



**УНИВЕРСИТЕТ  
ЛОБАЧЕВСКОГО**

Институт информационных технологий, математики и  
механики  
Кафедра математического обеспечения и  
суперкомпьютерных технологий

## Обработка изображений

# Источники биомедицинских изображений

Светлана Носова  
svetlana.nosova@itmm.unn.ru

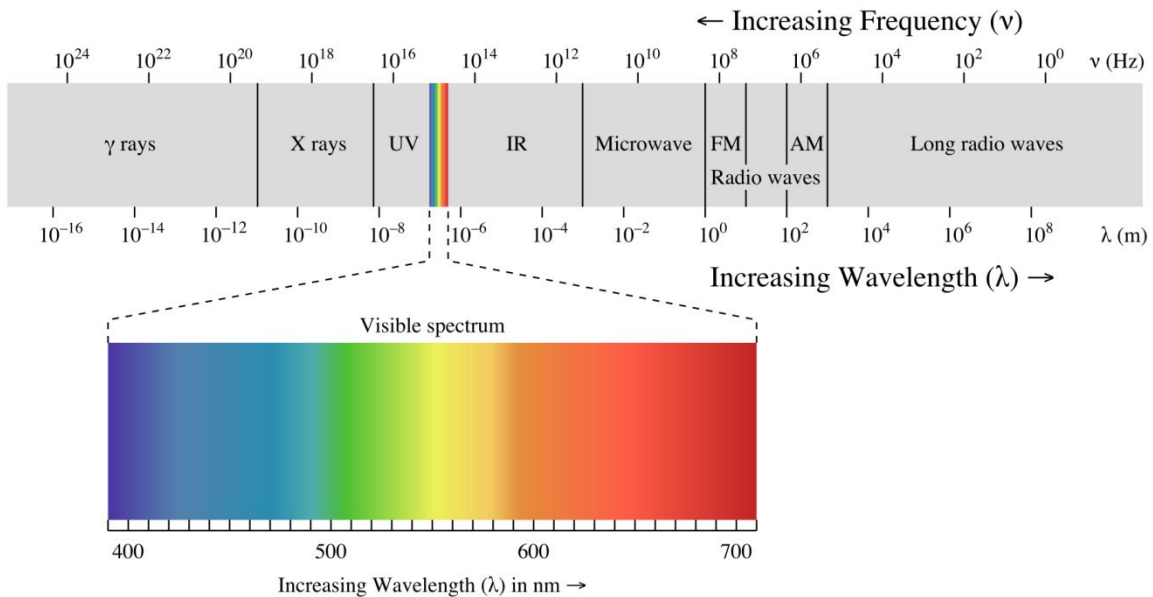
---

# Источники данных

---

# «Нормальное» разрешение





Особенность биомедицинских изображений:  
 всегда оттенки серого 😊

## Электромагнитны е волны

радиоволны (начиная со  
сверхдлинных),

терагерцевое излучение,

инфракрасное излучение,

ВИДИМЫЙ СВЕТ,

ультрафиолетовое  
излучение,

рентгеновское  
излучение и жёсткое  
(гамма-излучение) (см.  
ниже, см. также рисунок).



Доброкачественная

Злокачественная

Симметричная



Ассиметричная

Четко очерченная



Нечетко очерчен

Равно окрашенная



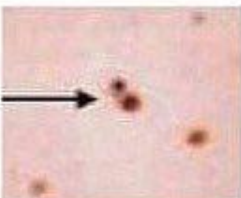
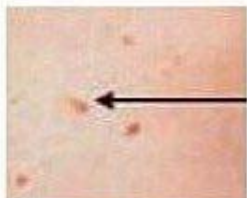
Неровно окрашен

Менее 6 мм



Более 6 мм

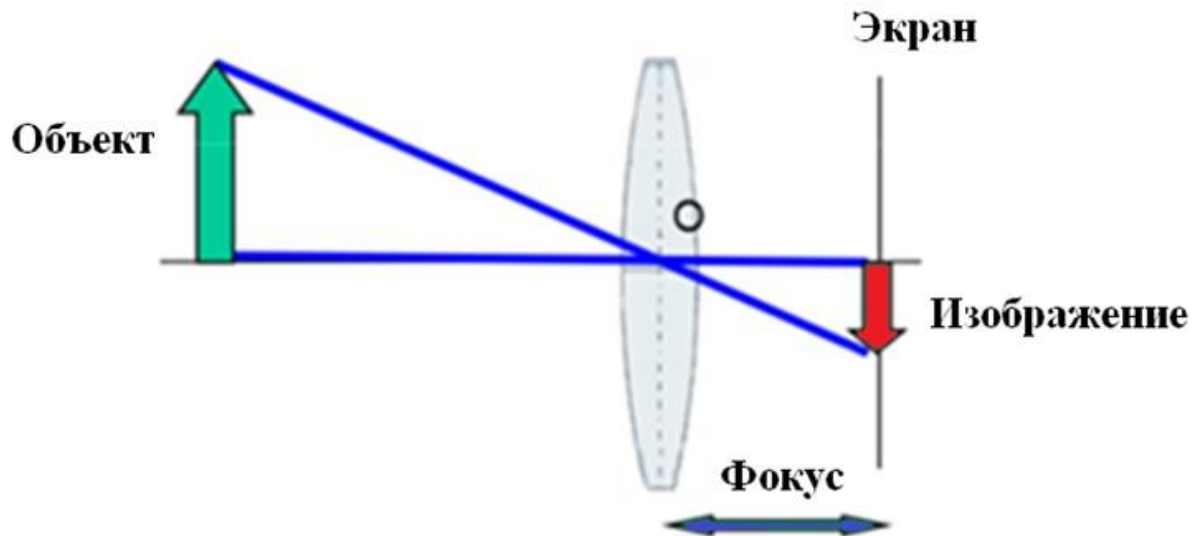
Обычная родинка



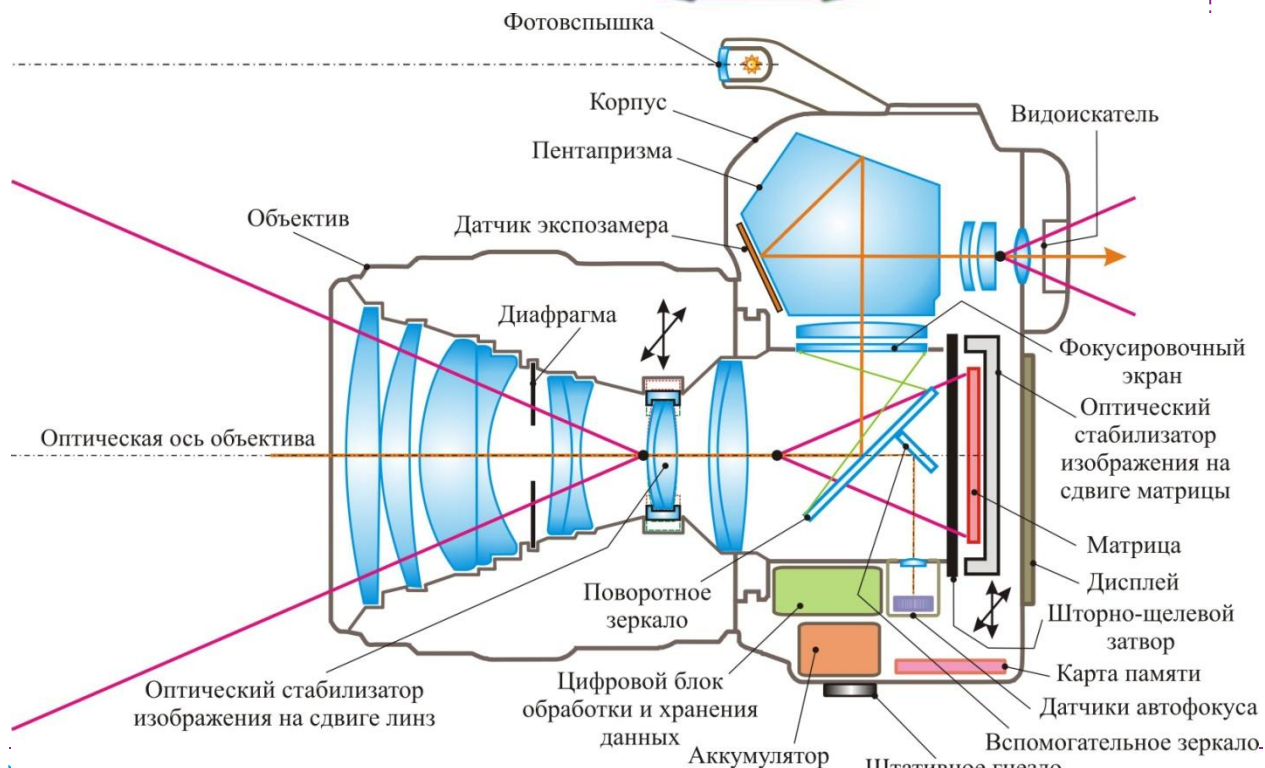
Растущая родинка

## Фотоаппарат

Фото меланом

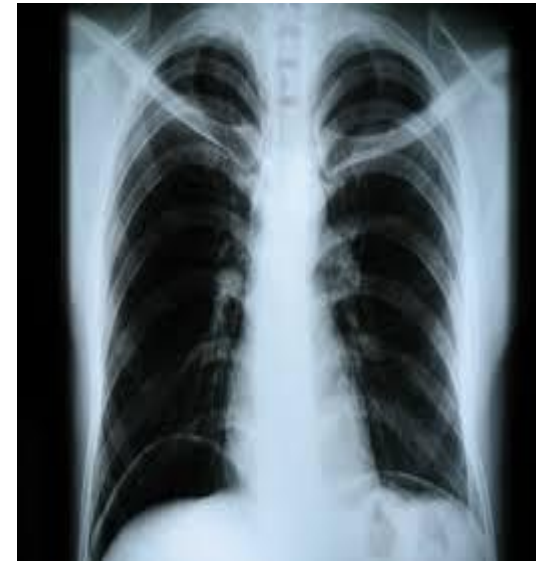


## Формирование изображения



- Особенности:
  - Низкая стоимость;
  - Высокое разрешение;
  - Небольшие дозы облучения.
- **Рентгеновское излучение** – электромагнитные волны, энергия фотонов которых лежит на шкале электромагнитных волн между ультрафиолетовым излучением и гамма-излучением (от  $-100 \text{ эВ}$  до  $-1 \text{ МэВ}$ ), (от  $-10$  до  $-10^{-3} \text{ нм}$ )<sup>[1]</sup>.
- В результате прохождения через образования разной плотности и состава пучок излучения рассеивается и тормозится, в связи с чем на плёнке формируется изображение разной степени интенсивности. В результате, на плёнке получается усреднённое, суммационное изображение всех тканей (тень).
- ▮ *Рентгеноскопия,*
  - Используется контраст.
  - В реальном времени.
  - Обратная связь.
- ▮ *Проекционные рентгенограммы ,*
  - Переломы.
  - Легкие, а также для выявления патологических изменений в легких.
  - С использованием контрастного вещества, такого как барий, они также могут быть использованы для визуализации структуры желудка и кишечника – это может помочь диагностировать язвы или некоторые виды рака толстой кишки .
- Компьютерная томография – аналогично, только 3D.
- В КТ рентгеновские лучи должны быть блокированы той или иной плотной тканью чтобы создать изображение, поэтому качество изображения мягких тканей будет плохим.

## Рентгенография



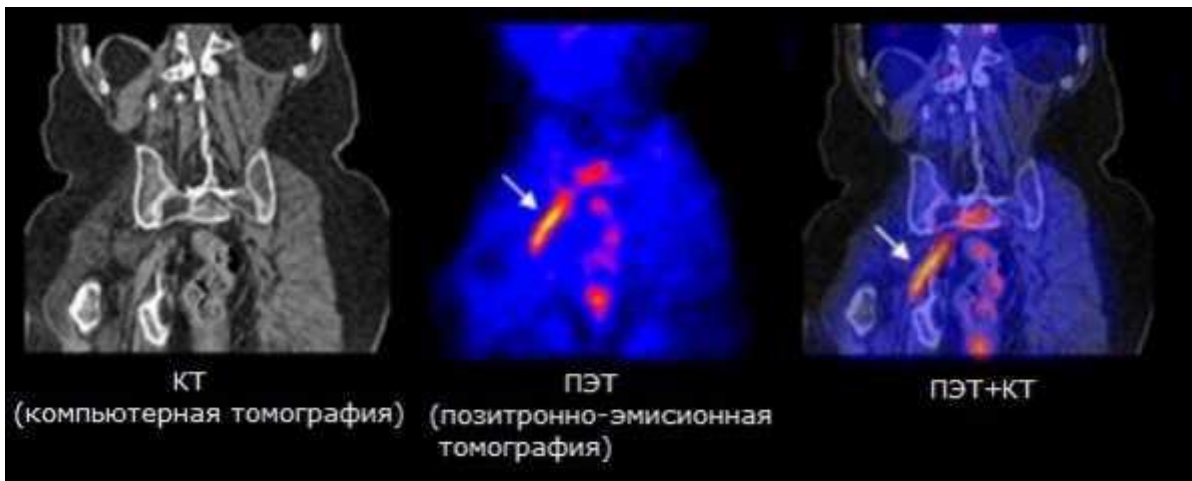
# Магнито-резонансная томография

- Машина МРТ излучает радиочастотный (РЧ) импульс на резонансной частоте атомов водорода молекул воды.
- Радиочастотные антенны ("РЧ-катушки") отправляют импульс в области тела, подлежащие рассмотрению.
- РЧ-импульс поглощается протонами, в результате чего их направление изменяется относительно основного магнитного поля.
- Когда радиочастотный импульс выключается, протоны "расслабляются" и обратно выравниваются первичным магнитом и излучают радиоволны в этом процессе.
- Это радиоизлучение от водородных атомов на воде обнаруживается и реконструируется в изображение.
- Резонансная частота вращающегося магнитного диполя, называется частотой Лармора и определяется силой основного магнитного поля и химической среды ядер, представляющих интерес.
- МРТ использует три вида электромагнитных полей:
  - очень сильные (как правило, от 1,5 до 3 **тесла**) статические магнитные поля для поляризации ядер водорода (первичное поле);
  - градиентные поля, которые могут изменяться в пространстве и времени (порядка 1 кГц) для пространственного кодирования, часто называют просто градиенты;
  - пространственно однородное радиочастотное поле для манипуляций с ядрами водорода для получения измеримых сигналов, собранных через РЧ антенны.
- МРТ традиционно создает двухмерное изображение тонкого "среза" тела и поэтому считается томографическим методом визуализации.
- Современные МРТ инструменты способны производить изображения в виде 3D-блоков, которые можно считать обобщением одного среза.
- МРТ не использует ионизирующее излучение и, следовательно, не представляет опасности для здоровья. (в отличие от рентгена и КТ)
- Опасность: нагревание и кардио-приборы.
- Отличная контрастность мягких тканей.**





- ▣ Молекулярная медицина
- ▣ Ядерная медицина использует определенные свойства изотопов и частиц, испускаемых от радиоактивного материала для диагностики и лечения различных патологий.
- ▣ Для проведения исследования пациенту вводят относительно короткоживущий изотоп, например,  $^{99m}\text{Tc}$ .
- ▣ Эти изотопы поглощаются преимущественно биологически активной тканью, и могут быть использованы для выявления опухолей или переломов в кости.
- ▣ Изображение получается после того как коллимированные фотоны регистрируются кристаллом, испускающим световой сигнал, который, в свою очередь, усиливается и преобразуется в данные для подсчета.
- ▣ Ядерная медицина делится на:
  - ▣ Сцинтиграфия – это форма диагностического теста, в котором радиоизотопы принимаются внутрь, например, внутривенно или перорально. Затем гамма-камеры захватывают сигналы от излучения, испускаемого препаратами, и образуют двумерные изображения.
  - ▣ ОФЭКТ – трехмерный томографический метод, который использует данные гамма-камеры со многих проекций и реконструирует их в разных плоскостях. Гамма-камера с двумя детекторами в сочетании с КТ-сканером, который обеспечивает локализацию функциональных данных ОФЭКТ, называется ОФЭКТ-КТ-камерой, и показывает полезность в продвижении области молекулярной визуализации.
  - ▣ *Позитронно-эмиссионная томография (ПЭТ)* использует распознавание совпадений для изображений функциональных процессов. Короткоживущий позитрон, излучаемый изотопом, такой как  $^{18}\text{F}$ , объединяется с органическим веществом, таким как глюкоза, создавая F18-фтордезоксиглюкозы, и может быть использован в качестве маркера метаболических реакций. Изображение распределения активности по всему телу может показать быстро растущую ткань, например, опухоли или инфекцию. ПЭТ-изображения можно просматривать вместе с изображениями компьютерной томографии чтобы определить анатомическую корреляцию.
  - ▣ Фидуциарные маркеры. Изображения одного и того же объекта, созданного с помощью двух разных систем обработки изображений, могут быть скоррелированы, если поместить фидуциарный маркер в область, отображаемую обеими системами.



КТ  
(компьютерная томография)      ПЭТ  
(позитронно-эмиссионная томография)      ПЭТ+КТ

Обработка изображений

## Ядерная медицина

25.12.2019

Высокочастотные звуковые волны, которые по-разному отражаются от ткани и образуют изображения.

Обычно это используется для визуализации плода у беременных женщин, визуализацию органов брюшной полости, сердца, груди, мышц, сухожилий, артерий и вен.

УЗИ может обеспечить меньшую анатомическую детализацию, чем такие методы, такие как КТ или МРТ

Измерения в реальном времени!!!

. Ультразвук отличается от других методов медицинской визуализации тем, что он является звуковой волной высокой частоты, которая посылается в ткани, и, в зависимости от состава различных тканей, сигнал будет ослаблен и вернется через разные промежутки времени.

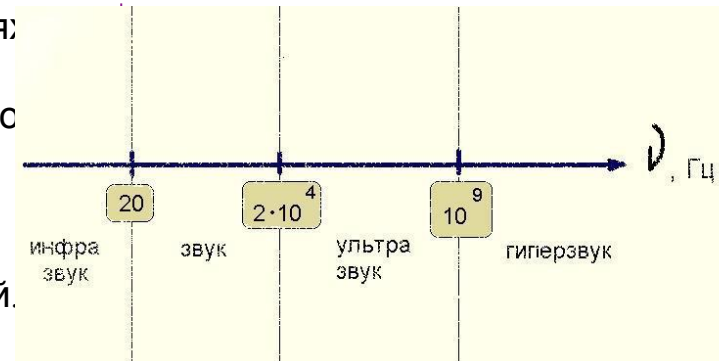
Путь отраженных звуковых волн в многослойной структуре может быть определен с помощью входного акустического импеданса и коэффициента отражения и пропускания относительных структур.

Современные сканеры позволяют показать кровоток в артериях и венах.

**Ультразвук** – упругие колебания в среде с частотой за предел слышимости человека. Обычно под **ультразвуком** понимают частоты выше 20 000 Герц.

В случае, когда ультразвук используется для получения изображения сердца, процесс называется эхордией.

## Ультразвук



# Другие

---

## □ Диффузная оптическая томография

□ Диффузная оптическая томография (англ.)русск. (ДОТ) — способ медицинской визуализации, использующий инфракрасное излучение для изображения тела человека. Технология измеряет оптическую абсорбцию гемоглобина и опирается на его спектр поглощения в зависимости от насыщения кислородом.

## □ Магнитоэнцефалография

□ Магнитоэнцефалография (МЭГ) — нейровизуализационная технология, используемая для измерения магнитных полей, которую производит электрическая активность головного мозга посредством особо чувствительных устройств, таких как СКВИД. МЭГ использует непосредственное измерение электроактивности нейронов, более точное, чем например ФМРТ, с очень высоким разрешением во времени, но маленьким в пространстве. Преимущество измерения таких магнитных полей в том, что они не искажаются окружающей тканью, в отличие от электрических полей, измеряемых ЭЭГ.

---

# Увеличение



# Микроскопы

## Оптические микроскопы

- Человеческий [глаз](#) представляет собой естественную оптическую систему, характеризующуюся определённым разрешением, то есть наименьшим расстоянием между элементами наблюдаемого объекта (воспринимаемыми как точки или линии), при котором они ещё могут быть отличны один от другого. Для нормального глаза при удалении от объекта на т. н. расстояние наилучшего видения ( $D = 250$  мм), среднестатистическое нормальное разрешение составляет  $\sim 0,2$  мм. Размеры микроорганизмов, большинства растительных и животных клеток, мелких [кристаллов](#), деталей микроструктуры [металлов](#) и сплавов и т. п. значительно меньше этой величины.
- До середины XX века работали только с видимым оптическим излучением, в диапазоне  $400\text{--}700$  [нм](#), а также с ближним ультрафиолетом ([люминесцентный микроскоп](#)). Оптические микроскопы не могли давать разрешающей способности менее полупериода волны опорного излучения (диапазон длин волн  $0,2\text{--}0,7$  [мкм](#), или  $200\text{--}700$  нм). Таким образом, оптический микроскоп способен различать структуры с расстоянием между точками до  $\sim 0,20$  мкм, поэтому максимальное увеличение, которого можно было добиться, составляло  $\sim 2000$  крат.

## Электронные микроскопы

- Пучок [электронов](#), которые обладают свойствами не только частицы, но и волны, может быть использован в микроскопии.
- Длина волны электрона зависит от его энергии, а энергия электрона равна  $E = Ve$ , где  $V$  — разность потенциалов, проходимая электроном,  $e$  — заряд электрона. Длины волн электронов при прохождении разности потенциалов  $200\ 000$  В составляет порядка  $0,1$  нм. Электроны легко фокусировать электромагнитными линзами, так как электрон — заряженная частица. Электронное изображение может быть легко переведено в видимое.
- Разрешающая способность электронного микроскопа в  $1000\text{--}10000$  раз превосходит разрешение традиционного светового микроскопа и для лучших современных приборов может быть меньше одного ангстрема.

## Сканирующие зондовые микроскопы

- Класс микроскопов, основанных на сканировании поверхности зондом.
- Сканирующие зондовые микроскопы (СЗМ) — относительно новый класс микроскопов. На СЗМ изображение получают путём регистрации взаимодействий между зондом и поверхностью. На данном этапе развития возможно регистрировать взаимодействие зонда с отдельными атомами и молекулами, благодаря чему СЗМ по разрешающей способности сопоставимы с электронными микроскопами, а по некоторым параметрам превосходят их.

## Рентгеновские микроскопы

- Рентгеновский микроскоп** — устройство для исследования очень малых объектов, размеры которых сопоставимы с длиной рентгеновской волны. Основан на использовании электромагнитного излучения с длиной волны от  $0,01$  до  $1$  нанометра.
- Рентгеновские микроскопы по разрешающей способности находятся между электронными и оптическими микроскопами. Теоретическая разрешающая способность рентгеновского микроскопа достигает  $2\text{--}20$  [нанометров](#), что на порядок больше разрешающей способности [оптического микроскопа](#) (до  $150$  [нанометров](#)). В настоящее время существуют рентгеновские микроскопы с разрешающей способностью около  $5$  нанометров<sup>[5]</sup>.
- Галерея оптических микроскопов

---

# Коллекции данных

(включая материалы <https://github.com/beamandrew/medical-data>)

# База данных «OmniMedicalSearch».

---

- Различные тематические источники данных:
  - интерактивный анатомический атлас;
  - разнообразные коллекции медицинских изображений;
  - результаты дерматологических исследований;
  - библиотека эндоскопических видеороликов.
- Доступ: [http://www.omnimedicalsearch.com/image\\_databases.html](http://www.omnimedicalsearch.com/image_databases.html)

# Национальная медицинская библиотека MedPix

---

- The National Library of Medicine MedPix.
- База данных содержит 53 тыс. медицинских изображений 13 тыс. пациентов с аннотациями. Требуется регистрация.  
Ссылка: <https://medpix.nlm.nih.gov/home>

# База данных рентгенологических исследований MURA

---

- **musculoskeletal radiographs.**
- Набор данных скелетно-мышечных рентгенограмм, состоящих из 14 863 исследований от 12173 пациентов, в общей сложности 40 561 многократных рентгенографических изображений. Каждое из них относится к одному из 7 стандартных типов рентгенологического исследования верхней конечности: локоть, палец, предплечье, рука, плечевая кость, плечо и запястье. Каждое исследование было вручную помечено как нормальное или ненормальное со стороны сертифицированных по радиологии специалистов из Стэнфордской больницы в период с 2001 по 2012 год. Описание:
- <https://stanfordmlgroup.github.io/competitions/mura/>
-

# Система обмена изображениями мозга при аутизме ABIDE

---

- (The Autism Brain Imaging Data Exchange).
- Содержит МРТ-изображений 539 лиц, страдающих от аутизма и 573 типичных элементов управления. Эти 1112 наборов данных состоят из структурных и функциональных данных МРТ, а также обширного массива фенотипической информации. Требуется регистрация.  
Описание: <http://www.ncbi.nlm.nih.gov/pubmed/23774715>. Предварительно обработанная версия: <http://preprocessed-connectomes-project.org/abide/>

# Инициатива по накоплению МРТ-изображений в болезни Альцгеймера

---

- (Alzheimer's Disease Neuroimaging Initiative – ADNI).
- База данных МРТ пациентов с болезнью Альцгеймера. Включает клинические, геномные и биомаркерные данные. Требуется регистрация.  
Описание: <http://www.neurology.org/content/74/3/201.short>.  
Доступ: <http://adni.loni.usc.edu/data-samples/access-data/>

# Цифровые изображения сетчатки глаза в экстракции сосудов

---

- (Digital Retinal Images for Vessel Extraction – DRIVE).
- База данных DRIVE предназначена для сравнительных исследований по сегментации кровеносных сосудов в изображениях сетчатки. Она состоит из фотографий, на которых показаны признаки легкой ранней диабетической ретинопатии.  
Описание: <http://www.isi.uu.nl/Research/Publications/publicationview/id=855.html>.  
Доступ: <http://www.isi.uu.nl/Research/Databases/DRIVE/download.php>



# Открытая библиотека серий изображений МРТ

---

- (The Open Access Series of Imaging Studies – OASIS).
  - Включает 2 набора данных: поперечное сечение и продольный набор.
- Доступ: <http://www.oasis-brains.org/>

# Консенсусные данные SCMR

---

- (SCMR Consensus Data).
- Набор данных 15 исследований сердечно-сосудистых заболеваний смешанных патологий (5 здоровых, 6 инфарктов миокарда, 2 сердечной недостаточности и 2 гипертрофии), которые были получены у разных МР-аппаратов (4 GE, 5 Siemens, 6 Philips).

Доступ: <http://www.cardiacatlas.org/studies/>

# Консорциум базы данных изображений легких

---

- (Lung Image Database Consortium - LIDC)
- Спиральное КТ-сканирование легких может улучшить раннее выявление рака легких у лиц с высоким риском.
- Алгоритмы обработки изображений могут помочь в обнаружении повреждений на спиральных КТ-исследованиях и оценить стабильность или изменение размера поражения при серийных исследованиях КТ. Использование таких компьютерных алгоритмов могло бы значительно повысить чувствительность и специфичность скрининга спирального КТ легких, а также снизить затраты за счет сокращения времени врача, необходимого для интерпретации.

Доступ: <http://imaging.cancer.gov/programsandresources/informationsystems/lidc>

# Набор цифровых снимков грудной клетки

---

- (NIH Chest X-ray Dataset of 14 Common Thorax Disease Categories).
- Набор данных содержит 112 тыс. изображений 30 тыс. уникальных пациентов с фронтальным обзором и примерами 14 торакальных патологий.  
Доступ: <http://academictorrents.com/details/557481faacd824c83fbf57dcf7b6da9383b3235a>

# Коллекции TCIA

---

- (The Cancer Imaging Archive (TCIA) Collections).
- Содержит данные по различным типам рака (например, карцинома, рак легких, миелома) и различные методы визуализации. Данные TCIA организованы в целенаправленные коллекции предметов. У субъектов обычно есть тип рака и / или анатомическая карта (легкие, мозг и т. д.). Каждая ссылка в приведенной таблице содержит информацию о научной ценности коллекции, информацию о том, как получить любые поддерживающие данные без изображения, которые могут быть доступны, и ссылки для просмотра или загрузки данных изображения. Для поддержки воспроизводимости научных исследований TCIA поддерживает идентификаторы цифровых объектов (DOI), которые позволяют пользователям делиться подмножествами данных TCIA, на которые ссылается исследование.  
Доступ: <http://www.cancerimagingarchive.net/>

# Белорусский туберкулезный портал

---

- (Belarus tuberculosis portal).
- Туберкулез является серьезной проблемой системы здравоохранения Беларуси. Многие тяжелые формы распространены по всей стране и наблюдаются в различных противотуберкулезных диспансерах. Ожидается, что способность ведущих белорусских специалистов по туберкулезу следить за такими пациентами может быть значительно улучшена за счет использования общей базы данных, содержащей радиологические изображения пациентов, лабораторные работы и клинические данные. Это также значительно улучшит соблюдение стандартов лечения и приведет к лучшим результатам лечения. В наборе данных есть рентгенографии грудной клетки и компьютерная томография одного и того же пациента. Доступ: <http://tuberculosis.by/>

# Цифровая база данных скрининга маммографии

- (Digital Database for Screening Mammography – DDSM).
- Этот ресурс создан сообществом исследований маммографических изображений. Первичная поддержка этого проекта была обеспечена грантом Программы исследований рака молочной железы (Breast Cancer Research Program) в США. Проект DDSM – это совместные усилия, связанные с исследованиями в больнице Массачусетса (D. Kopans, R. Moore), Университете Южной Флориды (K. Bowyer) и Sandia National Laboratories (P. Kegelmeyer). Дополнительные случаи включены Школой медицины Вашингтонского университета. В число дополнительных сотрудничающих учреждений входят Школа медицины Университета Уэйк Форест (Отделения медицинской инженерии и радиологии), Госпиталь Святого Сердца. Основная цель базы данных – облегчить обоснованные исследования в разработке компьютерных алгоритмов, помогающих в скрининге. Еще одной целью ведения базы является поддержка в разработке алгоритмов помощи в диагностике и развитии учебных пособий. База данных содержит около 2500 исследований. Каждое исследование включает в себя два изображения каждой груди, а также некоторую связанную с этим информацию о пациенте (возраст во время исследования, рейтинг плотности сердечной деятельности ACR и т.д.) и информацию об изображении (сканер, пространственное разрешение и т.д.). Изображения, содержащие подозрительные области, имеют связанную с «пикселем» информацию о местах и типах подозрительных областей. Также предоставляются программное обеспечение как для доступа к изображениям маммограммы, а также для расчета показателей эффективности для автоматизированных алгоритмов анализа изображений.  
Доступ: <http://marathon.csee.usf.edu/Mammography/Database.html>



# База данных МРТ-изображений рака предстательной железы.

---

- Магнитно-резонансная томография (МРТ) обеспечивает методы визуализации, позволяющие диагностировать и локализовать рак предстательной железы. Предоставляет многопараметрический набор данных МРТ, предназначенный для помощи в разработке систем автоматизированного обнаружения и диагностики этих заболеваний.

Доступ: <http://i2cvb.github.io/>



# База данных «Сегментация в рентгенограммах грудной клетки»

---

- (Segmentation in Chest Radiographs – SCR).  
Автоматическая сегментация анатомических структур в рентгенограммах грудной клетки имеет большое значение для компьютерной диагностики. База данных SCR была создана для облегчения сравнительных исследований по сегментации легочных полей, сердца и ключиц в стандартных рентгенограммах грудной клетки.  
Доступ: <http://www.isi.uu.nl/Research/Databases/SCR/>

# Публичные базы данных VIA Group

---

- (VIA Group Public Databases).
- Включает документированные базы данных изображений, пригодные для разработки инструментов количественного анализа изображений, особенно в системах поддержки принятия врачебных решений (ССПВР). Создана в сотрудничестве с группой I-ELCAP. Содержат легкие КТ-изображения в формате DICOM вместе с описанными радиологами аномалиями.  
Доступ: <http://www.via.cornell.edu/databases/>

# База данных изображений USC-SIPI

---

- (The USC-SIPI Image Database).
- Представляет собой набор оцифрованных изображений. Предназначен в основном для поддержки исследований в области обработки изображений, анализа и машинного зрения. Первое издание базы данных изображений USC-SIPI было распространено в 1977 году, и с тех пор было добавлено много новых изображений. База данных разделена на тома, основанные на основном характере изображений. Изображения в каждом томе имеют различные размеры, такие как 256x256 пикселей, 512x512 пикселей или 1024x1024 пикселей. Все изображения имеют 8 бит / пиксель для черно-белых изображений, 24 бит / пиксель для цветных изображений. Доступ: <http://sipi.usc.edu/database/>

# Визуальная концепция извлечения данных в радиологии

---

- (Visual Concept Extraction Challenge in Radiology).
- Включает вручную аннотированные радиологические данные нескольких анатомических структур (например, почек, легких, мочевого пузыря и т. Д.) Из нескольких различных методов визуализации (например, КТ и МР). Можно использовать для разработки и оценки моделей по сравнению с эталонами.  
Доступ: <http://www.visceral.eu/>

# Изображения диабетической ретинопатии

---

- (diabetic retinopathy).
- База данных включает изображения сетчатки высокого разрешения, которые клиницисты аннотируют по шкале тяжести 0–4, для выявления диабетической ретинопатии. Этот набор данных является частью завершившегося конкурса Kaggle, который, как правило, является отличным источником общедоступных наборов данных. Доступ: <https://www.kaggle.com/c/diabetic-retinopathy-detection>

# Скрининг рака шейки матки

---

- (Cervical Cancer Screening).
- Еще один источник данных конкурсов kaggle, на этот раз использовался при разработке алгоритмов для правильной классификации типов шейки матки на основе соответствующих изображений. Набор данных включает различные типы шейки матки, которые считаются нормальными (не раковыми), но поскольку зоны трансформации не всегда видны, некоторые пациенты нуждаются в дополнительном тестировании, а некоторые нет.

Доступ: <https://www.kaggle.com/c/intel-mobileodt-cervical-cancer-screening/data>

# База данных сегментации мозговых опухолей

---

- (Multimodal Brain Tumor Segmentation Challenge).
- Включает большой набор данных магнитно-резонансного сканирования опухолей головного мозга. Авторы постоянно расширяют этот набор данных с 2012 года.

Доступ: <http://braintumorsegmentation.org/>

# DDSM: Цифровая база данных для скрининговой маммографии

---

- База данных содержит около 2500 исследований.
- Каждое исследование включает в себя два изображения каждой молочной железы, а также некоторую связанную информацию о пациенте (возраст на момент исследования, рейтинг плотности груди ACR, рейтинг тонкости для отклонений, описание отклонений от нормы для ключевого слова ACR) и информацию об изображении (сканер, пространственное разрешение, ... ).
- Изображения, содержащие подозрительные области, имеют связанную с пиксельной о расположении и типах подозрительных областей.
- Доступ: <http://marathon.csee.usf.edu/Mammography/Database.html>



# INbreast: Database for Digital Mammography

---

- INbreast включает 115 случаев (410 изображений).
- Несколько типов поражений (массы, кальцификации, асимметрии и искажения) включены.
- Точные контуры поражений в формате XML.
- Access: [http://medicalresearch.inescporto.pt/breastresearch/index.php/Get\\_INbreast\\_Database](http://medicalresearch.inescporto.pt/breastresearch/index.php/Get_INbreast_Database)
-

## mini-MIAS: MIAS MiniMammographic Database

---

- Общество анализа маммографических изображений (MIAS) .
- База данных содержит 322 оцифрованных фильма и доступна на ленте 2,3 ГБ 8 мм (ExaByte).
- Включает в себя маркировку местлюбых отклонений, которые могут присутствовать.
- Изображения - 1024x1024.

# Предстательная железа

---

- По сообщениям, во всем мире рак простаты (CaP) является вторым наиболее часто диагностируемым раком мужчин, составляя 13,6% (Ferlay et al. (2010)).
- Согласно статистическим данным, в 2008 году число новых диагностированных случаев составило 899 000 с не менее чем 258 100 случаев смерти (Ferlay et al. (2010)).
- Магнитно-резонансная томография (МРТ) обеспечивает методы визуализации, позволяющие диагностировать и локализовать CaP. I2CVB предоставляет многопараметрический набор данных МРТ, (CAD).
- Доступ: <http://i2cvb.github.io/>

# Broad Bioimage Benchmark Collection

---

- Коллекция изображений микроскопии.
- Описаны различные биологические применения.
- Источник: <https://data.broadinstitute.org/bbbc/>

---

# Конкурсы

# Данные конкурсов

---

- <http://www.visceral.eu/> Visual Concept Extraction Challenge in Radiology
- <http://www.grand-challenge.org/> Grand Challenges in Biomedical Image Analysis
- <http://dreamchallenges.org/> Dream Challenges
- <https://www.kaggle.com/c/diabetic-retinopathy-detection> Kaggle diabetic retinopathy
- <https://www.kaggle.com/c/intel-mobileodt-cervical-cancer-screening/data> Cervical Cancer Screening
- <http://www.ia.unc.edu/MSseg/> Multiple sclerosis lesion segmentation
- <http://braintumorsegmentation.org/> Multimodal Brain Tumor Segmentation Challenge
- <http://coding4cancer.org/> Coding4Cancer
- Challenges track in MICCAI Conference
- International Symposium on Biomedical Imaging (ISBI)
- <https://continuousregistration.grand-challenge.org/home/> Continuous Registration Challenge (CRC)

---

# Источники/литература

# Источники/литература

---

- РМС Open Access Subset  
Полный текст
- Открытый доступ  
Information: <http://www.ncbi.nlm.nih.gov/pmc/tools/openftlist/>  
Archived  
files: [http://www.ncbi.nlm.nih.gov/pmc/tools/ftp/#Data\\_Mining](http://www.ncbi.nlm.nih.gov/pmc/tools/ftp/#Data_Mining)
- <https://www.ncbi.nlm.nih.gov/research/bionlp/APIs/BioC-PubMed/>
- <https://www.ncbi.nlm.nih.gov/research/bionlp/APIs/BioC-PMC/>





**УНИВЕРСИТЕТ  
ЛОБАЧЕВСКОГО**

Институт информационных технологий, математики и  
механики  
Кафедра математического обеспечения и  
суперкомпьютерных технологий

**Обработка изображений**

# Шумоподавление

Светлана Носова  
svetlana.nosova@itmm.unn.ru

# Рассматриваемые вопросы

---

- Цели удаления шума
- Формирование изображений
- Источники и типы шума
- Моделирование шума
- Модели шума
- Алгоритмы удаления шума
- Оценка качества алгоритмов удаления шума

# Цели удаления шума

---

- Улучшение визуального восприятия
- Специальные применения:
  - Увеличение резкости изображений на рентгеновских снимках;
  - Предобработка для последующего распознавания объектов;
  - Для сжатия изображений (удалить шумовые пиксели изображения как неважные детали).
- Основная цель: удалить информацию, мешающую восприятию и обработке (человеком или алгоритмом). Необходимую – оставить.

---

# Оценки качества алгоритмов шумоподавления

# Метрики

---

- Среднеквадратичная ошибка

$$MSE = \frac{1}{N} \sum_{i=1}^N (x_i - y_i)^2$$

- Пиковое отношение сигнал/шум

$$PSNR_{dB} = 10 \lg \frac{M^2}{MSE}$$

$M$  – максимальное значение пикселя

# Метрики

---

$$NQM = 10 \log_{10} \left( \frac{\sum_i \sum_j f^2(i, j)}{(\sum_i \sum_j (f(i, j) - g(i, j))^2)} \right)$$

Общий индекс качества  
искаженного  
изображения  
 $i, j$  – средняя  
ИНТЕНСИВНОСТЬ

$$Q = \frac{4\sigma_{ij}\bar{i}\bar{j}}{(\sigma_i^2 + \sigma_j^2)[\bar{i}^2 + \bar{j}^2]}$$

# SSIM

□ Интенсивность

$$\mu_x = \frac{1}{N} \sum_{n=1}^N x_n,$$

□ Контрастность

$$\sigma_x = \sqrt{\frac{1}{N-1} \sum_{n=1}^N (x_n - \mu_x)^2},$$

□ Структура

$$\zeta_x = \frac{x - \mu_x}{\sigma_x}.$$

$$SSIM(x, y) = l(x, y)^\alpha \cdot c(x, y)^\beta \cdot s(x, y)^\gamma,$$

$$l(x, y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1}$$

$$c(x, y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2}$$

$$s(x, y) = \frac{\langle \zeta_x, \zeta_y \rangle + C_3}{\sigma_x\sigma_y + C_3} = \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3}$$

---

# Источники и типы шумов



# Источники шума

---

- ❑ Ошибки оборудования
- ❑ Плохие условия съёмки
  - ❑ Пример: ночная съёмка
- ❑ помехи при передаче сигнала по аналоговым каналам: наводки от источников электромагнитных полей, собственные шумы активных компонентов (усилителей) линии передачи;
- ❑ Дефекты (примеси и др.) потенциального барьера вызывают утечку заряда сгенерированного за время экспозиции — т. н. чёрный дефект. Такие дефекты видны на светлом фоне в виде тёмных точек

# Шум считывания

---



**Каждый раз**, когда вы считываете изображение с камеры, в изображение включается некоторый шум.

**Даже если нет никакого сигнала** (никакого света, никакого заметного темнового тока), у вас все равно **будет шум в изображении**, оно всё равно не будет идеальным.

Этот шум называют шумом считывания. Его генерирует и электроника устройства в целом, и непосредственно электроника сенсора.



# Дробовой шум

---



Если съемка данных производится в течение секунды, может оказаться, что какой-то пиксель в сенсоре поразили 10 фотонов, а до другого доберутся 12, до третьего – 9.

- Такие отклонения всегда возникают на световом пути. Их и называют **дробовым шумом** (другое название – *фотонный шум*), который подчиняется распределению Пуассона.

Избежать дробового шума невозможно, при этом его величина растет как квадратный корень из интенсивности. Т.е. чем более яркий объект вы снимаете, тем больше получаете шума.



# Темновой шум

---

- Закройте объектив камеры крышкой и сделайте несколько снимков с разной выдержкой. Вы заметите, что изображение тем ярче, чем больше выдержка. Это из-за темнового тока.
- *Темновой ток – малый электрический ток, который протекает через фоточувствительный детектор, например, фотодиод, фотозлектронный умножитель, полупроводниковый детектор гамма-квантов и др. при отсутствии поглощенных фотонов.*
- Интенсивность удваивается, когда вы удваиваете длительность экспозиции, и, кроме того, почти удваивается каждые 6 градусов по шкале Цельсия.
- Некоторые пиксели светлеют быстрее других («горячие» пиксели светлеют очень быстро), в результате чего появляется узор стабильного пространственного шума.

# Ошибка квантования

---

- Когда мы считываем напряжение с сенсора, это аналоговый сигнал. Чтобы превратить его в цифровой, используется аналогово-цифровой преобразователь (АЦП).
- Предположим, у нас есть 8-битный АЦП.
- Установив его, мы можем записать 256 градаций интенсивности ( $2^8$  в 8-й степени). Предположим далее, что наша ПЗС-матрица может накопить до насыщения 10 000 электронов.
- Если мы хотим использовать весь динамический диапазон ПЗС-матрицы, то должны установить АЦП так, чтобы каждое деление шкалы соответствовало примерно 25 фотонам ( $10\,000/256$ ). Так, 10 на АЦП означает, что было захвачено примерно 250 фотонов, а 11 – около 275.

Не нужно много усилий, чтобы заметить здесь некоторую проблему. Для нас теперь нет разницы между 250 фотонами и 251, 255 или 260 фотонами. Им всем соответствует одно и то же значение. Эту проблему называют ошибкой квантования

# Что влияет на величину шума

---

- Повышение светочувствительности элемента
- Повышение длины экспозиции
- Повышение температуры
  
- Ведет к увеличению шума

# Классификация шума

---

- Случайный
  - Колебания интенсивности выше и ниже реальных
  - Сильно зависит от числа ISO
  - Проявляется при любой экспозиции
  - Рисунок меняется даже при одинаковых настройках камеры
- Структурный
  - Проявляются «горячие пиксели»
  - Выдержка большой длины, высокие температуры вызывают усиление
  - Значительно превосходят колебания, вызванные случайным шумом
  - Идентичное распространение при съемке в идентичных условиях
- Линеичатый
  - Зависит от камеры, вносит сама камера в процессе считывания сигнала с цифрового сенсора
  - Высокие ISO, в тенях, когда изображение было излишне осветлено

# Некоторые особенности

---

- Чувствительность глаза к деталям цветовой информации (не к яркостной)
- Шум сильная зашумленность в цветовой компоненте (не в яркостной)
- Выход: работать в HSV,  $L^*a^*b$  (а не в RGB)
- Величина шума – от одного до нескольких пикселей



---

# Моделирование шума



# Моделирование шума

---

$$f(i, j) = s(i, j) + n(i, j)$$

$f(i, j)$  – результирующее изображение

$s(i, j)$  – идеальное изображение сцены

$n(i, j)$  - шум

# Метод Монте-Карло

- Положения «шумовых пикселей» искать при помощи генератора псевдослучайных чисел (например, 5% от общего количества пикселей)
- Имеем генератор псевдослучайных чисел с равномерным законом распределения на интервале (0,1) (A)
- Надо получить новую случайную величину  $N$  с распределением  $P$ 
$$\begin{bmatrix} n_1 n_2 \dots n_N \\ p_1 p_2 \dots p_N \end{bmatrix}$$
- интервал значений (0,1) разбивают на  $N$  интервалов, длины которых соответственно равны  $[p_1 p_2 \dots p_N]$
- Получаем значение случайной величины  $a$
- Если это значение попадает в интервал  $i$ , наступило событие  $n_i$
- Есть еще метод обратных функций, но мы его рассматривать не будем)

---

# Модели шума

# Типы шума

---

- Яркостной шум
  - Отличие в оттенке цвета
  
- Хроматический шум
  - Отличие по цвету

# Шум типа «Соль перец» (импульсный)

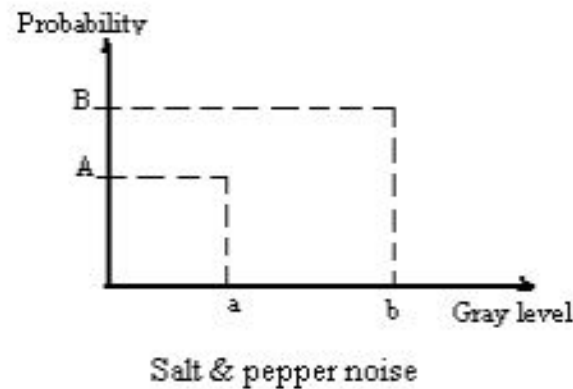


Fig. 10.1 Probability density function for the salt & pepper noise model.

$$PDF_{salt \& \ pepper} = \begin{cases} A & \text{for } g = a \text{ ("pepper")} \\ B & \text{for } g = b \text{ ("salt")} \end{cases}$$

# Шум Гаусса

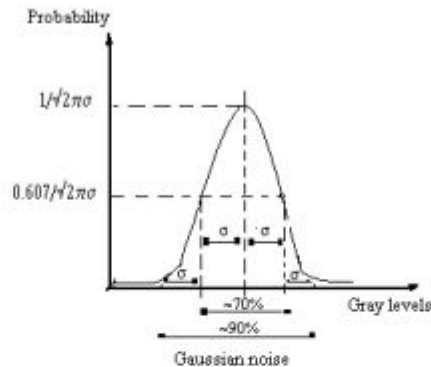


Fig. 10.2 Probability density function for the Gaussian noise model

$$PDF_{Gaussian} = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(g-\mu)^2}{2\sigma^2}} \quad (10.3)$$

where:

$g$  = gray level;

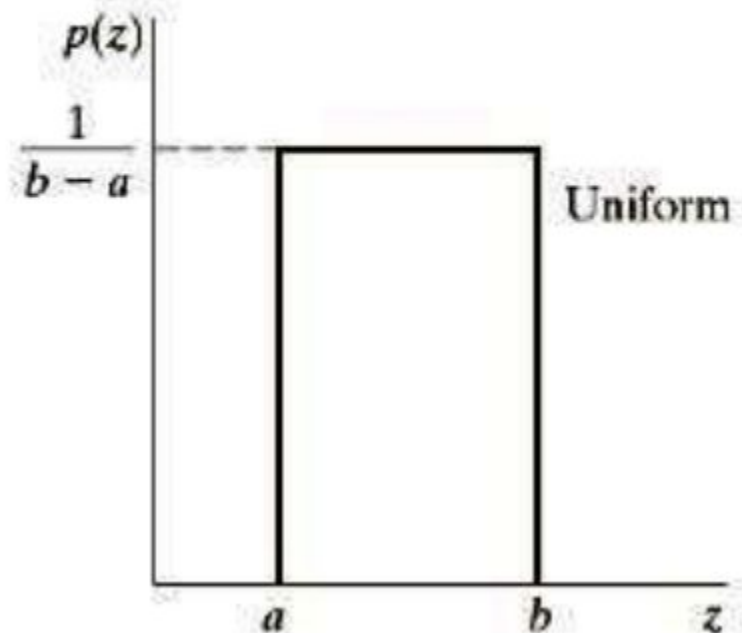
$\mu$  = mean;

$\sigma$  = standard deviation;

- ( в прикладных областях называется электронным шумом
- возникает в усилителях или детекторах.

# Постоянный шум

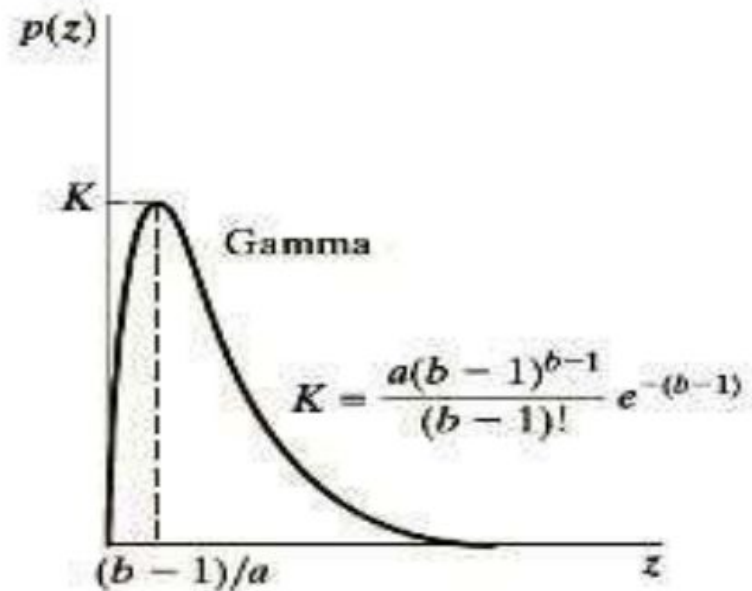
$$p(z) = \begin{cases} \frac{1}{b-a} & \text{if } a \leq z \leq b \\ 0 & \text{otherwise} \end{cases} \quad [5]$$





# Гамма-шум (Шум Эрланга)

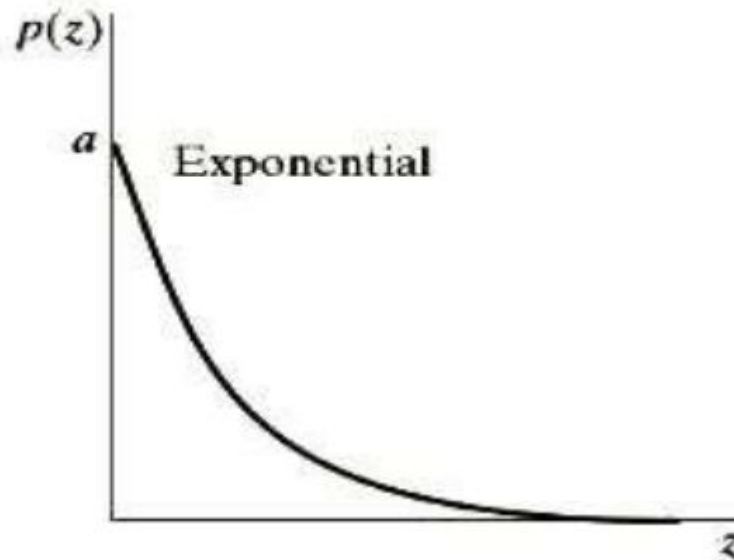
$$p(z) = \begin{cases} \frac{a^b z^{b-1}}{(b-1)!} e^{-az}, & \text{for } z \geq 0 \\ 0, & \text{otherwise} \end{cases}$$



Возникает из-за несовершенства электронных схем и датчика

# Экспоненциальный шум

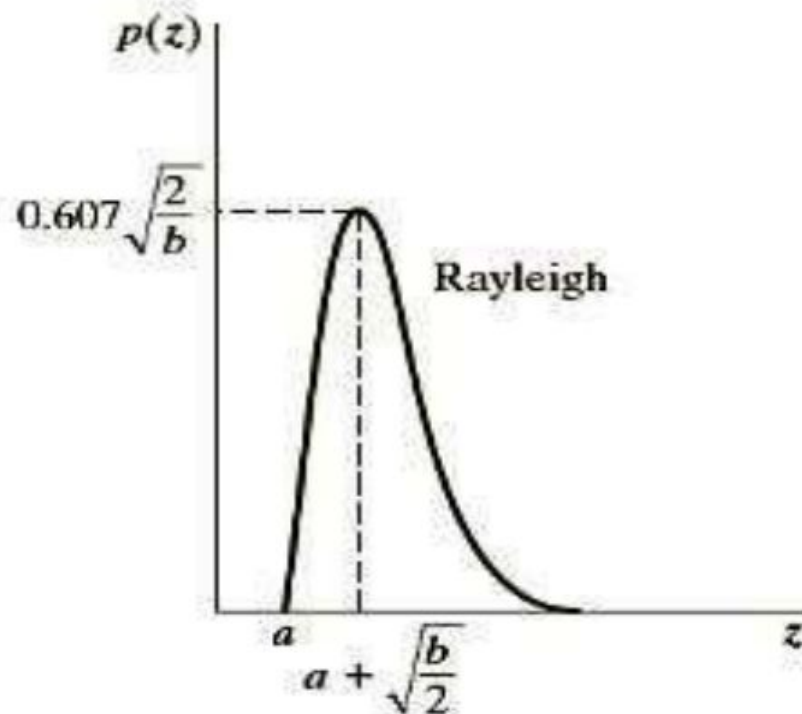
$$p(z) = \begin{cases} a e^{-az}, & z \geq 0 \\ 0, & z < 0 \end{cases} \quad [5]$$



Гамма-шум при  $b=1$

# Шум Райли

$$p(z) = \begin{cases} \frac{2}{b} (z - a) e^{-(z-a)^2/b}, & \text{for } z \geq a \\ 0, & \text{for } z < a \end{cases} \quad [5]$$



# Другие шумы

---

- Периодический шум
  - Генерируется во время «захвата» изображения
  - Параметры описываются при помощи преобразования Фурье
  - Удаление происходит при помощи анализа и изменения спектра Фурье
- Квантовый(фотонный) шум.
  - Возникает на рентгеновских снимках

# Оценка распределения

---

- Для каждого сенсора рассчитать свою функцию распределения, сделав снимок поверхности с постоянным значением цветовой компоненты.
- Для периодического шума – оценка спектра Фурье.

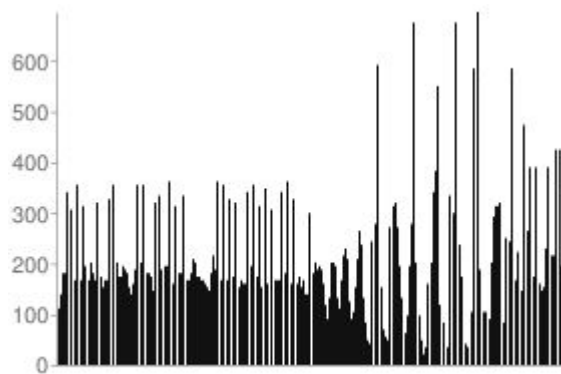
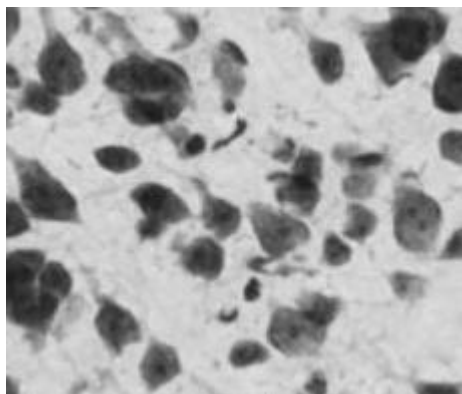
---

# Методы удаления шума

# Гистограмма изображения

---

- Распределение интенсивностей.
- В «обычном» изображении  $[0..255]$  или  $[0..1]$
- В биомедицине уровней гораздо больше, часто уровень важен.



# Виды применяемых фильтров

---

- Точечные?
- Локальные – ДА!
- Глобальные?
  
- Основное понятие – свертка (но не всегда!!!).
- Например, упорядочивающие фильтры – не сверточные.



# Методы удаления шума

---

- Сглаживающие фильтры
- Фильтры Винера
- Медианные фильтры
- Билатериальный фильтр
- Фильтр, управляющий величиной коррекции
- Фильтрация спектра (фурье)
- Фильтры, выделяющие доминантное направление (например, фильтрация в направлении перпендикулярном градиенту)
- Математическая морфология
  
- Классификация: линейные и нелинейные

# Сглаживающие фильтры

□ Арифметическое среднее  $\hat{f}(x, y) = \frac{1}{mn} \sum_{(s,t) \in S_{xy}} g(s, t)$

□ Геометрическое среднее

□ Меньше теряет деталей изображения

$$\hat{f}(x, y) = \left[ \prod_{(s,t) \in S_{xy}} g(s, t) \right]^{\frac{1}{mn}}$$

□ Гармоническое среднее

□ Гауссов и “белый” шум +

□ “Черный” шум –

$$\hat{f}(x, y) = \frac{mn}{\sum_{(s,t) \in S_{xy}} g(s, t)}$$

□ Контргармоническое среднее

□  $Q < 0$  – удаление “белого” шума

□  $Q > 0$  – удаление “черного” шума

$$\hat{f}(x, y) = \frac{\sum_{(s,t) \in S_{xy}} g(s, t)^{Q+1}}{\sum_{(s,t) \in S_{xy}} g(s, t)^Q}$$

# Адаптивная фильтрация

---

- В фильтрации не участвуют пиксели, значение которых слишком отличается от значения центрального рассматриваемого пикселя.
- Или участвуют, но со специально подсчитанными весами.

# Метод Гаусса

- Сепарабелен
- Фильтр низких частот
- Подходит для удаления постоянного и гауссовского шума
- Удаляет границы на изображении
- Как выбрать сигма? Зависит от задачи и что Вы знаете о ней
  - Например, 2% от размера диагонали изображения
- Важно лишь взаимное расположение пикселей

$$G_{\sigma} = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}}$$

$$GB[I]_p = \sum_{q \in S} G_{\sigma}(\|p - q\|) I_q,$$

$$G_{\sigma}(x) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2}{2\sigma^2}\right).$$

# Билатериальный фильтр (адапт)

---

- Не всегда лучший
- Часто очень хорош

$$BF[I]_p = \frac{1}{W_p} \sum_{q \in \mathcal{S}} G_{\sigma_s}(\|p - q\|) G_{\sigma_r}(I_p - I_q) I_q$$

$$W_p = \sum_{q \in \mathcal{S}} G_{\sigma_s}(\|p - q\|) G_{\sigma_r}(I_p - I_q)$$

- Важно и взаимное расположение пикселей и разница в цветовой компоненте
- Как найти оптимальные параметры?

# Метод нелокальных средних

Non-local means (NL-means) – веса зависят от близости целых блоков, а не отдельных пикселей

$v(x_{i,j})$  – блок вокруг пикселя  $x_{i,j}$

$$y_{i,j} = \sum_{k,m \in Q} x_{i+k,j+m} \cdot W(i,j,k,m)$$

$$W(i,j,k,m) \approx \exp\left(-\frac{\|v(x_{i,j}) - v(x_{i+k,j+m})\|^2}{h^2}\right)$$

# Упорядочивающие фильтры

□ Медианный фильтр  $\hat{f}(x, y) = \underset{(s,t) \in S_{xy}}{\text{median}}\{g(s, t)\}$

□ Удаление шума “соль-перец”

□ Фильтр максимума

$$\hat{f}(x, y) = \underset{(s,t) \in S_{xy}}{\max}\{g(s, t)\}$$

□ Удаление “черного” шума

□ Фильтр “минимума”

$$\hat{f}(x, y) = \underset{(s,t) \in S_{xy}}{\min}\{g(s, t)\}$$

□ Удаление “белого” шума

□ Фильтр “средней точки”

□ Удаление Гауссова шума и

$$\hat{f}(x, y) = \frac{1}{2} \left[ \underset{(s,t) \in S_{xy}}{\max}\{g(s, t)\} + \underset{(s,t) \in S_{xy}}{\min}\{g(s, t)\} \right]$$

□ Постоянного шума

□ Усреднение с альфа-сдвигом

$$\hat{f}(x, y) = \frac{1}{mn - d} \sum_{(s,t) \in S_{xy}} g_r(s, t)$$

□ Удалить  $d/2$  наибольших и  $d/2$  наименьших значений

# Упорядочивающие фильтры

---

- Let  $x_{(1)} \leq x_{(2)} \leq \dots \leq x_{(p)}$  be the sorted values.
- These values are known as order statistics.
- Let  $a_{(1)}, a_{(2)}, \dots, a_{(p)}$  be the **corresponding** weights.

Find  $i^*$  such that the following equations hold

$$a_{i^*} + \sum_{i=1}^{i^*-1} a_{(i)} \geq \sum_{i=i^*+1}^p a_{(i)}$$
$$\sum_{i=1}^{i^*-1} a_{(i)} \leq \sum_{i=i^*+1}^p a_{(i)} + a_{i^*}$$

Then the value  $x_{(i^*)}$  is the weighted median value.





# Методы на основе вейвлет-преобразования. Фильтр Винера

---

- Алгоритмы на основе вейвлет-преобразования и метода главных компонент применяются, в основном, для обработки изображений
- Фильтр Винера
- Отличное качество
- Медленно !!!

# Метод главных компонент

---

- KLT –преобразование – на следующих лекциях.
- Основная идея – найти такой базис, в котором разброс значений будет минимален, проводить фильтрацию в этом базисе.

# Нейронные сети

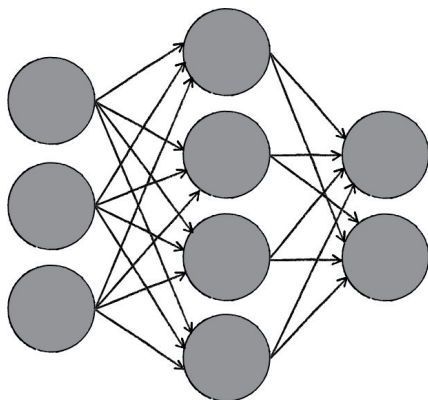
---

- По материалам
- <https://habr.com/ru/company/oleg-bunin/blog/340184/>

# Нейронные сети

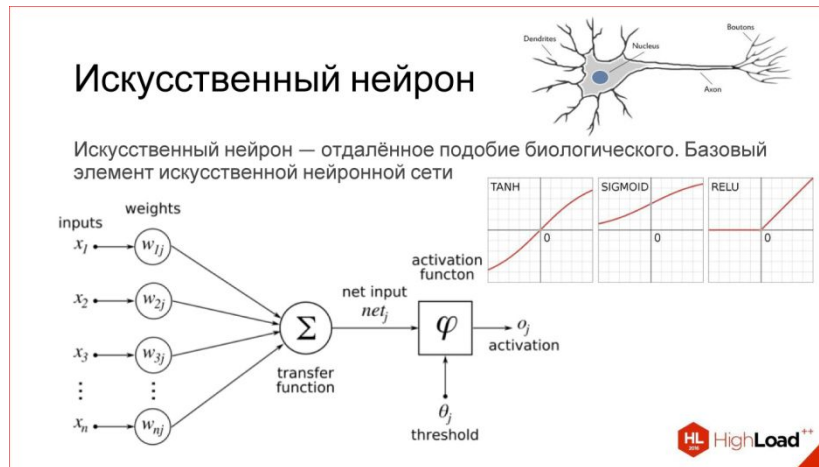
---

- Математическая модель мозга
  - Биологическая структура – схема/архитектура сетей
  - Электрические сигналы – числа.
  - Синапсы – веса. Усиливают или ослабляют проходящий сигнал.
- Задача: по заданному входу выдать ответ.



# Искусственный нейрон

- Нейрон должен как-то сформировать адекватный выходной сигнал. Именно для этих целей и используют функцию активации. Она преобразует взвешенную сумму в какое-то число, которое и является выходом нейрона (выход нейрона обозначим переменной  $out$ ).



- 
- Сети прямого распространения
  - Сети обратного распространения
    - -восстановление и дополнение сигнала
    - свойства кратковременной памяти (как у человека).
  
  - Поиск набора весовых коэффициентов, при котором входной сигнал после прохода по сети преобразуется в нужный нам выходной – обучение.

# Архитектуры нейронных сетей

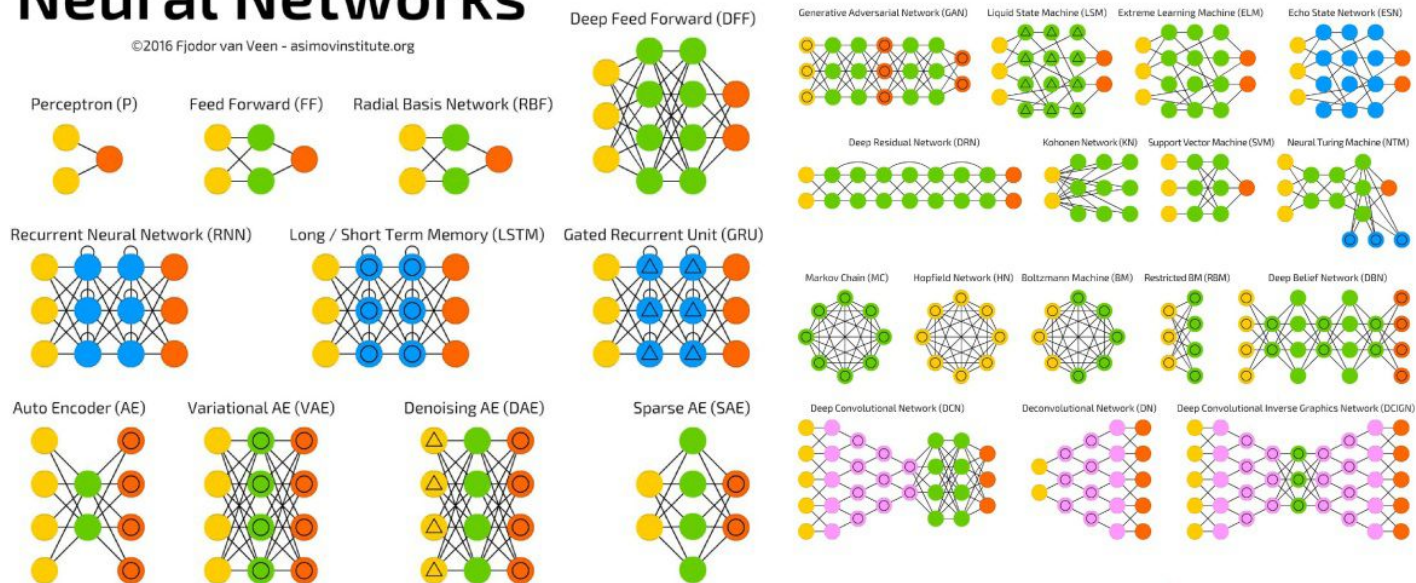
## Существует многообразие архитектур

A mostly complete chart of

### Neural Networks

©2016 Fjodor van Veen - asimovinstitute.org

- Backfed Input Cell
- Input Cell
- △ Noisy Input Cell
- Hidden Cell
- Probabilistic Hidden Cell
- △ Spiking Hidden Cell
- Output Cell
- Match Input Output Cell
- Recurrent Cell
- Memory Cell
- △ Different Memory Cell
- Kernel
- Convolution or Pool



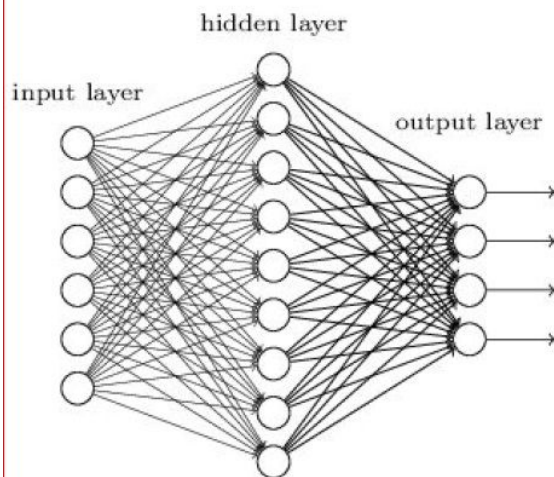
<http://www.asimovinstitute.org/neural-network-zoo/>



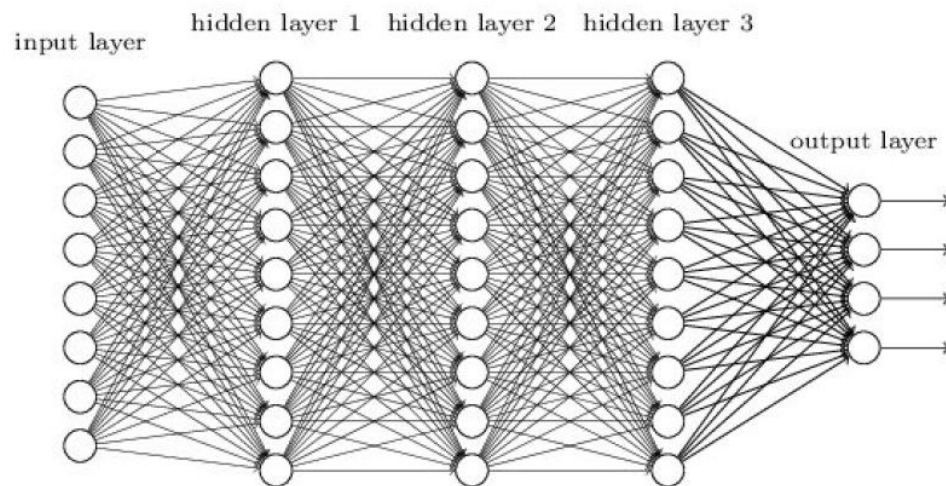
# Глубокие и неглубокие сети

## Искусственная нейросеть

"Non-deep" feedforward neural network



Deep neural network





# Все растёт

---

- Точность сетей растёт
- Сложность сетей растёт
- Объёмы данных растут
  
- Класс!
- Больше – лучше (шутка).

# Convolutional Neural Networks, CNN

---

- Сверточные сети

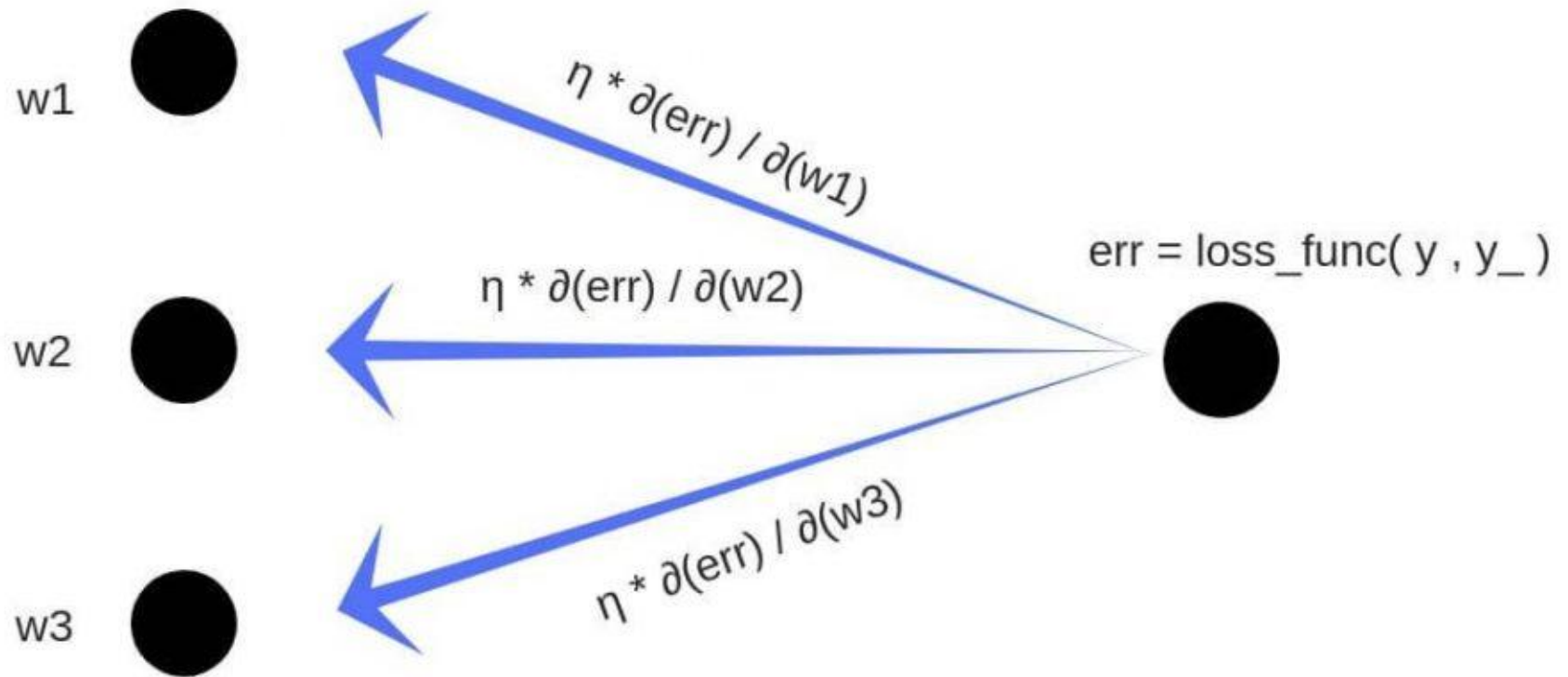
- Сверточные нейросети решают 3 основные задачи:

**Классификация.** Вы подаете картинку, и нейросеть просто говорит — у вас картинка про собаку, про лошадь, еще про что-то, и выдает класс.

- **Детекция** – это более продвинутая задача, когда нейросеть не просто говорит, что на картинке есть собака или лошадь, но находит еще Bounding box — где это находится на картинке.

- **Сегментация.** На мой взгляд, это самая крутая задача. По сути, это попиксельная классификация. Здесь мы говорим про каждый пиксель изображения: этот пиксель относится к собаке, этот — к лошади, а этот еще к чему-то. На самом деле, если вы умеете решать задачу сегментации, то остальные 2 задачи уже автоматически даны.

# Обучение



# Обучение

---

- $f(x)=A(B(C(x)))$
- A, B, и C – функции активации на различных слоях.
- $f'(x)=f'(A) \cdot A'(B) \cdot B'(C) \cdot C'(x)$
- $f'(B)=f'(A) \cdot A'(B)$
- Этот простой метод распространяется на любую переменную внутри функции, и позволяет нам **в точности определить влияние** каждой переменной на общий результат.
- Учитывая сеть, состоящую из одного нейрона, общая потеря нейросети может быть рассчитана как:
- $Cost=C(R(Z(XW)))$
- Используя правило цепочки, мы легко можем найти производную потери относительно веса  $W$ .
- $C'(W)=C'(R) \cdot R'(Z) \cdot Z'(W)=(\hat{y}-y) \cdot R'(Z) \cdot X$

---

Function	Formula	Derivative
Weighted input	$Z = XW$	$Z'(X) = W$ $Z'(W) = X$
ReLU activation	$R = \max(0, Z)$	$R'(Z) = \begin{cases} 0 & Z < 0 \\ 1 & Z > 0 \end{cases}$
Cost function	$C = \frac{1}{2}(\hat{y} - y)^2$	$C'(\hat{y}) = (\hat{y} - y)$

# Фильтрация

H. Chen *et al.*, "Low-dose CT denoising with convolutional neural network," *2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)*, Melbourne, VIC, 2017, pp. 143-146.  
doi: 10.1109/ISBI.2017.7950488

Свертка – раз (перевод в пространство фич)  
Свертка – два (фильтрация)  
Свертка для восстановления – три  
Придумывайте что угодно, лишь бы возможно было посчитать производную.

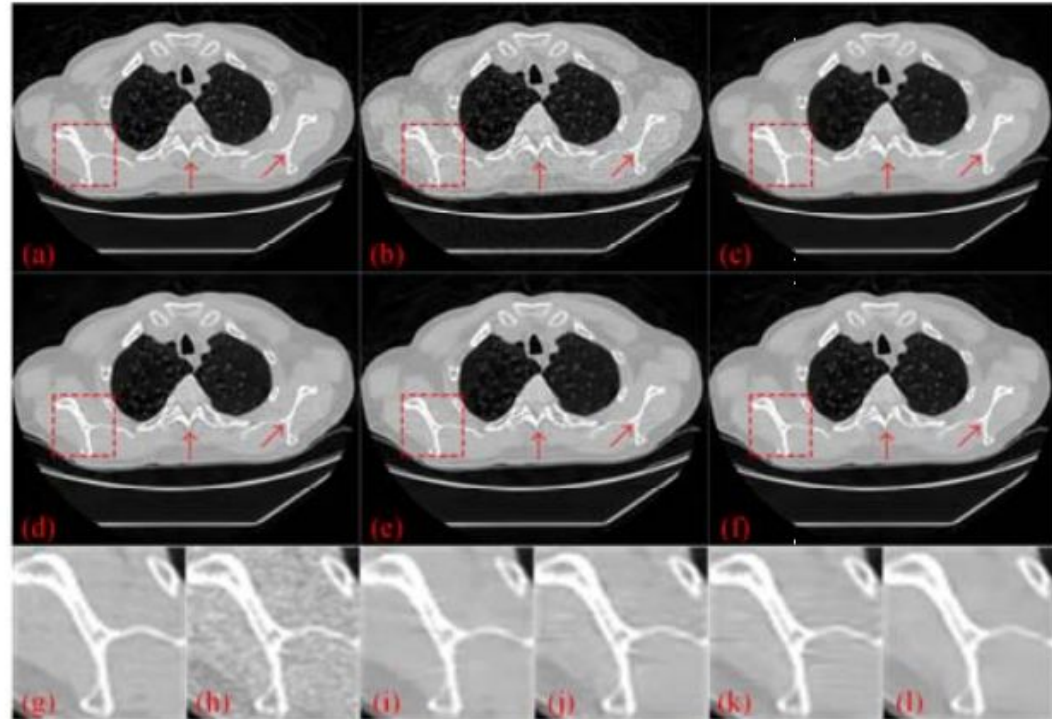


Fig. 2. Results of a slice of chest images. (a) Original normal-dose image; (b) the low-dose images; (c) the ASD-POCS reconstructed image; (d) the KSVD processed low-dose image; (e) the BM3D processed low-dose image; (f) the CNN processed low-dose image; and (g)-(l) the zoomed regions specified with the red boxes in (a)-(f).



**УНИВЕРСИТЕТ  
ЛОБАЧЕВСКОГО**

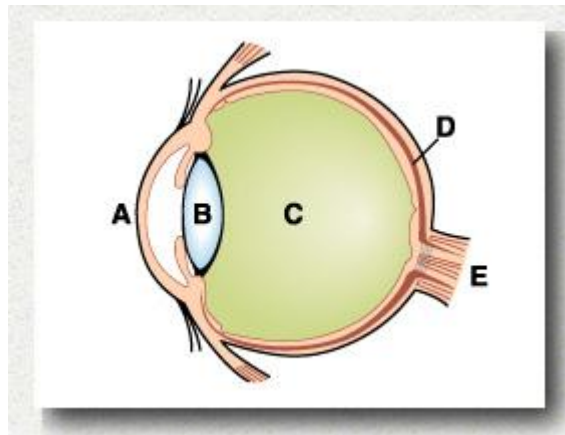
Институт информационных технологий, математики и  
механики  
Кафедра математического обеспечения и  
суперкомпьютерных технологий

**Обработка изображений**

**Теория цвета**

Светлана Носова  
svetlana.nosova@itmm.unn.ru

# Восприятие цвета человеком



A: роговая оболочка глаза

B: хрусталик

C: стекловидное тело

D: сетчатка

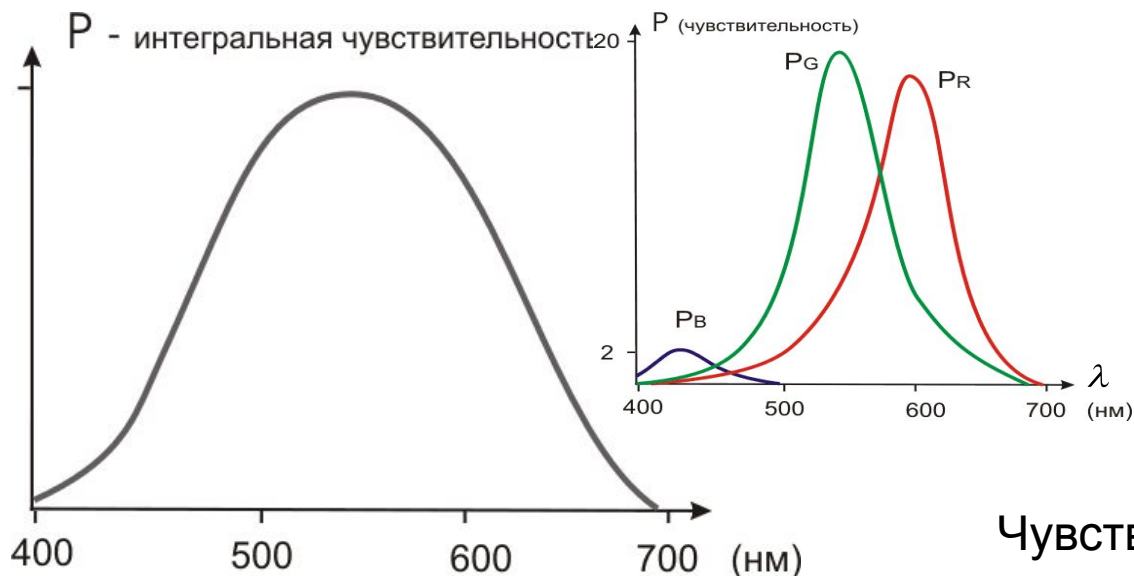
E: оптический нерв

Палочки (rods): 75–150 млн

Колбочки (cones): 6–7 млн, в ямке (fovea), – воспринимают цвет. К каждой – свой нерв.



# Источники света



$$R = \int I(\lambda) \cdot P_R(\lambda) d\lambda$$

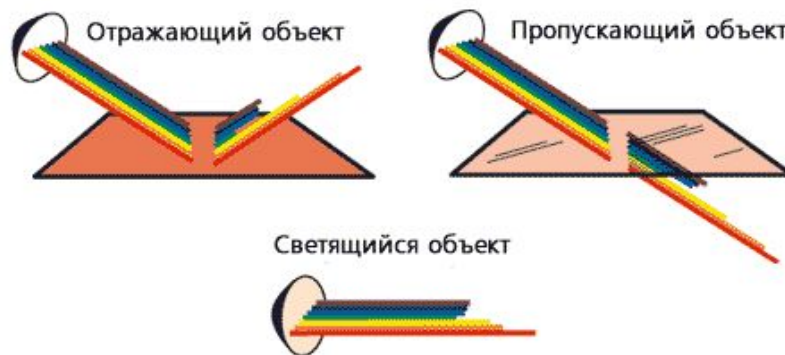
$$G = \int I(\lambda) \cdot P_G(\lambda) d\lambda$$

$$B = \int I(\lambda) \cdot P_B(\lambda) d\lambda$$

(1)

$I(\lambda)$  - зависимость  
интенсивности  
света от длины волны

Чувствительность колбочек



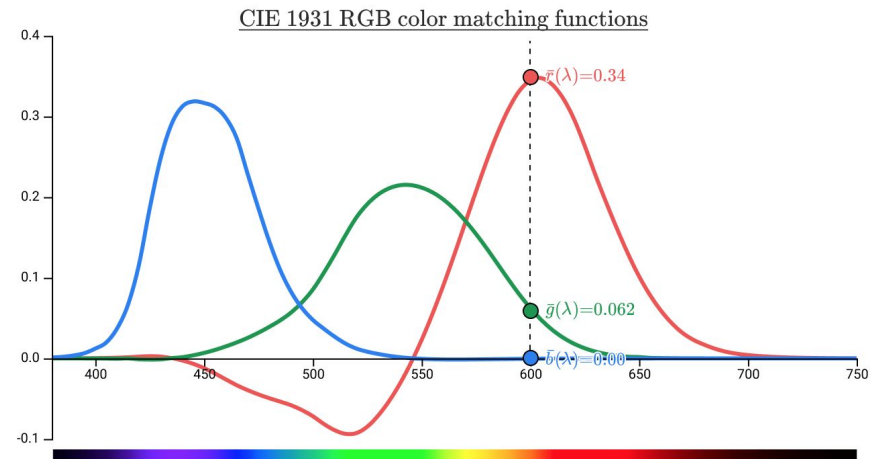
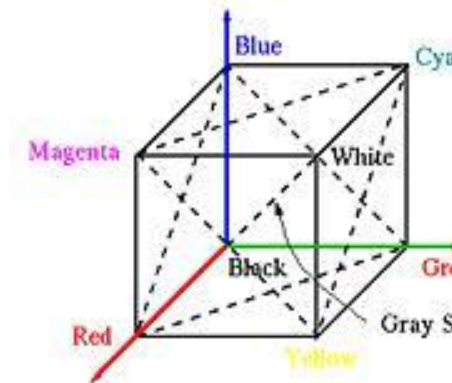
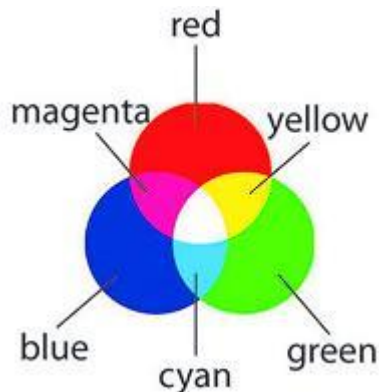
# Законы Германа Грассмана (аддитивной модели света)

---

- **1.Закон трехмерности:** Любой цвет однозначно выражается тремя, если они линейно независимы. Линейная независимость заключается в том, что ни один из этих трех цветов нельзя получить сложением двух остальных.
- **2.Закон непрерывности:** При непрерывном изменении излучения цвет смеси также меняется непрерывно. Не существует такого цвета, к которому нельзя было бы подобрать бесконечно близкий.
- **3.Закон аддитивности:** Цвет смеси излучений зависит только от компонентов их цвета, а не от спектрального состава.
- Как следствие – аддитивность цветковых уравнений: для смеси двух цветов  $C_1$  и  $C_2$  имеет место равенство:  
$$C = C_1 + C_2 = (r_1 + r_2)R + (g_1 + g_2)G + (b_1 + b_2)B$$

# RGB

- Основные принципы восприятия человеческим глазом.
- Отобразить цвет так, как видит человек.
- Аддитивная модель.
- Каждая компонента – 1 байт [0..255].
- Возможны отрицательные координаты.



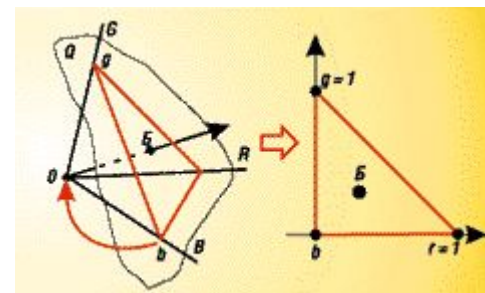
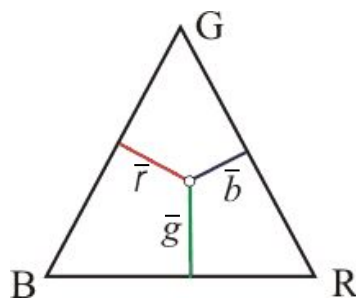
# RGB треугольник Максвелла

$$\bar{r} = \frac{r}{r+g+b}$$

$$\bar{g} = \frac{g}{r+g+b}$$

$$\bar{b} = \frac{b}{r+g+b}$$

$$\bar{r} + \bar{g} + \bar{b} = 1$$



$$\lambda_R = 630\text{нм}, \lambda_G = 528\text{нм}, \lambda_B = 457\text{нм}$$

# Перевод из RGB обратно

- Не все видимые цвета представимы
- Или некоторые цвета имеют отрицательные коэффициенты
- Неудобно работать.

$$\begin{vmatrix} Y(i, j) \\ X(i, j) \\ Z(i, j) \end{vmatrix} = \begin{vmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{vmatrix} \times \begin{vmatrix} R(i, j) \\ G(i, j) \\ B(i, j) \end{vmatrix}.$$

$$\begin{vmatrix} R'(i, j) \\ G'(i, j) \\ B'(i, j) \end{vmatrix} = \begin{vmatrix} c_{11}^{-1} & c_{12}^{-1} & c_{13}^{-1} \\ c_{21}^{-1} & c_{22}^{-1} & c_{23}^{-1} \\ c_{31}^{-1} & c_{32}^{-1} & c_{33}^{-1} \end{vmatrix} \times \begin{vmatrix} Y(i, j) \\ X(i, j) \\ Z(i, j) \end{vmatrix}.$$

# RGBs

---

*Красный*



*Канал  
Зелёный*



*Синий*



*Исходное  
изображение*



# YIQ, YUV, YCbCr

---

- Стандарты аналогового цифрового телевидения
- Уменьшение полосы пропускания для цветного сигнала.
- БОльшая доля яркостной Y компоненты.
- Позже – YCbCr для JPEG.
- В формате JPEG применяется модель YIQ телевизионного стандарта NTSC (USA). В канале Y яркость подобрана так, что она также близка к цветовой чувствительности глаза. Канал I соответствует цветам от голубого до оранжевого (теплым тонам). Канал Q – от зеленого до пурпурного. В качестве опорного белого был взят источник с температурой 6500K.

# YIQ, YUV, YCbCr

	Матрица прямого преобразования			Матрица обратного преобразования			max	min
	R	G	B	Y	I	Q		
Y	0,299	0,587	0,114	1	0,956	0,621	255	0
I	0,596	-0,274	-0,322	1	-0,272	-0,647	151,98	-151,98
Q	0,211	-0,523	0,312	1	-1,106	1,703	133,37	-133,37
	R	G	B	Y	U	V		
Y	0,299	0,587	0,114	1	-0,048	1,138	255	0
U	-0,147	-0,289	0,437	1	-0,37	-0,579	111,44	-111,44
V	0,615	-0,515	-0,09	1	2,028	0	156,83	-154,49
	R	G	B	Y	D <sub>r</sub>	D <sub>b</sub>		
Y	0,299	0,587	0,114	1	0	0,526	255	0
D <sub>r</sub>	-0,449	-0,881	1,329	1	-0,1295	-0,2681	338,9	-339,15
D <sub>b</sub>	1,332	-1,115	-0,217	1	0,667	0	339,15	-339,15
	R	G	B	Y	C <sub>b</sub>	C <sub>r</sub>		
Y	0,299	0,587	0,114	1	0	1,402	255	0
C <sub>r</sub>	0,5	-0,4187	-0,0813	1	1,772	0	127,5	-127,5
C <sub>b</sub>	-0,1687	-0,3313	0,5	1	-0,3441	-0,7141	127,5	-127,5



# YIQ

---

$$R, G, B, Y \in [0, 1], \quad I \in [-0.5957, 0.5957], \quad Q \in [-0.5226, 0.5226]$$

От RGB до YIQ

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.274 & -0.322 \\ 0.211 & -0.523 & 0.312 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

От YIQ до RGB

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 & 0.956 & 0.621 \\ 1 & -0.272 & -0.647 \\ 1 & -1.106 & 1.703 \end{bmatrix} \begin{bmatrix} Y \\ I \\ Q \end{bmatrix}$$

# YCbCr

## □ Cb,Cr

$$Y = k_r R + k_g G + k_b B$$

$$Cb = B - Y,$$

$$Cr = R - Y,$$

$$Cg = G - Y.$$

$$Y = 0,299R + 0,587G + 0,114B,$$

$$Cb = 0,564(B - Y),$$

$$Cr = 0,713(R - Y);$$

(2.5)

$$R = Y + 1,402Cr,$$

$$G = Y - 0,344Cb - 0,714Cr,$$

$$B = Y + 1,772Cb.$$

(2.6)

$$Y = k_r R + (1 - k_b - k_r)G + k_b B,$$

$$Cb = \frac{0,5}{1 - k_b} (B - Y),$$

$$Cr = \frac{0,5}{1 - k_r} (R - Y);$$

(2.3)

$$R = Y + \frac{1 - k_r}{0,5} Cr,$$

$$G = Y - \frac{2k_b(1 - k_b)}{1 - k_b - k_r} Cb - \frac{2k_r(1 - k_r)}{1 - k_b - k_r} Cr,$$

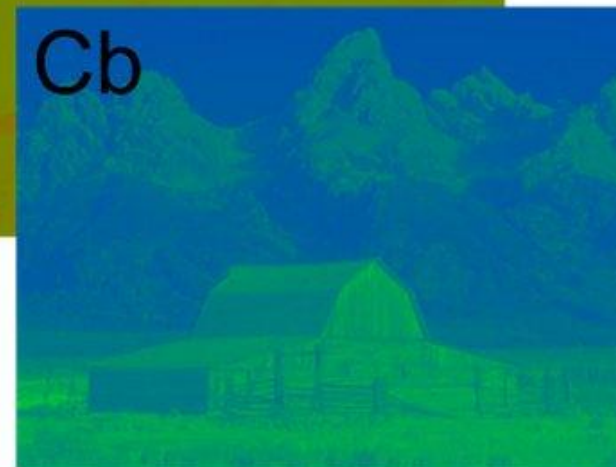
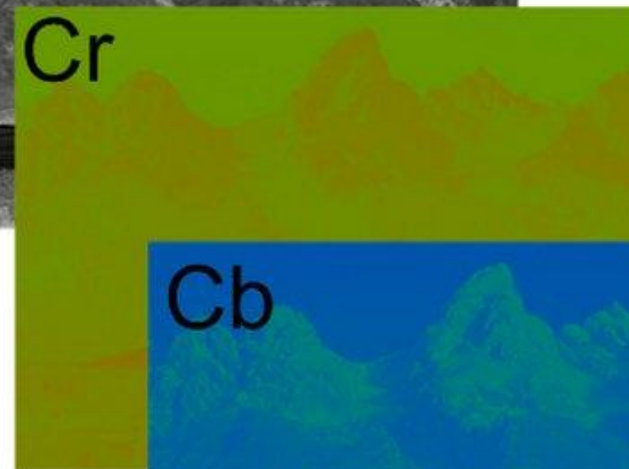
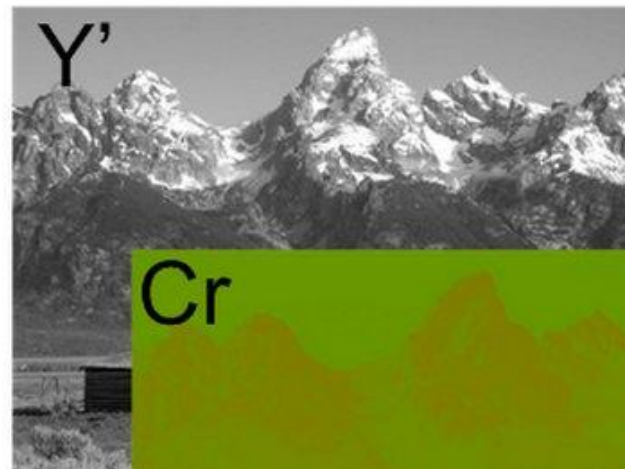
$$B = Y + \frac{1 - k_b}{0,5} Cb.$$

(2.4)

# YIQ, YUV, YCbCr



RGB  
to  
Y'CrCb  
→



	Normal Range	White	Yellow	Cyan	Green	Magenta	Red	Blue	Black
<b>R</b>	0 to 255	255	255	0	0	255	255	0	0
<b>G</b>	0 to 255	255	255	255	255	0	0	0	0
<b>B</b>	0 to 255	255	0	255	0	255	0	255	0

# XYZ

---

- Основа аппаратнонезависимого преобразования моделей
- **Координаты цветности CIE представляют точный стандарт определения цвета.**
- Их необходимо знать для преобразования координат CIE в другие цветовые модели и обратно.

# XYZ

- Преобразование RGB - CIE XYZ задается следующей формулой:

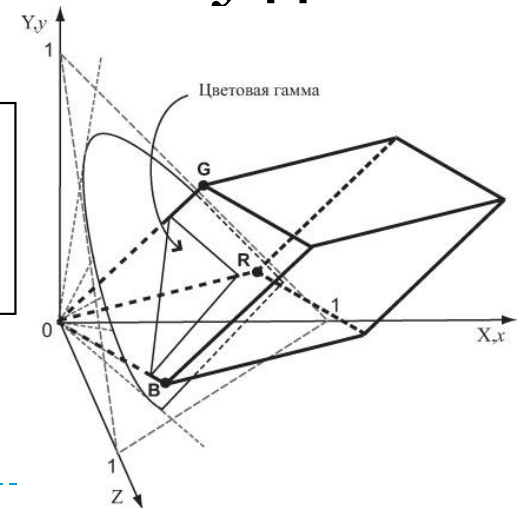
$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} X_r & X_g & X_b \\ Y_r & Y_g & Y_b \\ Z_r & Z_g & Z_b \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}.$$

где  $x_r, y_r, z_r$  - цвета для получения координат единичного основного цвета R, аналогично и для G и B.

**Если известны координаты цветности CIE  $x$  и  $y$  для основных цветов RGB**

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} x_r C_r & x_g C_g & x_b C_b \\ y_r C_r & y_g C_g & y_b C_b \\ (1-x_r-y_r)C_r & (1-x_g-y_g)C_g & (1-x_b-y_b)C_b \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

$$C_g = X_g + Y_g + Z_g, C_b = X_b + Y_b + Z_b, C_r = X_r + Y_r + Z_r$$



# RGB – XYZ

RGB Working Space	Reference White	RGB to XYZ [M]	XYZ to RGB [M] <sup>-1</sup>
Adobe RGB (1998)	D65	0.5767309 0.1855540 0.1861852 0.2973769 0.6273491 0.0752741 0.0270343 0.0706872 0.9911085	2.0413690 -0.5649464 -0.3446944 -0.9692660 1.8760108 0.0415560 0.0134474 -0.1183897 1.0154096
AppleRGB	D65	0.4497288 0.3162486 0.1844926 0.2446525 0.6720283 0.0833192 0.0251848 0.1411824 0.9224628	2.9515373 -1.2894116 -0.4738445 -1.0851093 1.9908566 0.0372026 0.0854934 -0.2694964 1.0912975
Best RGB	D50	0.6326696 0.2045558 0.1269946 0.2284569 0.7373523 0.0341908 0.0000000 0.0095142 0.8156958	1.7552599 -0.4836786 -0.2530000 -0.5441336 1.5068789 0.0215528 0.0063467 -0.0175761 1.2256959
Beta RGB	D50	0.6712537 0.1745834 0.1183829 0.3032726 0.6637861 0.0329413 0.0000000 0.0407010 0.7845090	1.6832270 -0.4282363 -0.2360185 -0.7710229 1.7065571 0.0446900 0.0400013 -0.0885376 1.2723640
Bruce RGB	D65	0.4674162 0.2944512 0.1886026 0.2410115 0.6835475 0.0754410 0.0219101 0.0736128 0.9933071	2.7454669 -1.1358136 -0.4350269 -0.9692660 1.8760108 0.0415560 0.0112723 -0.1139754 1.0132541
CIE RGB	E	0.4887180 0.3106803 0.2006017 0.1762044 0.8129847 0.0108109 0.0000000 0.0102048 0.9897952	2.3706743 -0.9000405 -0.4706338 -0.5138850 1.4253036 0.0885814 0.0052982 -0.0146949 1.0093968

# Lab

---

- Точная установка цвета в печатной продукции
- $L^*a^*b^*$  – линеаризованная и приведенная модель XYZ
- Ось **a** проходит от зеленого ( $-a$ ) до красного ( $+a$ ), а ось **b** – от синего ( $-b$ ) до желтого ( $+b$ ).
- Яркость (**L**) возрастает снизу вверх вдоль диагонали RGB-куба.
- По сравнению с цветовой моделью XYZ цвета CIE  $L^*a^*b^*$  более совместимы с цветами, воспринимаемыми человеческим глазом.
- В модели CIE  $L^*a^*b^*$  яркость (**L**), цветовой тон и насыщенность (**a**, **b**) могут рассматриваться отдельно.
- В результате цвет изображения можно изменять без изменения самого изображения или его яркости.
- Поскольку CIE  $L^*a^*b^*$  не зависит от устройства, то, при переходе от RGB к CMYK или от CMYK к RGB, полезно во избежание потерь использовать модель CIE  $L^*a^*b^*$  в качестве промежуточной.

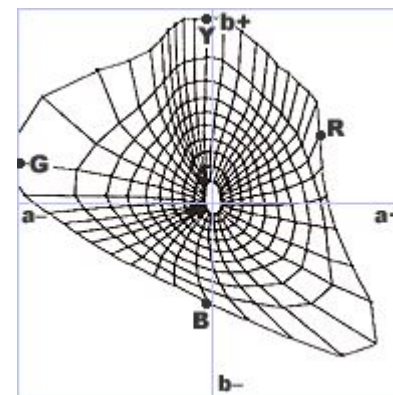
# Lab

$$L^* = \begin{cases} 116(Y/Y_n)^{1/3} - 16, & Y/Y_n > 0.008856 \\ 903.3(Y/Y_n), & Y/Y_n \leq 0.008856 \end{cases}$$

$$a^* = 500 \left[ (X/X_n)^{1/3} - (Y/Y_n)^{1/3} \right]$$

$$b^* = 200 \left[ (Y/Y_n)^{1/3} - (Z/Z_n)^{1/3} \right]$$

, где  $X_n, Y_n, Z_n$  - координаты белого цвета





# Lab



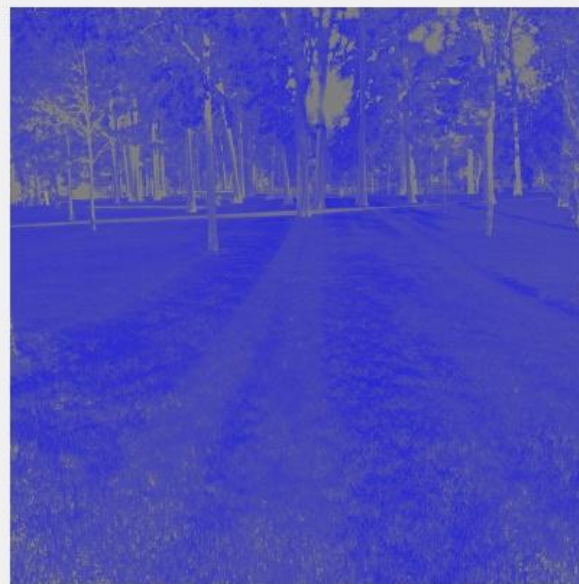
Source



L



a



b

# HSV

$$R' = R/255$$

$$G' = G/255$$

$$B' = B/255$$

$$C_{max} = \max(R', G', B')$$

$$C_{min} = \min(R', G', B')$$

$$\Delta = C_{max} - C_{min}$$

Hue calculation:

$$H = \begin{cases} 0^\circ & \Delta = 0 \\ 60^\circ \times \left(\frac{G'-B'}{\Delta} \bmod 6\right) & , C_{max} = R' \\ 60^\circ \times \left(\frac{B'-R'}{\Delta} + 2\right) & , C_{max} = G' \\ 60^\circ \times \left(\frac{R'-G'}{\Delta} + 4\right) & , C_{max} = B' \end{cases}$$

Saturation calculation:

$$S = \begin{cases} 0 & , C_{max} = 0 \\ \frac{\Delta}{C_{max}} & , C_{max} \neq 0 \end{cases}$$

Value calculation:

$$V = C_{max}$$

$$C = V \times S$$

$$X = C \times (1 - |(H / 60^\circ) \bmod 2 - 1|)$$

$$m = V - C$$

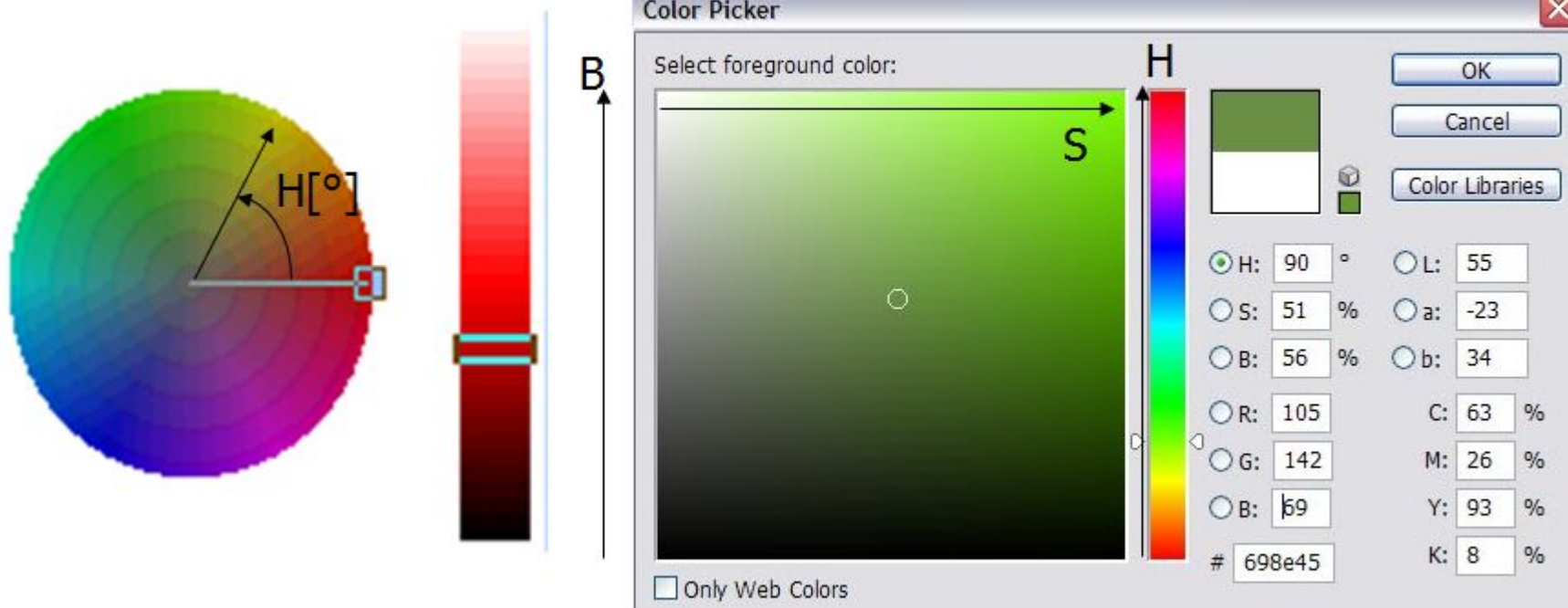
$$(R', G', B') = \begin{cases} (C, X, 0) & , 0^\circ \leq H < 60^\circ \\ (X, C, 0) & , 60^\circ \leq H < 120^\circ \\ (0, C, X) & , 120^\circ \leq H < 180^\circ \\ (0, X, C) & , 180^\circ \leq H < 240^\circ \\ (X, 0, C) & , 240^\circ \leq H < 300^\circ \\ (C, 0, X) & , 300^\circ \leq H < 360^\circ \end{cases}$$

$$(R, G, B) = ((R'+m) \times 255, (G'+m) \times 255, (B'+m) \times 255)$$

# HSB (HSL, HSV)

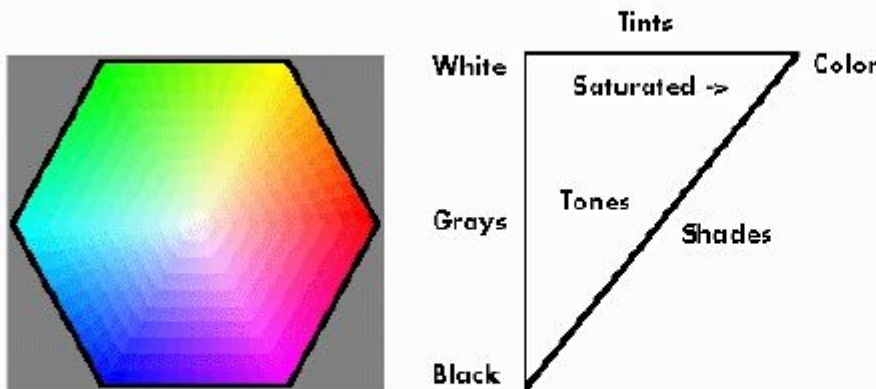
- «Интуитивный» выбор цветов дизайнерами и художниками

HSB – (Hue-°, Saturation-%, Brightness -%)



# HIS\HSV: Hue – Saturation – Intensity (Volume)

## Intuitive Color Spaces



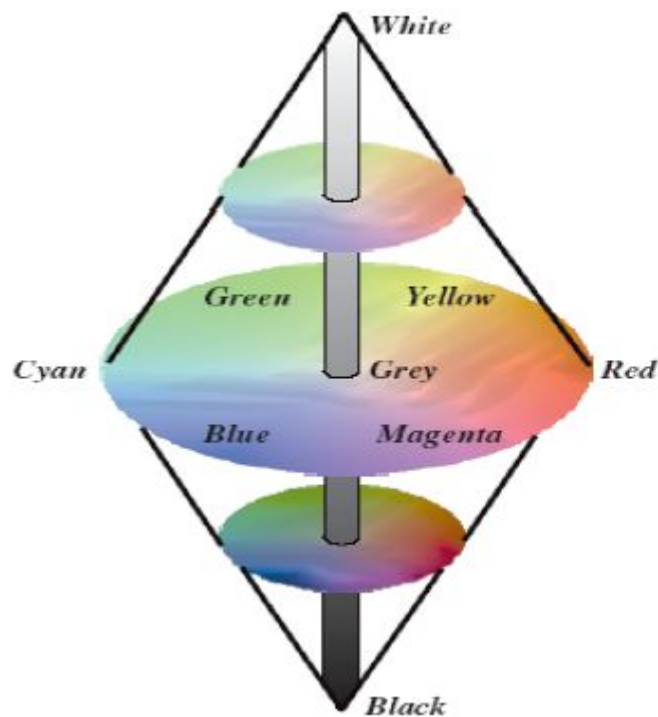
Hexagon is a diagonal Cross-Section of the Color Cube

Цветовой тон (H) измеряется углом вокруг вертикальной оси, причем красному, зеленому и синему цветам соответствует  $H = 0, 120$  и  $240$  градусов. Интенсивность (V) вдоль оси возрастает от 0 в вершине до 1 на верхней грани, где она максимальна для всех цветов. Насыщенность (S) определяется расстоянием от оси. На вертикальной оси находятся ахроматические, серые цвета.



# HLS: Hue–Lightness–Saturation

---



Цветовая модель HLS (Hue, Lightness, Saturation) в виде двойной шестигранной пирамиды является расширением одиночной пирамиды HSV.

# HLS

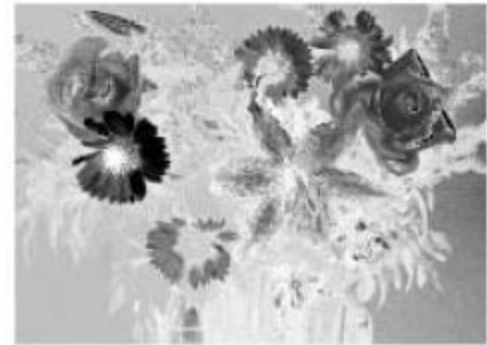
---



e



f



g



**Original and**  
(e) hue component;  
(f) intensity component;  
(g) saturation component

# Гамма коррекция. Глубина цвета

## Гамма-коррекция ИНТЕНСИВНОСТИ И ЦВЕТА:

$Y \sim U^{\text{Gamma}}$

## Глубина цвета:

- ❑ Черно-белое изображение (монохромное) – 1 bit/pix
- ❑ Полутоновое (256 оттенков серого) – 8 bit/pix
- ❑ Полноцветные изображения (High Color, True Color) – до 24 (3x8) – 32 (4x8) bit/pix
- ❑ Индексированный цвет (палитровое) – 8 bit/pix:

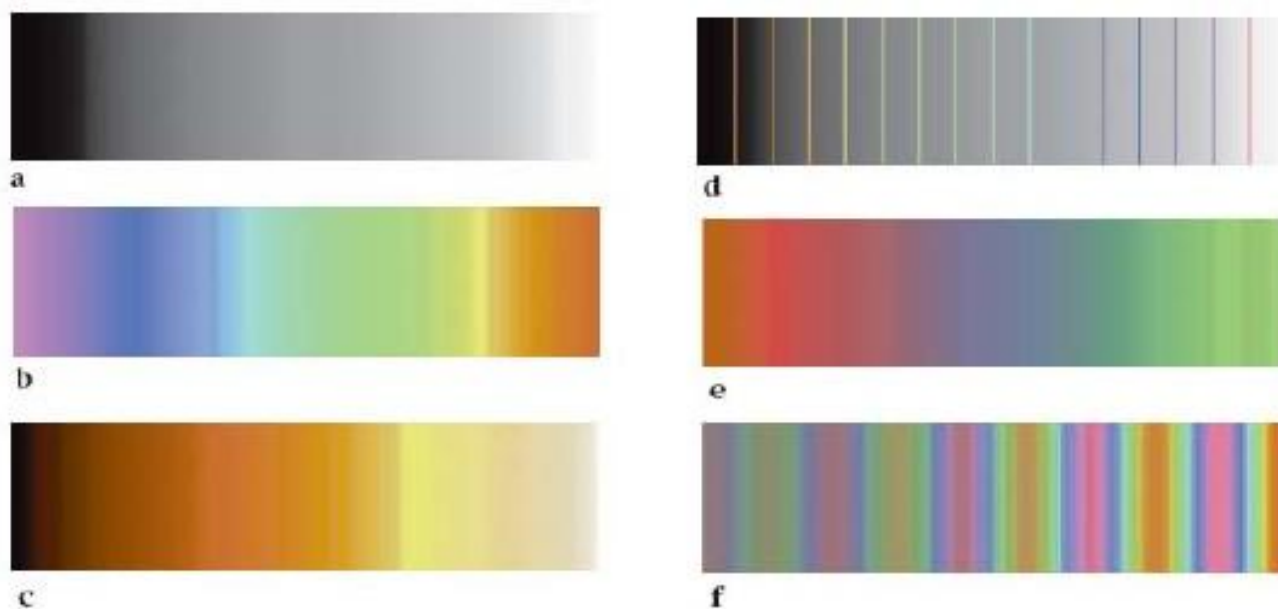
```
3 3 3 3 3 3 3 3
0 1 4 1 4 1 4 0
0 4 1 4 1 4 1 0
0 5 5 5 5 5 5 0
0 5 5 5 5 5 5 0
0 1 4 1 4 1 4 0
0 4 1 4 1 4 1 0
2 2 2 2 2 2 2 2
```



0	000000	Black
1	FF0000	Red
2	00FF00	Green
3	0000FF	Blue
4	FFFFFF	White
5	FFFF00	Yellow
6	FF00FF	Magenta
7	00FFFF	Cyan
8	FF0080	Pink
9	FF8040	Orange
A	804000	Brown
B	008080	Teal
C	800000	Dark Red
D	800080	Purple
E	8080FF	Light Blue

# Палитры

---



**6 примеров представления палитр (look-up tables, LUTs):** (a) monochrome (grey-scale); (b) спектральная радуга (variation of hue, with maximum saturation and constant intensity); (c) горячая шкала (heat scale); (d) monochrome with contour lines (rainbow colors substituted every 16th value); (e) плавное смешивание трех основных цветов; (f) синусоидальное изменение оттенка с линейным изменением насыщенности и интенсивности.



# Перенос цветовой палитры

---



## Задачи: Коррекция (подбор) цвета

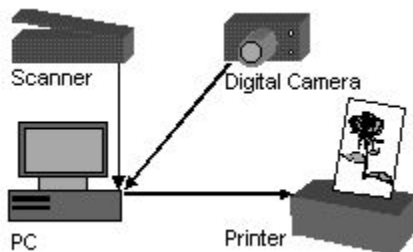
---



**Color matching by shifting hue values:**

- (a) изображение цветка в тени с областью для измерения оттенка (hue);
- (b) подобный цветок на прямом солнечном свете с областью для подгонки;
- (c) изображение (a), повернутое на  $12^\circ$  для выравнивания оттенка в двух указанных областях.

# Система управления цветом



Профиль монитора – колориметр (спектрофотометр), снимающий образцы цветов [см. также 2,6].

sRGB – усредненный профиль.

Профиль сканера – стандартный эталон IT-8

Профиль принтера

Печатается эталон IT-873. Цвета на нем измеряются, для данного режима печати и бумаги, с помощью прибора денситометра и передаются в программу вычисляющую цветовой профиль.

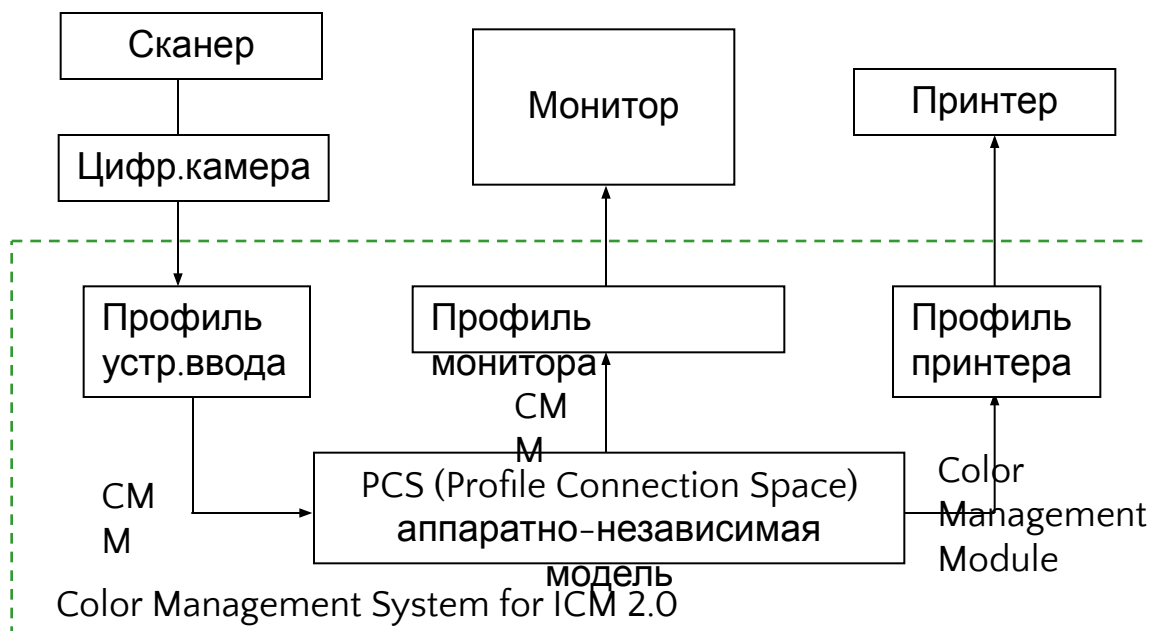


Image Color Management, ICM 2.0

ICC Profile Format Specification, available from the International Color Consortium (1993) <http://www.color.org>.

Понятия калибровки, идентификации, профиля.

Значение в полиграфии.

Расширение возможностей многоцветной печати.



**УНИВЕРСИТЕТ  
ЛОБАЧЕВСКОГО**

Институт информационных технологий, математики и  
механики  
Кафедра математического обеспечения и  
суперкомпьютерных технологий

## Обработка изображений

Преобразование цветовых  
пространств. Глубина цвета

Светлана Носова  
svetlana.nosova@itmm.unn.ru

# Глубина цвета

- Черно-белое изображение (монохромное) – 1 bit/pix
- Полутоновое (256 оттенков серого) – 8 bit/pix
- Полноцветные изображения (High Color, True Color) – до 24 (3x8) – 32 (4x8) bit/pix
- Индексированный цвет (палитровое) – 8 bit/pix:

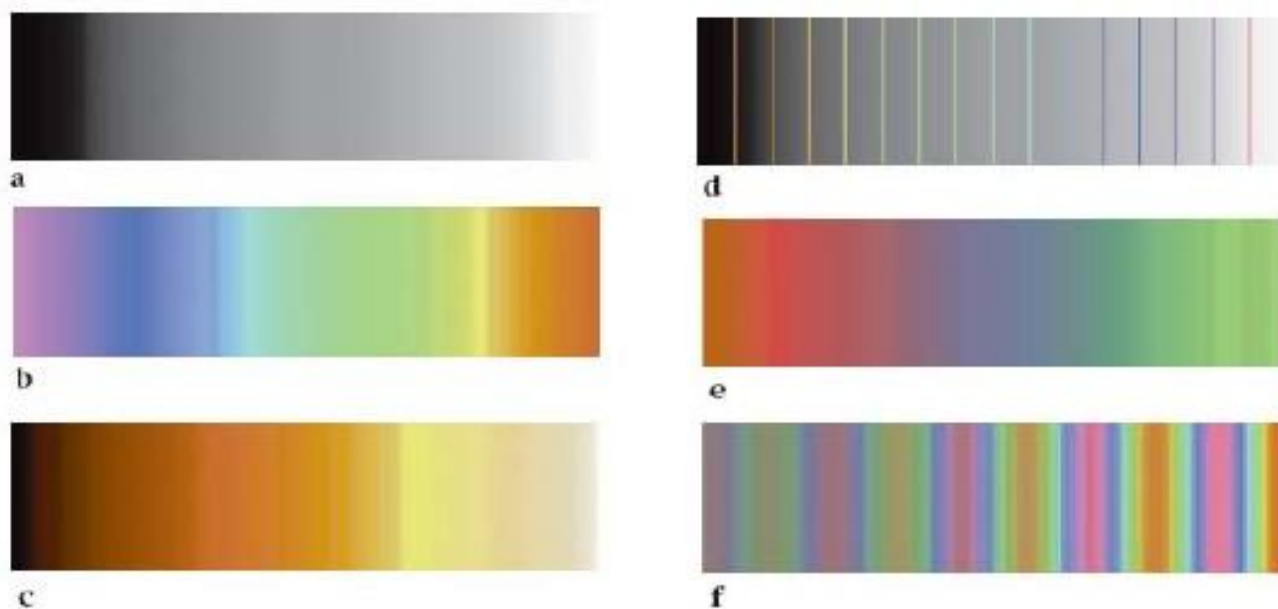
```
3 3 3 3 3 3 3 3
0 1 4 1 4 1 4 0
0 4 1 4 1 4 1 0
0 5 5 5 5 5 5 0
0 5 5 5 5 5 5 0
0 1 4 1 4 1 4 0
0 4 1 4 1 4 1 0
2 2 2 2 2 2 2 2
```



0	000000	Black
1	FF0000	Red
2	00FF00	Green
3	0000FF	Blue
4	FFFFFF	White
5	FFFF00	Yellow
6	FF00FF	Magenta
7	00FFFF	Cyan
8	FF0080	Pink
9	FF8040	Orange
A	804000	Brown
B	008080	Teal
C	800000	Dark Red
D	800080	Purple
E	8080FF	Light Blue

# Палитры

---



**6 примеров представления палитр (look-up tables, LUTs):** (a) monochrome (grey-scale); (b) спектральная радуга (variation of hue, with maximum saturation and constant intensity); (c) горячая шкала (heat scale); (d) monochrome with contour lines (rainbow colors substituted every 16th value); (e) плавное смешивание трех основных цветов; (f) синусоидальное изменение оттенка с линейным изменением насыщенности и интенсивности.

---

# Перевод в оттенки серого

# Классы фильтров

---

- Точечный
- Локальный
- Глобальный



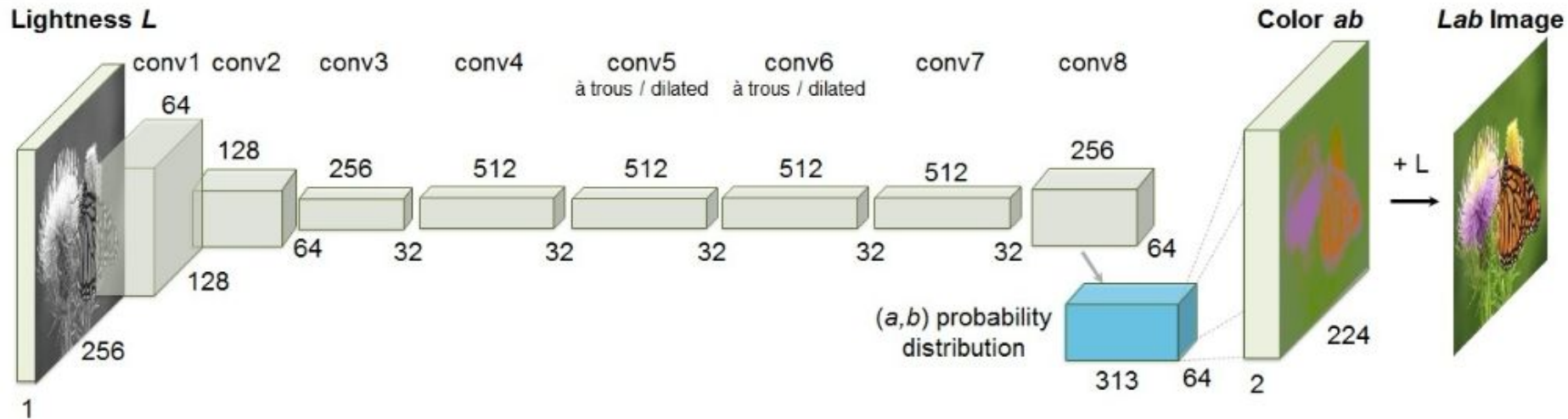
# RGB-Gray

---

- Точечный фильтр
- $R=G=B=I$ ,  $[0..255]$
- average  $(R + G + B) / 3$
- lightness  $(\max(R, G, B) + \min(R, G, B)) / 2$
- luminosity  $(0.21 R + 0.72 G + 0.07 B)$
- Photoshop, GIMP
  - $(0.3 \text{ Red} + 0.59 \text{ Green} + 0.11 \text{ Blue})$
- ITU-R, BT.709
  - $(0.2126 \text{ Red} + 0.7152 \text{ Green} + 0.0722 \text{ Blue})$
- $\text{Max}(\text{Red}, \text{Green}, \text{Blue})$
- $\text{Min}(\text{Red}, \text{Green}, \text{Blue})$
- $0.2952 R + 0.5547 G + 0.148 B$

# Gray - RGB

- Zhang, Isola, Efros. Colorful Image Colorization. In ECCV, 2016 (oral). (hosted on [arXiv](#))



• 2 или 3 повтора:

- Свертка
- Активация ReLU
- Пулинг (субдискретизация) – замена

# Gray - RGB



---

# Бинаризация

# Бинарные изображения

---

- Два возможных значений интенсивности (черный и белый).
- Часто: выбирается порог  $T$  для цветного или изображения в оттенках серого.
- Как найти  $T$ ?
  
- Алгоритмы сегментации на выходе: бинарное изображение (объект/фон).
- Алгоритмы выделения границ на выходе : бинарные изображения (есть граница/нет границы).
- В этот раз: делаем предположения относительно гистограммы изображения и вычисляем порог.

# Обработка бинарных изображений

---

- Подавление шума
- Run Length Coding
- Разметка связных компонент
- Выделение контуров
- Расчет главных диагоналей
- Утоньшение
- Фильтрация (морфологические операции)
- Извлечение признаков (размер, ориентация и т. Д.)

# Пример



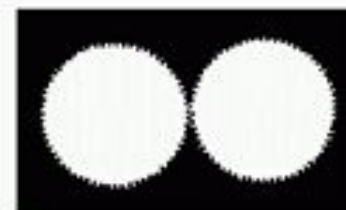
a) original image B



b)  $B1 = B \ominus \text{hole\_ring}$



c)  $B2 = B1 \oplus \text{hole\_mask}$



d)  $B3 = B \text{ OR } B2$



e) B7 (see text)



f)  $B8 = B \text{ AND } B7$



g)  $B9 = B8 \ominus \text{tip\_spacing}$



h)  $\text{RESULT} = ((B7 - B9) \oplus \text{defect\_cue}) \text{ OR } B9$

# Пример

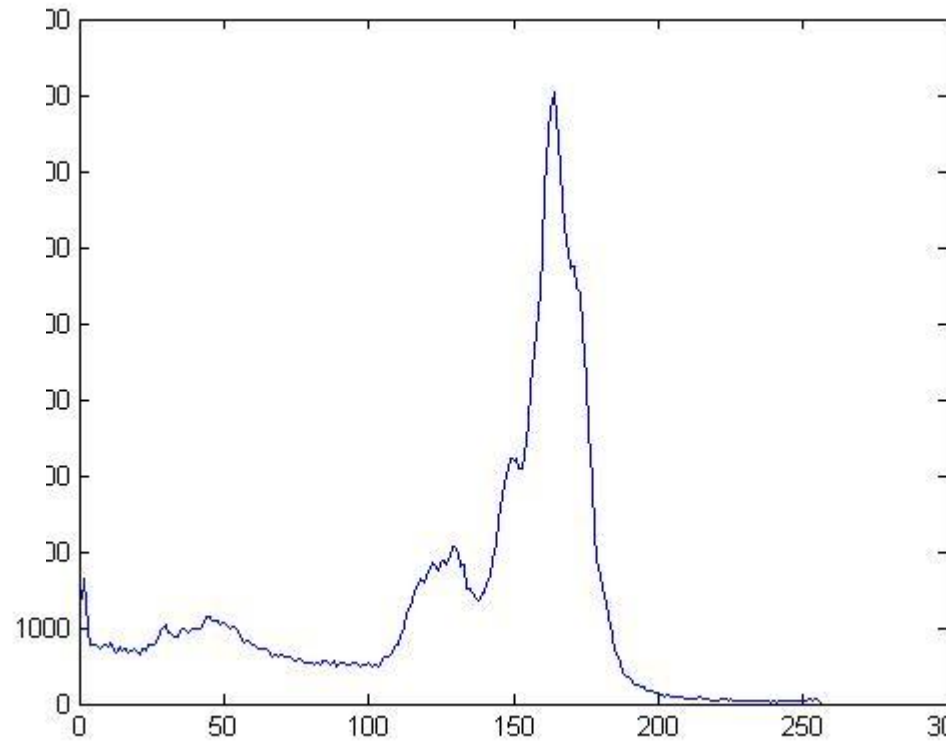
---





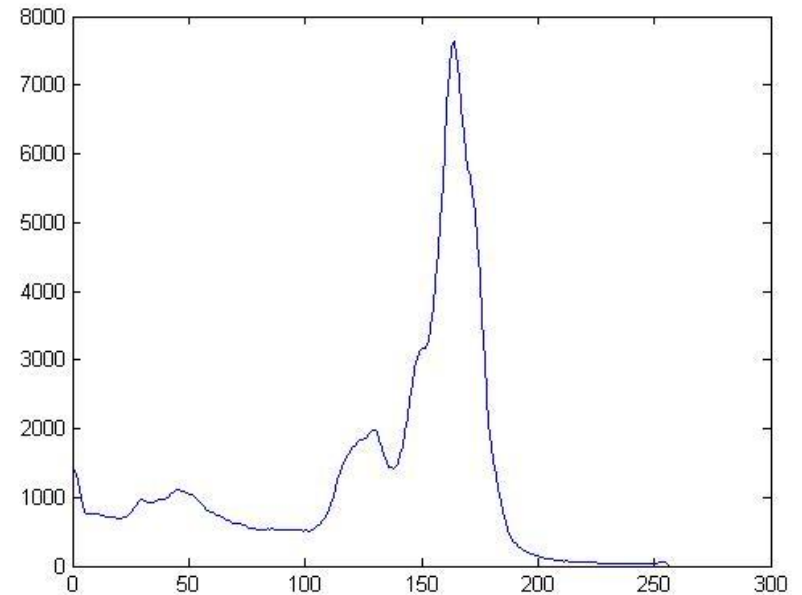
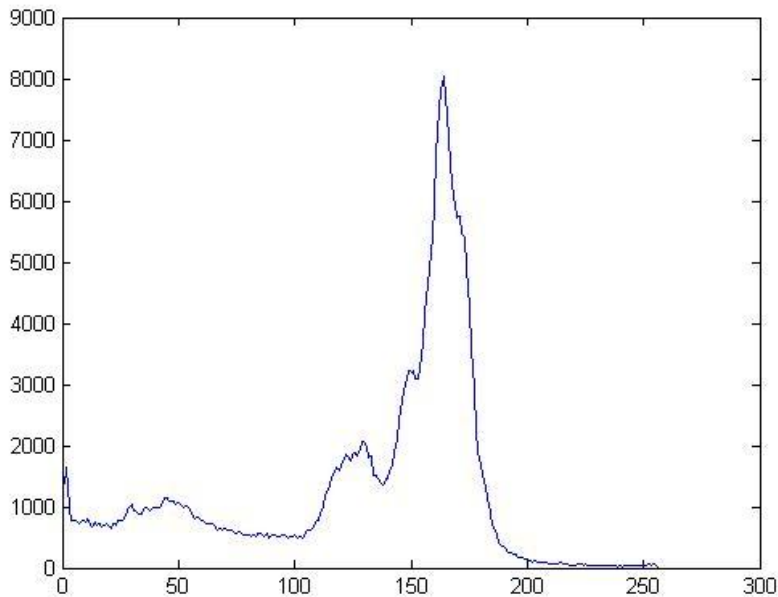


604x527 pixels



# Предобработка

- Первый шаг – удалить шум/сгладить гистограмму



# Пороговая фильтрация

---

- Выбрать порог  $T$  из диапазона  $[0,255]$ .
- ✓ Если  $I(x,y) > T$  – белый (объект);
  - ✓ В противном случае – черный.
  
- ✓ Как выбрать порог?
- ✓ Простейший случай: вручную, пользователем.

# Выбор порога «вручную»

---



$T = 0.25$



$T = 0.5$



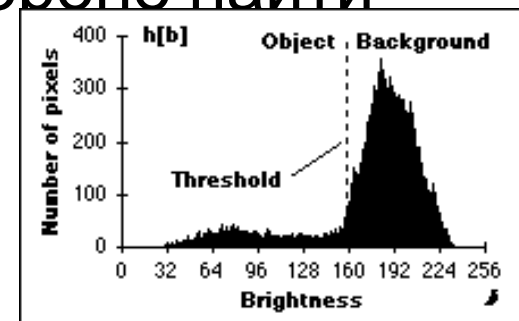
$T = 0.6$

# Анализ симметричного пика гистограммы

1. Вычислить гистограмму.
2. Найти  $(h_{max}, i_{max})$ .
3. На противоположной (от конца) стороне найти

1.  $(h_p, i_p)$ , который

$$\sum_{i=i_p}^{255} h_i \geq p\% \text{ от } \sum_{i=i_{max}}^{255} h_i$$



4.  $T = i_{max} - (i_p - i_{max})$

- Применение: фон изображения ярко выражен, симметричен относительно своего максимума.

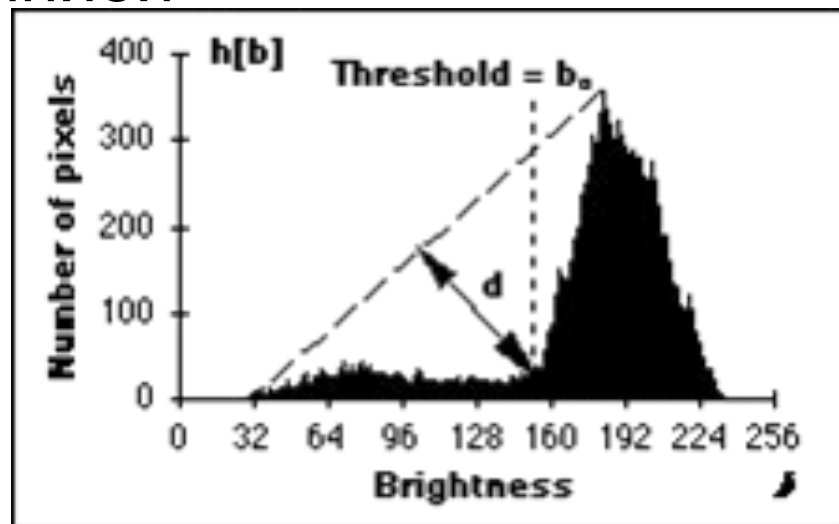
# Пример

---



# Метод треугольника

1. Вычислить сглаженную гистограмму.
2. Найти  $(h_{\max}, i_{\max})$ .
3. На левой стороне гистограммы найти  $(h_{\min}, i_{\min})$  при том, что  $h_{\min} \neq 0$
4. Найти  $(h_d, i_d)$  - с максимальным расстоянием  $d$  между  $(h_d, i_d)$  точкой и линией  $(h_{\max}, i_{\max}) - (h_{\min}, i_{\min})$ .
5.  $T = i_d$ .



# Пример

---





# Метод Отсу

- Метод Оцу находит порог, который уменьшает дисперсию в классе, которая определяется как взвешенная сумма дисперсий двух классов

$$\sigma_w^2(t) = \omega_1(t)\sigma_1^2(t) + \omega_2(t)\sigma_2^2(t)$$

- $\omega_i$  – вероятность попадания в первый класс при пороге  $t_i$ ;
- $\sigma_i^2$  – разброс значений каждого из двух классов (variance)
- То е самое, что и

$$\sigma_b^2(t) = \sigma^2 - \sigma_w^2(t) = \omega_1(t)\omega_2(t) [\mu_1(t) - \mu_2(t)]^2$$

- $\mu_i$  Среднее класса

# Пример

---



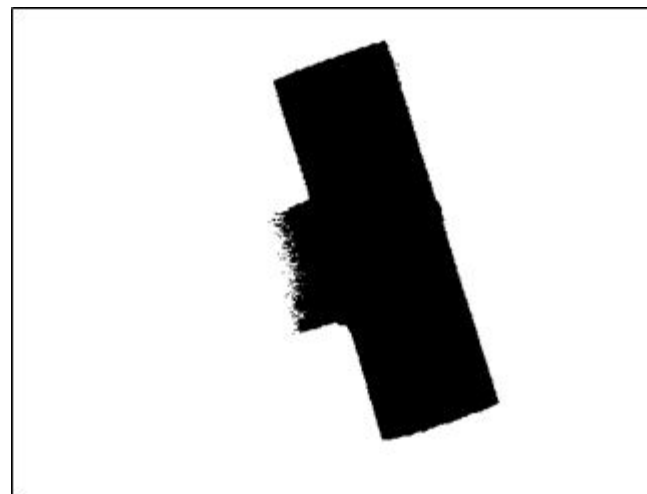
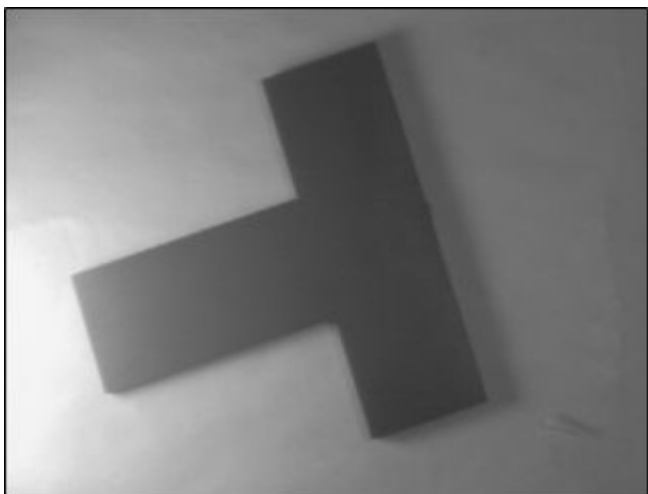
# Адаптивная фильтрация

---

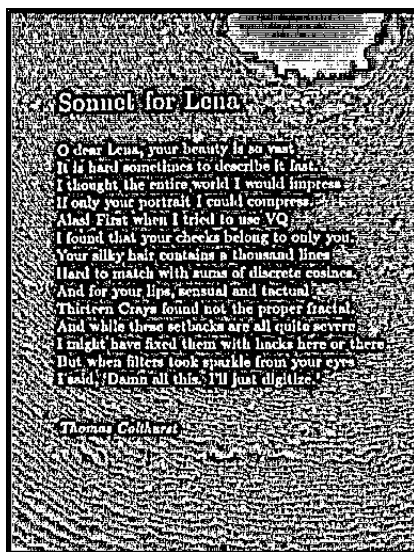
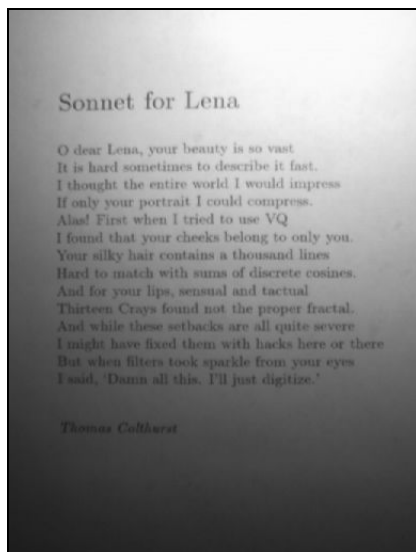
- Для каждого пикселя  $I(x,y)$ :
  1. Вычислить порог  $T$ , используя интенсивность соседей в радиусе  $R$ .
  2. Если  $\text{abs}(I(x,y)-T) > C$ , - белый. Иначе - черный
  
- Как вычислить  $T$ ?
  1. Среднее;
  2. Медиана;
  3.  $T = (I_{\min} + I_{\max}) / 2$ .
  4. Любое среднее (см. шумоподавление).

# Пример

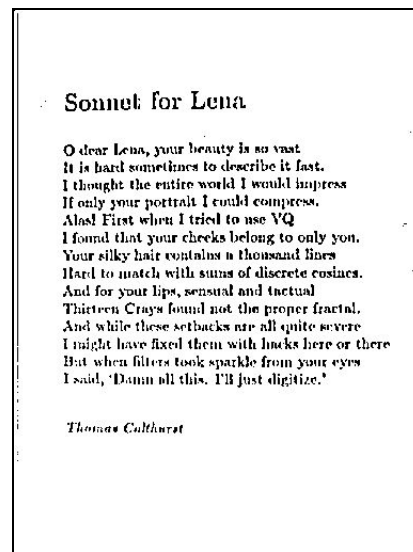
---



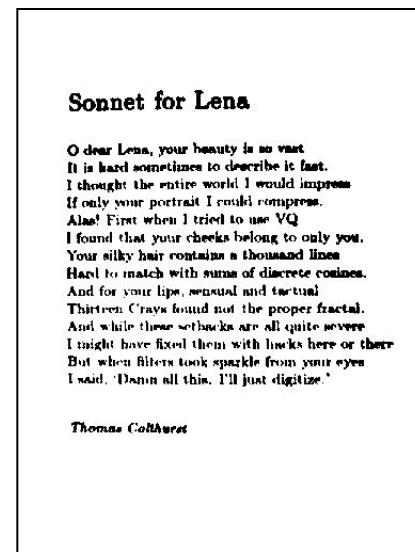
# Пример



R=7, C=0



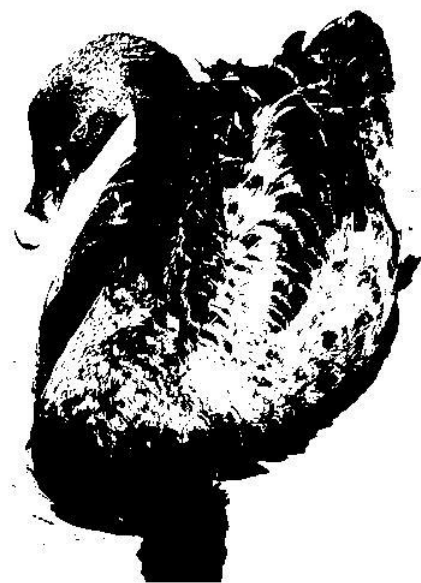
R=7, C=7



R=75, C=10

# Пример

---



Mean



Median



$(I_{max}+I_{min})/2$



УНИВЕРСИТЕТ  
ЛОБАЧЕВСКОГО

Институт информационных технологий, математики и  
механики  
Кафедра математического обеспечения и  
суперкомпьютерных технологий

Обработка изображений

## Lecture 3. Basic Binary Images Analysis

Svetlana Nosova  
svetlana.nosova@itmm.unn.ru

# What is a binary image?

---

- Binary images are images whose pixels have only two possible intensity values.
- They are normally displayed as black and white. Numerically, the two values are often 0 for black, and either 1 or 255 for white.
- Binary images are often produced by thresholding a grayscale or color image, in order to separate an object in the image from the background



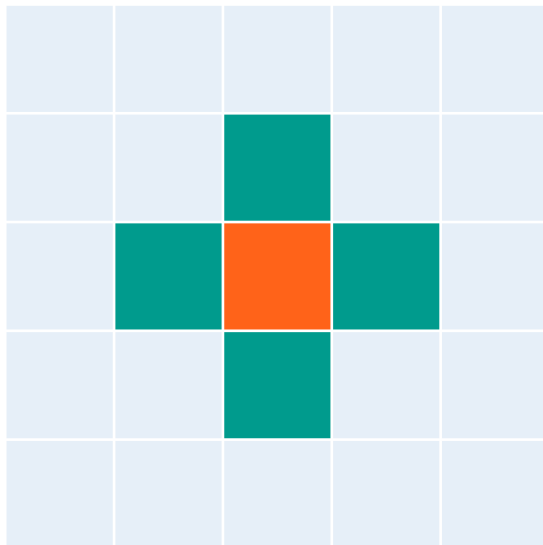
# Binary image analysis task

---

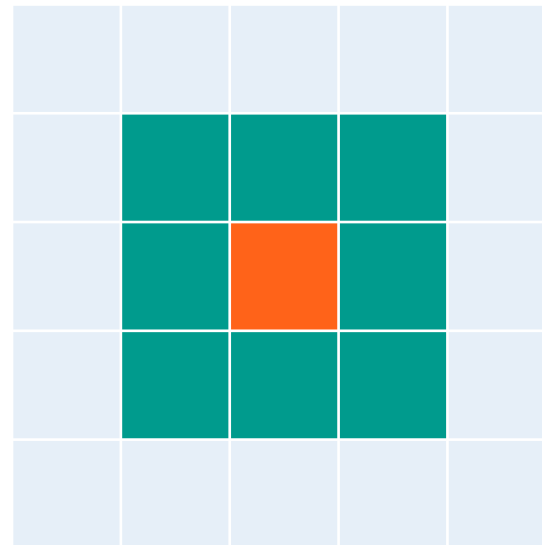
- Noise suppression
- Run-length encoding
- Component labeling
- Contour extraction
- Medial axis computation
- Thinning
- Filtering (morphological operations)
- Feature extraction (size, orientation etc.)

# Pixel Neighborhoods

---



four-neighborhood  
 $d$   
N4



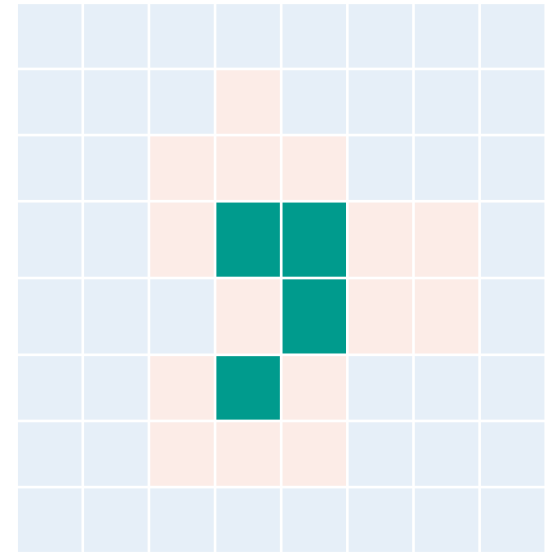
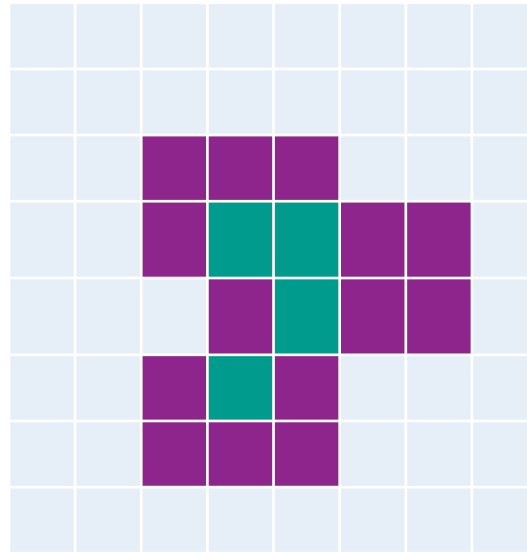
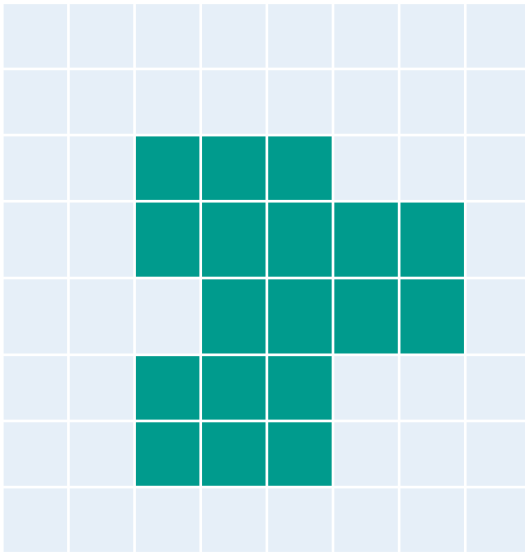
eight-neighborhood  
 $d$   
N8

# Basic morphological operations

Erosion, Dilation, Opening, Closing

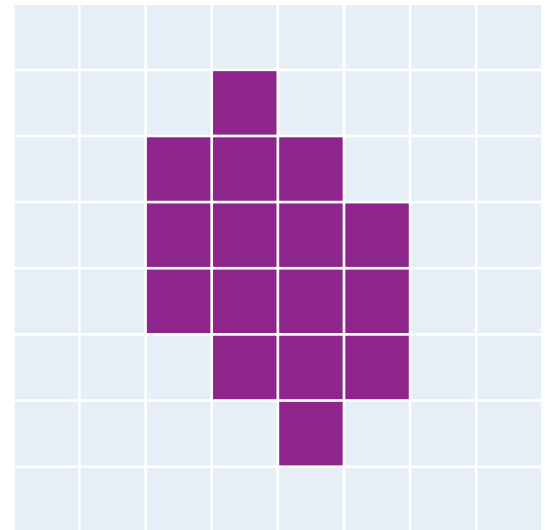
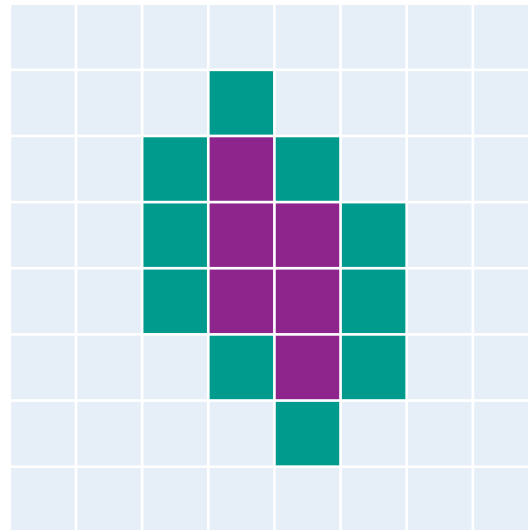
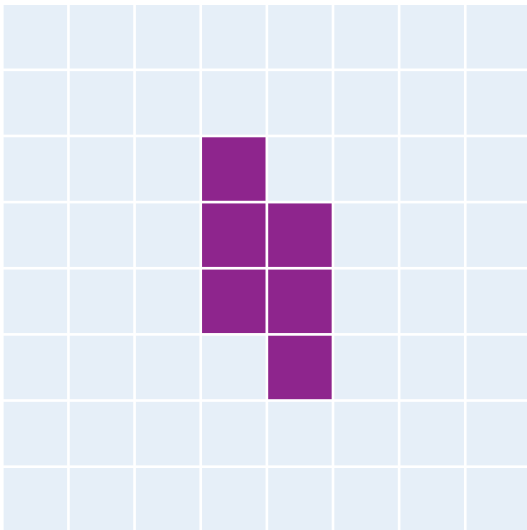
# Erosion Example

- Change a foreground pixel to background if it has a background pixel as a 4-neighbor.



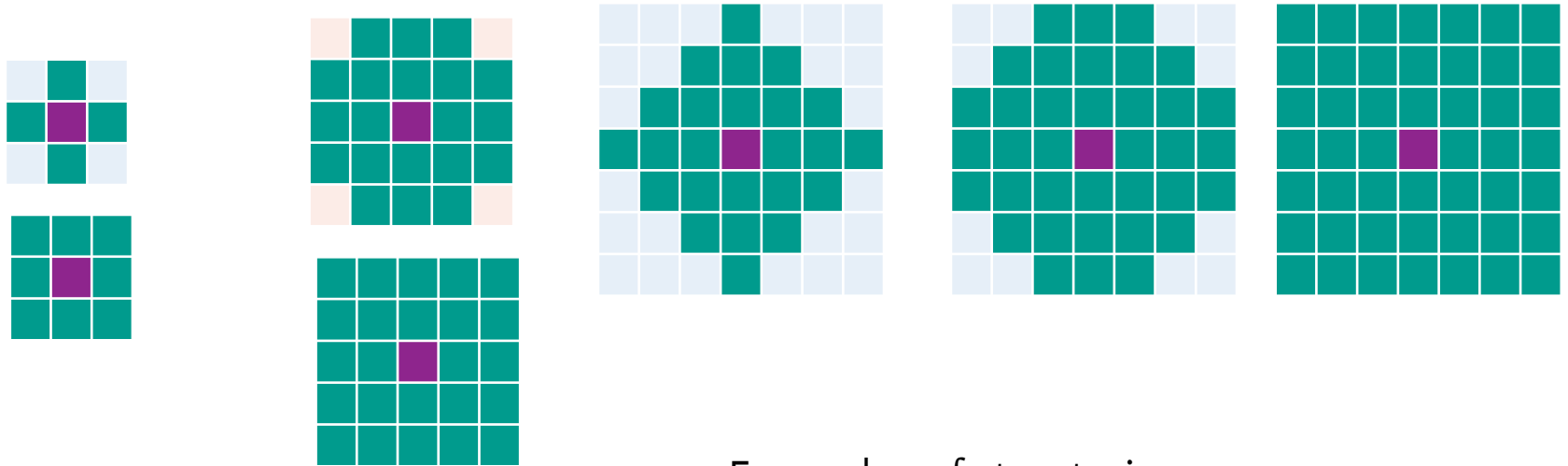
# Dilation Example

- Change a background pixel to foreground if it has a foreground pixel as a 4-neighbor.



# Structuring element

- A structuring element is simply a binary image (or mask) that allows us to define arbitrary neighborhood structures.



Examples of structuring elements

# Binary Images as Sets

---

- We can think of a binary image  $I(u, v)$  as the set of all pixel locations in the foreground:

$$Q_I = \{(u, v) \mid I(u, v) = 1\}$$

- To simplify notation, we'll use a single variable for a coordinate pair,  $p = (u, v)$ .

$$Q_I \{p \mid I(p) = 1\}$$

# Set Operations = Point Operations

---

## □ Complement = Inversion

- Let  $I$  denote image inversion (pointwise NOT)

$$Q_{\bar{I}} = \bar{Q}_I = \{p \in Z^2 \mid p \notin Q_I\}$$

## □ Union = OR

- Let  $I_1 \vee I_2$  be pointwise OR operation

$$Q_{I_1 \vee I_2} = Q_{I_1} \cup Q_{I_2}$$

## □ Intersection = AND

- Let  $I_1 \wedge I_2$  be pointwise AND operation

$$Q_{I_1 \wedge I_2} = Q_{I_1} \cap Q_{I_2}$$



# More Image Operations

---

- Use  $I$  to denote the set
- Translation

$$d \in \mathbb{Z}^2; I_d = \{(p + d) \mid p \in I\}$$

- Reflection:

$$I^* = \{-p \mid p \in I\}$$

# Dilation

---

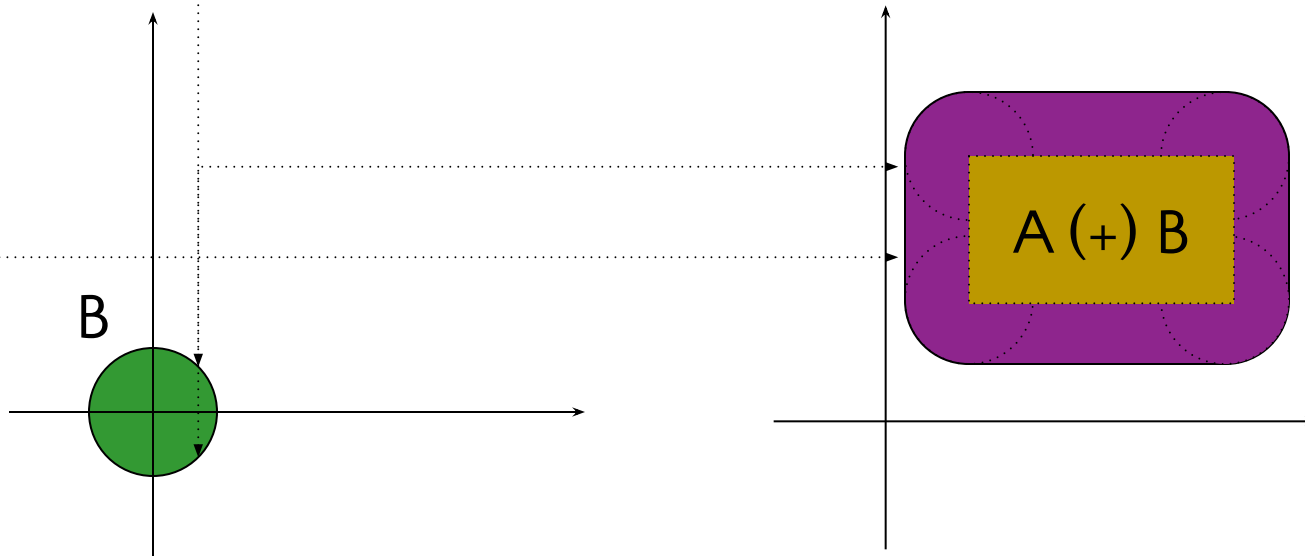
- A **dilation** of an image  $I$  by the structure element  $H$  is given by the set operation

$$I \oplus H = \{(p + q) \mid p \in I, q \in H\}$$

- Alternative definition: Take the union of copies of the structuring element,  $H_p$ , centered at every pixel location  $p$  in the foreground:

$$I \oplus H = \bigcup_{p \in I} H_p$$

# Dilation



# Dilation Algorithm

---

- Uses equivalent formula  $I \oplus H = \bigotimes_{q \in H} I_q$
- Input: Image  $I$ , structuring element  $H$
- Output: Image  $I' = I \oplus H$ 
  1. Start with all-zero image  $I$
  2. Loop over all  $q \in H$ 
    1. Compute shifted image  $Iq$
    2. Update  $I' = I' \vee I_q$

# Erosion

---

- A erosion of an image  $I$  by the structure element  $H$  is given by the set operation

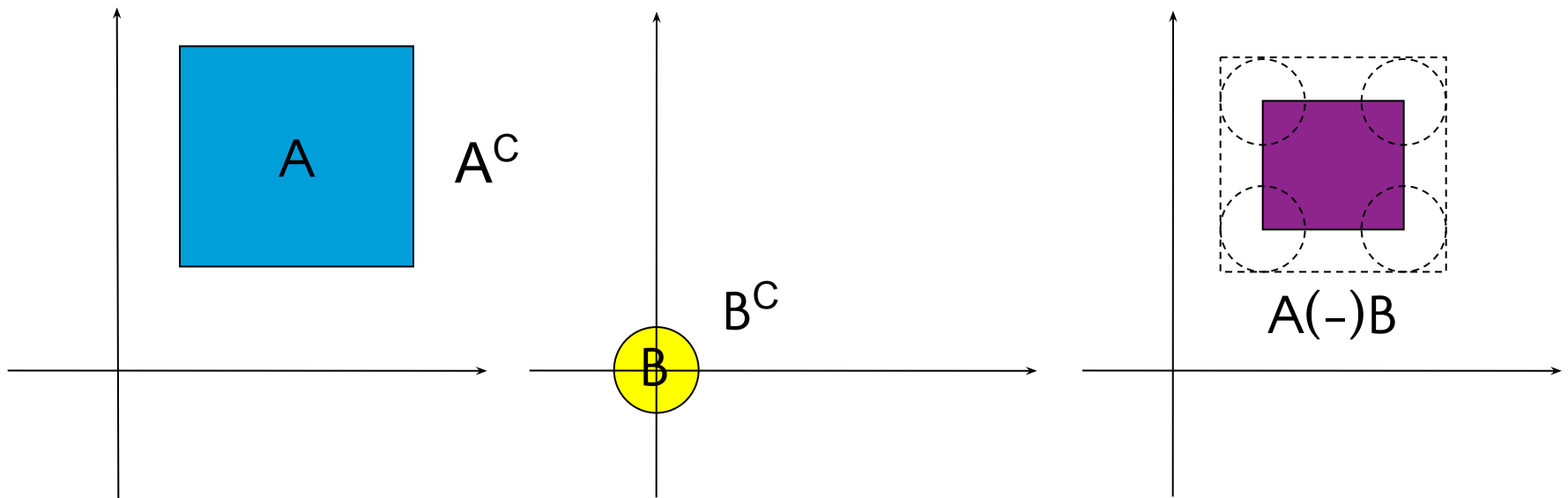
$$I(-)H = \{p \in Z^2 \mid (p + q) \in I, \text{ for every } q \in H\}$$

- Alternative definition: Keep only pixels  $p \in I$  such that  $H_p$  fits inside  $I$ :

$$I(-)H = \{p \mid H_p \subseteq I\}$$

# Erosion

---



# Duality of Erosion and Dilation

---

- Erosion can be computed as a dilation of the background:

$$I(-)H = \overline{\bar{I} \oplus H^*}$$

- Same duality for dilation:

$$I \oplus H = \overline{\bar{I}(-)H^*}$$

# Erosion Algorithm

---

- Uses dual  $I(-)H = \overline{\bar{I} \oplus H^*}$
- Input: Image  $I$ , structuring element  $H$
- Output: Image  $I' = I(-)H$ 
  1. Start with inversion  $I' = \bar{I}$
  2. Dilate  $I'$  with reflected structure element,  $H^*$
  3. Invert  $I'$



# Properties of Dilation

---

## □ Commutativity:

□ Means we can switch the roles of the structuring element and the image

□ Associativity:  $I \oplus H = H \oplus I$

□ Means that we can sometimes break up a big structuring element into smaller ones:

□ If  $I_1 \oplus (I_2 \oplus I_3) = (I_1 \oplus I_2) \oplus I_3$

□ Then

$$H = H_1 \oplus H_2 \oplus \dots \oplus H_n$$

$$I \oplus H = (\dots((I \oplus H_1) \oplus H_2) \oplus \dots \oplus H_n)$$

# Properties of Erosion

---

- It is NOT commutative:

$$I(-)H \neq H(-)I$$

- It is NOT associative, but:

$$(I(-)H_1)(-)H_2 = I(-)(H_1 \oplus H_2)$$

# Some Particular Dilation Operators

---

- **Identity:**  $id = \{(0, 0)\}$ 
  - $I \oplus id = id \oplus I = I$
- **Shift by  $k$  pixels in  $x$ :**  $S_x = \{(k, 0)\}$
- **Shift by  $k$  pixels in  $y$ :**  $S_y = \{(0, k)\}$

# Opening

---

- Stray foreground structures that are smaller than the H structure element will disappear. Larger structures will remain.

$$I \boxtimes H = (I(-)H) \oplus H$$

# Closing

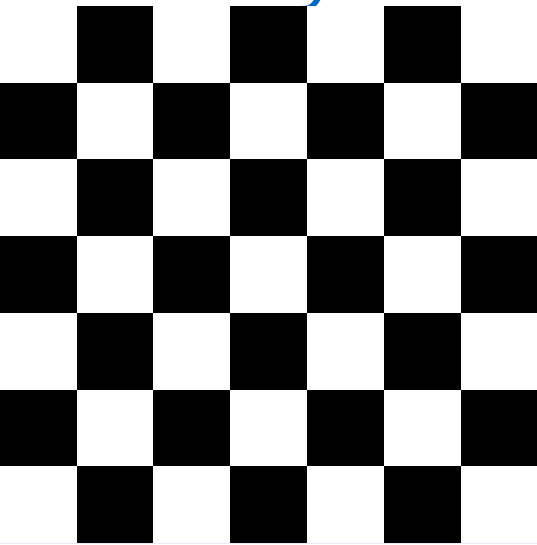
---

- Holes in the foreground that are smaller than  $H$  will be filled.

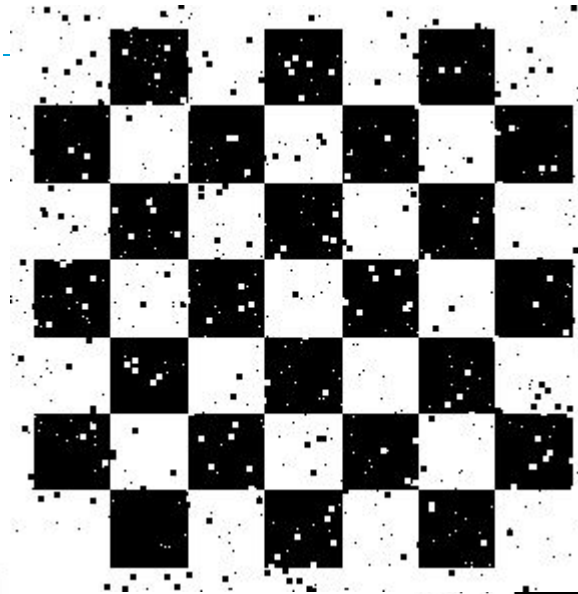
$$I \bullet H = (I \oplus H)(- )H$$



# Binary noise

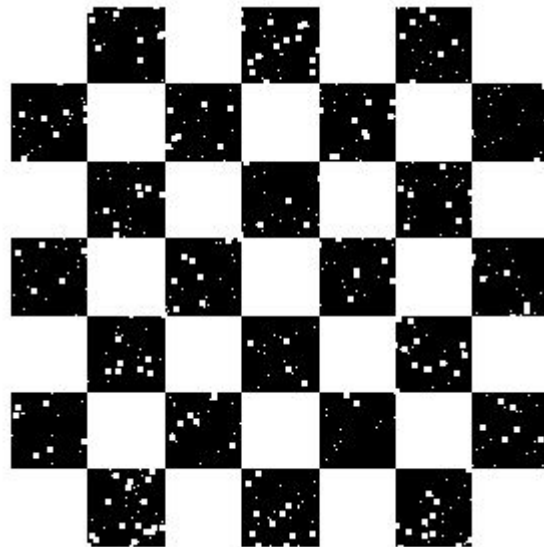


Original  
image

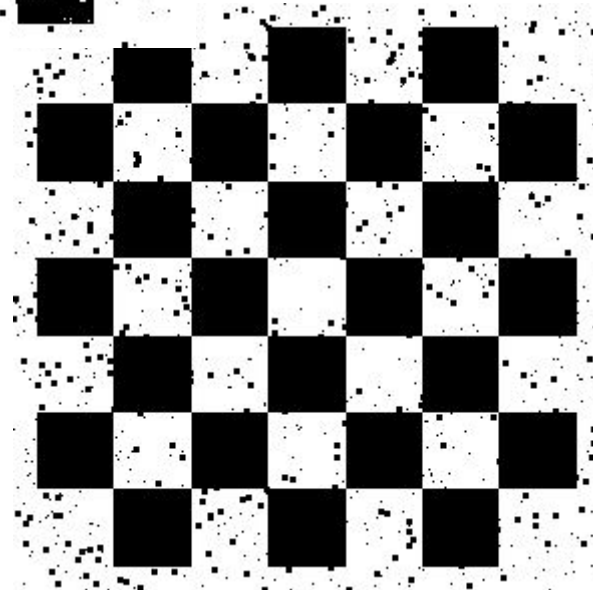


Salt  
and  
pepper  
noise

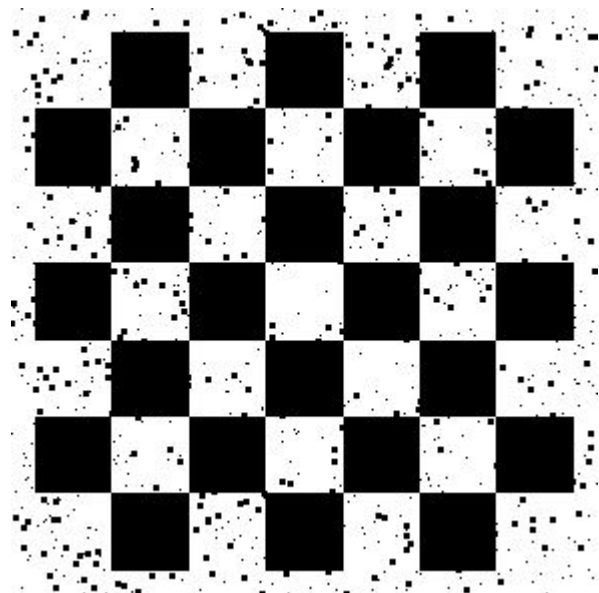
WHITE  
NOISE



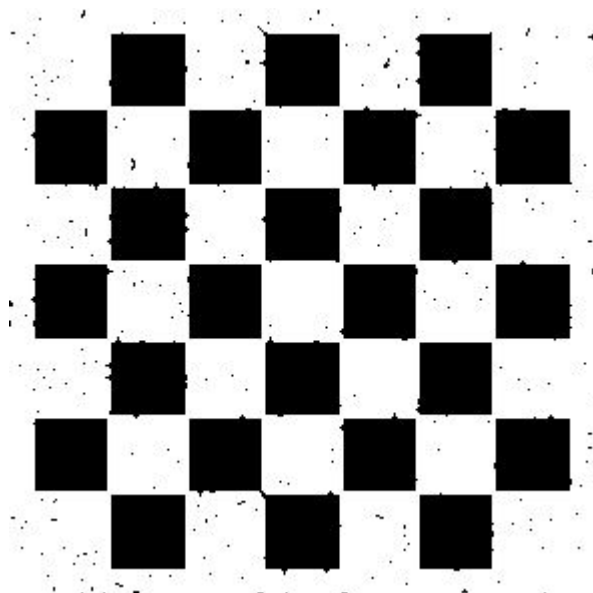
BLACK  
NOISE



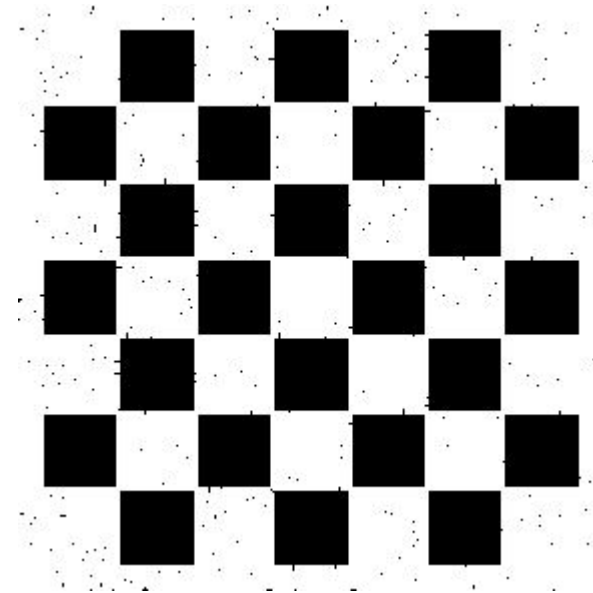
# Erosion results



Original: black  
noise



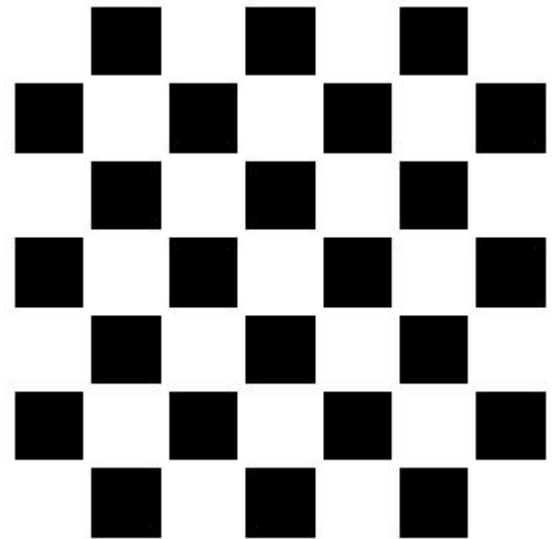
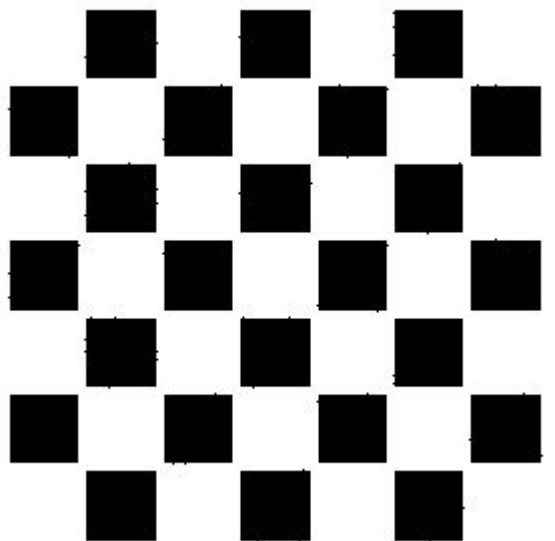
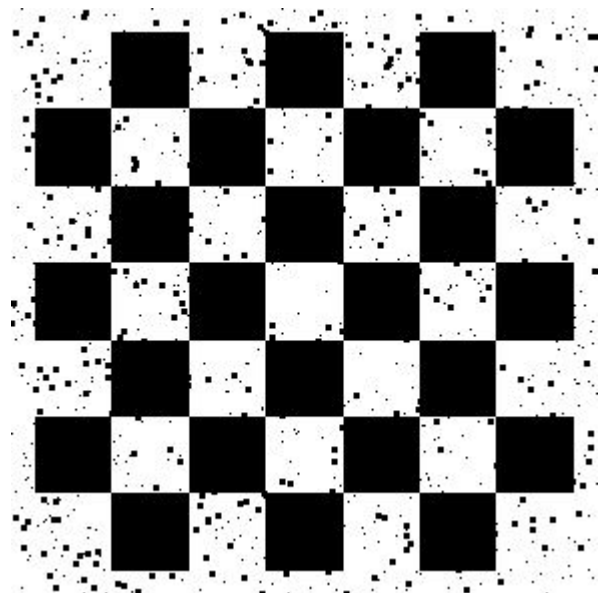
mask



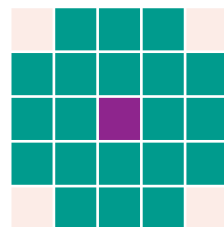
mask



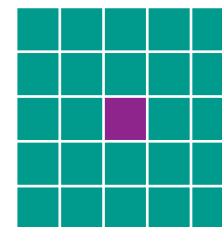
# Erosion results



Original: black  
noise

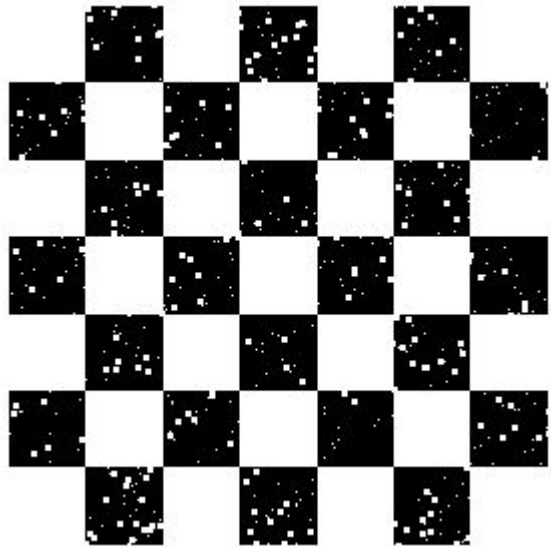


mask

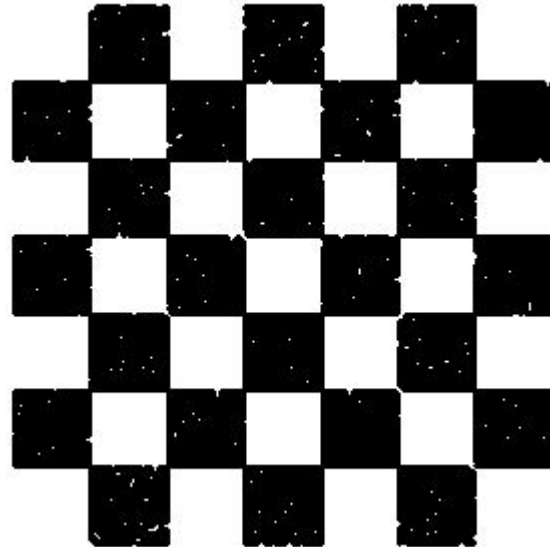


mask

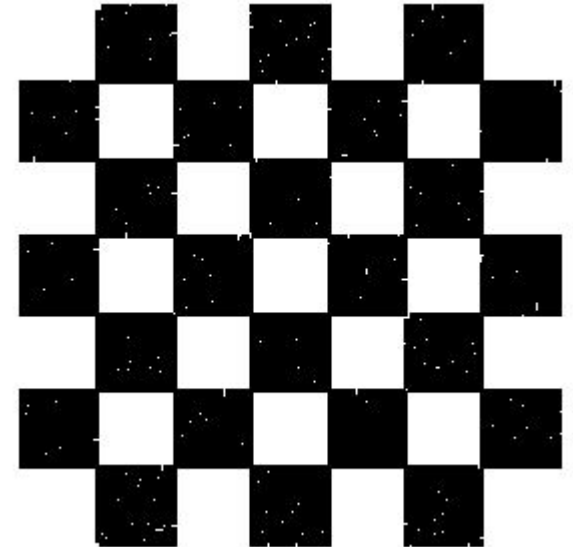
# Dilation results



Original: white noise

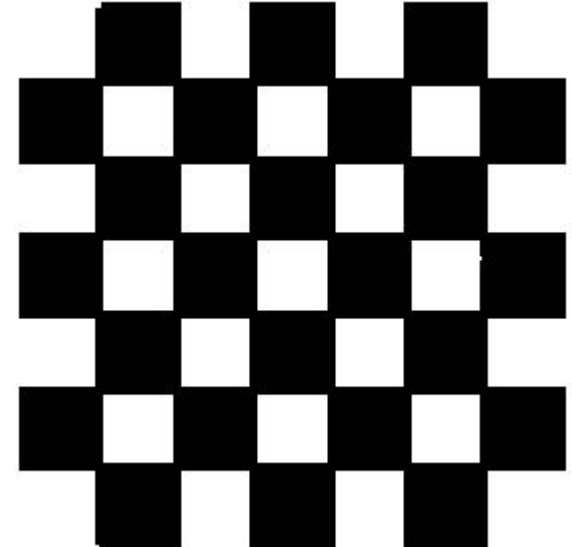
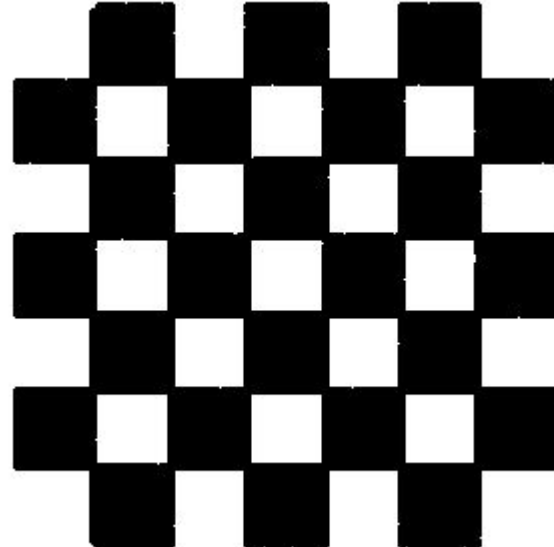
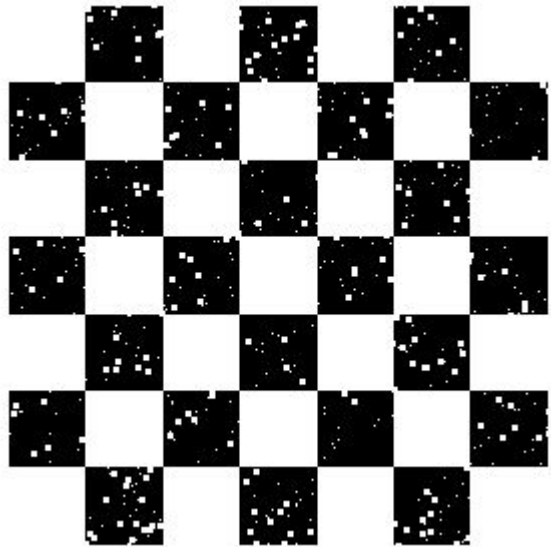


mask

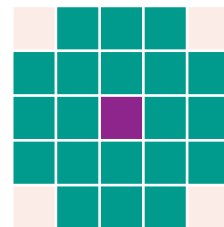


mask

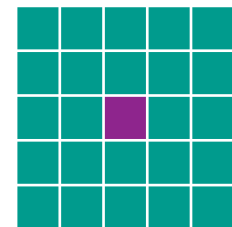
# Dilation results



Original: white noise



mask

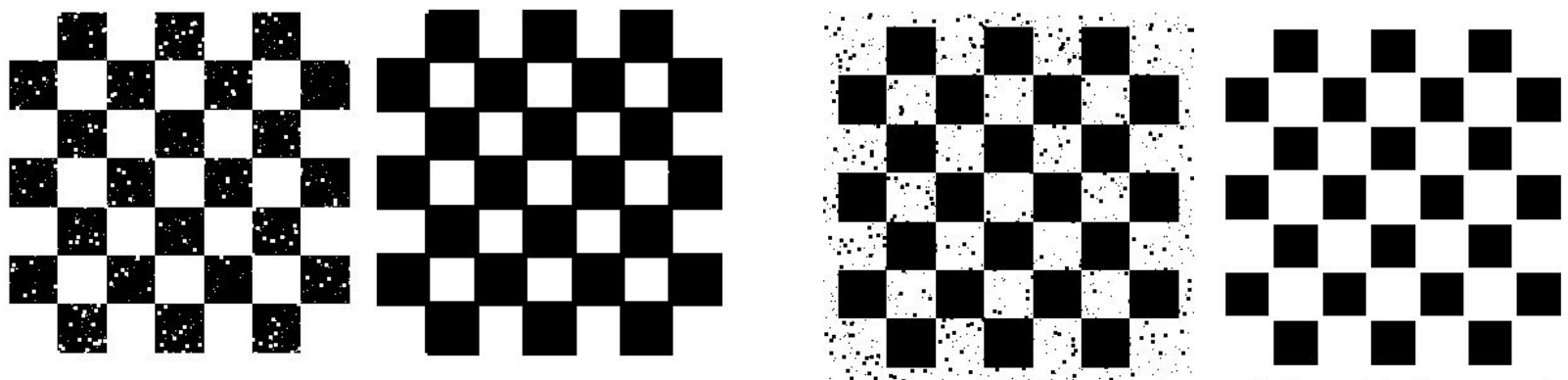


mask

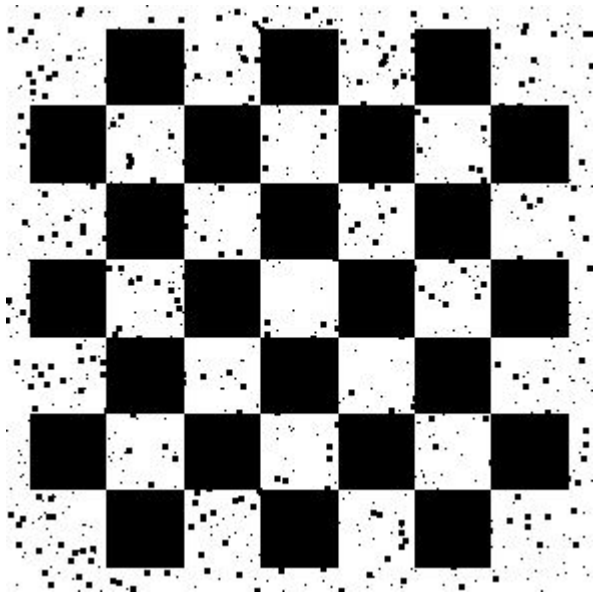
# Problems

---

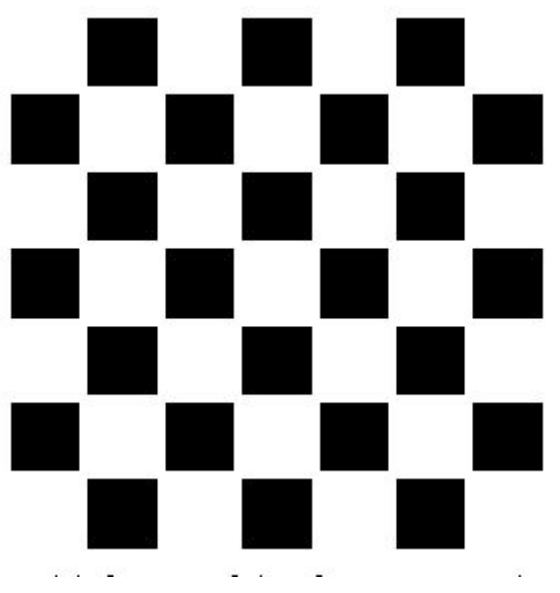
- You see that foreground size is changed after erosion or dilation. If you want to save original size, you should use opening or closing operations.



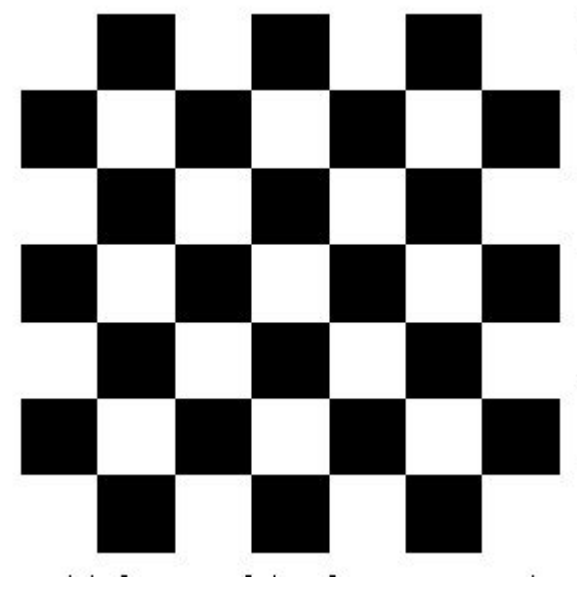
# Closing results



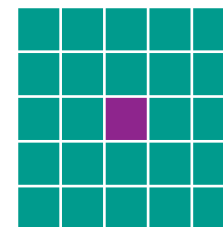
Original: black noise



Dilation

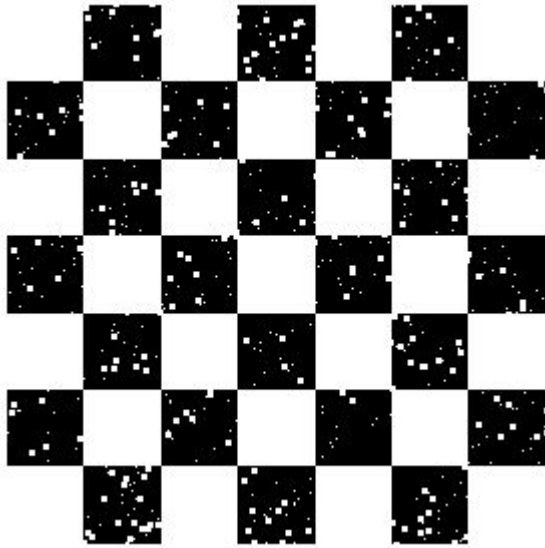


Closing

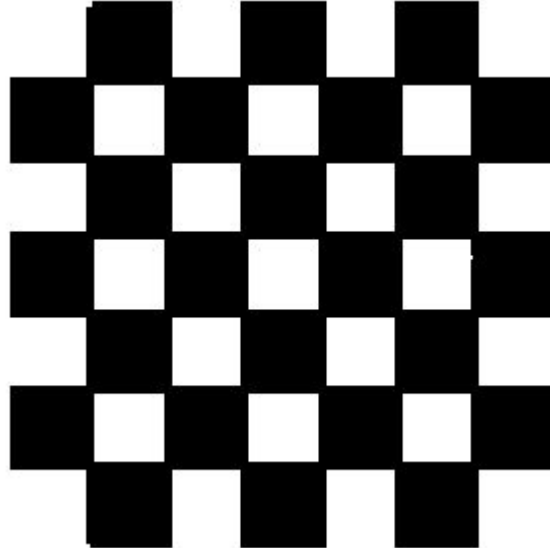


mask

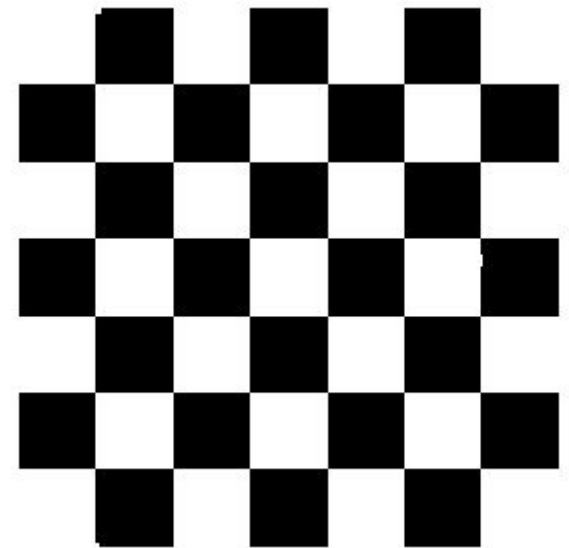
# Opening results



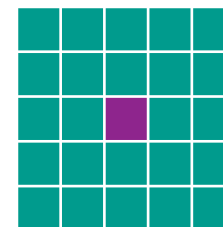
Original: white noise



Erosion



Opening



mask

# Other examples: opening



$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & [1] & 1 \\ 0 & 1 & 0 \end{bmatrix}$$



$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & [1] & 1 \\ 1 & 1 & 1 \end{bmatrix}$$



$$\begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & [1] & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix}$$

# Other examples: closing



$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & [1] & 1 \\ 0 & 1 & 0 \end{bmatrix}$$



$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & [1] & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

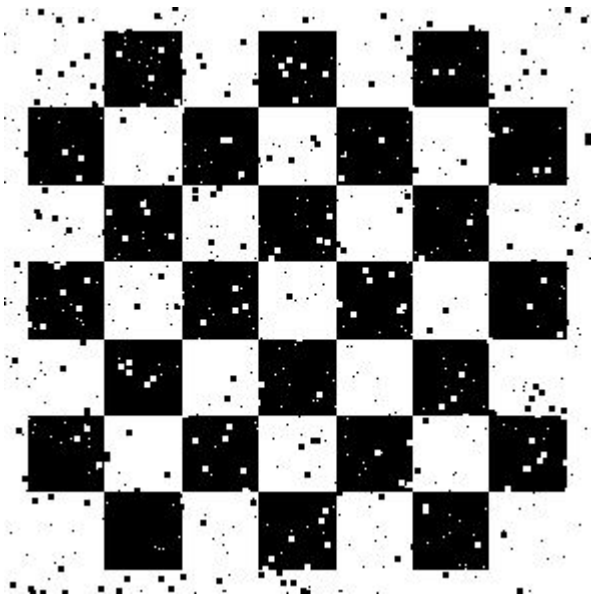
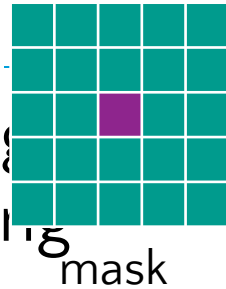


$$\begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & [1] & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix}$$

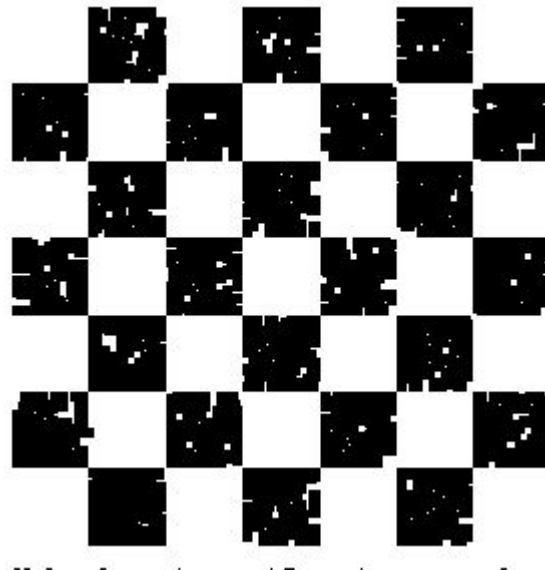


# Salt and pepper noise

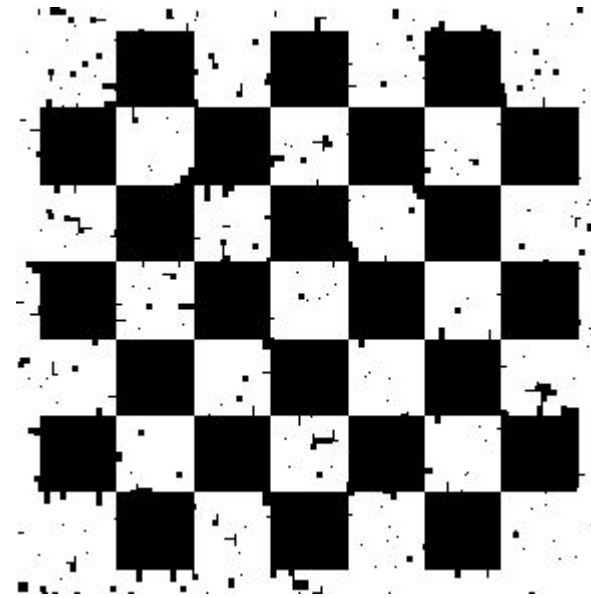
- We can't delete salt pepper noise from binary image using only math. morphology operations with saving foreground size and form.



Original:  
Salt and pepper  
noise



Closin



Openin

What can we do? See on later slides.



# Boundary extraction

---

- Internal border

$$ib(I) = I - (I(-)H)$$

- External border

$$eb(I) = (I \oplus H) - I$$

# Boundary extraction

---



**Hello,  
students!**

Original  
|



**Hello,  
students!**

Internal  
border



**Hello,  
students!**

External  
border

# Other morphological operations

- For color images too.



Source



Erosion  $I(-)B$



Dilatation  $I\oplus B$



Opening  $I\ominus B=(I(-)B)(+)B$



Closing  $I\bullet B=(I(+)B)(-)B$



Grad(I) =  $(I(+)B)-(I(-)B)$



Top Hat (I) =  $I-(I(-)B)$



Black Hat (I) =  $(I(+)B)-I$



# Thinning

---

- thinning  $A$  symmetrically is based on a sequence of structuring elements:

$$\{B\} = \{B^1, B^2, \dots, B^n\}$$

- Where  $B^i$  are rotated version of  $B^{i-1}$
- The thinning by a sequence of SEs:

$$A(N)B = A - (A \otimes B)$$

$$A(N)\{B\} = (\dots((A(N)B^1)(N)B^2)\dots)(N)B^n$$

# Thinning

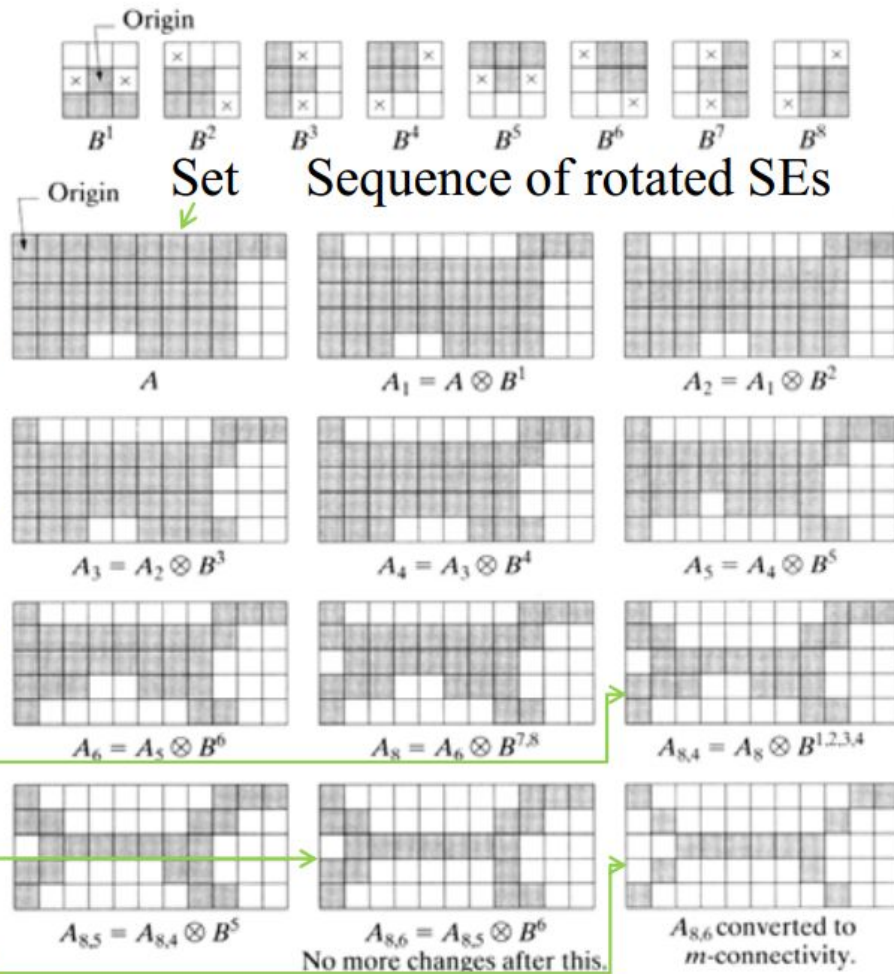
The process is to thin  $A$  by one pass with  $B^1$ , then thin the result with one pass of  $B^2$ , and so on, until  $A$  is thinned with one pass of  $B^n$ . The entire process is repeated until no further changes occur.

Results of thinning with each SE one after another

Using 4 first SEs again

Result after convergence

Conversion to  $m$ -connectivity







# Median filter

---

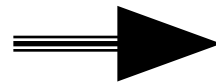
- In median filtering, the neighboring pixels are ranked according to brightness (intensity) and the median value becomes the new value for the central pixel.
- Median filters can do an excellent job of rejecting certain types of noise, in particular, “shot” or impulse noise in which some individual pixels have extreme values.
- In the median filtering operation, the pixel values in the neighborhood window are ranked according to intensity, and the middle value (the median) becomes the output value for the pixel under evaluation.

# Median Filter: example

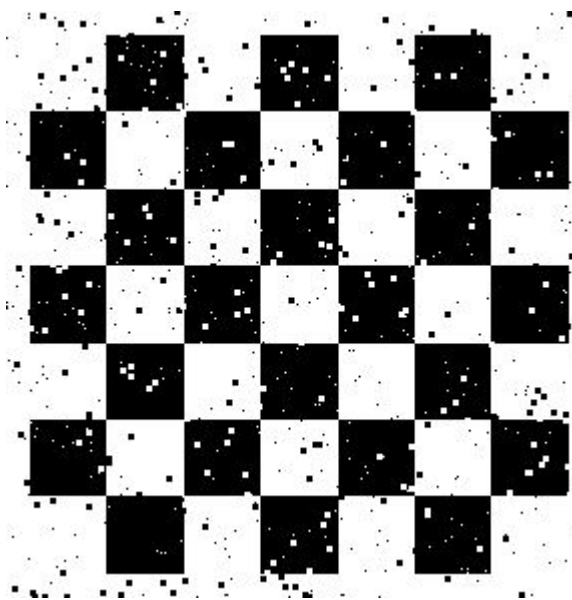
---



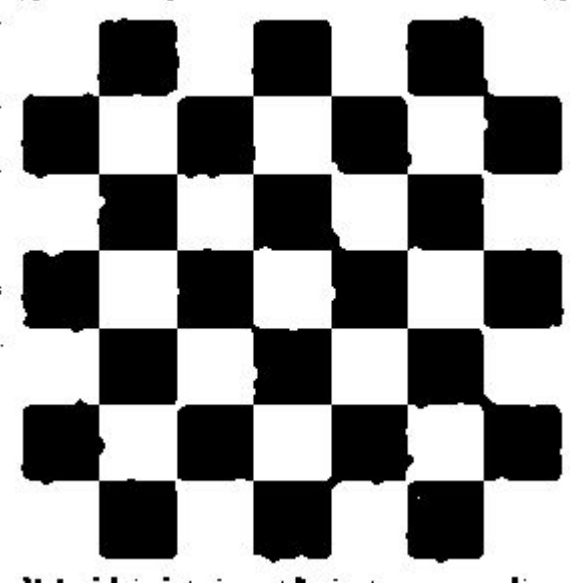
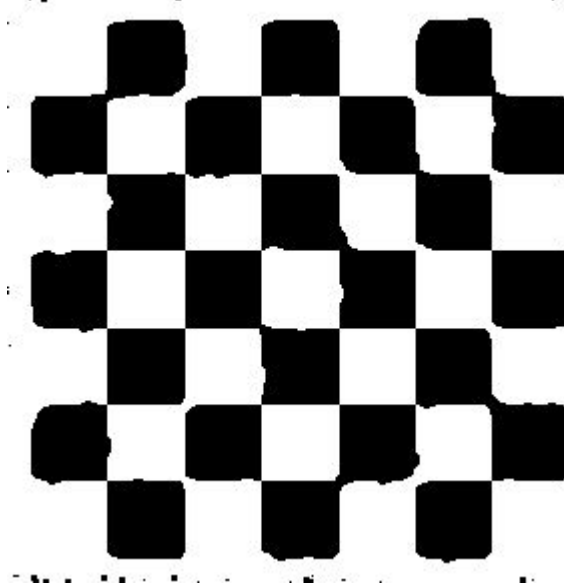
mask



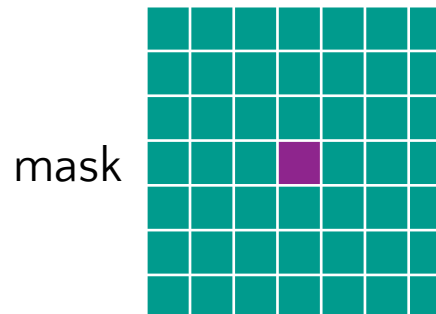
# Median filter: example



Source

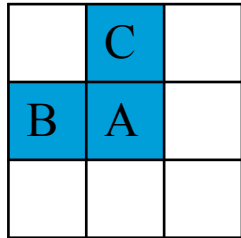


Result

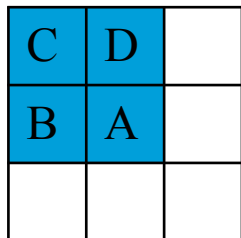




# Region growing (for N4 case)



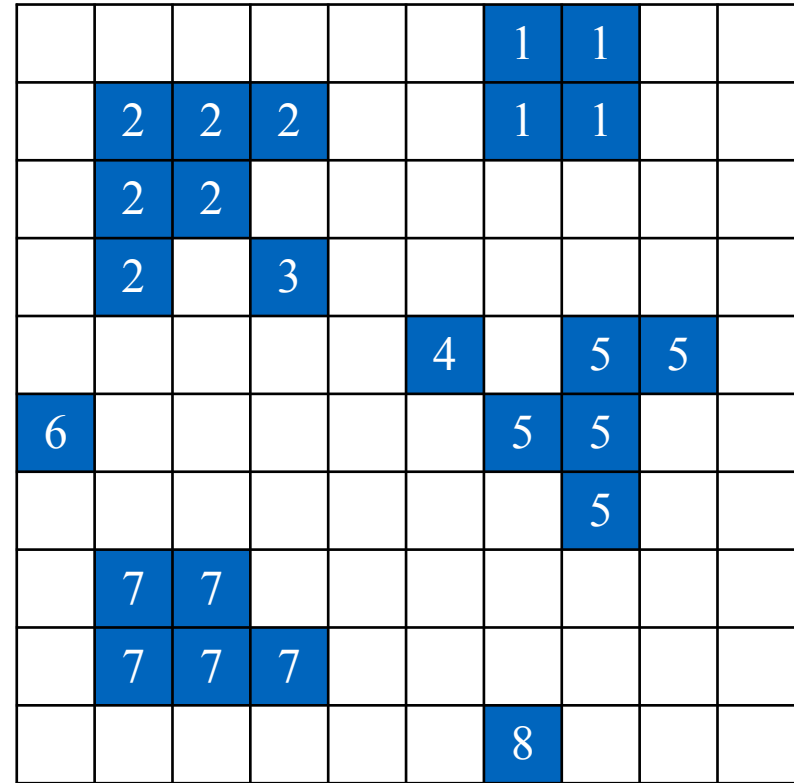
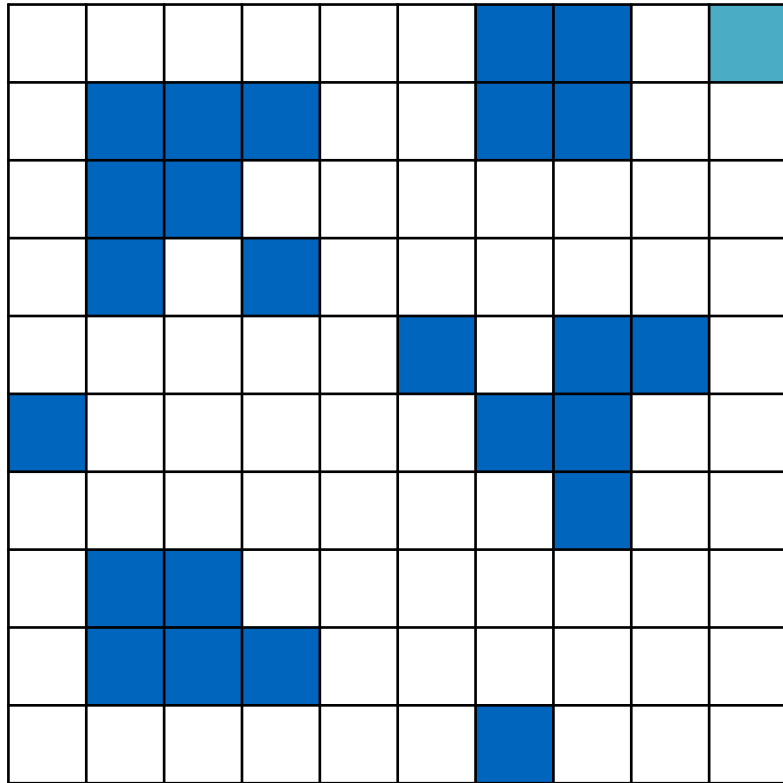
N4



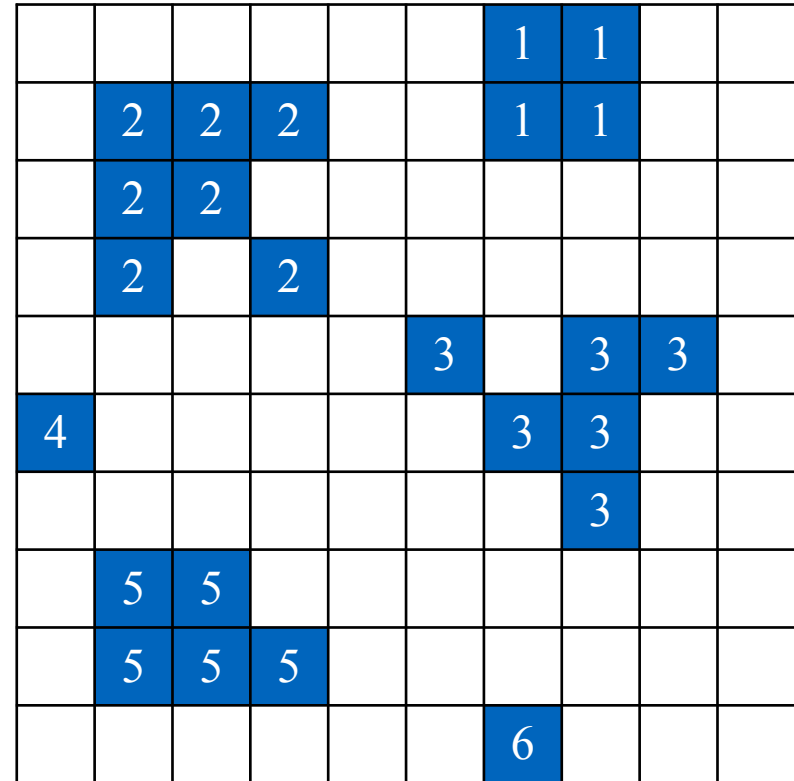
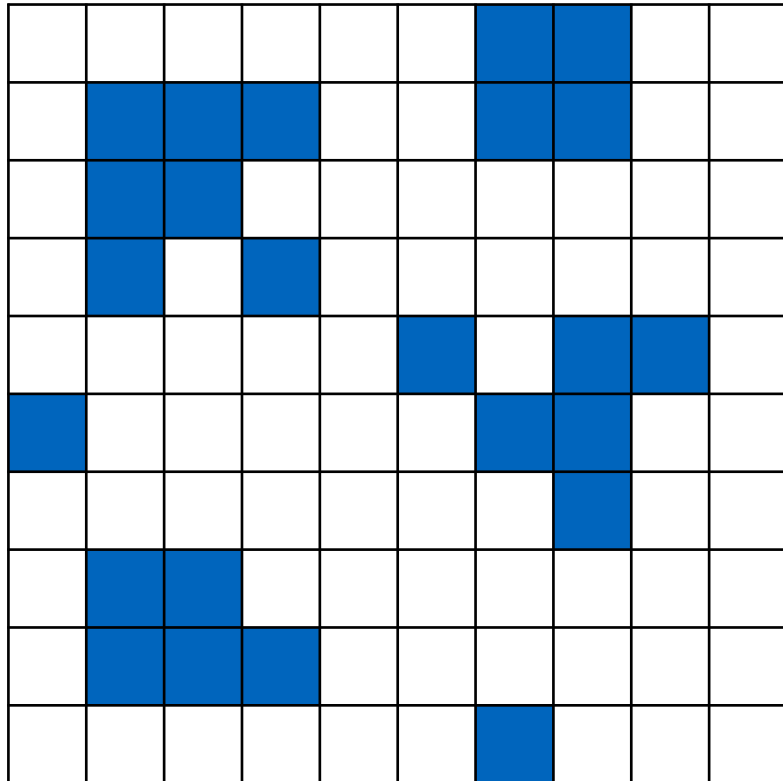
N8

- Current steps you should do for every pixel from top to bottom and from left to right.
- Create new region with A pixel if
  - $|I(A) - Clavg(B)| > \delta$  and  $|I(A) - Clavg(C)| > \delta$
- Add A to B region if
  - $|I(A) - Clavg(B)| \leq \delta$  and  $|I(A) - Clavg(C)| > \delta$
- Add A to C region if
  - $|I(A) - Clavg(B)| > \delta$  and  $|I(A) - Clavg(C)| \leq \delta$
- If  $|I(A) - Clavg(B)| \leq \delta$  and  $|I(A) - Clavg(C)| \leq \delta$ 
  - If  $|Clavg(B) - Clavg(C)| \leq \delta$   
merge B and C regions, add A to the result region.
  - $|Clavg(B) - Clavg(C)| > \delta$   
add A to the nearest region ( “nearest “ means nearest by mean intensity)
- Where:  $I(A)$  – pixel A intensity
- $Clavg(B)$ - mean intensity in B region

# Connected-component labeling: example



# Connected-component labeling: example







# Object parameters

---

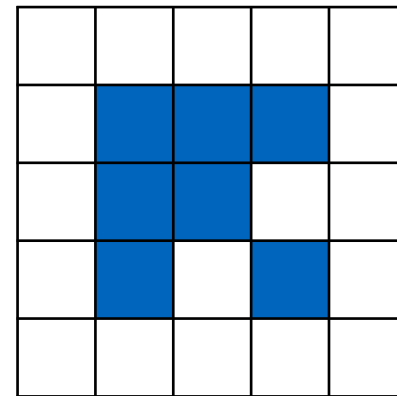
- Border characteristics Number of "holes" in
- The center of mass
- Perimeter
- Density
- Moments
- Orientation of the principal axes
- Color / brightness

# Area

---

For binary image

$$A = \sum_{x=0}^m \sum_{y=0}^n I(x, y)$$



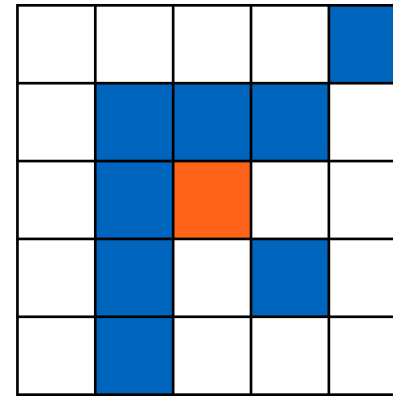
Result = 7

# The center of mass

---

$$\bar{x} = \frac{\sum_{x=0}^m \sum_{y=0}^n xI(x, y)}{A}$$

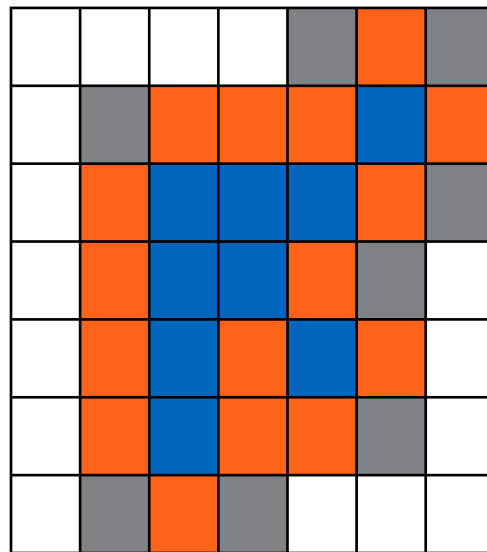
$$\bar{y} = \frac{\sum_{x=0}^m \sum_{y=0}^n yI(x, y)}{A}$$



Result

# Perimeter

- Border pixel count
- You can use both external or internal border.

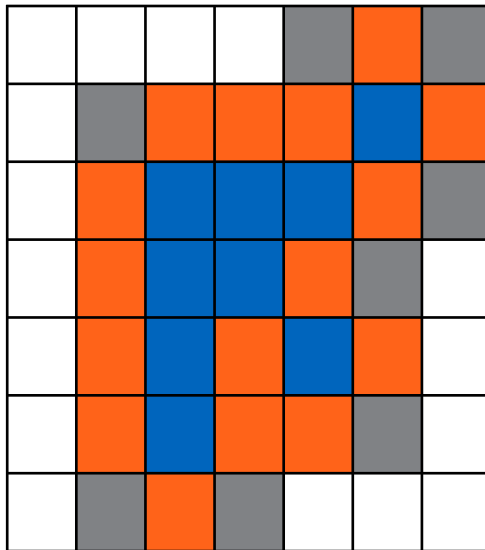


Result = 16; N4  
Result = 24; N8

# Compactness

$$C = \frac{P^2}{A}$$

The most compact shape is a circle,  $C = 4\pi$



Result = 28,4 ; N4

Result = 64; N8

# Moments

- Discrete moments

$$m_{ij} = \sum_{x,y \in S}^n x^i y^j B(x, y)$$

- Central moments

$$\mu_{pq} = \int \int (x - \bar{x})^p (y - \bar{y})^q B(x, y) d(x - \bar{x}) d(y - \bar{y})$$

- Hu moments

$$\phi_1 = \mu_{20} + \mu_{02}$$

$$\phi_2 = (\mu_{20} - \mu_{02})^2 + \mu_{11}^2$$

$$\phi_3 = (\mu_{30} - 3\mu_{12})^2 + (3\mu_{12} - \mu_{03})^2$$

$$\phi_4 = (\mu_{30} + \mu_{12})^2 + (\mu_{21} + \mu_{03})^2$$

⊠

# Moments: basic example

$$m_{ij} = \sum_{x,y \in S} x^i y^j B(x,y)$$

0	1	2	3	4	5
1					
2					
3					
4					
5					

<b>i</b>	<b>j</b>	<b>M<sub>ij</sub></b>
0	0	7
1	0	18
0	1	16
2	0	58
0	2	46
1	1	49

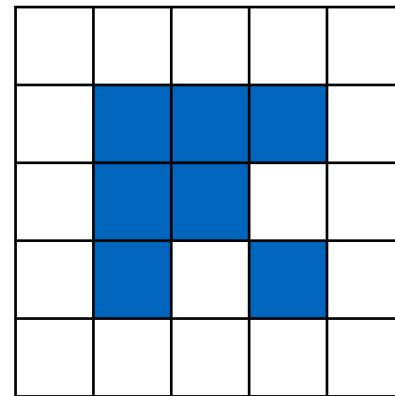


# Area

---

For binary image

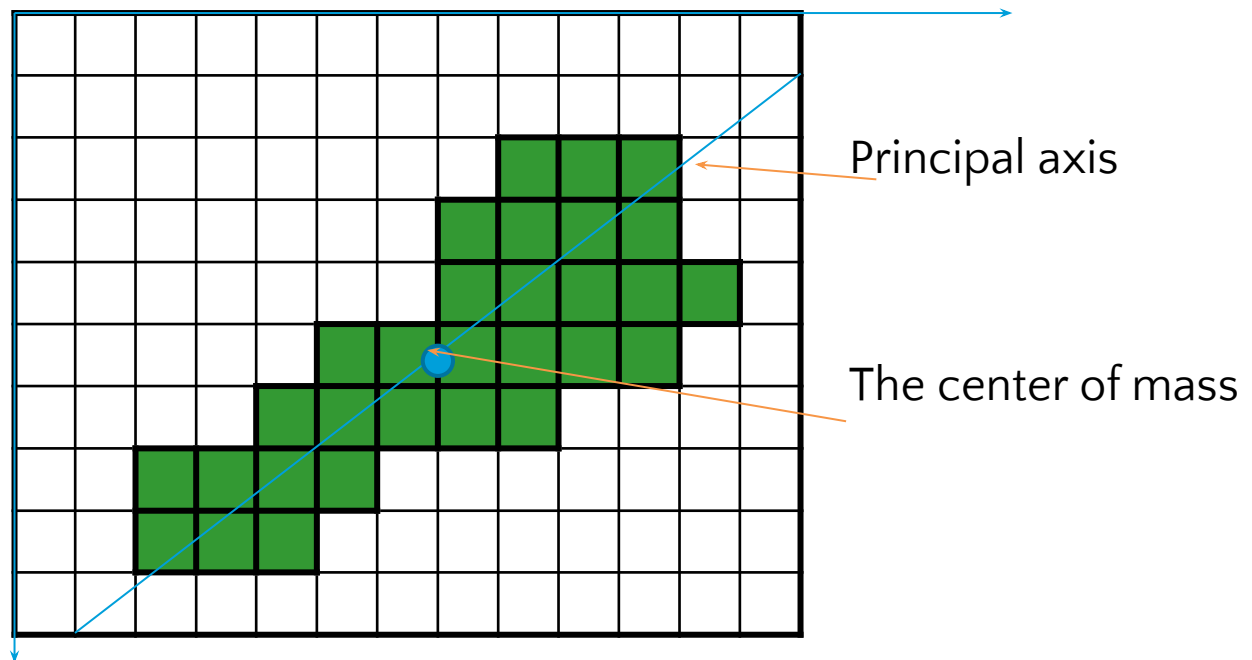
$$A = \sum_{x=0}^m \sum_{y=0}^n I(x, y)$$



Result = 7

# Principal axes orientation

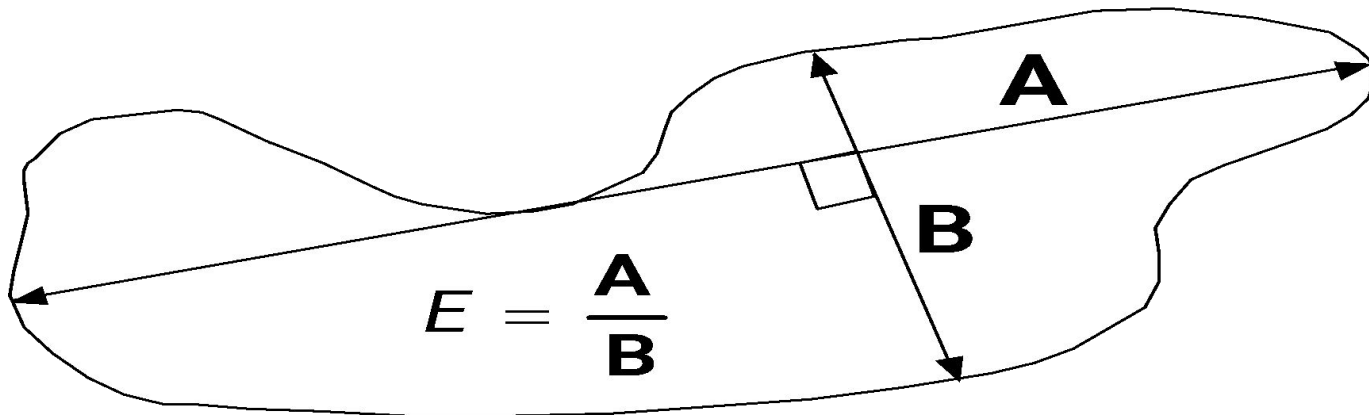
$$\theta = \frac{1}{2} \arctan \left( \frac{2m_{11}}{m_{20} - m_{02}} \right)$$



# Invariant area characteristics

---

$$\textit{elongation} = \frac{m_{20} + m_{02} + \sqrt{(m_{20} - m_{02})^2 + 4m_{11}^2}}{m_{20} + m_{02} - \sqrt{(m_{20} - m_{02})^2 + 4m_{11}^2}}$$



# Skeletons

The skeleton  $A$  can be expressed in terms of erosion and opening

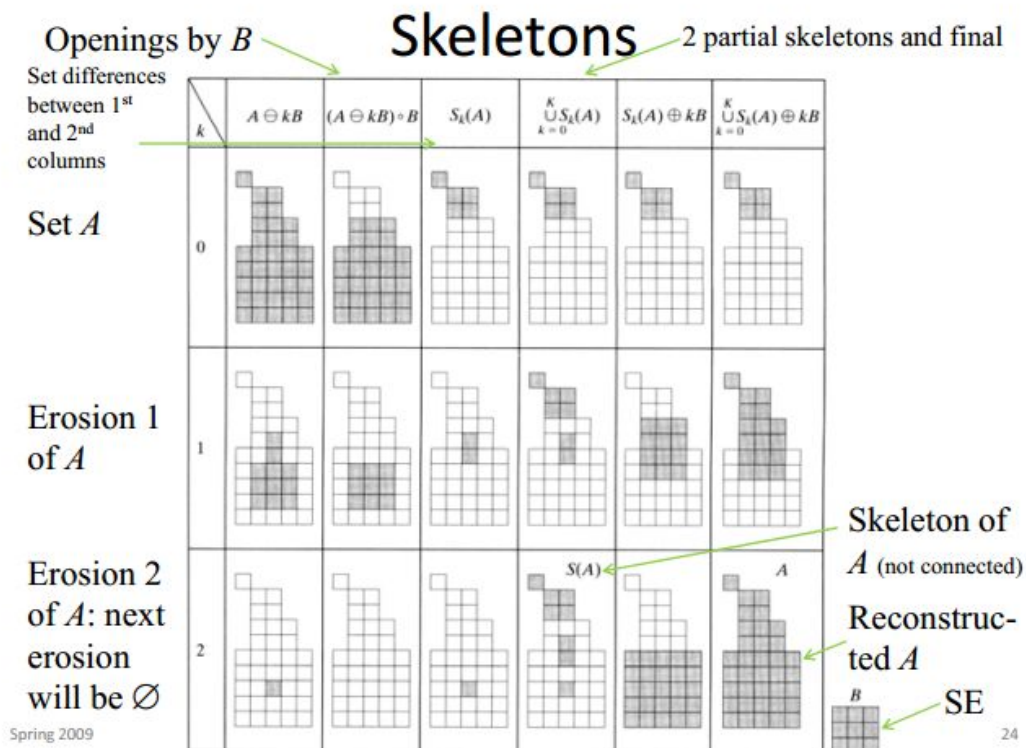
$$S(A) = \bigcap_{k=0}^K S_k(A)$$

$$S_k(A) = (A(-)kB) - (A(-)kB) \boxtimes B$$

$$A(-)kB = (\dots((A(-)B)(-)B)(-)\dots)(-)$$

$K$  is the last iterative step before  $A$  erodes to an empty set:

$$K = \max \{k \mid (A(-)kB) \neq \emptyset\}$$





**УНИВЕРСИТЕТ  
ЛОБАЧЕВСКОГО**

Институт информационных технологий, математики и  
механики  
Кафедра математического обеспечения и  
суперкомпьютерных технологий

**Обработка изображений**

**Выделение границ**

Светлана Носова  
svetlana.nosova@itmm.unn.ru

# Выделение границ

---

- Выделение границ – процесс поиска значимых переходов на изображении.
- Места, в которых происходит резкий перепад яркости, обычно образуют границу между различными объектами или частями сцены.
- Дальнейшая обработка ребер с целью определения линий, кривых и дуг окружности приводят к полезным результатам сопоставления и распознавания.
- Начальный этап зрительных систем млекопитающих состоит в обнаружении краев и местных особенностей.

# Выделение границ

---

- Резкие перепады яркостей:
  - Границы объекта:
    - Светлый объект располагается на темном фоне, или темном объект – на светлом фоне;ы
  - Изменения структуры (текстуры):
    - У зебры – линии, у леопарда – пятна;
  - Перепад тени;
  - Рельеф поверхности.

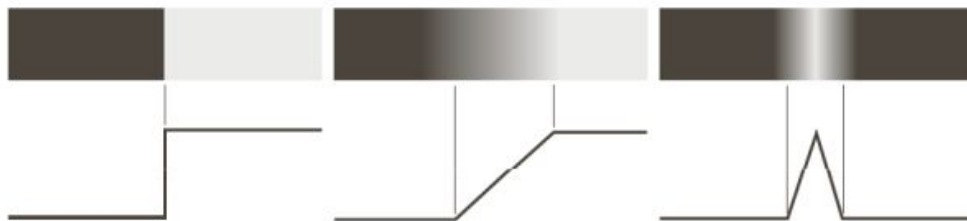
# Выделение границ

- Основная идея: искать места с сильным признаком перепада функции изображения.
- Проблемы:
  - Размер окрестности;
  - Как обнаружить перепады?
- Дифференциальные операторы:
  - Попытка приближения градиента изображения с использованием маски;
  - Пороговая фильтрация (постпроцессинг), для того, чтобы определить границу.

81	82	26	24
82	33	25	25
81	82	26	24



# Модели границ



a b c

**FIGURE 10.8** From left to right, models (ideal representations) of a step, a ramp, and a roof edge, and their corresponding intensity profiles.



**FIGURE 10.9** A  $1508 \times 1970$  image showing (zoomed) actual ramp (bottom, left), step (top, right), and roof edge profiles. The profiles are from dark to light, in the areas indicated by the short line segments shown in the small circles. The ramp and “step” profiles span 9 pixels and 2 pixels, respectively. The base of the roof edge is 3 pixels. (Original image courtesy of Dr. David R. Pickens, Vanderbilt University.)

# 1D операторы

mask  $M = [-1, 0, 1]$

$S_1$			12	12	12	12	12	24	24	24	24	24
$S_1$	$\otimes$	$M$	0	0	0	0	12	12	0	0	0	0

(a)  $S_1$  is an upward step edge

$S_2$			24	24	24	24	24	12	12	12	12	12
$S_2$	$\otimes$	$M$	0	0	0	0	-12	-12	0	0	0	0

(b)  $S_2$  is a downward step edge

$S_3$			12	12	12	12	15	18	21	24	24	24
$S_3$	$\otimes$	$M$	0	0	0	3	6	6	6	3	0	0

(c)  $S_3$  is an upward ramp

$S_4$			12	12	12	12	24	12	12	12	12	12
$S_4$	$\otimes$	$M$	0	0	0	12	0	-12	0	0	0	0

(d)  $S_4$  is a bright impulse or “line”

Figure 5.11: Cross correlation of four special signals with first derivative edge detecting mask  $[-1, 0, 1]$ ; (a) upward step edge, (b) downward step edge, (c) upward ramp, and (d) bright impulse. Note that, since the coordinates of  $M$  sum to zero, output must be zero on a constant region.

Adapted from Shapiro and Stockman

# 1D операторы

mask  $M = [-1, 2, -1]$

$S_1$			12	12	12	12	12	24	24	24	24	24
$S_1$	$\otimes$	$M$	0	0	0	0	-12	12	0	0	0	0

(a)  $S_1$  is an upward step edge

$S_2$			24	24	24	24	24	12	12	12	12	12
$S_2$	$\otimes$	$M$	0	0	0	0	12	-12	0	0	0	0

(b)  $S_2$  is a downward step edge

$S_3$			12	12	12	12	15	18	21	24	24	24
$S_3$	$\otimes$	$M$	0	0	0	-3	0	0	0	3	0	0

(c)  $S_3$  is an upward ramp

$S_4$			12	12	12	12	24	12	12	12	12	12
$S_4$	$\otimes$	$M$	0	0	0	-12	24	-12	0	0	0	0

(d)  $S_4$  is a bright impulse or "line"

Figure 5.12: Cross correlation of four special signals with second derivative edge detecting mask  $M = [-1, 2, -1]$ ; (a) upward step edge, (b) downward step edge, (c) upward ramp, and (d) bright impulse. Since the coordinates of  $M$  sum to zero, response on constant regions is zero. Note how a *zero-crossing* appears at an output position where different trends in the input signal join.

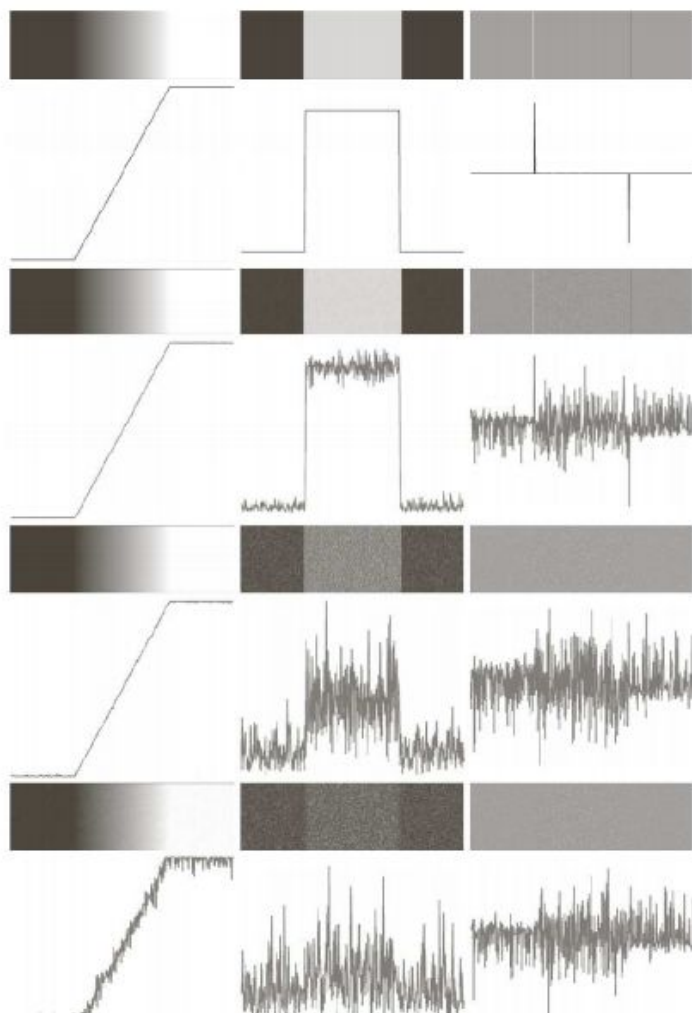
Adapted from Shapiro and Stockman

# Замечания

---

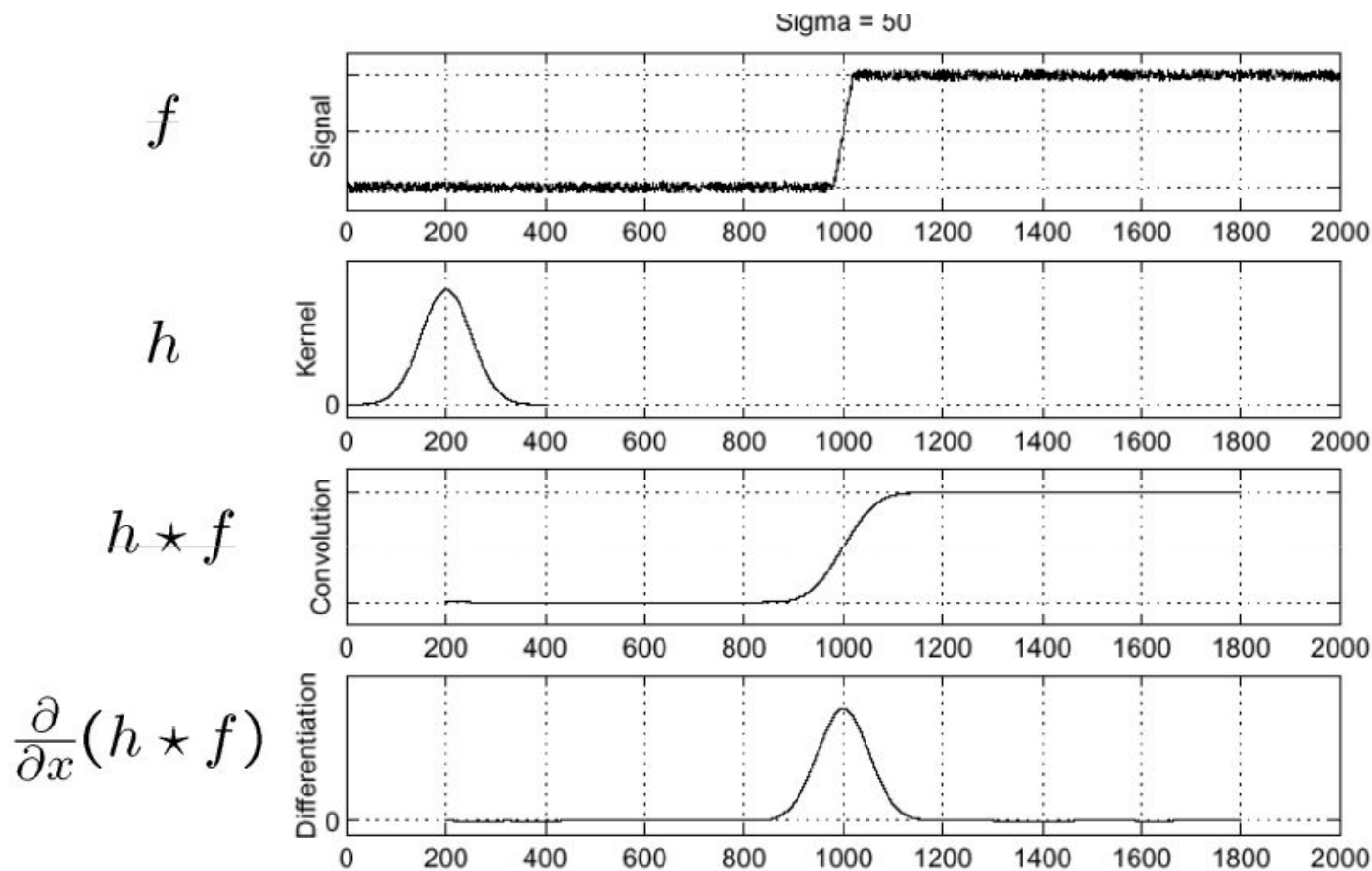
- Особенности масок для производных:
  - Весовые коэффициенты имеют противоположные знаки, чтобы получить высокий отклик в местах высокого контраста.
  - Сумма коэффициентов равна нули. На неизменяемых областях – нулевой отклик.
  - Первые производные: большие абсолютные значения в точках высокой контрастности.
  - Вторые производные: пересечение с 0 в точка высокой контрастности.

# Границы и шум



**FIGURE 10.11** First column: Images and intensity profiles of a ramp edge corrupted by random Gaussian noise of zero mean and standard deviations of 0.0, 0.1, 1.0, and 10.0 intensity levels, respectively. Second column: First-derivative images and intensity profiles. Third column: Second-derivative images and intensity profiles.

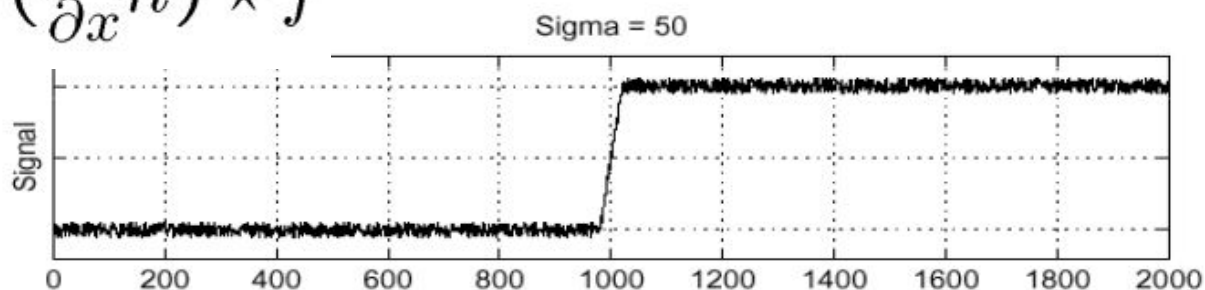
# Шум, выделение границ и сглаживание



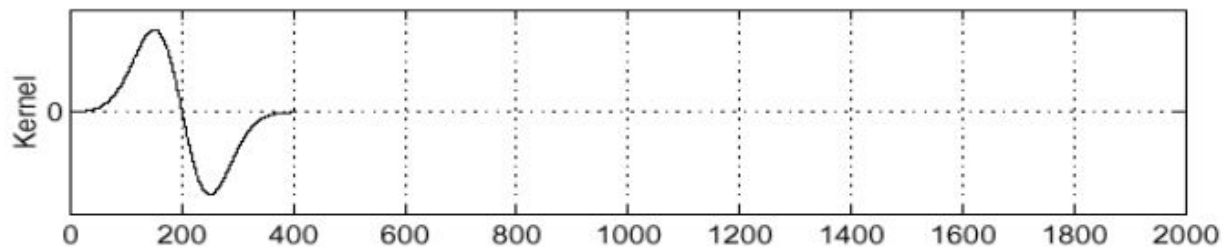
# Шум, выделение границ при сглаживании

$$\frac{\partial}{\partial x}(h \star f) = \left(\frac{\partial}{\partial x}h\right) \star f$$

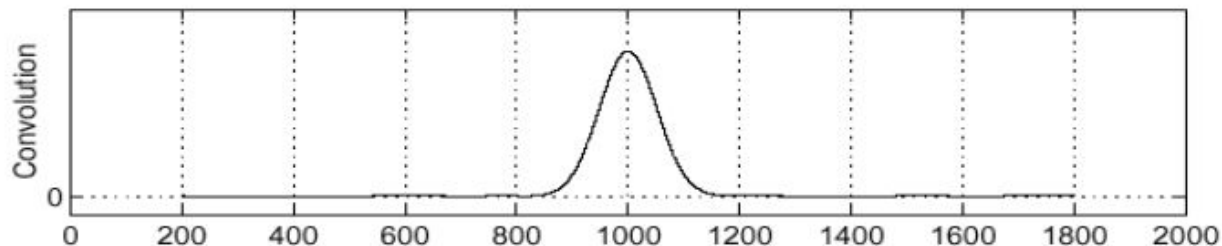
$f$



$\frac{\partial}{\partial x}h$



$\left(\frac{\partial}{\partial x}h\right) \star f$



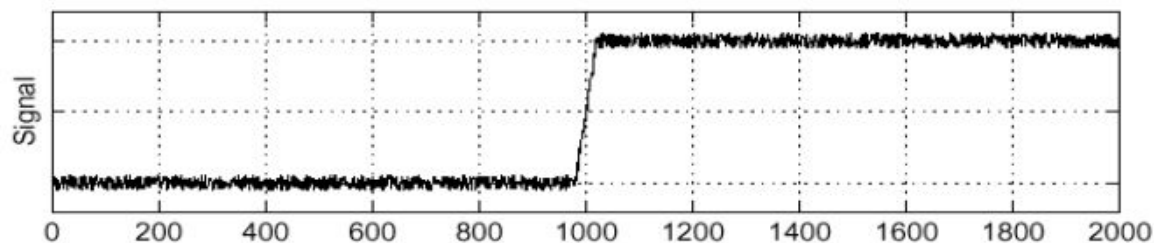


# Шум, выделение границ и сглаживание

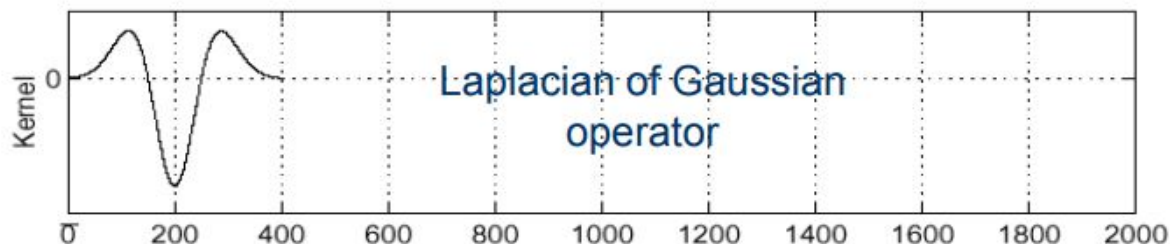
$$\frac{\partial^2}{\partial x^2}(h \star f)$$

Sigma = 50

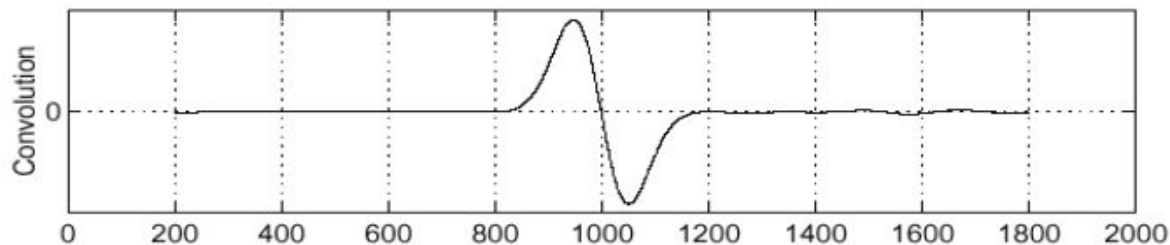
$f$



$\frac{\partial^2}{\partial x^2}h$



$(\frac{\partial^2}{\partial x^2}h) \star f$

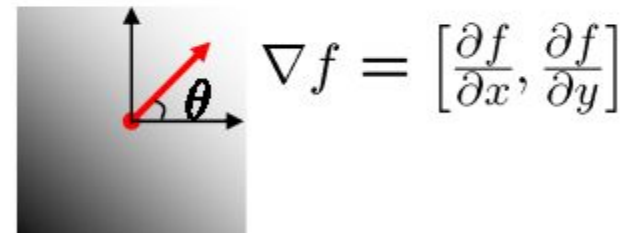
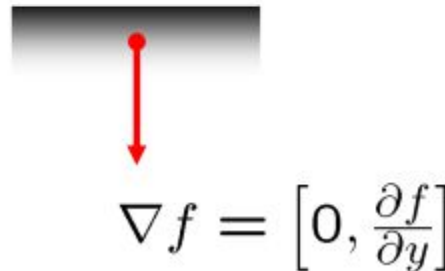
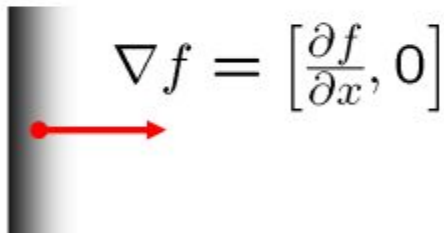




## 2D операторы

- Контрастность для 2D функции изображения может рассматриваться в различных направлениях.
- Из математики: можно вычислить наибольшее изменение функции по направлению.
- Градиент функции в точке  $(x,y)$ :

$$\nabla f = \left[ \frac{\partial f}{\partial x} \quad \frac{\partial f}{\partial y} \right]^T .$$



# Характеристика градиента

---

□ Норма градиента:

$$|\nabla f| = \left( \left( \frac{\partial f}{\partial x} \right)^2 + \left( \frac{\partial f}{\partial y} \right)^2 \right)^{1/2}$$

□ Направление градиента:

$$\angle(\nabla f) = \tan^{-1} \left( \frac{\partial f}{\partial y} / \frac{\partial f}{\partial x} \right)$$

# 2D операторы

$z_1$	$z_2$	$z_3$
$z_4$	$z_5$	$z_6$
$z_7$	$z_8$	$z_9$

-1	0	0	-1
0	1	1	0

Roberts

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

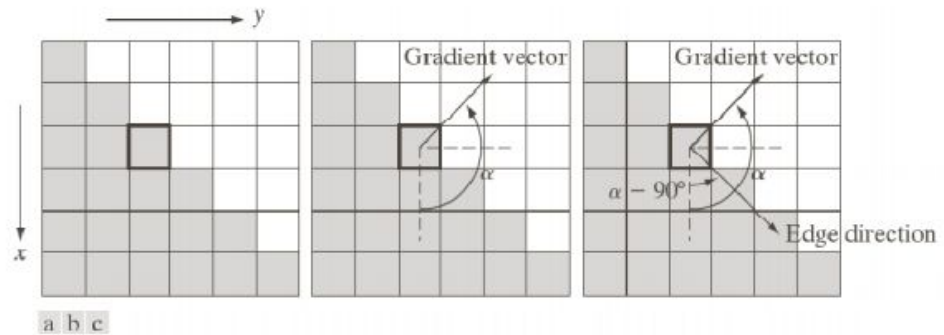
Prewitt

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

Sobel

a
b c
d e
f g

**FIGURE 10.14**  
A  $3 \times 3$  region of an image (the  $z$ 's are intensity values) and various masks used to compute the gradient at the point labeled  $z_5$ .



**FIGURE 10.12** Using the gradient to determine edge strength and direction at a point. Note that the edge is perpendicular to the direction of the gradient vector at the point where the gradient is computed. Each square in the figure represents one pixel.

Adapted from Gonzales and Woods

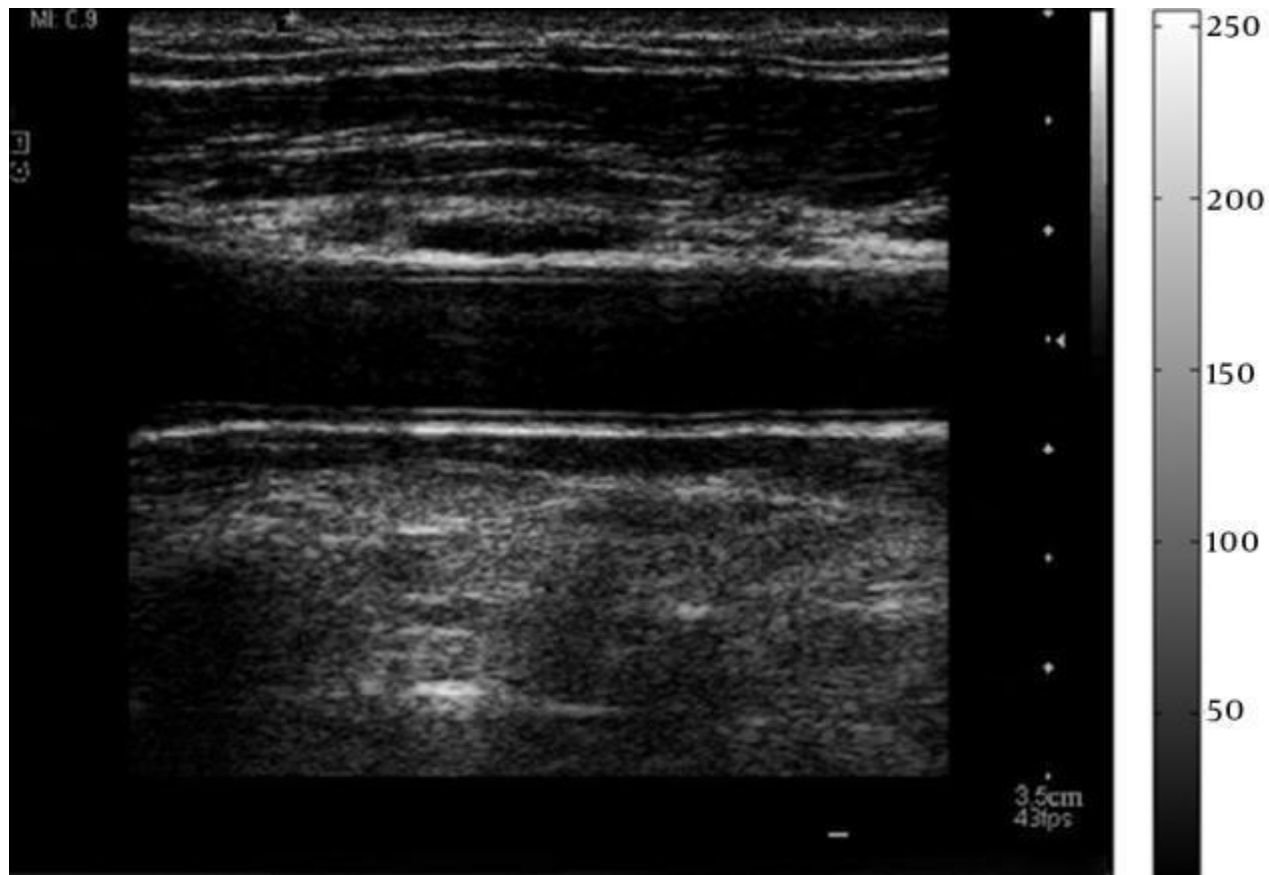
## 2D операторы

---

- Rafati M, Arabfard M, Rafati-Rahimzadeh M. Comparison of different edge detections and noise reduction on ultrasound images of carotid and brachial arteries using a speckle reducing anisotropic diffusion filter. *Iran Red Crescent Med J*. 2014;16(9):e14658. Published 2014 Sep 5. doi:10.5812/ircmj.14658

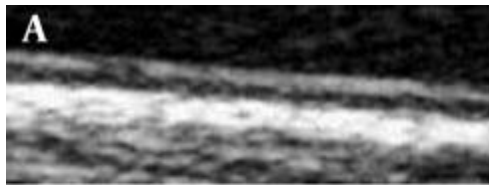
# 2D операторы

---



## 2D операторы

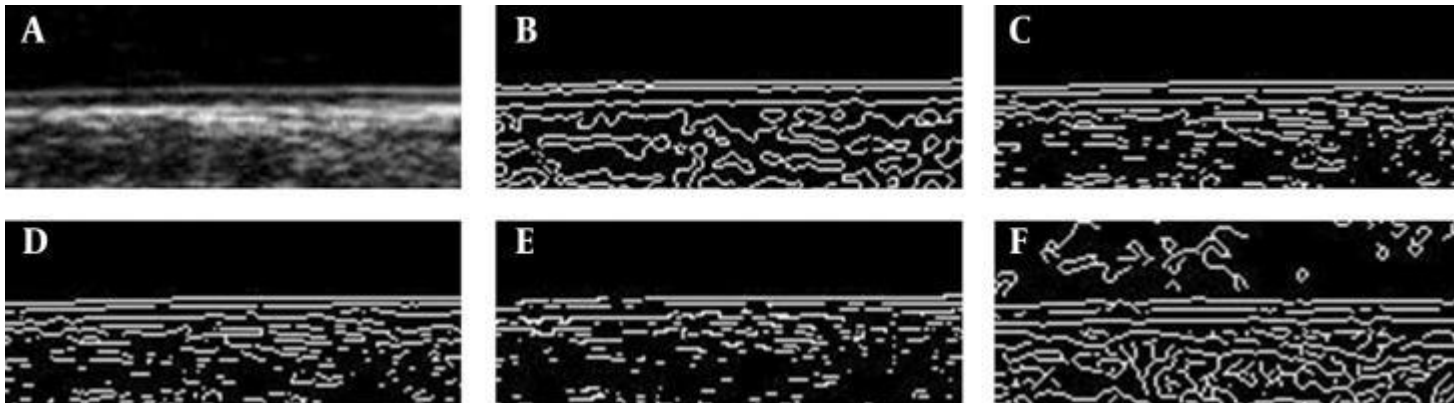
- **Ultrasound IMT of Left Common Carotid Artery**  
A) Original image, B) SRAD filter–Canny edge detection, C) SRAD filter–Sobel edge detection, D) SRAD filter–Prewitt edge detection, E) SRAD filter–Roberts edge detection, F) SRAD filter–LOG edge detection



## 2D операторы

---

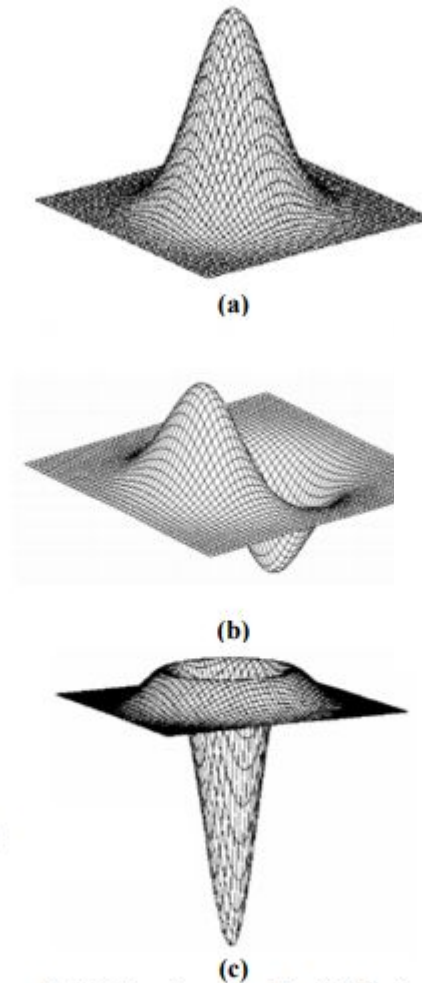
- **Ultrasound IMT of Left Brachial Artery** A) Original image, B) SRAD filter-Canny edge detection, C) SRAD filter-Sobel edge detection, D) SRAD filter-Prewitt edge detection, E) SRAD filter-Roberts edge detection, F) SRAD filter-LOG edge detection



# 2D операторы

Лапласиан в «чистом виде»:  
«-» чувствителен к шуму,  
«-» формирует двойные границы,  
«-» невозможно вычислить направление градиента.

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}.$$



$$h_{\sigma}(u, v) = \frac{1}{2\pi\sigma^2} e^{-\frac{u^2+v^2}{2\sigma^2}}$$

$$\frac{\partial}{\partial x} h_{\sigma}(u, v)$$

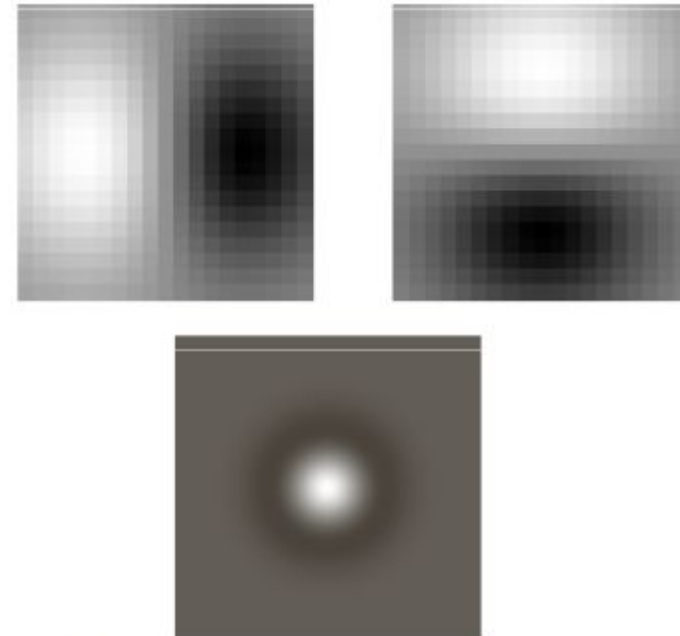
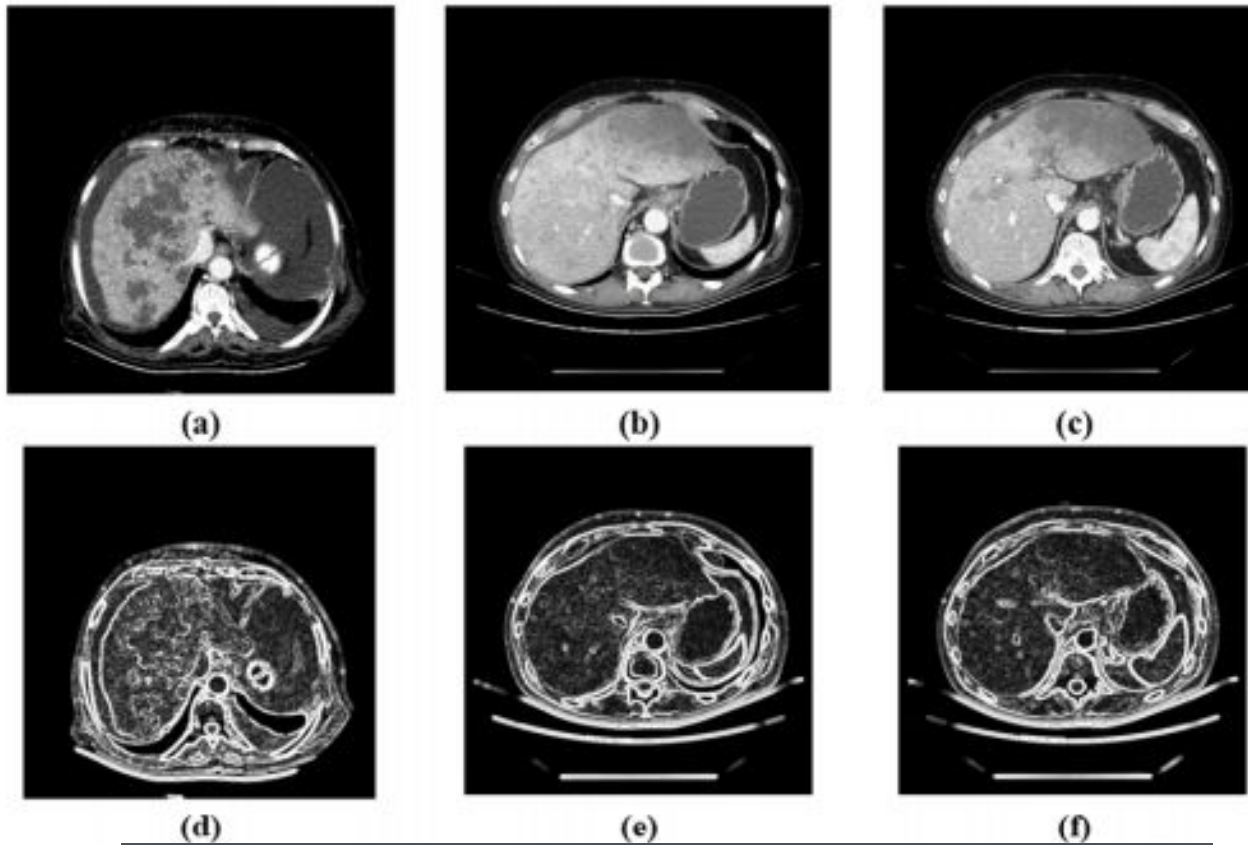


Figure 5. (a) Gaussian function (b) Derivative of Gaussian function (c) Laplacian of Gaussian function



# 2D операторы



S.N Kumar et al /J. Pharm. Sci. & Res. Vol. 9(5), 2017, 695-704

**ISSN:0975-1459**

Journal of Pharmaceutical  
Sciences and Research

[www.jpsr.pharmainfo.in](http://www.jpsr.pharmainfo.in)

Medical image Edge detection using Gauss  
Gradient operator

# 2D операторы

---

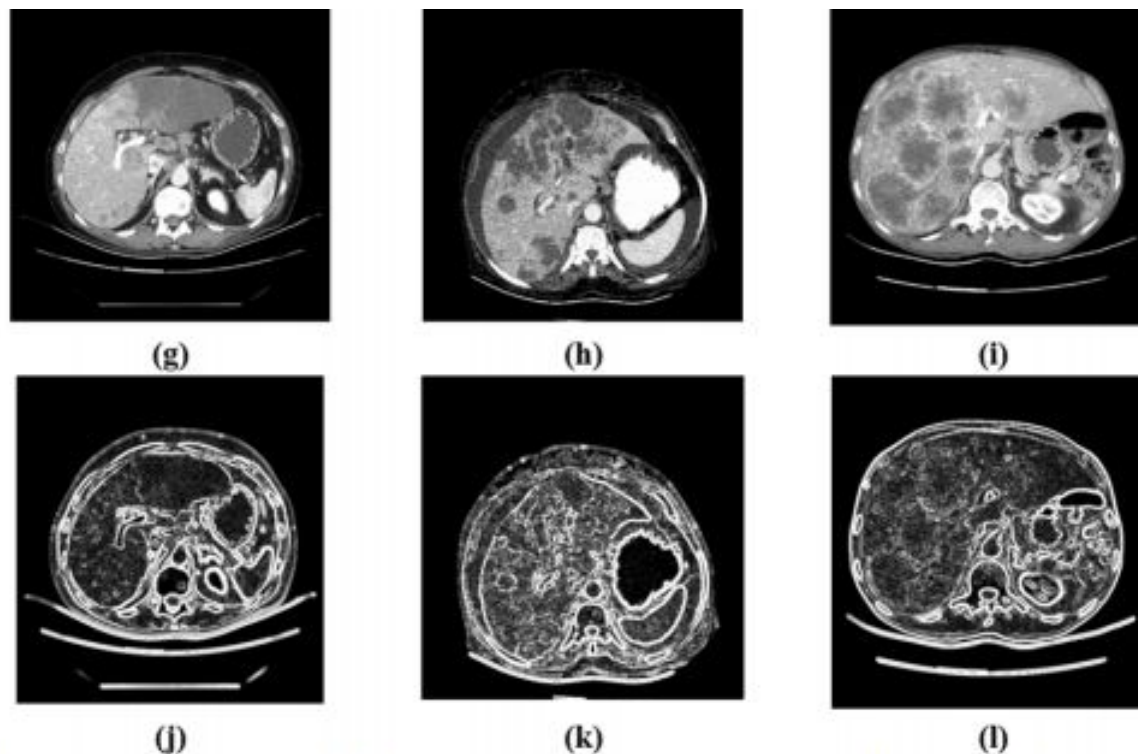


Figure 17. Dicom input images I 14 – I 19 (a,b,c, g, h, i) and their corresponding Gauss Gradient output images (d,e,f,j,k,l)

## Выделение границ: 3 основных шага

---

- 1. Сглаживание изображения. Шумоподавление.
- 2. Обнаружение граничных точек. Найти все точки- кандидаты на то, чтобы быть границей на изображении.
- 3. Уточнение границ. Оставить истинные границы на изображении.

# Детектор границ Кэнни( Canny)

---

- Характеристика границы по Кэнни:
- 1. Низкий уровень ошибок: все границы должны быть найдены, и не должно быть никаких ложно найденных границ.
- 2. Верная локализация граничных точек: граничные пиксели должны располагаться как можно ближе к истинным границам.
- 3. Характеристика толщины граничной линии: детектор должен возвращать только одну точку для каждой точки истинной границы. То есть количество локальных максимумов вокруг истинного края должно быть минимальным.

# Детектор границ

---

1. Удалить шум на изображении при помощи фильтра Гаусса.
2. Вычислить магнитуду и направление градиента в каждой точке изображения.
3. Подавить немаксимумы: установить в 0 те пиксели, которые не являются строгими максимумами вдоль направления градиента.
4. Выполнить двойную пороговую фильтрацию.

# Детектор границ Кэнни

1. Compute  $x$  and  $y$  derivatives of image

$$I_x = G_\sigma^x * I \quad I_y = G_\sigma^y * I$$

2. Compute magnitude of gradient at every pixel

$$M(x, y) = |\nabla I| = \sqrt{I_x^2 + I_y^2}$$

3. Eliminate those pixels that are not local maxima of the magnitude in the direction of the gradient

4. Hysteresis Thresholding

- Select the pixels such that  $M > T_h$  (high threshold)

Adapted from Martial Hebert, CMU

- Collect the pixels such that  $M > T_l$  (low threshold) that are neighbors of already collected edge points

!012

32

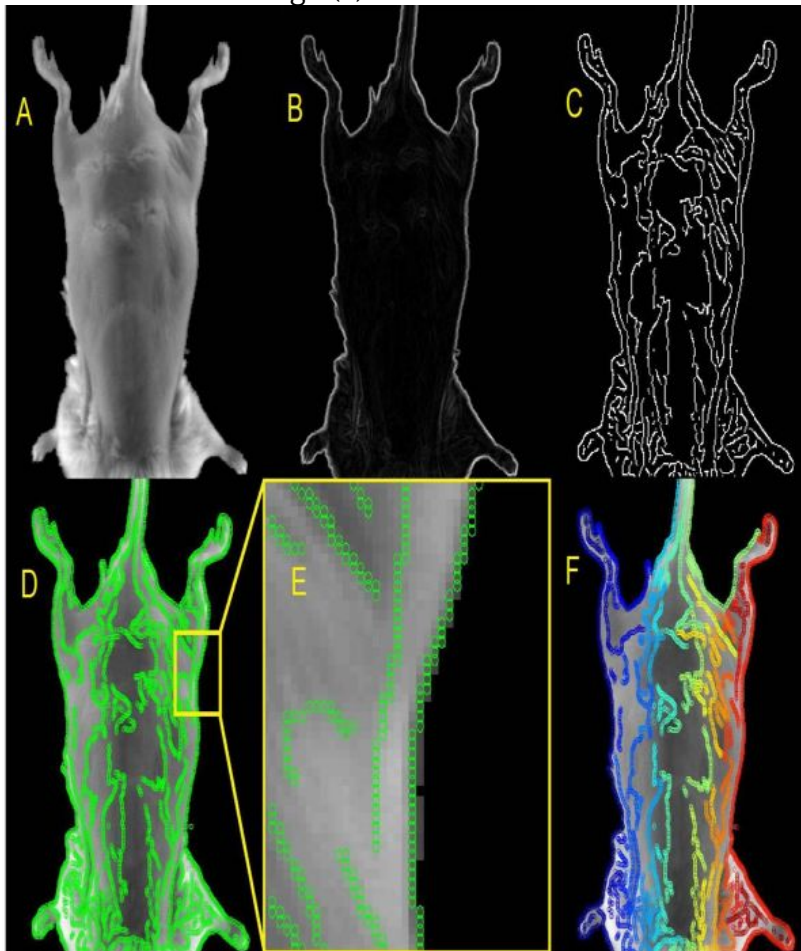
# Детектор границ Кэнни

---

- Kearney V, Huang Y, Mao W, Yuan B, Tang L. Canny edge-based deformable image registration. *Phys Med Biol*. 2017;62(3):966–985.  
doi:10.1088/1361-6560/aa5342

# Детектор границ Кэнни

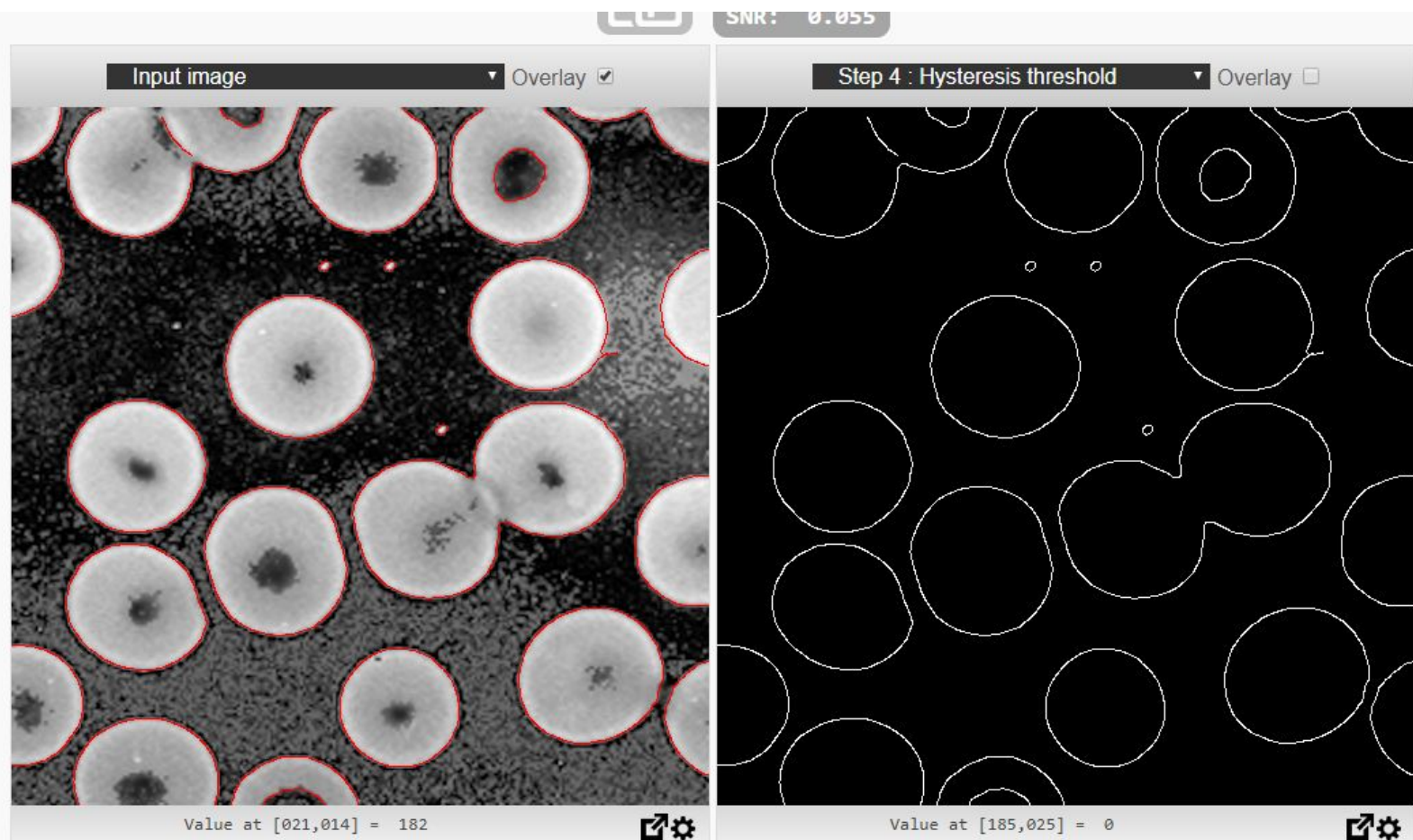
- ▣ The static image (A). The magnitude of the gradients (B). The Canny edge extraction (C). The points associated with the Canny edges superimposed on the static image (D). Zoomed in region of image D (E). Color coded separated canny edges, are shown superimposed on the static image (F).





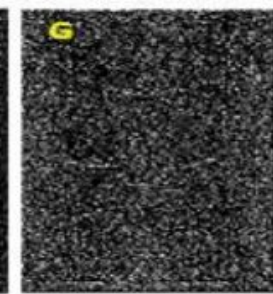
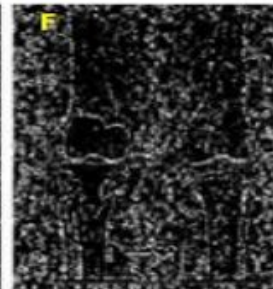
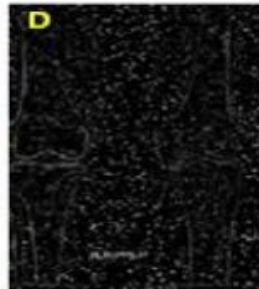
# Детектор границ Кэнни

□ <http://bigwww.epfl.ch/demo/ip/demos/edgeDetector/>



# Детектор границ Кэнни

A. Original X-Ray Image, B. X-Ray Image with salt & pepper noise, C. X-Ray image with Gaussian noise, D. Laplacian edge detected (salt & pepper), E. Laplacian edge detected (Gaussian), F. LOG edge detected (salt & pepper), G. LOG edge detected (Gaussian), H. Perwitt edge detected (salt & pepper), I. Perwitt edge detected (Gaussian), J. Canny edge detected (salt & pepper), K. Canny edge detected (Gaussian).



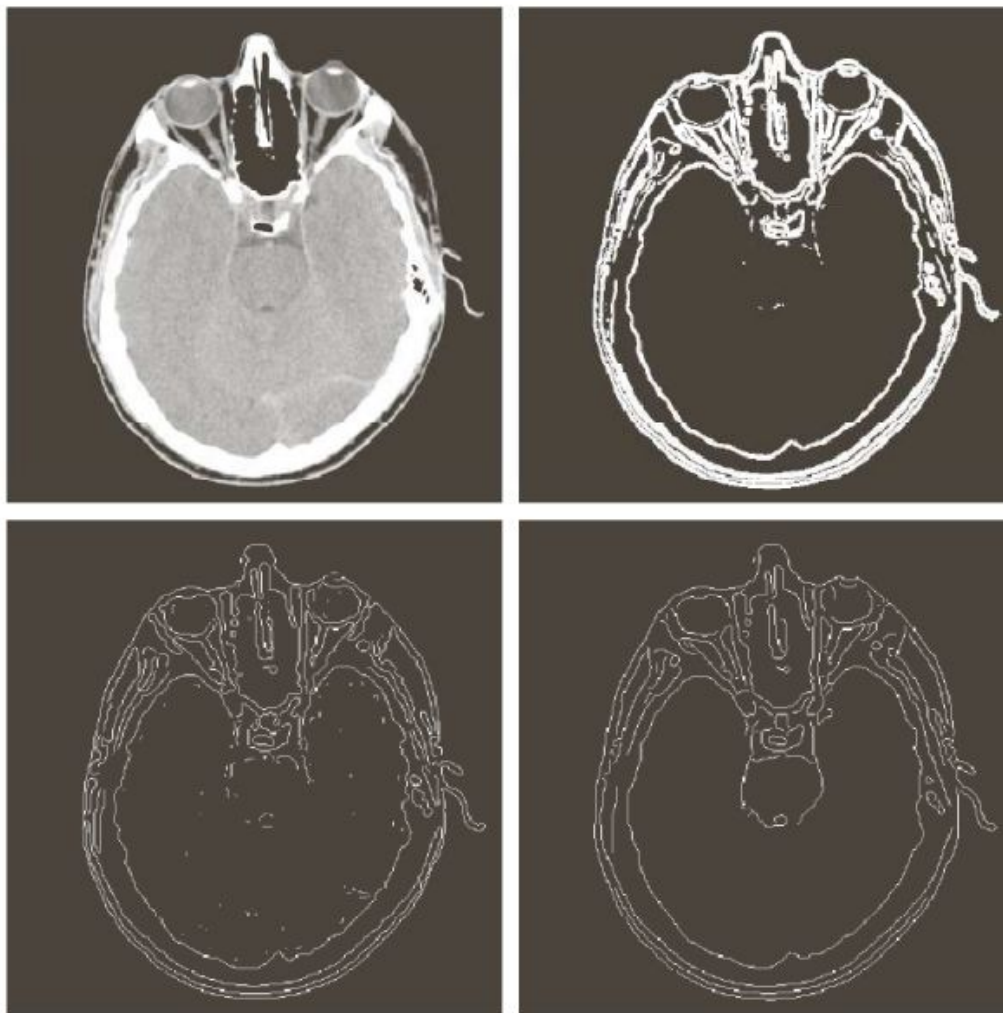
## Impact of Edge Detection Algorithms in Medical Image Processing

Ruba Anas, Hadeel A. Elhadi, Elmustafa Sayed Ali\*

Department of Electrical and Electronics Engineering, Faculty of Engineering,  
Red Sea University, Port Sudan, Sudan

\*E-mail address: elmustafasayed@rsu.edu.sd

# Детектор границ Кэнни



a b  
c d

**FIGURE 10.26**

(a) Original head CT image of size  $512 \times 512$  pixels, with intensity values scaled to the range  $[0, 1]$ .  
(b) Thresholded gradient of smoothed image.  
(c) Image obtained using the Marr-Hildreth algorithm.  
(d) Image obtained using the Canny algorithm.  
(Original image courtesy of Dr. David R. Pickens, Vanderbilt University.)

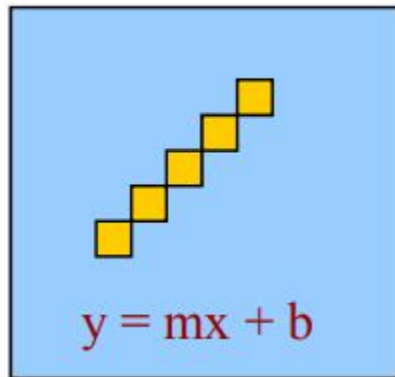
# Преобразование Хафа. Голосование

---

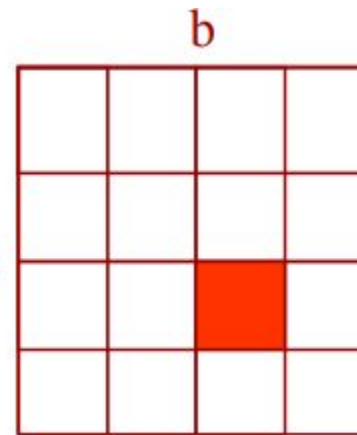
- Проблема «неточных форм», разрывных (недетектированных) участков границ и т.д.
- Контур описывается при помощи модели.
- Идея: каждый пиксель голосует за те модели, которым он подходит.
- Параметры модели – полный перебор.
- Выбираем модели с наибольшим количеством голосов, или модели, получившие определенный процент голосов (в зависимости от постановки задачи).
- Шумовые пиксели, конечно, вносят свой (теоретически незначительный) вклад в общий результат.

# Преобразование Хафа

image



m



accumulator

- 
- Точка в пространстве изображения соответствует линии (набору параметров) в пространстве Хафа. Объяснение: через одну точку проходит бесконечное множество прямых.
  - Точке в пространстве Хафа соответствует линия в пространстве изображения.



# Преобразование Хафа: линии

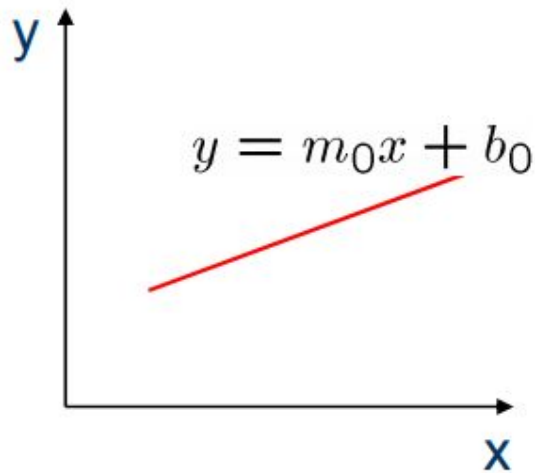
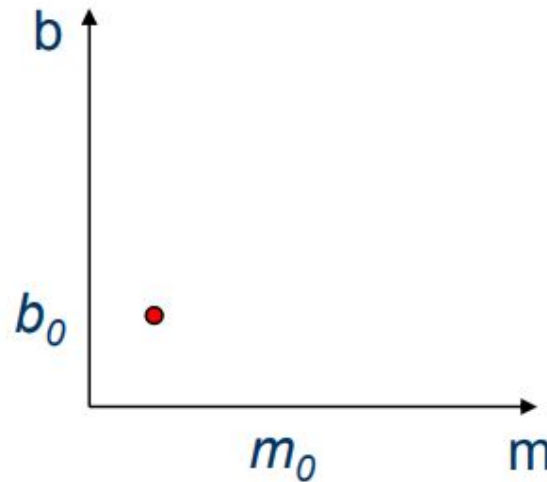


Image space



Hough (parameter) space

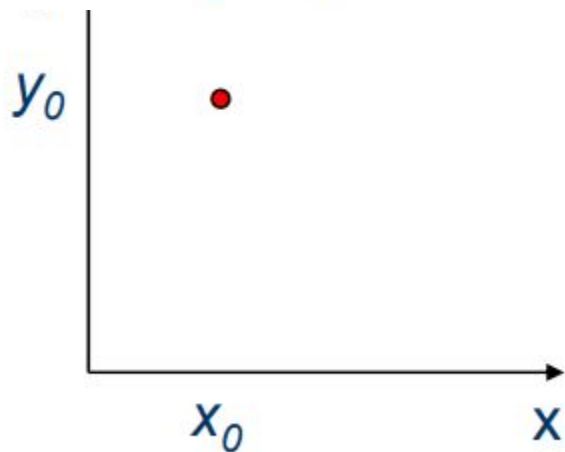
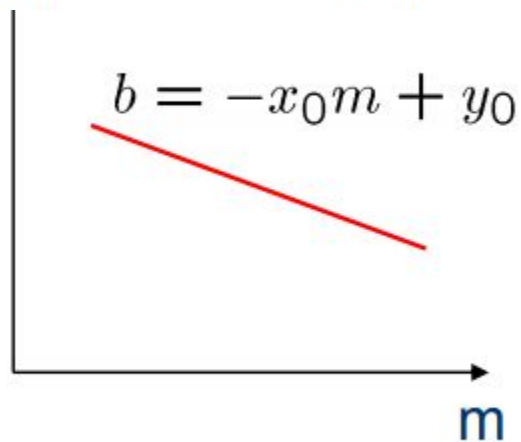


Image space



Hough (parameter) space

# Преобразование Хафа: линии

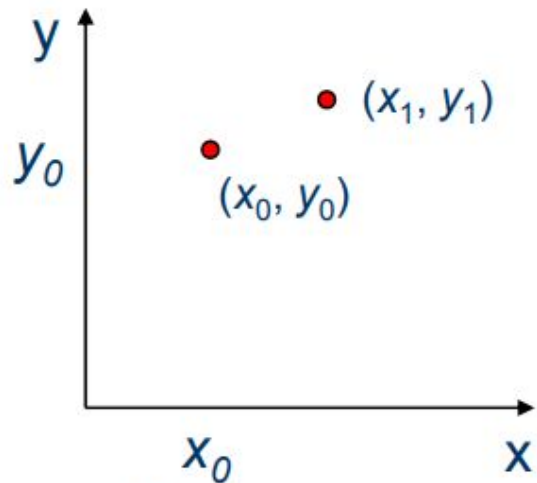
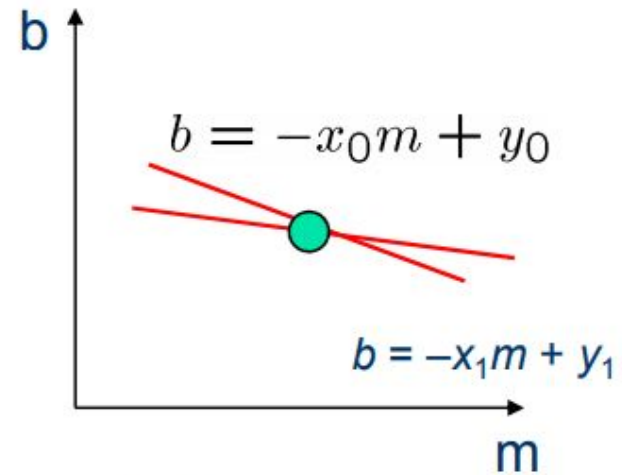


Image space

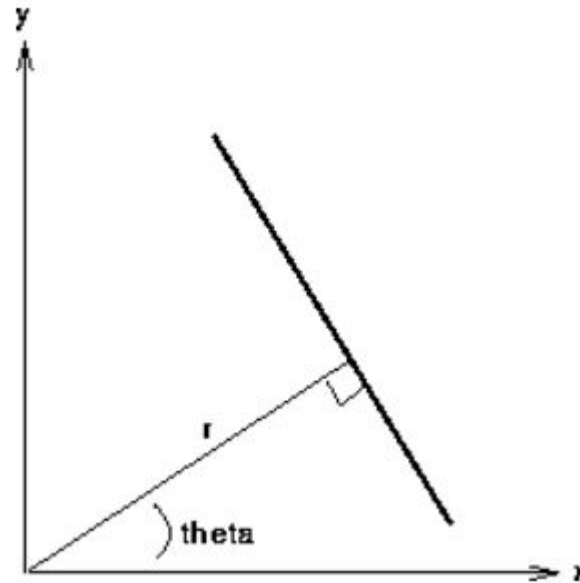
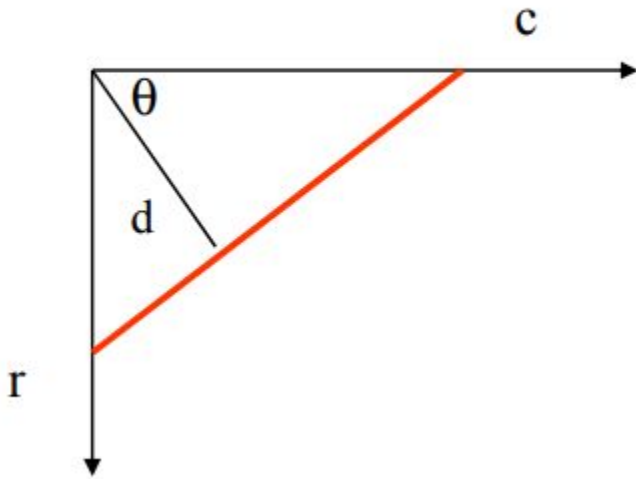


Hough (parameter) space



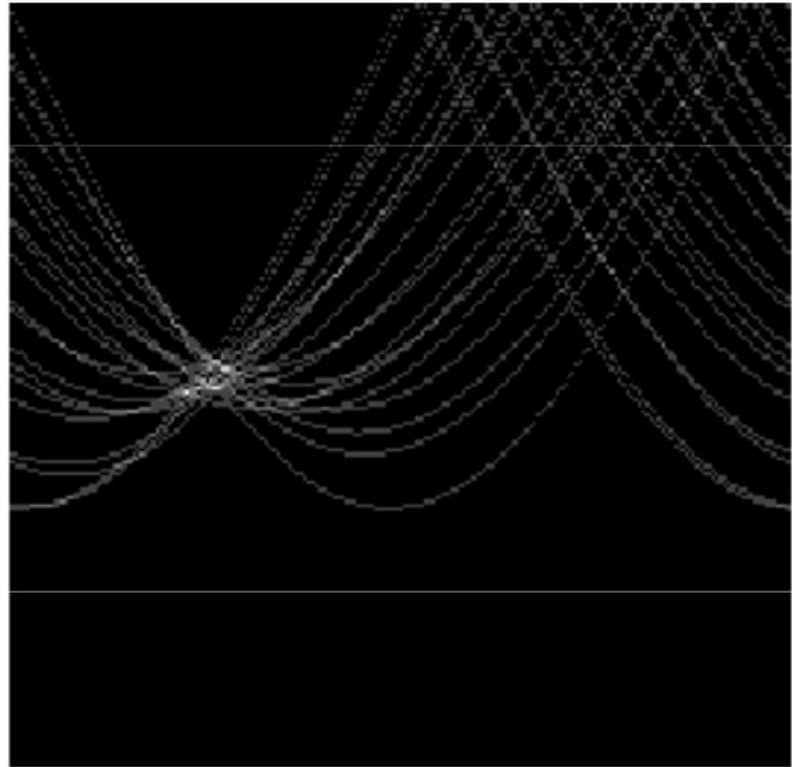
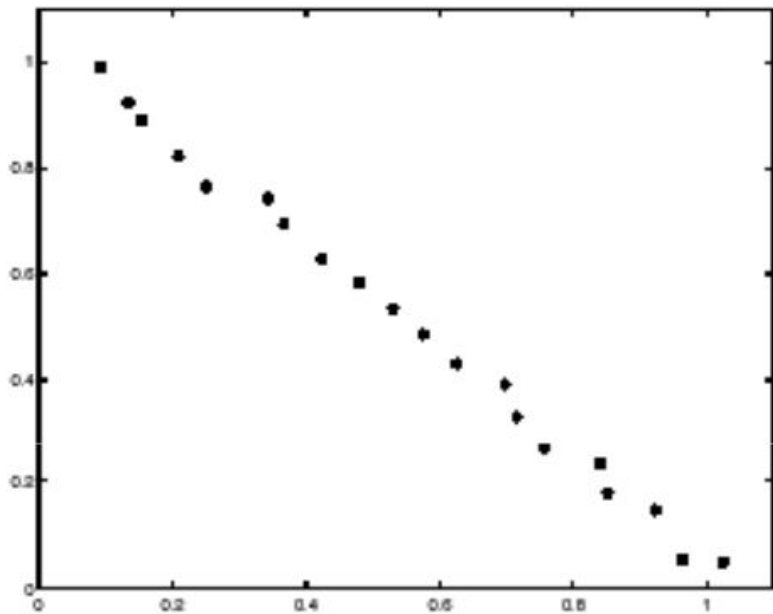
# Преобразование Хафа: отрезок

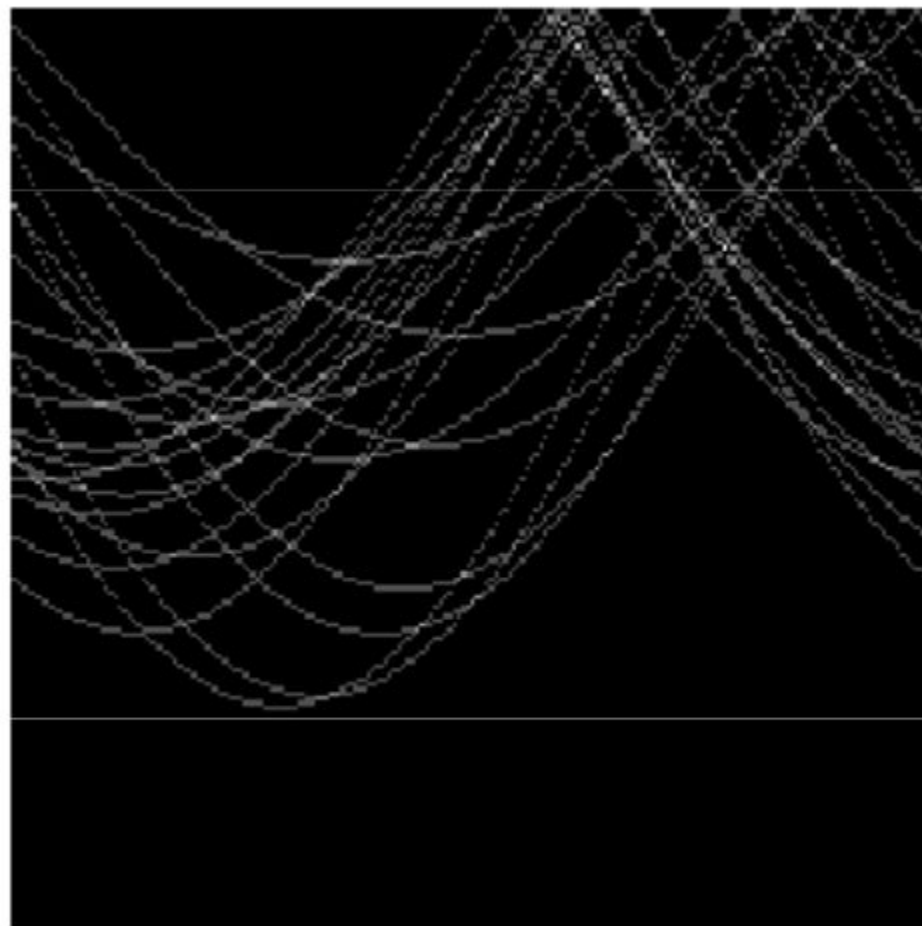
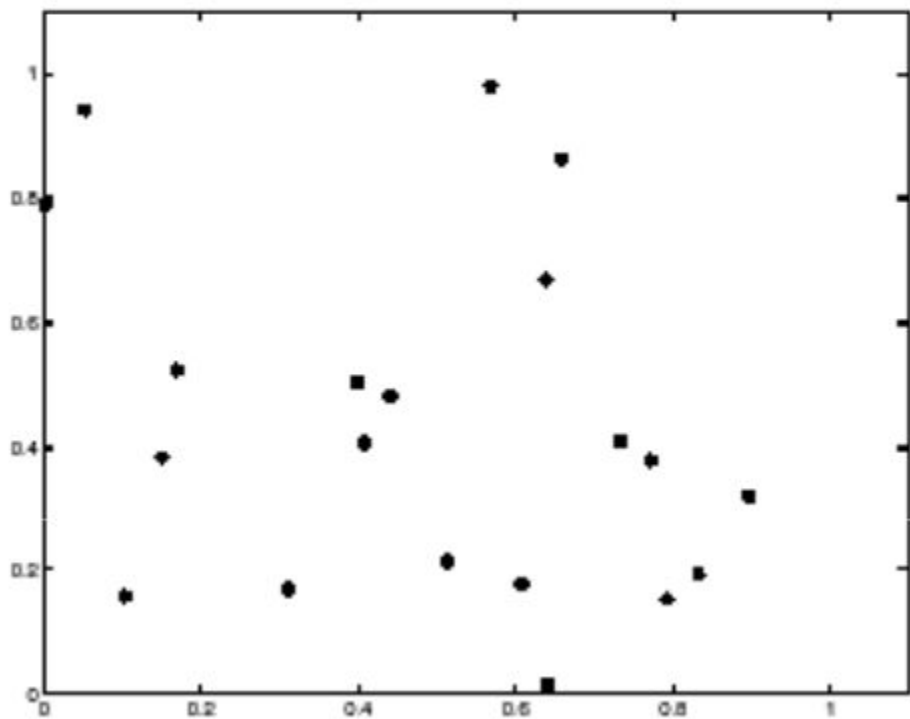
$$x \cos \theta + y \sin \theta = r$$



$$d = r \sin(\theta) + c \cos(\theta).$$

# Преобразование Хафа: отрезок





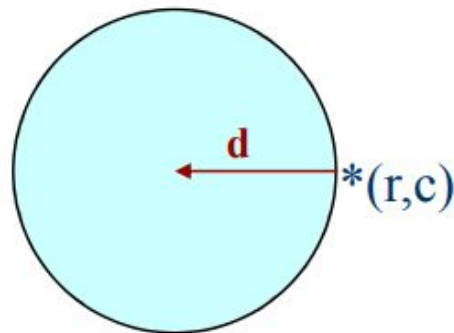
# Преобразование Хафа: окружности

---

- $r = r_0 + d \sin(\theta)$

$r_0, c_0, d$  are parameters

- $c = c_0 + d \cos(\theta)$



# Преобразование Хафа

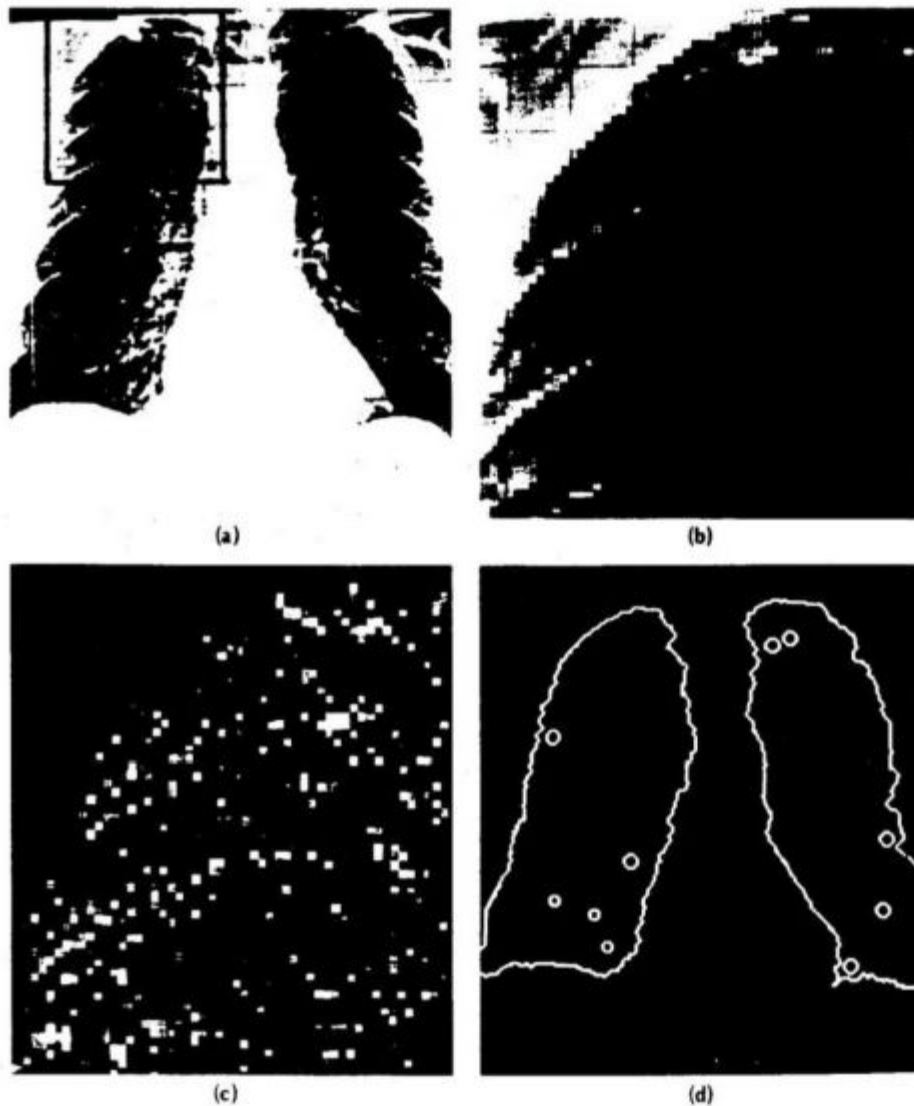
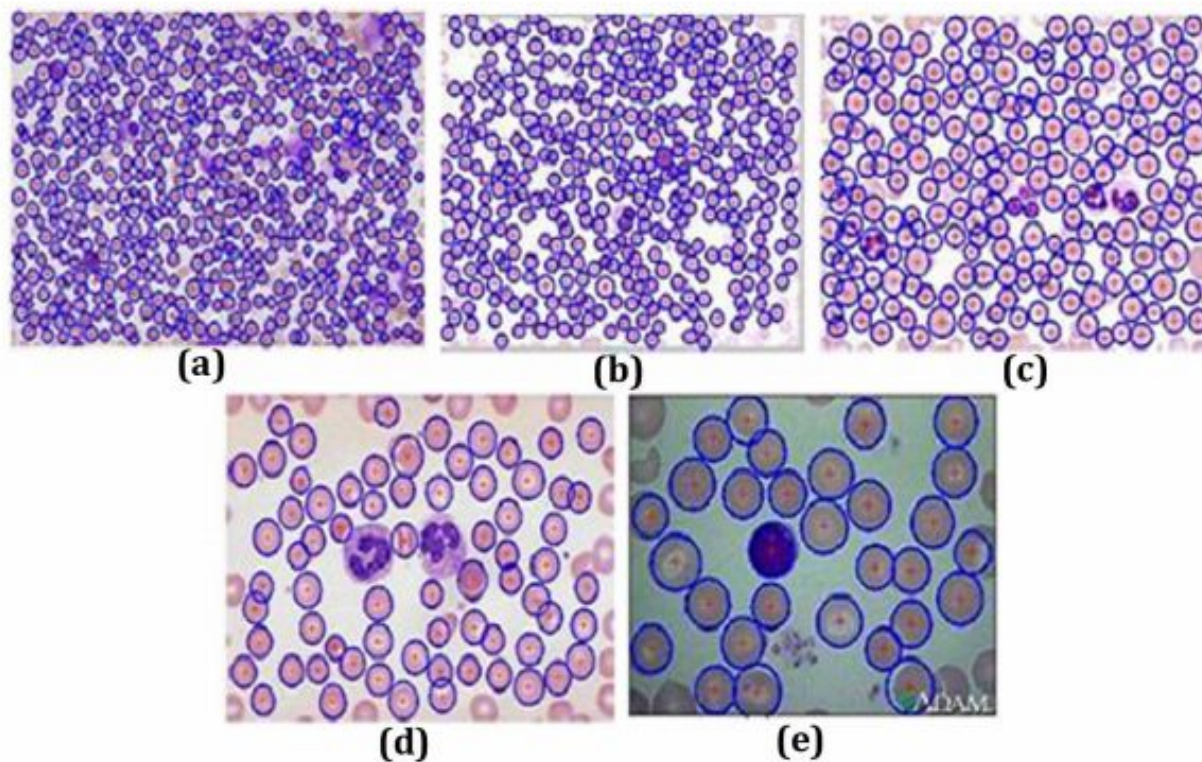


Fig. 4.7 Using the Hough technique for circular shapes. (a) Radiograph. (b) Window. (c) Accumulator array for  $r = 3$ . (d) Results of maxima detection.

# Преобразование Хафа

- <https://www.researchgate.net/publication/316202410>  
Application of Biomedical Image Processing in Blood Cell Counting Using Hough Transform



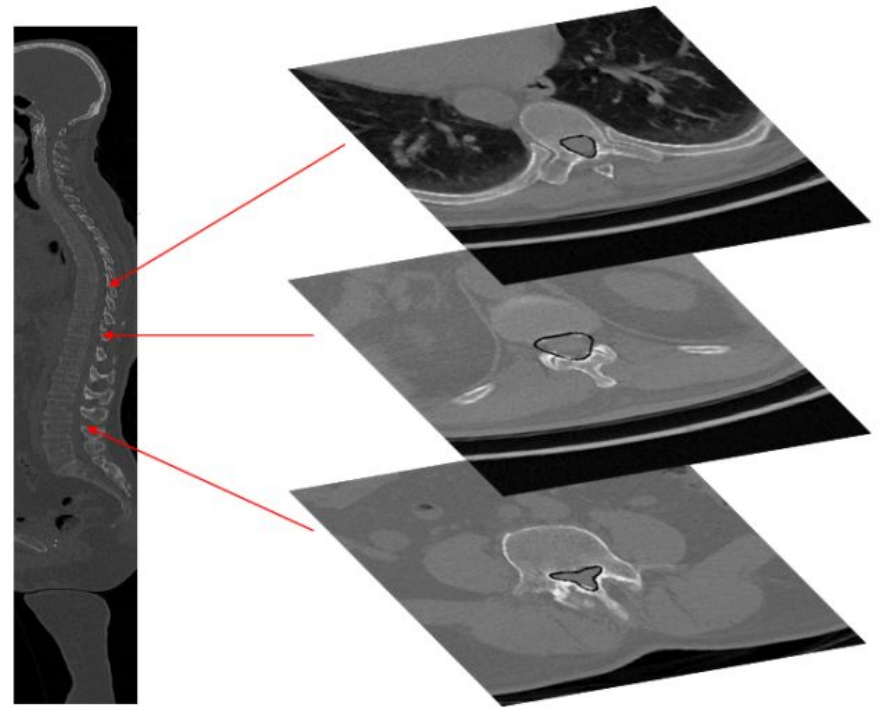
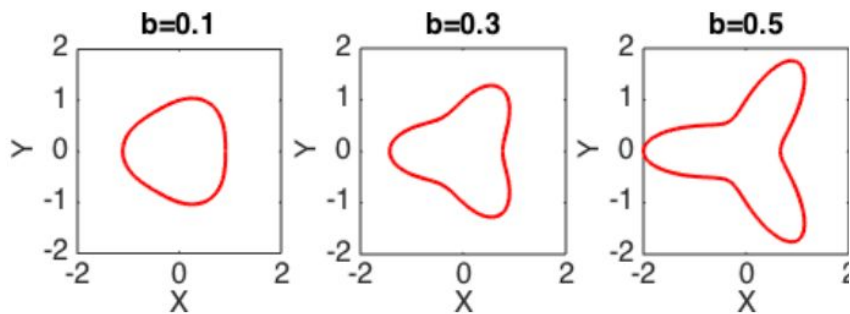
# Преобразование Хафа

□ <http://www1.mat.uniroma1.it/ricerca/seminari/mdn/PRESENTAZIONI/Campi18.pdf>

$$C_{a,b} : (X^2 + Y^2)^3 = [a(X^2 + Y^2) - b(X^3 - 3XY^2)]^2$$

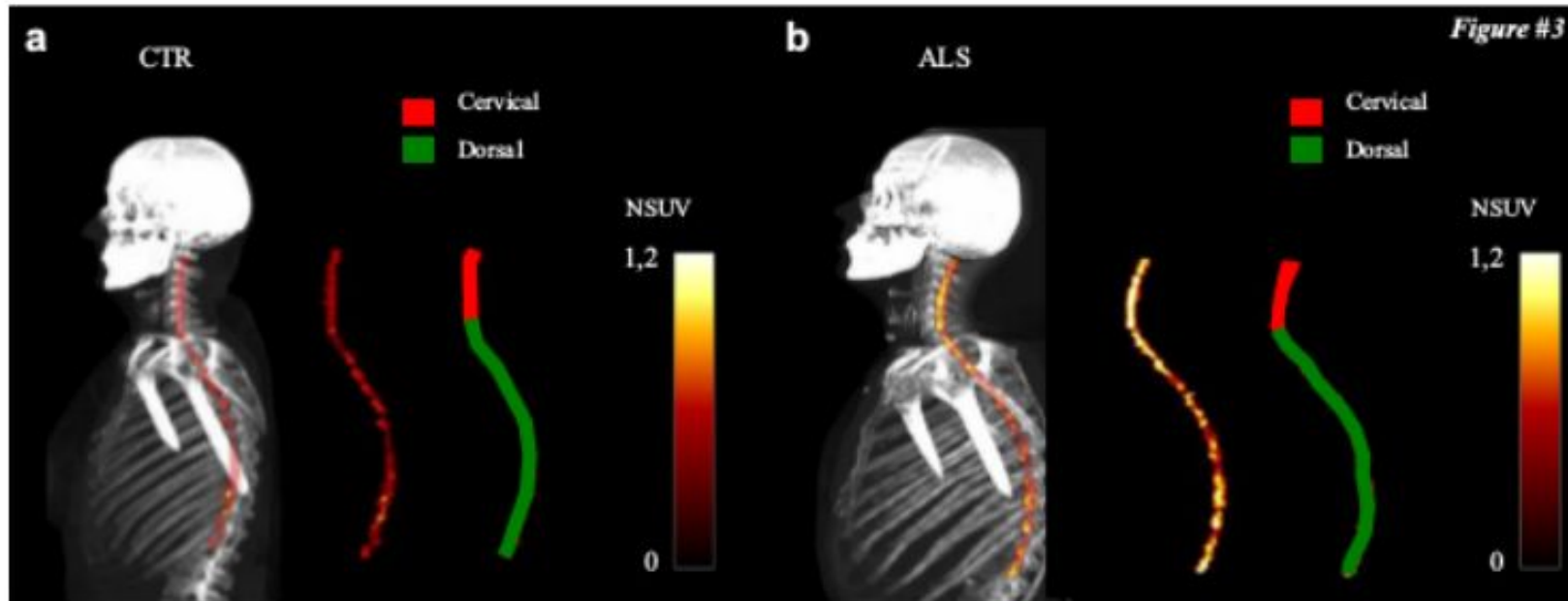
$$\Gamma_P(A, B) : (sA - tB + s^{3/2})(sA - tB - s^{3/2}) = 0$$

$$\text{with } s = x^2 + y^2, t = x^2 - 3xy^2$$





# Преобразование Хафа





# Подгонка моделей. Метод наименьших квадратов

## □ Для линии

$$LSE = \sum_{i=1}^n (ax_i + b - y_i)^2$$

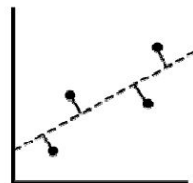
## □ Параметры для оптимума

$$\begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^n x_i^2 & \sum_{i=1}^n x_i \\ \sum_{i=1}^n x_i & \sum_{i=1}^n 1 \end{bmatrix}^{-1} \begin{bmatrix} \sum_{i=1}^n x_i y_i \\ \sum_{i=1}^n y_i \end{bmatrix}.$$

- If we use the geometric distance where  $ax + by + c = 0$  and  $a^2 + b^2 = 1$ , the solution for  $[a \ b]^T$  is the eigenvector corresponding to the smallest eigenvalue of

$$\begin{bmatrix} \sum_{i=1}^n x_i^2 - \left(\sum_{i=1}^n x_i\right)^2 & \sum_{i=1}^n x_i y_i - \left(\sum_{i=1}^n x_i\right)\left(\sum_{i=1}^n y_i\right) \\ \sum_{i=1}^n x_i y_i - \left(\sum_{i=1}^n x_i\right)\left(\sum_{i=1}^n y_i\right) & \sum_{i=1}^n y_i^2 - \left(\sum_{i=1}^n y_i\right)^2 \end{bmatrix}$$

and  $c = -a \sum_{i=1}^n x_i - b \sum_{i=1}^n y_i$ .



# RANSAC

---

На вход алгоритма поступает:

1. набор исходных данных  $X$
2. функция  $M$ , позволяющая вычислить параметры  $\theta$  модели  $P$  по набору данных из  $n$  точек
3. функция оценки  $E$  соответствия точек полученной модели
4. порог  $t$  для функции оценки
5. количество итераций метода  $k$

Весь алгоритм состоит из одного цикла, каждую итерацию которого можно логически разделить на два этапа.

- Первый этап — выбор точек и подсчёт модели.
  - Из множества исходных точек  $X$  случайным образом выбираются  $n$  различных точек.
  - На основе выбранных точек вычисляются параметры  $\theta$  модели  $P$  с помощью функции  $M$ , построенную модель принято называть гипотезой.
- Второй этап — проверка гипотезы.
  - Для каждой точки проверяется её соответствие данной гипотезе с помощью функции оценки  $E$  и порога  $t$
  - Каждая точка помечается инлаером или выбросом
  - После проверки всех точек, проверяется, является ли гипотеза лучшей на данный момент, и если является, то она замещает предыдущую лучшую гипотезу.

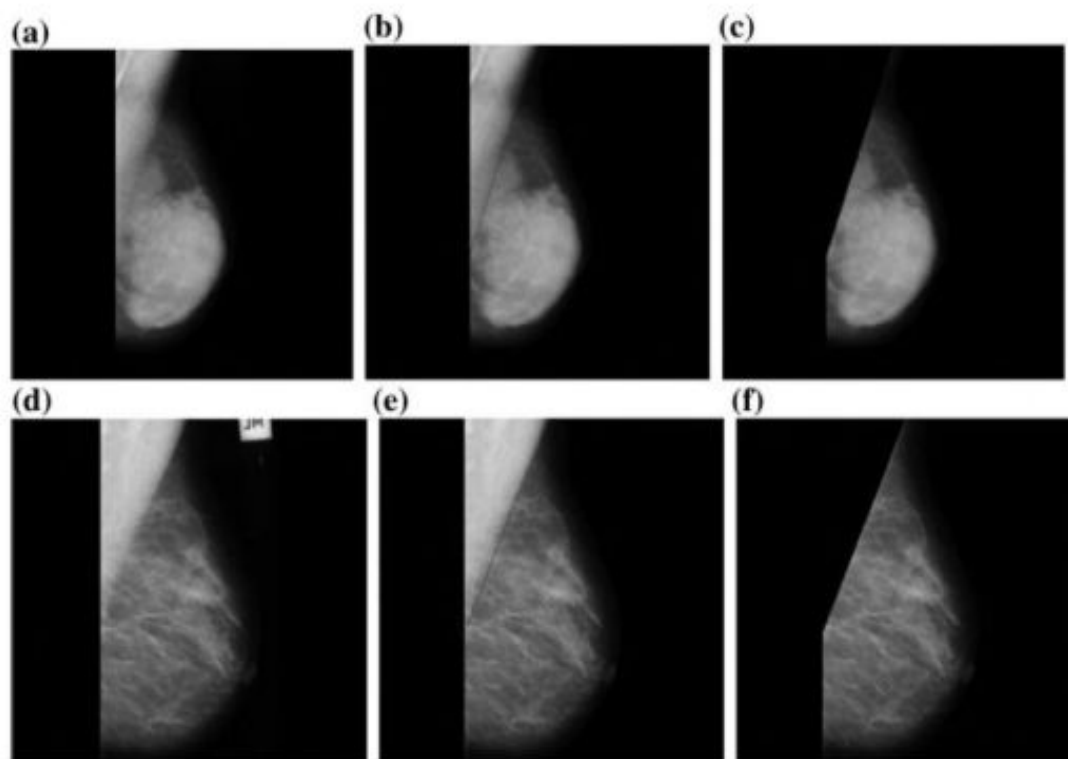
В конце работы цикла оставляется последняя лучшая гипотеза.

Результатом работы метода являются:

1. Параметры  $\theta$  модели  $P$
2. Точки исходных данных, помеченные инлаерами или выбросами.

# Примеры

- [https://books.google.ru/books?id=F6FPDAAA0BAI&pg=PA34&lpg=PA34&dq=edge+fitting+RANSAC+in+biomedical+application&source=bl&ots=xkOZ5XekG6&sig=ACfU3U3jZVBU26\\_igyllszc3oznEiK-ceO&hl=ru&sa=X&ved=2ahUKEwjt2fyEsf3kAhUqxosKHTVVD08O6AEwAXoECAkOAQ#v=onepage&q=edge%20fitting%20RANSAC%20in%20biomedical%20application&f=false](https://books.google.ru/books?id=F6FPDAAA0BAI&pg=PA34&lpg=PA34&dq=edge+fitting+RANSAC+in+biomedical+application&source=bl&ots=xkOZ5XekG6&sig=ACfU3U3jZVBU26_igyllszc3oznEiK-ceO&hl=ru&sa=X&ved=2ahUKEwjt2fyEsf3kAhUqxosKHTVVD08O6AEwAXoECAkOAQ#v=onepage&q=edge%20fitting%20RANSAC%20in%20biomedical%20application&f=false)

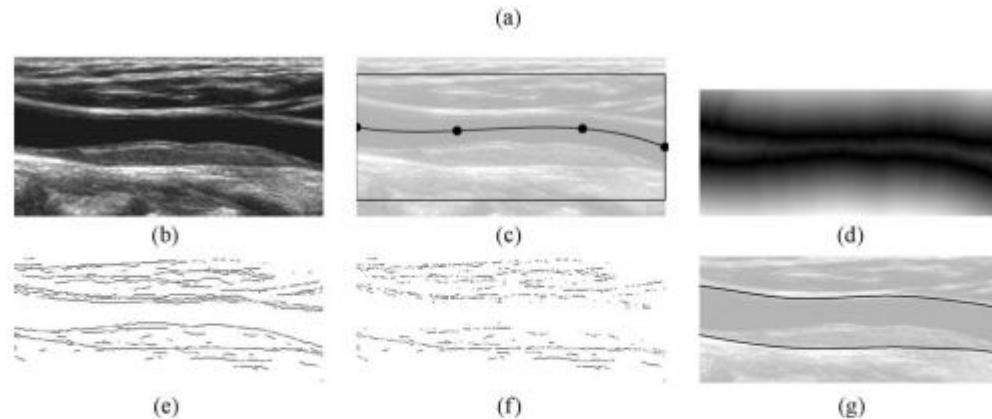


**Fig. 3** Experimental results of **RANSAC** Algorithm. **a** Original image mdb038. **b** Line by **RANSAC**. **c** Extracted ROI of mdb-038. **d** Original image mdb046. **e** Line by **RANSAC**. **f** Extracted ROI of mdb-046



## Segmentation of ultrasound images of the carotid using RANSAC and cubic splines

Rui Rocha<sup>a,b,\*</sup>, Aurélio Campilho<sup>a,c</sup>, Jorge Silva<sup>a,c</sup>, Elsa Azevedo<sup>d,e</sup>, Rosa Santos<sup>e</sup>

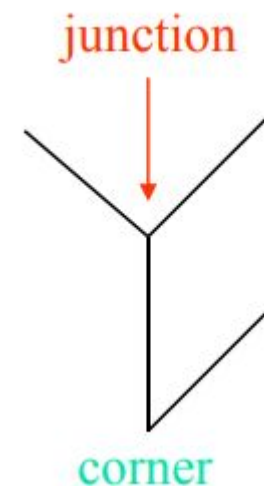


**Fig. 1** – Main steps of the method: (a) block diagram; (b) input image; (c) points entered by the user, the interpolated lumen medial axis and the ROI (region inside the rectangle); (d) the absolute values of SDL,  $|SDL|$ ; (e) edge map; (f) valley edge map; (g) estimated adventitia boundaries.

# Трэкинг границы

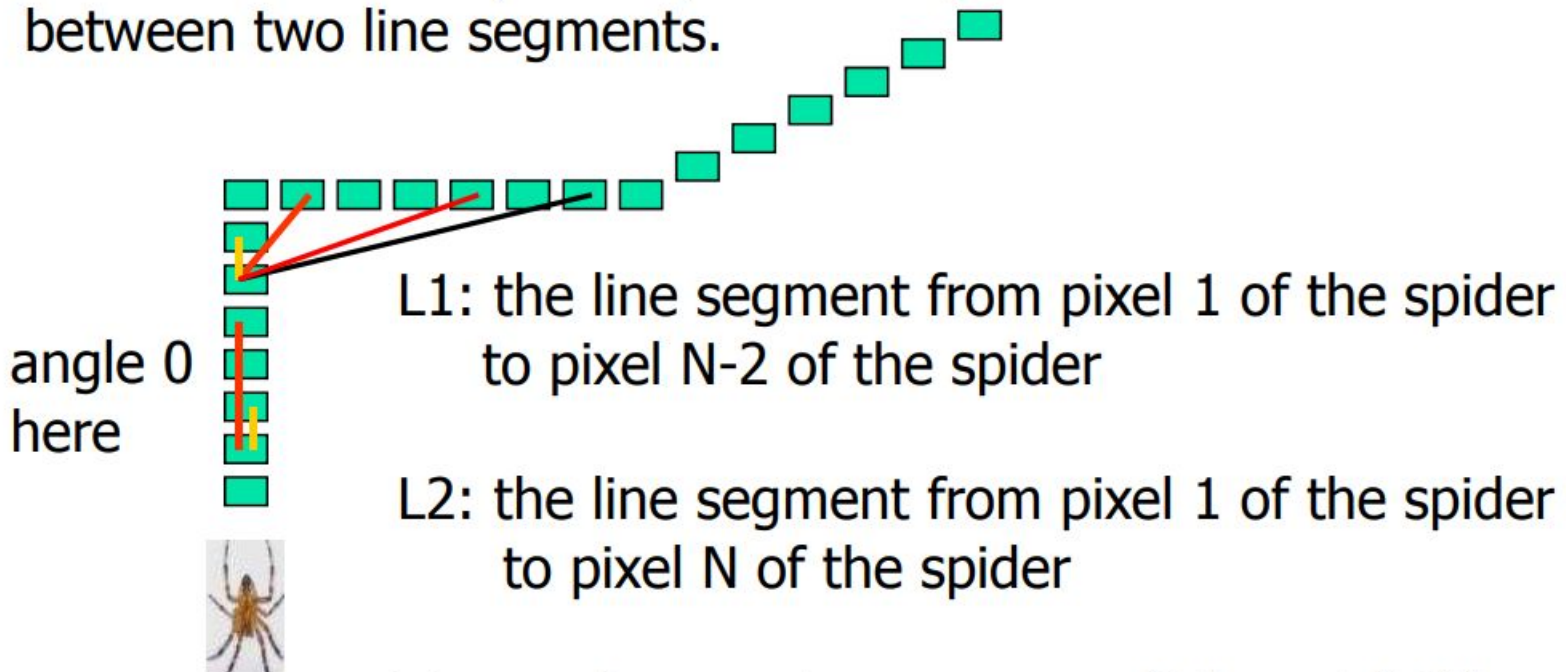
---

- Трэкинг на основе маски. Возможны следующие случаи:
  - начало нового отрезка,
  - Внутренняя точка отрезка,
  - конец отрезка
  - соединение нескольких отрезков
  - угол, поворот отрезка (или разбиение сегмент части)



# Алгоритм паучка 😊

The measure of asymmetry is the angle between two line segments.



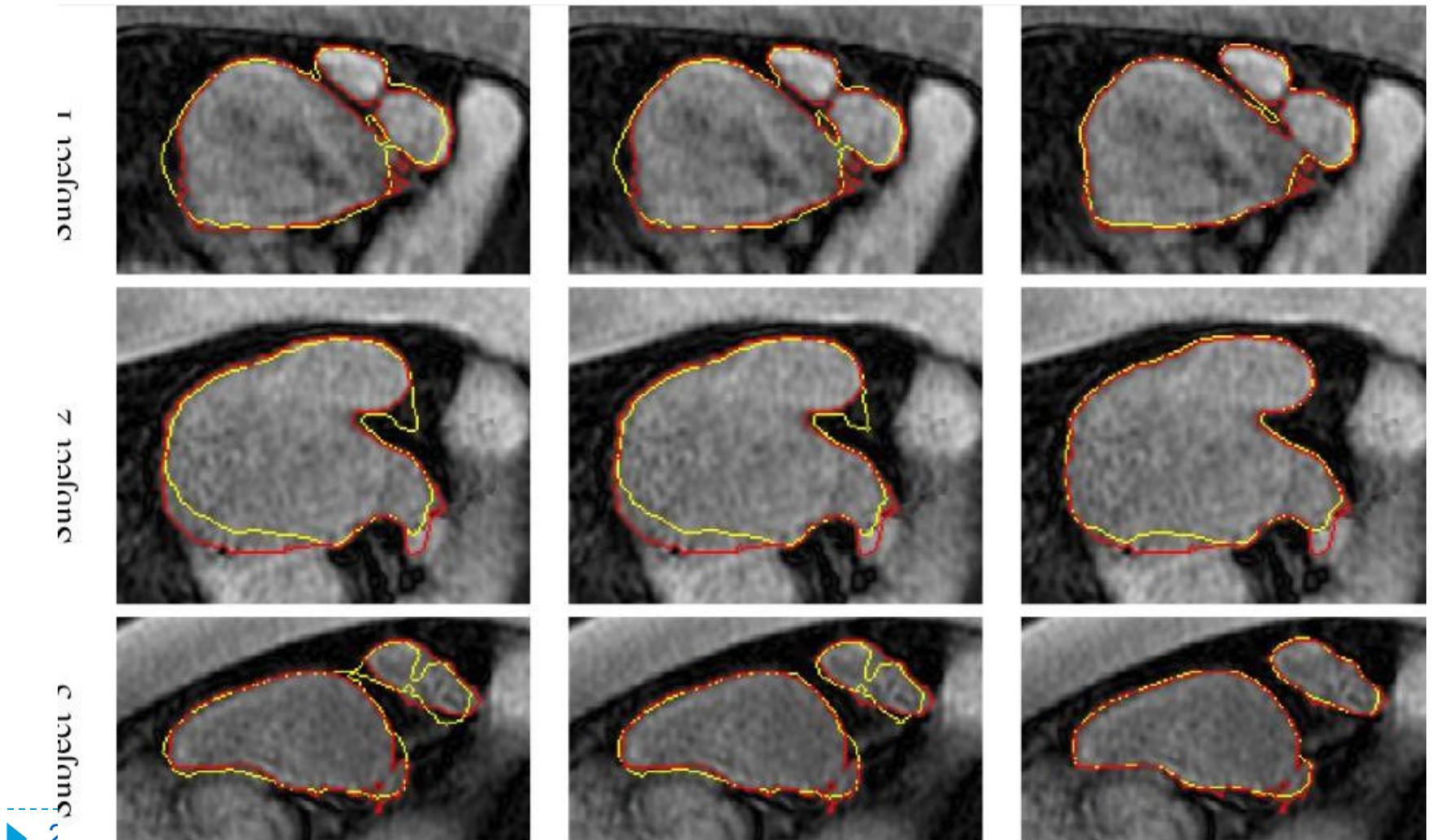
The angle must be  $\leq \arctan(2/\text{length}(L2))$

**Longer spiders allow less of an angle.**



# Сегментация на основе model fitting

- Examples slices for the atlas-based segmentation and contour-driven refinement of the left atrium. Automatic segmentation results are shown in yellow, manual delineations are shown in red.



(a) Majority voting

(b) Intensity voting

(c) Intensity + UCM label



**УНИВЕРСИТЕТ  
ЛОБАЧЕВСКОГО**

Институт информационных технологий, математики и  
механики  
Кафедра математического обеспечения и  
суперкомпьютерных технологий

**Обработка изображений**

# Вейвлет-преобразование

Светлана Носова  
svetlana.nosova@itmm.unn.ru



## Example - Haar basis (revisited)

<i>Resolution</i>	<i>Averages</i>	<i>Detail Coefficients</i>
4	[9 7 3 5]	[]
2	[8 4]	[1 -1]
4	[6]	[2]



[6 2 1 -1]

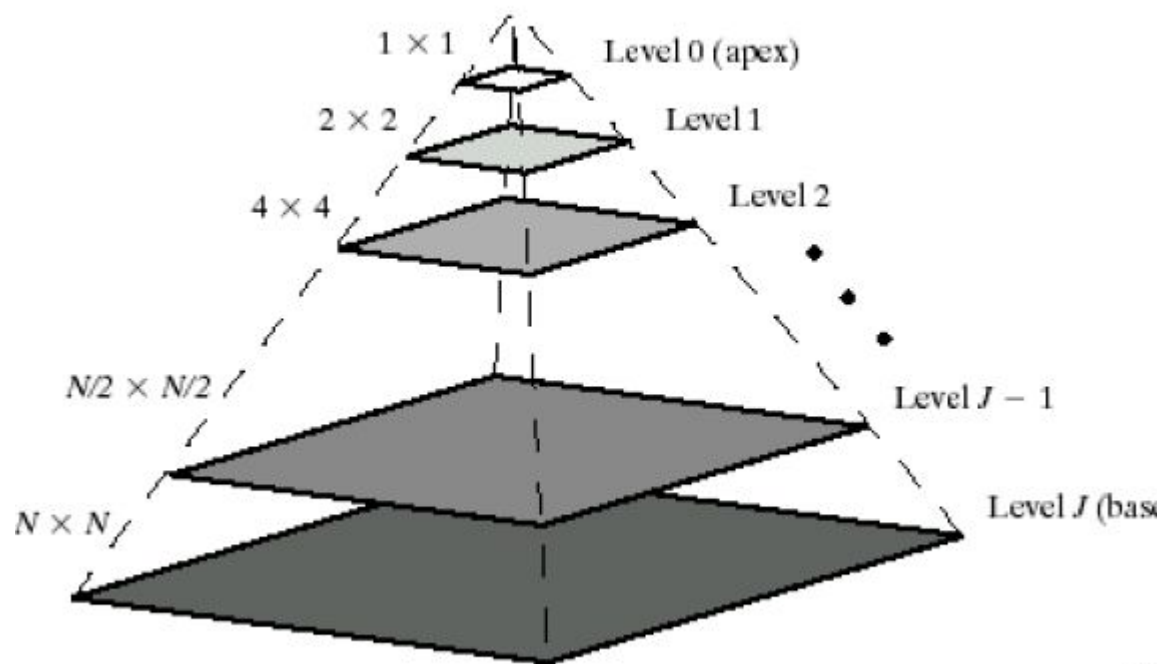
# Вейвлет-преобразование и многомасштабный анализ

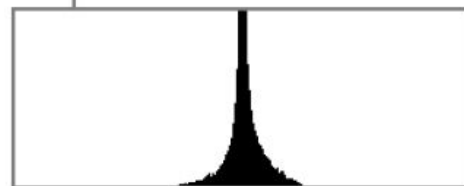
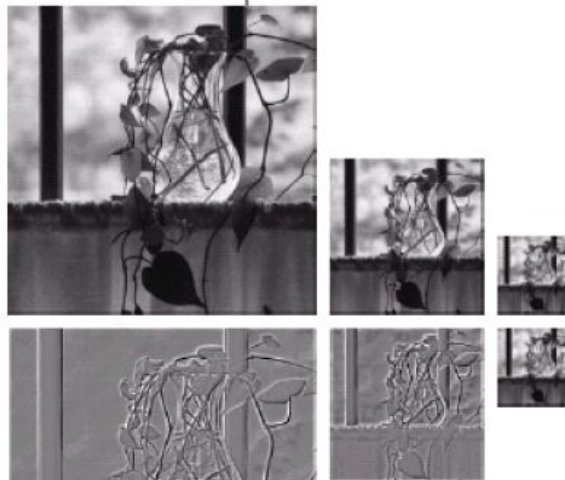
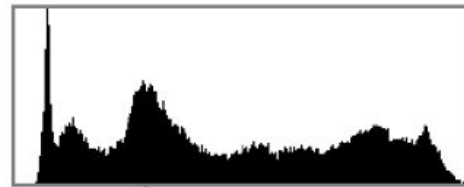
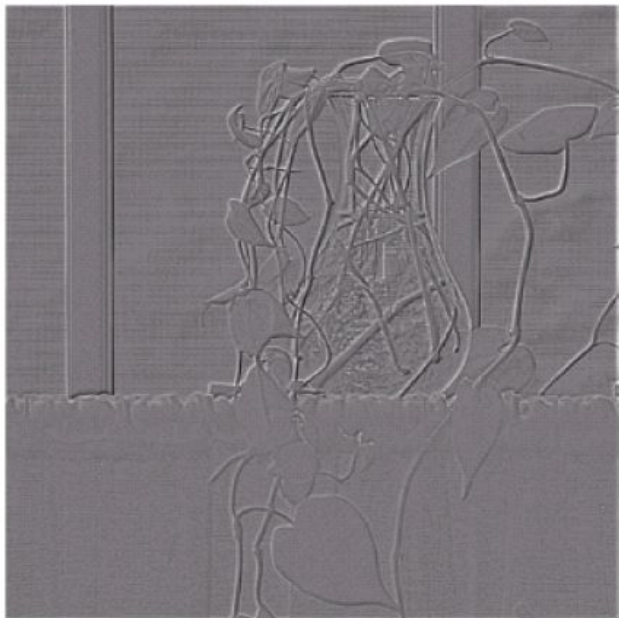
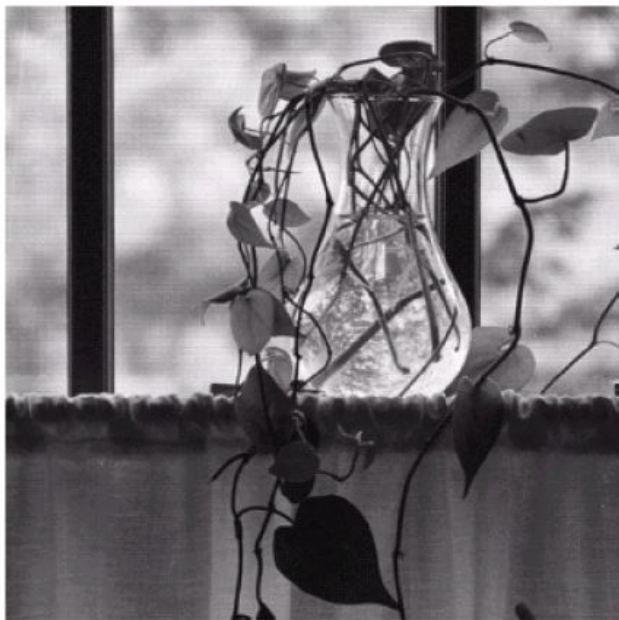
---

- Задача: найти такое представление данных, в которое облегчит анализ изображения
- Решение: разложение сигнала/изображения на набор базовых функций перед обработкой.
- Применение в обработке изображений:
  - Улучшение качества изображения;
  - Восстановление изображения;
  - Кодирование/сжатие изображений;
  - Описание/представление данных.

# Вейвлет-преобразование для изображений

- Создаем пирамиду изображений
- На каждом уровне храним разницу в данных с предыдущим уровнем





a  
b

**FIGURE 7.3** Two image pyramids and their statistics: (a) a Gaussian (approximation) pyramid and (b) a Laplacian (prediction residual) pyramid.

# Преобразование Фурье

---

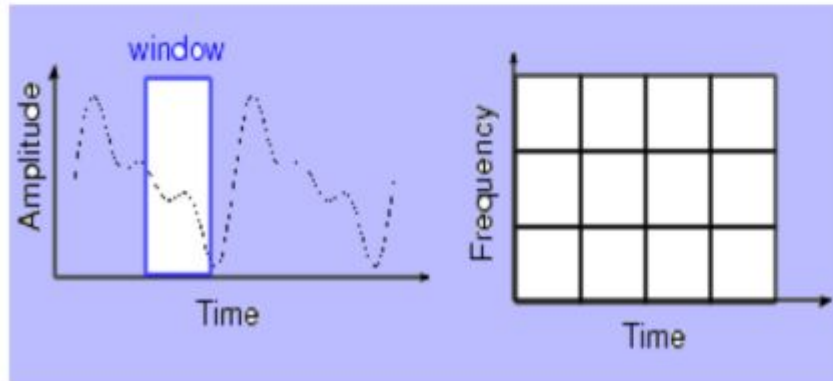
- Исторически: первым для анализа амплитудно-частотной составляющей использовалось преобразование Фурье

$$\hat{f}(\omega) = \int_{-\infty}^{+\infty} f(t)e^{-i\omega t} dt,$$
$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \hat{f}(\omega)e^{i\omega t} d\omega.$$

- Одна частота – для всего временного (пространственного) представления.
- Сложно локализовать особенности поведения функций.

# SHORT TIME FOURIER TRANSFORM (STFT)

- Dennis Gabor (1946) Used STFT
  - To analyze only a small section of the signal at a time -- a technique called *Windowing the Signal*.
- The Segment of Signal is Assumed *Stationary*
- A 3D transform



$$\text{STFT}_x^{(\omega)}(t', f) = \int [x(t) \cdot \omega^*(t-t')] \cdot e^{-j2\pi ft} dt$$

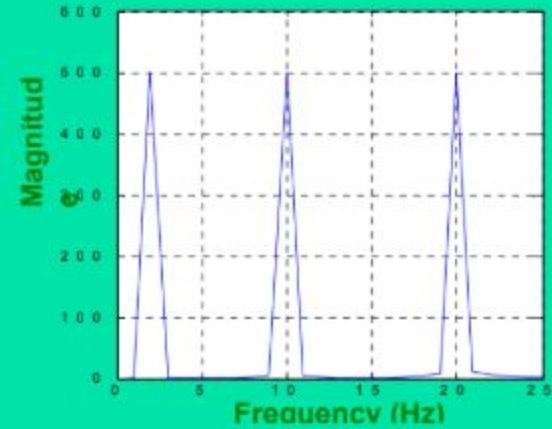
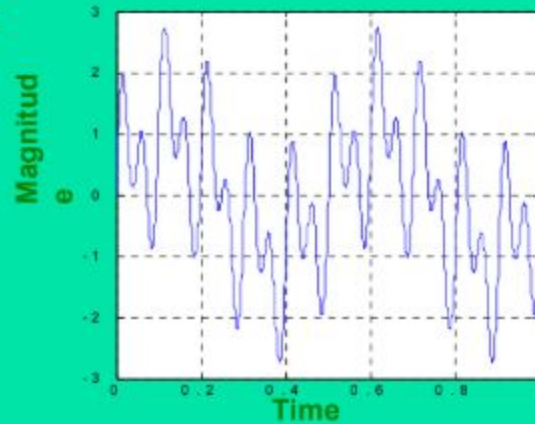
$\omega(t')$ : the window function

**A function of time and frequency**



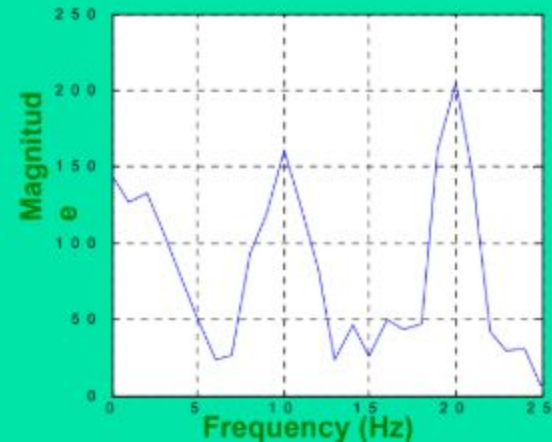
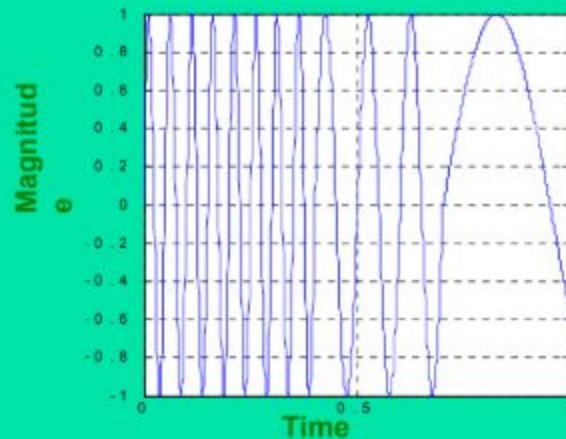
2 Hz + 10 Hz + 20Hz

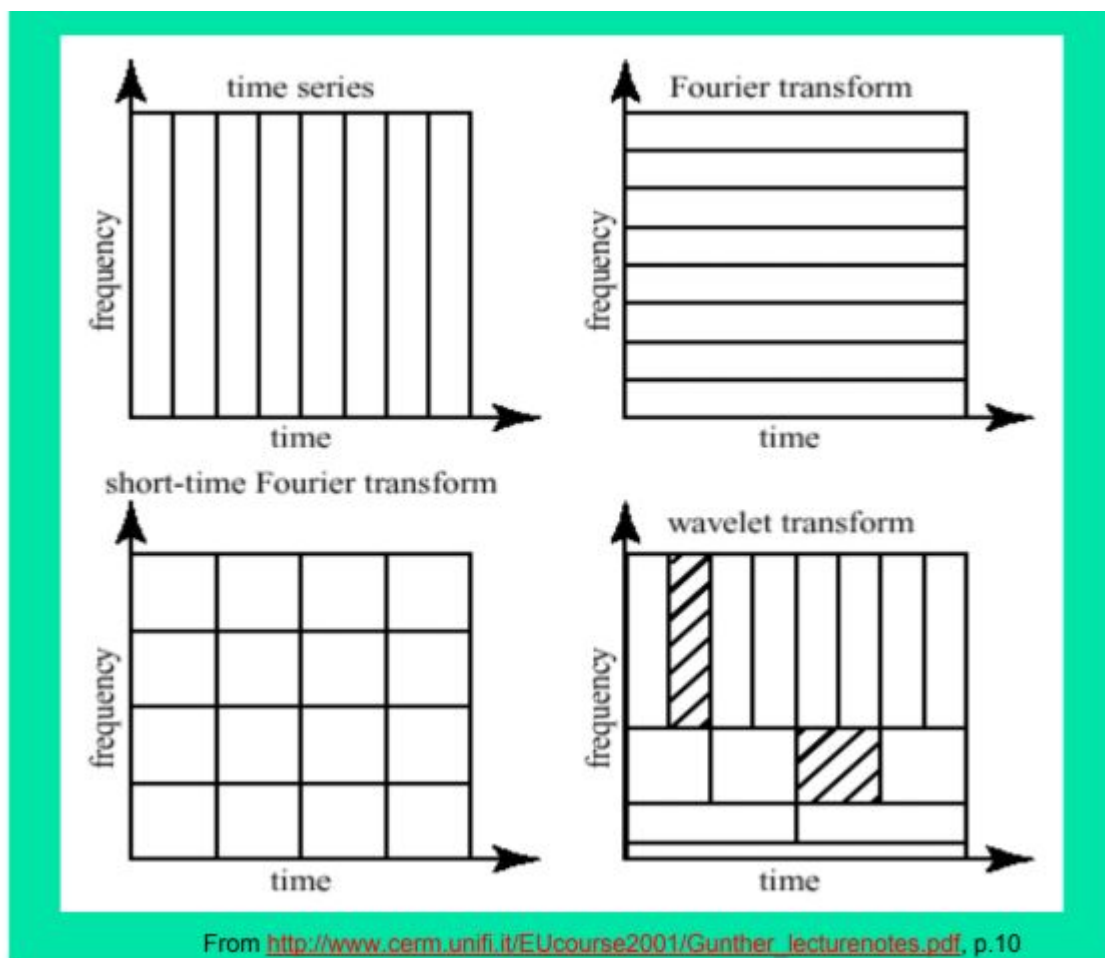
Stationary



0.0-0.4: 2 Hz +  
0.4-0.7: 10 Hz +  
0.7-1.0: 20Hz

Non-Stationary







# Непрерывное вейвлет-преобразование

---

□ Вейвлет-функция:  $\psi \in L^2(\mathbb{R})$

$$\int_{-\infty}^{+\infty} \psi(t) dt = 0.$$

$$\|\psi\| = 1$$

□ Множество частотно-временных базисных элементов получается путем масштабирования базисной вейвлет-функции на  $s$  и сдвигом на  $t$  по  $u$ :

$$\psi_{u,s}(t) = \frac{1}{\sqrt{s}} \psi\left(\frac{t-u}{s}\right)$$

# Вейвлет-преобразование

---

- Непрерывное вейвлет-преобразование раскладывает сигнал в сумму базовых вейвлет-функций, полученных путем масштабирования и сдвига материнской вейвлет-функции.

$$Wf(u, s) = \langle f, \psi_{u,s} \rangle = \int_{-\infty}^{+\infty} f(t) \frac{1}{\sqrt{s}} \psi^* \left( \frac{t-u}{s} \right) dt = 0.$$

- Временное (пространственное) представление пропорционально  $S$ , частотное представление пропорционально  $1/s$ .
- Для многомасштабного анализа:
  - Низкочастотное приближение

# Низкочастотное приближение

---

- Для многомасштабного анализа: низкочастотное приближение сигнала  $f$  в масштабе  $s$

$$Lf(u, s) = \langle f(t), \phi_s(t - u) \rangle$$

$$\phi_s(t) = \frac{1}{\sqrt{s}} \phi\left(\frac{t}{s}\right) \quad |\hat{\phi}(\omega)|^2 = \int_1^{+\infty} |\hat{\psi}(s\omega)|^2 \frac{ds}{s}.$$

- Для одномерного сигнала результат вейвлет-преобразования – двумерное представление.

# Вейвлет-преобразование

- Может быть построено ортогональное вейвлет-преобразование.
- Кратность 2 – удобно.  $s, k$  – целые числа.

$$s = 2^i \text{ and } u = 2^i \cdot k$$

$$\left\{ \psi_{j,n}(t) = \frac{1}{\sqrt{2^j}} \psi \left( \frac{t - 2^j n}{2^j} \right) \right\}_{(j,n) \in \mathbb{Z}^2}$$

- На практике  $\psi$  преобразование проводится для ограниченного набора масштабов. Коэффициенты каждого масштаба характеризуют изменение базовой функции изображения по сравнению с предыдущим масштабом.
- Низкочастотная  $Lf(u, 2^j)$  для самого малого масштаба добавляется в описание сигнала.

# Дискретное вейвлет – преобразование

- Дискретное вейвлет-преобразование для одномерного сигнала  $f=[f_1, f_2, \dots, f_N]$  – набор дискретных преобразований

$$\{L_J f, \{W_j f\}_{j \in [1, J]}\} \quad s = 2^j$$

$$L_J f = L f(2^J n, 2^J)$$

$$W_j f = W f(2^j n, 2^j)$$

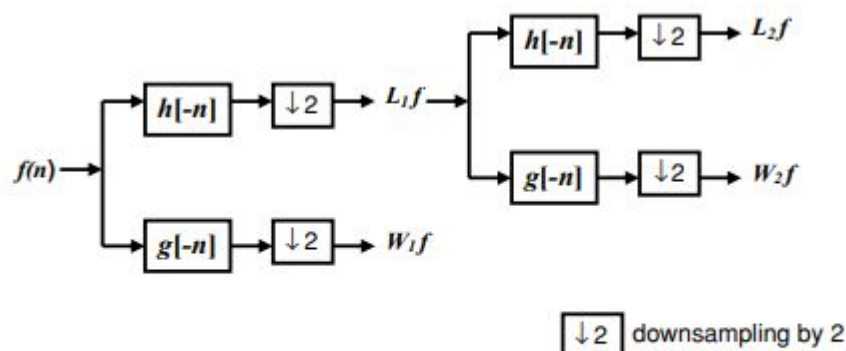


Figure 6.2: Illustration of orthogonal wavelet transform of a discrete signal  $f(n)$  with CMF. A two-level expansion is shown.

# Построение ядер фильтрации

---

- Н – низкочастотный фильтр
- G – высокочастотный фильтр

$$h[n] = \left\langle \frac{1}{\sqrt{2}} \phi \left( \frac{t}{2} \right), \phi(t - n) \right\rangle \quad \text{and} \quad g[n] = \left\langle \frac{1}{\sqrt{2}} \psi \left( \frac{t}{2} \right), \phi(t - n) \right\rangle.$$

# Восстановление сигнала

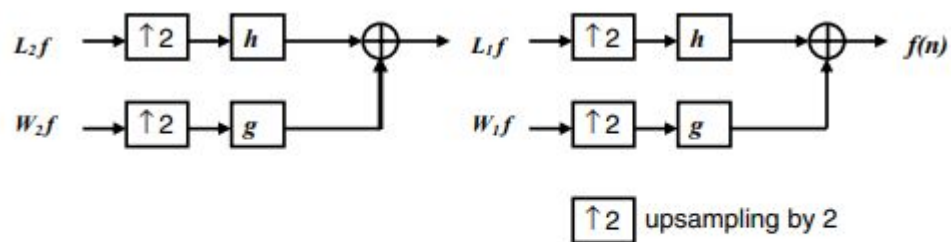
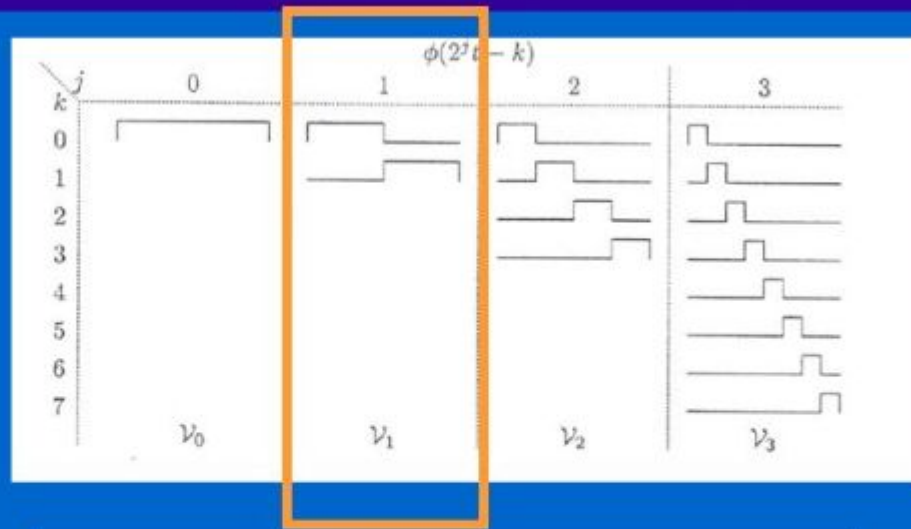


Figure 6.3: Illustration of inverse wavelet transform implemented with CMF. A two-level expansion is shown.

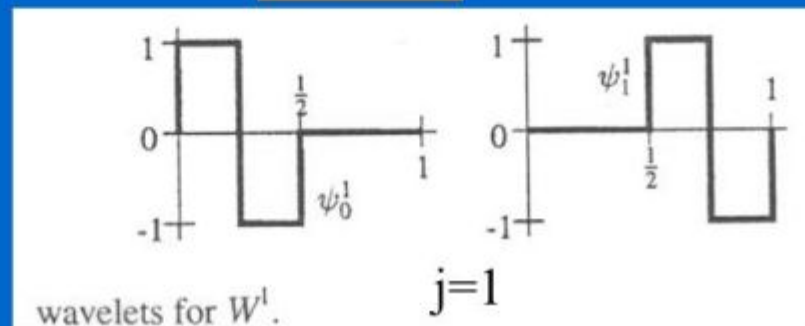
# Вейвлеты Хаара

## 1D Haar Wavelets (cont'd)

basis for  $V_1$ :



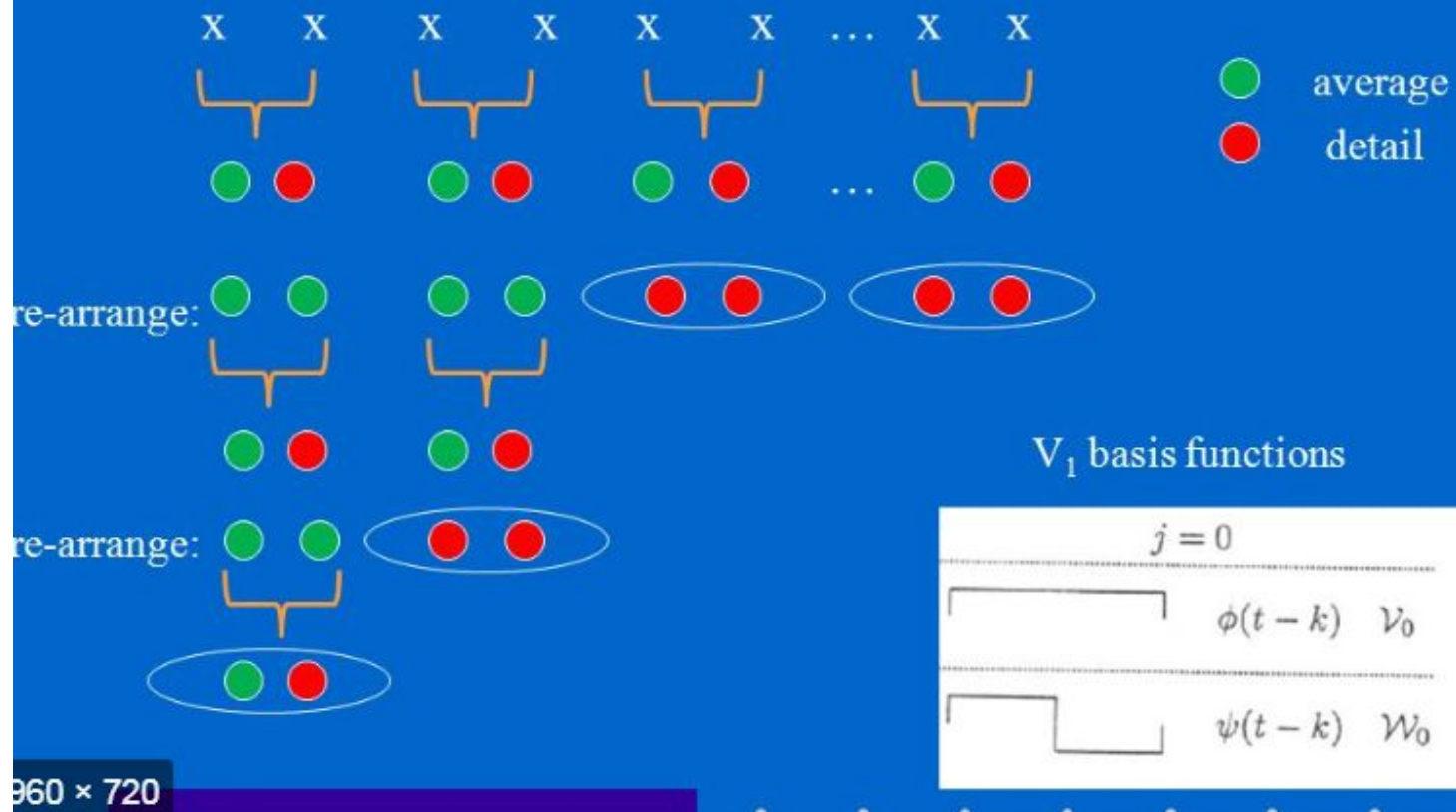
basis  $W_1$ :



Note that inner product is zero!

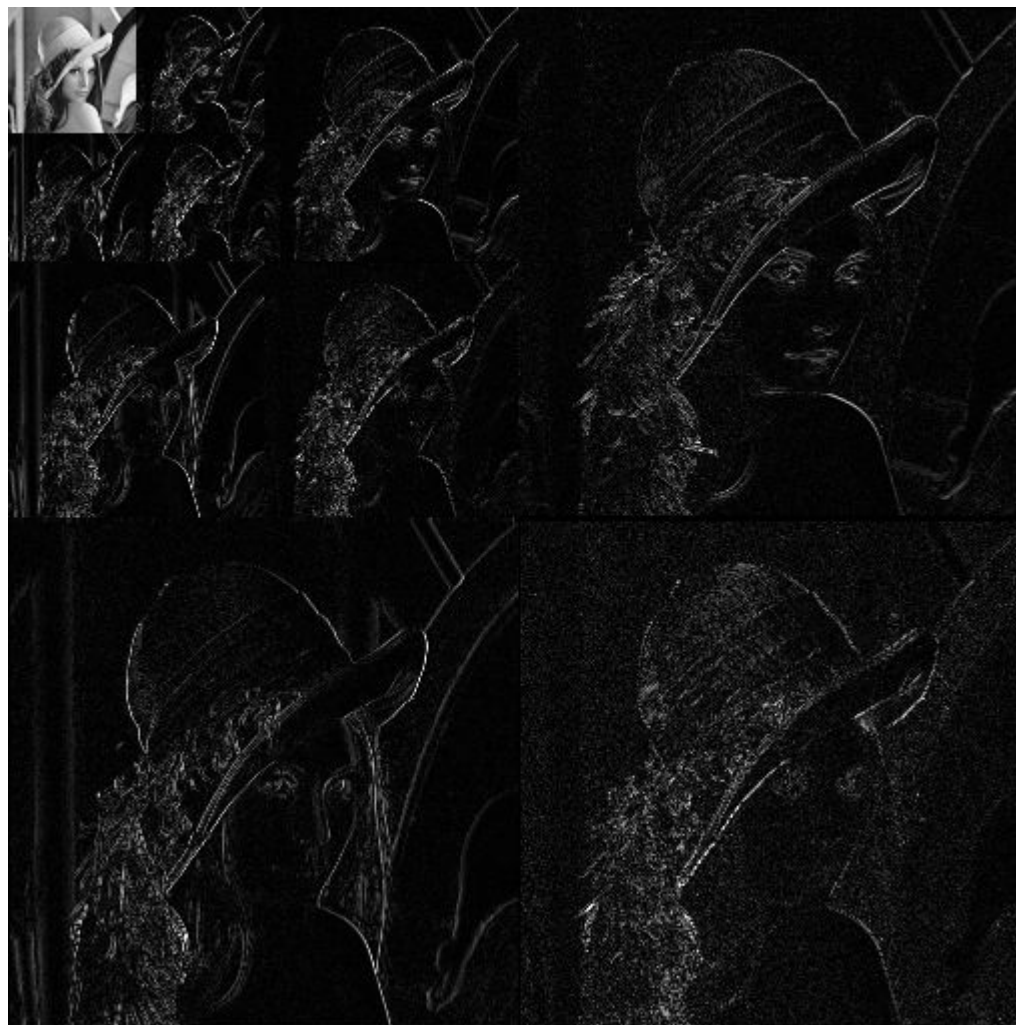


# Convention for illustrating 1D Haar wavelet decomposition



960 × 720

$I_{LL3}$	$I_{LH3}$	$I_{LH2}$	$I_{LH1}$
$I_{HL3}$	$I_{HH3}$		
$I_{HL2}$		$I_{HH2}$	
$I_{HL1}$			$I_{HH1}$



# Вейвлет-преобразование

---

- Общий объем данных после дискретного вейвлет-разложения имеет ту же длину, что и входной сигнал.
- ВП обеспечивает компактное представление сигнала, подходящего для сжатия данных, поскольку вейвлет-преобразование обеспечивает лучшую пространственную частотную локализацию.
- Низкочастотная составляющая – важная, ее не трогаем.
- Высокочастотные коэффициенты – детали, меньше разброс значений.
  - Легче сжимать.
  - Неважные детали обращаем в 0.

# Вейвлет – преобразование

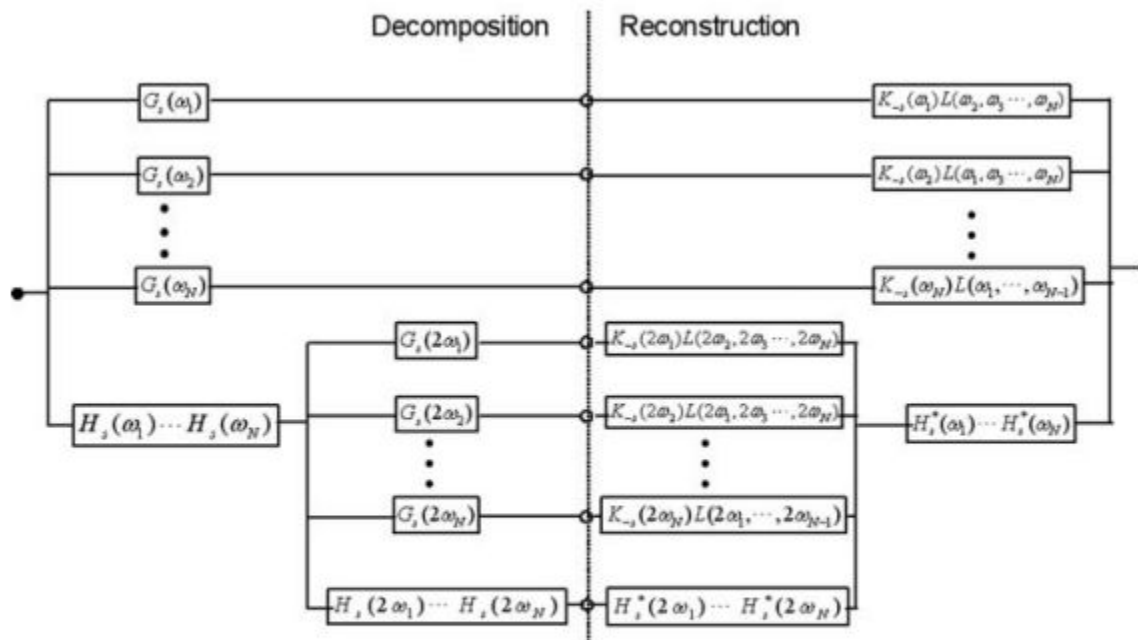


Figure 6.5: Filter bank implementation of a multidimensional discrete dyadic wavelet transform decomposition (left) and reconstruction (right) for two levels of analysis.

# В обработке изображений

---

- Биомедицинские изображения : размерность больше 2
- Вейвлет-преобразование сепарабельно

$$\{S_M s, \{W_m^1 s, W_m^2 s, \dots, W_m^N s\}_{m=\{1, M\}}\},$$

# Удаление шума

- T1,T2,T3 – пороговая фильтрация

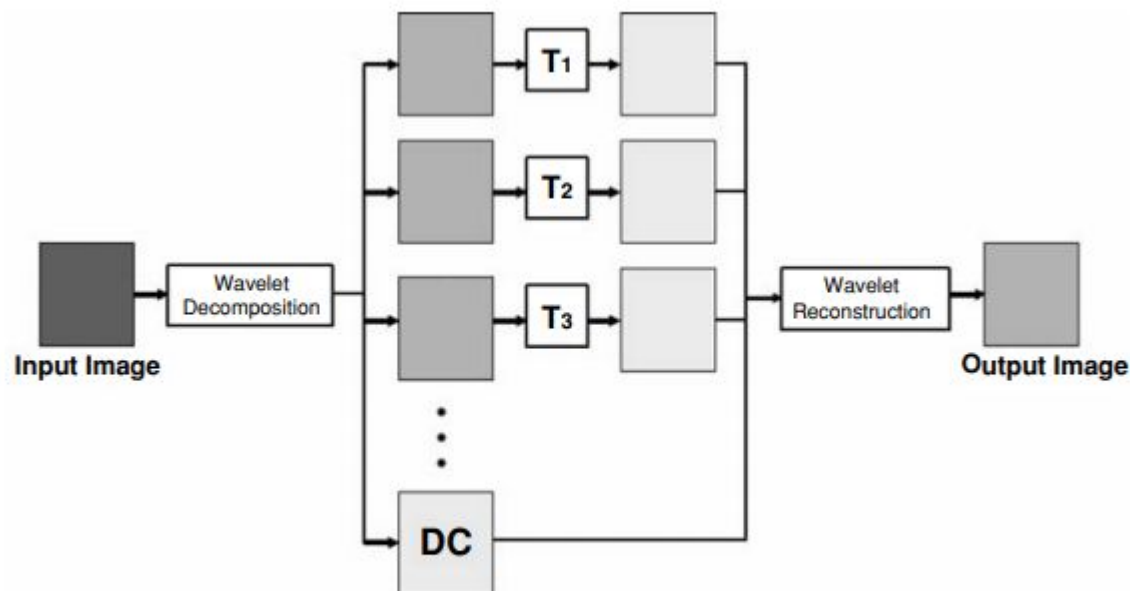


Figure 6.9: A Multiscale framework of denoising and enhancement using discrete dyadic wavelet transform. A three-level decomposition was shown.

# Способы пороговой фильтрации

---

Typical threshold operators for denoising include *hard thresholding*:

$$\rho_T(x) = \begin{cases} x, & \text{if } |x| > T \\ 0, & \text{if } |x| \leq T \end{cases},$$

*soft thresholding* (wavelet shrinkage) [33]:

$$\rho_T(x) = \begin{cases} x - T, & \text{if } x \geq T \\ x + T, & \text{if } x \leq -T, \\ 0, & \text{if } |x| < T \end{cases}$$

and *affine (firm) thresholding* [34]:

$$\rho_T(x) = \begin{cases} x, & \text{if } |x| \geq T \\ 2x + T, & \text{if } -T \leq x \leq -T/2 \\ 2x - T, & \text{if } T/2 \leq x \leq T \\ 0, & \text{if } |x| < T \end{cases}.$$



# Image enhancement

---

- Критерии:
  - Низкочастотная составляющая должна быть размыта сильнее.
  - Реальные границы не должны быть размыты.
- Дополнительные ограничения на функцию:  
монотонно-возрастающая
  - сохранение относительной силы колебаний сигнала и позволяет избежать изменения местоположения экстремумов или создания новых экстремумов
- и антисимметричность ( $E(-x) = -E(x)$ )
  - Позволяет сохранить края.



# Example of enhancement functions

$$E(x) = \begin{cases} x - (K - 1)T, & \text{if } x < -T \\ Kx, & \text{if } |x| \leq T. \\ x + (K - 1)T, & \text{if } x > T \end{cases}$$

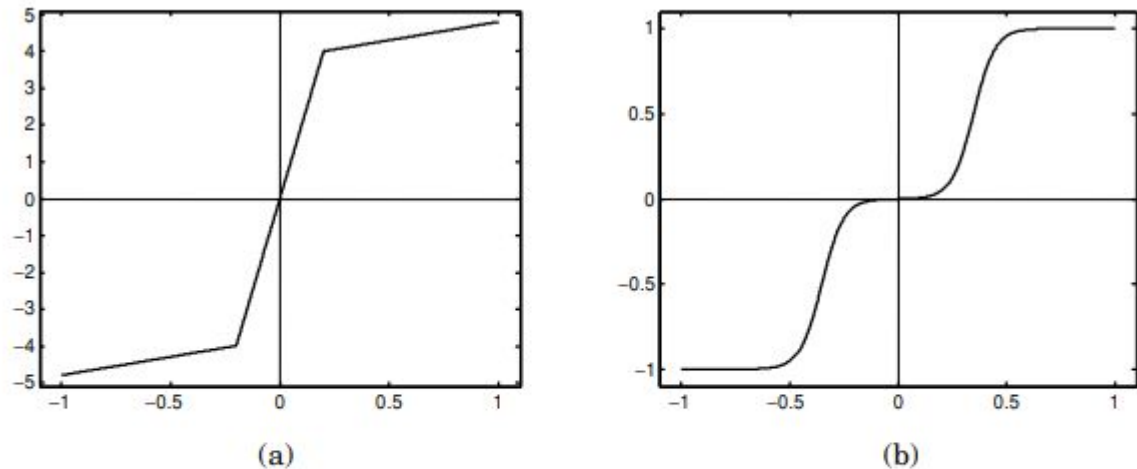


Figure 6.11: Example of enhancement functions, assuming that the input data was normalized to the range of  $[-1, 1]$ . (a) Piecewise linear function,  $T = 0.2$ ,  $K = 20$ . (b) Sigmoid enhancement function,  $b = 0.35$ ,  $c = 20$ . Notice the different scales of the  $y$ -axis for the two plots.

# Шум и увеличение контрастности

---

- Вейвлет-коэффициенты малой величины скорее всего связаны с шумом. Простое усиление коэффициентов по формуле также увеличит шум.
- Этот оператор улучшения поэтому ограничен усилением контраста данных с очень низким уровнем шума, таких как маммография или компьютерная томография.
- Решение: сочетать с оператором шумоподавления.

# Выбор порога фильтрации

---

- Глобальный порог,  $T = \text{const}$
- Зависимость от уровня. Для каждого уровня анализа порог свой.
- Пространственно-адаптивный порог. Зависит от пространственных свойств каждого вейвлет-коэффициента.  $T = T_j(x, y, z)$ .

# Примеры порогов

---

- $T = \sigma \sqrt{2 \log n}$   $n$  – масштаб
  - Не учитывает сами данные, учитывает только размер
  - Просто в реализации
- $T = \sigma T_n$   $T_n$  – мин-макс фильтр
- $T = \sigma^2 / \sigma_X$  [44], where  $\sigma_X$  is the local variance

$$\sigma = \frac{\text{median}(|W_1(x, y, z)|)}{0.6745},$$

where  $W_1$  is the most detailed level of wavelet coefficients.

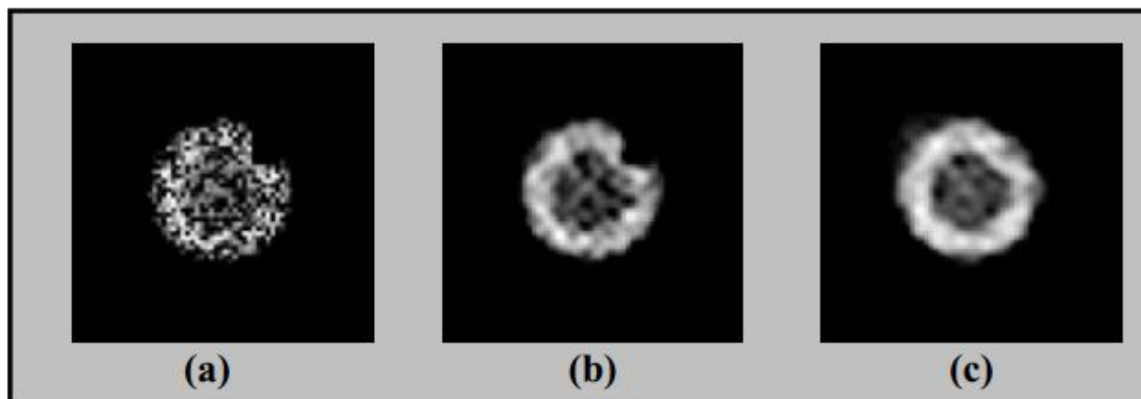


Figure 6.17: (a) Original noisy slice with defect, (b) denoised slice with 3D brushlet analysis, and (c) denoised slice with 4D brushlet analysis.

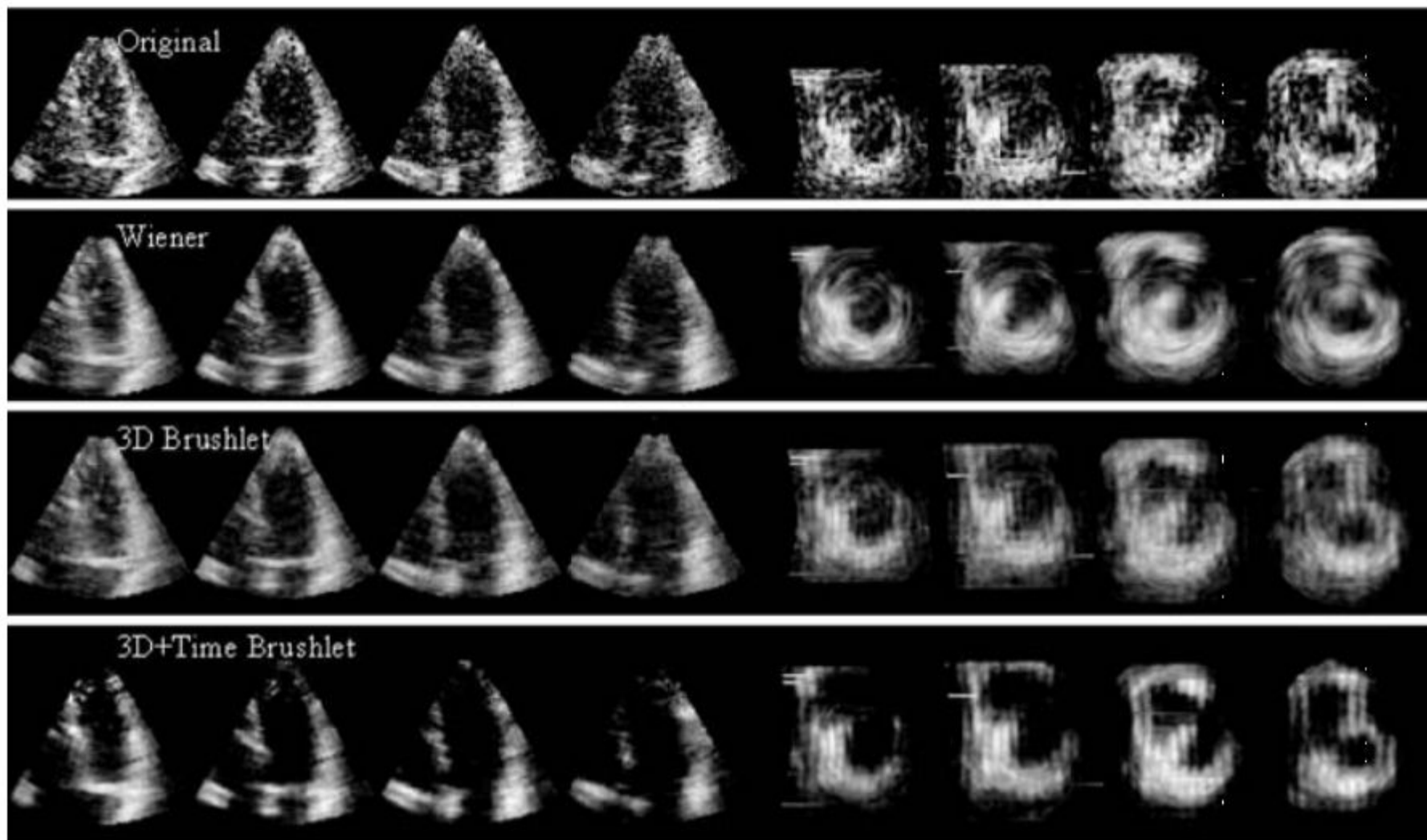


Figure 6.18: Spatio-temporal denoising with brushlet expansion on RT3D ultrasound data illustrated on four long-axis and four short-axis slices.

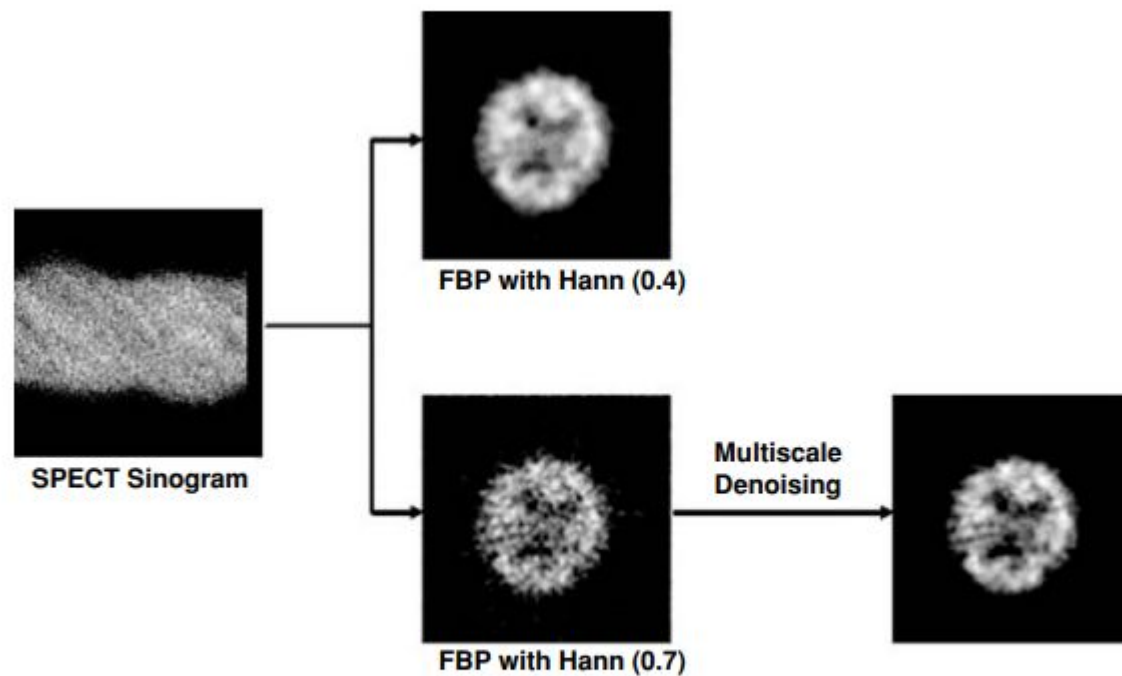


Figure 6.19: Illustration, on a clinical brain SPECT slice, of the combination of multiscale denoising and traditional FBP with higher cut-off frequency to improve tomographic reconstruction.

# Примеры

---

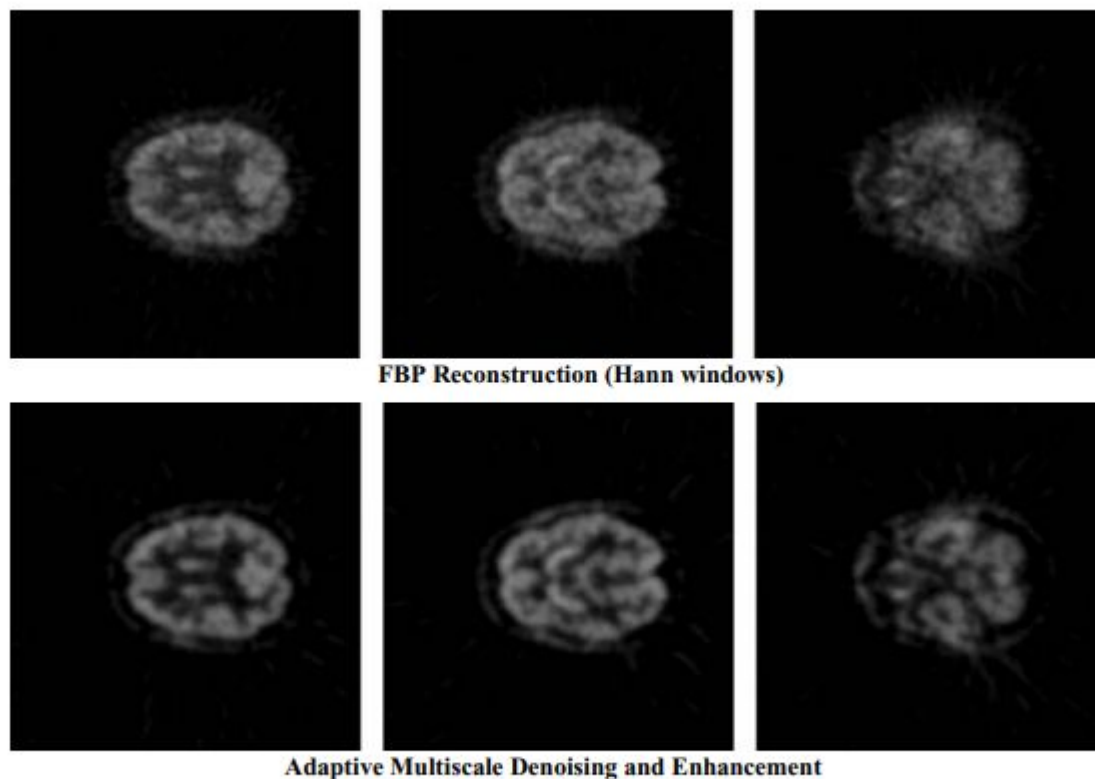


Figure 6.22: Denoising of PET brain data and comparison between unprocessed and multiscale processed images.



The algorithm used for this example from [75] uses the combination of *wavelet packets frame* with Lemarié–Battle filters, *multiscale envelope* features, and a *K-mean* classifier.

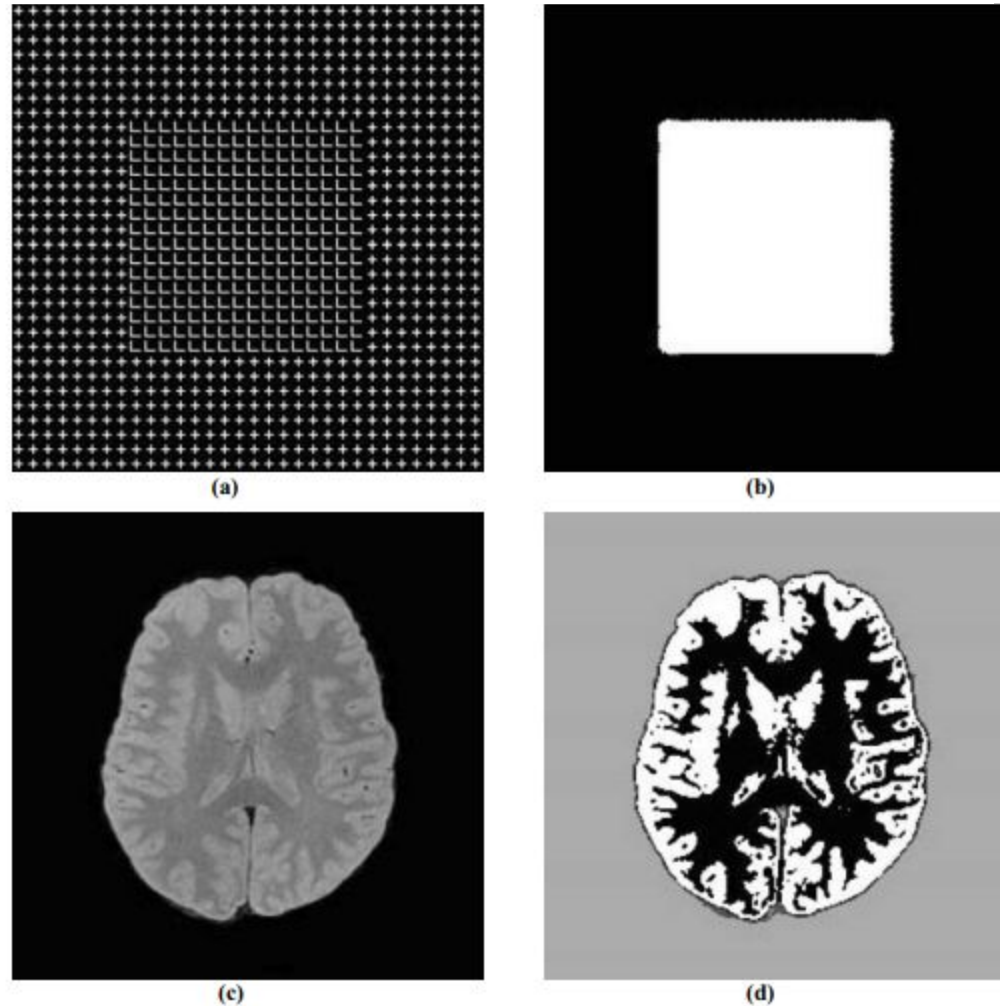


Figure 6.23: Sample results using multiscale texture segmentation. (a) Synthetic texture image. (b) Segmentation result for image (a) with a 2-class labeling. (c) MRI T1 image of a human brain. (d) Segmentation result for image (c) with a 4-class labeling.

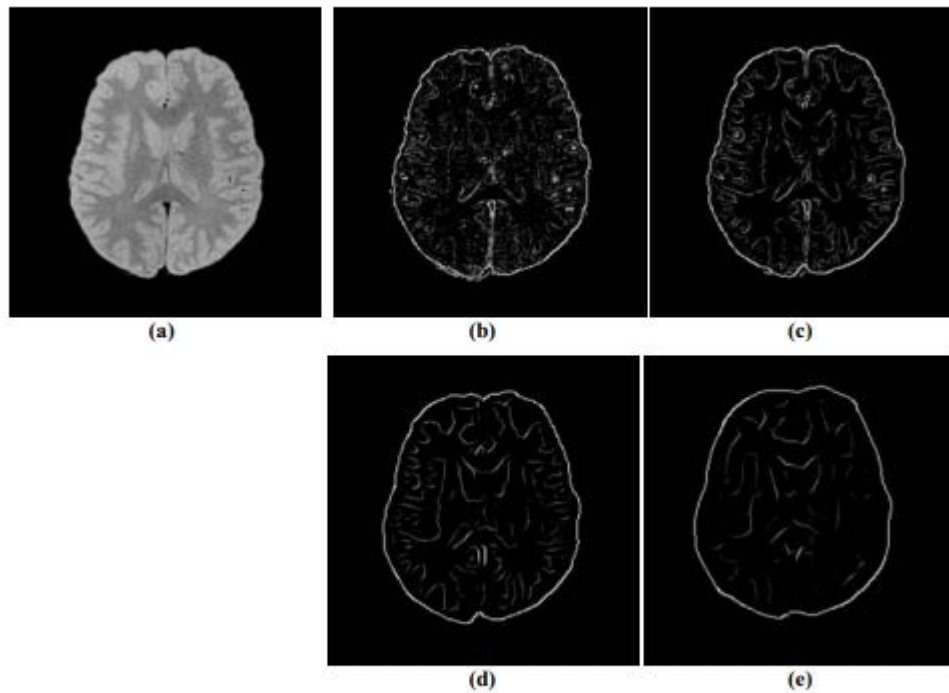
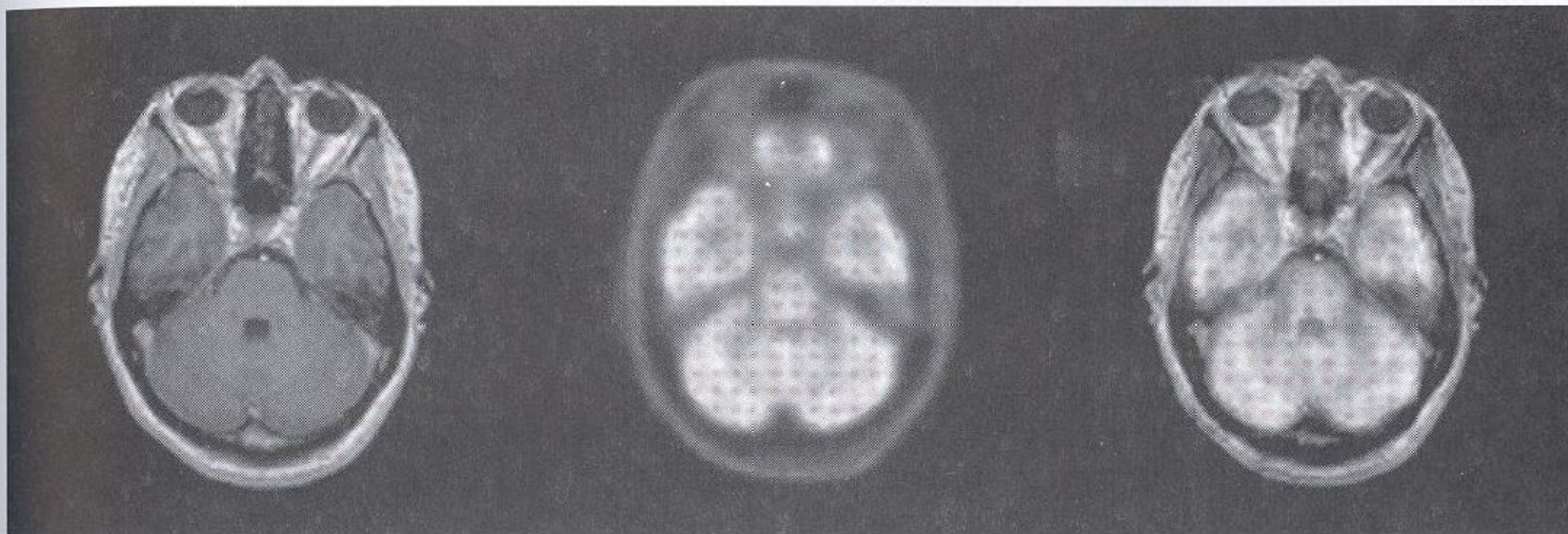
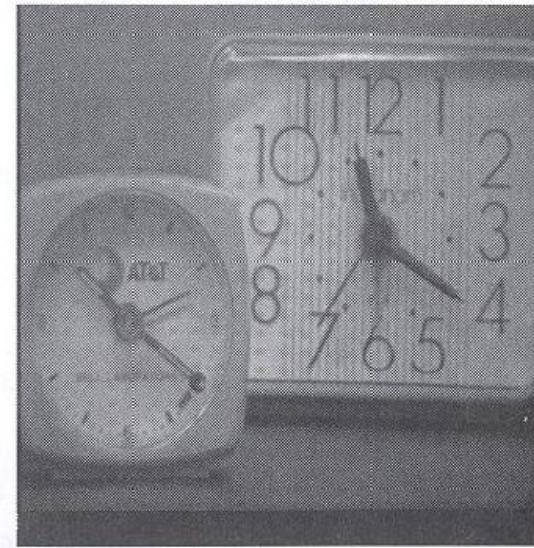
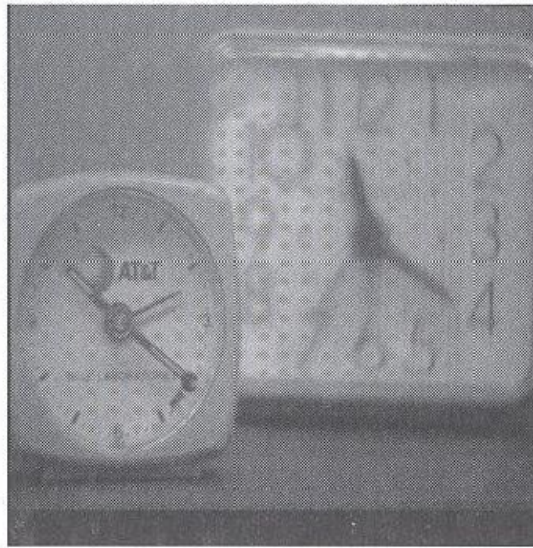
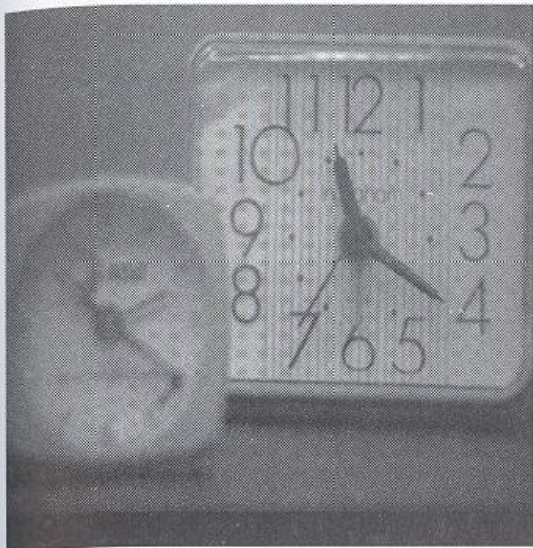


Figure 6.24: Example of a multiscale edge detection method finding local maxima of wavelet modulus, with a first-derivative wavelet function. (a) Input image and (b)–(e) multiscale edge map at expansion scale 1 to 4.





**Figure 14–36** Wavelet transform image fusion: (a), (b) images taken at different focus settings; (c) fused image; (d) MRI image; (e) PET image; (f) fused image (Courtesy Henry Hui Li, reprinted by permission from [28])



**УНИВЕРСИТЕТ  
ЛОБАЧЕВСКОГО**

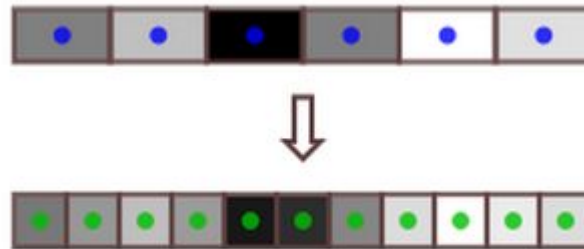
Институт информационных технологий, математики и  
механики  
Кафедра математического обеспечения и  
суперкомпьютерных технологий

## Обработка изображений

Rescaling

Светлана Носова  
svetlana.nosova@itmm.unn.ru

- 
- Size reduction for email sending, publishing on the Internet
  - Increasing of an image resolution or a printer
    - In a case, when printer driver doesn't have enough quality
  - Image resizing for future analysis



# Upsampling

---

- Get a new information from old image

Increasing image resolution

## Downsampling

Decreasing image resolution

Loss of information from old image

The three most important hypotheses for interpolation are:

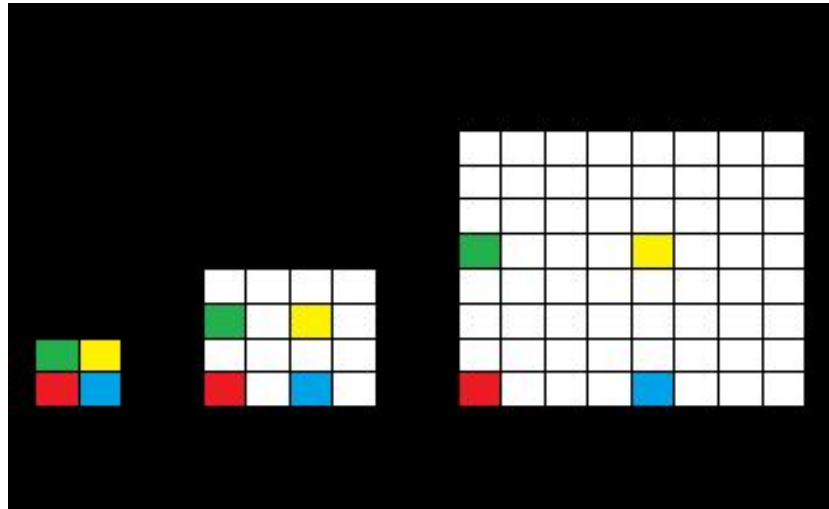
1. The underlying data is continuously defined;
2. Given data samples, it is possible to compute a data value of the underlying continuous function at any abscissa value;
3. The evaluation of the underlying continuous function at the sampling points yields the same value as the data themselves.



---

Can we get same image after applying of such steps:

1. Downsampling
  2. Upsampling
- ?



---

How does downsampling algorithm work? What do you think?





---

We will talk only about separable methods (in our 2D case). It's mean that we have two step in every rescaling algorithm:

1. Calculate rescaling only for rows.
2. Calculate rescaling for columns from step 1.



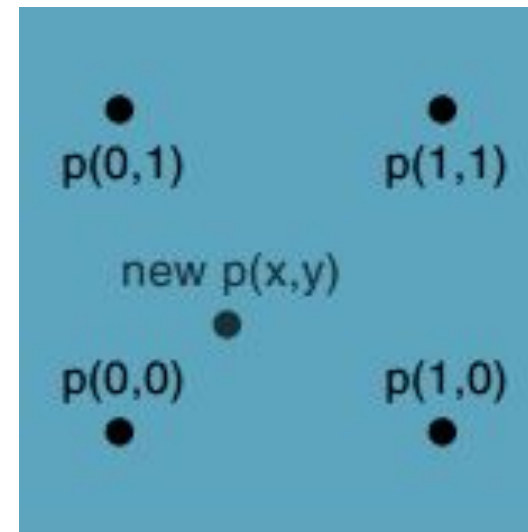
# Bilinear Interpolation

---

We want to add some new pixels to our image  $I(n,m)$ .

We have 3 thesis for this:

1. New pixel must have at least 4 neighbors.
2. The simplest case is if we have this 4 points as vertexes of a square.
3. We have interpolation problem.



---

We can't get new data (information) after applying of an interpolation algorithm. The best thing that we can do is a reduction of artifacts.





LOBACHEVSKY STATE UNIVERSITY  
of NIZHNI NOVGOROD  
National Research University

---

# Linear interpolation methods



# Nearest Neighbor method

---

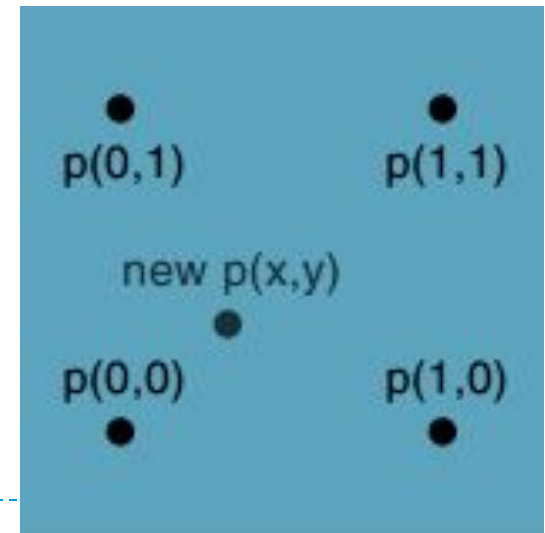
The  $p(x,y)$  color is a color of the nearest pixel from  $p(0,0)$ ,  $p(0,1)$ ,  $p(1,1)$  and  $p(1,0)$ .

▣ **Advantages:**

- ✓ Very fast algorithm

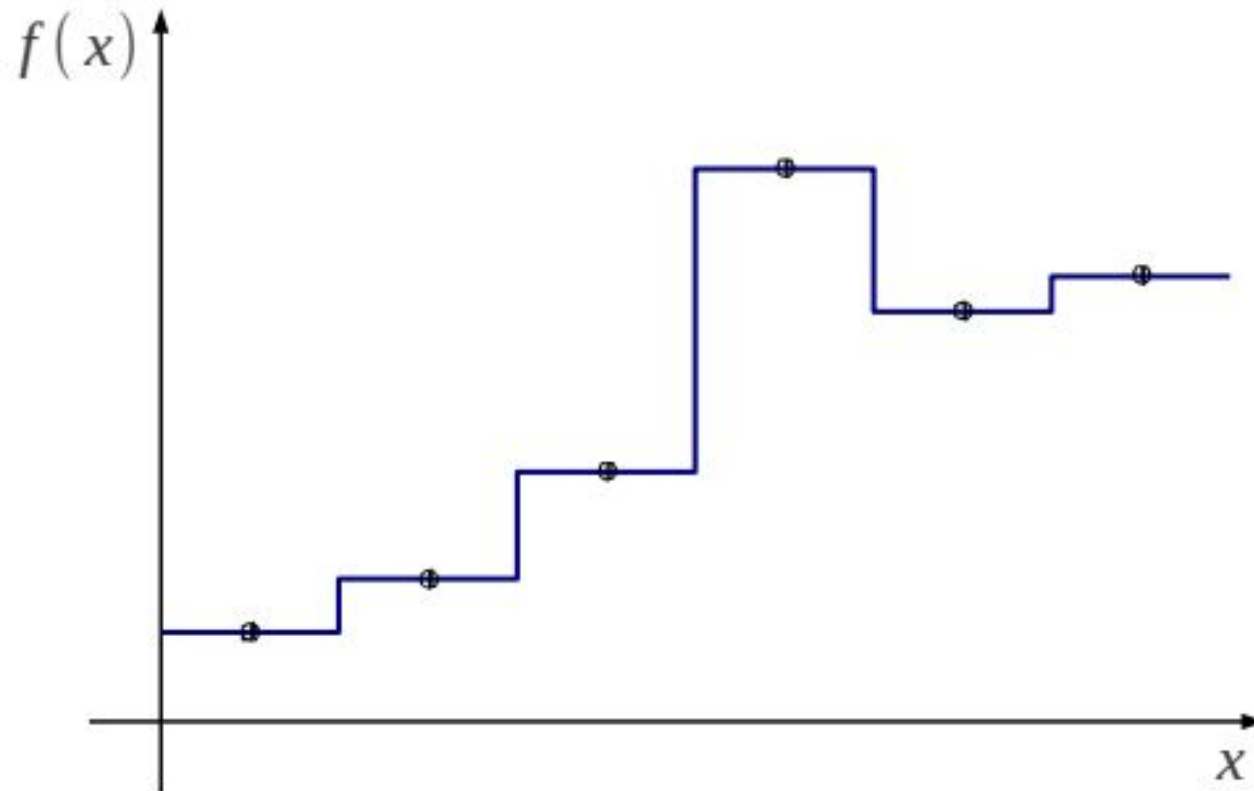
▣ **Drawbacks:**

- Upsampling – blockiness of a resulting image.
- Downsampling – graininess of the resulting image.



# Nearest Neighbor method

---



# Bilinear Interpolation

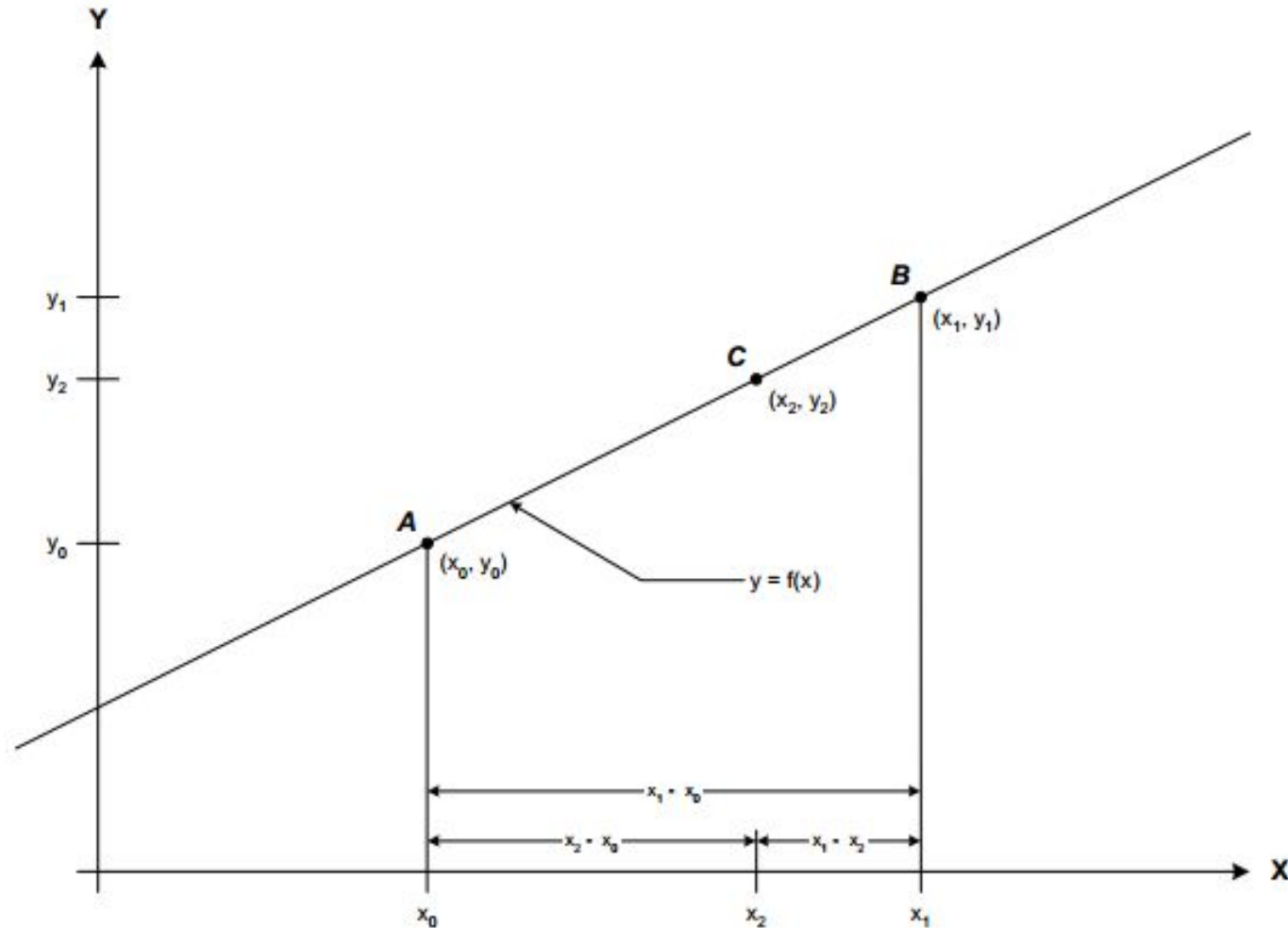
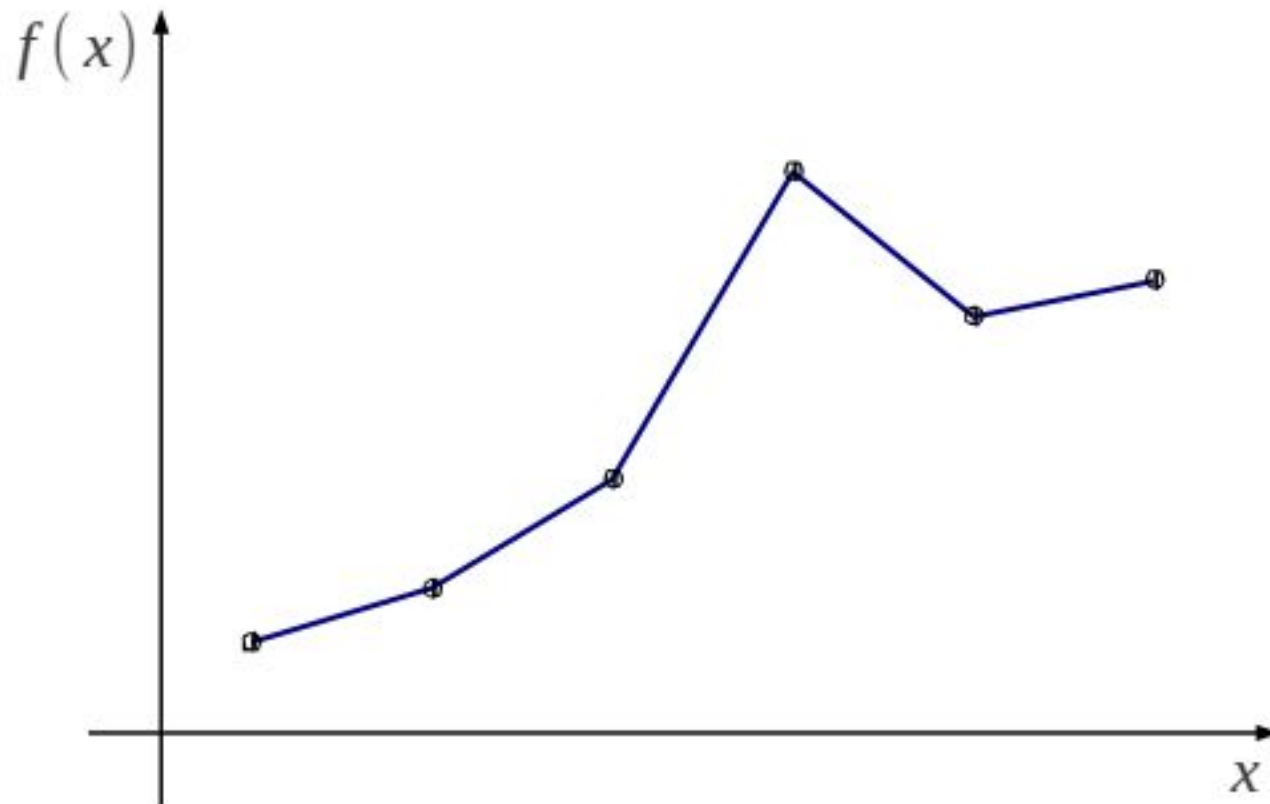


Figure 1: Linear interpolation: given points  $A$  and  $B$  and  $x_2$ , compute  $y_2$ , yielding point  $C$ .

# Bilinear Interpolation

---



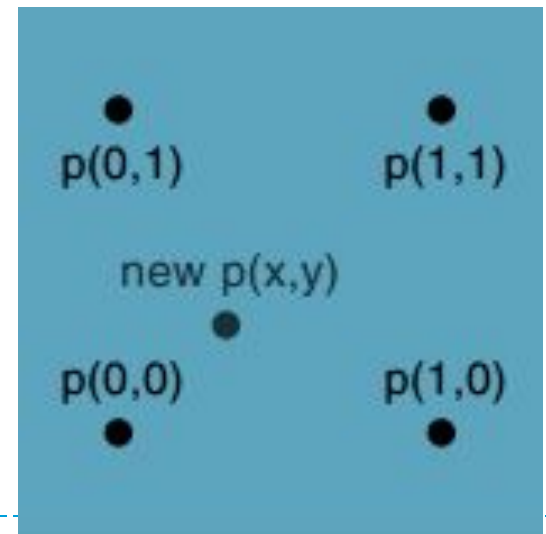
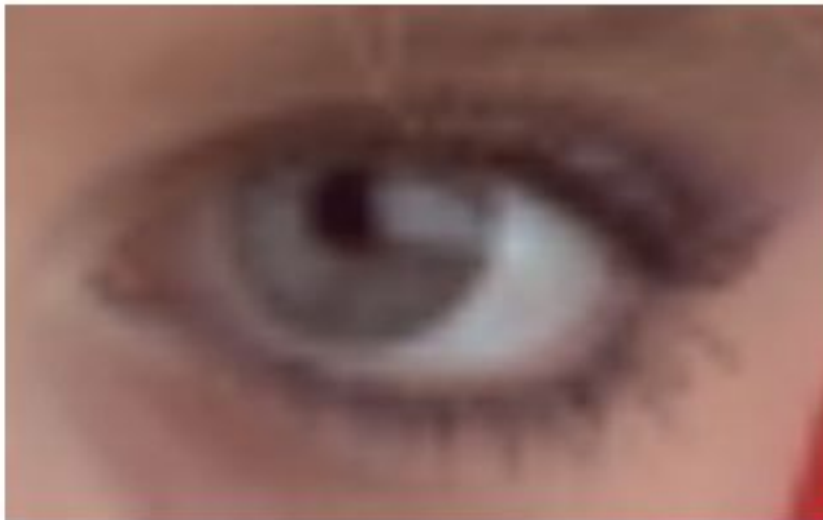


# Bilinear Interpolation

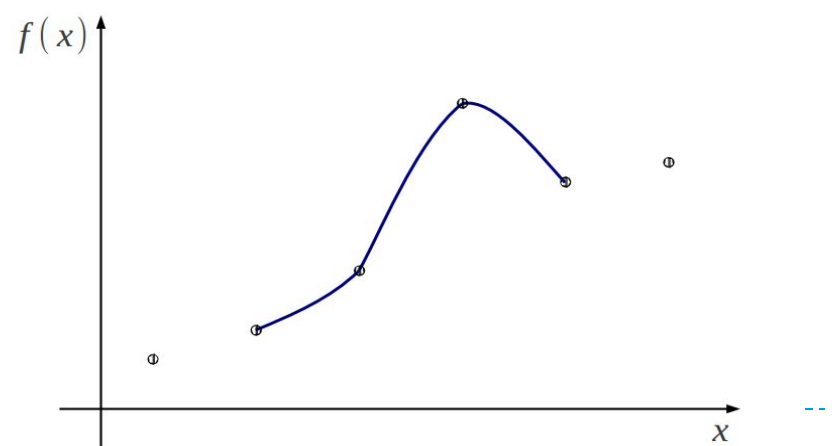
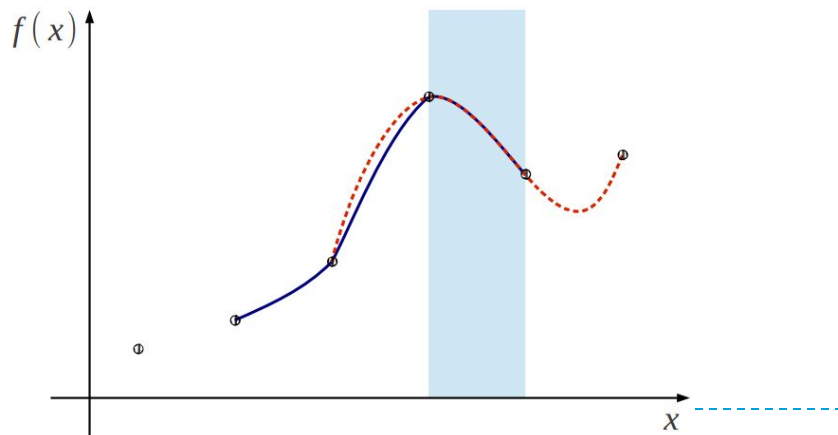
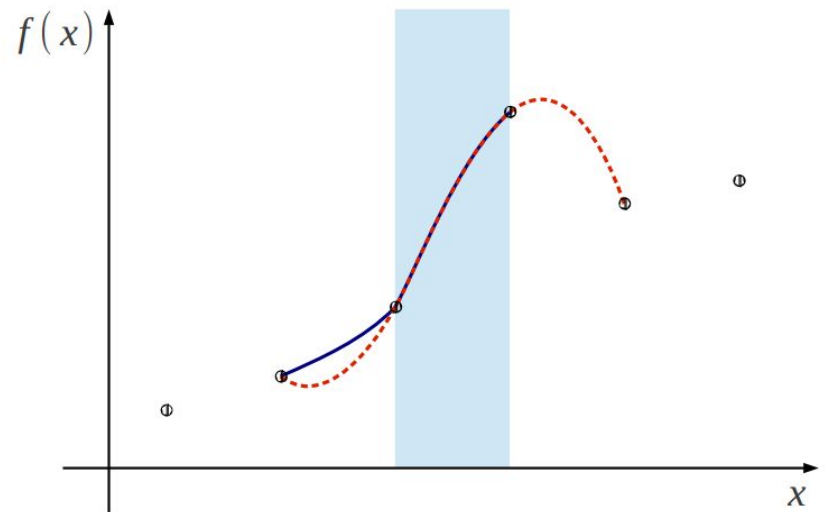
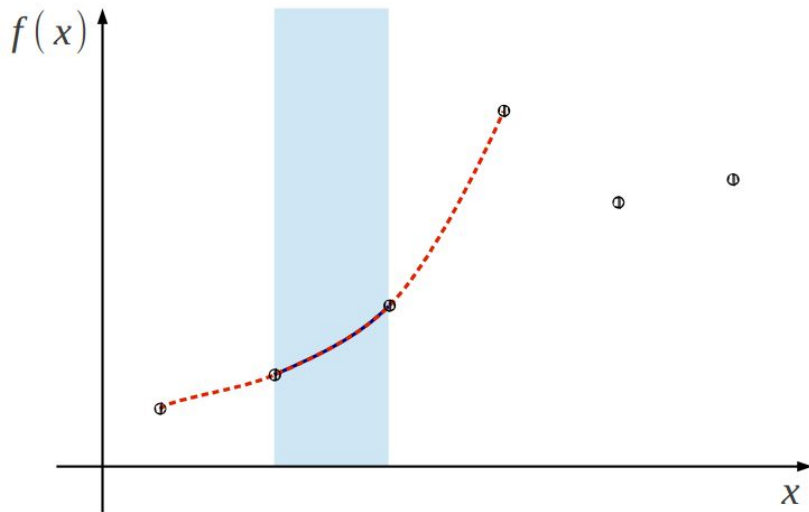
---

The rule of new pixel calculating:

$$f(x,y) = f(0,0)(1-x)(1-y) + f(1,0)x(1-y) + f(0,1)(1-x)y + f(1,1)xy$$



# Bicubic Spline interpolation



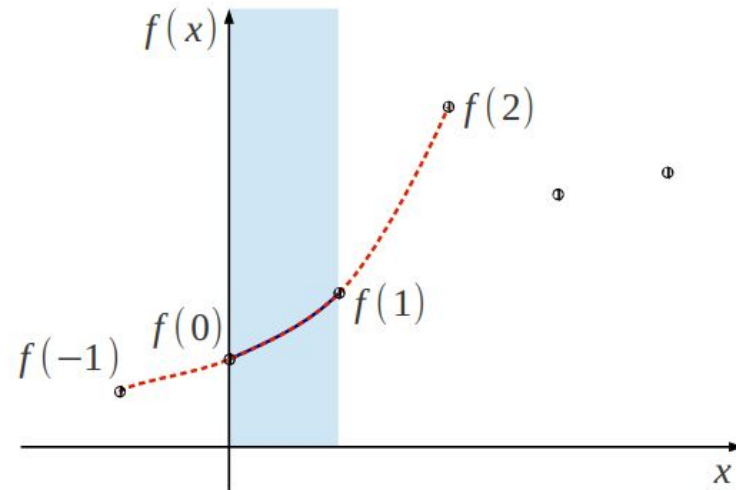
# Bicubic Spline interpolation

In this case you need 4 points in 1D case ( and 16 in 2D case) to calculate new point.

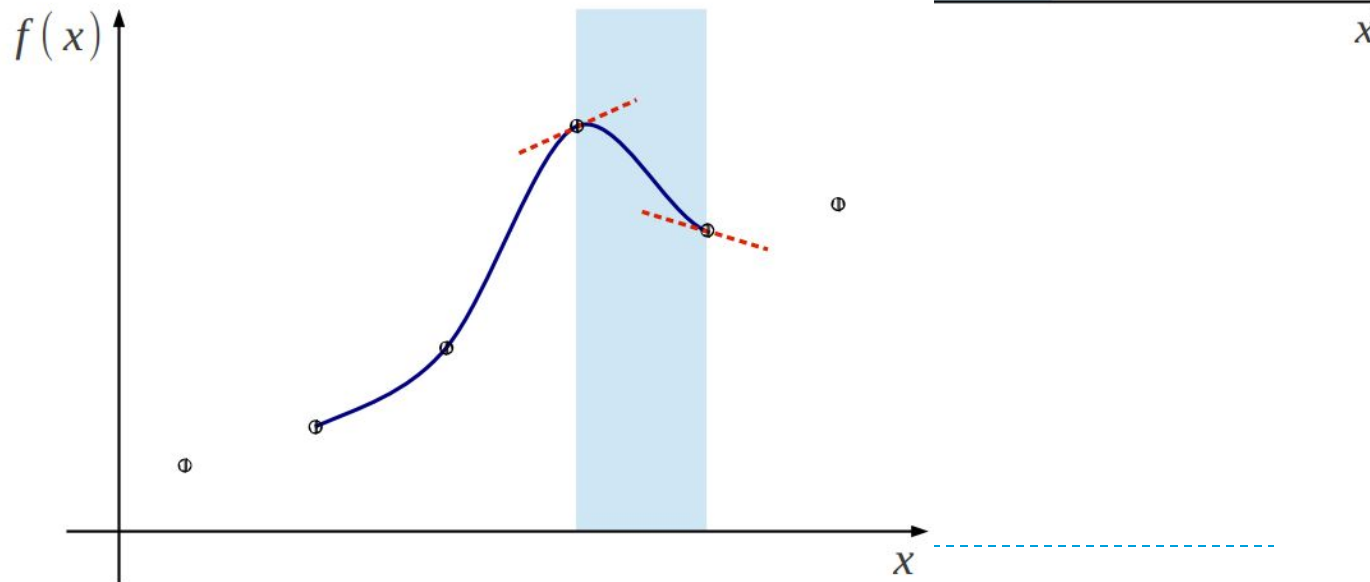
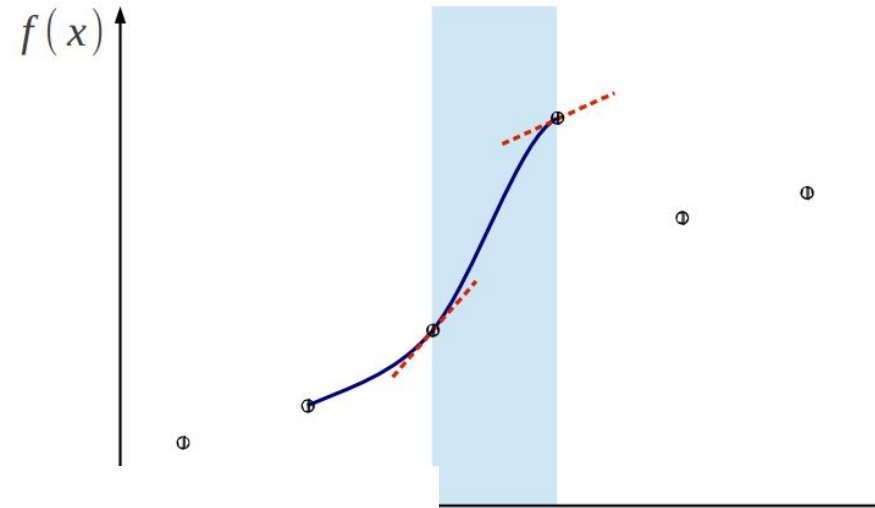
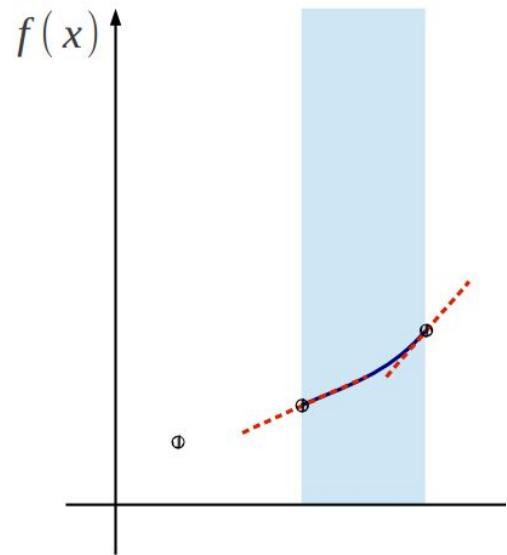
You can see the model on this slide.

But first of all you should calculate  $a_0$ ,  $a_1$ ,  $a_2$  and  $a_3$  from linear equations system.

- Model: 
$$f(x) = \sum_{i=0}^3 a_i x^i = a_3 x^3 + a_2 x^2 + a_1 x^1 + a_0$$
- $$\begin{cases} f(-1) = a_3 \cdot (-1)^3 + a_2 \cdot (-1)^2 + a_1 \cdot (-1)^1 + a_0 \cdot (-1)^0 \\ f(0) = a_3 \cdot 0^3 + a_2 \cdot 0^2 + a_1 \cdot 0^1 + a_0 \cdot 0^0 \\ f(1) = a_3 \cdot 1^3 + a_2 \cdot 1^2 + a_1 \cdot 1^1 + a_0 \cdot 1^0 \\ f(2) = a_3 \cdot 2^3 + a_2 \cdot 2^2 + a_1 \cdot 2^1 + a_0 \cdot 2^0 \end{cases}$$



# Bicubic Spline interpolation



# Bicubic Spline interpolation

---

- Model:

- $f(x) = \sum_{i=0}^3 a_i x^i = a_3 x^3 + a_2 x^2 + a_1 x^1 + a_0$

- $f'(x) = \sum_{i=1}^3 i a_i x^{i-1} = 3a_3 x^2 + 2a_2 x^1 + a_1$

- $$\begin{cases} f(0) = a_3 \cdot 0^3 & + a_2 \cdot 0^2 & + a_1 \cdot 0^1 & + a_0 \cdot 0^0 \\ f(1) = a_3 \cdot 1^3 & + a_2 \cdot 1^2 & + a_1 \cdot 1^1 & + a_0 \cdot 1^0 \\ f'(0) = a_3 \cdot 3 \cdot 0^2 & + a_2 \cdot 2 \cdot 0^1 & + a_1 \cdot 1 \cdot 0^0 \\ f'(1) = a_3 \cdot 3 \cdot 1^2 & + a_2 \cdot 2 \cdot 1^1 & + a_1 \cdot 1 \cdot 1^0 \end{cases}$$

- $$\begin{cases} f(0) = f(0) \\ f(1) = f(1) \\ f'(0) \approx \frac{1}{2} f(1) - \frac{1}{2} f(-1) \\ f'(1) \approx \frac{1}{2} f(2) - \frac{1}{2} f(0) \end{cases}$$



# Bicubic Spline interpolation

$$\begin{aligned}
 a_{00} &= w_0, \\
 a_{01} &= y_0, \\
 a_{02} &= -3w_0 + 3w_2 - 2y_0 - y_2 \\
 a_{03} &= 2w_0 - 2w_2 + y_0 + y_2 \\
 a_{10} &= x_0 \\
 a_{11} &= z_0 \\
 a_{12} &= -3x_0 + 3x_2 - 2z_0 - z_2 \\
 a_{13} &= 2x_0 - 2x_2 + z_0 + z_2 \\
 a_{20} &= -3w_0 + 3w_1 - 2x_0 - x_1 \\
 a_{21} &= -3y_0 + 3y_1 - 2z_0 - z_1 \\
 a_{22} &= 9w_0 - 9w_1 - 9w_2 + 9w_3 + 6x_0 + 3x_1 + \\
 &\quad -6x_2 - 3x_3 + 6y_0 - 6y_1 + 3y_2 - 3y_3 + \\
 &\quad 4z_0 + 2z_1 + 2z_2 + z_3 \\
 a_{23} &= -6w_0 + 6w_1 + 6w_2 - 6w_3 - 4x_0 - 2x_1 + \\
 &\quad 4x_2 + 2x_3 - 3y_0 + 3y_1 - 3y_2 + 3y_3 + \\
 &\quad -2z_0 - z_1 - 2z_2 - z_3 \\
 a_{30} &= 2w_0 - 2w_1 + x_0 + x_1 \\
 a_{31} &= 2y_0 - 2y_1 + z_0 + z_1 \\
 a_{32} &= -6w_0 + 6w_1 + 6w_2 - 6w_3 - 3x_0 - 3x_1 + \\
 &\quad 3x_2 + 3x_3 - 4y_0 + 4y_1 - 2y_2 + 2y_3 + \\
 &\quad -2z_0 - 2z_1 - z_2 - z_3 \\
 a_{33} &= 4w_0 - 4w_1 - 4w_2 + 4w_3 + 2x_0 + 2x_1 + \\
 &\quad -2x_2 - 2x_3 + 2y_0 - 2y_1 + 2y_2 - 2y_3 + \\
 &\quad z_0 + z_1 + z_2 + z_3
 \end{aligned}$$

$$\begin{aligned}
 w_0 &= f(0,0) = a_{00} \\
 w_1 &= f(1,0) = a_{00} + a_{10} + a_{20} + a_{30} \\
 w_2 &= f(0,1) = a_{00} + a_{01} + a_{02} + a_{03} \\
 w_3 &= f(1,1) = a_{00} + a_{10} + a_{20} + a_{30} + \\
 &\quad a_{01} + a_{11} + a_{21} + a_{31} + \\
 &\quad a_{02} + a_{12} + a_{22} + a_{32} + \\
 &\quad a_{03} + a_{13} + a_{23} + a_{33} \\
 x_0 &= f_x(0,0) = a_{10} \\
 x_1 &= f_x(1,0) = a_{10} + 2a_{20} + 3a_{30} \\
 x_2 &= f_x(0,1) = a_{10} + a_{11} + a_{12} + a_{13} \\
 x_3 &= f_x(1,1) = 1*(a_{10} + a_{11} + a_{12} + a_{13}) + \\
 &\quad 2*(a_{20} + a_{21} + a_{22} + a_{23}) + \\
 &\quad 3*(a_{30} + a_{31} + a_{32} + a_{33}) \\
 y_0 &= f_y(0,0) = a_{01} \\
 y_1 &= f_y(1,0) = a_{01} + a_{11} + a_{21} + a_{31} \\
 y_2 &= f_y(0,1) = a_{01} + 2a_{02} + 3a_{03} \\
 y_3 &= f_y(1,1) = 1*(a_{01} + a_{11} + a_{21} + a_{31}) + \\
 &\quad 2*(a_{02} + a_{12} + a_{22} + a_{32}) + \\
 &\quad 3*(a_{03} + a_{13} + a_{23} + a_{33}) \\
 z_0 &= f_{xy}(0,0) = a_{11} \\
 z_1 &= f_{xy}(1,0) = a_{11} + 2a_{21} + 3a_{31} \\
 z_2 &= f_{xy}(0,1) = a_{11} + 2a_{12} + 3a_{13} \\
 z_3 &= f_{xy}(1,1) = 1*a_{11} + 2*a_{12} + 3*a_{13} + \\
 &\quad 2*a_{21} + 4*a_{22} + 6*a_{23} + \\
 &\quad 3*a_{31} + 6*a_{32} + 9*a_{33}
 \end{aligned}$$

$$\begin{aligned}
 p(x,y) &= a_{00}x^0y^0 + a_{01}x^0y^1 + \\
 &\quad a_{02}x^0y^2 + a_{03}x^0y^3 + \\
 &\quad a_{10}x^1y^0 + a_{11}x^1y^1 + \\
 &\quad a_{12}x^1y^2 + a_{13}x^1y^3 + \\
 &\quad a_{20}x^2y^0 + a_{21}x^2y^1 + \\
 &\quad a_{22}x^2y^2 + a_{23}x^2y^3 + \\
 &\quad a_{30}x^3y^0 + a_{31}x^3y^1 + \\
 &\quad a_{32}x^3y^2 + a_{33}x^3y^3
 \end{aligned}$$



# Bicubic Spline interpolation

---

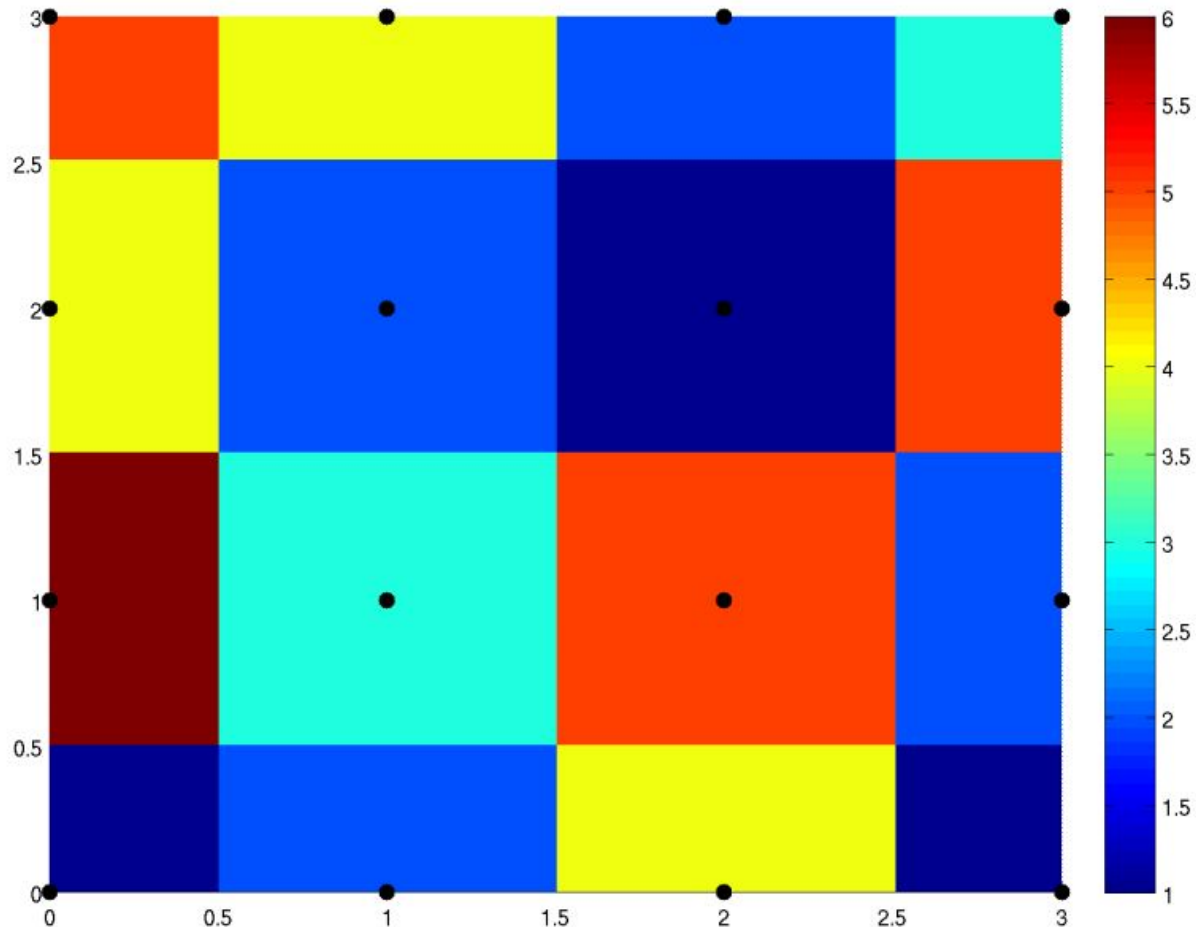
Drawbacks:

- 10 times slower than the nearest neighbor interpolation.
- 7 times slower than bilinear interpolation.



# Интерполяция методом ближайшего соседа

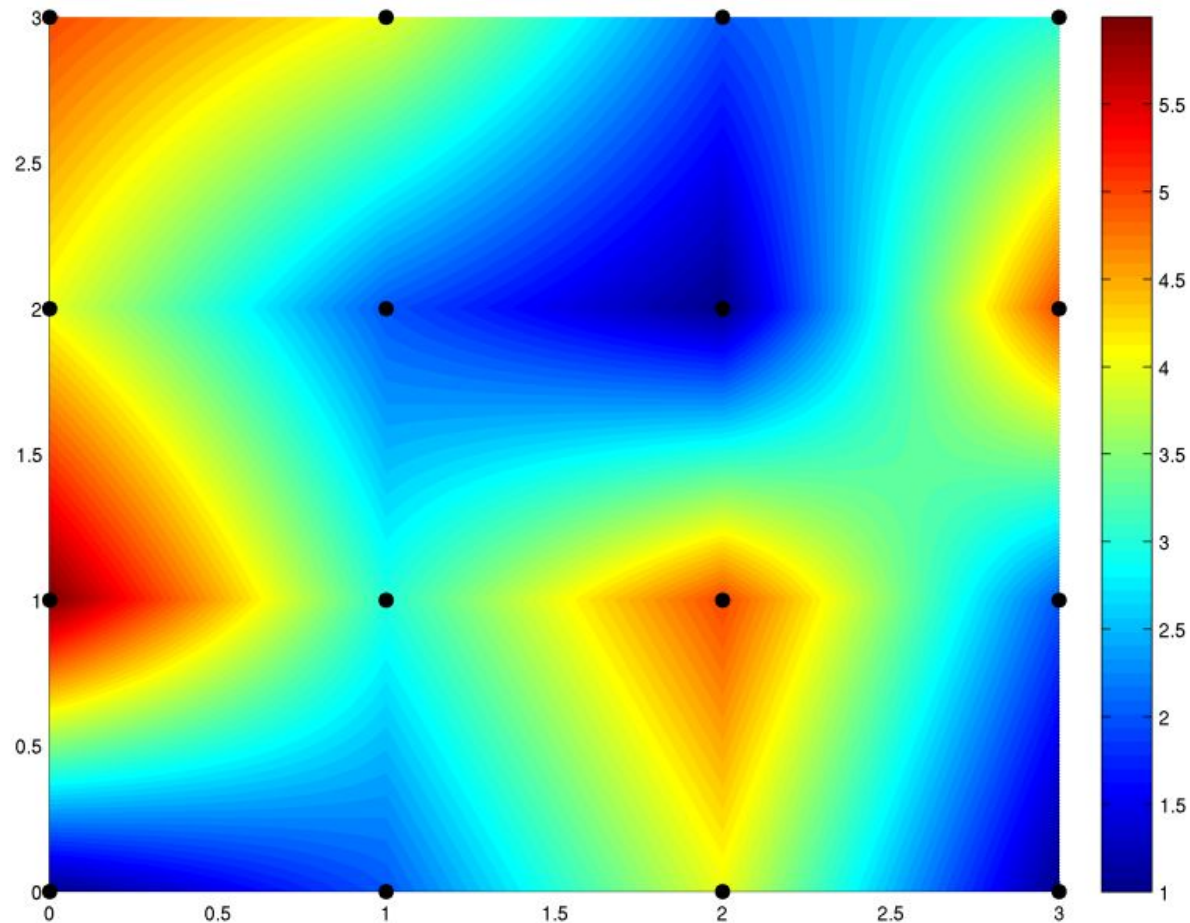
---





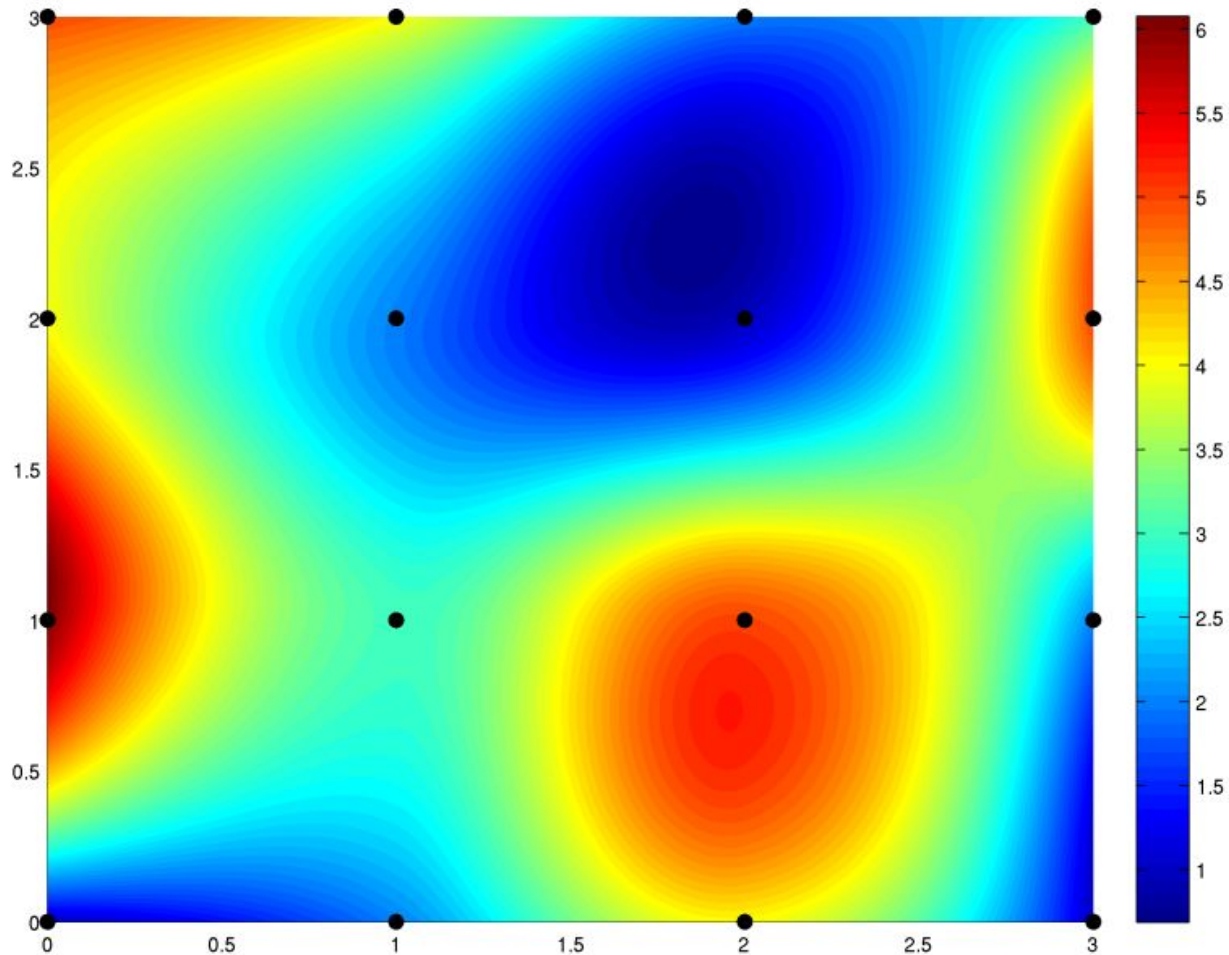
# Билинейная интерполяция

---



# Бикубическая сплайн-интерполяция

---



# Примеры

---

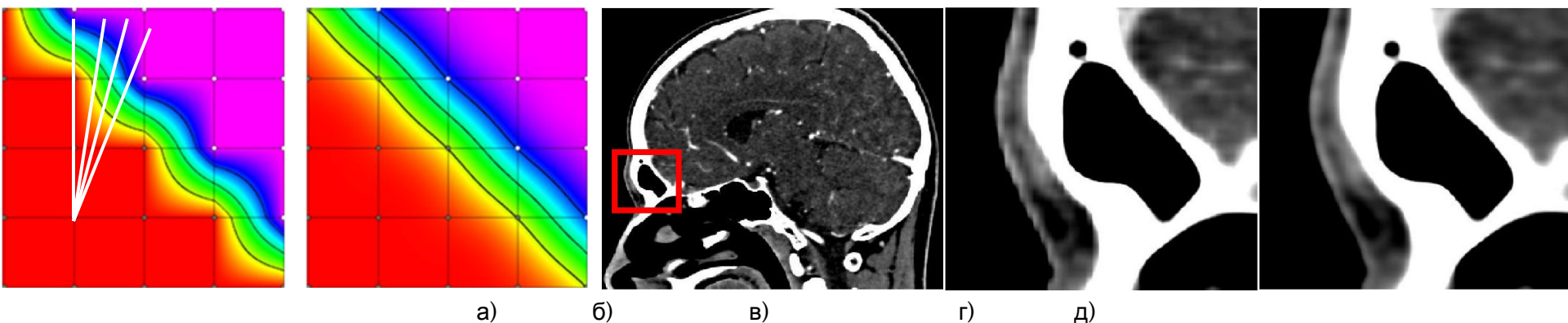


Рис.

Пример 2D искажений, вносимых фильтрацией в границу между зоной нулей (серые узлы) и единиц (белые узлы):

а) билинейной; б) бикубической; черным цветом показаны изолинии для интерполированных значений, белым – лучи, вдоль которых выполняется интегрирование.

Проявление тех же эффектов в плоском медицинском изображении на черно-серой и серо-белой границах во фрагментах г) и д).





LOBACHEVSKY STATE UNIVERSITY  
of NIZHNI NOVGOROD  
National Research University

---

# Нелинейная интерполяция



# Lanczos filters 4X4, 6X6 and 8X8

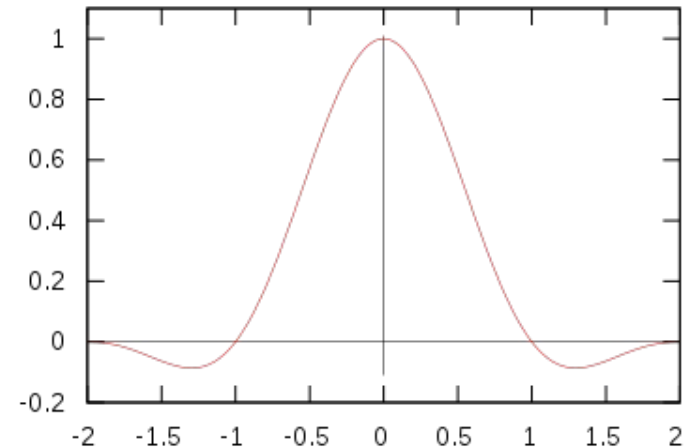
$$L(x) = \begin{cases} \text{sinc}(\pi x) \text{sinc}(\pi x/a) & \text{if } -a < x < a \\ 0 & \text{otherwise} \end{cases}$$

$$S(x, y) = \sum_{i=\lfloor x \rfloor - a + 1}^{\lfloor x \rfloor + a} \sum_{j=\lfloor y \rfloor - a + 1}^{\lfloor y \rfloor + a} s_{ij} L(x - i) L(y - j).$$

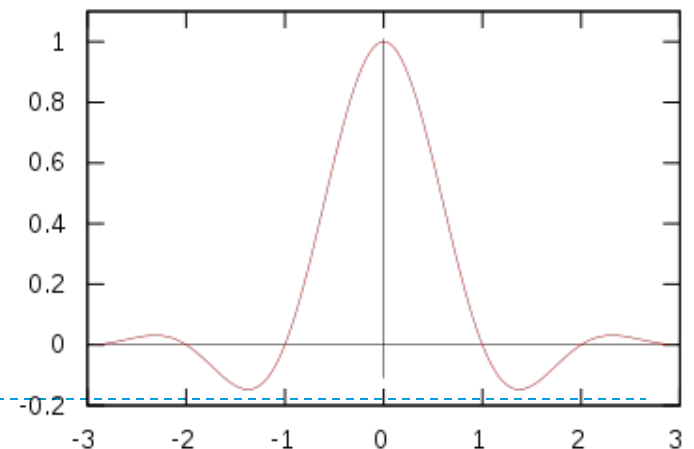
$$L(x, y) = L(x) \cdot L(y). \quad \text{Separable filter!!!}$$

$$S(x) = \sum_{i=\lfloor x \rfloor - a + 1}^{\lfloor x \rfloor + a} s_i L(x - i),$$

Lanczos kernel for a=2



Lanczos kernel for a=3



# Lanczos filters 4X4, 6X6 and 8X8

---





**УНИВЕРСИТЕТ  
ЛОБАЧЕВСКОГО**

Институт информационных технологий, математики и  
механики  
Кафедра математического обеспечения и  
суперкомпьютерных технологий

**Обработка изображений**

# Детекторы и дескрипторы

Светлана Носова  
svetlana.nosova@itmm.unn.ru

# Обзор

---

- Инварианты
- локализация особенных точек:
  - детектор Hessian
  - детектор Harris
- Инвариантность к масштабу
  - детектор (LOG)
  - детектор (DOG)
- Дескрипторы локальных особенностей
  - (SIFT)
  - (GLOH)
- Другое



# Введение

---

- Анализируем локальные особенности
- Применяется:
  - Обнаружение (детектирование) объекта
  - Нахождение соответствий на изображениях (поиск объектов)
  - «склейка» изображений (панорамы)

# Требования к дескрипторам

---

- Необходимо: обнаружить одинаковые точки (независимо друг от друга) на каждом изображении, участвующем в обработке.
- Инвариантность к переносу, повороту, масштабу.
- Инвариантность к аффинным преобразованиям.
- Инвариантность к наличию шума, размытости и т. Д.
- Положение: устойчивость к изменению формы и освещенности.
- Различимость: область должна содержать «интересную», «отличимую» информацию.
- Количество: детектор должен выдавать достаточно точек, чтобы описать/охарактеризовать изображение.
- Эффективность (по времени).

# Особые точки

---

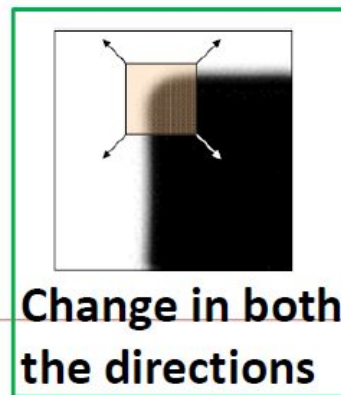
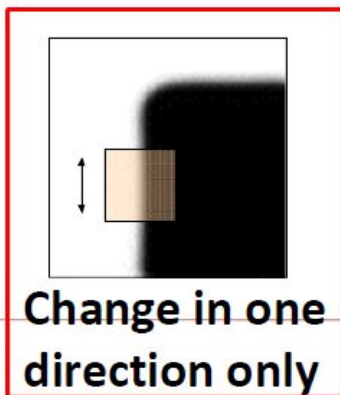
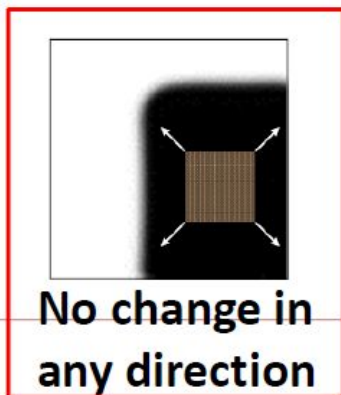
- В 1992 Haralick и Shapir [10] выделили следующие требования к особым точкам в виде следующих свойств:
  - Отличимость (distinctness)* – особая точка должна явно выделяться на фоне и быть отличимой (уникальной) в своей окрестности.
  - Инвариантность (invariance)* – определение особой точки должно быть независимо к аффинным преобразованиям.
  - Стабильность (stability)* – определение особой точки должно быть устойчиво к шумам и ошибкам.
  - Уникальность (uniqueness)* – кроме локальной отличимости, особая точка должна обладать глобальной уникальностью для улучшения различимости повторяющихся паттернов.
  - Интерпретируемость (interpretability)* – особые точки должны определяться так, чтобы их можно было использовать для анализа соответствий и выявления интерпретируемой информации из изображения.

# Особые точки

- ▣ Tuytelaars и Mikolajczyk (2006) выделили следующие свойства, которыми должны обладать особые точки:
  - Повторяемость (repeatability)* – особая точка находится в одном и том же месте сцены или объекта изображения, несмотря на изменения точки обзора и освещённости.
  - Отличительность / информативность (distinctiveness/informativeness)* – окрестности особых точек должны иметь большие отличия друг от друга, так, чтобы возможно было выделить и сопоставить особые точки.
  - Локальность (locality)* – особая точка должна занимать небольшую область изображения, чтобы быть уменьшит вероятность чувствительности к геометрическим и фотометрическим искажениям между двумя изображениями, снятых в различных точках обзора.
  - Количество (quantity)* – число обнаруженных особых точек должно быть достаточно большим, так чтобы их хватило для обнаружения даже небольших объектов. Однако оптимальное количество особых точек зависит от предметной области. В идеале количество обнаруженных особых точек должно адаптивно определяться с использованием простого и интуитивного порога. Плотность расположения особых точек должна отражать информационное содержимое изображения, чтобы обеспечить его компактное представление.
  - Точность (accuracy)* – обнаруженные особые точки должны точно локализовываться, как в исходном изображении, так и взятом в другом масштабе.
  - Эффективность (efficiency)* – время обнаружения особых точек на изображении должно быть допустимым в критичных по времени приложениях.

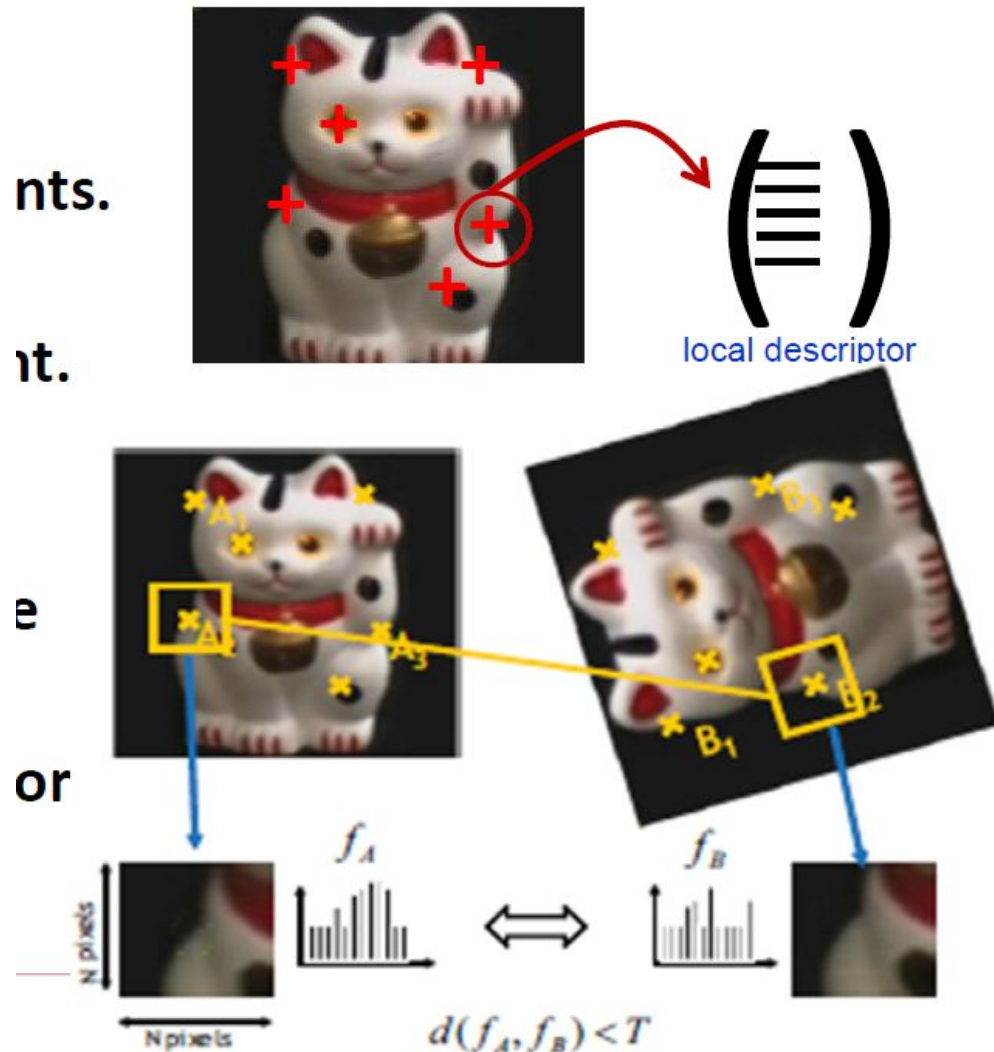
# Список детекторов

- ❑ **Смотри изменение функции градиента в двух направлениях!!!!**
  - ❑ Moravec detector
  - ❑ Hessian/ Harris corner detection
  - ❑ Laplacian of Gaussian (LOG) detector
  - ❑ Difference of Gaussian (DOG) detector
  - ❑ Hessian/ Harris Laplacian detector
  - ❑ Hessian/ Harris Affine detector
  - ❑ Maximally Stable Extremal Regions (MSER)



# Последовательность действий

1. Найти особенные точки на изображении.
2. Рассмотреть область вокруг каждой особенной точки
3. Вычислить дескриптор особенной точки, нормализовать значения.
4. Сопоставить дескрипторы двух изображений.



# Moravec detector

1. Для каждого пикселя  $(x,y)$  в изображении вычислить изменение интенсивности

$$V_{u,v}(x,y) = \sum_{\forall a,b \in w} (I(x+u+a,y+v+b) - I(x+a,y+b))^2,$$
$$(u,v) \in \{(1,0),(1,1),(0,1),(-1,1),(-1,0),(-1,-1),(0,-1),(1,-1)\}$$

2. Построить карту вероятности нахождения углов в каждом пикселе  $(x,y)$  изображения посредством вычисления оценочной функции. То есть определяется направление, которому соответствует наименьшее изменение интенсивности, т.к. угол должен иметь смежные ребра.
3. Отсечь пиксели, в которых значения  $C(x,y)$  ниже порогового значения  $T$ .
4. Подавление не максимумов. Все полученные ненулевые элементы карты соответствуют углам на изображении.

# Hessian corner detector

---

- Поиск мест на изображении, которые имеют сильное изменение вдоль двух ортогональных направлений.

$$H(x, \sigma) = \begin{bmatrix} I_{xx}(x, \sigma) & I_{xy}(x, \sigma) \\ I_{xy}(x, \sigma) & I_{yy}(x, \sigma) \end{bmatrix}$$

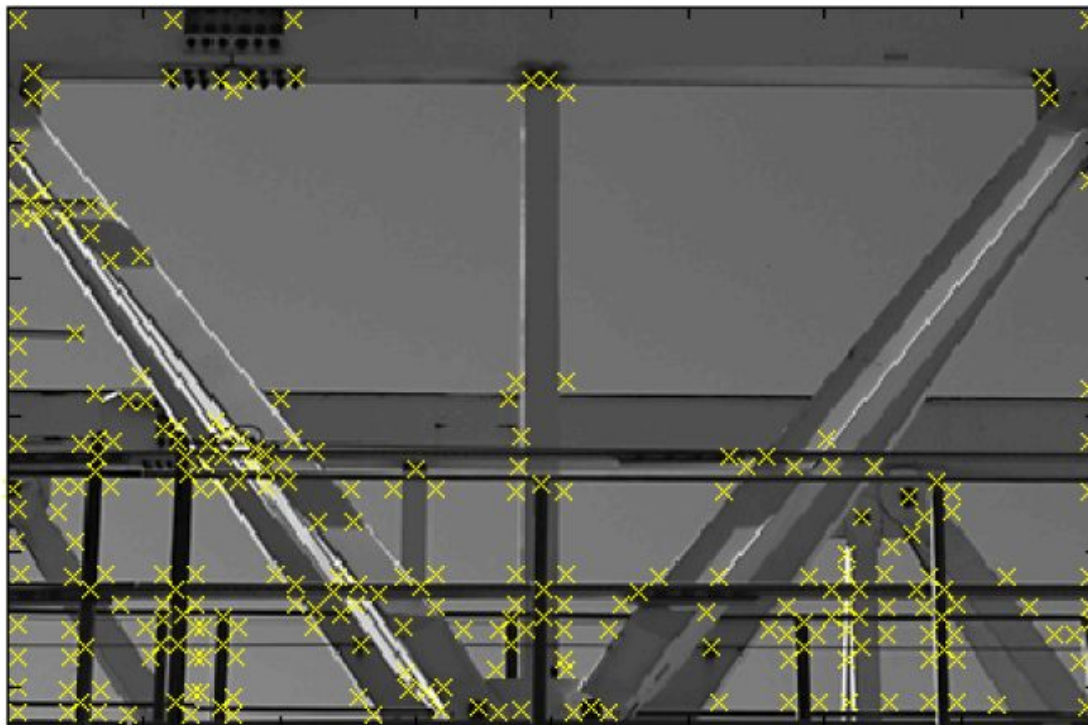
$$\det(H) = I_{xx}I_{yy} - I_{xy}^2$$

1. Подсчитать градиент.
2. Выполнить подавление немаксимумов в окне 3 \* 3.
3. Оставить точки, в которых  $\det(H) > \theta$  ( $\theta$  – порог)



# Harris detector

---



- Выделение «истинных» углов и сильно-текстурированных областей.

# Harris Corner

- Локализация мест, имеющих изменение информации в двух основных направлениях.
- Рассмотрим матрицу автокорреляции:

$$C(\mathbf{x}, \sigma, \tilde{\sigma}) = \underbrace{G(\mathbf{x}, \tilde{\sigma})}_{\text{triangle}} * \begin{bmatrix} I_x^2(\mathbf{x}, \sigma) & I_x I_y(\mathbf{x}, \sigma) \\ I_x I_y(\mathbf{x}, \sigma) & I_y^2(\mathbf{x}, \sigma) \end{bmatrix} \quad \tilde{\sigma} \approx 2\sigma$$

- C – гауссова сумма

$$C = \begin{bmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{bmatrix} = \mathbf{R}^{-1} \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \mathbf{R}$$

$$E(u,v) = \sum_{(x,y) \in W} w(x,y)(I(x+u,y+v) - I(x,y))^2 \approx \sum_{(x,y) \in W} w(x,y)(I_x(x,y)u + I_y(x,y)v)^2 \approx (x \ y)M \begin{pmatrix} x \\ y \end{pmatrix},$$

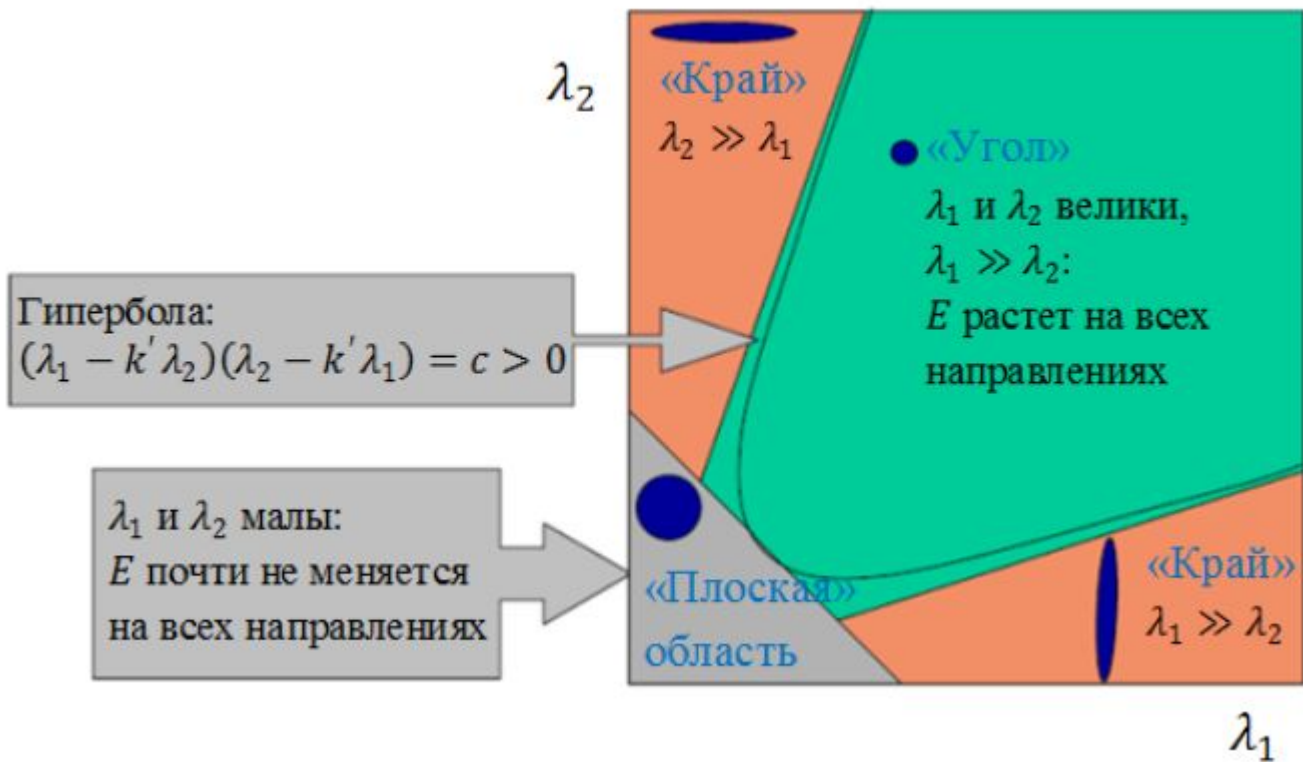
где  $w(x,y)$  – весовая функция (обычно используется функция Гаусса или бинарное окно).



$M$  – автокорреляционная матрица:

$$M = \sum_{(u,v) \in W} w(u,v) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}$$

Угол характеризуется большими изменениями функции  $E(x,y)$  по всем возможным направлениям  $(x,y)$ , что эквивалентно большим по модулю собственным значениям матрицы  $M$ . Расположение собственных значений приведено на следующем рисунке.



---

Поскольку напрямую считать собственные значения является трудоёмкой задачей, Харрисом и Стефеном была предложена мера отклика [7]:

$$\det(C) = \lambda_1 \lambda_2$$

$$R = \det M - k(\text{tr}M)^2 > k,$$

$$\text{trace}(C) = \lambda_1 + \lambda_2$$

где  $k$  – эмпирическая константа,  $k \in [0,04;0,06]$ .

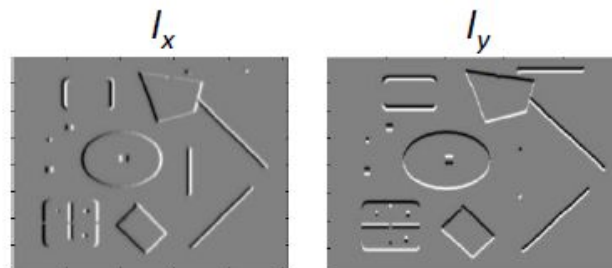
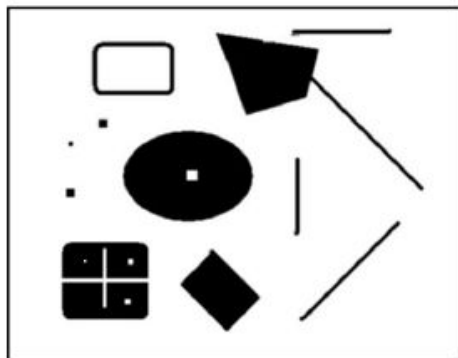
- значение  $R$  положительно для угловых особых точек.
- отсечение точек по найденному порогу
- Подавление не-максимумов по окрестности

+ инвариантен к поворотам,

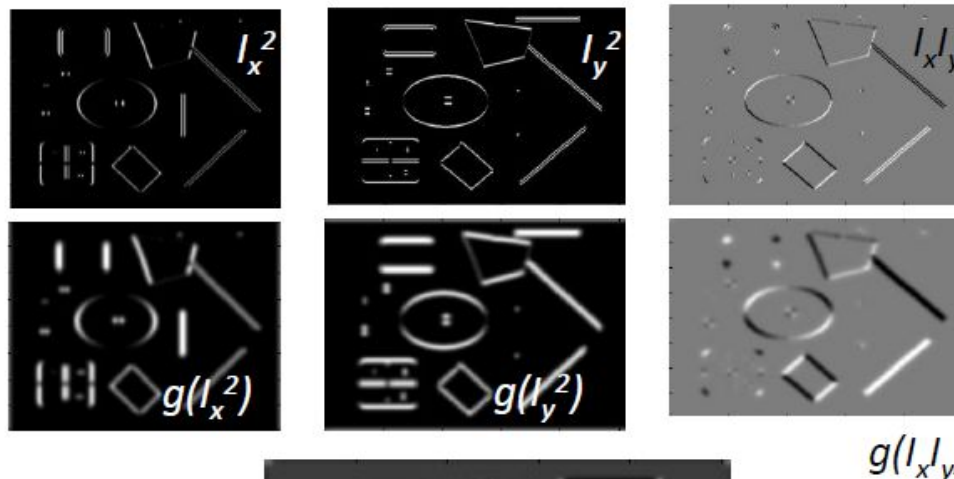
+частично инвариантен к аффинным изменениям интенсивности.

-чувствительность к шуму и

-зависимость детектора от масштаба изображения (для устранения этого недостатка используют многомасштабный детектор Харриса (multi-scale Harris detector)).



**1. Image derivatives**

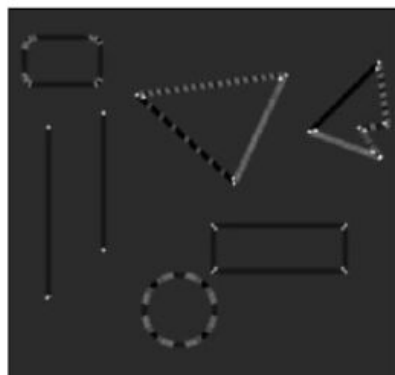
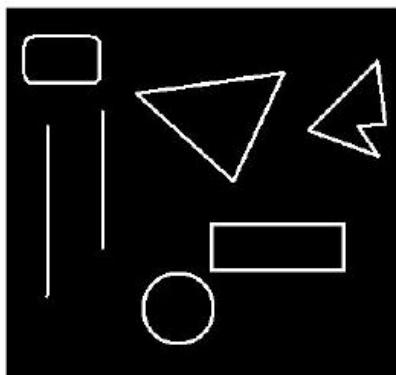


**2. Square of derivatives**

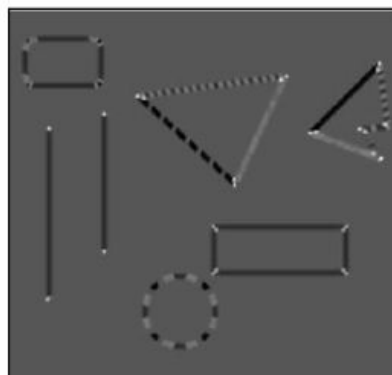
**3. Gaussian filter  $G(\sigma_I)$**



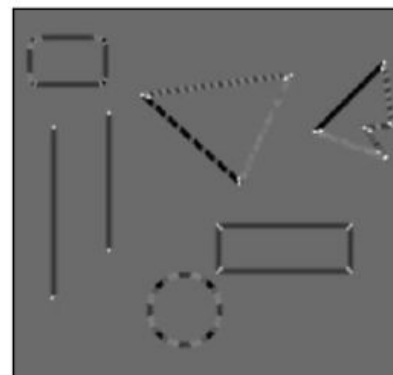
**4. Cornerness function – both eigenvalues are strong**



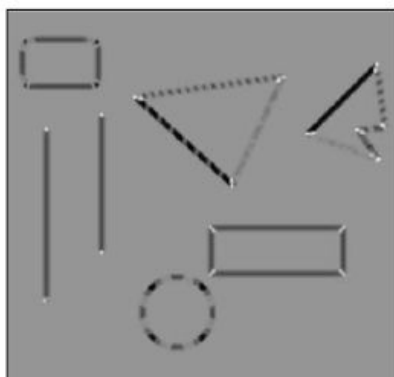
$\alpha = .04$



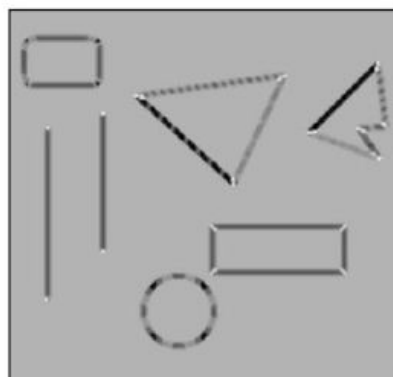
$\alpha = .08$



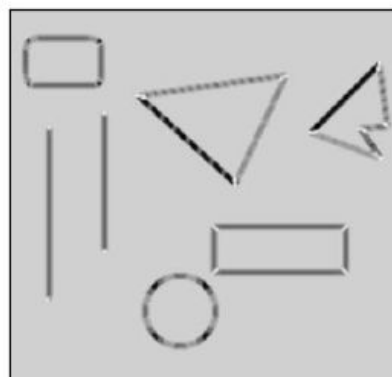
$\alpha = .1$



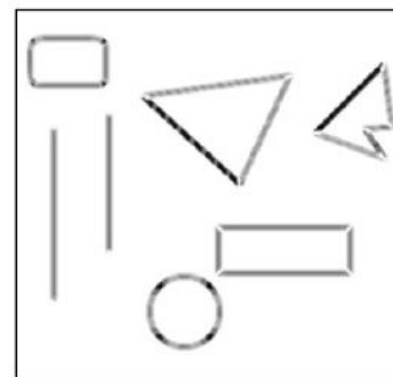
$\alpha = .14$



$\alpha = .17$



$\alpha = .2$



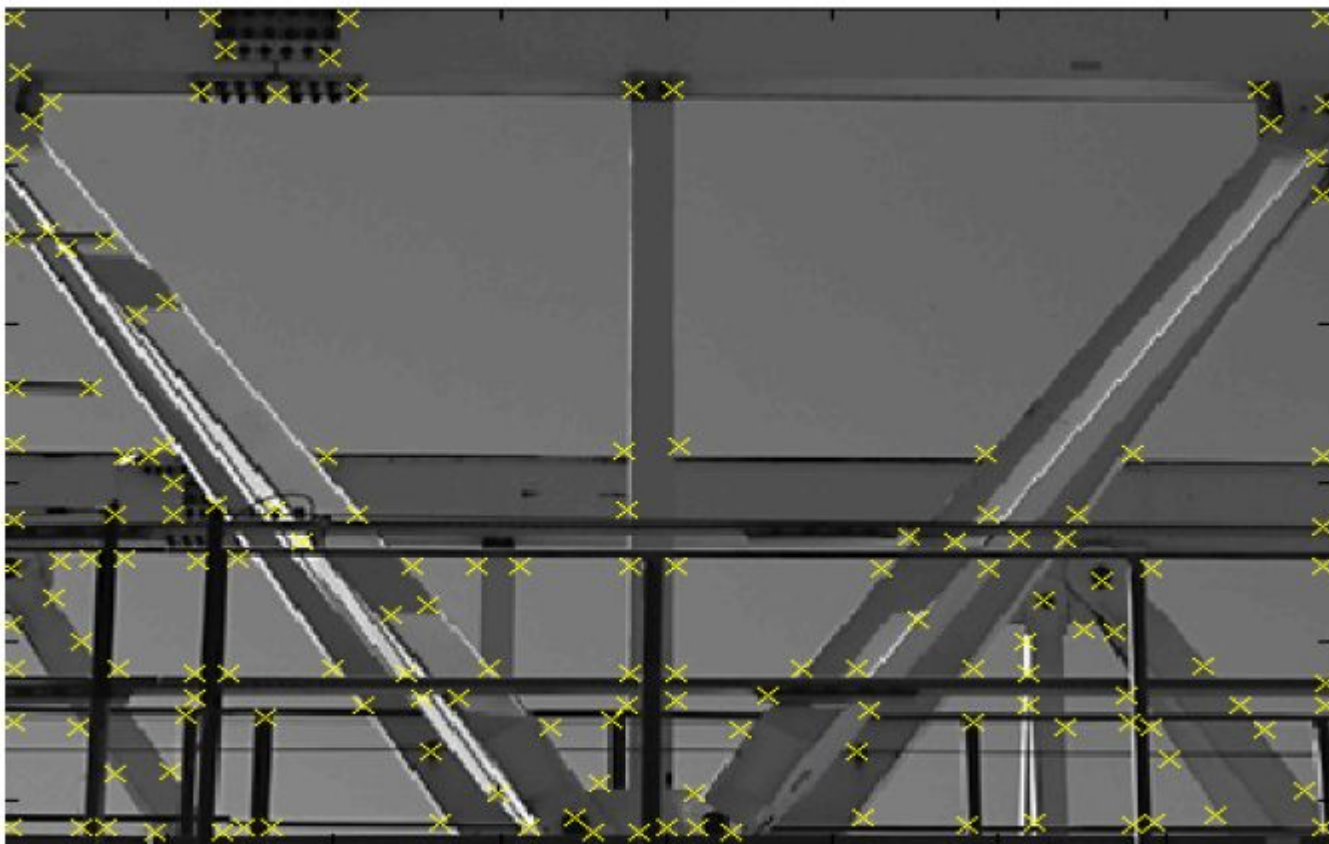
$\alpha = .25$



# Harris detector

---

- Точные углы

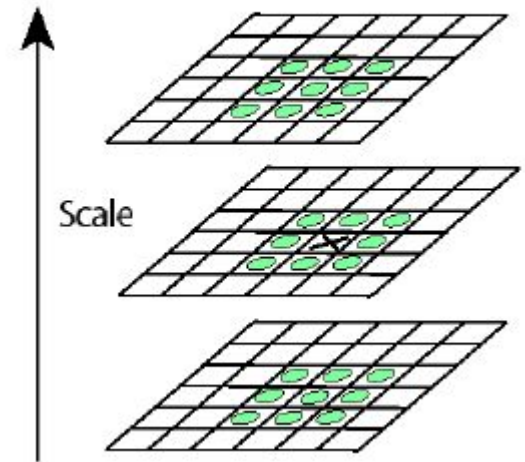


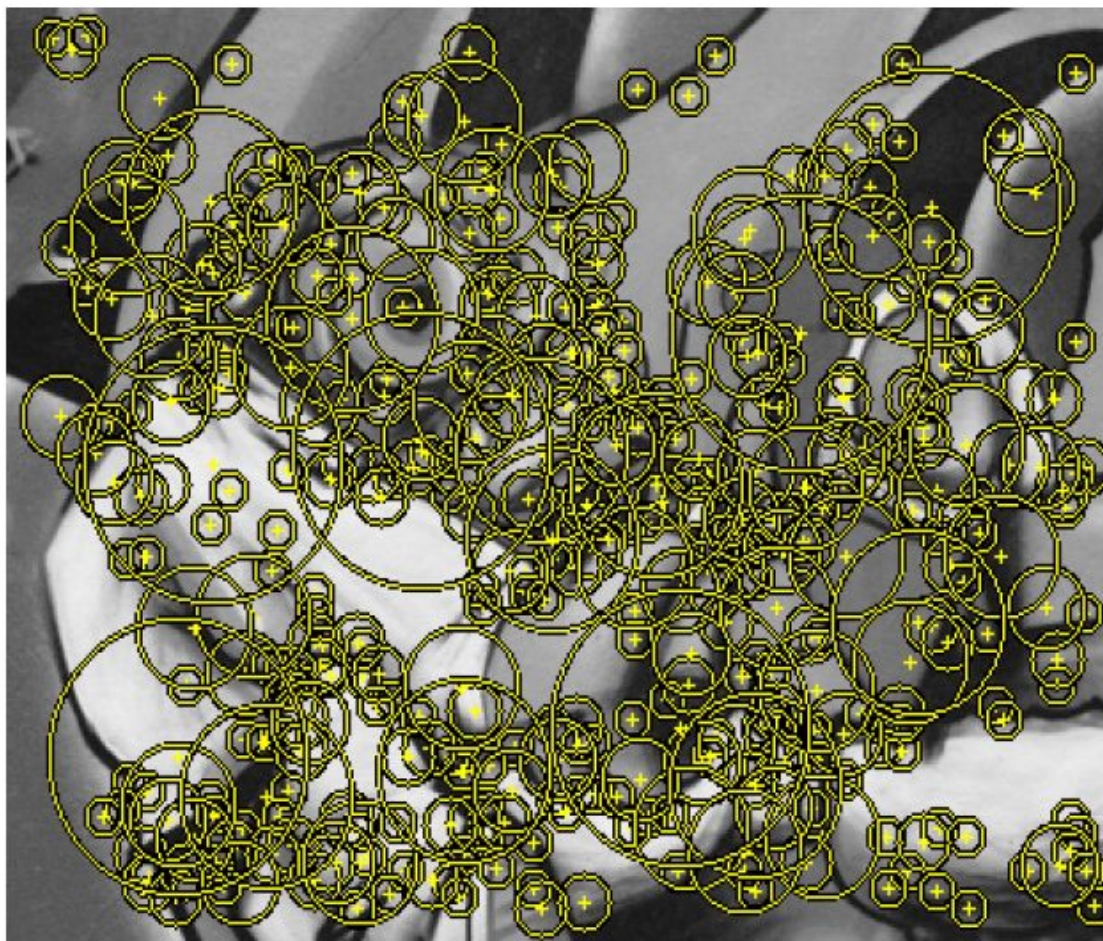


# Laplacian of Gaussian (LOG)

- Многомасштабность.
- Вместо пересечения с 0 ( для детектирования границ), рассматривать точку, которая максимум по 26 соседям (9+8+9).
- LOG можно использовать для поиска характеристики масштаба объекта в данной точке.
- LOG может использоваться для нахождения масштабного инварианта регионы путем поиска экстремумов 3D (местоположение + масштаб)
- LOG также используется для обнаружения краев

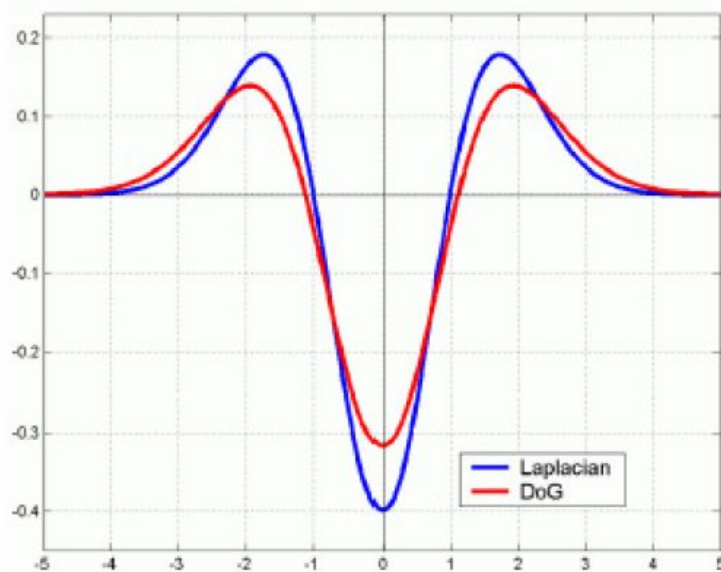
$$L(x, \sigma) = \sigma^2 (I_{xx}(x, \sigma) + I_{yy}(x, \sigma))$$





# Difference of Gaussian (DOG) Detector

- Приближение LoG - DoG (ускорение вычислений)
- Рассмотрим регион, где DOG больше, чем заданный порог и масштаб относится к определенному диапазону



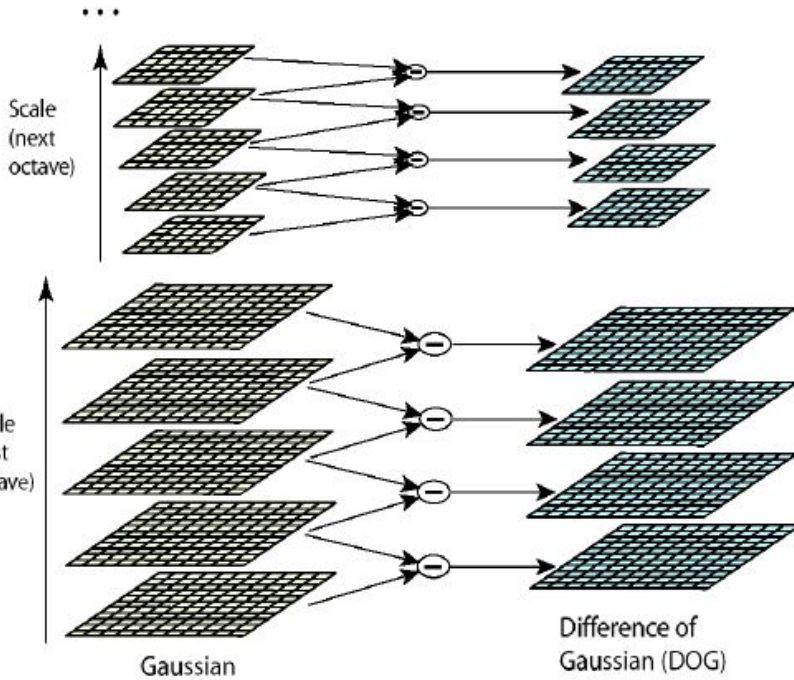
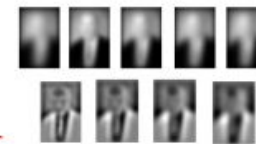
$$D(x, \sigma) =$$

$$(G(x, k\sigma) - G(x, \sigma)) * I(x)$$

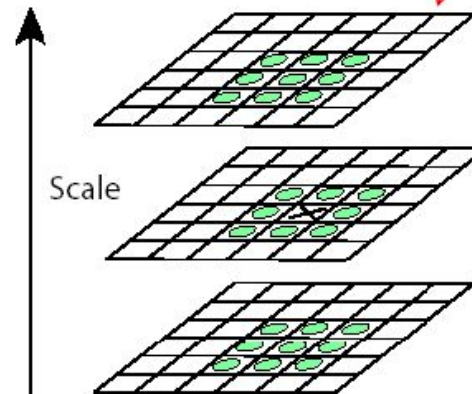
$$k = 2^{1/K}$$

$$K = 0, 1, 2, \dots, \text{constant}$$





**X is selected if it is larger or smaller than all neighbors.**

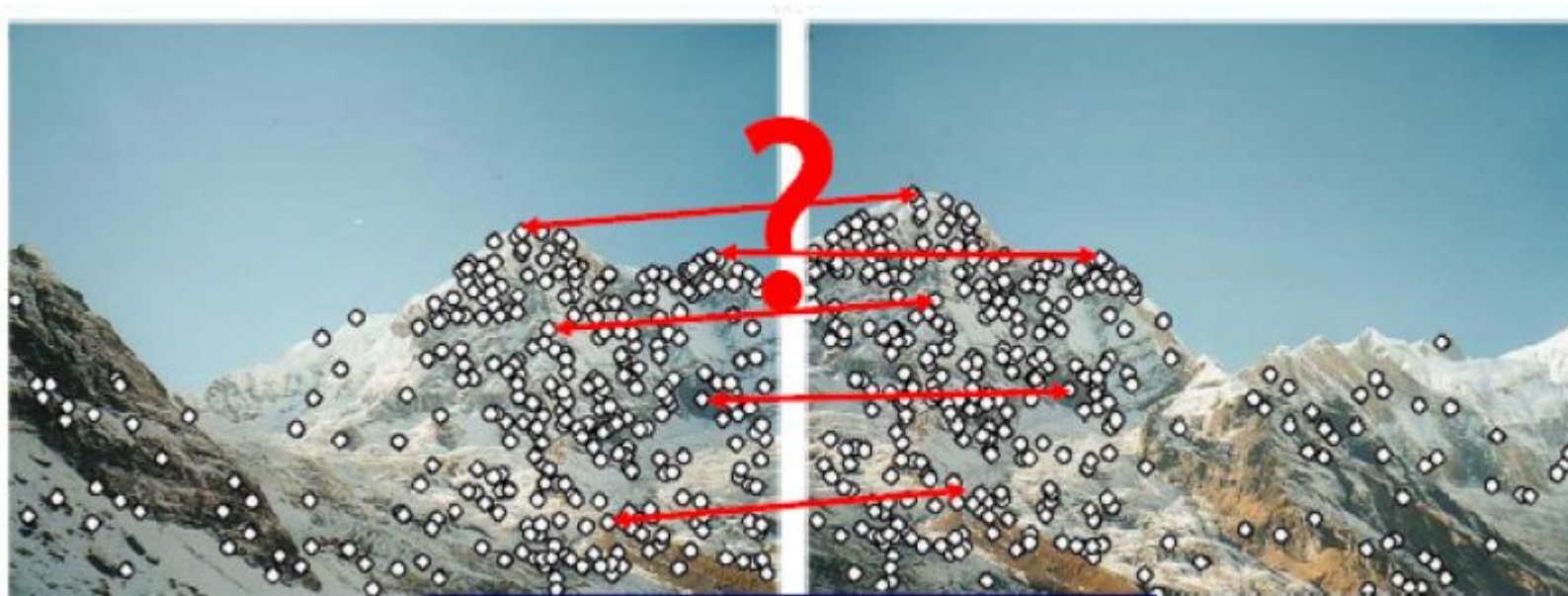




Feature detector	<u>Edge</u>	<u>Corner</u>	<u>Blob</u>
<u>Canny</u>	X		
<u>Sobel</u>	X		
<u>Harris &amp; Stephens / Plessey</u>	X	X	
<u>SUSAN</u>	X	X	
<u>Shi &amp; Tomasi</u>		X	
<u>Level curve curvature</u>		X	
<u>FAST</u>		X	X
<u>Laplacian of Gaussian</u>		X	X
<u>Difference of Gaussians</u>		X	X
<u>Determinant of Hessian</u>		X	X
<u>MSER</u>			X
<u>PCBR</u>			X
<u>Grey-level blobs</u>			X



- 
- Как найти соответствие между особенными точками?



# List of local feature descriptors

---

- **Scale Invariant Feature Transform (SIFT)**
- Speed-Up Robust Feature (SURF)
- Histogram of Oriented Gradient (HOG)
- **Gradient Location Orientation Histogram (GLOH)**
- PCA-SIFT
- Pyramidal HOG (PHOG)
- Pyramidal Histogram Of visual Words (PHOW)
- Others...(shape Context, Steerable filters, Spin images)



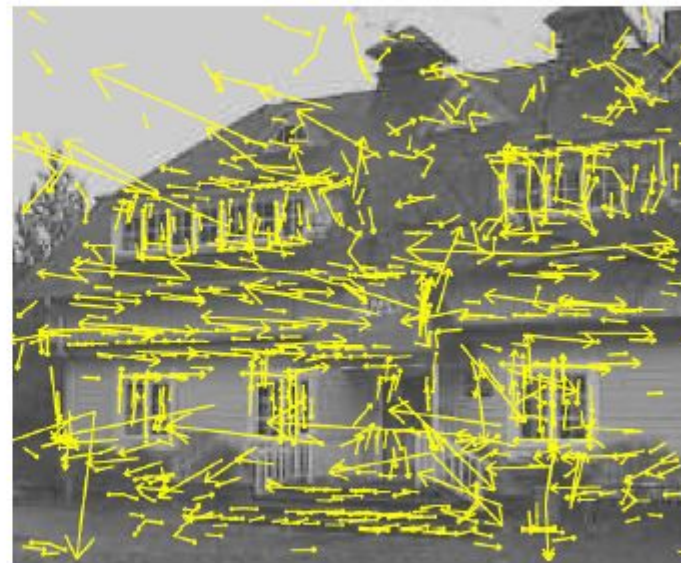
# Scale Invariant Feature Transform (SIFT)

---

- Step 1: Scale-space extrema Detection – детектировать особенные точки с использованием DoG.
- Step 2: Keypoint Localization – Определите местоположение и масштаб в каждой особенной точке.
- Step 3: Orientation Estimation – Рассчитать локальные градиенты изображения для присвоения ориентации каждой локализованной ключевой точке.
- Сохранить направление, масштаб и положение.
- Step 4: Keypoint Descriptor – Извлечь локальные градиенты изображения в выбранном масштабе вокруг ключевой точки и сформировать представление, инвариантное к локальному искажению формы и освещению.

# детектировать особенные точки с использованием DoG.

---



832 DOG extrema

Step 2: Keypoint Localization – Определите местоположение и масштаб в каждой особенной точке.

---

- Цель: отбросить точки низкой контрастности и точки, лежащие на границе (не угол).
- О контрастности

$$D(\underline{x}) = D + \frac{\partial D^T}{\partial \underline{x}} \underline{x} + \frac{1}{2} \underline{x}^T \frac{\partial^2 D^T}{\partial \underline{x}^2} \underline{x}$$
$$\mathbf{D}(\mathbf{x}, \boldsymbol{\sigma}) =$$
$$(\mathbf{G}(\mathbf{x}, k\boldsymbol{\sigma}) - \mathbf{G}(\mathbf{x}, \boldsymbol{\sigma})) * \mathbf{I}(\mathbf{x})$$

- Вычислить локальные максимумы. Удалить минимумы.

$$\text{local minima (for contrast)} \quad D(\underline{\hat{x}}) < 0.03$$

- DoS дает слишком много граничных точек
- Решение:
- -для границ одно собственное значение много больше другого.
- Для углов данного эффекта не наблюдается.

Hessian  
Matrix

$$H = \begin{bmatrix} I_{xx} & I_{xy} \\ I_{xy} & I_{yy} \end{bmatrix}$$

Harris  
corner  
criterion

$$\frac{\text{Tr}(H)^2}{\text{Det}(H)} < \frac{(r+1)^2}{r}$$

Discard points with  
response below threshold;  
Value of  $r = 10$ , is used;

Step 3: Orientation Estimation – Рассчитать локальные градиенты изображения для присвоения ориентации каждой локализованной ключевой точке

---

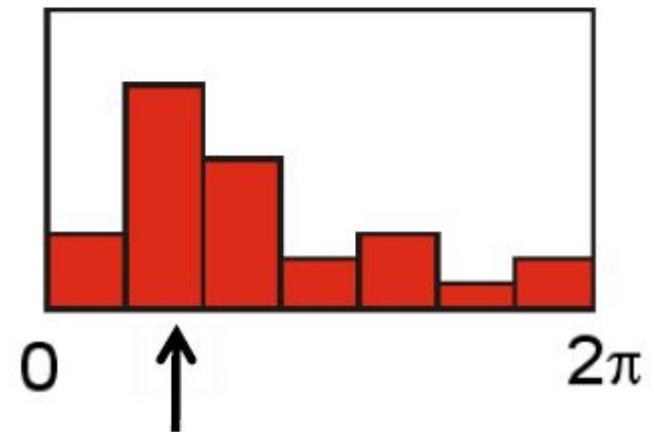
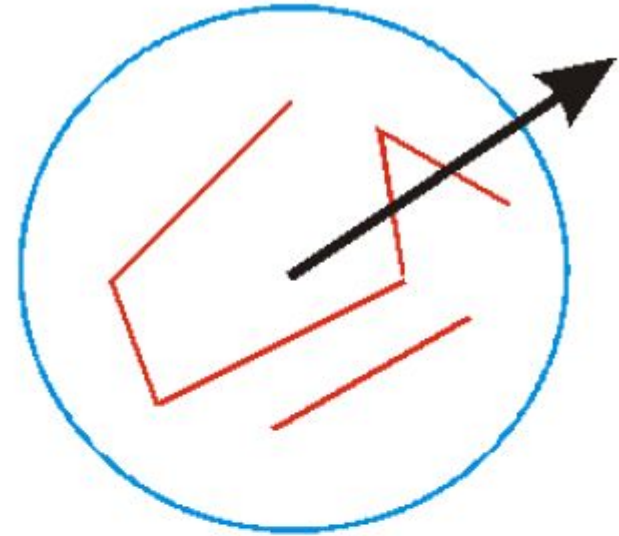
□ Цель: получить инвариантность к повороту.

$$m(x, y) = \sqrt{(I(x+1, y) - I(x-1, y))^2 + (I(x, y+1) - I(x, y-1))^2}$$

$$\theta(x, y) = \tan^{-1} \frac{I(x, y+1) - I(x, y-1)}{I(x+1, y) - I(x-1, y)}$$



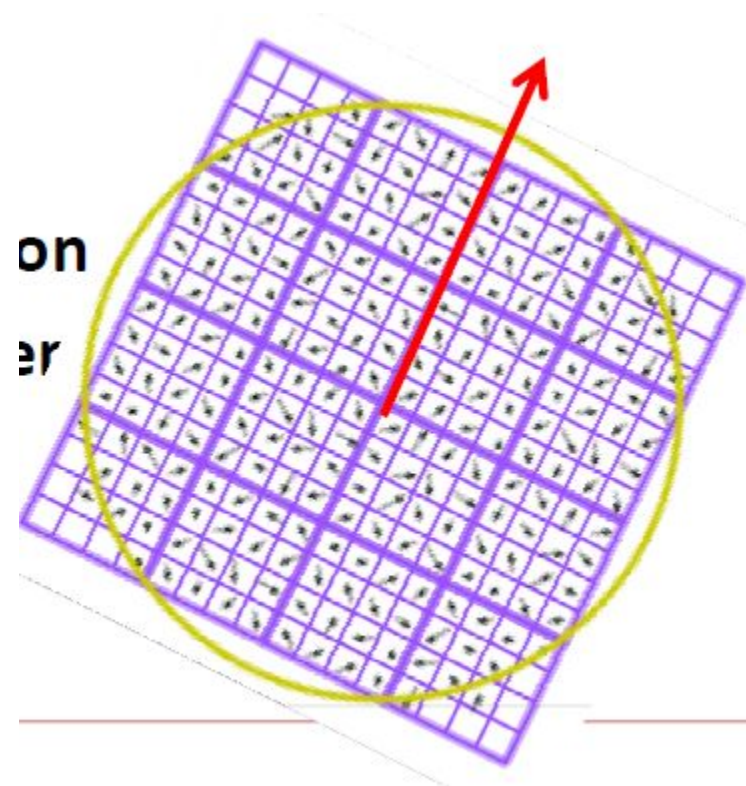
- Вычислить взвешенную (магнитуда, коэффициенты ядра гаусса) гистограмму направлений локального градиента, рассчитанных в выбранном масштабе
- Назначить ориентацию региона – пик сглаженной гистограммы
- Несколько пиков – несколько ключевых точек.



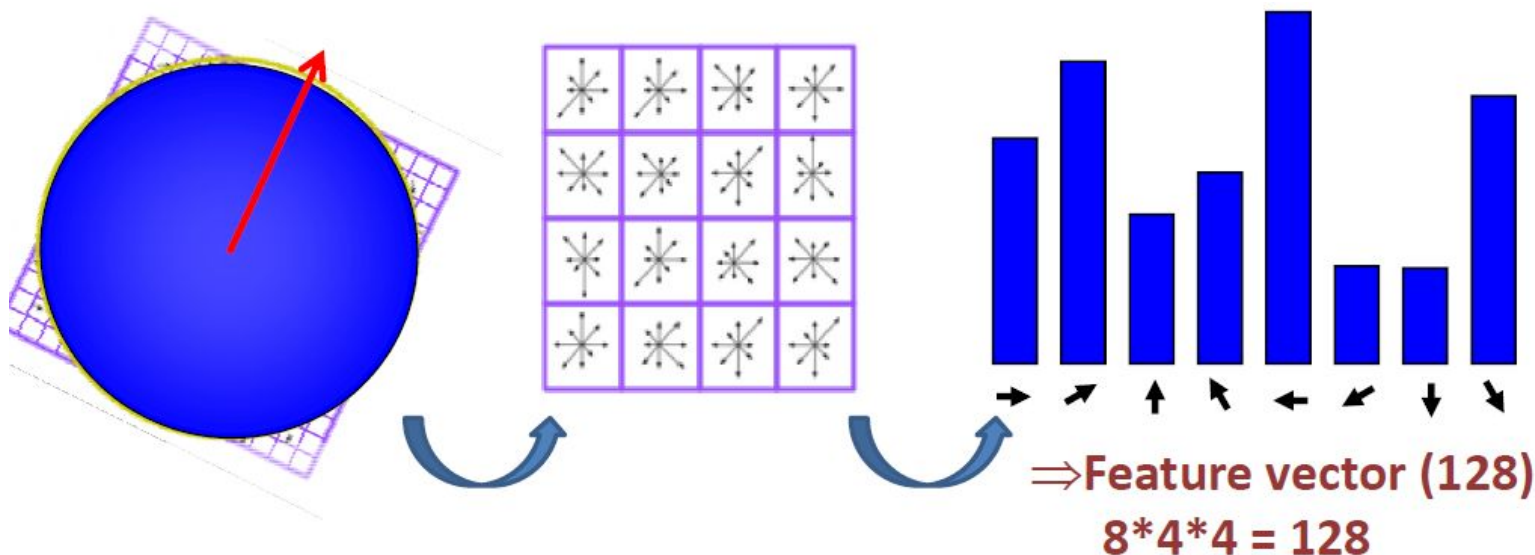


Step 4: Keypoint Descriptor – Извлечь локальные градиенты изображения в выбранном масштабе вокруг ключевой точки и сформировать представление, инвариантное к локальному искажению формы и освещению.

- Получаем локальный дескриптор (отличающийся), и инвариантный к изменению типа освещения и аффинного изменения.
- Рассматривается прямоугольная сетка  $16 * 16$  в направлении доминирующей ориентации региона.
- Разделить регион на  $4 * 4$  субрегиона.
- Гауссова фильтрация области, что дает больший вес центральным пикселям дескриптора.



- Вычислить гистограмму ( 8 бинов) для каждого субрегиона, взвешенного по величине и гауссову окну ( $\sigma$  – половина размера окна).



- Нормализовать 128-компонентный вектор (инвариантность к освещению)







---

Log-polar coordinates in the plane consist of a pair of real numbers  $(\rho, \theta)$ , where  $\rho$  is the logarithm of the distance between a given point and the origin and  $\theta$  is the angle between a line of reference (the  $x$ -axis) and the line through the origin and the point. The angular coordinate is the same as for polar coordinates, while the radial coordinate is transformed according to the rule

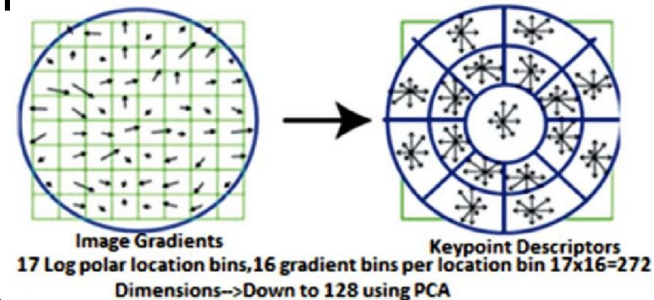
$$r = e^\rho,$$

where  $r$  is the distance to the origin. The formulas for transformation from Cartesian coordinates to log-polar coordinates are given by

$$\begin{cases} \rho = \log \sqrt{x^2 + y^2}, & \text{[dubious - discuss]} \\ \theta = \arctan y/x \text{ if } x > 0. \end{cases}$$

# Gradient Location Orientation Histogram (GLOH)

- Первые 3 шага – аналогичны SIFT
- Создание дескриптора
  - Рассматриваются лог-полярные координаты
    - 3 различных радиуса
    - 8 направлений углов
    - Итого : 17 значений
  - Для каждого региона: сформировать гистограмму (16 бин)
    - Итого  $16 * 17 = 272$  значения.
  - Квантовать вектор значений до 128 бинов.



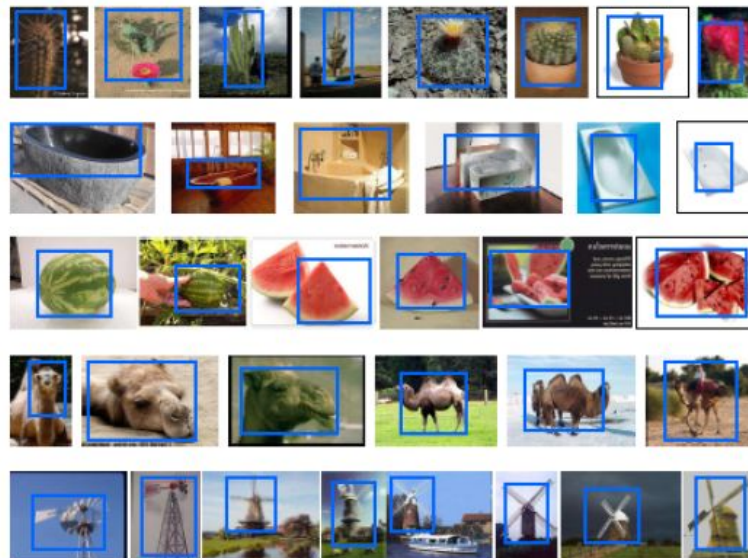




**192 correct  
matches  
(yellow)  
208 false  
matches  
(blue).**



SURF



PHOW



HOG

- 
- Other Feature descriptors :
  - - LBP, LTP and variants, HAAR;
  - - PCA-SIFT, VLAD, MOSIFT,
  - - deep features, CNN, Fisher vector,
  - - SV-DSIFT, BF-DSIFT, LL-MO1SIFT, 1SIFT, VM1SIFT, VLADSIFT,
  - - DECAF, Fisher vector pyramid, IFV
  - - Dirichlet Histogram
  - - Simplex based STV (3-D), MSDR;
  - BOV-W, Steak flow, tracklets, spatio-temporal gradients,
  - LCS, LTDS, MRF, LDA, RFT, LCSS, MDA, DFM, Dynamic textures,
  - BOAW, HFST, SRC based MHOF, LBPTOPS, HOP



**УНИВЕРСИТЕТ  
ЛОБАЧЕВСКОГО**

Институт информационных технологий, математики и  
механики  
Кафедра математического обеспечения и  
суперкомпьютерных технологий

**Обработка изображений**

# Сегментация изображений

По материалам презентации коллектива авторов:

**Владимир Вежневек, Антон Конушин**

**Александр Вежневек**

**Курс – «Введение в компьютерное зрение»**

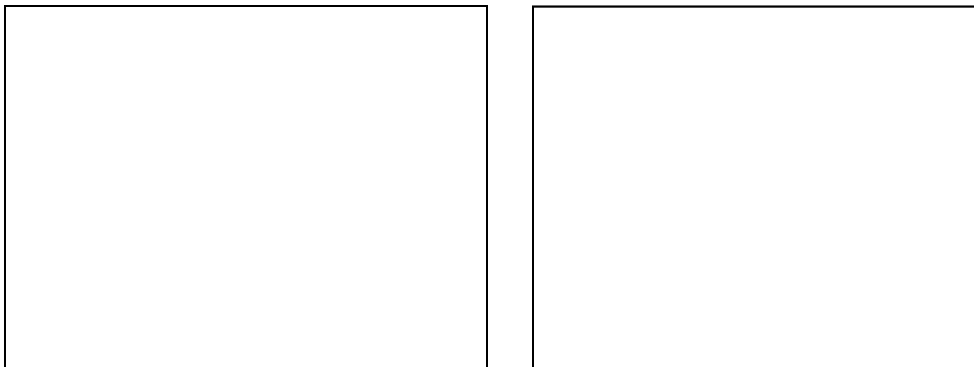
МГУ ВМК, Graphics & Media Lab,  
осень 2006



# Что такое сегментация?

---

- Анализ низкого уровня:
  - разбиение на области «похожих» между собой пикселей
- Анализ высокого уровня:
  - отделение находящихся на изображении объектов от фона и друг от друга



# Автоматика и интерактивность

---

## □ Подразделяем

### □ Автоматическая сегментация

□ Сегментация производимая без взаимодействия с пользователем

- Картинка на входе, регионы на выходе

### □ Интерактивная сегментация

□ Сегментация, управляемая пользователем, допускающая и/или требующая ввода дополнительной информации

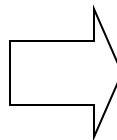
- Пример – «волшебная палочка» в Photoshop



# Применение сегментации

---

- Измерение параметров объектов



# Применение сегментации

---

- Предобработка перед высокоуровневым анализом



# Определение сегментации 1

---

- «Жесткая» сегментация
  - Разбиение изображения на неперекрывающиеся области, покрывающие все изображение и однородные по некоторому признаку
- Формально:
  - Разбиение изображения на набор областей

□

□

□

□



# Рассмотрим семейства методов:

---

- Основанные на поиске краев
- Основанные на формировании однородных областей
- Метод водораздела / tobogganing
- Методы из теории графов



# Автоматическая сегментация

---

- Как можно сформировать однородные области?
  - Отталкиваясь от неоднородности на границах
    - Пример – ищем резкие переходы яркости, берем их как границы областей
  - Отталкиваясь от однородности внутри областей
    - Пример – объединяем в одну область пиксели, близкие по яркости



# Однородность

---

- Варианты однородности:

- По яркости

- По цвету

- По близости на изображении

- По текстуре

- По глубине

- (Если есть 3D информация)

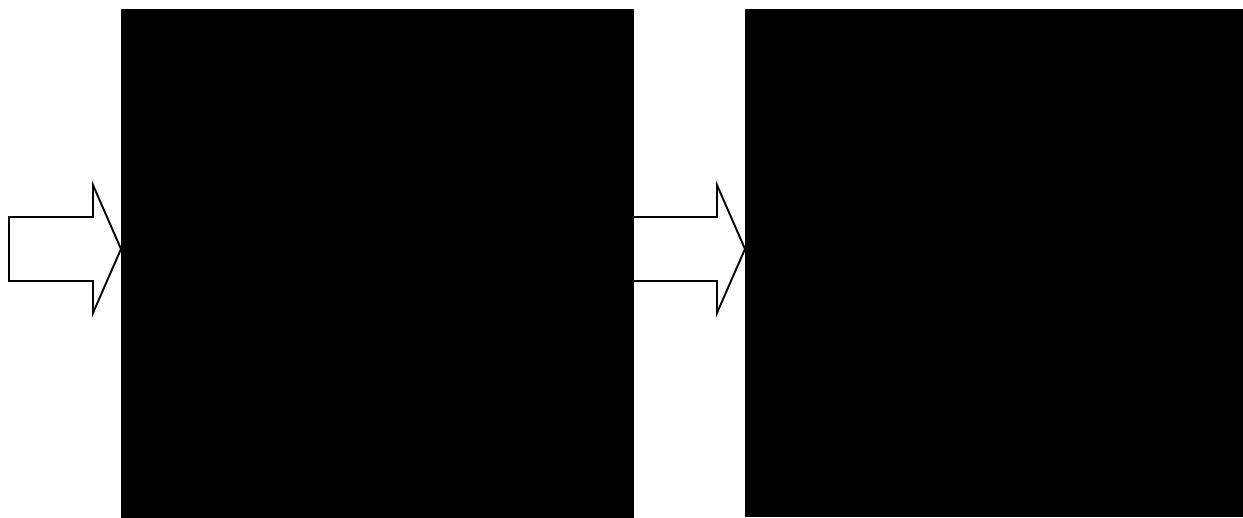




# Сегментация через поиск неоднородностей

---

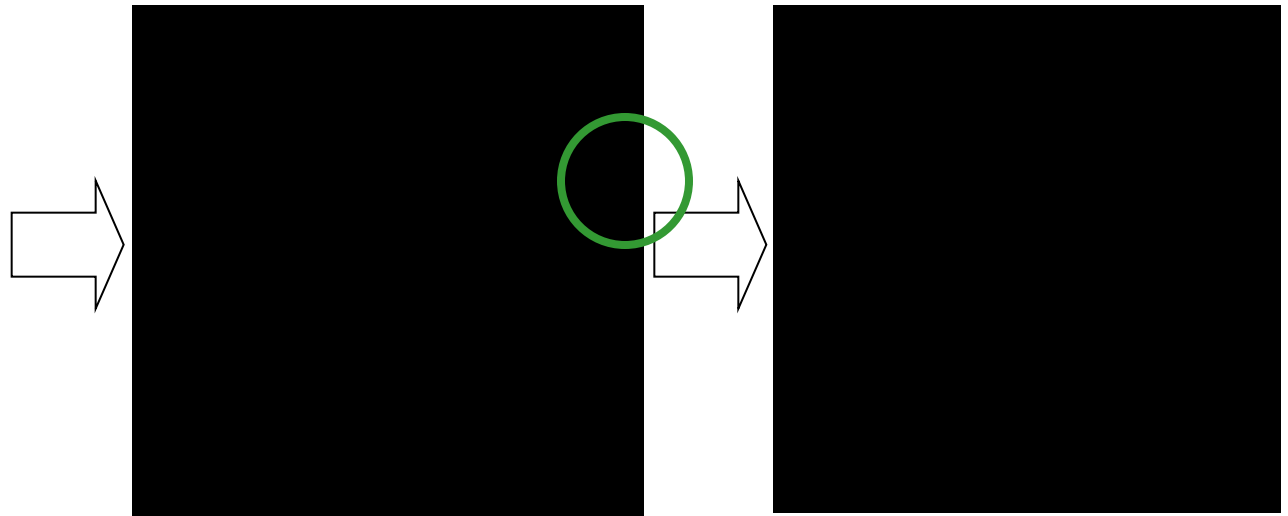
- Наиболее простой и чаще всего используемый вариант:
  - Поиск неоднородностей яркости через выделение краев



# Алгоритм

---

1. Найдём все контура на изображении алгоритмом Canny;
2. Найдём все замкнутые контура;
3. «Внутренности» замкнутых контуров являются искомыми однородными областями;



# Сегментация через поиск однородных областей

---

## □ План

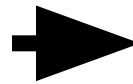
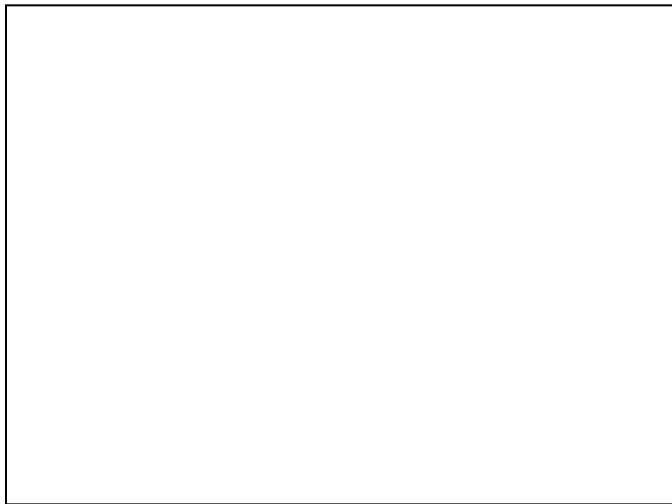
- Сегментация без учета пространственных связей
  - Пороговая фильтрация
  - Кластеризация по цвету
- Сегментация с учетом пространственных связей
  - Разрастание областей (region growing)
  - Слияние/разделение областей (region merging/splitting)



# Пороговая фильтрация

---

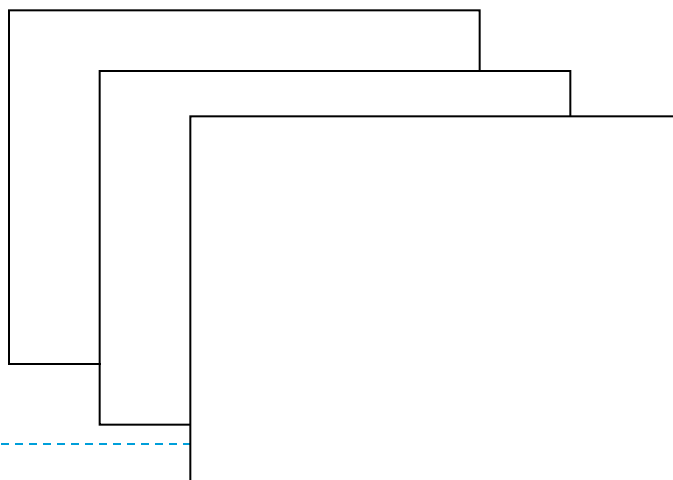
- Разделение пикселей на  $n$  классов по их яркости
  - Чаще всего используется 2 класса (бинаризация)



# Гистограммы

---

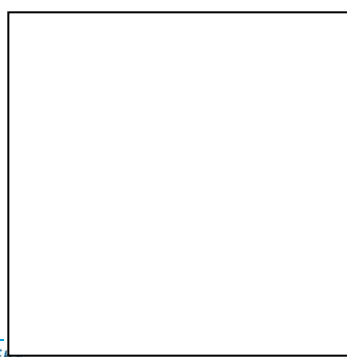
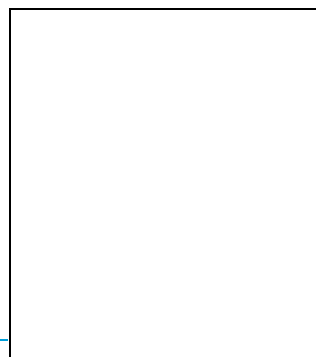
- Свойства:
  - Рассчитываются глобально для всего изображения
  - Пространственная информация (расположение пикселей различной яркости) полностью игнорируется
- Это можно использовать для сравнения изображений (см. слайд 21 – мера пиковости):



# Гистограммы

---

- Свойства:
  - Рассчитываются глобально для всего изображения
  - Пространственная информация (расположение пикселей различной яркости) полностью игнорируется
- Однако при анализе сложных сцен это может мешать
  - Сильно различные «с виду» сцены могут иметь очень похожие гистограммы (но дает число классов объектов)



# Пороговая фильтрация

---

- Яркий объект на темном фоне
  - Выбрать величину  $T$  разделяющую яркость объекта и фона
  - Каждый пиксель  $(x,y)$  яркость которого  $I(x,y) > T$  принадлежит объекту



0

255



# Как определить величину $T$ ?

---

- В каждом конкретном случае хотим уметь рассчитать правильный порог
- Вариант решения – анализ гистограммы изображения





# Автоопределение величины $T$

---

- Можно использовать следующее:
  1. Предположение о яркости объектов
  2. Размеры объектов
  3. Площадь изображения занятого объектом
  4. Количество различных типов объектов
  
- Вопрос – как?



# Автоопределение величины $T$

---

- Метод P-tile:
  - Если знаем (предполагаем) что объект занимает  $P\%$  площади
  - $T$  устанавливаем так, чтобы отсечь  $P\%$  пикселей на гистограмме



# Поиск пиков в гистограмме

---

1. Найти соседние локальные максимумы в гистограмме  $g_i$
2. Рассчитать меру «пиковости» для  $g_i$
3. Отфильтровать пики со слишком маленькой «пиковостью».
4. Для оставшихся найти самые «низкие» точки между пиками – это и будут пороги.



---

**ОВОСТИ»**

- 
- Это проблема – много «лишних» локальных максимумов

\_\_\_\_\_

**лленно**

\_\_\_\_\_

93 пика



- 
- Сглаживание посредством усреднения соседних значений
    - *Свертка одномерным box-фильтром*

ГГ

ГИС

Сглажено 1 раз

2 раза

3 раза

54 пика

21 пика

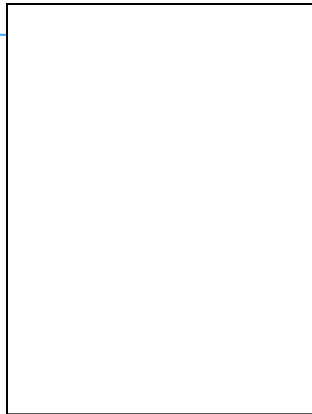
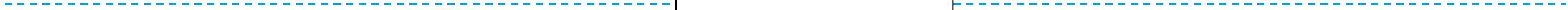
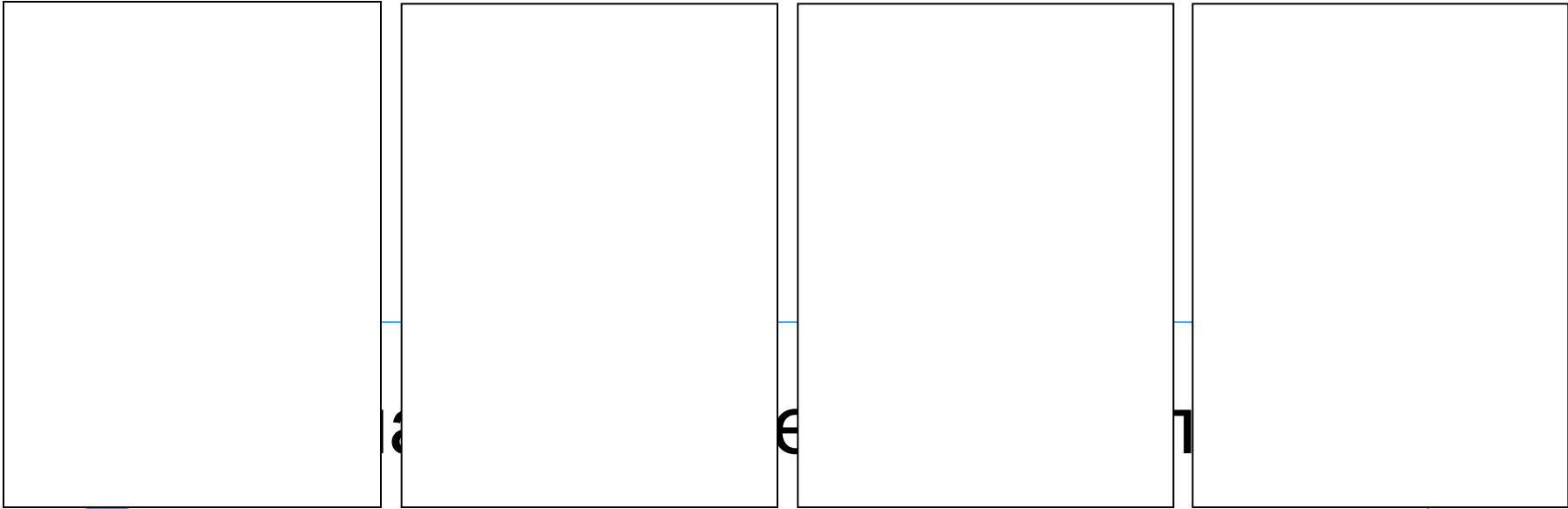
11 пиков

«Пиковость» проходят 18

«Пиковость» проходят 7

«Пиковость» проходят 4 peaks



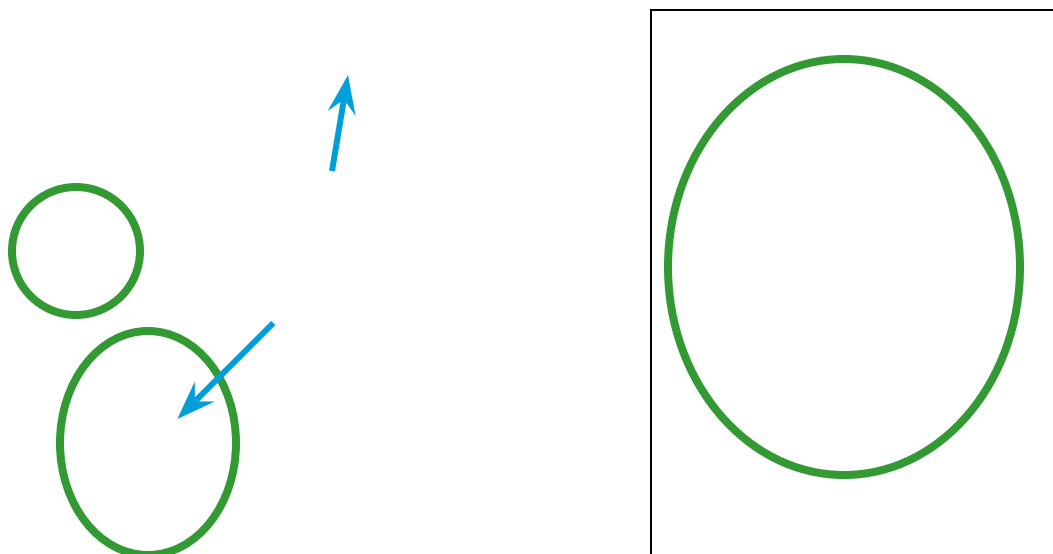


# Адаптивный порог

---

- Проблема:

- Яркость фона может быть разной в разных частях изображения
- Единый порог не подойдет





- 
1. Для каждого пикселя изображения  $I(x, y)$ :
    - 1) В окрестности пикселя радиуса  $r$  высчитывается индивидуальная для данного пикселя величина  $C$ ;
    - 2) Если  $I(x, y) - C > T$ , результат 1, иначе 0;

Варианты выбора  $C$  по окрестности  $(x, y)$ :

- $C = \text{среднее}$
- $C = \text{медиана}$
- $C = (\text{min} + \text{max}) / 2$

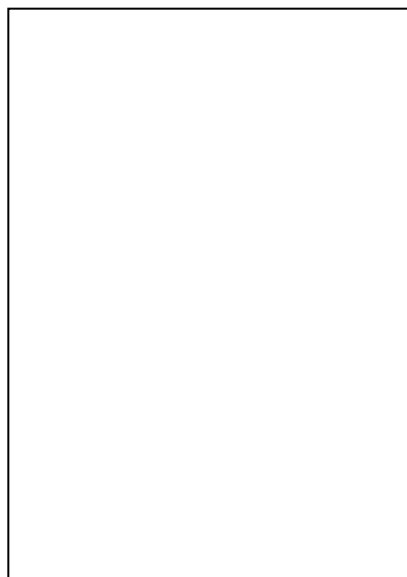
Адаптивный порог

Обратите внимание – начинаем учитывать пространственную информацию

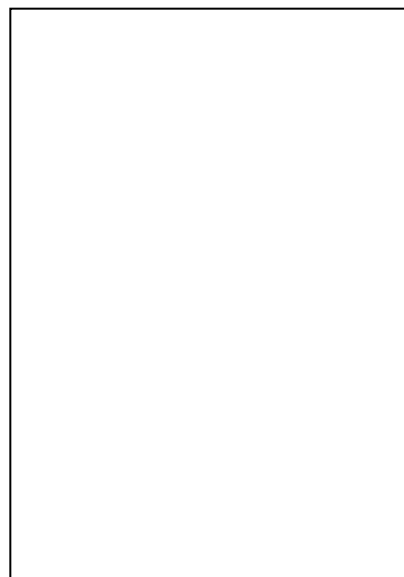
# Адаптивный порог

---

- Другая формулировка
  - Приближение фона усреднением
  - Вычитание фона -  $I(x, y) - C(x, y) > T$



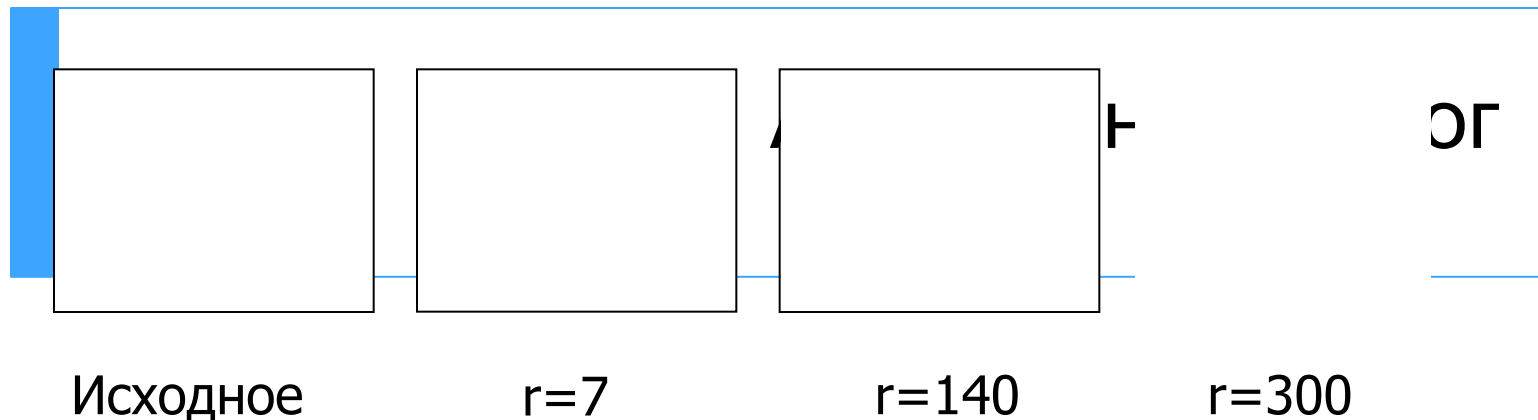
Исходное



$I(x, y) - C(x, y), r=18$



- Хорошо работает
  - Когда размер искомого объекта заметно меньше размера оцениваемой окрестности
- Хуже работает,
  - Когда объект велик по сравнению с самим изображением



# Метод Оцу (Otsu, Википедия)

В области компьютерного распознавания образов и обработки изображения, Метод Оцу используется для выполнения пороговой бинаризации полутоновых изображений. Алгоритм предполагает наличие в изображении двух классов пикселей (например, текстовые и фоновые) и ищет оптимальный порог, разделяющий эти два класса так, чтобы их внутриклассовая дисперсия была минимальна [1]. Оригинальный метод был улучшен для поддержки многоуровневых порогов [2], и на него ссылаются как на Мульти Оцу Метод.

Метод Оцу ищет порог ( $t$ ), уменьшающий дисперсию внутри класса, которая определяется как взвешенная сумма дисперсий двух классов:

Веса  $\omega_i$  - это вероятности двух классов разделенных порогом  $t$ , а  $\sigma_i^2$  - дисперсия этих классов.

Оцу показал, что минимизация дисперсии внутри класса - это то же самое, что и максимизация дисперсии между классами [1]:

которая выражается в терминах вероятности  $\omega_i$  и средне арифметического класса  $\mu_i$ , которые в свою очередь могут обновляться итеративно. Эта идея привела к эффективному алгоритму:

1. Вычислить гистограмму и вероятность для каждого уровня интенсивности.
2. Вычислить начальные значения для  $\omega_i(0)$  и  $\mu_i(0)$ .
3. Для каждого значения порога от  $t = 1 \dots$  до максимальной интенсивности:  
Обновляем  $\omega_i$  и  $\mu_i$   
Вычисляем  $\sigma_b^2(t)$ .  
Если  $\sigma_b(t)$  больше, чем имеющееся, то запоминаем  $\sigma_b$  и значение порога  $t$ .
4. Искомый порог соответствует максимуму  $\sigma_b^2(t)$ .

## Ссылки

[1] N.Otsu (1979). «A threshold selection method from gray-level histograms». *IEEE Trans. Sys., Man., Cyber.* **9**: 62-66.

[2] Ping-Sung Liao and Tse-Sheng Chen and Pau-Choo Chung (2001). «A Fast Algorithm for Multilevel Thresholding». *J. Inf. Sci. Eng.* **17**: 713-727.

# Метод Оцу. Пример применения (Википедия)

---

Original image

An example image thresholded using  
Otsu's algorithm



# Кластеризация k-средних

---

- Способ определения нескольких порогов одновременно
- Нужно заранее знать  $k$  – количество диапазонов яркостей
  - Можно найти  $k$  по гистограмме с помощью анализа «ПИКОВОСТИ»



- 
- **Входные данные** – набор векторов  $n$ -мерного пр-ва  $v_i$   $i=1, \dots, p$ .
  - **Выходные данные** – центры кластеров  $m_j$   $j=1, \dots, k$  и принадлежность  $v_i$  к кластерам

1. Случайным образом выбрать  $k$  средних  $m_j$   $j=1, \dots, k$ ;
2. Для каждого  $v_i$   $i=1, \dots, p$  подсчитать расстояние до каждого из  $m_j$   $j=1, \dots, k$ ,
3. Отнести (приписать) те  $v_i$  к кластеру  $j$ , для которых расстояние до  $m_j$  меньше, чем до других;
4. Пересчитать средние  $m_j$   $j=1, \dots, k$  по всем кластерам;
5. Повторять шаги 2, 3 пока кластеры не перестанут изменяться;

### Алгоритм $k$ -средних



# k-средних для сегментации

---

- Если изображение одноканальное
  - $v_i = I(x, y)$  – работаем в одномерном пространстве
  - Получается итеративный алгоритм пересчета порога
- Если изображения трехканальное (RGB)
  - $v_i = (R(x, y), G(x, y), B(x, y))$  – работаем в трехмерном пространстве
- Можно работать и с многоканальными изображениями
  - Например – RGB + инфракрасный канал





## Общие недостатки описанного

---

- Игнорируется пространственное расположение пикселей
  - За исключением адаптивного порога, но и там соседство не учитывается
- Перейдем к методам, учитывающим взаимное расположение пикселей



# Понятие связности

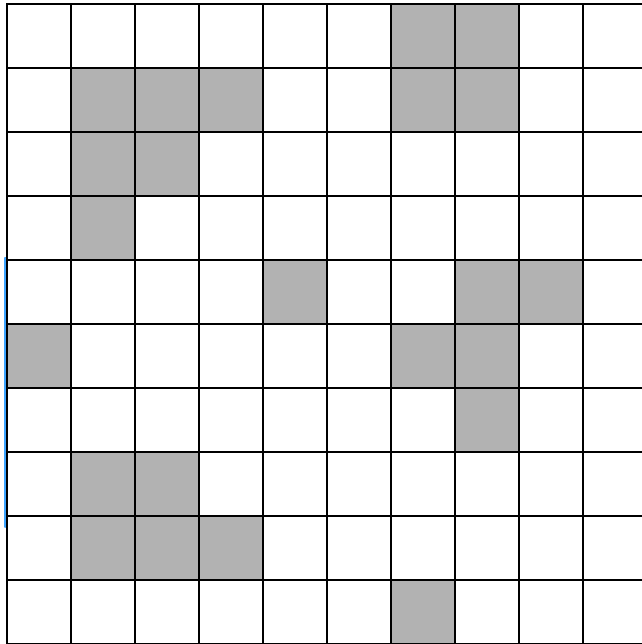
---

- Определение связной области:
  - Множество пикселей, у каждого пикселя которого есть хотя бы один сосед, принадлежащий данному множеству.
  - Соседи пикселей:

4-СВЯЗНОСТЬ

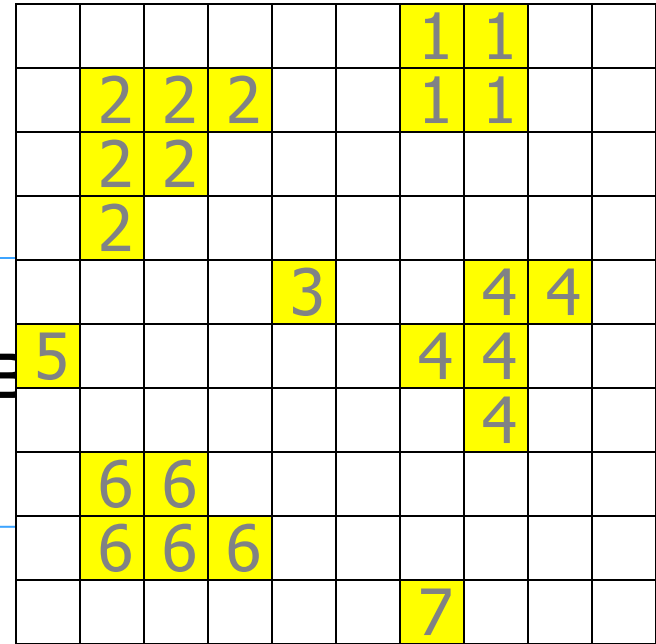
8-СВЯЗНОСТЬ





**Бинарное изображение**

Гка СВ



**Размеченное изображение**

# Разрастание регионов (Region growing)

---

- Простая идея – начиная с некоторого “семени” обходить пиксели и объединять в области пока выполняется условие однородности



# Что необходимо определить

---

- Критерий однородности
  - Гистограмма содержит не больше 1 значительного пика
  - Отклонение любого пикселя от средней яркости  $< T_{avg}$

□ Разн



- 
- ▶ □ «Слабая» граница между регионами (только для слияния) – позже

# Алгоритм разрастания регионов

Среднее: 1

Среднее: 1.125

1	1	1	1	1	1	1	2
1	1	1	1	1	1	1	0
3	1	4	9	9	8	1	0
1	1	8	8	8	4	1	0
1	1	6	6	6	3	1	0
1	1	5	6	6	3	1	0
1	1	5	6	6	2	1	0
1	1	1	1	1	1	0	0

Пример  $\delta = 1$



# Алгоритм разрастания регионов



Пример  $\delta = 1$



---

Сканируем изображение сверху вниз, слева направо:

1. if  $|I(A) - Cl_{avg}(B)| > \delta$  and  $|I(A) - Cl_{avg}(C)| > \delta$  -  
создаем новую область, присоединяем к ней пиксел A
2. if  $|I(A) - Cl_{avg}(B)| \leq \delta$  xor  $|I(A) - Cl_{avg}(C)| \leq \delta$  -  
добавить A к одной из областей
3. if  $|I(A) - Cl_{avg}(B)| \leq \delta$  and  $|I(A) - Cl_{avg}(C)| \leq \delta$  :

1. **Разрастание регионов**  
сливаем области B и C.

2.  $|Cl_{avg}(B) - Cl_{avg}(C)| > \delta$  -  
добавляем пиксел A к тому классу, отклонение от которого минимально.

$I(A)$  – яркость пиксела A

$Cl_{avg}(B)$  – средняя яркость области к которой принадлежит B



# Разделение областей

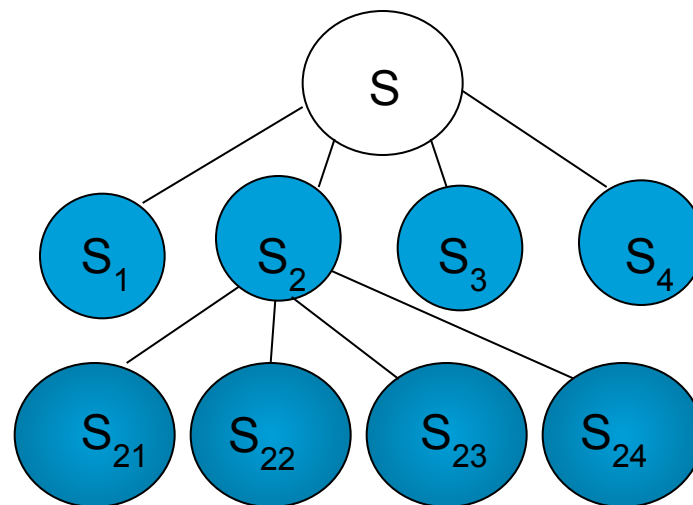
---

1. Первый шаг – всё изображение это одна область, поместить область в стек
2. Пока стек не пуст
  - Взять область  $S$  из стека
  - Проверить область на однородность
  - Если область неоднородна
    - разделить ее, новые области поместить в стек
  - Если область однородна
    - область больше не трогаем



# Что необходимо определить 2

- Правило разделения областей
  - Распространенный вариант – на 4 части, как квадродерево



Просто реализовать, но границы получившихся областей  
вряд ли будут соответствовать границам объектов

# Алгоритм разбиения (split)

---

1	1	1	1	1	1	1	2
1	1	1	1	1	1	1	0
3	1	4	9	9	8	1	0
1	1	8	8	8	4	1	0
1	1	6	6	6	3	1	0
1	1	5	6	6	3	1	0
1	1	5	6	6	2	1	0
1	1	1	1	1	1	0	0

Пример



# Алгоритм разбиения (split)

---

1	1	1	1	1	1	1	2
1	1	1	1	1	1	1	0
3	1	4	9	9	8	1	0
1	1	8	8	8	4	1	0
1	1	6	6	6	3	1	0
1	1	5	6	6	3	1	0
1	1	5	6	6	2	1	0
1	1	1	1	1	1	0	0

Первое разбиение



# Алгоритм разбиения (split)

---

1	1	1	1	1	1	1	2
1	1	1	1	1	1	1	0
3	1	4	9	9	8	1	0
1	1	8	8	8	4	1	0
1	1	6	6	6	3	1	0
1	1	5	6	6	3	1	0
1	1	5	6	6	2	1	0
1	1	1	1	1	1	0	0

Второе разбиение



# Алгоритм разбиения (split)

---

1	1	1	1	1	1	1	2
1	1	1	1	1	1	1	0
3	1	4	9	9	8	1	0
1	1	8	8	8	4	1	0
1	1	6	6	6	3	1	0
1	1	5	6	6	3	1	0
1	1	5	6	6	2	1	0
1	1	1	1	1	1	0	0

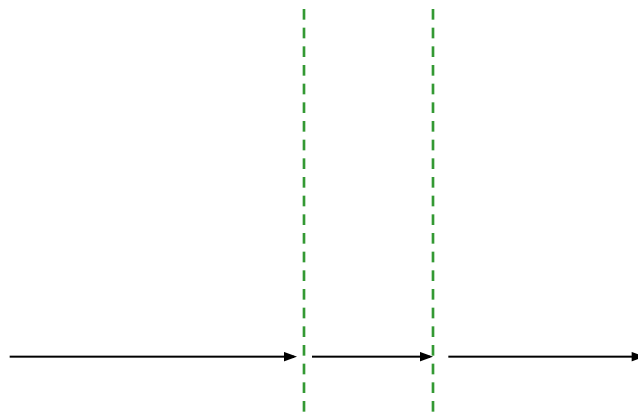
Третье разбиение



# Что необходимо определить 3

---

- Правило разделения областей – более умно
  - Найти в гистограмме пики, разделить гистограмму по ним
  - Для каждой части гистограммы найти связанные компоненты – это будут новые области



Реализовать сложнее, работает дольше



# Слияние областей

---

1. Первый шаг – каждый пиксель это отдельная область, поместить все области в стек
  
2. Пока стек не пуст
  - i. Взять область  $S$  из стека, для всех соседних областей  $S_i$ :
    - Проверить  $S' = S \cup S_i$  на однородность
    - Если  $S'$  однородна –
      - Слить  $S$  и  $S_i$ ,  $S'$  поместить в стек,  $S_i$  из стека удалить, перейти на 2
    - Если область неоднородна
      - Пробуем другого соседа

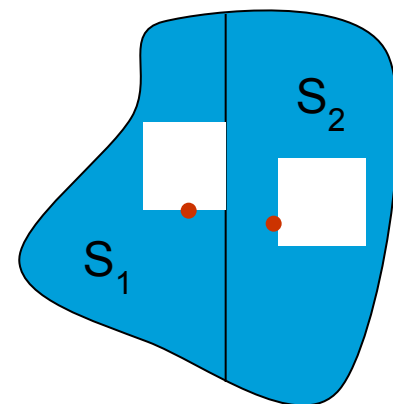




# Алгоритм «фагоцита»

---

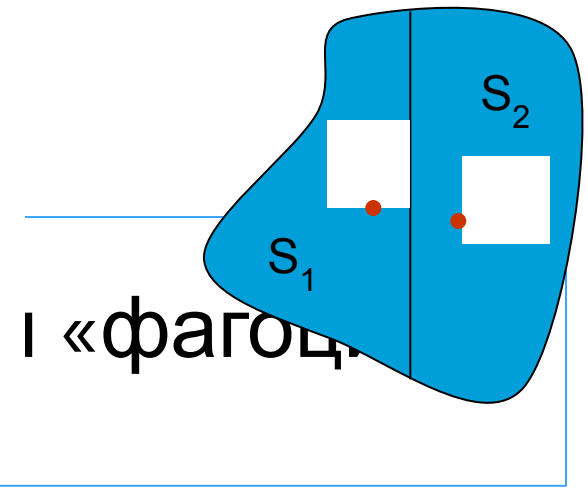
- Истаивание границ
  - Убирает слабые границы
- «Слабость границ» определяется по разности яркостей граничных пикселей



---

*клетка способная захватывать и  
переваривать посторонние тела*





# Алгоритм «фагоцита»

---

- Слить две области если:

Где  $r_1$  и  $r_2$  — радиусы областей  $S_1$  and  $S_2$

- Слить две области если:



# Алгоритмы разбиения и слияния

---

## □ Недостатки:

### □ Разбиение

- Может дать слишком много регионов
- Если использовать квадродерево, границы скорее всего будут неверны

### □ Слияние

- Долго работает, если начинать с индивидуальных пикселей

## □ Вывод:

- Нужен комбинированный метод!
- 



# Алгоритм разбиения/слияния (split and merge)

---

## □ Идея:

- Сначала провести разбиение на небольшие однородные области
  - Обычно используется принцип квадродерева
- Затем слить между собой те из них, которые вместе не нарушат требование однородности
  - Продолжать до тех пор, пока остаются регионы которые можно объединить



# Алгоритм разбиения/слияния (split and merge)

1	1	1	1	1	1	1	2
1	1	1	1	1	1	1	0
3	1	4	9	9	8	1	0
1	1	8	8	8	4	1	0
1	1	6	6	6	3	1	0
1	1	5	6	6	3	1	0
1	1	5	6	6	2	1	0
1	1	1	1	1	1	0	0

Слияние



# Алгоритм разбиения/слияния (split and merge)

1	1	1	1	1	1	1	2
1	1	1	1	1	1	1	0
3	1	4	9	9	8	1	0
1	1	8	8	8	4	1	0
1	1	6	6	6	3	1	0
1	1	5	6	6	3	1	0
1	1	5	6	6	2	1	0
1	1	1	1	1	1	0	0

Результат



# Сравним с разрастанием регионов

1	1	1	1	1	1	1	2
1	1	1	1	1	1	1	0
3	1	4	9	9	8	1	0
1	1	8	8	8	4	1	0
1	1	6	6	6	3	1	0
1	1	5	6	6	3	1	0
1	1	5	6	6	2	1	0
1	1	1	1	1	1	0	0

1	1	1	1	1	1	1	2
1	1	1	1	1	1	1	0
3	1	4	9	9	8	1	0
1	1	8	8	8	4	1	0
1	1	6	6	6	3	1	0
1	1	5	6	6	3	1	0
1	1	5	6	6	2	1	0
1	1	1	1	1	1	0	0

Результат





# Сравним подходы

---

- Сегментация на основе областей
  - В результате всегда замкнутые границы областей
  - Использование многоканальных изображений (RGB, RGB + ИК) обычно улучшает результаты
- Сегментация на основе границ
  - Границы обычно лучше локализованы



# Алгоритм водораздела (watershed)

---

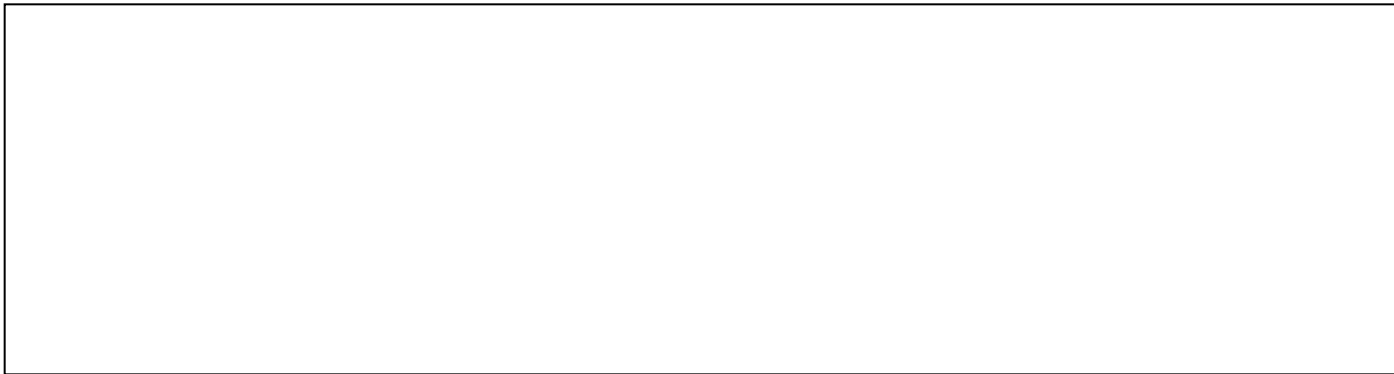
- Идея метода:
  - Большие значения градиента соответствуют резким переходам на изображении
  - Рассмотрим абсолютную величину градиента как карту высот ландшафта
  - Там где резкие границы – получатся «стены»
  - Будем «лить воду» в «ямы» и искать получающиеся «озера»



# Алгоритм водораздела

---

*Область водораздела, бассейн (catchment basin):* область в которой поток из всех точки «стекает» к одной общей точке



Слева – профиль интенсивностей изображения, справа – локальные минимумы определяют бассейны, локальные максимумы – линии водораздела.



# Алгоритм водораздела

---

- Алгоритм, как и разбиение дает множество небольших регионов
  - Очень чувствителен к шуму – ищет все локальные минимумы

Абс. величина  
градиента

Результат по  
данному градиенту

Градиент  $< 10$   
обращен в 0



# Алгоритм «погружения»

---

Алгоритм «погружения» (immersion) :

Начнем с самых «глубоких» (темных) пикселей  
(они определяют начальные бассейны)

Для каждой яркости  $k$ :

Для каждой связной компоненты пикселей яркости  $k$ :

Если прилежит только к одному существующему бассейну

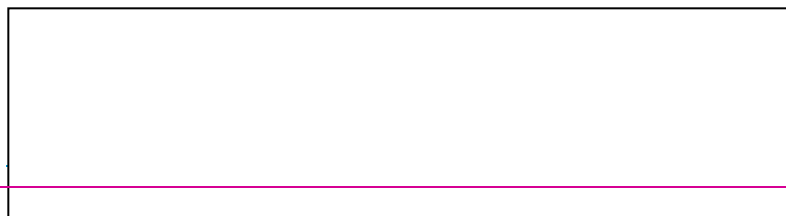
Добавить компоненту к бассейну

Если прилежит более чем к одному существующему  
бассейну

Пометить как границу (водораздел)

Иначе – создать новый бассейн

Аналог – вода медленно поднимается, пока не погрузятся в нее  
водоразделы



# Алгоритм tobogganing

---

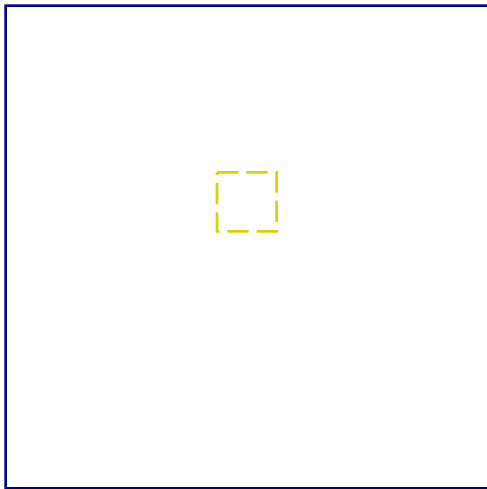
## □ Идея:

- Из каждого пикселя «спускаемся» в локальный минимум среди его соседей
- Спускаемся до тех пор, пока есть куда спускаться
- Пиксели «спустившиеся» в один минимум – одна область

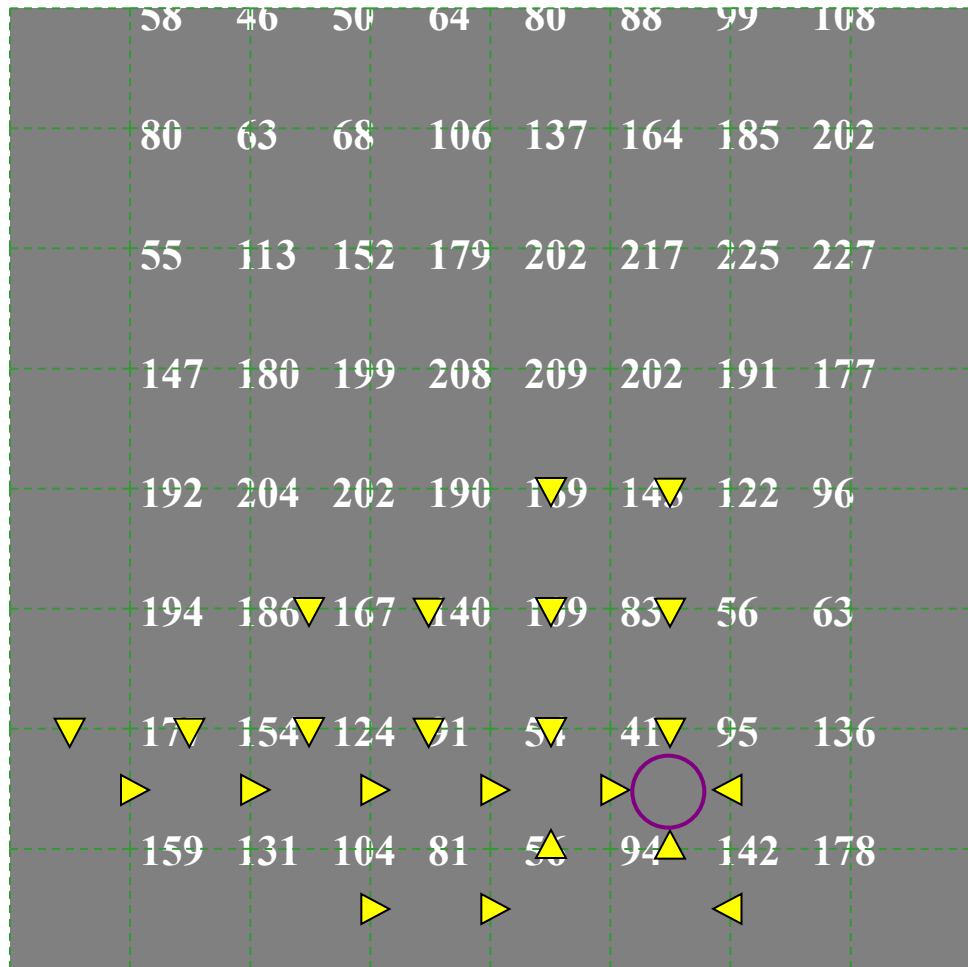
Как с горы на санках



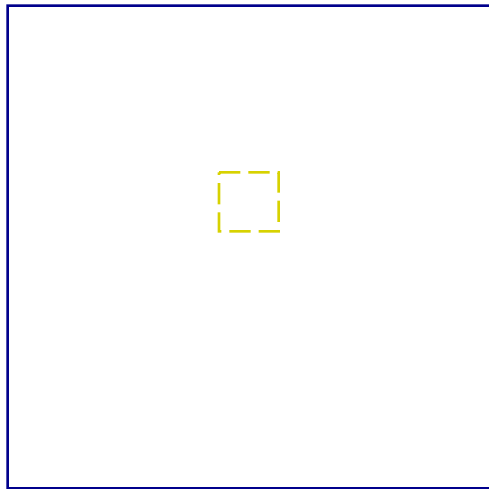
# Алгоритм tobogganing



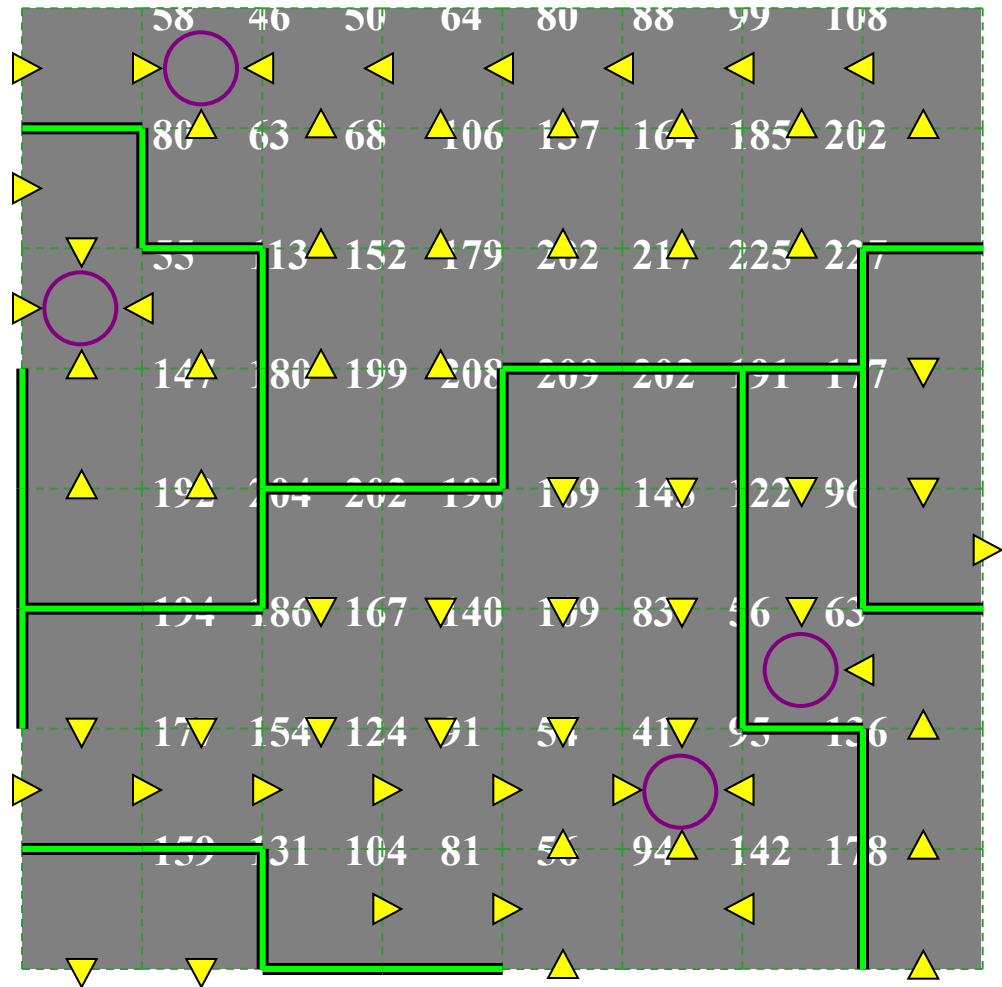
- Из каждого пикселя «спускаемся» в локальный минимум среди его соседей
- Спускаемся до тех пор, пока есть куда спускаться
- Пиксели «спустившиеся» в один минимум – одна область



# Алгоритм tobogganing



- Из каждого пикселя «спускаемся» в локальный минимум среди его соседей
- Спускаемся до тех пор, пока есть куда спускаться
- Пиксели «спустившиеся» в один минимум – одна область





# Tobogganing и водораздел

---

- В зависимости от задачи можно анализировать
  - само изображение
  - абсолютную величину его градиента
  - distance transform изображения (в каждой точке хранится расстояние до ближайшей границы)
  - Часто генерируют слишком много регионов, как и разделение
    - Требуется постобработка для слияния
  - В комбинации с distance transform хорошо для перекрывающихся регионов



---

# Методы теории графов



# Методы теории графов

---

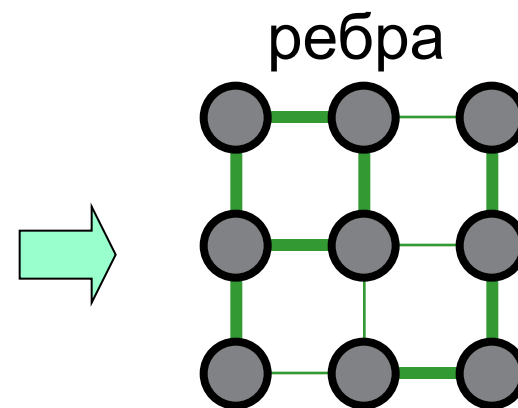
- Теория графов – хороший инструмент для работы с изображениями
  - Хорошая теоретическая база
  - Много проработанных методов
  - Изображение легко «превращается» в граф
- Математические модели теории графов хорошо применимы в частности для сегментации



# Граф и изображение

---

- Изображение превращается во взвешенный неориентированный граф
  - Пиксели – вершины графа
  - Ребра – связи между соседними пикселями
  - Вес ребер пропорционален «похожести» пикселей



# Критерии «похожести» пикселей

- По расстоянию
- По яркости
- По цвету
- По текстуре



# Сегментация с помощью разрезов графа

---

- Создать граф
- Разрезать граф
- Каждую связную компоненту после разреза рассматривать как отдельную область



# Разрез графа

---

- Разрез графа превращает граф в два несвязанных друг с другом подграфа



# Разрез графа

---

- $G=(V,E)$ 
  - Непересекающиеся подмножества вершин  $A$  и  $B$  из  $V$
  - Удаляем все ребра, связывающие  $A$  и  $B$

$Cut(A,B)$  – мн

множеств  $A$  и  $B$





# Разрез графа

---

- Если множества  $A$  и  $B$  не заданы заранее – разрезать граф можно по-разному:
- **Минимальный разрез** – разрез, превращающий граф в несвязный, с минимальной суммой весов удаленных ребер



# Минимальный разрез хорош не всегда

---


- На данном рисунке вес ребер графа показан расстоянием между вершинами



# Нормализованный разрез графа (Normalized cut)

---

- ▣ **Другая мера разреза** – измеряет «похожесть» двух групп вершин, нормированную на «объем», занимаемый ими в графе

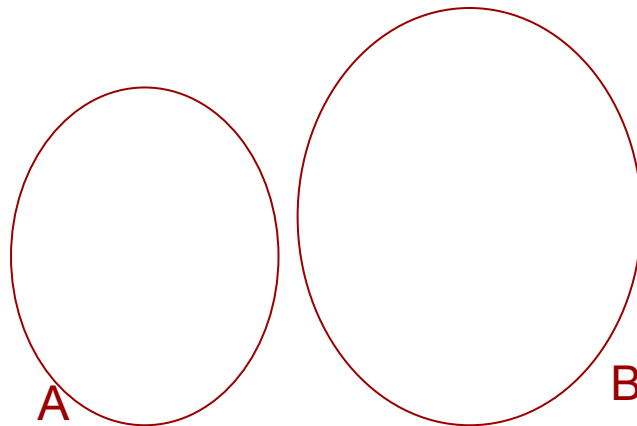
Все ребра графа 



# Минимальный нормализованный разрез

---

- ▣ **Минимальный нормализованный разрез** – разрез, превращающий граф в несвязный, с минимальной величиной  $NCut$
  
- ▣ *Как его найти?*



# New 2-step algorithm (cc-2011, local)

---



# Morphological amoebas

---

Gabor filtering ([ru](#), [en](#))



# Подытожим:

---

- Рассмотрели следующие методы
  - Использующие края
    - Edge-based
  - Пороговой фильтрации
    - Thresholding
  - k-средних
    - k-means
  - Разрастания регионов
    - Region growing
  - Разделения / слияния
    - Split and merge
  - Водораздела
    - Watershed, tobogganing
  - Нормализованный разрез графа
    - Normalized cut





**УНИВЕРСИТЕТ  
ЛОБАЧЕВСКОГО**

Институт информационных технологий, математики и  
механики  
Кафедра математического обеспечения и  
суперкомпьютерных технологий

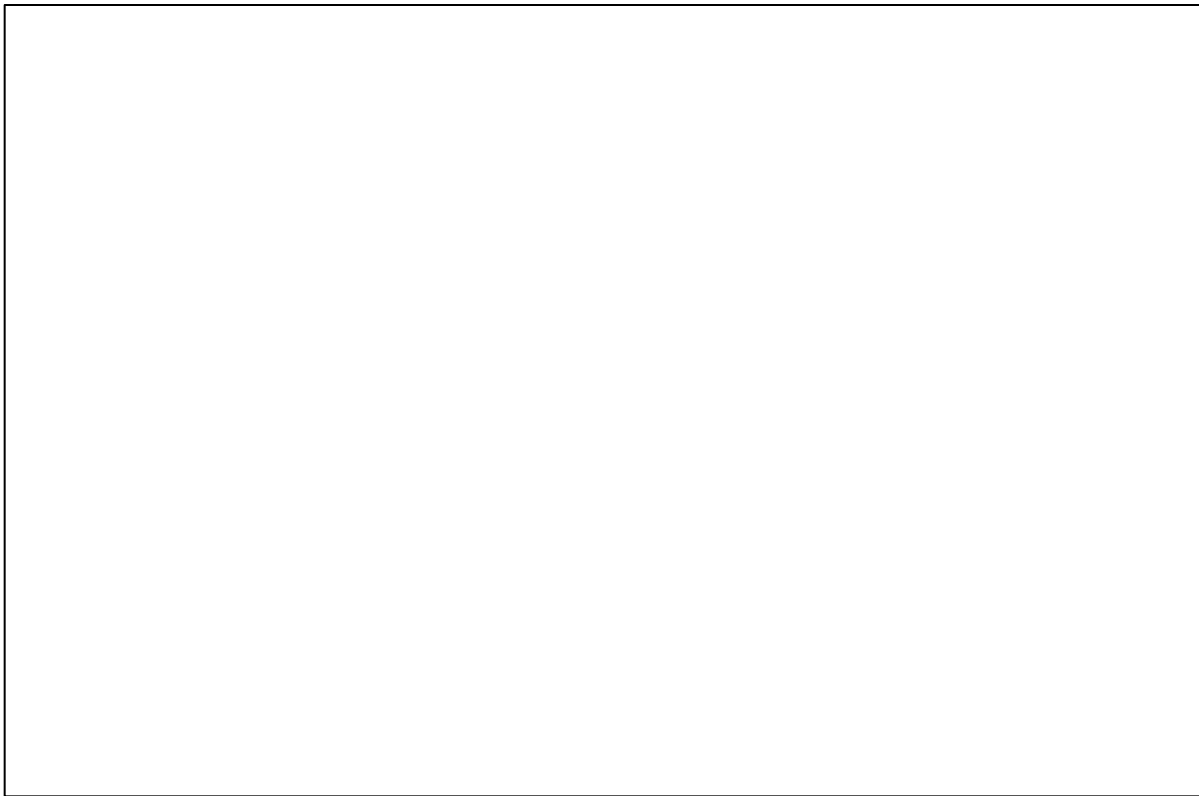
## Обработка изображений

**Анализ областей после сегментации**  
Владимир Вежневцев, Антон Коношин  
Александр Вежневцев

**Курс – «Введение в компьютерное зрение»**  
МГУ ВМК, Graphics & Media Lab,  
Осень 2006



Какие параметры формы областей помогут различить объекты на этом примере?



# Свойства области

---

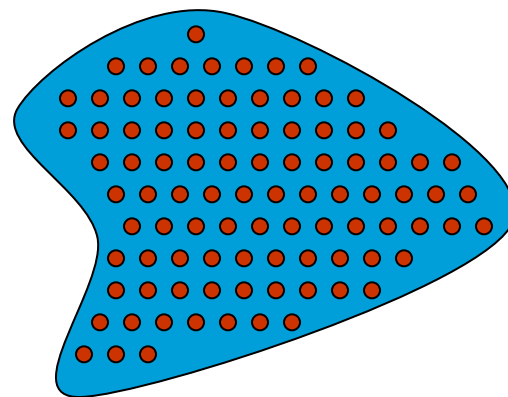
- Характеристики границы области
  - См. предыдущую лекцию
- Площадь
- Кол-во «дырок» внутри
- Центр масс
- Периметр
- Компактность
- Моменты
- Ориентация главной оси
- Цвет/яркость



# Площадь

---

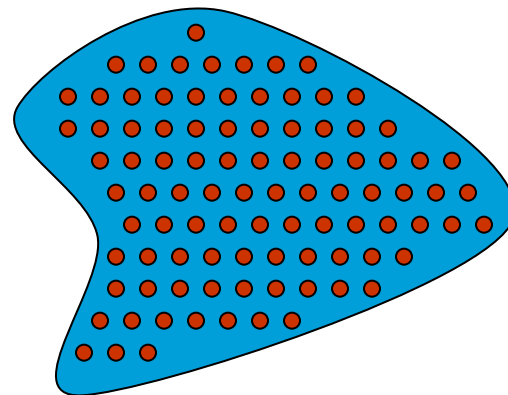
- Кол-во пикселей в области



# Центр масс

---

□ Центр масс:

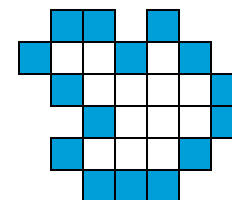


# Периметр и компактность

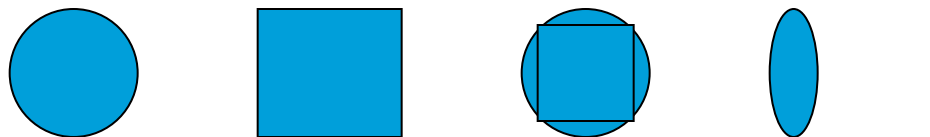
---

▣ **Периметр** – количество пикселей принадлежащих границе области

▣ **Компактность**



▣ На  $\mathbb{R}^2$  компактная фигура – круг,



- 
1. Пиксель лежит на границе области, если он сам принадлежит области и хотя бы один из его соседей области не принадлежит. (внутренняя граница)
  2. Пиксель лежит на границе области, если он сам **не** принадлежит области и хотя бы один из его соседей области принадлежит. (внешняя граница)

Периметр зависит также от того, как именно связность используется для определения соседей.

# Моменты

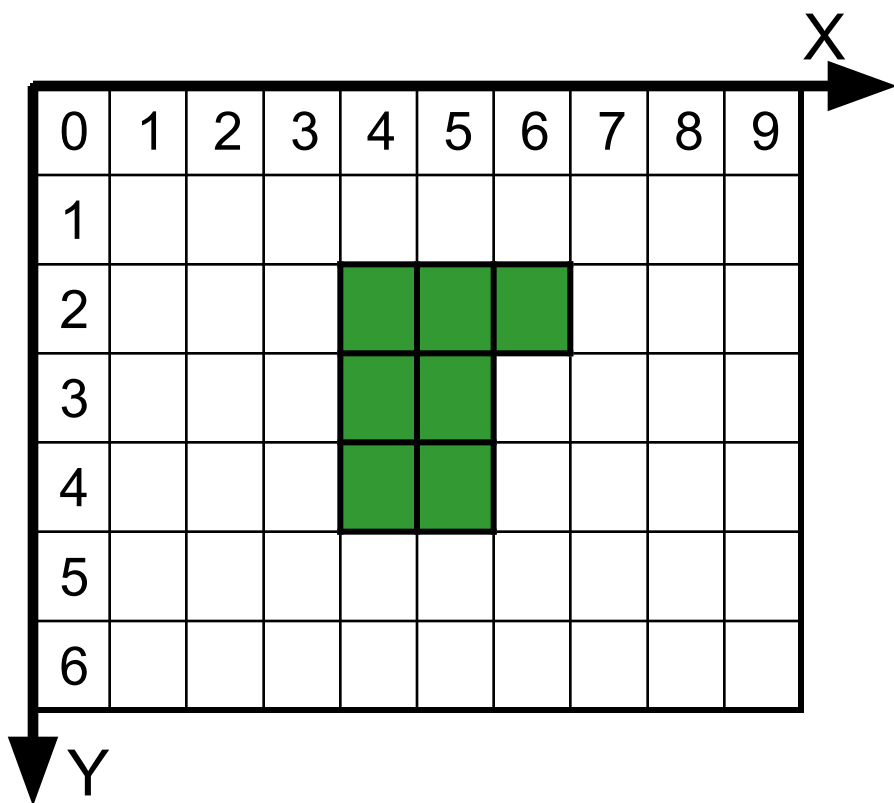
---

Дискретный момент  $m_{ij}$  области определяется следующим образом:

- значение пикселя изображения



# Моменты



i	j	$M_{ij}$
0	0	7
1	0	33
0	1	20
2	0	159
0	2	64
1	1	93

Площадь

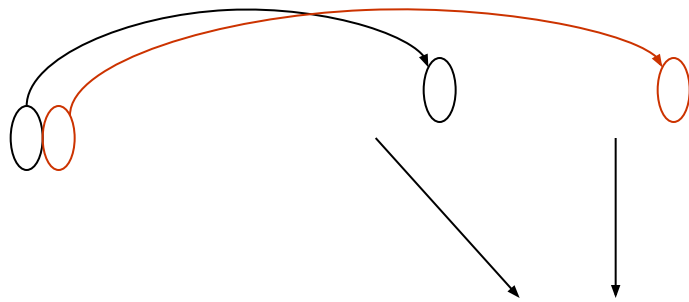
Моменты  
инерции



# Центральные моменты

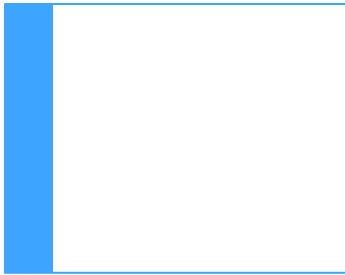
---

- Инвариантны к переносу



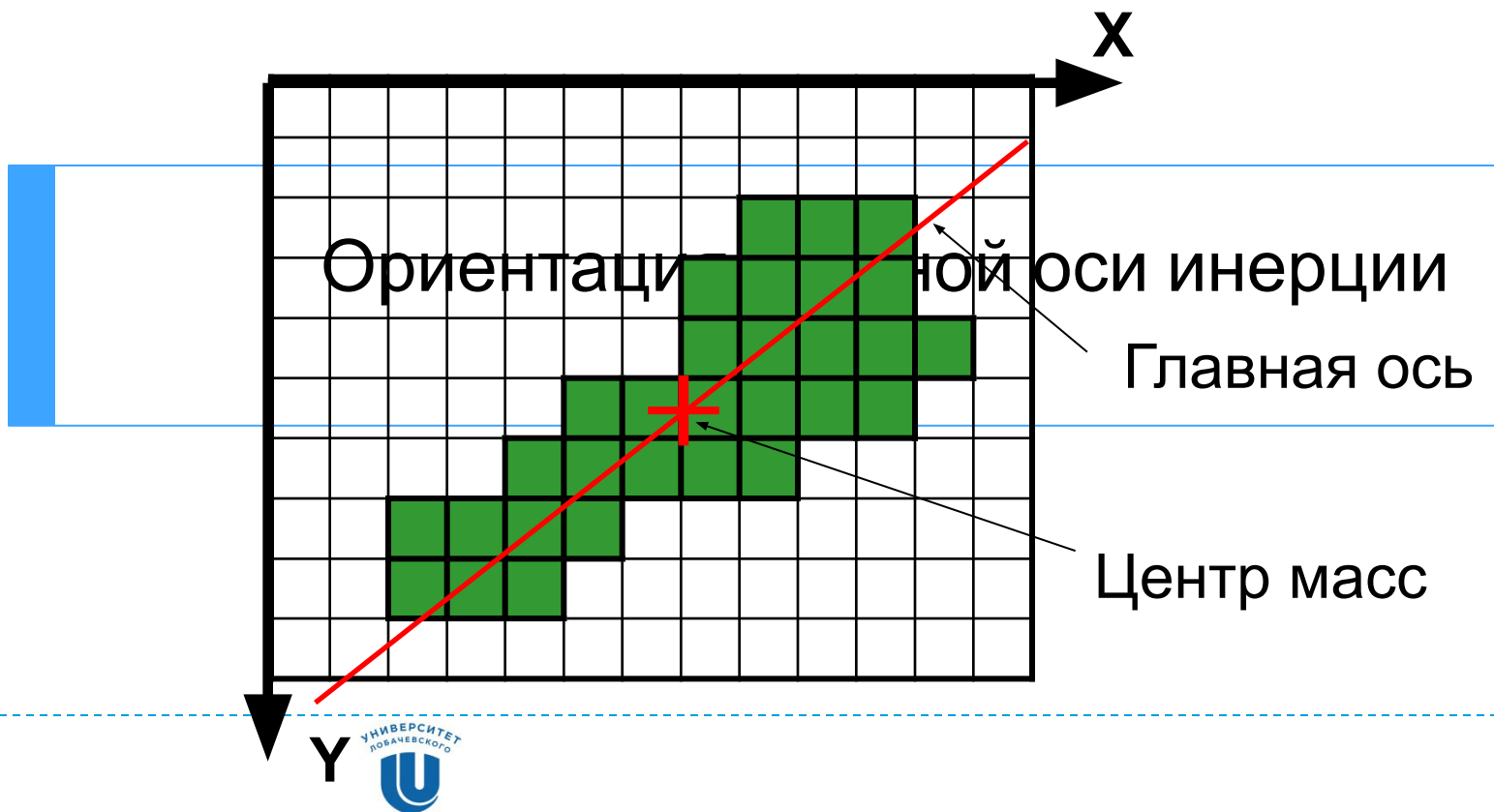
Центр масс области





**МОМЕНТЫ**





# Моменты $H_i$

---

□ ↓

---

алированию





original image



mirror

mirror, scale, rotate 4:

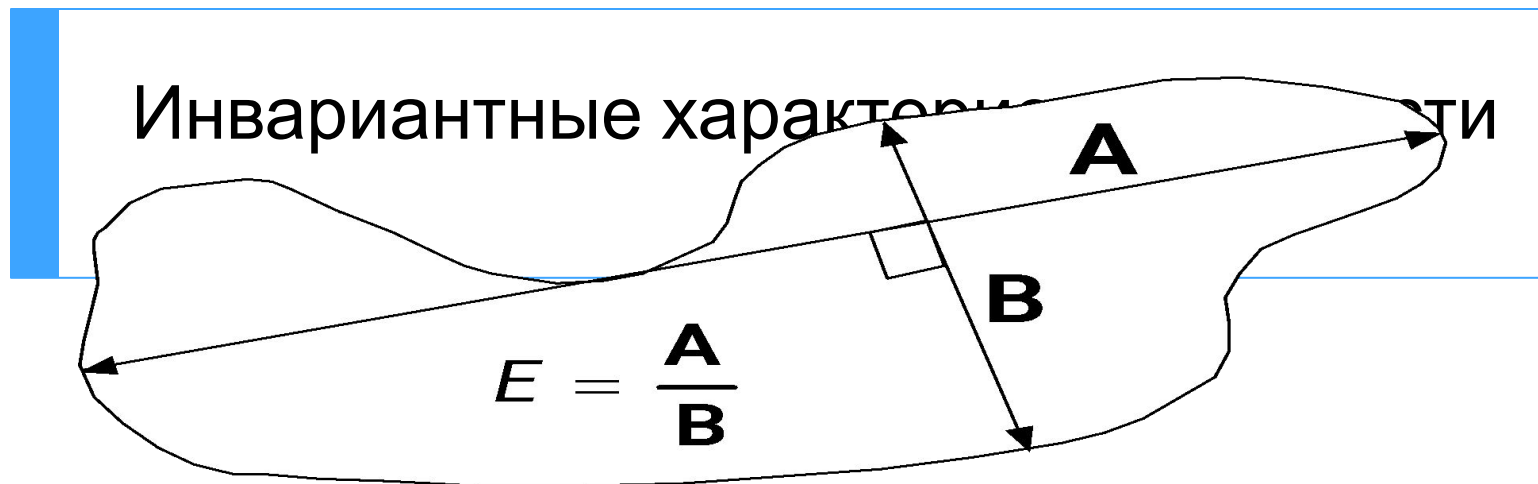
scale

scale, rotate 2°

имер



- Удлиненность, нецентрированность (эксцентриситет)



# Цвет, яркость

---

- Цвет и яркость области тоже хорошие признаки.

## Варианты

- Гистограмма яркости, цветов в данной области
- Средняя яркость, средний цвет
- Дисперсия яркости, цветов (R, G, B) внутри области



# Немного о машинном обучении

---

- Мы рассмотрели сейчас методы «низкого уровня»
  - Они анализируют небольшое кол-во «простой» информации
- При рассказе о машинном обучении будут упомянуты методы производящие более «умный» анализ изображения





# Благодарности

---

- ❑ В лекции использовались иллюстрации из курсов:
  - ❑ Nick Krouglicof
    - ❑ Memorial University of Newfoundland
  - ❑ K. K. Biswas
    - ❑ University of Central Florida
  - ❑ Alper Yilmaz
    - ❑ University of Central Florida
  - ❑ Dr. Boyer
    - ❑ Ohio State University

