

Компьютерное зрение. Бинаризация и метод Оцу

Лекция 3

Лектор: Демяненко Я.М.,
ФИИТ, 2026



От лекции 3 к лекции 4

- **Вспомним:**
 - • Гистограмма — распределение яркостей
 - • Эквиализация — выравнивание
 - • Бимодальная гистограмма → порог
- **Сегодня:**
 - • Как отделить объект от фона?
 - • Что такое бинаризация?
 - • Как автоматически выбрать порог?

Пороговое преобразование

Идея:

$f(v) = 1$, если $v > t$

$f(v) = 0$, иначе

где t — порог

Применение:

- Выделение объекта от фона
- Распознавание текста (OCR)
- Анализ микроскопических изображений



Виды пороговой бинаризации

1. Верхняя пороговая:

$v > t \rightarrow 1$, иначе 0

2. Нижняя пороговая:

$v < t \rightarrow 1$, иначе 0

3. По диапазону:

$t1 < v < t2 \rightarrow 1$, иначе 0



Вырезание цветового диапазона

$$s_i = \begin{cases} 0,5, & \text{если } \left[|r_j - a_j| > \frac{W}{2} \right] \text{ для любого } 1 \leq j \leq n; \\ r_i, & \text{в остальных случаях} \end{cases} \quad i = 1, 2 \dots n$$



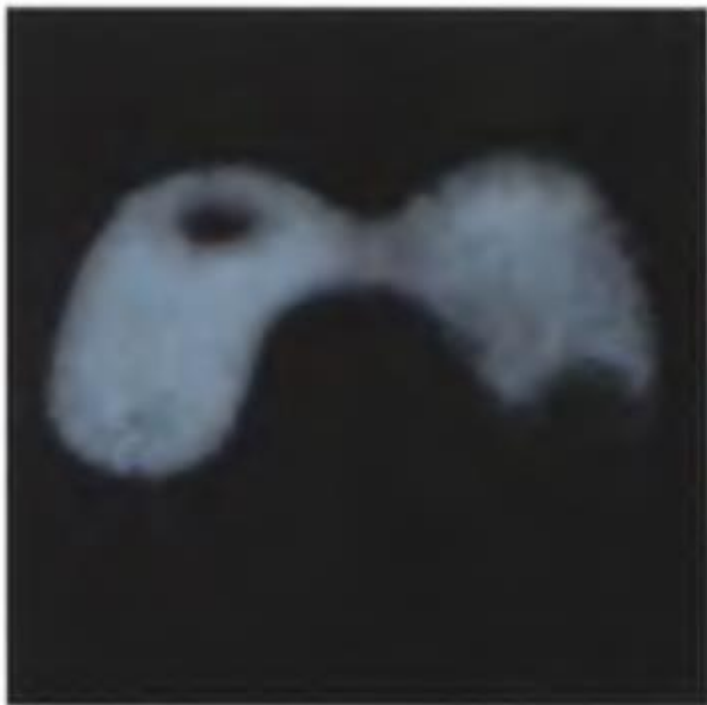
$$s_i = \begin{cases} 0,5, & \text{если } \sum_{j=1}^n (r_j - a_j)^2 > R_0^2; \\ r_i, & \text{в остальных случаях;} \end{cases} \quad i = 1, 2 \dots n$$



Применение:

- Выделение объектов определённого цвета
- Медицина (анализ клеток)
- Промышленность (контроль качества)

Квантование по яркости



Идея:

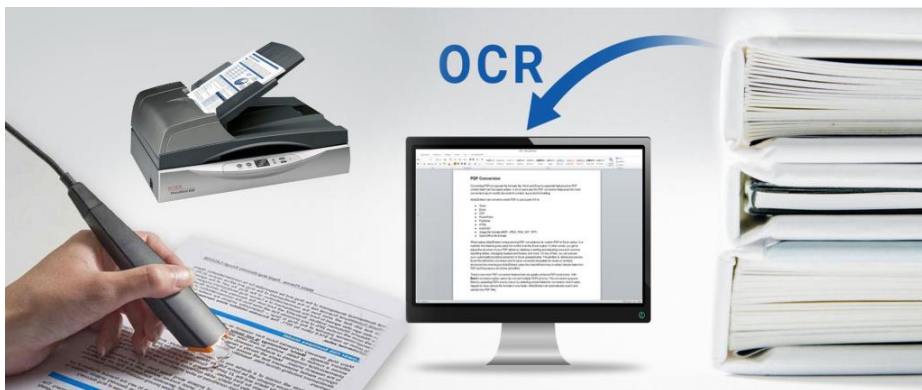
Разбить диапазон яркостей на N интервалов
Каждому интервалу — свой цвет

Применение:

- Визуализация медицинских данных
- Тепловые карты
- Анализ спутниковых снимков

Применение бинаризации

Распознавание текста (OCR)



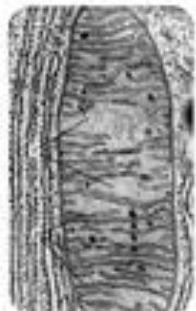
Сегментация объектов



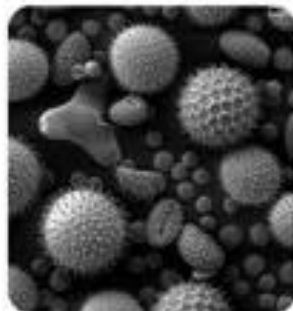
Анализ микроскопических изображений



«срез» хлоропласта
(ТЭМ)

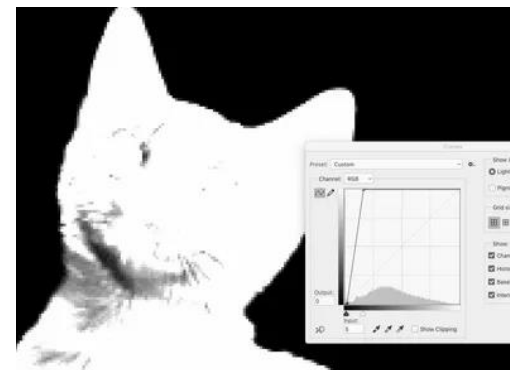


«срез» митохондрии
(ТЭМ)



поверхность пылцы
(СЭМ)

Любая задача, где нужно **отделить объект от фона**



Главная сложность бинаризации



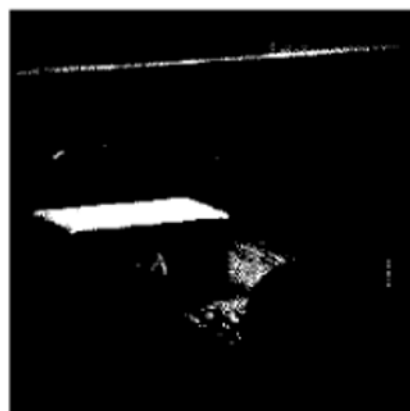
Threshold range 0 to 30



Threshold range 31 to 100



Threshold range 101 to 179



Threshold range 180 to 239

Не применение преобразования, а
Правильный выбор порога T

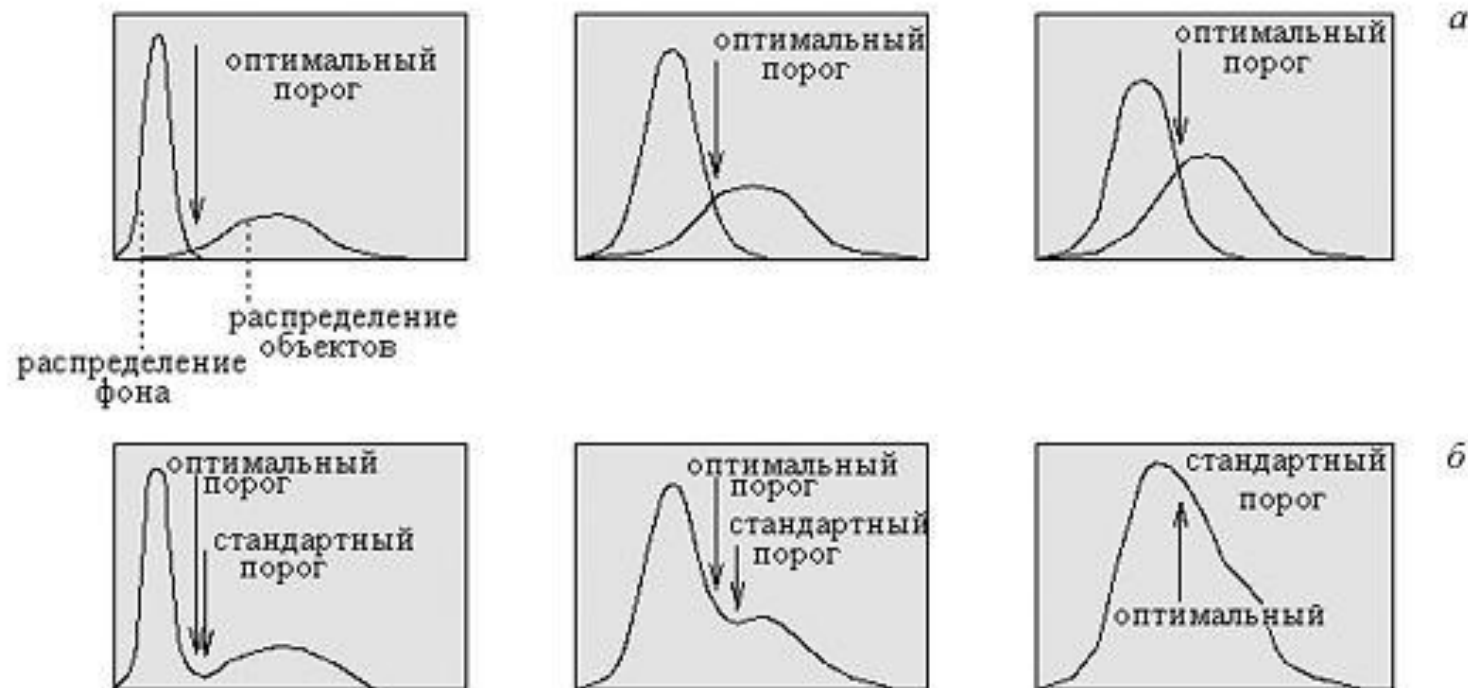
4 варианта:

- Threshold 0-30 → только тёмные объекты
- Threshold 31-100 → средние тона
- Threshold 101-179 → светлые тона
- Threshold 180-239 → очень светлые

Проблема:

- Вручную подбирать долго
- Для каждого фото — свой порог
- **Нужен автоматический метод!**

Выбор порога бинаризации по гистограмме



а - функции распределения объекта и фона

б - соответствующие гистограммы и оптимальный порог

Метод Оцу. Идея

Идея:

Найти автоматически найти порог, который **минимизирует внутриклассовую дисперсию** и **максимизирует межгрупповую дисперсию**

$$sc(t) = 1 - \frac{disp(0, t) + disp(t + 1, \max t)}{disp(0, \max t)}$$

$$T = \arg \max_{t \in 0.. \max t} sc(t)$$

Простыми словами:

- Разбиваем пиксели на 2 класса
- Ищем порог, при котором классы максимально различны



Метод Оцу. Формулы

Минимизация внутригрупповой дисперсии, где q_1 и q_2 — вероятности первого и второго классов соответственно

$$\sigma_w^2(t) = q_1(t)\sigma_1^2(t) + q_2(t)\sigma_2^2(t)$$

$$q_1(t) = \sum_{i=1}^t P(i), \text{ для } i=1..t$$

$$q_2(t) = \sum_{i=t+1}^T P(i)$$

Максимизация межгрупповой дисперсии:

$$\sigma_B^2(t) = q_1(t)[1 - q_1(t)][\mu_1(t) - \mu_2(t)]^2$$

$$\sigma_1^2(t) = \sum_{i=1}^t \frac{[i - \mu_1(t)]^2 P(i)}{q_1(t)}$$

$$\sigma_2^2(t) = \sum_{i=t+1}^T \frac{[i - \mu_2(t)]^2 P(i)}{q_2(t)}$$

$$T = \operatorname{argmax} \sigma_B^2(t)$$

$$\mu_1(t) = \sum_{i=1}^t \frac{iP(i)}{q_1(t)}$$

$$\mu_2(t) = \sum_{i=t+1}^T \frac{iP(i)}{q_2(t)}$$

Метод Оцу. Быстрый алгоритм

1. Вычислить гистограмму
(один проход по изображению)

2. Для $t = 1$ до 255:

- Пересчитать $\sigma^2_B(t)$
- Запомнить максимум

3. Искомый порог = T

Сложность: $O(M)$, где M — количество пикселей

$$\sigma_B^2(t) = q_1(t)[1 - q_1(t)][\mu_1(t) - \mu_2(t)]^2$$

Рекуррентные соотношения

$$q_1(t+1) = q_1(t) + P(t+1)$$

$$q_1 = P(1)$$

$$\mu_1(t+1) = \frac{q_1(t)\mu_1(t) + (t+1)P(t+1)}{q_1(t+1)}$$

$$\mu_1(0) = 0$$

$$\mu_2(t+1) = \frac{\mu - q_1(t+1)\mu_1(t+1)}{1 - q_1(t+1)}$$

Метод Оцу — пример

Исходное



Пиксели < 93



Пиксели > 93



Порог $T = 93$ найден автоматически

Результат:

- Объекты отделены от фона
- Не нужно подбирать ручную

Метод Оцу (Otsu)

Достоинства метода Оцу:

- простота реализации
- быстрое время выполнения ($O(N)$ операций, где N — количество пикселей в изображении)
- нет необходимости подбирать какие-либо коэффициенты, размеры окон для прохода по изображению и проч. (в MatLab это функция `graythresh()` без аргументов)
- дает предсказуемые результаты

Недостатки метода Оцу:

- не применим для сложных изображений с графикой, цветными надписями, различными градиентами
- пороговая бинаризация чувствительна к неравномерной яркости изображения
- решением такой проблемы может быть введение локальных порогов, вместо одного глобального

Метод Оцу — это не только классика

В 2025-2026 годах они используются как препроцессинг для CNN, что повышает точность детекции на 2–3%

Например, в задаче обнаружения малярии Оцу-маска + CNN дали 97.96% точности

Поэтому важно понимать и уметь применять эти методы — они остаются фундаментом современных систем

Виды порогов

1. ГЛОБАЛЬНЫЕ

Один порог для всего изображения

2. ЛОКАЛЬНЫЕ

Свой порог для каждой области

3. АДАПТИВНЫЕ

Порог зависит от окрестности пикселя



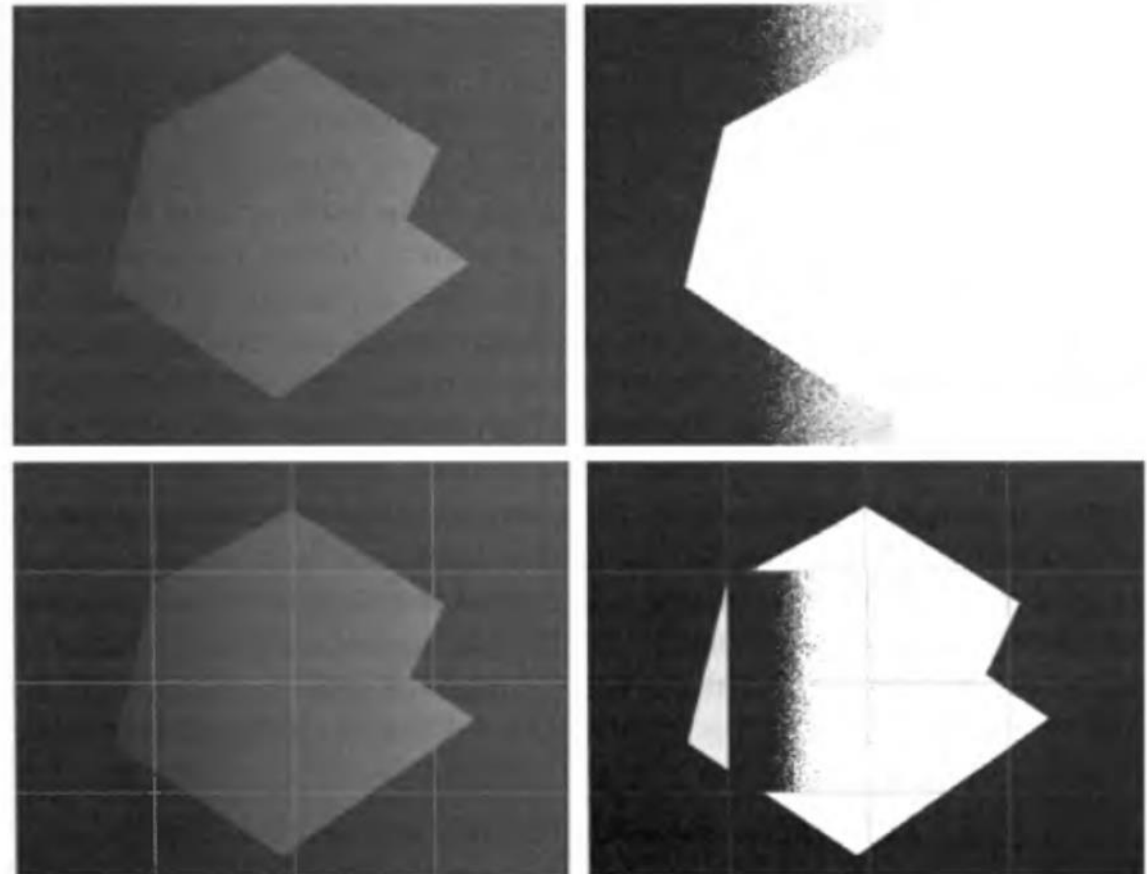
Глобальные vs локальные пороги

Глобальный порог:

- Фон слева тёмный → объект виден
- Фон справа светлый → объект теряется

Локальный порог:

- Для каждой области — свой порог
- Объект виден везде



Методы локальной (адаптивной) бинаризации

- метод Ниблэка
- метод Бернсена
- метод Эйквеля
- метод Саувола
- метод Кристиана

Методы используют окна различного размера, рассчитывают средние значения и средние квадратические отклонения для окрестностей и используют различные эмпирические статистики

Метод Ниблэка

Идея: Порог для каждого пикселя зависит от локального среднего и локального стандартного отклонения в окне.

Формула: $T(x, y) = m(x, y) + k \cdot s(x, y)$, где:

- $m(x, y)$ — среднее значение яркости в окне радиуса r вокруг пикселя
- $s(x, y)$ — стандартное отклонение в этом окне
- k — коэффициент (обычно $k \approx -0.2$)

Как работает:

- Если пиксель темнее, чем локальное среднее плюс поправка на разброс $\rightarrow$ объект (0)
- Если светлее $\rightarrow$ фон (1)

Плюсы:

- Работает при неравномерном освещении
- Прост в реализации

Минусы:

- **Сильно шумит** на однородных участках (там, где $s \approx 0$, порог становится непредсказуемым)
- Требуется подбора k и размера окна r

Применение: анализ текста на неравномерном фоне

Метод Саувола. Идея

Идея: Улучшение метода Ниблэка.

Добавляет **динамический диапазон яркости** R и использует локальное среднее как опорную точку.

Формула: $T(x, y) = m(x, y) \cdot \left[1 + k \cdot \left(\frac{s(x, y)}{R} - 1 \right) \right]$, где:

- $m(x, y)$ — локальное среднее
- $s(x, y)$ — локальное стандартное отклонение
- R — динамический диапазон стандартного отклонения (обычно $R = 128$ для 8-битных изображений)
- k — коэффициент (обычно $k \approx 0.5$)

Как работает:

- Если локальный разброс s мал (однородный фон) $\rightarrow$ порог близок к среднему m
- Если разброс велик (граница объекта) $\rightarrow$ порог сдвигается, чтобы лучше отделить текст
- Параметр R нормирует отклонение, делая метод устойчивым к шуму

Метод Саувола. Плюсы и минусы

Плюсы:

- **Меньше шума**, чем у Ниблэка
- Хорошо работает на документах с пятнами, тенями, неравномерным фоном
- Стал **стандартом де-факто** для бинаризации текста (OCR, исторические рукописи)

Минусы:

- Всё ещё требует подбора k и r
- Медленнее глобального метода Оцу (нужно считать статистику для каждого пикселя)

Применение: OCR, сканирование документов, обработка исторических рукописей, анализ микроскопии.

Сравнение метод Ниблэка и Саувола

Критерий	Ниблэк	Саувола
Формула	$T = m + k \cdot s$	$T = m \cdot [1 + k(s/R - 1)]$
Устойчивость к шуму	Низкая	Высокая
Однородный фон	Шумит	Работает корректно
Параметры	k, r	k, r, R
Применение	Простые задачи	OCR, документы, рукописи

Место в современном pipeline

Оба метода (Ниблэка и Саувола) — **локальные**, то есть порог для каждого пикселя свой

Это делает их незаменимыми для:

- Документов с неравномерным освещением (фото страниц, сканы)
- Исторических рукописей (выцветший текст, пятна, чернила на обороте)
- Медицинских изображений (микроскопия)

В современных системах они часто используются как **препроцессинг перед OCR или CNN**, потому что дают более чистую бинарную маску, чем глобальный Оцу.

Сравнительная таблица: глобальный vs локальный

Критерий	Глобальный порог	Локальный порог
Порог	Один для всего изображения	Индивидуальный для каждой точки/области
Скорость	Высокая	Ниже (особенно при пересчёте для каждой точки)
Точность на сложном фоне	Низкая	Высокая
Чувствительность к освещению	Чувствителен	Нечувствителен
Примеры методов	Метод Оцу	Метод Ниблэка, метод Саувола
Параметры настройки	Ручное или автоматическое (например, метод Оцу)	Требуют подбора (размер окна, константы)

Адаптивная бинаризация

Необходима в случае неравномерной яркости фона/объекта

Для каждого пикселя изображения $I(x, y)$:

- В окрестности пикселя радиуса r высчитывается индивидуальный для данного пикселя порог T
- Если $I(x, y) > T + C$, результат 1, иначе 0

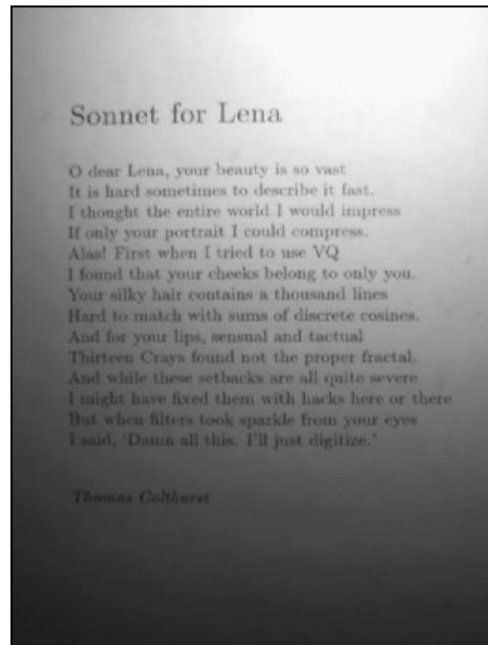
Варианты выбора T :

- $T = \text{mean}$
- $T = \text{median}$
- $T = (\text{min} + \text{max}) / 2$

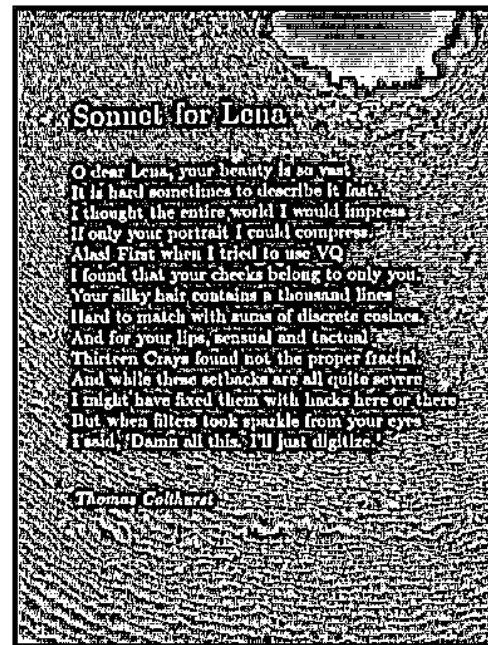
Параметры:

- r — радиус окна
- C — константа

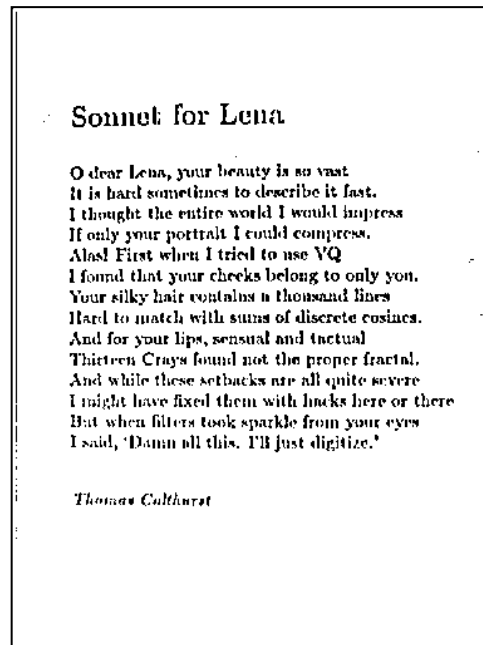
Пример адаптивной бинаризации



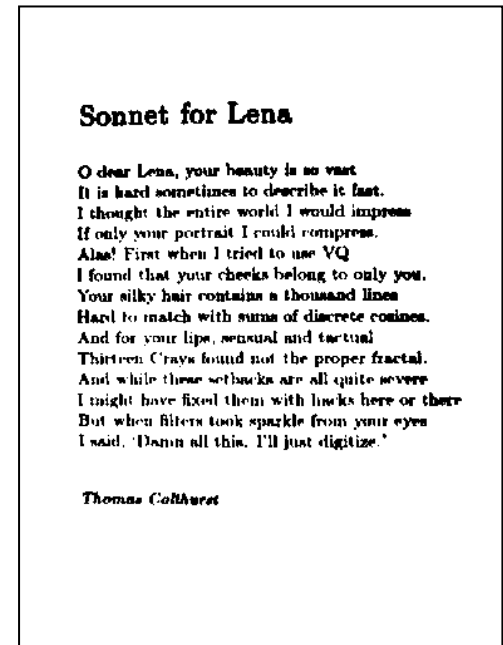
Исходное



$r=7, C=0$



$r=7, C=7$



$r=75, C=10$

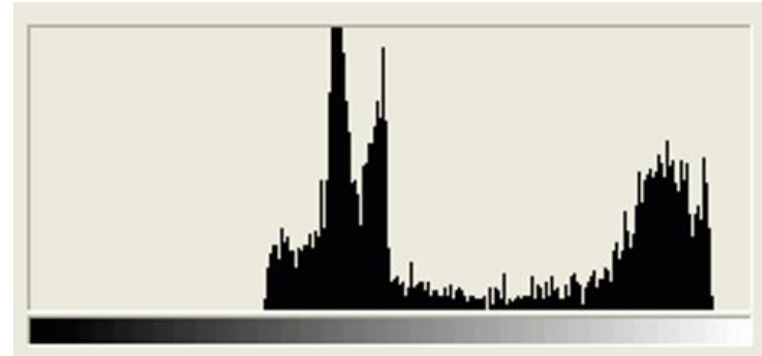
Иерархическая бинаризация

Идея:

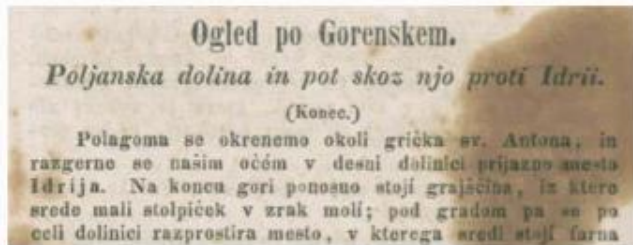
- Рекурсивно разделяем изображение
- На каждом уровне — свой порог
- Получаем иерархию объектов

Применение:

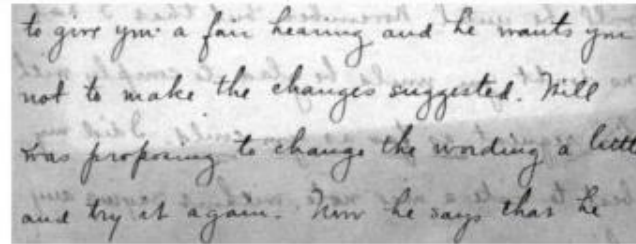
- Сегментация сложных сцен
- Медицинские изображения



Применение бинаризации (примеры)



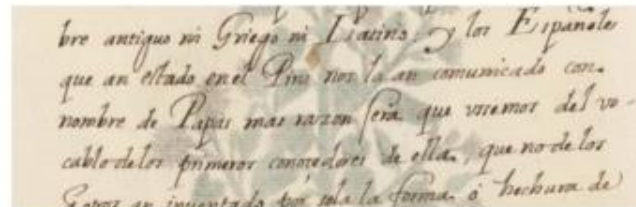
(a) page stain



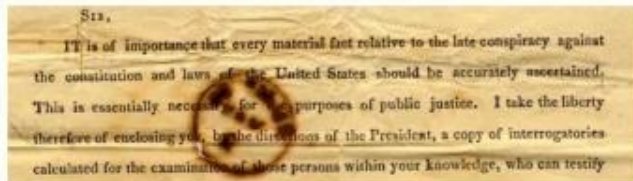
(b) ink bleed through



(c) text stroke fading



(d) background texture



(e) library seal



(f) text stroke changes

ПРИМЕНЕНИЕ: ИСТОРИЧЕСКИЕ ДОКУМЕНТЫ

Проблемы:

- Пятна на бумаге
- Чернила на обороте
- Выцветший текст
- Текстура фона
- Печати
- Изменения штрихов

Решение:

- Адаптивная бинаризация
- Локальные пороги

Сравнение результатов методов бинаризации

Бинаризация :

а), б), в) – метод Оцу;

г), д), е) – алгоритм Брэдли-Рота;

ж), з), и) –среднее арифметическое цветов пикселя;

к), л), м) –иерархический метод Оцу



Сравнение методов бинаризации

Вывод:

- Нет универсального метода
- Выбор зависит от задачи
- Оцу — для простых случаев
- Адаптивные — для сложных



Глобальный порог
Метод Оцу



Локальный порог
Метод Ниблэка

Выбор метода

Выбор метода зависит от задачи

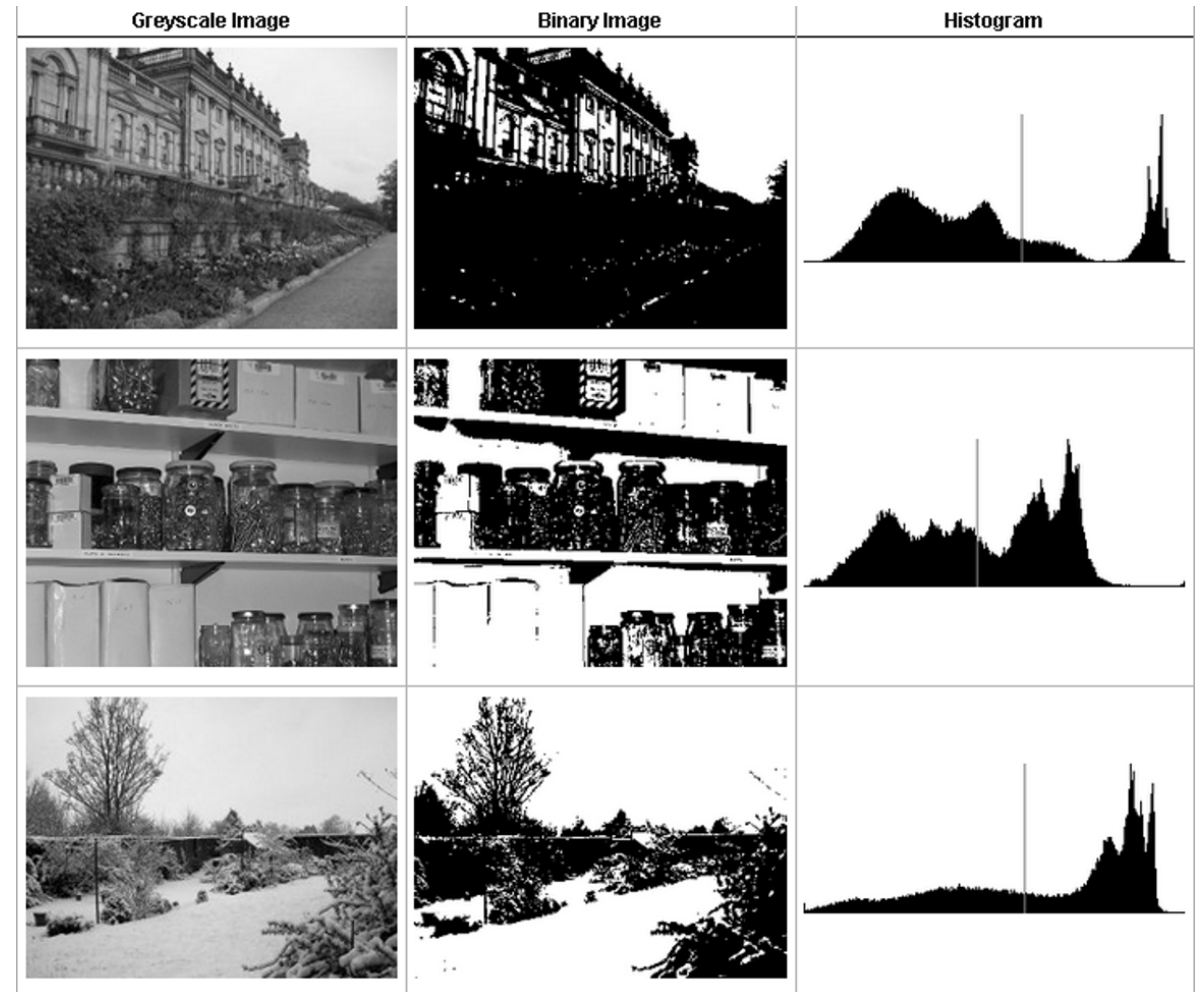
Но в современных системах часто используют **гибрид**:

- классический порог для быстрой грубой сегментации,
- нейросеть — для точной

Это сочетает скорость $O(N)$ и точность глубокого обучения

Резюме

1. Бинаризация — отделение объекта от фона
 $f(v) = 1$, если $v > t$
2. Виды: верхняя, нижняя, по диапазону
3. Метод Оцу — автоматический выбор порога
 - Минимизация внутриклассовой дисперсии
 - Быстрый ($O(M)$)
 - Не работает для сложных изображений
4. Виды порогов:
 - Глобальные
 - Локальные
 - Адаптивные
5. Применение: OCR, медицина, документы



Мост к современным подходам

Традиционная бинаризация (Оцу, Ниблэк, Саувола) — это не просто старый метод
В современных системах компьютерного зрения она занимает новое место

Понимание Оцу и адаптивных порогов — это не только теория, но и практический инструмент для
повышения качества ML-систем

Раньше	Сейчас
Финальный результат сегментации	Предварительный этап для глубокого обучения
Ручной подбор порога	Автоматический выбор + нейросеть
Работа с одним изображением	Работа с потоком данных в реальном времени

Бинаризация как препроцессинг для CNN

Проблема: Свёрточные нейросети (CNN) мощны, но чувствительны к шуму и фону. Если фон сложный, модель может отвлекаться на нерелевантные детали.

Решение: Использовать быстрые классические методы (Оцу, адаптивный порог) для **грубого выделения области интереса (ROI)** перед подачей в нейросеть.

Пример из исследования 2025 года (малярия):

- Вход: изображение мазка крови
- Шаг 1: бинаризация методом Оцу → маска паразитов
- Шаг 2: маска + исходное RGB-изображение → CNN
- Результат: точность выросла с **95% до 97.96%**
- Метрики качества сегментации: Dice = 0.848, IoU = 0.738

Вывод: Даже простая предварительная бинаризация может значительно улучшить работу глубокой модели.

Когда классика всё ещё лучше нейросетей

Критерий	Классические методы	Глубокое обучение
Скорость	$O(N)$, реальное время	Требует GPU
Данные для обучения	Не нужны	Нужны тысячи размеченных примеров
Интерпретируемость	Полная (формула, порог)	Чёрный ящик
Сложные сцены	Слабо	Сильно
Освещение	Чувствительны (глобальные)	Устойчивы после обучения

Практический вывод:

Оцу — для простых, равномерно освещённых задач и как быстрый препроцессинг

Саувола / Ниблэк — для документов с неравномерным фоном (OCR, исторические рукописи)

Нейросети — для сложных сцен, где классика не справляется

Гибридный подход — будущее

Современный тренд: не «или-или», а **симбиоз**.

1. Классика → Нейросеть:

- Оцу генерирует маску → CNN уточняет границы
- Адаптивный порог выделяет текст → Трансформер распознаёт

2. Нейросеть → Классика:

- CNN предсказывает карту вероятностей → Оцу выбирает оптимальный порог бинаризации

3. Примеры:

- Медицина: Оцу + U-Net для сегментации опухолей
- Промышленность: адаптивный порог + YOLO для контроля качества
- Документы: Саувола + CRNN для распознавания рукописей

Ключевая мысль: Классические методы бинаризации — это **быстрый и надёжный инструмент**, который в современном pipeline работает **вместе** с глубоким обучением, а не вместо него.