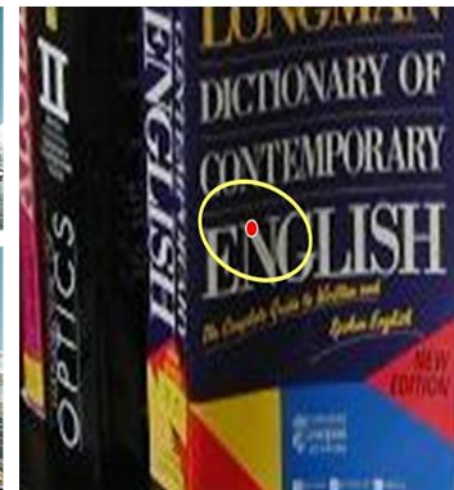
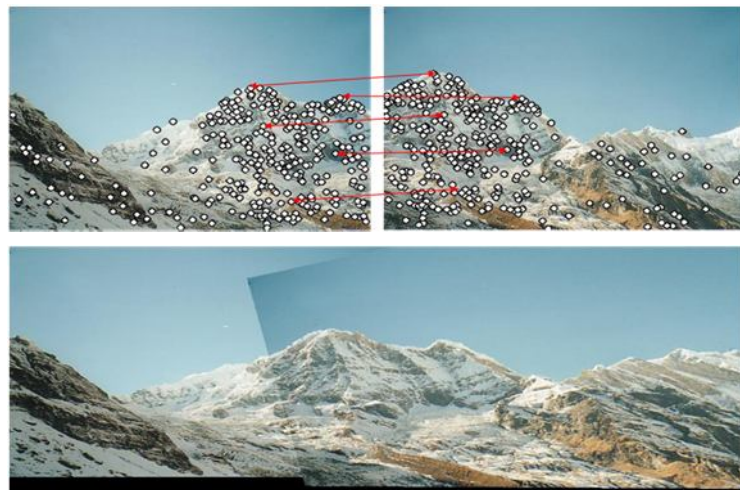
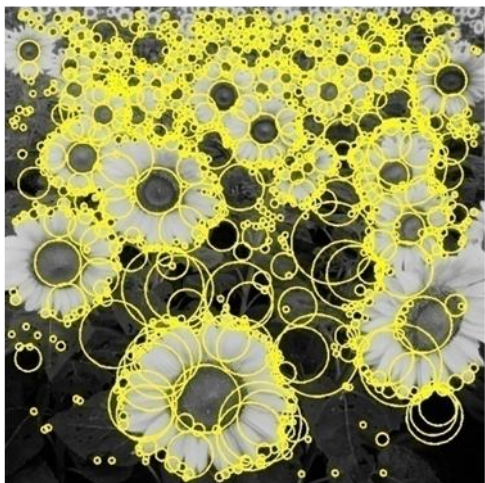
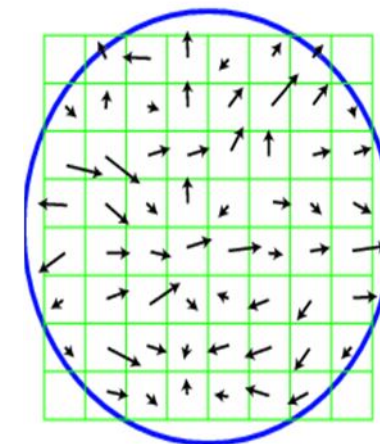
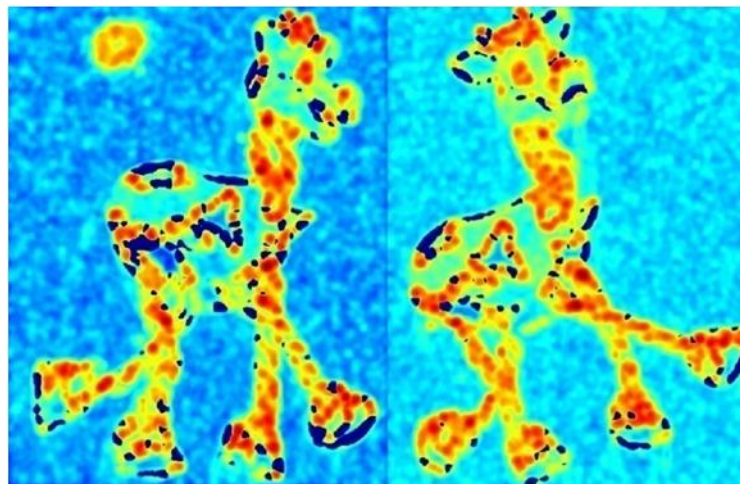




Локальные особенности



Many slides adopted from Svetlana Lazebnik, Steve Seitz and Alexey Efros



Общая информация

Microsoft
Research

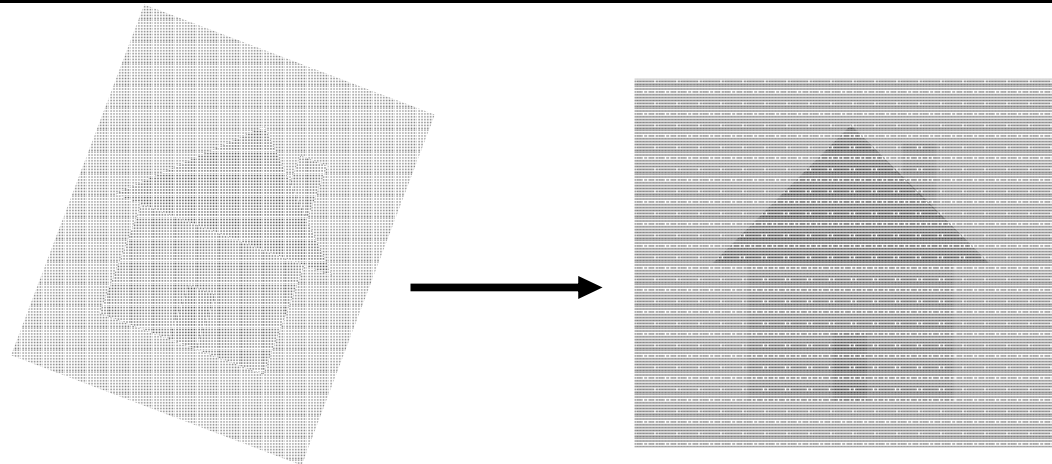
Этот курс
подготовлен и
читается при
поддержке Microsoft
Research

Microsoft
Research

- Страница курса
<http://courses.graphicon.ru/main/vision>



Выравнивание изображений



Есть два изображения одного и того же объекта.
Как нам совместить изображения автоматически?

Вариант:

- Прямое(попиксельное) согласование
- Поиск такого совмещения, при котором большинство пикселей совпадают
- Поиск шаблонов



Прямое согласование

Простейший подход – «грубой силы» (brute force)

- Выбрать модель преобразования (сдвиг, поворот) и набор параметров, описывающих преобразование
- Выбрать функцию сопоставления изображений
 - SSD, Нормализованная корреляция, сопоставление краев, и т.д.
- Перебор всевозможных значений параметров в разумных пределах:

Пример – параллельный перенос:

```
for tx=x0:step:x1,  
  for ty=y0:step:y1,  
    compare image1(x,y) to image2(x+tx,y+ty)  
  end;  
end;
```

Необходимо заранее выбрать x_0 , x_1 и $step$

- Что произойдет, если $step$ слишком велик?



Градиентный спуск

- Используем градиентный спуск для оптимизации параметров преобразования
- В качестве целевой функции используется мера согласования изображений
- Требуется точное начальное приближение
 - Ошибка менее 2х пикселей
- Можно улучшить с помощью многомасштабного подхода



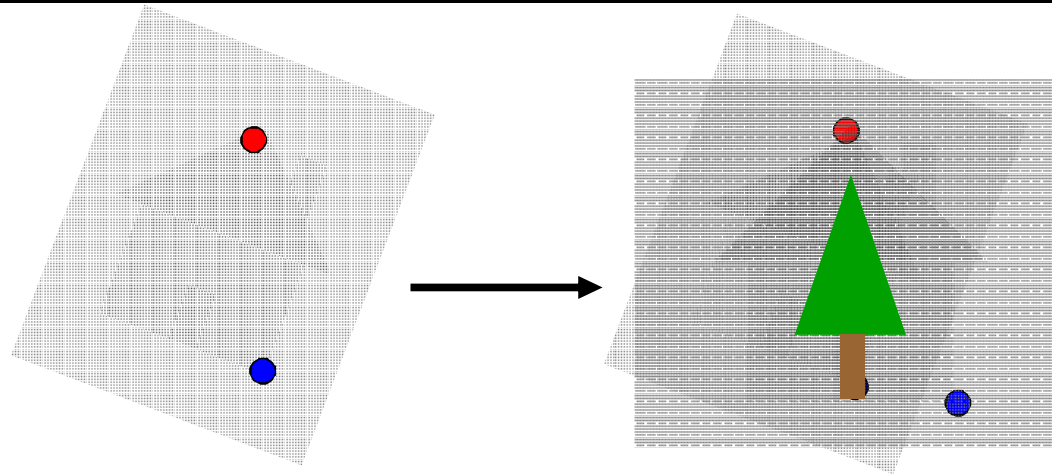
Многомасштабное сопоставление

- Идея метода
 - Строим пирамиду изображений
 - Ищем преобразование на самом низком уровне
 - Можно использовать полный перебор
 - Используем как начальное приближение для уточнения на следующих уровнях
 - Градиентный спуск

- Проблемы
 - Можем сойтись в локальный минимум



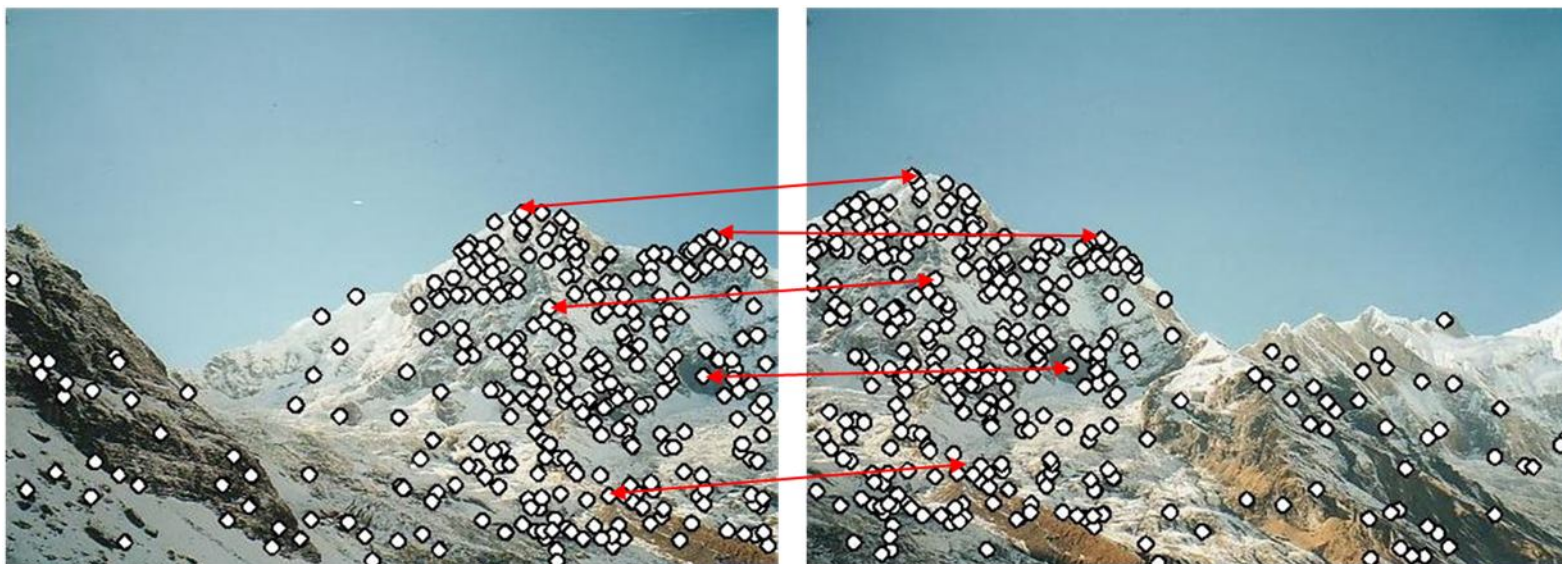
Локальные особенности



- Что делать, если часть объекта загорожена другим?
 - Елочка выросла....
- Найти хорошо различимые точки («характеристические точки», «особенности», «локальный особые точки»)
- Найти такое преобразование, которое совмещает найденные точки
- Можно верифицировать с помощью попиксельного сравнения



Мотивация



Создание мозаик



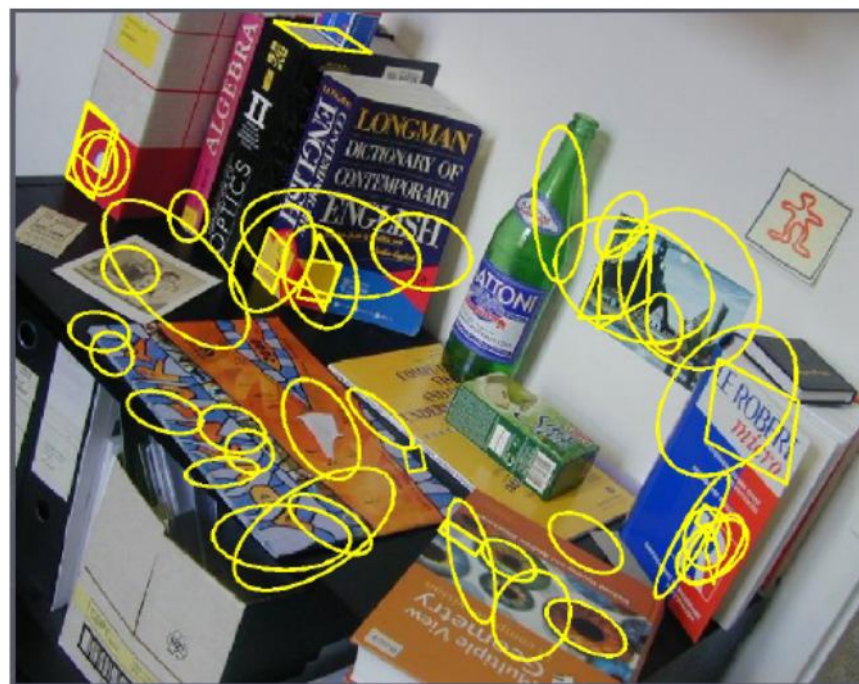
Мотивация



Создание мозаик



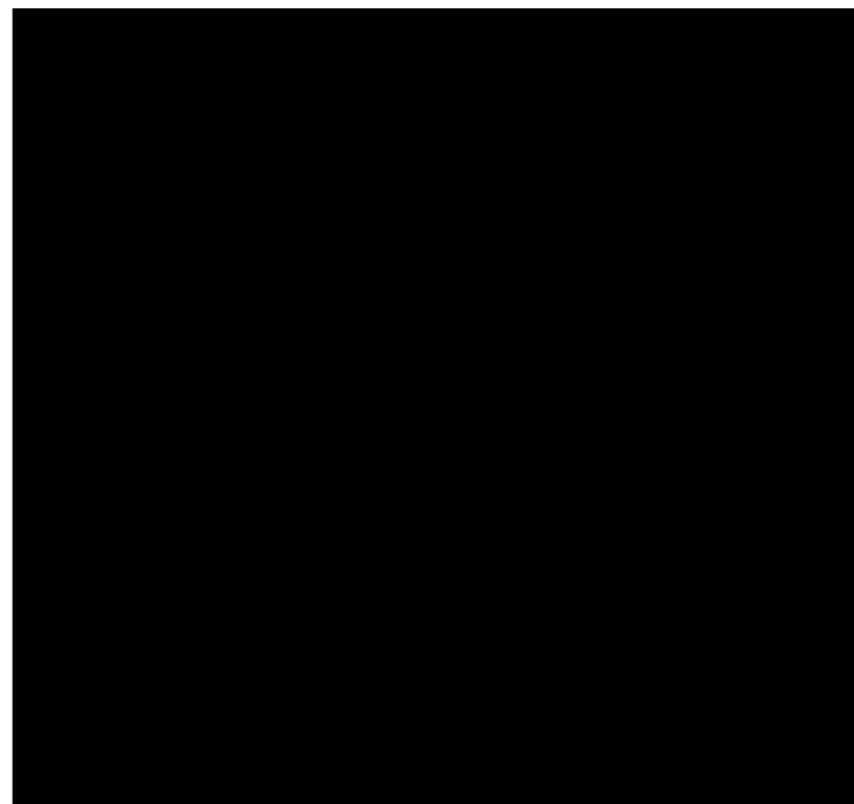
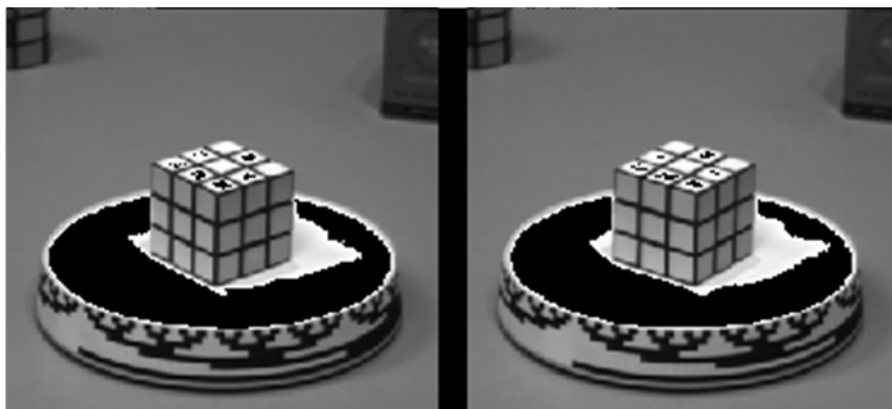
Мотивация



Стереотождество



Мотивация



Оценка движения, слежение за объектом



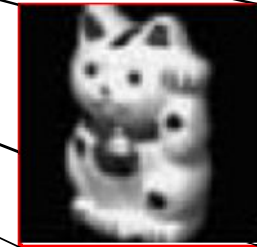
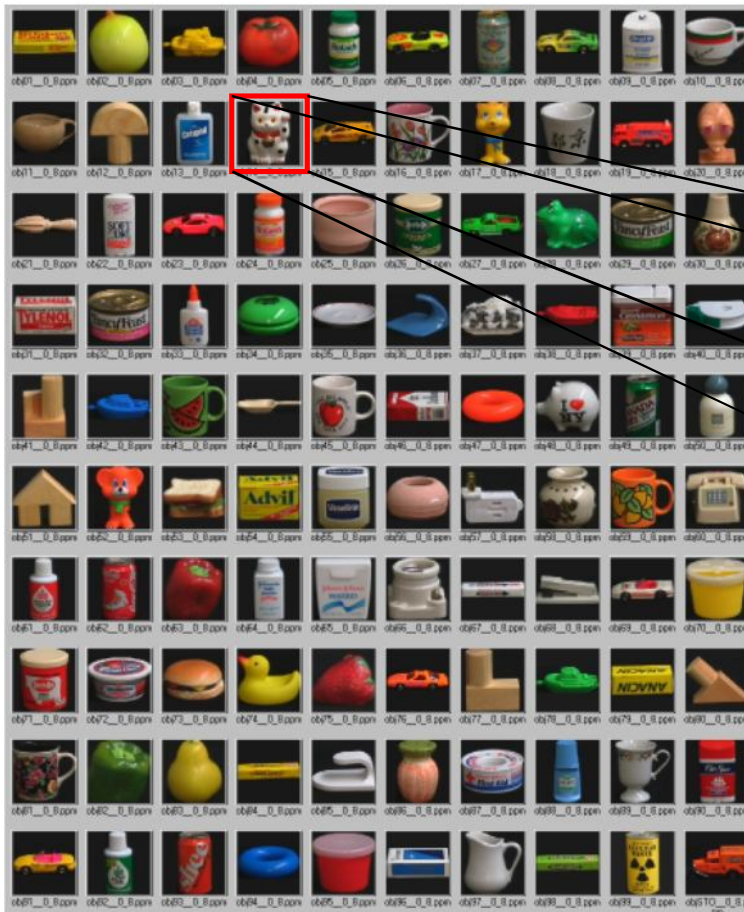
Мотивация



Классификация и поиск объектов



Мотивация



Поиск в базе изображений



Локальные особенности



Пример особой
точки

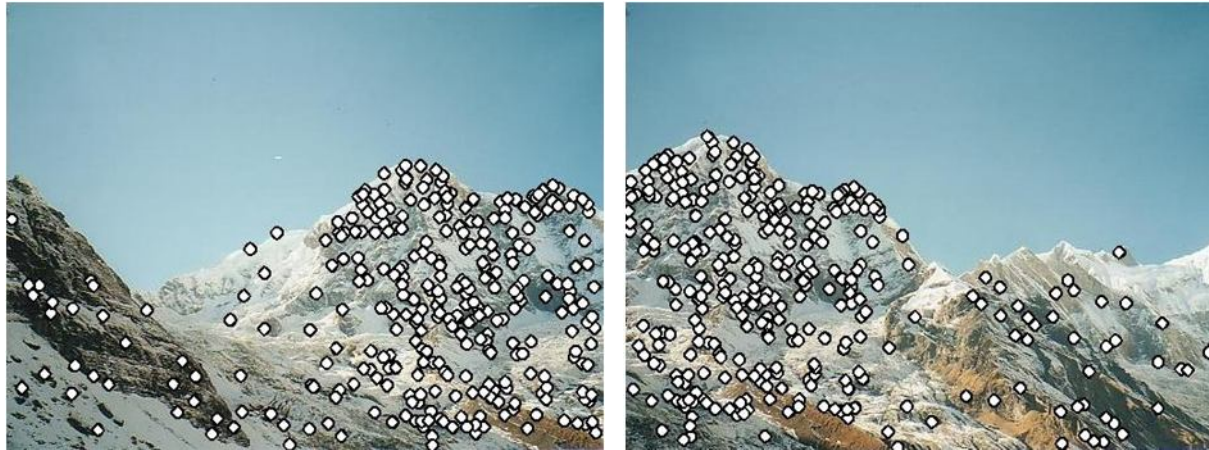


Пример точки, не
являющейся особой

- Локальная особая точка изображения (local image feature)
 - это точка с характерной (особой) окрестностью, т.е. отличающаяся от всех точек в некоторой окрестности



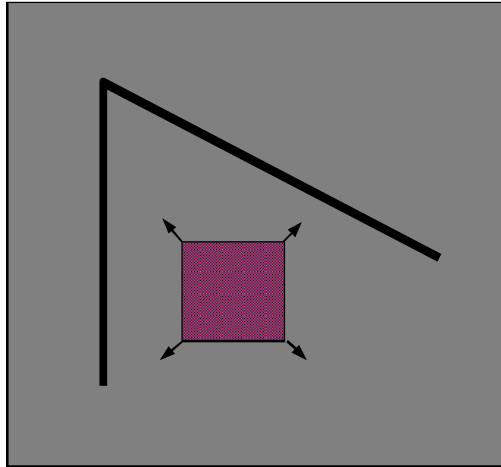
Требования к особенностям



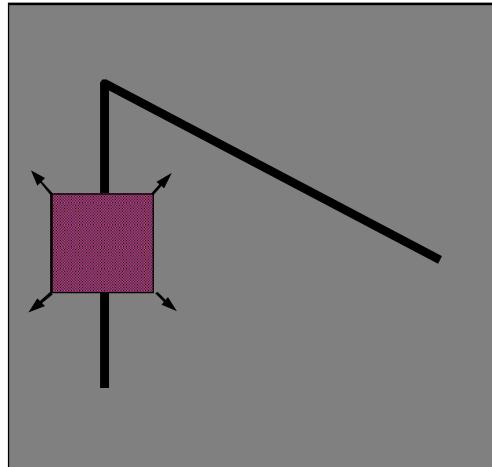
- **Повторяемость (Repeatability)**
 - Особенность (feature) находится в том же месте сцены не смотря на изменения точки обзора и освещения
- **Значимость (Saliency)**
 - Каждая особенность имеет уникальное (distinctive) описание
- **Компактность и эффективность**
 - Количество особенностей существенно меньше числа пикселей изображения
- **Локальность (Locality)**
 - Особенность занимает маленькую область изображения, поэтому работа с ней нечувствительна к перекрытиям



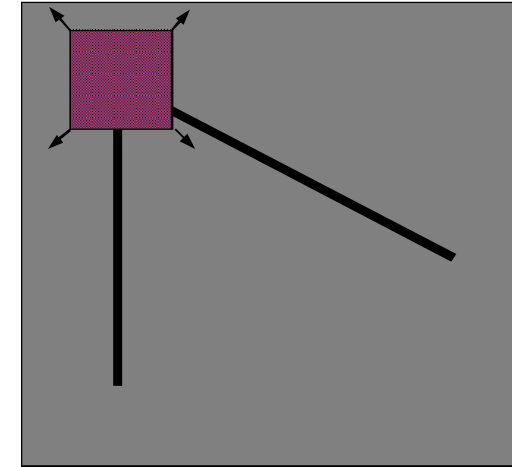
Углы



монотонный регион:
в любом направлении
изменений нет



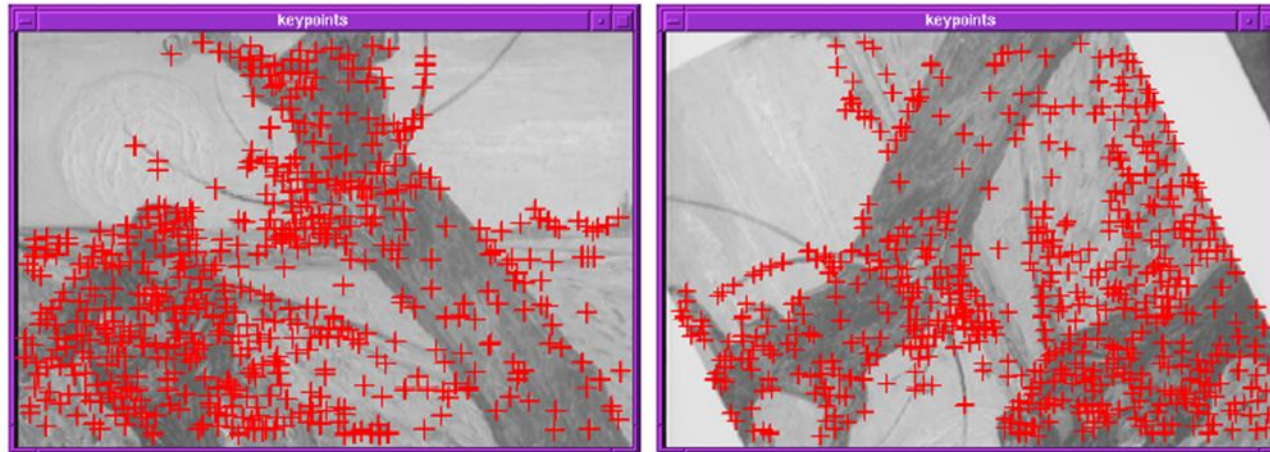
«*край*»:
вдоль края
изменений нет



«*уголок*»:
изменения при
перемещении
в любую сторону



Детектор Харриса



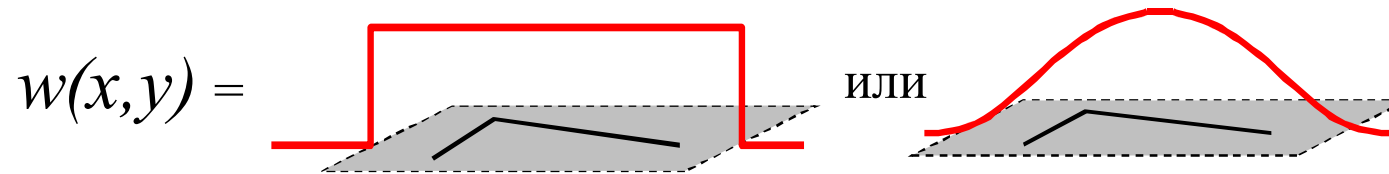
- Главное свойство угла
 - в области вокруг угла у градиента изображения два доминирующих направления
- Уголки хорошо повторимы и различимы
- Наиболее популярный детектор локальных особенностей точек – детектор Харриса (Harris)

C.Harris and M.Stephens. ["A Combined Corner and Edge Detector."](#)
Proceedings of the 4th Alvey Vision Conference: pages 147—151, 1988



Детектор Харриса

Изменение яркости $E(u, v) = \sum_{x, y} w(x, y) [I(x+u, y+v) - I(x, y)]^2$



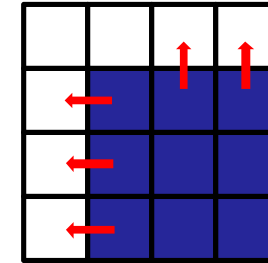
Для малых u, v $E(u, v) \cong [u, v] M \begin{bmatrix} u \\ v \end{bmatrix}$

$$M = \sum_{x, y} w(x, y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \quad \lambda_1, \lambda_2 \text{ собственные значения } M$$



Интерпретация матрицы моментов

Рассмотрим случай, когда градиенты выровнены по осям (вертикальные или горизонтальные)



$$M = \sum \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}$$

Если одно из λ близко к 0, тогда это не уголок, и нужно искать другие точки



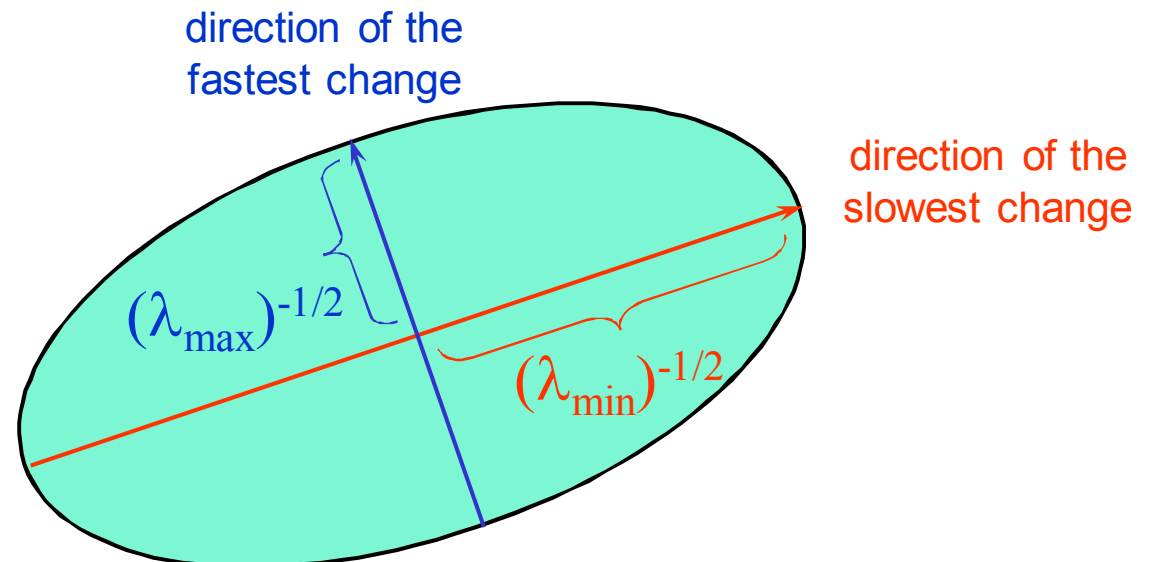
Общий случай

M – симметричная, поэтому $M = R^{-1} \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} R$

Матрицу M можно представить как эллипс, у которого длины осей определены собственными значениями, а ориентация определена матрицей R

Уравнение эллипса:

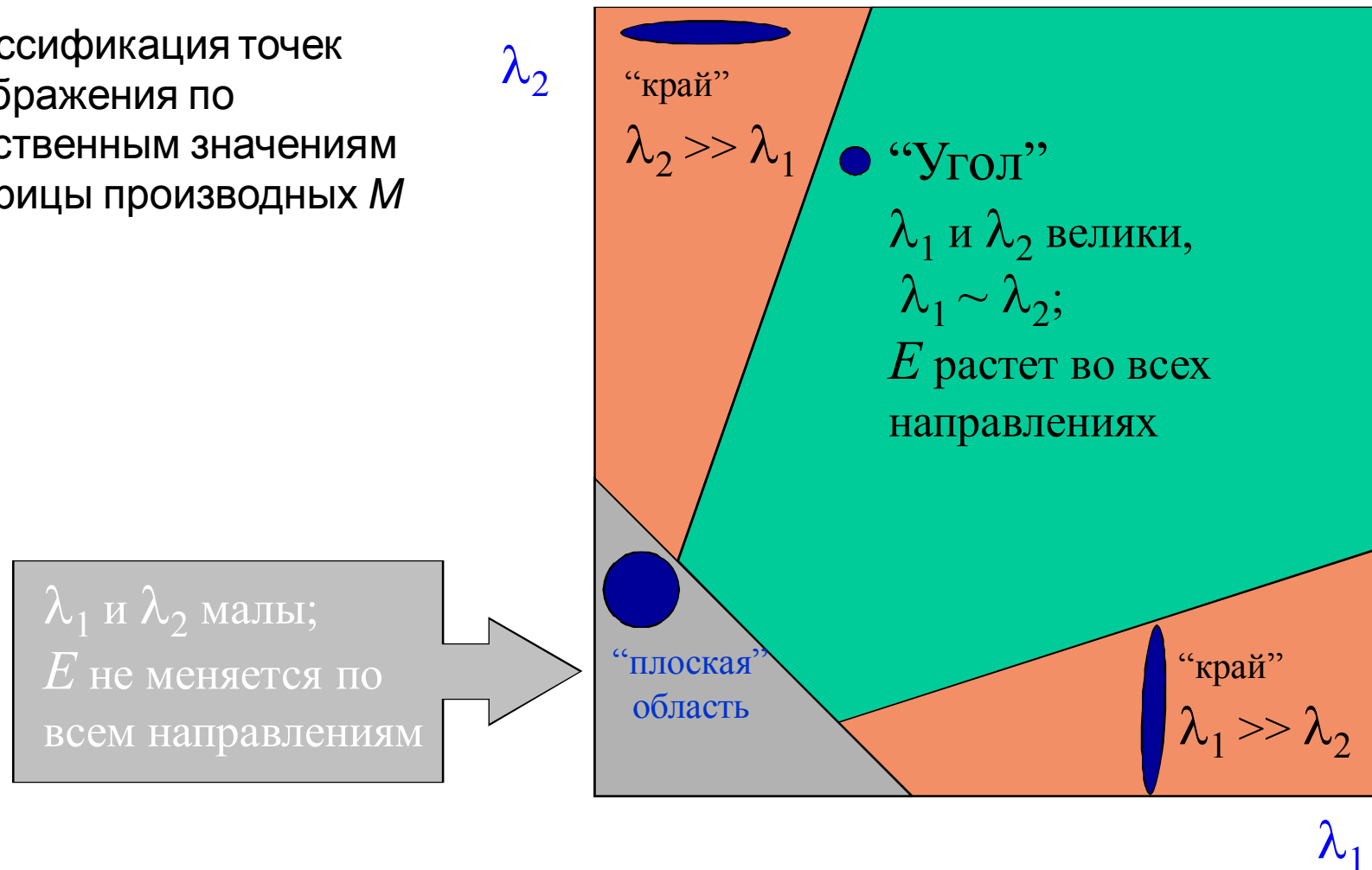
$$[u \ v] M \begin{bmatrix} u \\ v \end{bmatrix} = \text{const}$$





Детекторы углов

Классификация точек изображения по собственным значениям матрицы производных M





Детекторы углов

- Мера отклика угла по Харрису:

$$R = \det M - k (\text{trace } M)^2$$

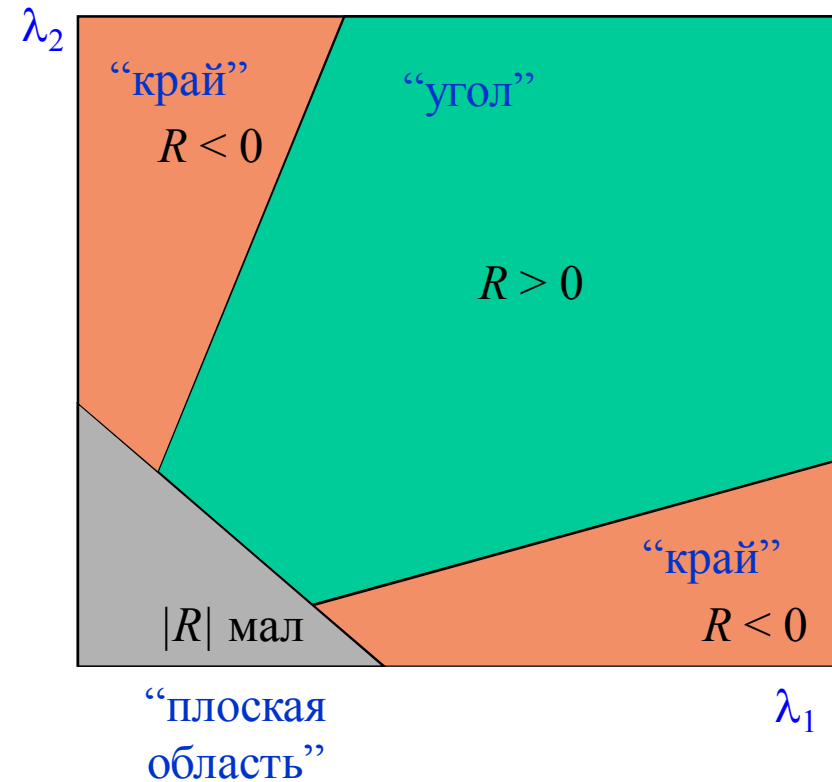
$$\det M = \lambda_1 \lambda_2$$

$$\text{trace } M = \lambda_1 + \lambda_2$$

$$(k = 0.04-0.06)$$

- Мера по Фёрстнеру (Forstner):

$$R = \det M / \text{trace } M$$



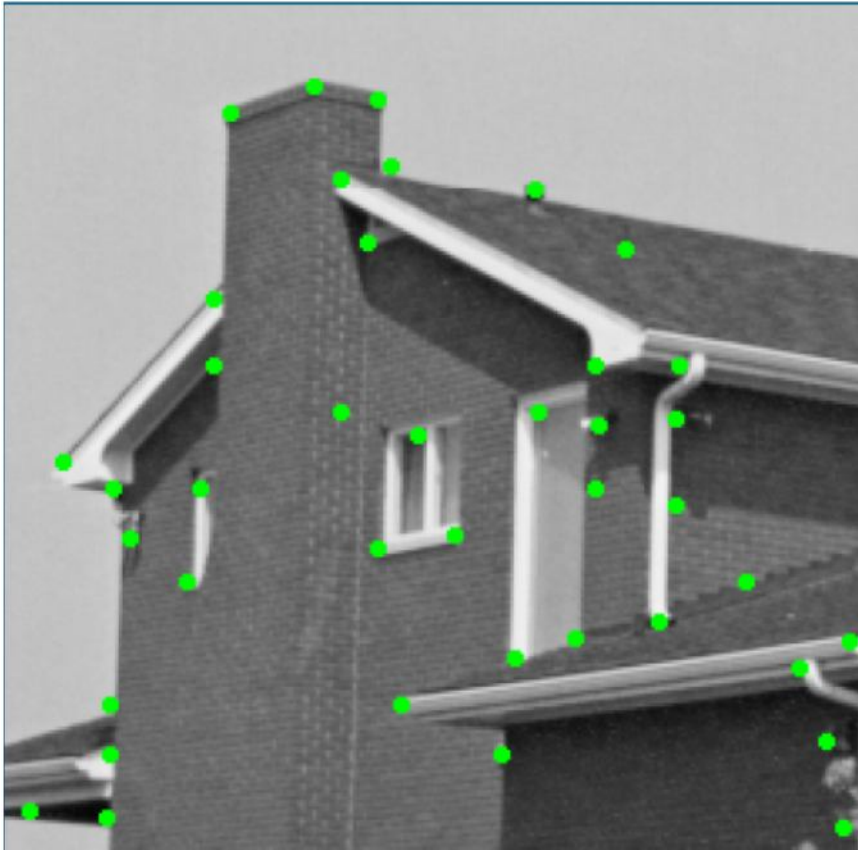


Алгоритм детектора Харриса

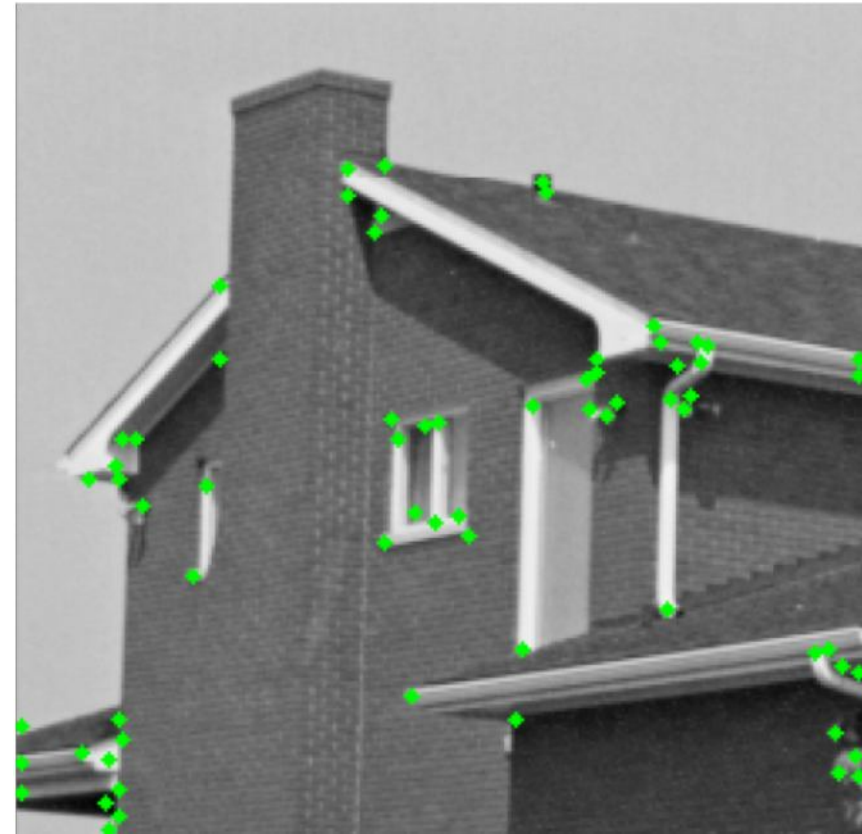
1. Вычислить градиент изображения в каждом пикселе
 - С использованием гауссова сглаживания
2. Вычислить матрицу вторых моментов M по окну вокруг каждого пикселя
3. Вычислить отклик угла R
4. Отрезать по порогу R
5. Найти локальные максимумы функции отклика (nonmaximum suppression) по окрестности заданного радиуса
6. Выбор N самых сильных локальных максимумов



Результат работы детектора



детектор Фёрстнера

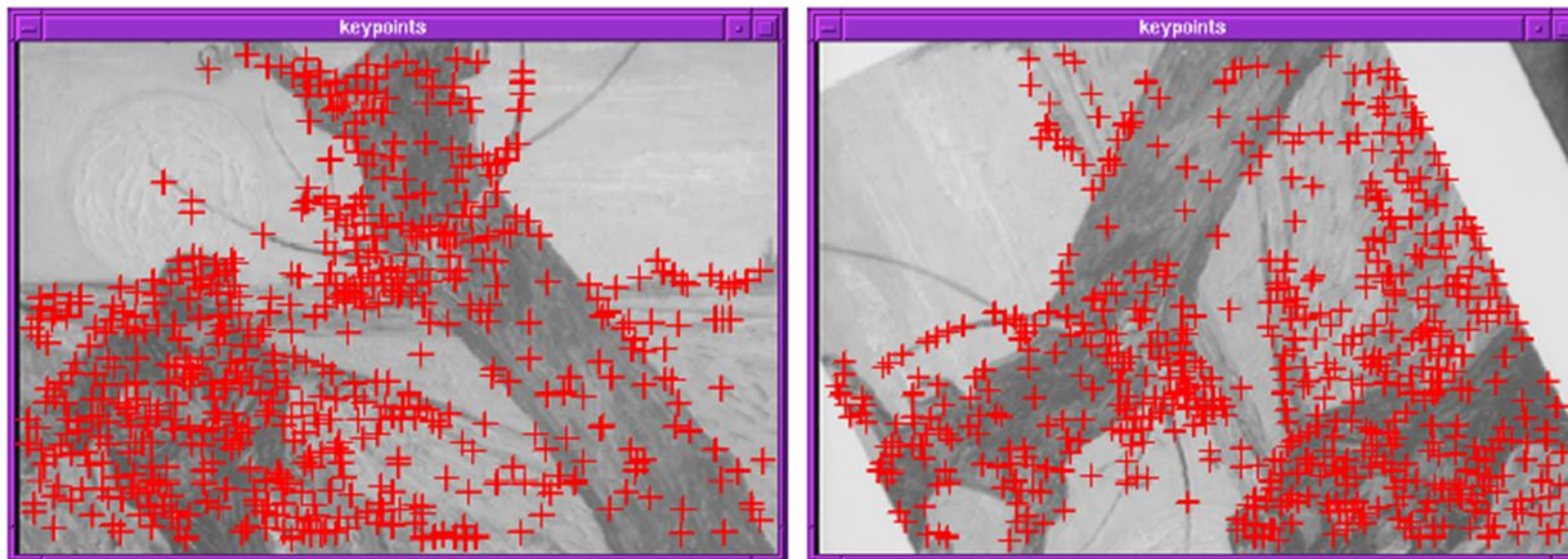


детектор Харриса



Инвариантность

- Хорошо бы чтобы особенности находились всегда, не смотря на геометрические или фотометрические искажения изображения.
- На двух преобразованных изображениях должно находиться одно и то же множество уголков

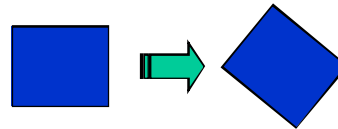




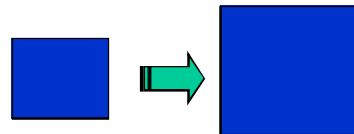
Модели преобразования

Геометрические

- **Поворот**

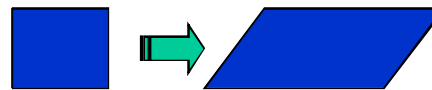


- **Масштаб**



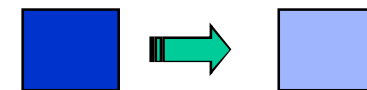
- **Аффинное**

годно для ортографической камеры, локально-плоского объекта



Фотометрические

- **Аффинное изменение яркости ($I \rightarrow a I + b$)**



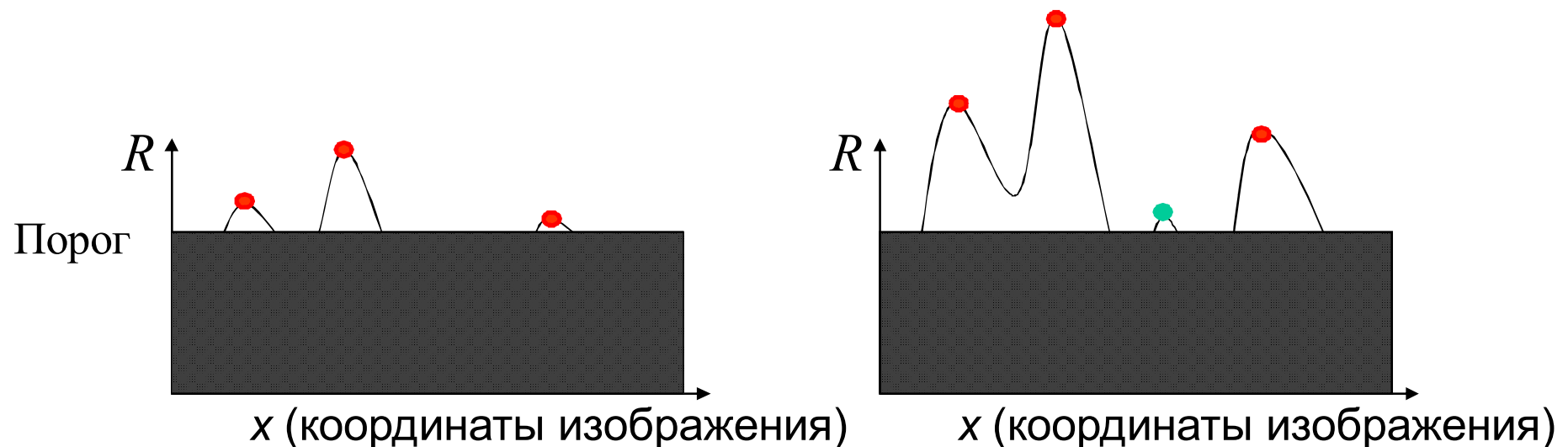


Детекторы Харриса

- Частичная инвариантность к изменению освещенности

- ✓ Используются только производные
=> инвариантность к сдвигу $I \rightarrow I + b$

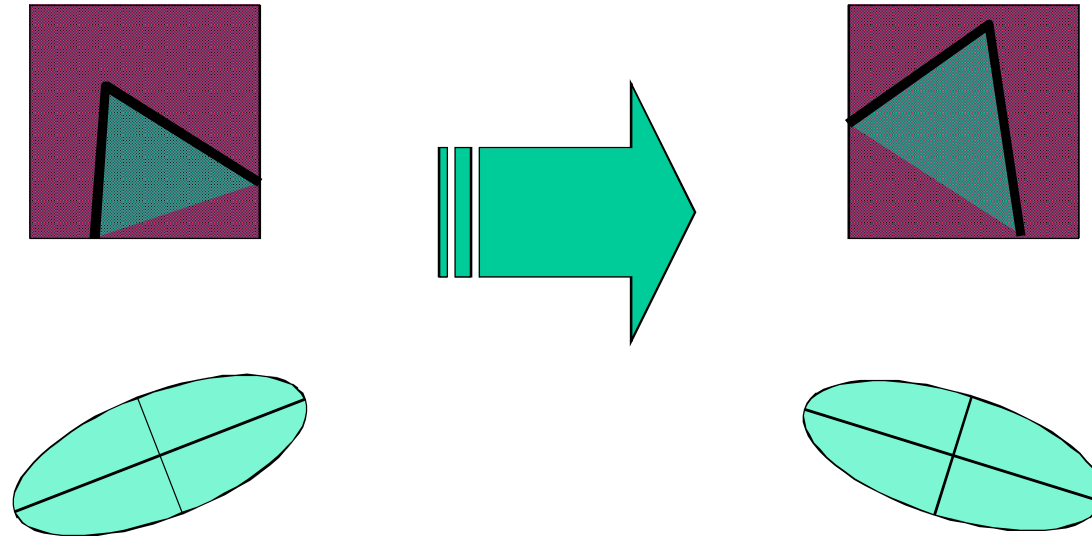
- ✓ Масштабирование: $I \rightarrow a I$





Детектор Харриса: Инвариантность

Поворот



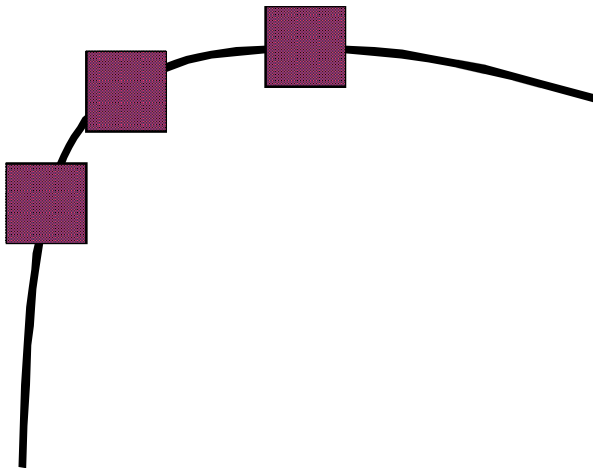
Эллипс вращается, но его форма
(собственные значения) остаются

*Отклик угла R инвариантен к повороту
изображения*

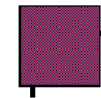
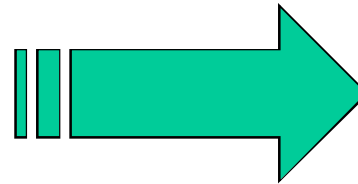


Масштабирование?

- Угол или нет? - Зависит от масштаба изображения!



Все эти точки будут помечены как *края*

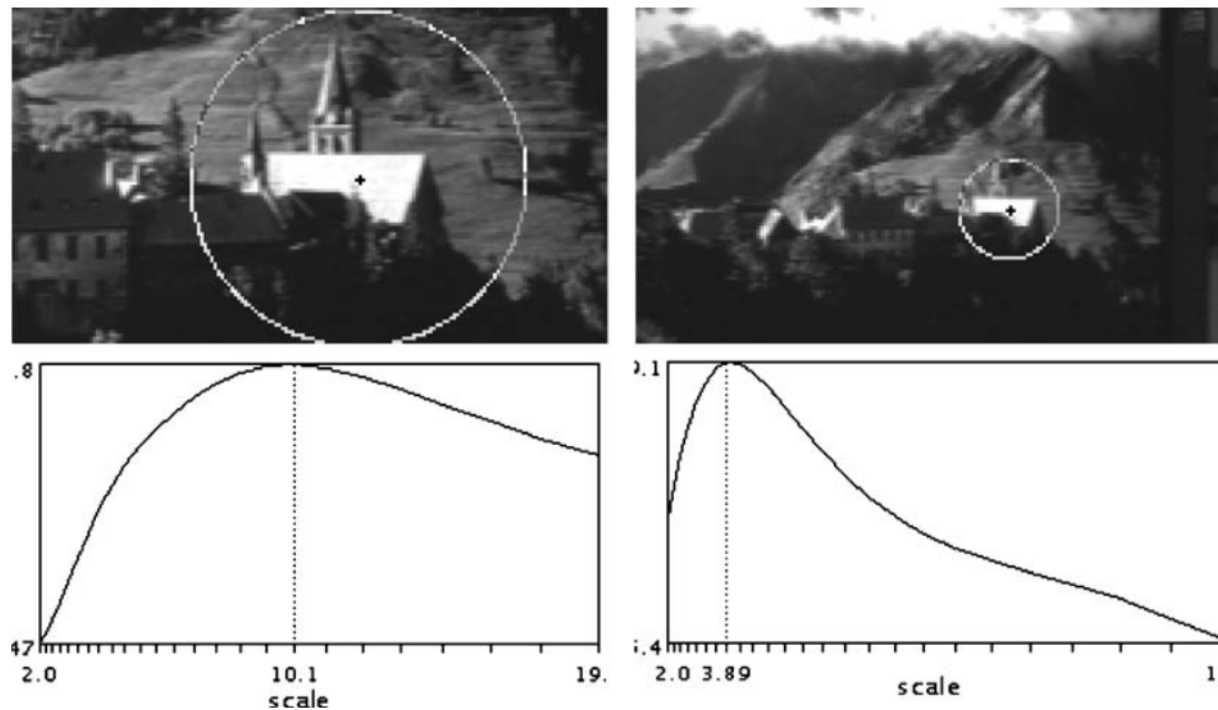


Угол !



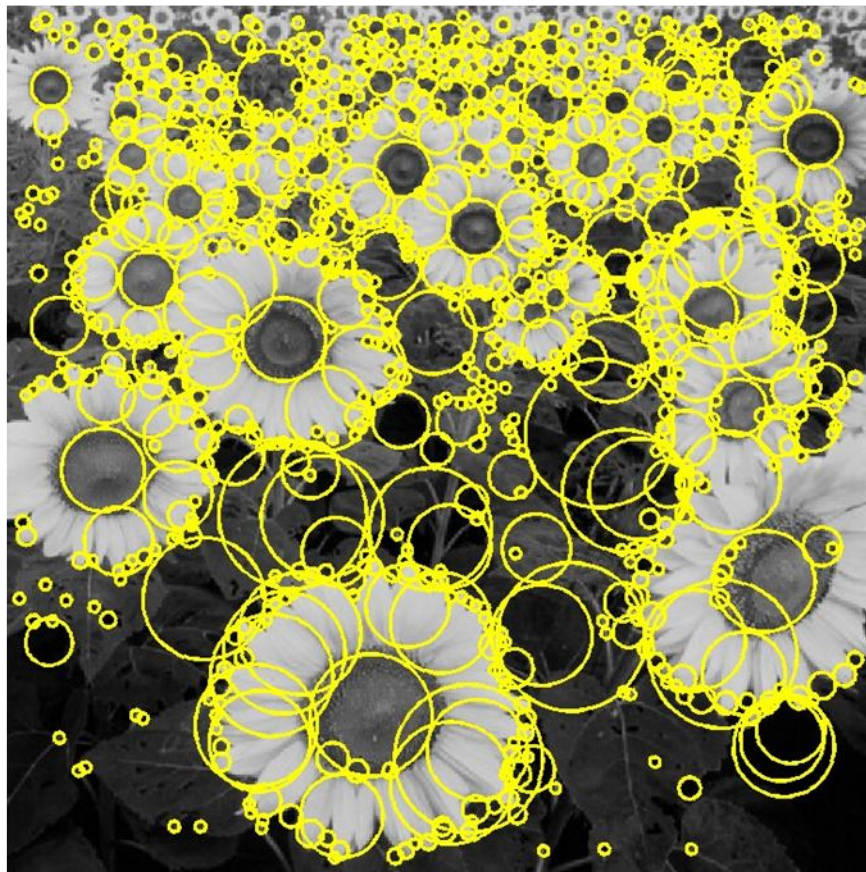
Инвариантность к масштабу

- Цель: независимо находить область в масштабированных версиях одного и того же изображения
- Требуется метод выбора размера характеристической области





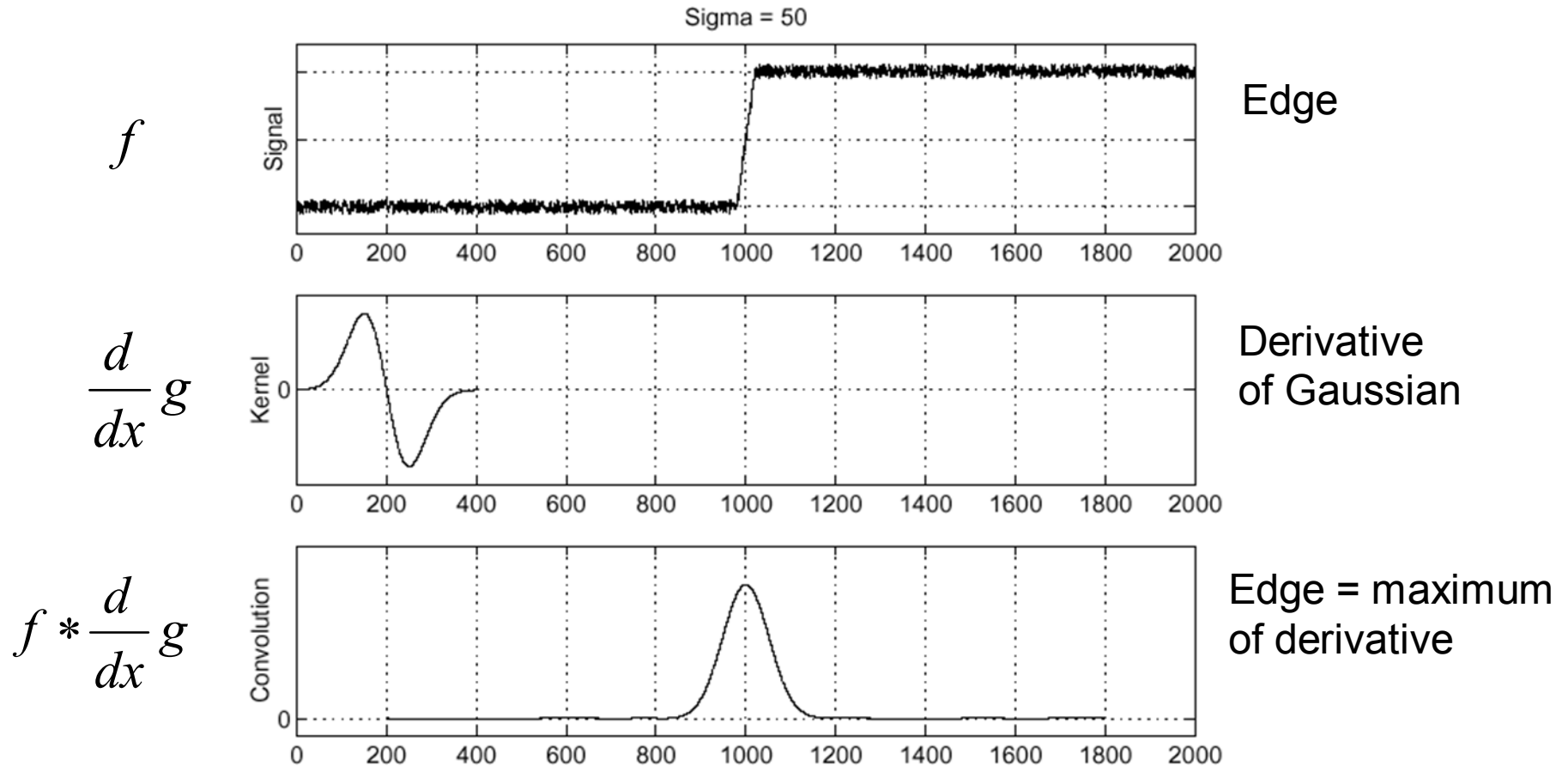
Блобы



«Капля», «Vlob» - вначале для особенностей такого типа была разработана теория выбора характерного размера

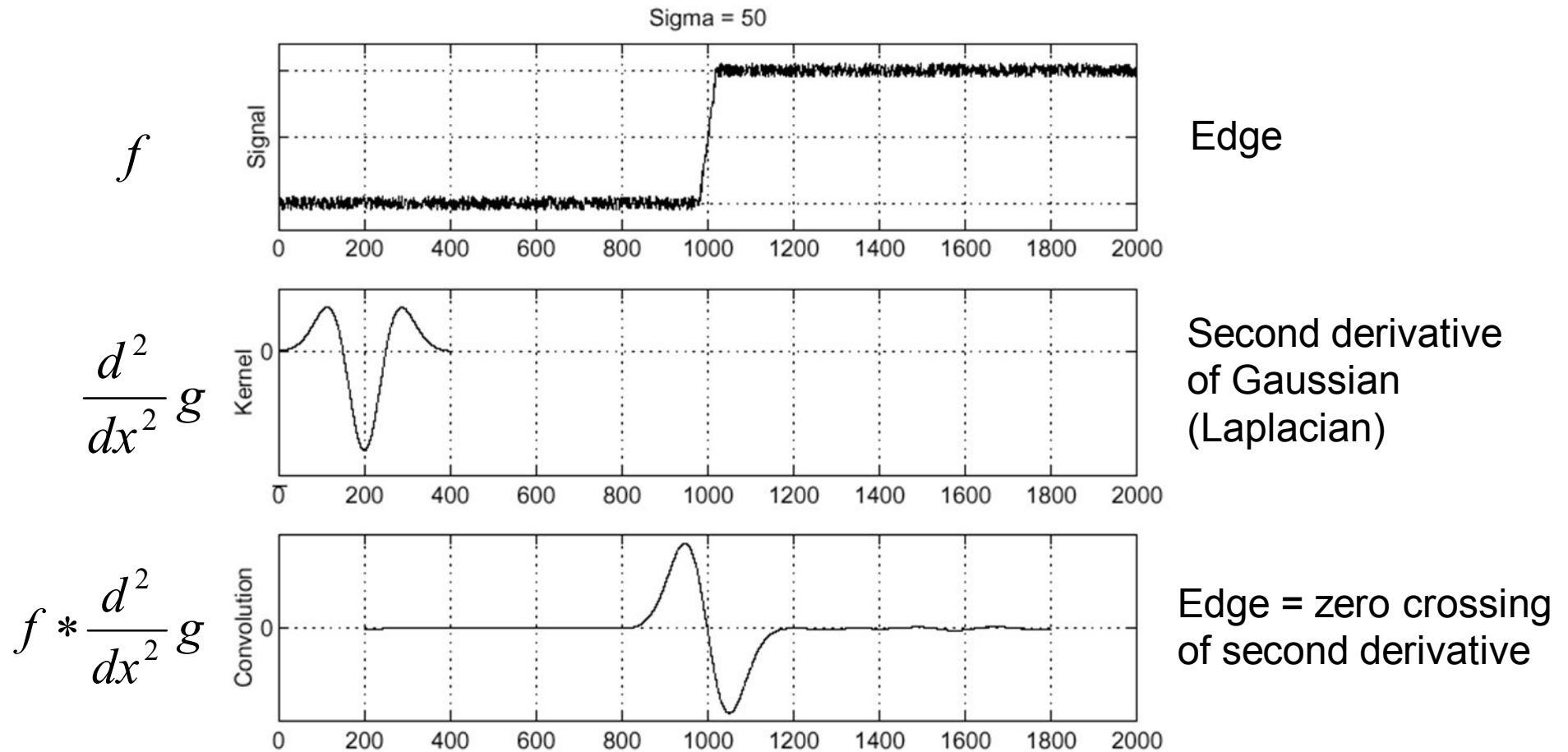


Поиск краев





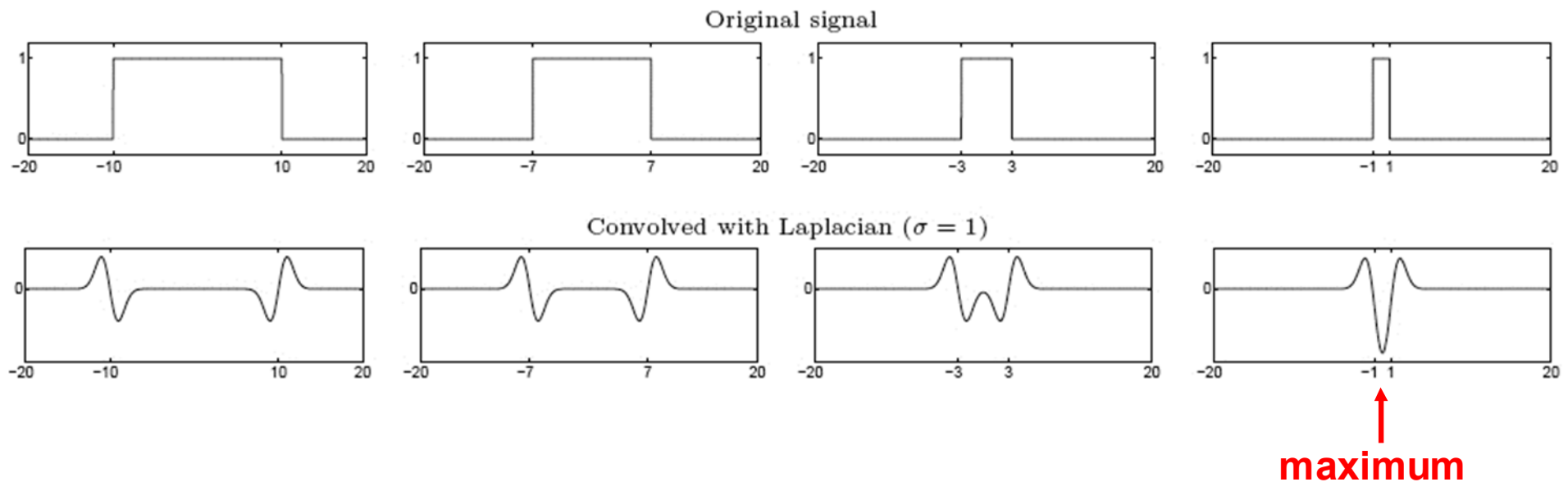
Второй проход





От краев к блобам

- Край = «всплеск»
- Блоб = совмещение двух «всплесков»

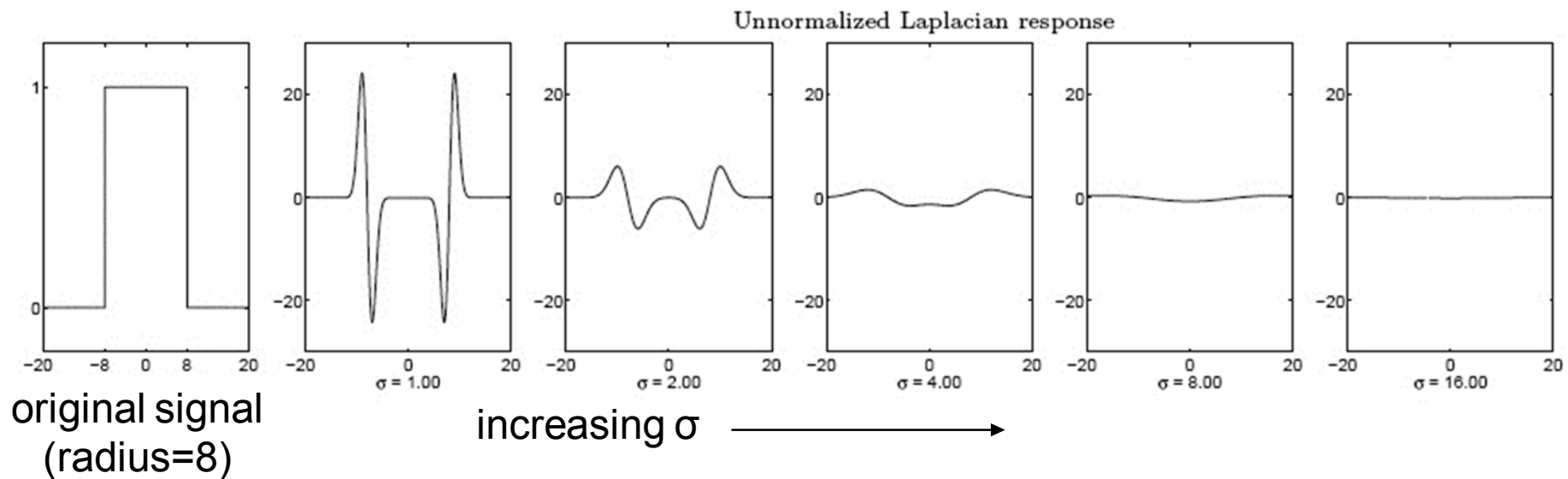


Выбор масштаба: величина отклика Лапласиана достигает максимума в центре блоба в том случае, если размер Лапласиана «соответствует» размеру блоба



Выбор масштаба

- Нужно найти характеристический размер блока путем свертки с Лапласианом в нескольких масштабах и найти максимальные отклики
- Однако, отклик Лапласиана затухает при увеличении масштаба:

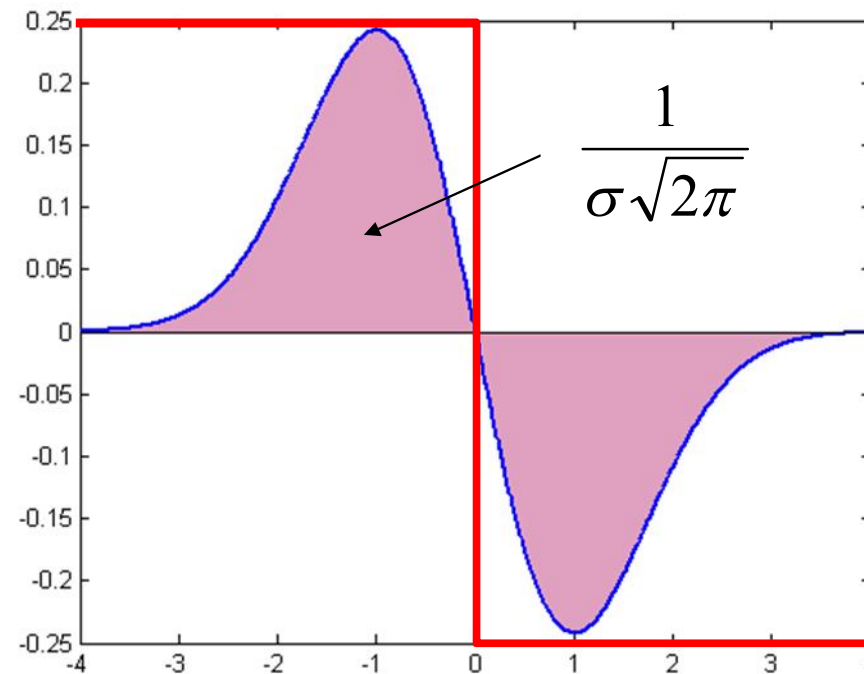


Почему так происходит?



Нормализация по масштабу

- Отклик производной фильтра гаусса на идеальный край затухает с увеличением масштаба σ





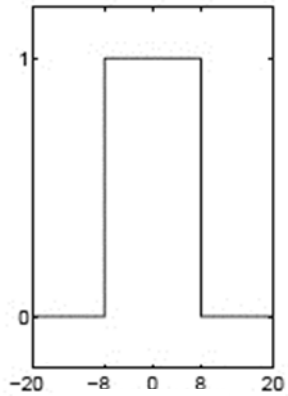
Нормализация масштаба

- Отклик производной фильтра гаусса на идеальный край затухает при увеличении σ
- Нужно домножить производную на σ для достижения инвариантности к масштабу
- Лапласиан это вторая производная фильтра гаусса, поэтому домножаем на σ^2

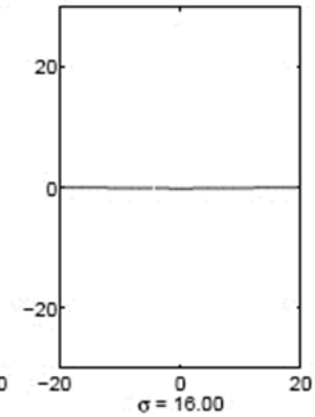
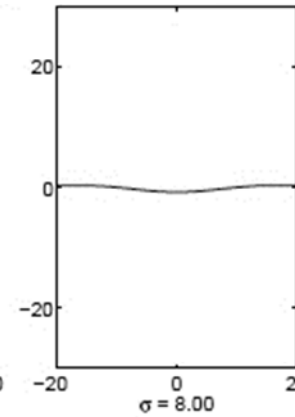
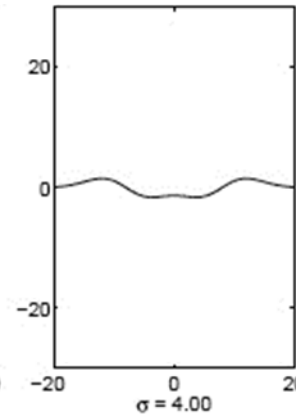
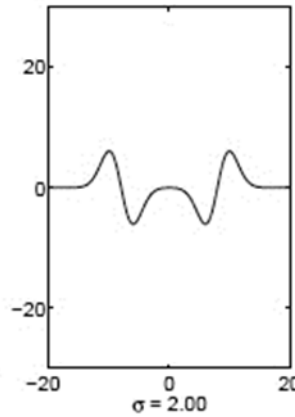
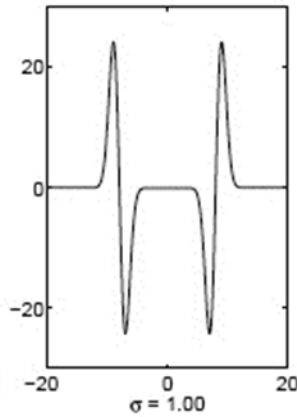


Эффект нормализации

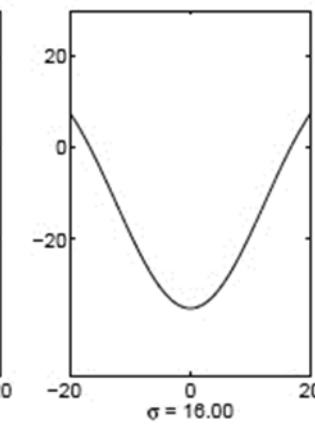
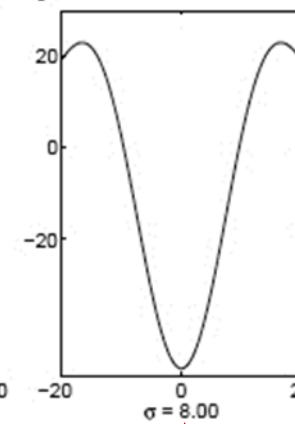
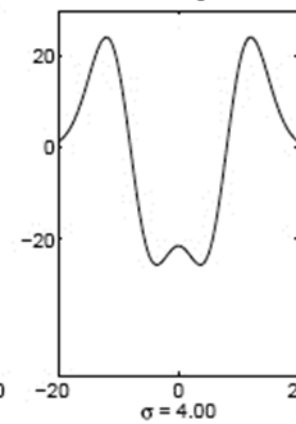
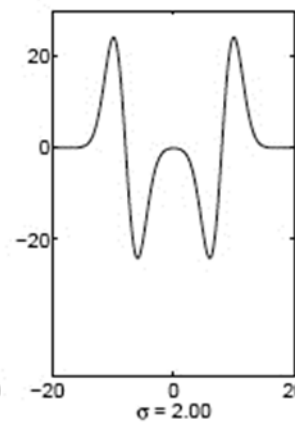
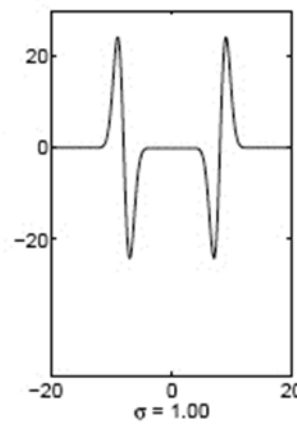
Original signal



Unnormalized Laplacian response



Scale-normalized Laplacian response

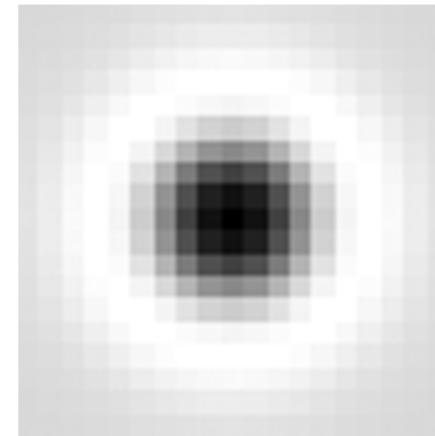
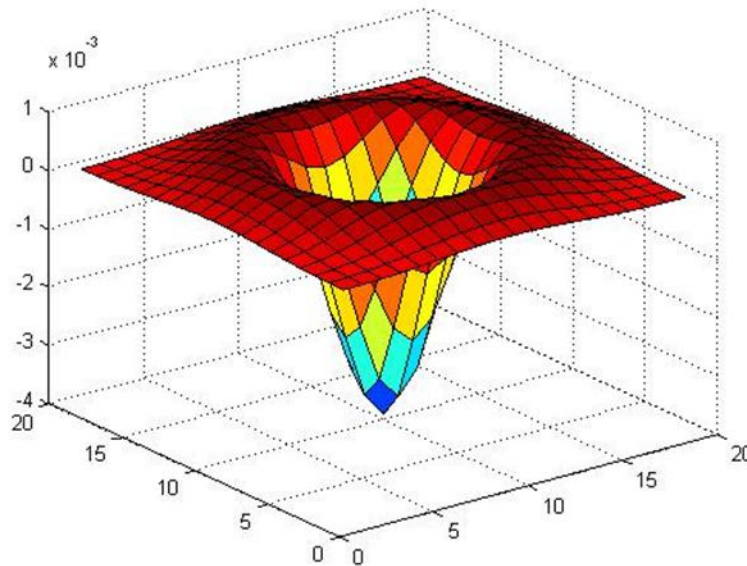


↑
maximum



Поиск блобов в 2D

Лапласиан Гауссиана: Центральено симметричный оператор поиска блобов в 2D

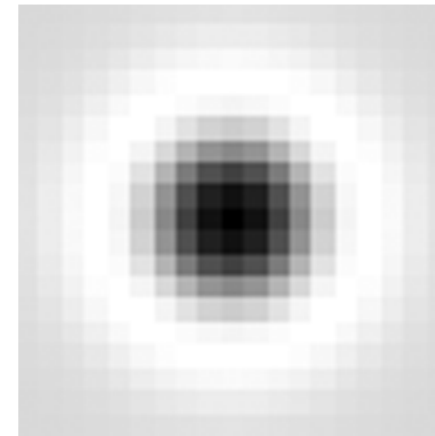
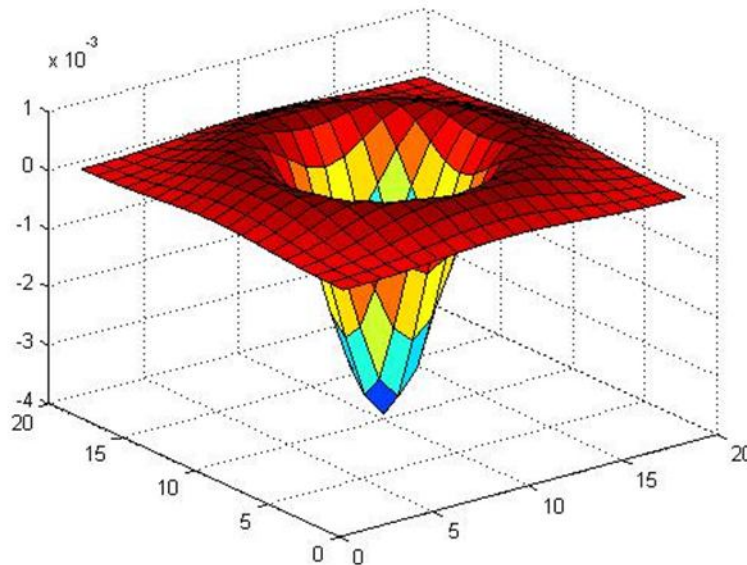


$$\nabla^2 g = \frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2}$$



Поиск блоков в 2D

Лапласиан Гауссиана: Центральное симметричный оператор поиска блоков в 2D



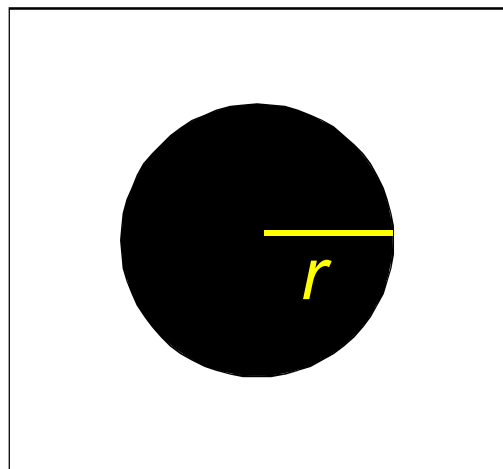
Нормализация:

$$\nabla_{\text{norm}}^2 g = \sigma^2 \left(\frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2} \right)$$

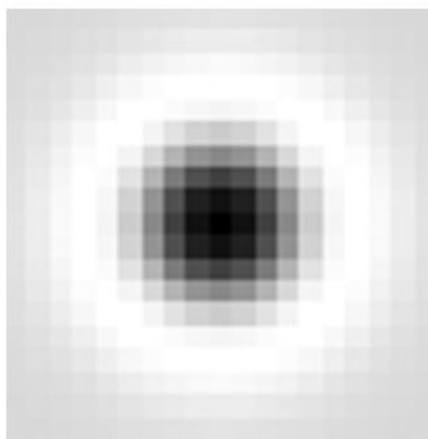


Выбор масштаба

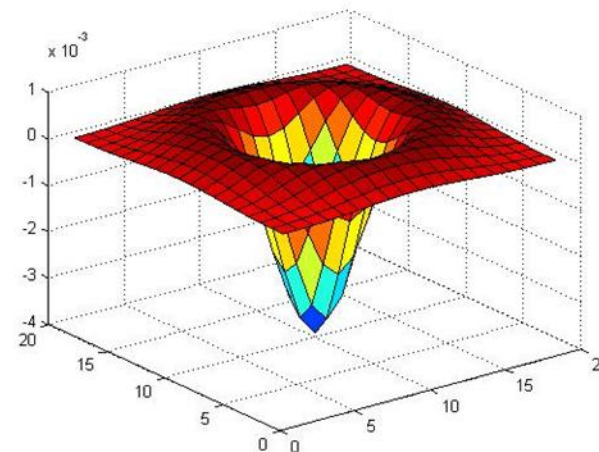
- На каком масштабе Лапласиан достигает максимума отклика на бинарный круг радиуса r ?



image



Laplacian



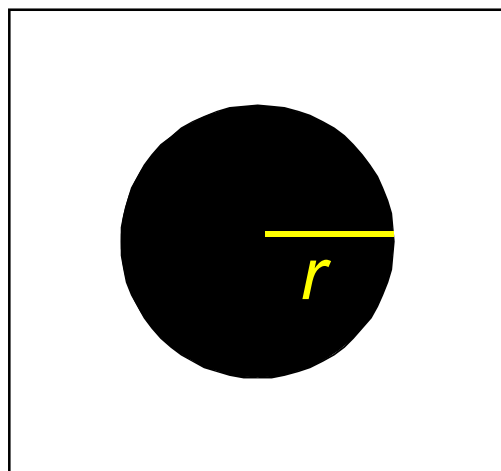


Выбор масштаба

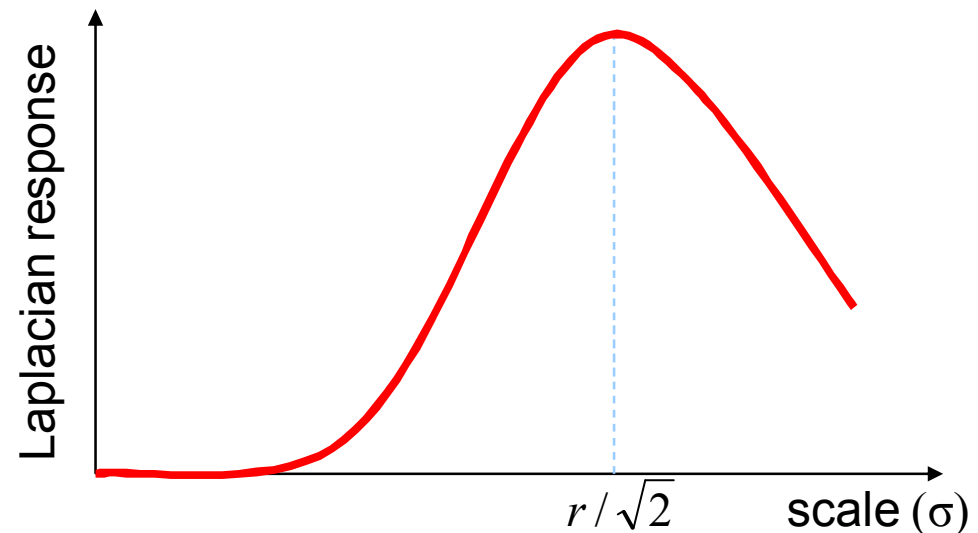
- 2D Лапласиан задается формулой:

$$(x^2 + y^2 - 2\sigma^2) e^{-(x^2 + y^2)/2\sigma^2} \quad (\text{с точностью до масштаба})$$

- Для бинарного круга радиуса r , Лапласиан достигает максимума в $\sigma = r / \sqrt{2}$



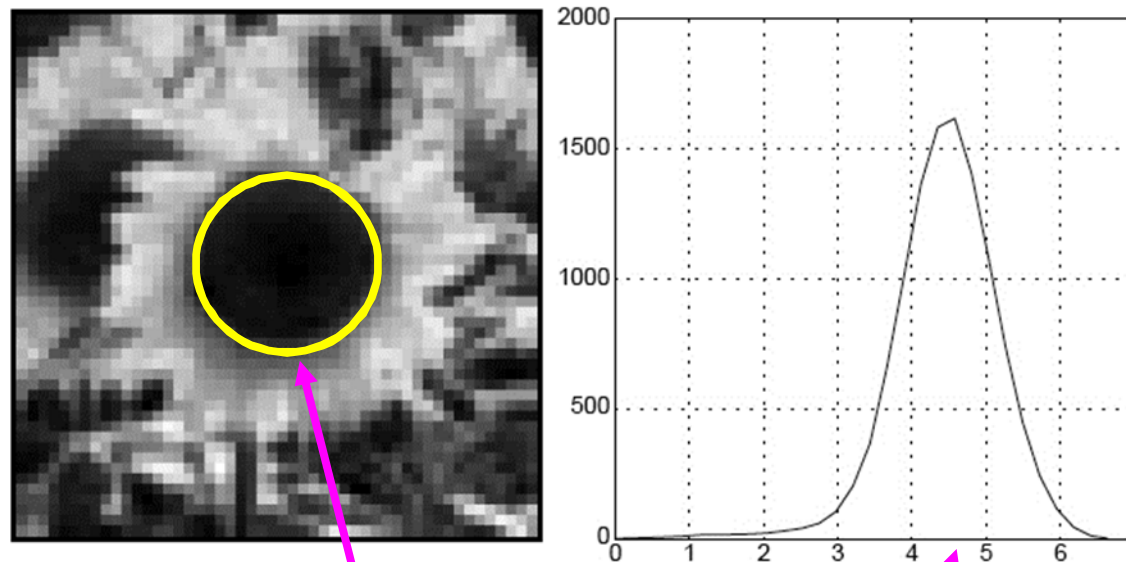
image





Характеристический размер

- Характеристический размер определяется как масштаб, на котором достигается максимум отклика Лапласиана

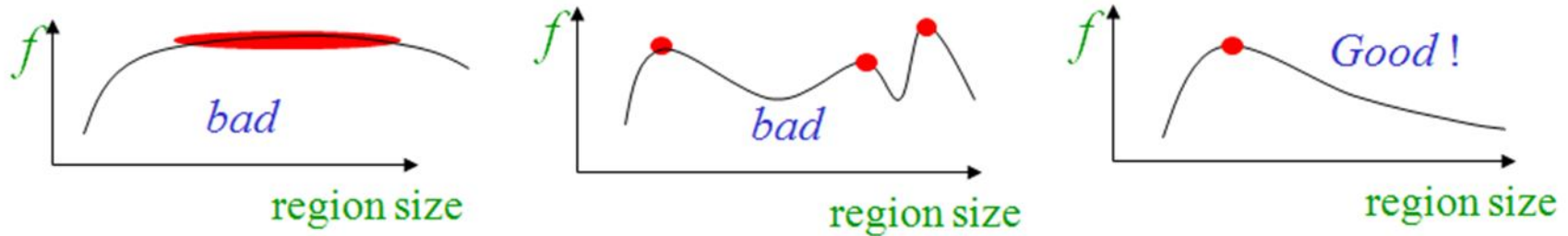


Характеристический масштаб

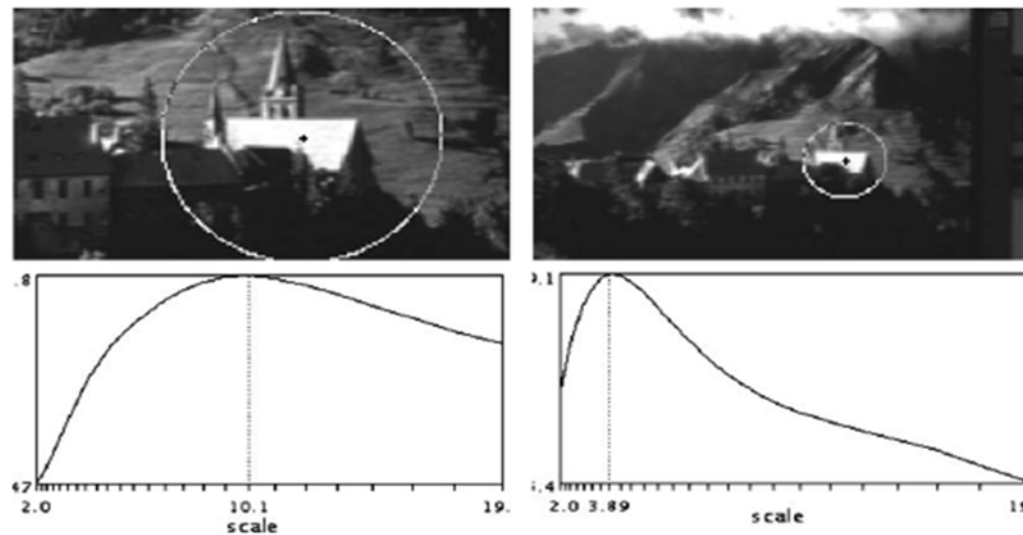
T. Lindeberg (1998). ["Feature detection with automatic scale selection."](#)
International Journal of Computer Vision **30** (2): pp 77--116.



Характеристический размер



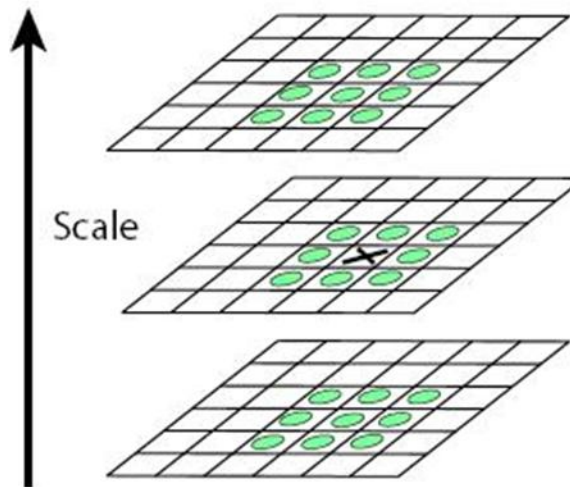
У «хорошего блоба»– один ярко выраженный пик функции





Многомасштабный детектор блобов

1. Свертываем изображение нормализованным фильтром Лапласианом на разных масштабах
2. Ищем максимум отклика Лапласиана в 3D





Пример





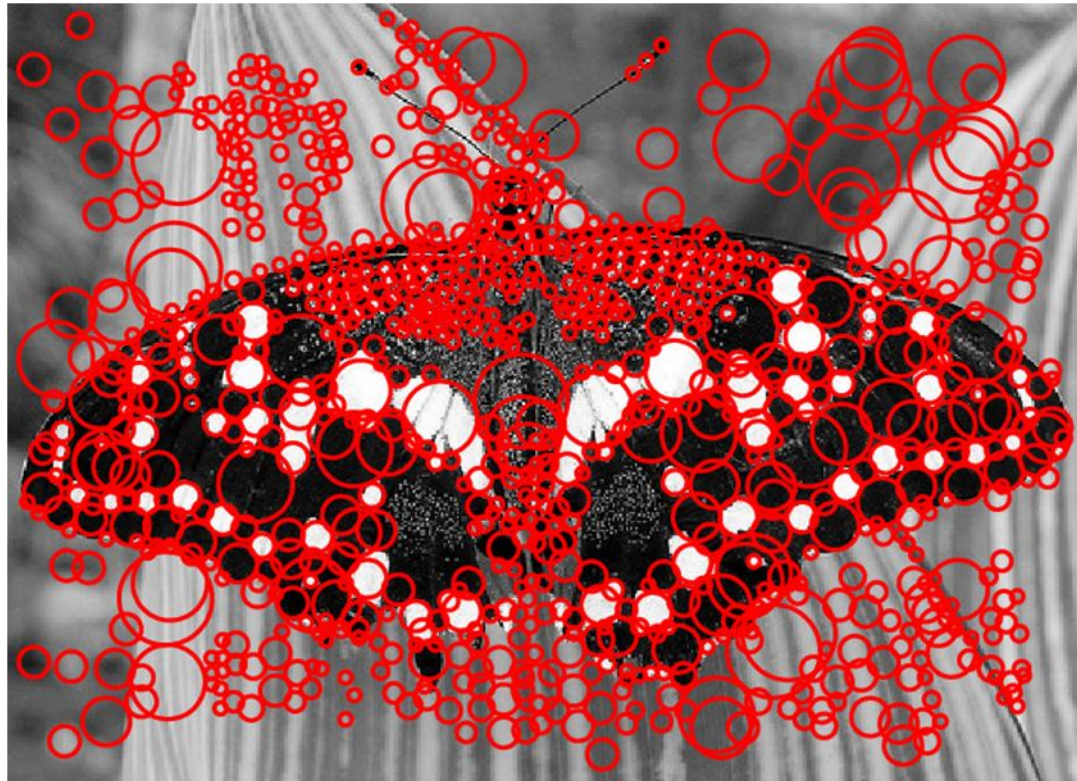
Пример



sigma = 11.9912



Пример





Эффективная реализация (DoG)

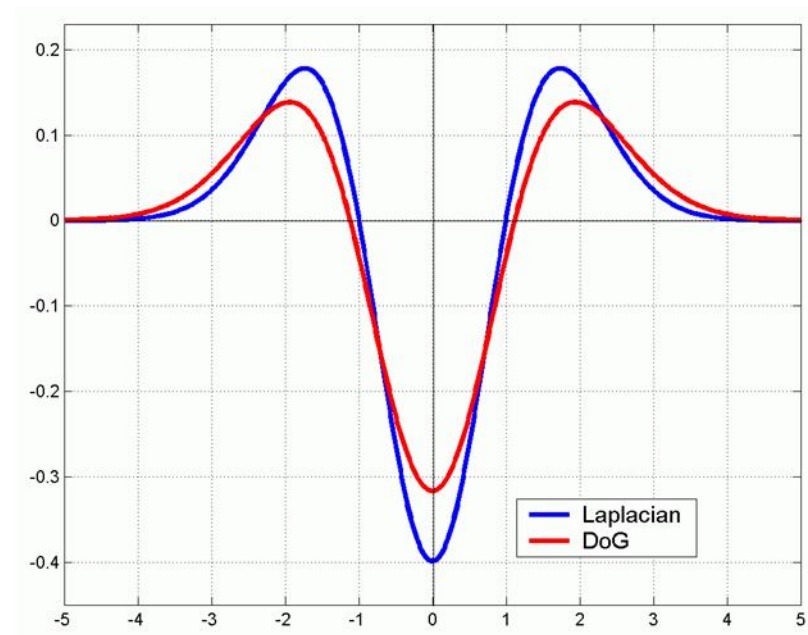
Приближение Лапласиана с помощью разницы гауссиан:

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

(Laplacian)

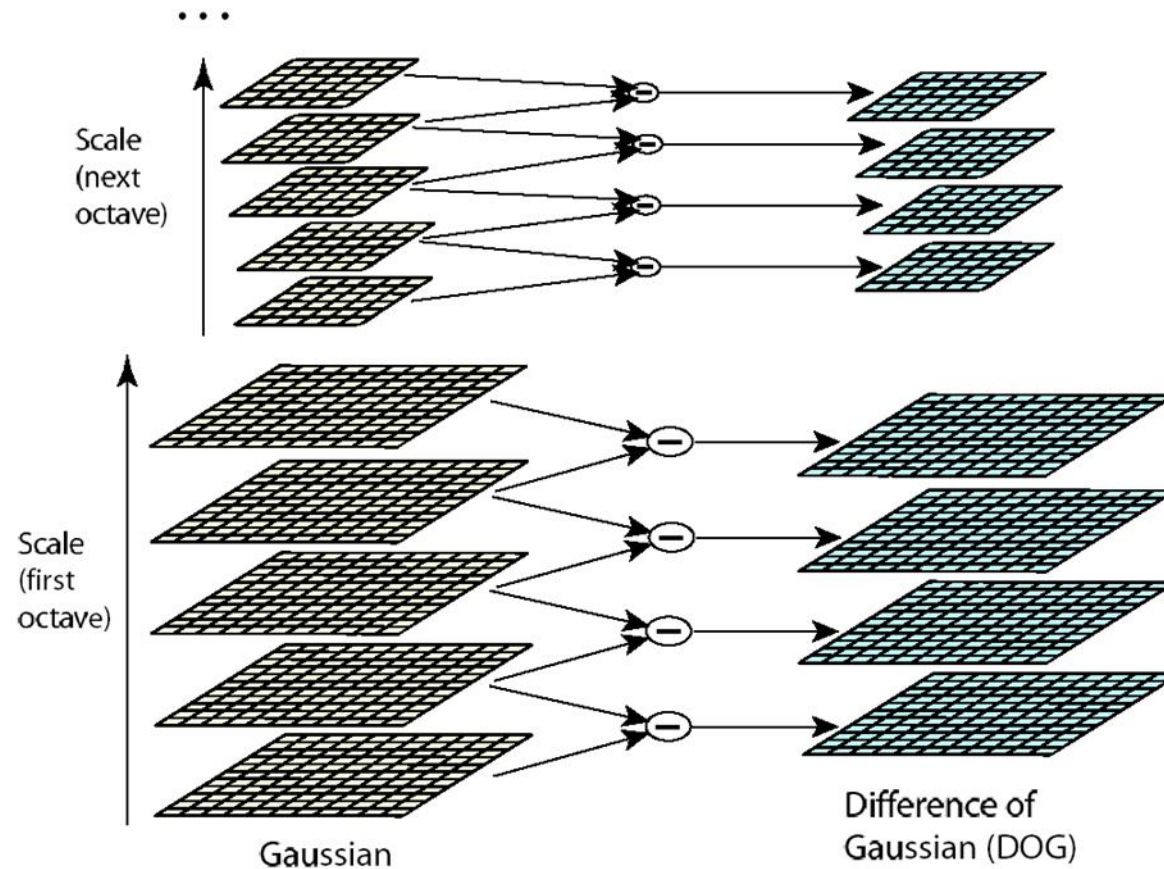
$$DoG = G(x, y, k\sigma) - G(x, y, \sigma)$$

(Difference of Gaussians)





Эффективная реализация (DoG)

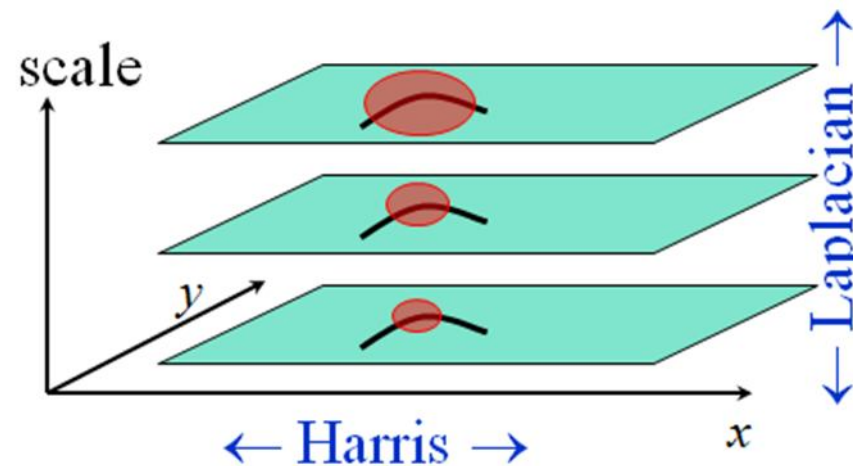


David G. Lowe. ["Distinctive image features from scale-invariant keypoints."](#) *IJCV* 60 (2), pp. 91-110, 2004.



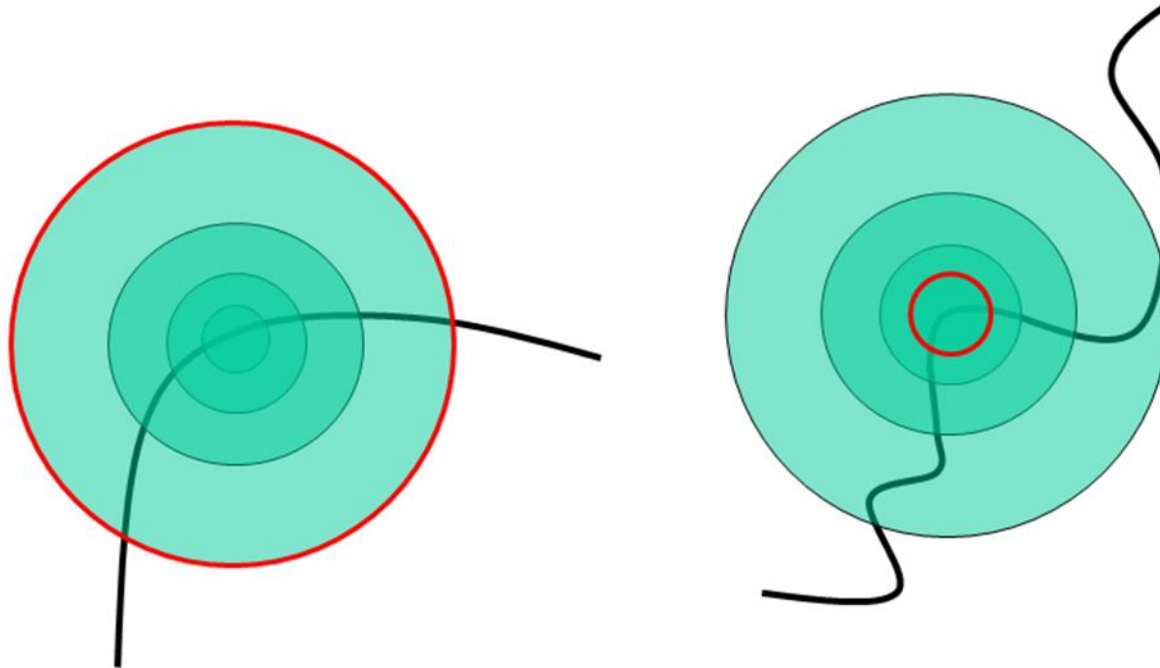
Детектор Harris-Laplacian

- Максимизация:
 - По изображению – откликов углов Харриса
 - По масштабу – Лапласиана
- Разные варианты чередования вычисления функции Харриса и Лапласиана





Детектор Harris-Laplacian



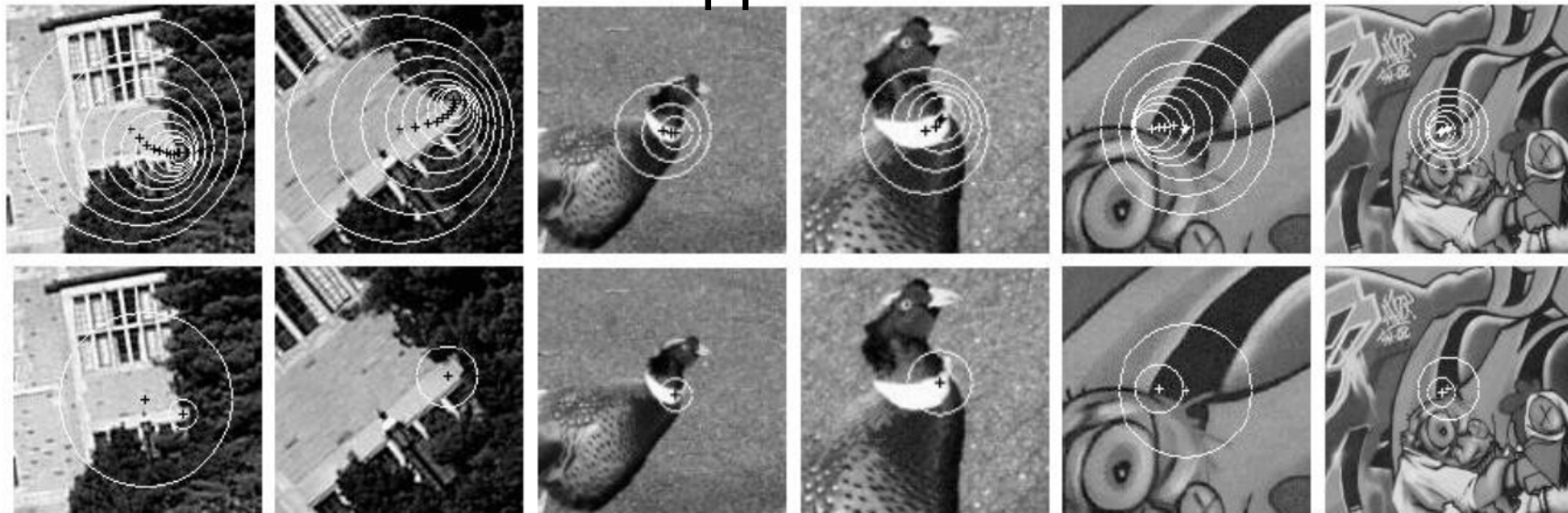
- Находим уголки, а не блобы!



Сравнение

- Сравнение простого детектора Харриса и Харрис-Лапласиана

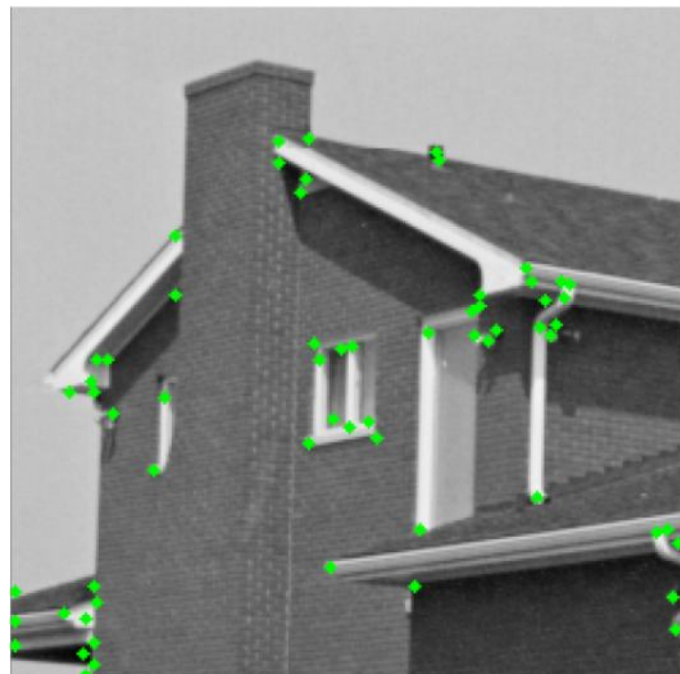
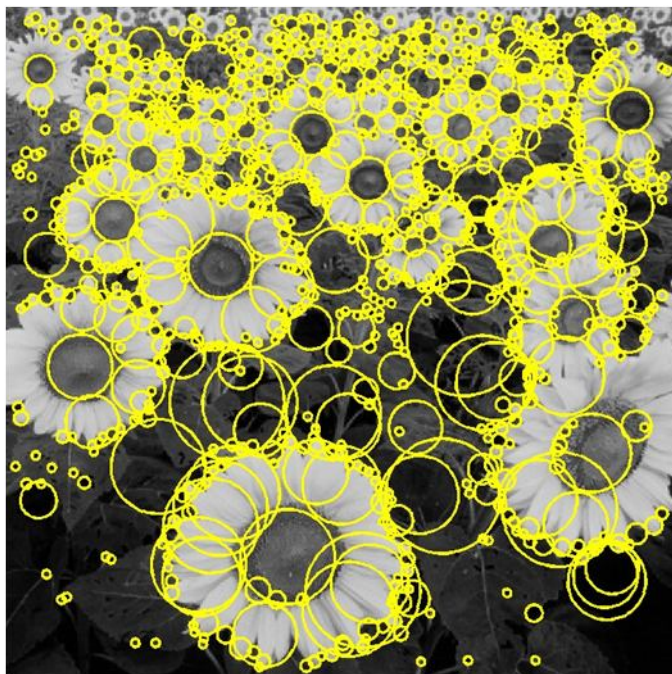
Харрис



Харрис-Лаплас



Углы и блобы

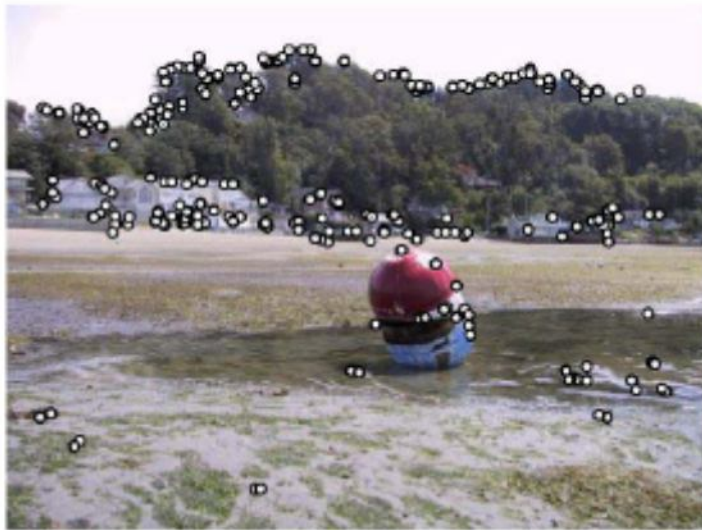


- Углы и блобы – разные виды локальных особенностей
- Детекторы Харрис-Лапласиан и LoG (DoG) находят разные множества особенностей
- Можно применять их параллельно

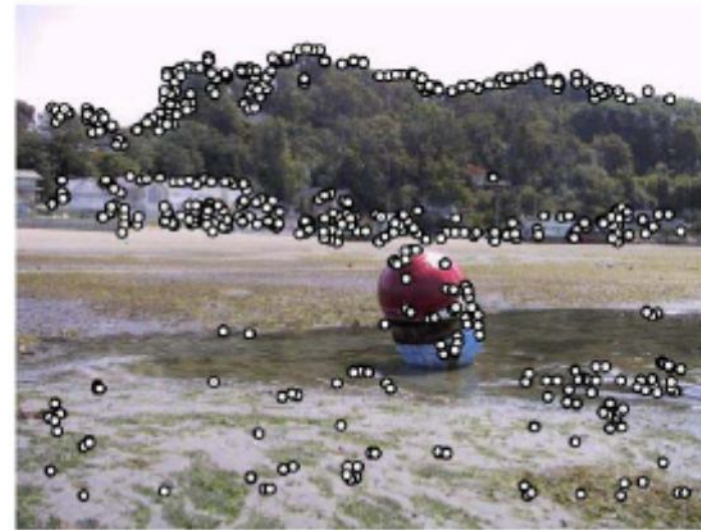


Выбор точек

- Цель: выбрать фиксированное кол-во точек на изображении
 - Точки должны быть равномерно распределены по изображению
 - Самые сильные отклики обычно расположены в текстурированных областях, неравномерно распределенных по изображению



(a) Strongest 250



(b) Strongest 500



Адаптивный радиус

- Пройдёмся по всем точкам в порядке качества
- Для каждой точки выкинем из списка всех соседей в окрестности радиуса r
- Посчитаем количество оставшихся точек
- Выберем такой радиус r , при котором получим нужное нам количество точек





Детекторы областей

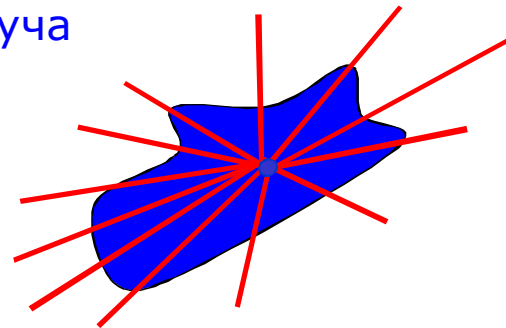
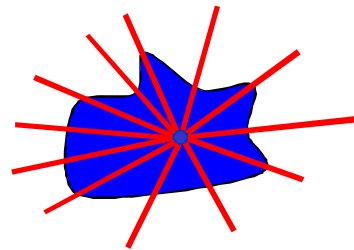
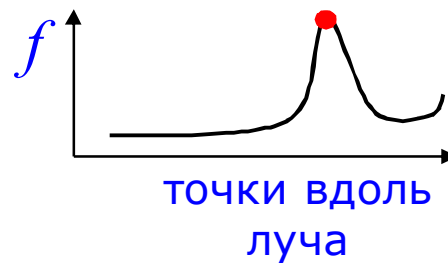
- Стоит попробовать работать с более уникальными характеристиками изображения – с областями
- Интересных областей гораздо меньше, но они более точно характеризуют сцену или объект



Детекторы областей

- Детектор областей IBR
- Идти от локального экстремума яркости по лучам, считая некоторую величину f
- Остановка при достижении пика f

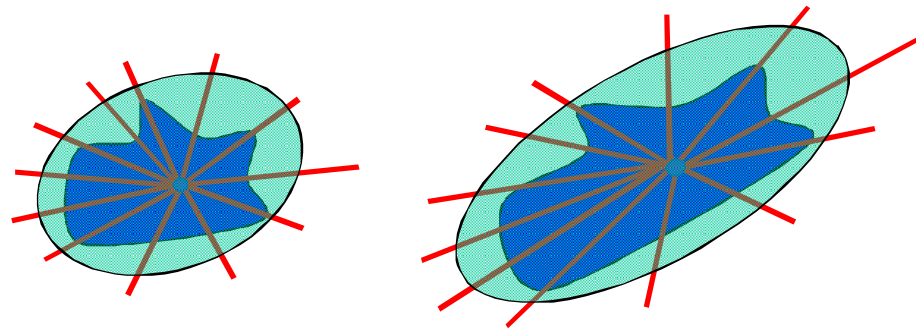
$$f(t) = \frac{|I(t) - I_0|}{\frac{1}{t} \int_0^t |I(t) - I_0| dt}$$





Детекторы областей

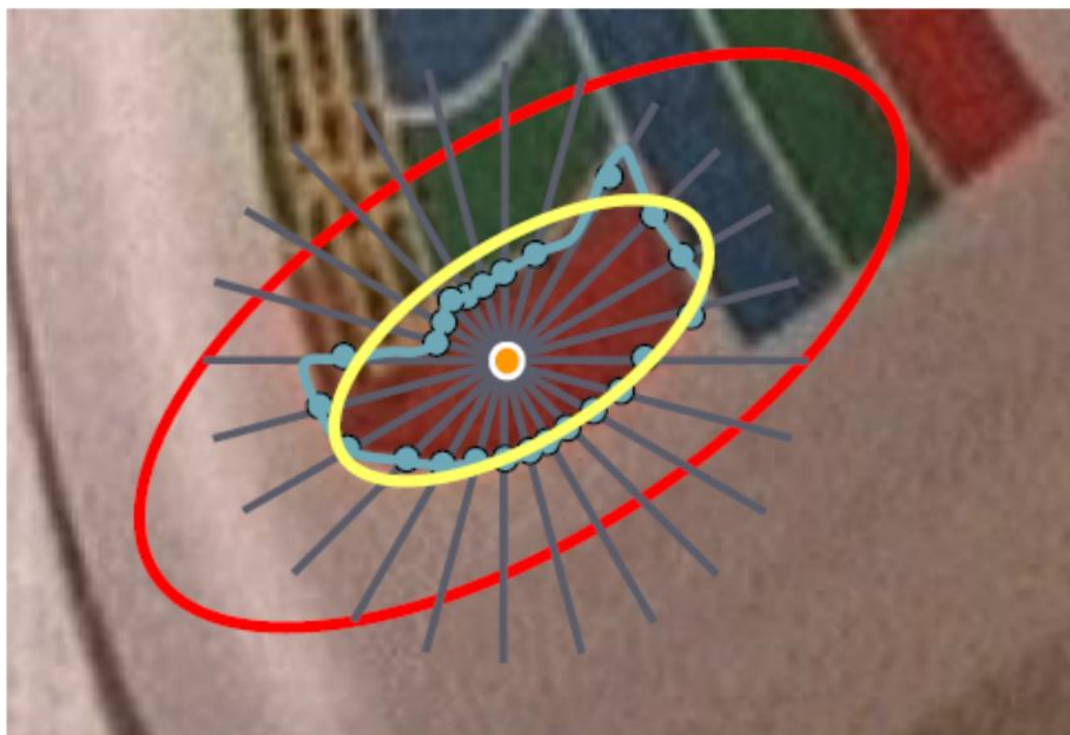
- Области на паре изображений могут различаться, поэтому опишем вокруг них эллипсы



- Если эллипсы превратить в окружности, то получим полное сходство с точностью до поворота (об этом позже)



Детекторы областей





Детекторы областей

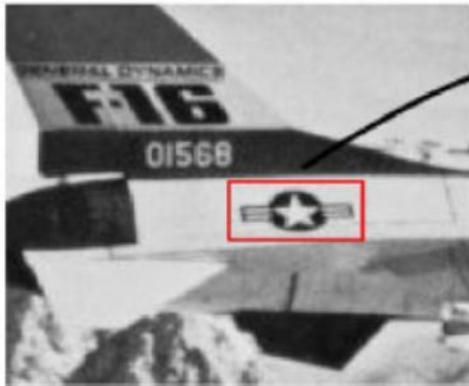
MSER = maximally stable
extremal regions

- Задать порог яркости T
- Провести сегментацию
- Извлечь области
- Для каждой области найти порог, при котором рост площади минимален
- Описать вокруг области эллипс



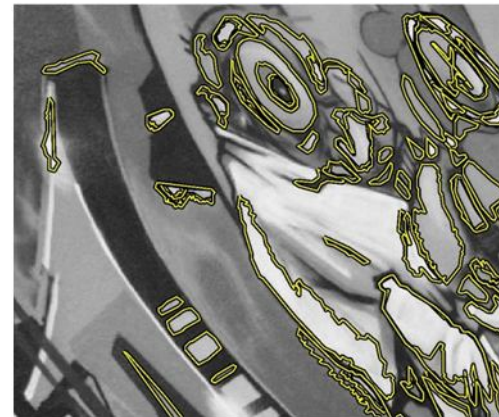
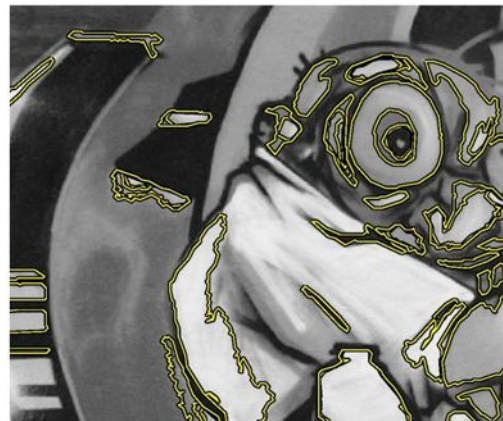


Детекторы областей





Детекторы областей





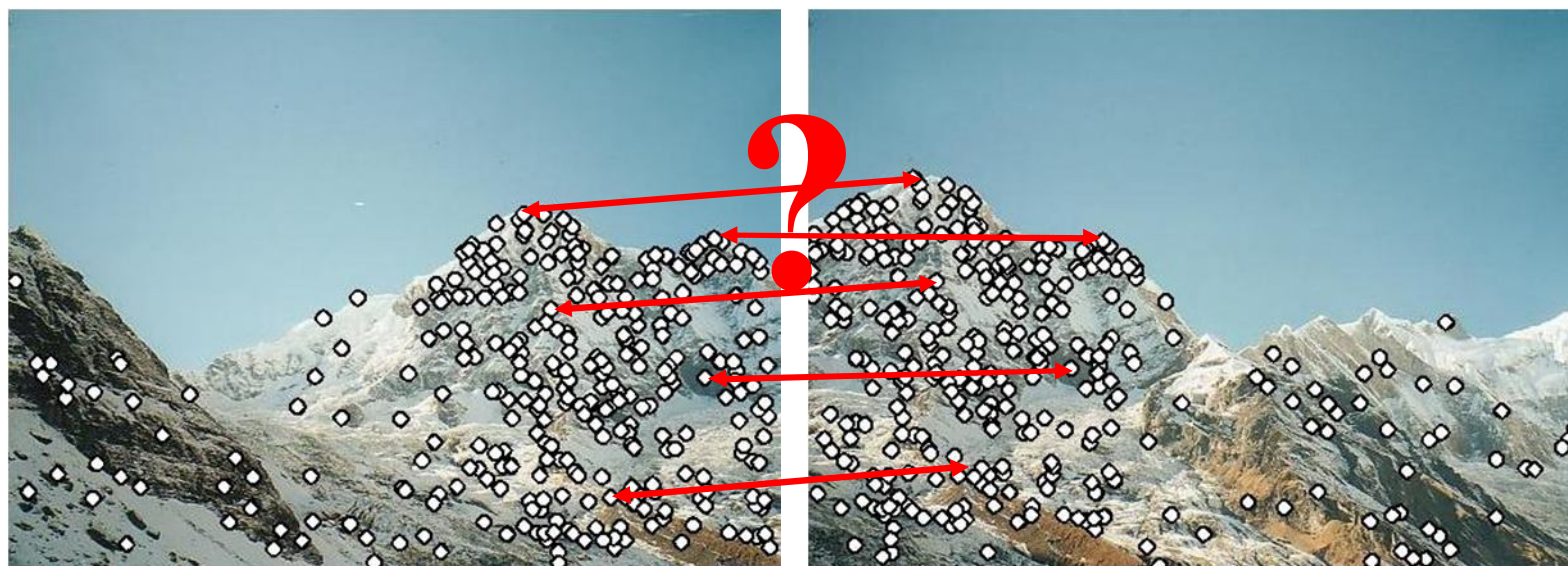
Резюме детекторов

- Необходима инвариантность алгоритма поиска интересных точек к искажениям изображения
- Уголки
 - Harris (Forstner)
 - Harris-Laplace
- Блобы
 - LoG
 - DoG
- Области
 - IBR
 - MSER



Дескрипторы

Точки найдены – как их сопоставить?

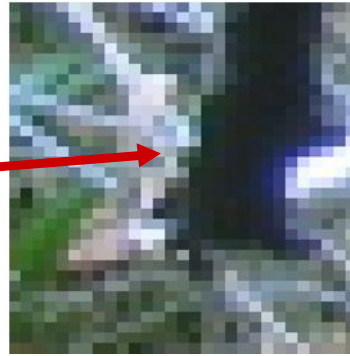
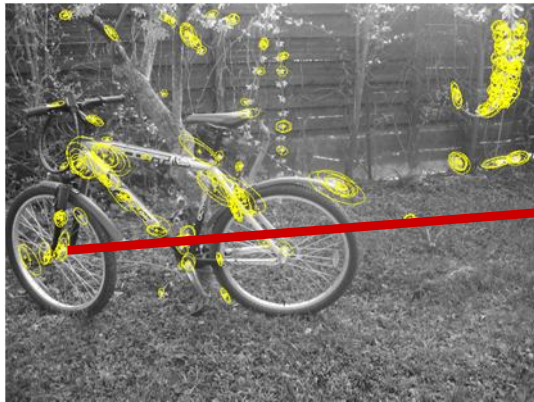


- Нужно как-то описать каждую точку, чтобы можно было отличать одну от другой!
- Дескрипторы (Descriptors)



Дескрипторы

Необходимо каждую интересную точку или область описать набором параметров



$$\mathbf{f}_n = (f_{n,1}, \dots, f_{n,j})^T$$

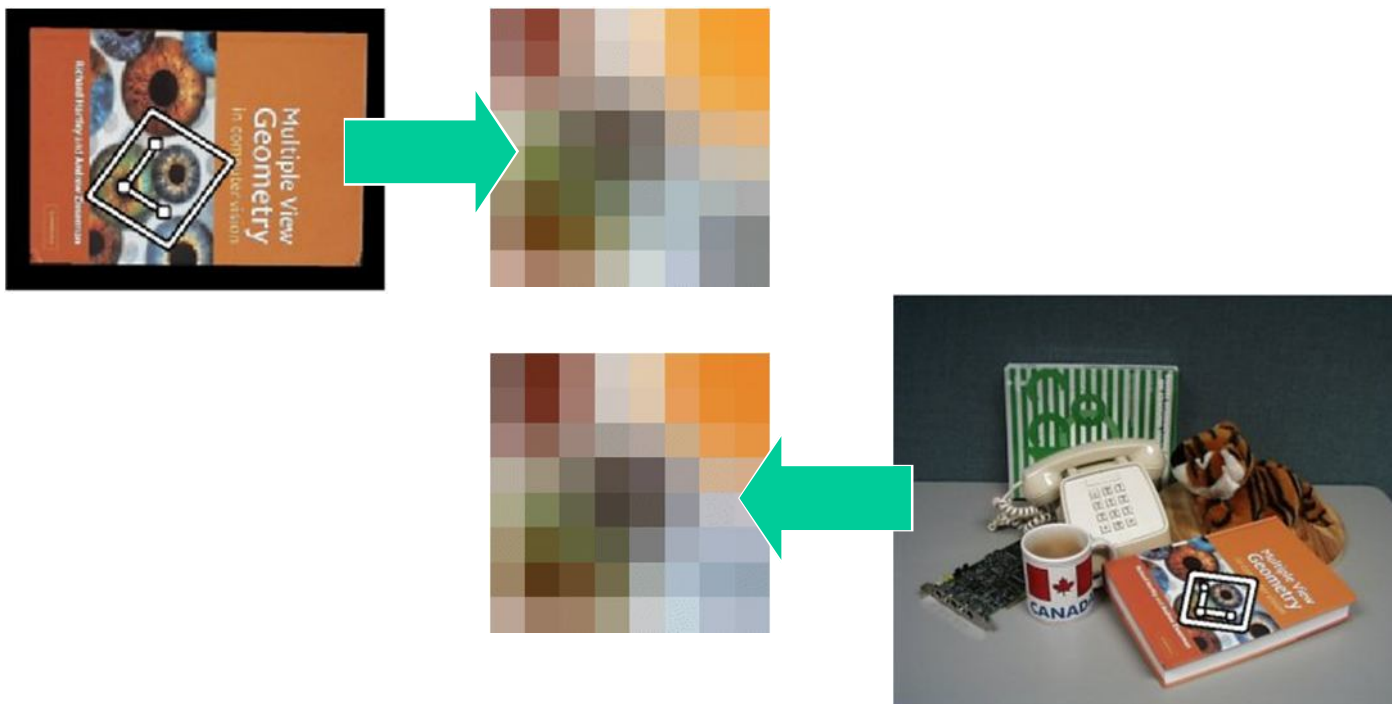
Дескрипторы должны быть:

- специфичны (отличаем разные точки)
- локальны (зависеть только от небольшой окрестности)
- инвариантны (к искажениям/изменению освещенности)
- просты в вычислении



Дескрипторы

- Простейший случай – вектор интенсивностей пикселей окрестности
- Сравниваем, как обычные изображения (SAD, SSD)
- Модель – только сдвиг
- Также верно, если окрестность уже нормализована (геометрическое преобразование)





Дескрипторы

- Инвариантность по отношению к изменению яркости:
 - Локальная нормализация гистограммы
 - Дескрипторы, основанные на градиенте яркости, инвариантны к сдвигу яркости
 - Нормирование яркости - вычесть среднее значение, поделить на дисперсию

$$I' = (I - \mu) / \sigma$$



нормализация





Недостаток простой окрестности

- Небольшие сдвиги приводит к существенному изменению
- Детектор инвариантен к повороту, а дескриптор - нет



- Необходимо:
 - Инвариантность к поворотам
 - Инвариантность к небольшим сдвигам
 - Желательно: устойчивость к другим геометрическим искажениям



SIFT

- Scale-Invariant Feature Transform:
 - Детектор DoG
 - Определение положения и масштаба особенности
 - Ориентация
 - Определение доминантной ориентации по градиентам
 - Дескриптор
 - Использование статистик по направлению градиентам

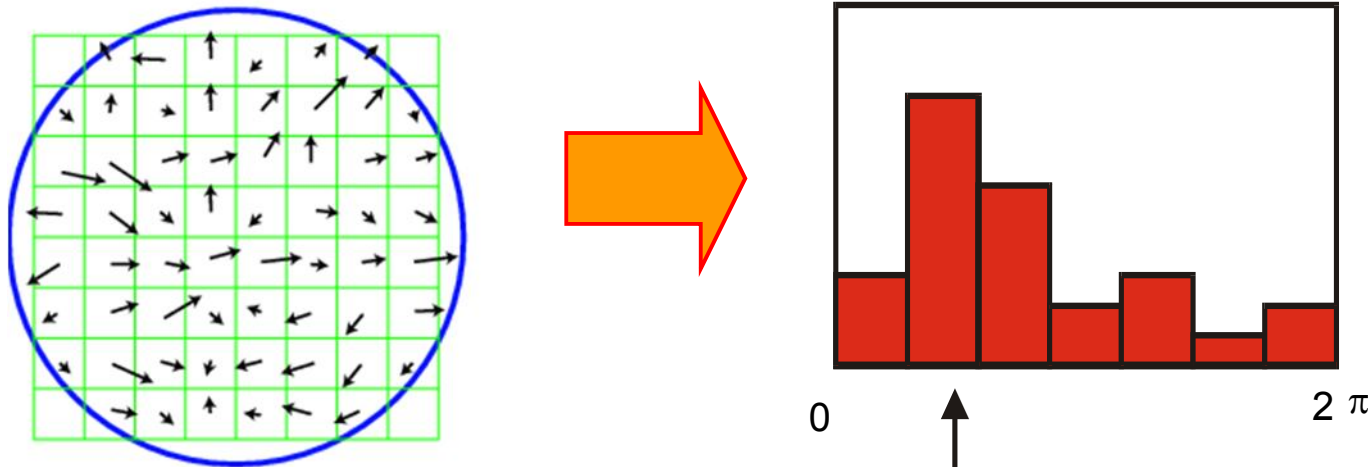
- Устойчив к изменениям освещенности и небольшим сдвигам

David G. Lowe. ["Distinctive image features from scale-invariant keypoints."](#) *IJCV* 60 (2), pp. 91-110, 2004.

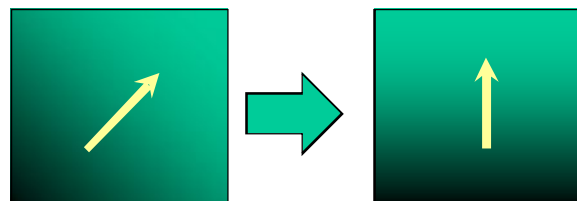


Ориентация

- Идея: найти основное (доминантное) направление градиентов пикселей окрестности



- Повернуть фрагмент так, чтобы доминантное направление градиента было направлено вверх



- Если локальных максимумов несколько – считаем, что несколько точек с разной ориентацией



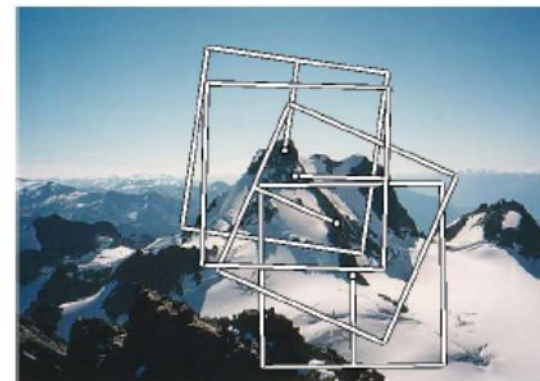
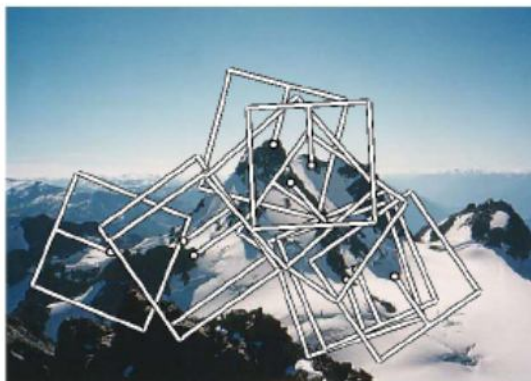
Окрестность особенности



- Для каждой найденной особенности теперь знаем характеристические масштаб и ориентацию
- Выберем соответствующую прямоугольную окрестность
 - (Rotation Invariant Frame)
- Приведем окрестность к стандартному размеру (масштабируем)

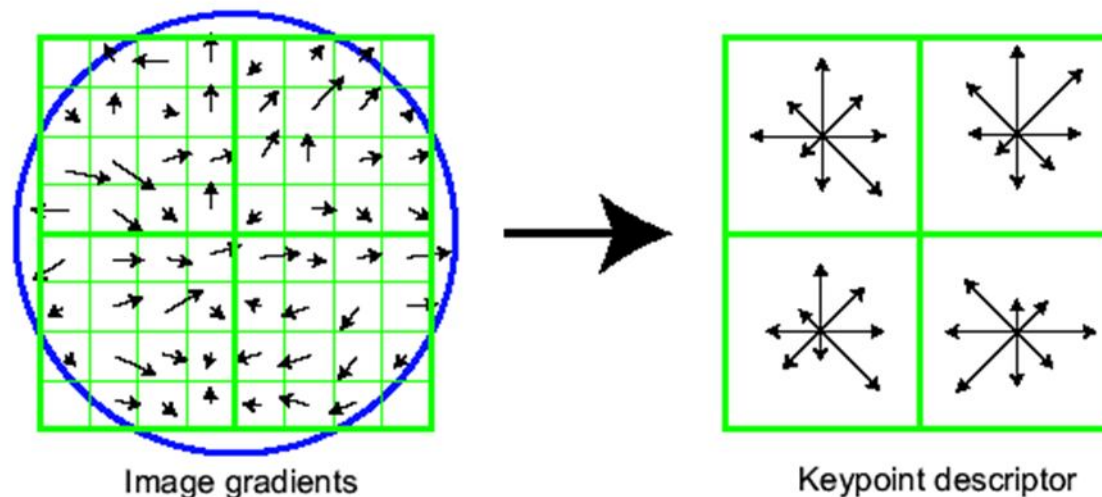


Пример локальных особенностей





Гистограммы градиентов



- Вычисляем градиент в каждом пикселе
- Строим гистограммы направлений градиентов по прямоугольным областям
 - Вклад каждого пикселя взвешиваем по гауссиане с центром в центре окрестности
- Обычно – сетка 4x4, в каждой гистограмма с 8ю ячейками
- Стандартная длина вектора-дескриптора – 128 ($4*4*8$)
- Сравниваем как вектор (разные метрики)



Резюме SIFT

- Детектор SIFT весьма специфичен, устойчив к изменениям освещения, небольшим сдвигам
- Вся схема SIFT (детектор, выбор окрестностей, дескриптор) оказалась очень эффективным инструментом для анализа изображений
- Очень широко используется

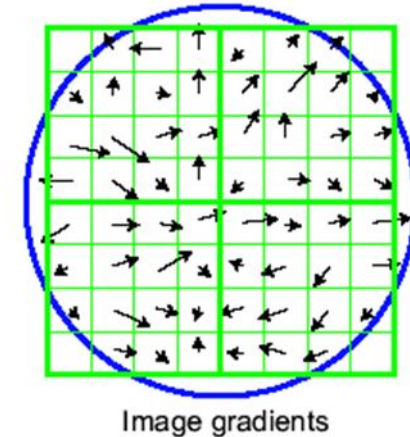


Image gradients



PCA-SIFT

- Изменяется только сам дескриптор
- Идея: после нормализации масштаба и поворота окрестности можно описать гораздо компактнее
- Используем МГК (PCA)
 - Окрестность 41×41
 - Считаем градиент в каждом пикселе
 - МГК для большой выборки фрагментов
 - Выбираем 20 главных компонент
- Более компактное описание
 - 20 вместо 128

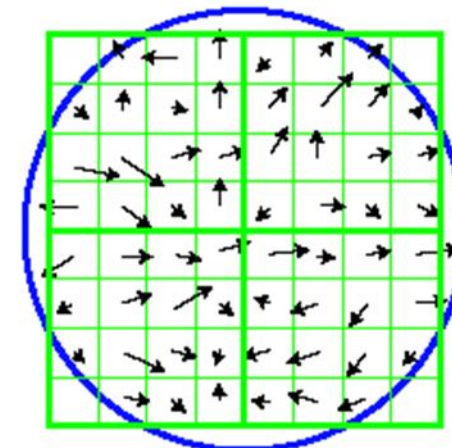
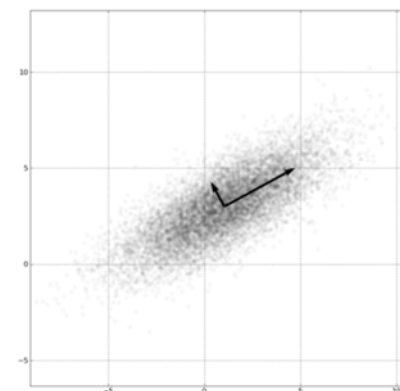


Image gradients





SIFT vs PCA-SIFT



(A1) SIFT:
4/10 correct



(A2) PCA-SIFT ($n=20$):
9/10 correct



(B1) SIFT:
6/10 correct



(B2) PCA-SIFT ($n=20$):
10/10 correct

- Утверждается, что лучше по точности
- Быстрее сравнение
- (?) исследуется

	time (sec)	σ
Localization and I/O	2.63	0.09
SIFT representation	1.59	0.06
PCA-SIFT representation	1.64	0.04
SIFT matching	2.20	0.03
PCA-SIFT matching	0.58	0.05



Использование цвета

- RGB-SIFT

- 3 дескриптора SIFT для каждого канала

- C-SIFT

- Каналы O_1 и O_2

$$\begin{pmatrix} O_1 \\ O_2 \\ O_3 \end{pmatrix} = \begin{pmatrix} \frac{R-G}{\sqrt{2}} \\ \frac{R+G-2B}{\sqrt{6}} \\ \frac{R+G+B}{\sqrt{3}} \end{pmatrix}$$

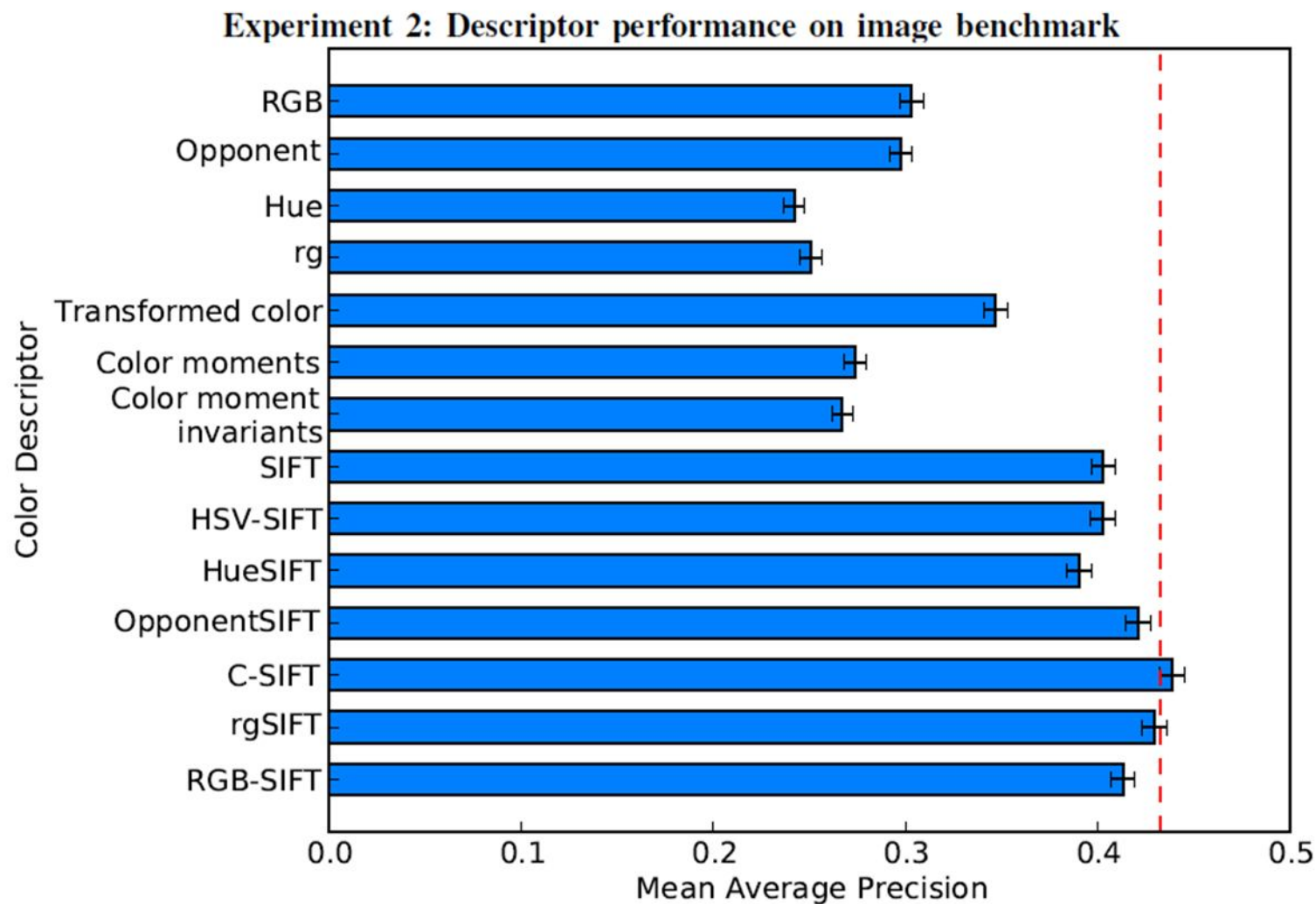
- rgSIFT

- Каналы r и g

$$\begin{pmatrix} r \\ g \\ b \end{pmatrix} = \begin{pmatrix} \frac{R}{R+G+B} \\ \frac{G}{R+G+B} \\ \frac{B}{R+G+B} \end{pmatrix}$$

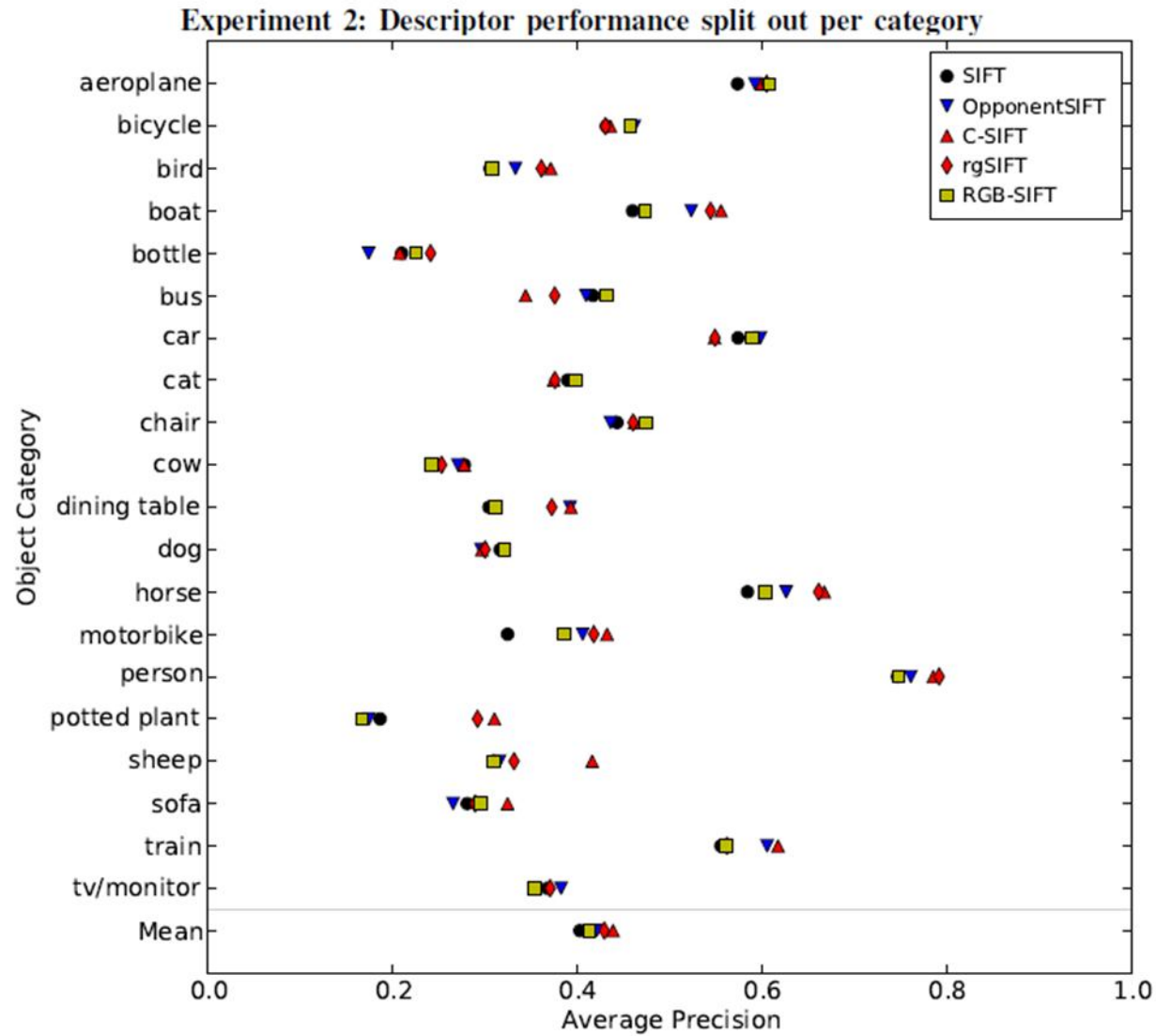


Использование цвета





Использование цвета





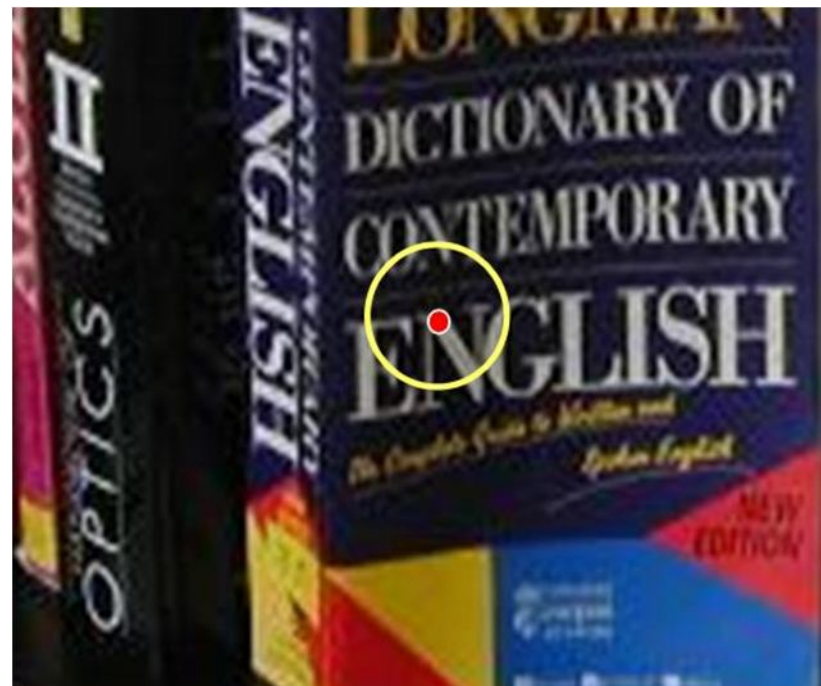
ИСПОЛЬЗОВАНИЕ ЦВЕТА

Combinations on image benchmark

Author	Point sampling	Descriptor	Spatial pyramid	Mean average precision
<i>This paper</i>	Harris-Laplace, dense sampling	SIFT	1x1+2x2+1x3	0.558
<i>This paper</i>	Harris-Laplace, dense sampling	C-SIFT	1x1+2x2+1x3	0.566
Marszałek <i>et al.</i> [38]	Harris-Laplace, dense sampling, Laplacian	SIFT, HueSIFT, other	1x1+2x2+1x3	0.575
Marszałek <i>et al.</i> [38]	Harris-Laplace, dense sampling, Laplacian	SIFT, HueSIFT, other; with feature selection	1x1+2x2+1x3	0.594
<i>This paper</i>	Harris-Laplace, dense sampling	SIFT, OpponentSIFT, <i>rg</i> SIFT, C-SIFT, RGB-SIFT	1x1+2x2+1x3	0.605

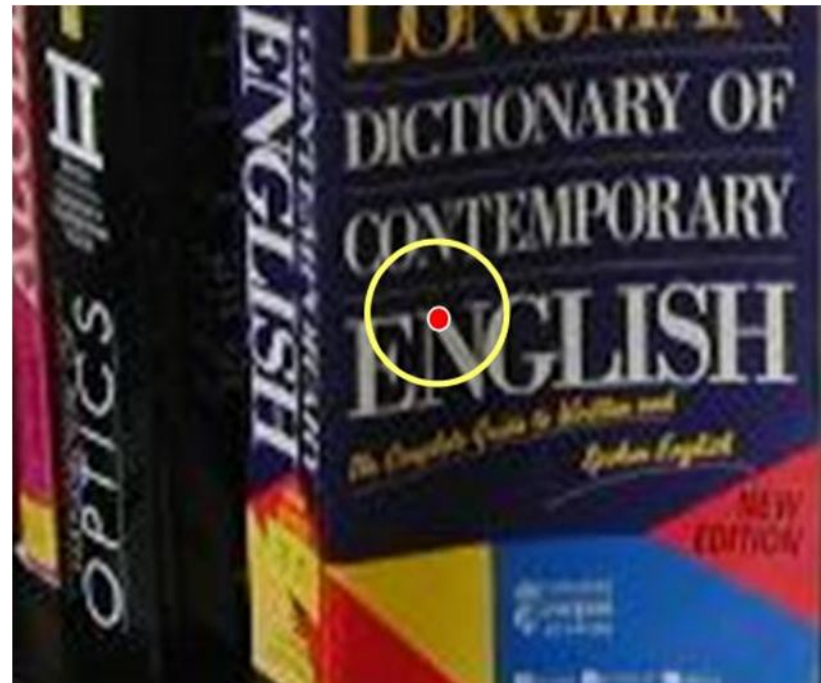


Перспективные искажения





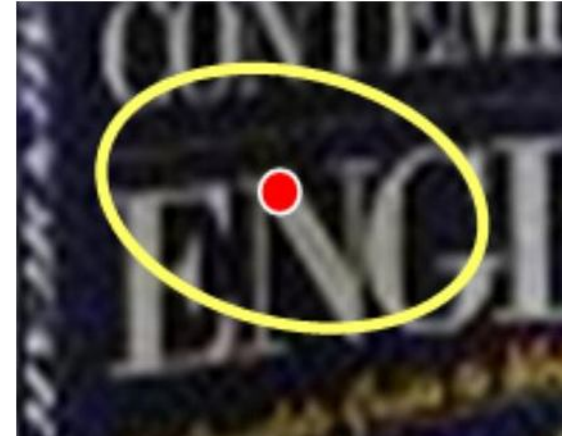
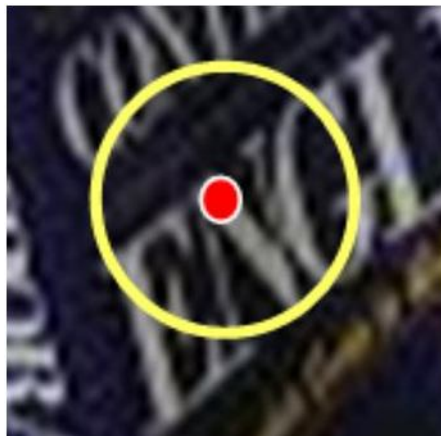
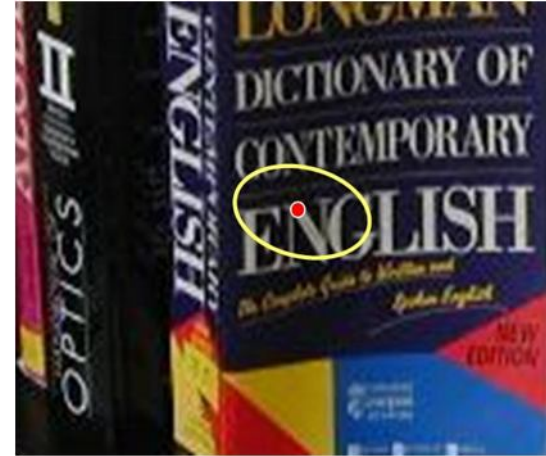
Дескрипторы



В круглую окрестность попадают разные фрагменты – в левом снимке внутрь окружности попала половина буквы G, в правом он почти не попала



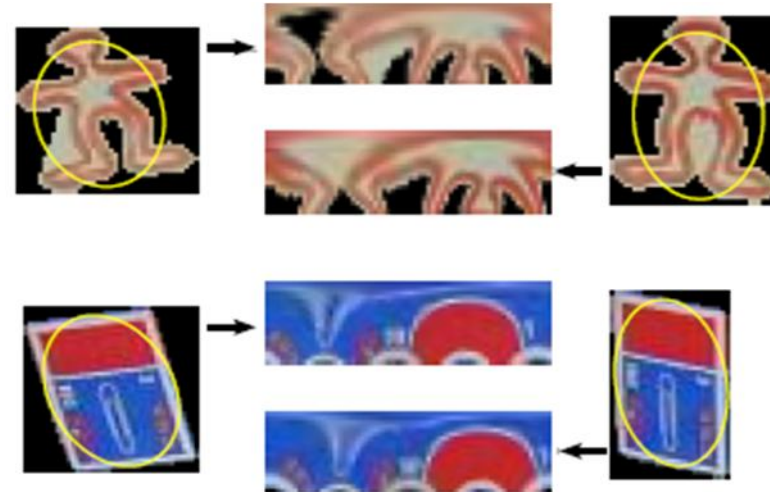
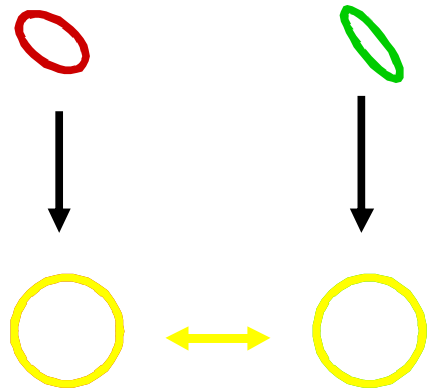
Дескрипторы



Найти соответствующие окрестности, с учетом аффинных преобразований описав их эллипсом



Дескрипторы



Для облегчения сравнения фрагментов изображения необходимо найти параметры эллипса вокруг интересной точки или области и привести эллипсы к «каноническому» виду – «общему знаменателю»



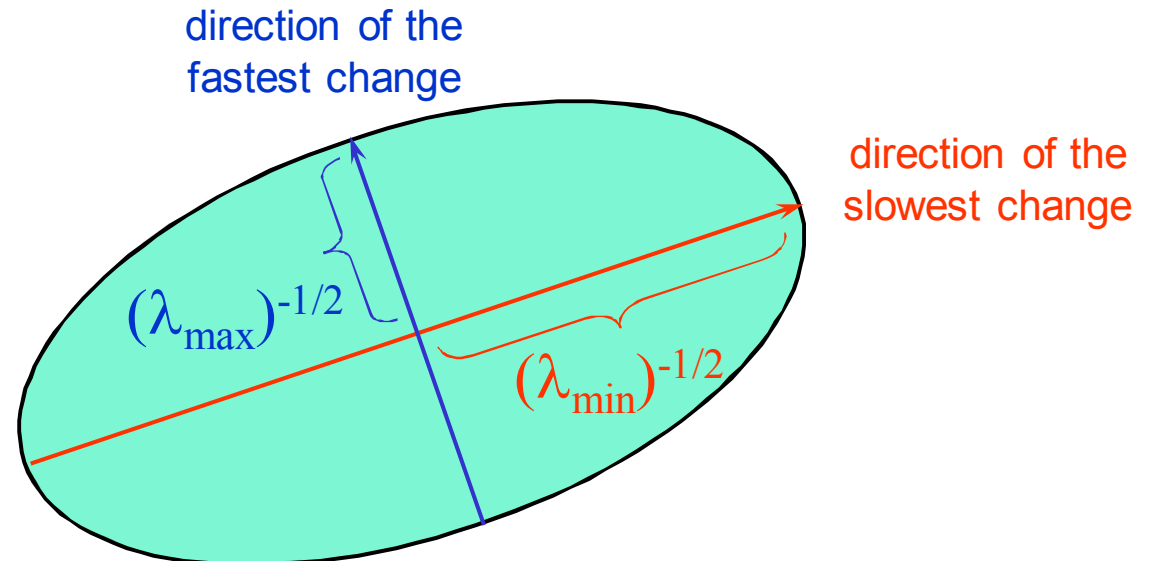
Аффинная адаптация

Помним:
$$M = \sum_{x,y} w(x,y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} = R^{-1} \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} R$$

Матрицу M можно представить как эллипс, у которого длины осей определены собственными значениями, а ориентация определена матрицей R

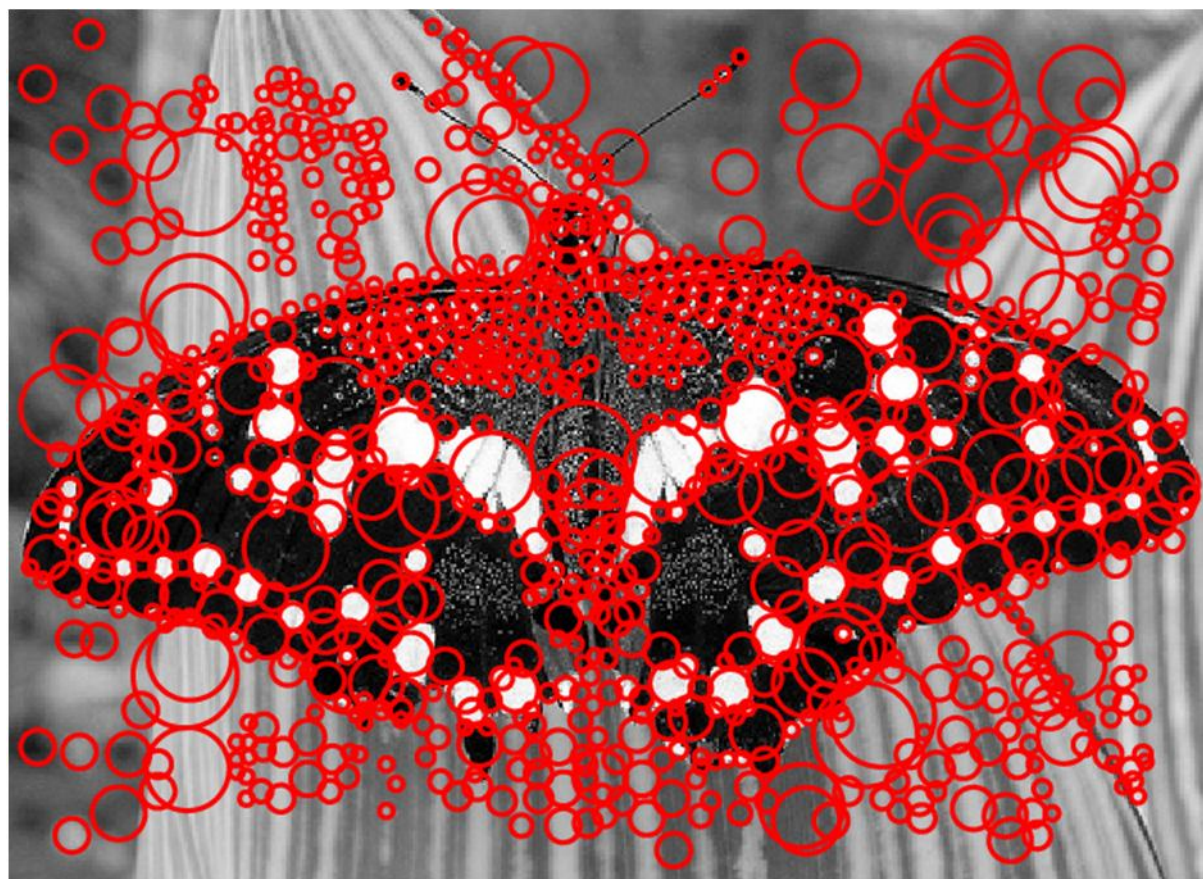
Уравнение эллипса:

$$[u \ v] M \begin{bmatrix} u \\ v \end{bmatrix} = \text{const}$$





Пример аффинной адаптации



Независимые от масштаба области (блобы)



Пример аффинной адаптации

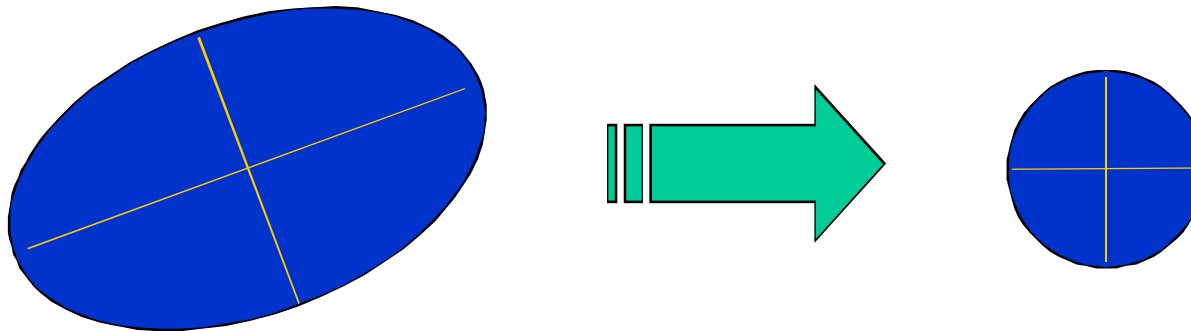


Уточненные окрестности блобов



Нормализация

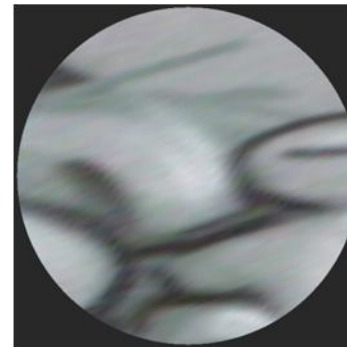
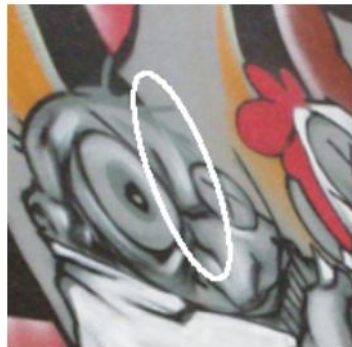
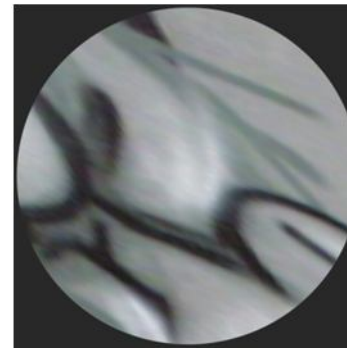
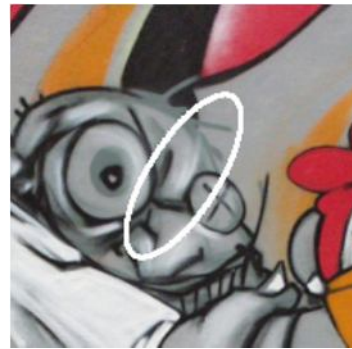
- Эллипс вторых моментов можно считать «характеристической формой» области
- Нормализуем окрестности путем преобразования эллипса в единичный круг





Неопределенность ориентации

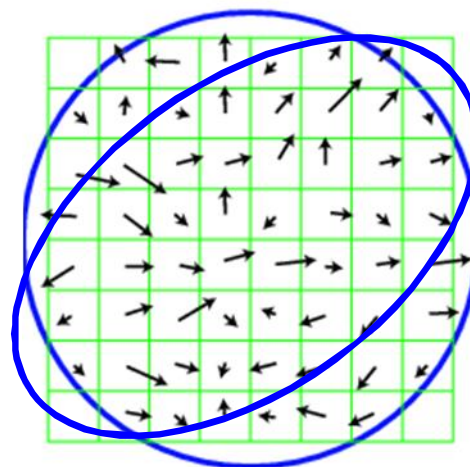
- Нет уникального преобразования из эллипса в единичный круг
- Мы можем вращать и отражать единичный круг, и он останется единичным кругом





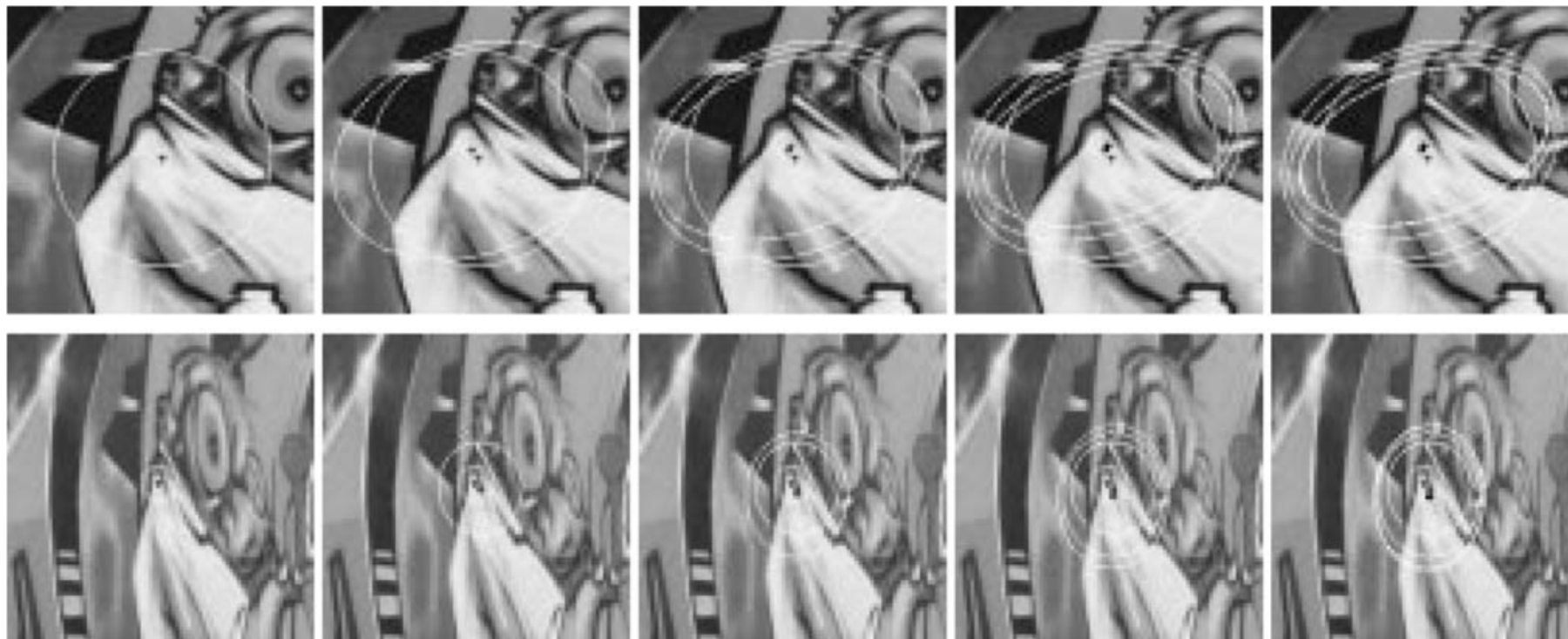
Аффинная адаптация

- Задача: матрица вторых моментов, определенная весами $w(x,y)$ должна совпадать с характерной формой области
- Решение: итеративное уточнение
 - Считаем матрицу моментов по круглому окну
 - Применяем аффинную адаптацию для получения эллиптического окна
 - Пересчитываем матрицу моментов по нормализованной окрестности. Повторяем.





Итеративная афинная адаптация



Initial

1

2

3

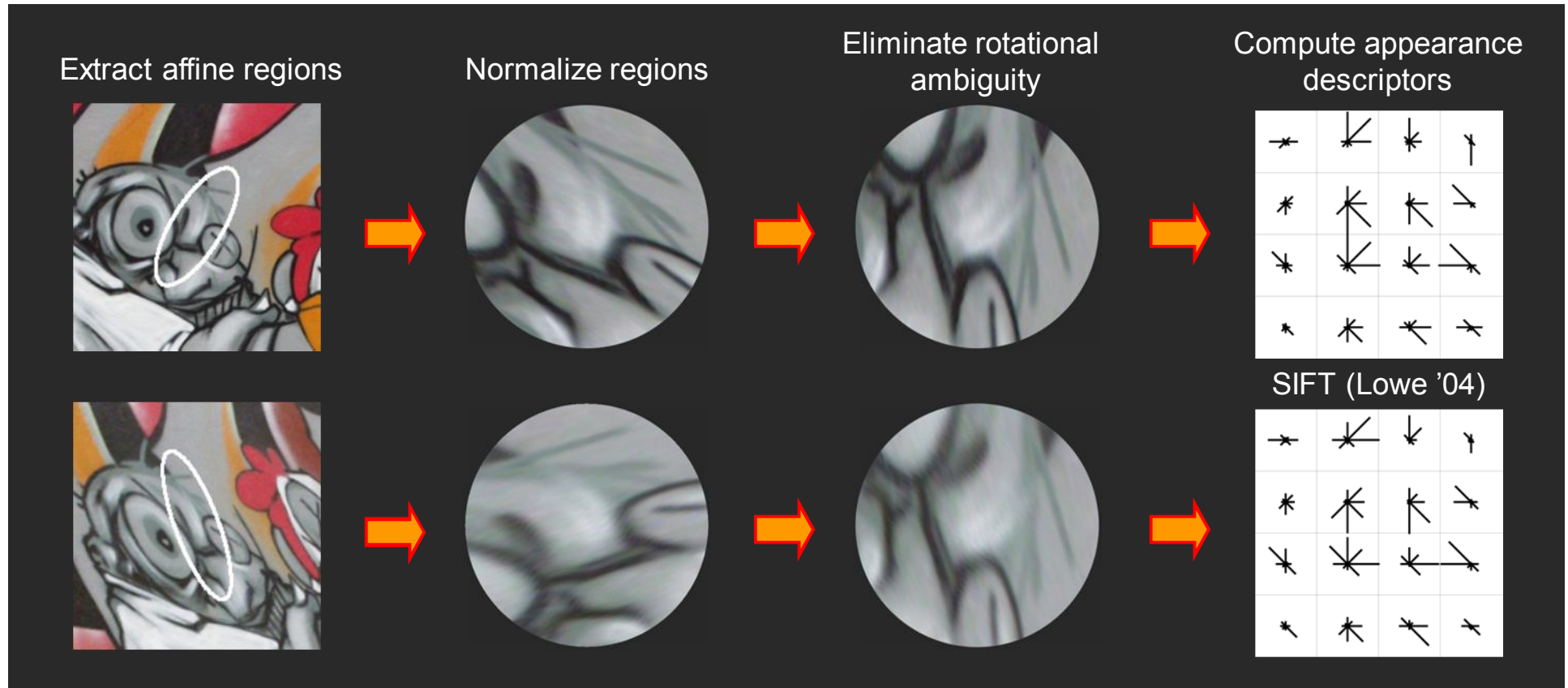
4

K. Mikolajczyk and C. Schmid, [Scale and Affine invariant interest point detectors](#), IJCV 60(1):63-86, 2004.

<http://www.robots.ox.ac.uk/~vgg/research/affine/>



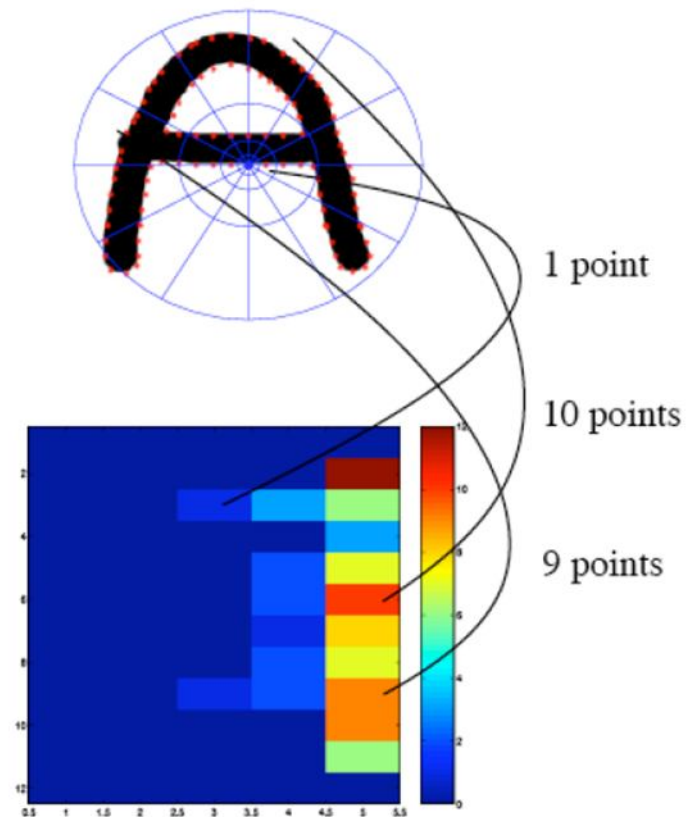
Пример:





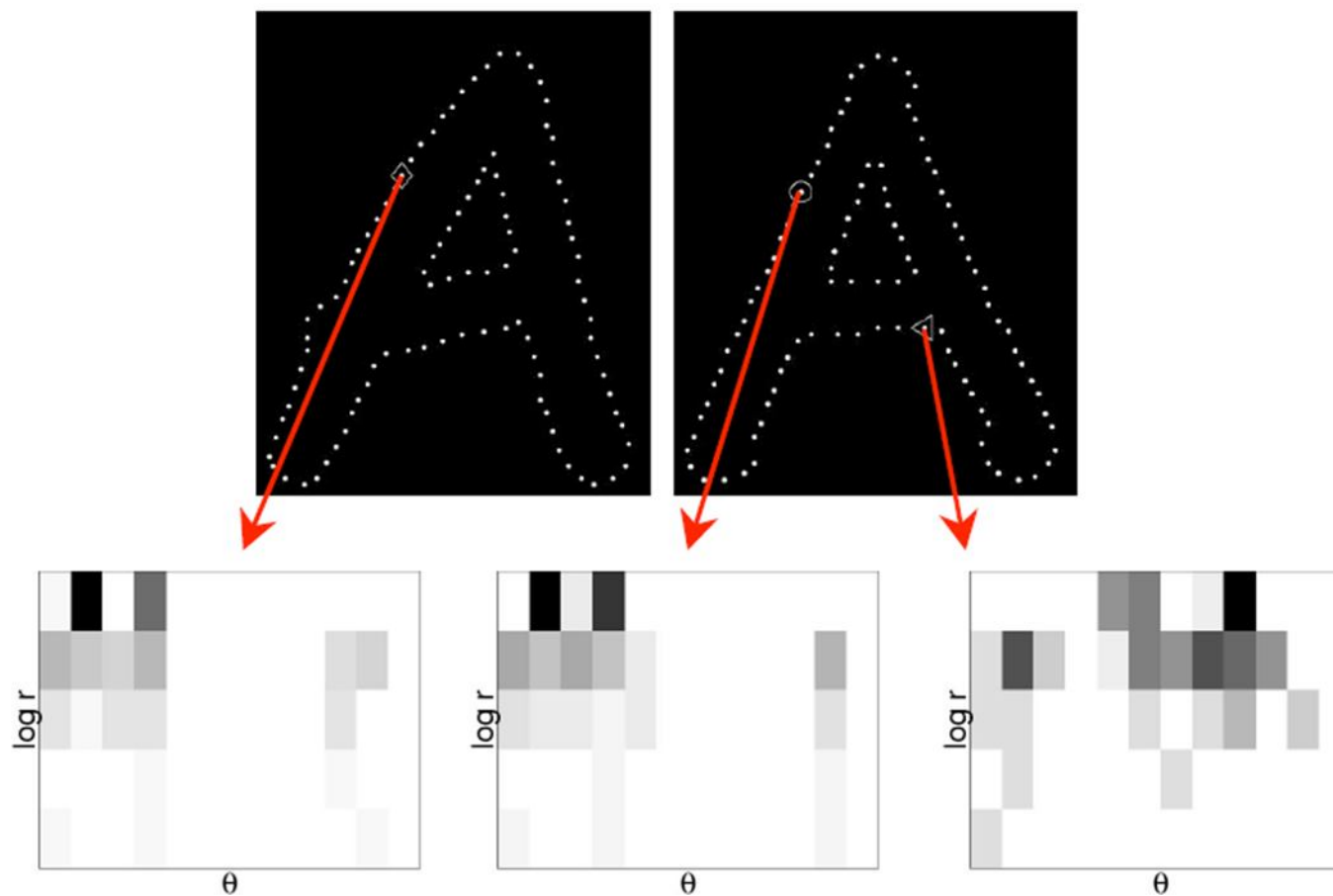
Дескрипторы – контекст формы

1. Вокруг интересной точки строится круглая сетка («мишень»)
2. Подсчитываем количество других интересных точек, попавших в окрестность точки





Дескрипторы – контекст формы



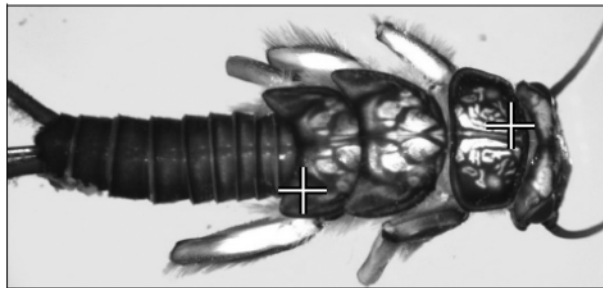
Сходство!



Дескрипторы – глобальный контекст



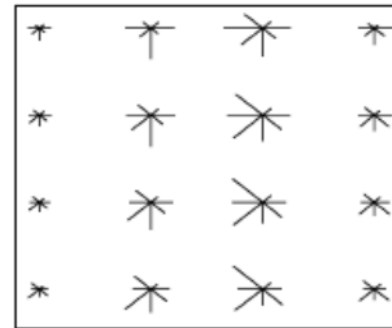
(a)



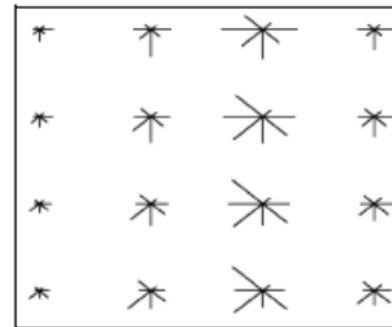
(b)



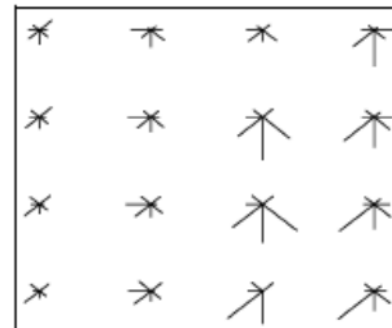
(c)



(d)

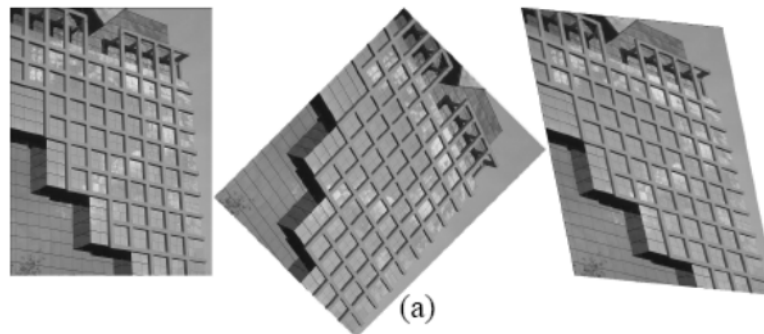


(e)

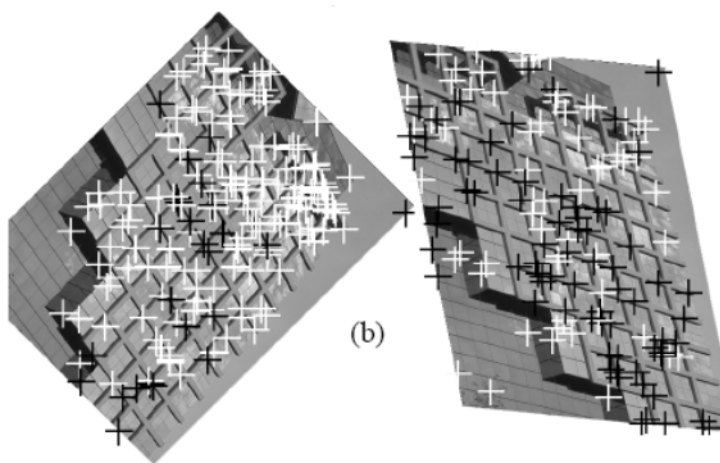




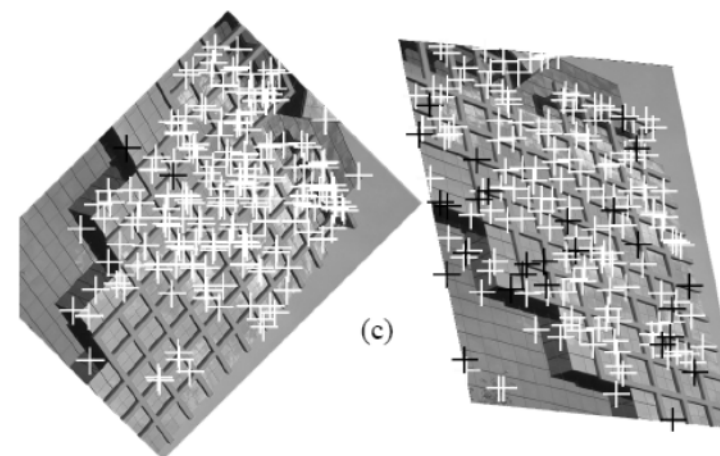
Дескрипторы – глобальный контекст



Ищем соответствия между снимками (1) и (2, 3)



Только SIFT: 85% и 37%
верных соответствий

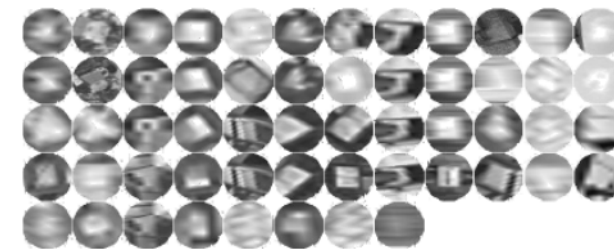
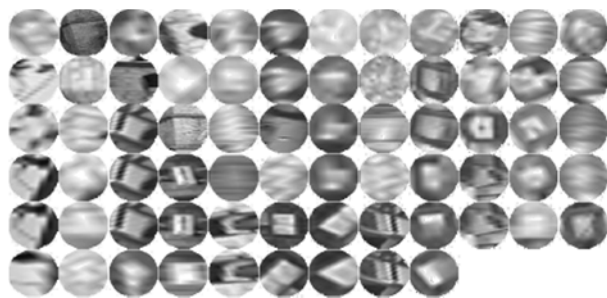


SIFT+Global context: 99% и
83% верных соответствий



Сопоставление

- Имеем набор точек и дескрипторов
- Как будем сопоставлять?





Сопоставление особенностей

- Генерируем пары-кандидаты: для каждого патча в одном изображении, находим несколько наиболее похожих по выбранной метрике патчей на другом изображении
- Как выбирать пары?
 - Полный перебор
 - Для каждой особенности вычисляем расстояния до всех особенностей второго изображения и берем лучшую
 - Ускоренные приближенные меры
 - Иерархические структуры (kd-trees, vocabulary trees)
 - Хэширование



Метрики

- Стандартные L1, L2
- Специальные для гистограмм:

- Пересечение гистограмм $D(h_1, h_2) = \sum_{i=1}^N \min(h_1(i), h_2(i))$

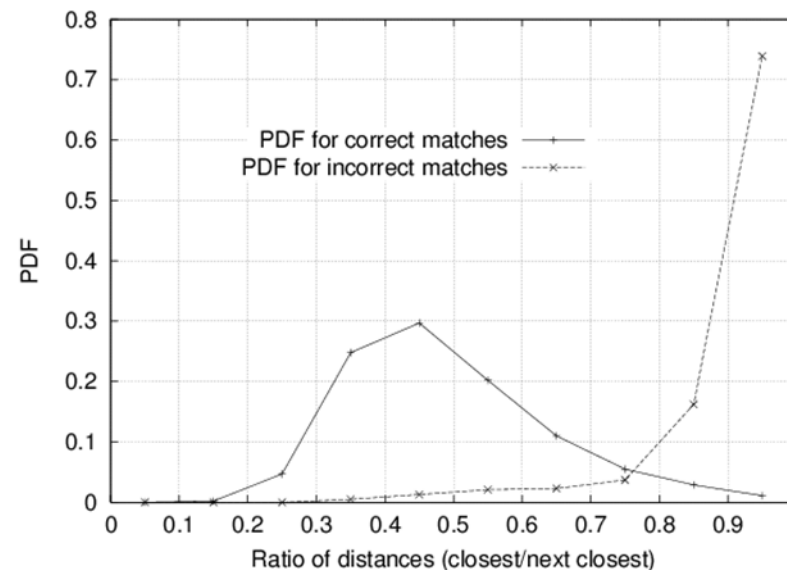
- Расстояние χ^2 $D(h_1, h_2) = \sum_{i=1}^N \frac{(h_1(i) - h_2(i))^2}{h_1(i) + h_2(i)}$

- Earth-Mover Distance



Фильтрация выбросов

- Как можно отбраковать выбросы?
- Эвристика: сравнить расстояния до ближайшей особенности с второй ближайшей по метрики
 - Отношения будет большим для не очень «выделенных» особенностей
 - Порог 0.8 дает неплохое разделение



David G. Lowe. ["Distinctive image features from scale-invariant keypoints."](#) *IJCV* 60 (2), pp. 91-110, 2004.



Расстояние MND

(MND = mutual neighbor distance)

Если y – ближайший к x элемент (сосед), то $NN(\mathbf{x}, \mathbf{y}) = 1$

Если y – следующий по близости x элемент, то $NN(\mathbf{x}, \mathbf{y}) = 2$

Если y – k -й по близости x элемент, то $NN(\mathbf{x}, \mathbf{y}) = k$

Эта мера несимметрична! Т.е., вообще говоря, $NN(\mathbf{x}, \mathbf{y}) \neq NN(\mathbf{y}, \mathbf{x})$

Однако можно использовать меру, которая симметрична:

$$MND(\mathbf{x}, \mathbf{y}) = NN(\mathbf{x}, \mathbf{y}) + NN(\mathbf{y}, \mathbf{x})$$

«Близость» определяется по «обычному» расстоянию, напр. Евклидову.



Резюме лекции

- Точечные особенности (интересные точки)
 - Базовый примитив для анализа изображений
 - Harris, LoG, DoG, Harris-Laplace
- Детекторы
 - SIFT, PCA-SIFT, C-SIFT
 - Контекст формы, глобальный контекст
- Сопоставление по дескрипторам
 - Выбор метрики (L1, L2, ...)
 - Стратегия перебора вариантов
 - Эвристики для фильтрации ложных



На следующей лекции

- Робастные методы оценки параметров моделей
 - M-оценки
 - Рандомизированные методы
 - Схемы голосования, преобразование Хафа
- Поиск линий, окружностей
- Сопоставление особенностей