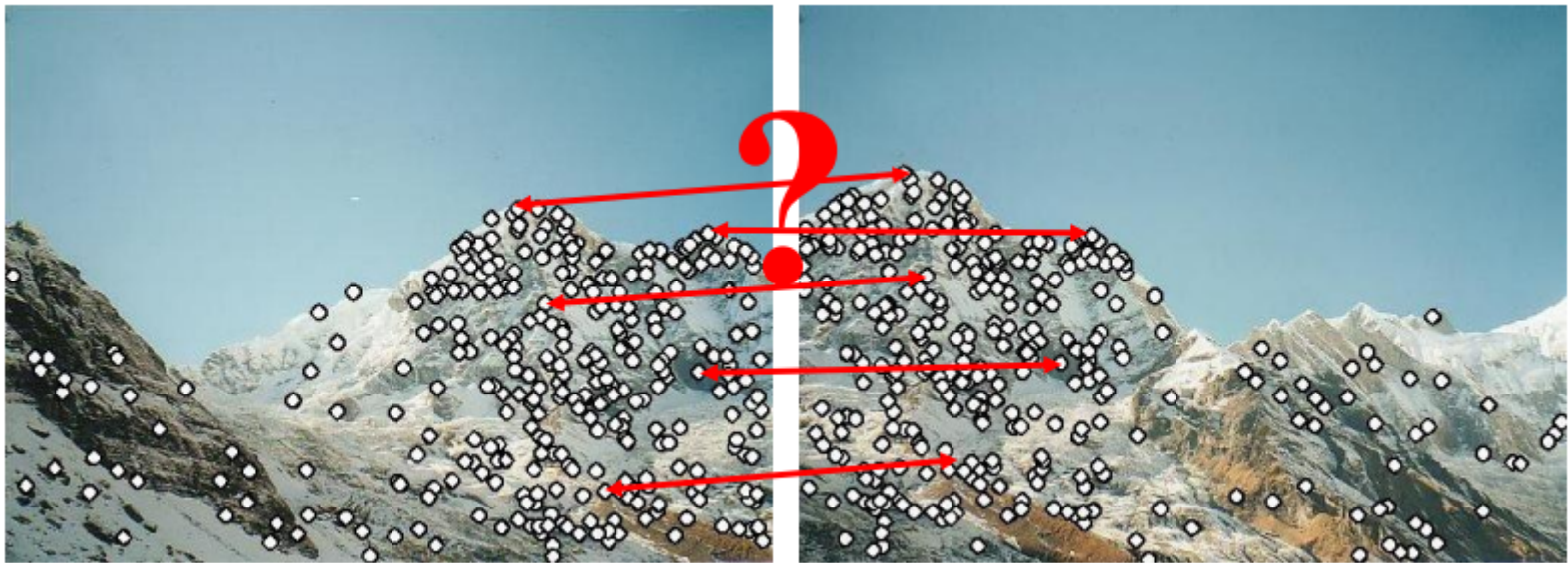


# Выделение и сопоставление особых точек в обработке изображений



Александр Мордвинцев, СПбГУ ИТМО, НИИ НКТ

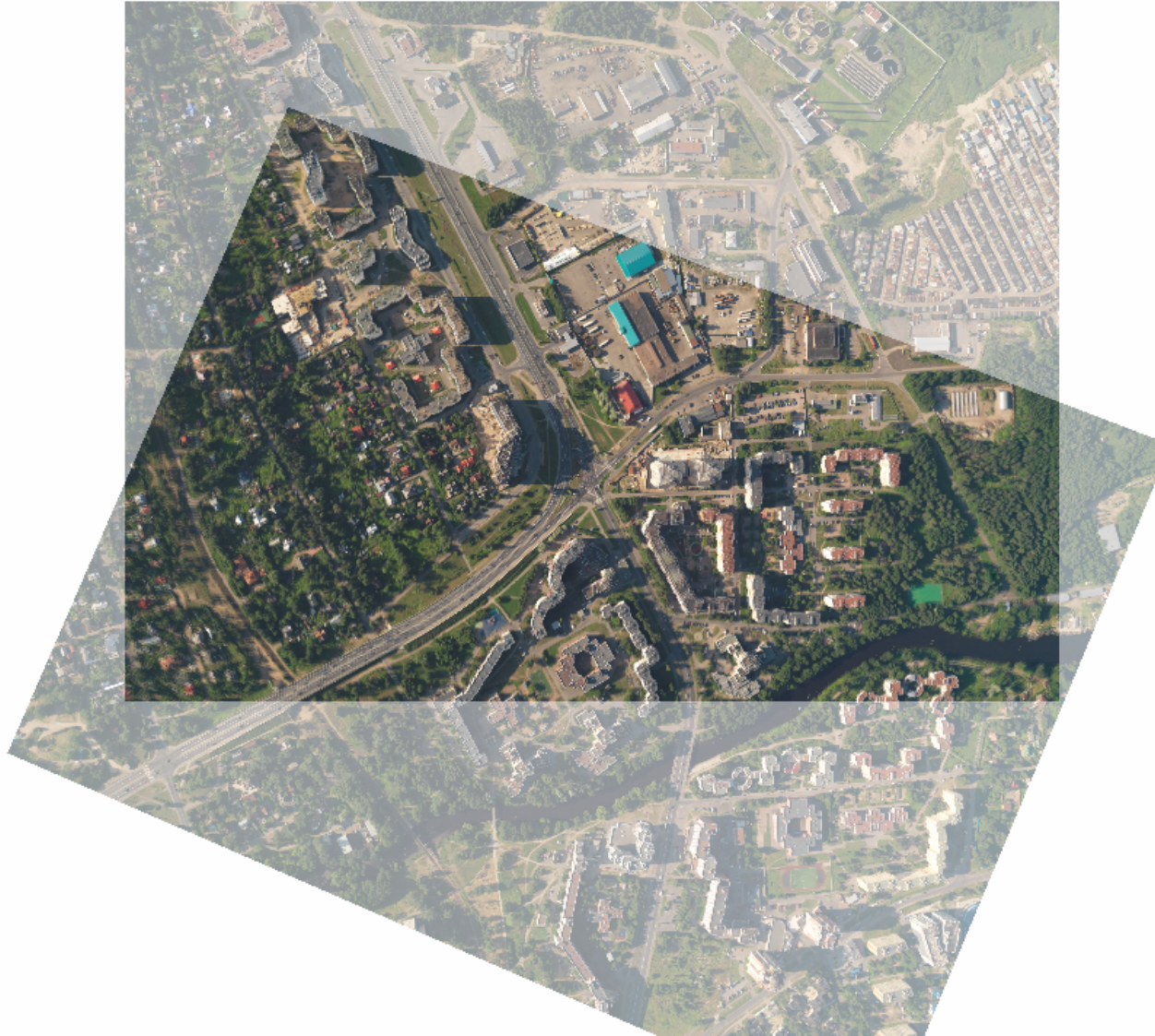
[zzznah@gmail.com](mailto:zzznah@gmail.com)

<http://znah.net>

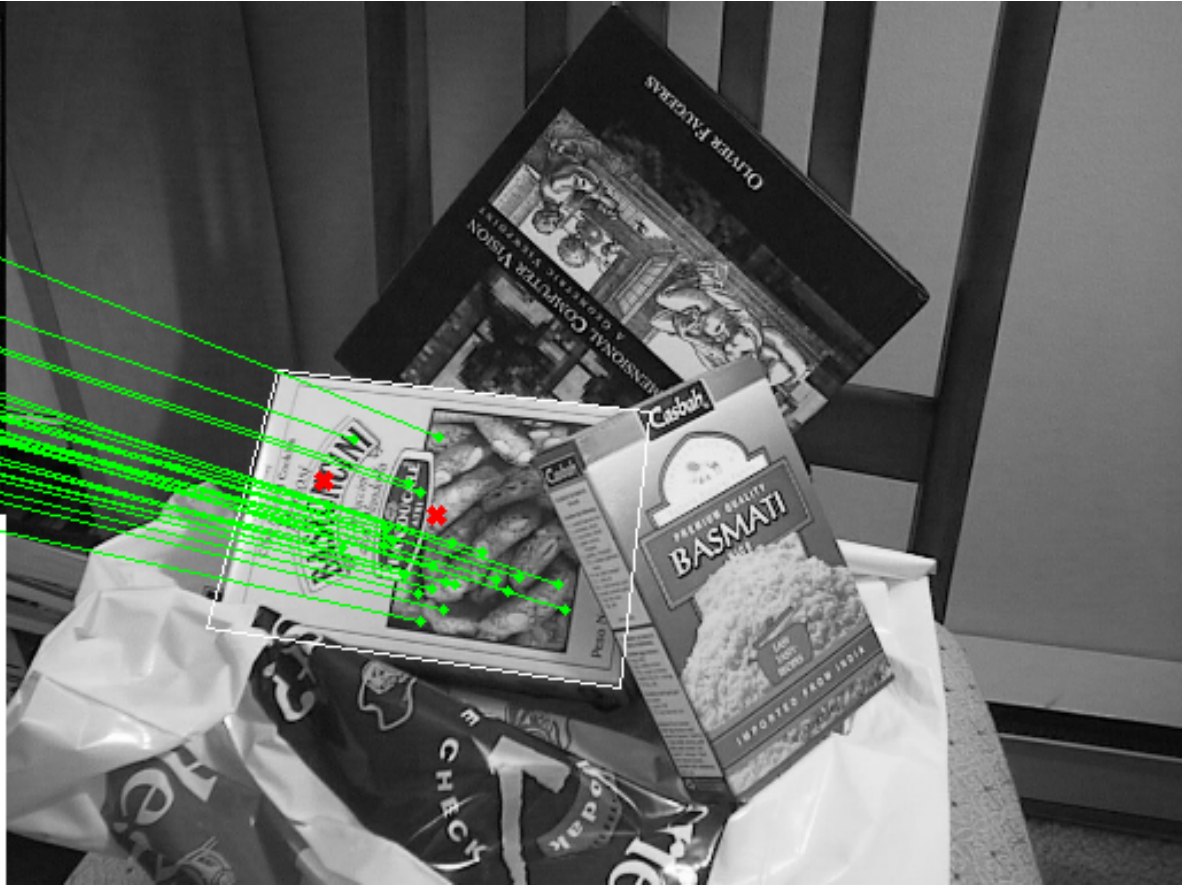
# Мотивация



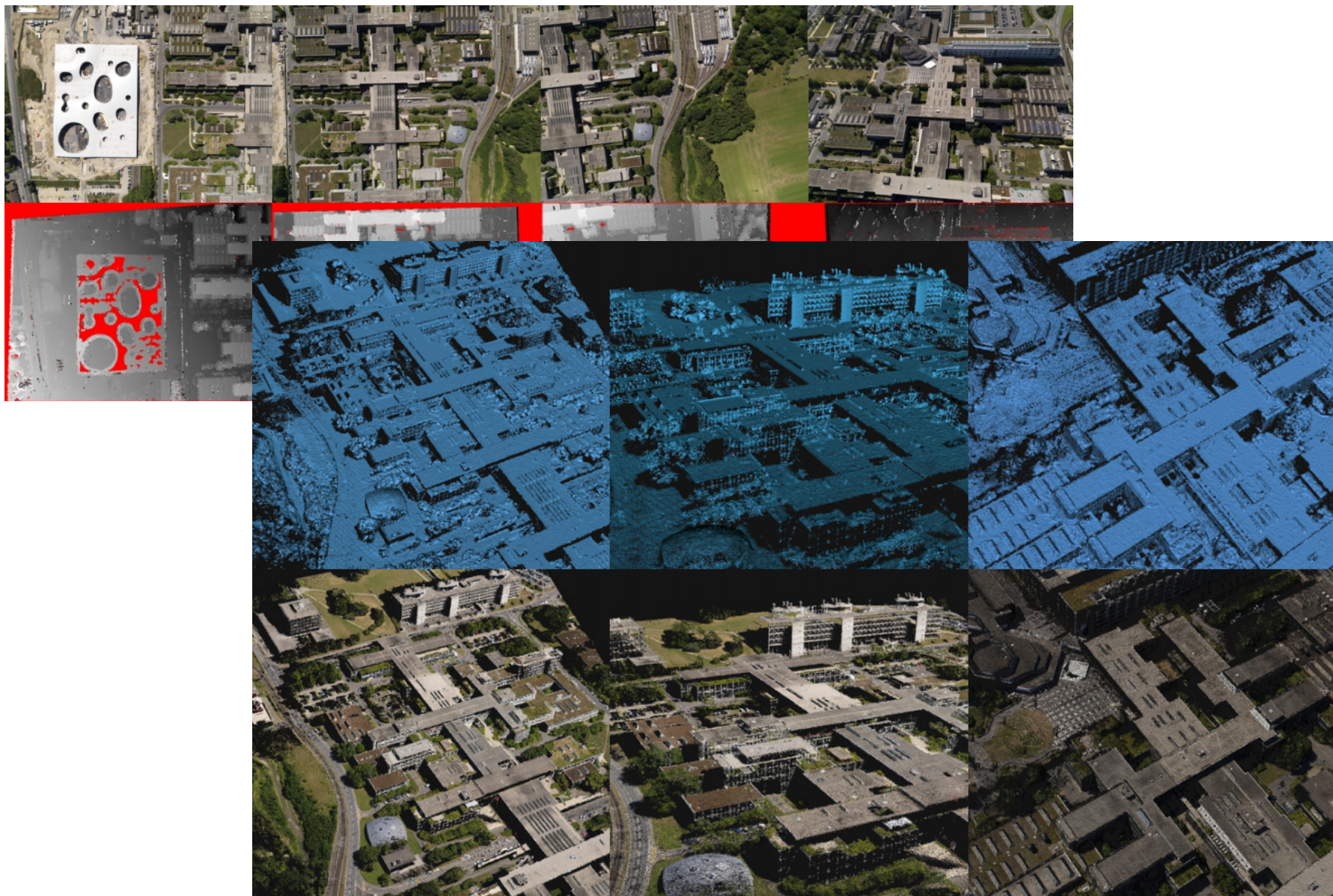
# Мотивация



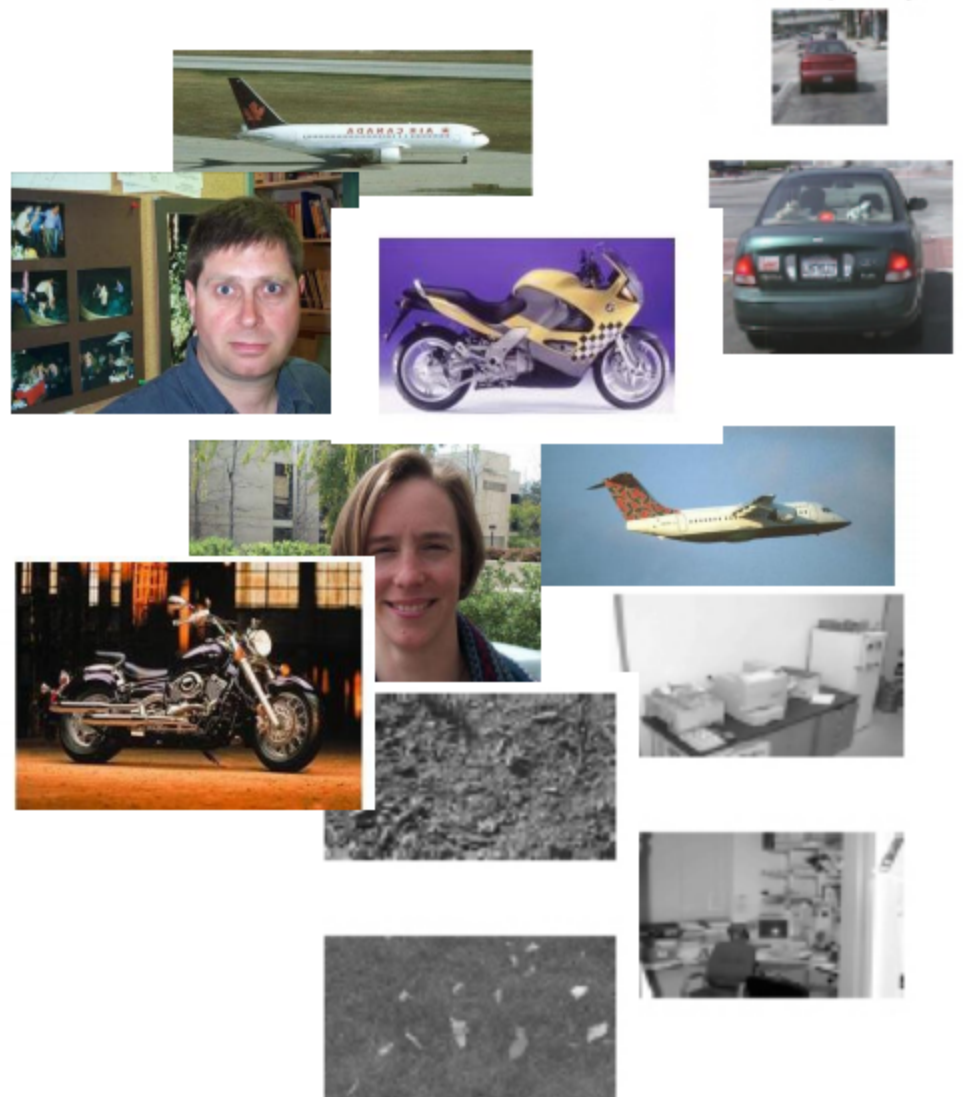
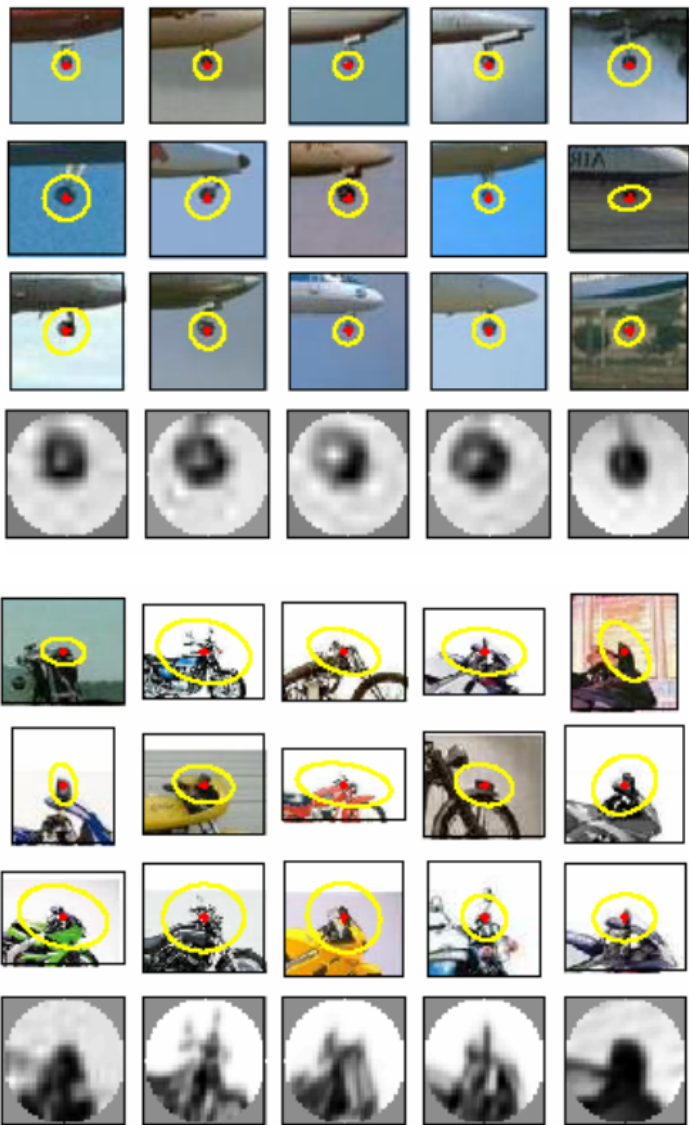
# Мотивация



# Мотивация



# Мотивация



# План

- Выделение особых точек (feature point detection)
  - Harris, LoG, DoG, MSER, FAST ...
- Построение дескрипторов (feature description)
  - SIFT, SURF, BRIEF, DAISY ...
- Feature matching
- Applications

# Поиск особых точек

- Повторяемость
  - Детектор находит одни и те же точки на разных кадрах
- Эффективность
  - Особых точек значительно меньше, чем пикселей



# Поиск особых точек

## традиционный подход



Функция  
«особенности»

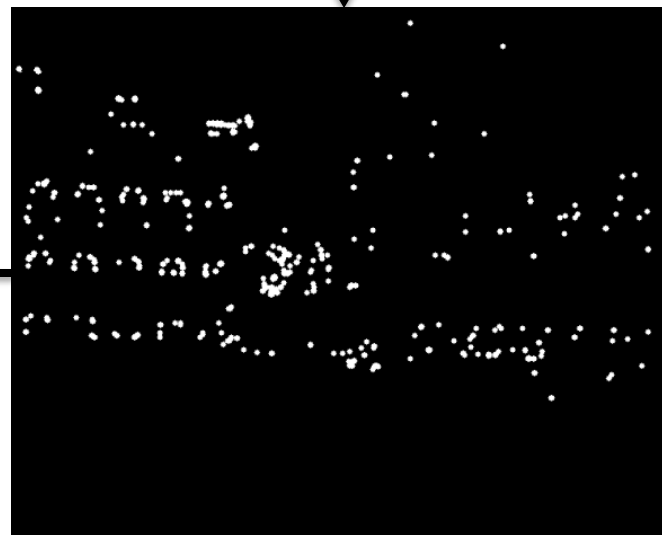
Saliency function



↓ Порог

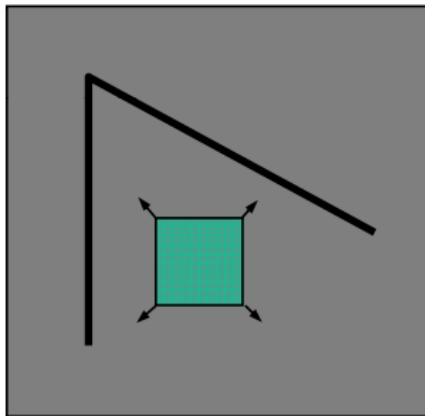


Локальные  
максимумы

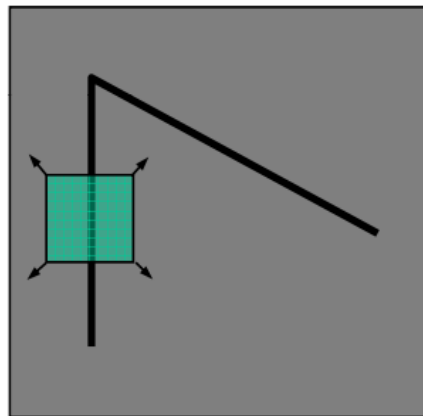


# Локальная особенность: угол

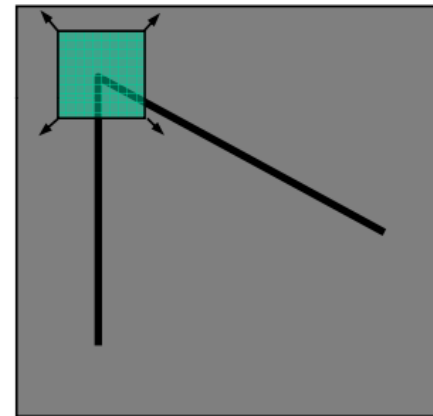
- Ищем точки, окрестность которых сильно изменяется при смещении в любом направлении



Монотонный  
регион (-)



Край (-)



Угол (+)

# Детектор углов Харриса

## Harris Corner Detector

Окно  
(напр. Гауссиан)

Яркость точки  
окрестности

Яркость  
смещенной точки

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

$$\approx [u, v] M \begin{bmatrix} u \\ v \end{bmatrix} \quad M = \sum_{x, y} w(x, y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}$$

Аппроксимация  $E$   
квадратичной формой

Матрица моментов,  
структурный тензор...

Матрица произведений  
производных в точке  $(x, y)$

# Детектор углов Харриса

## функции “особенности”

- $\lambda_1$  и  $\lambda_2$  – собственные числа матрицы  $M$
- Инвариантны к повороту
- Частично инвариантны к перемене освещенности

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

$$R_{\text{Brown}} = \frac{\det M}{\text{trace } M}$$

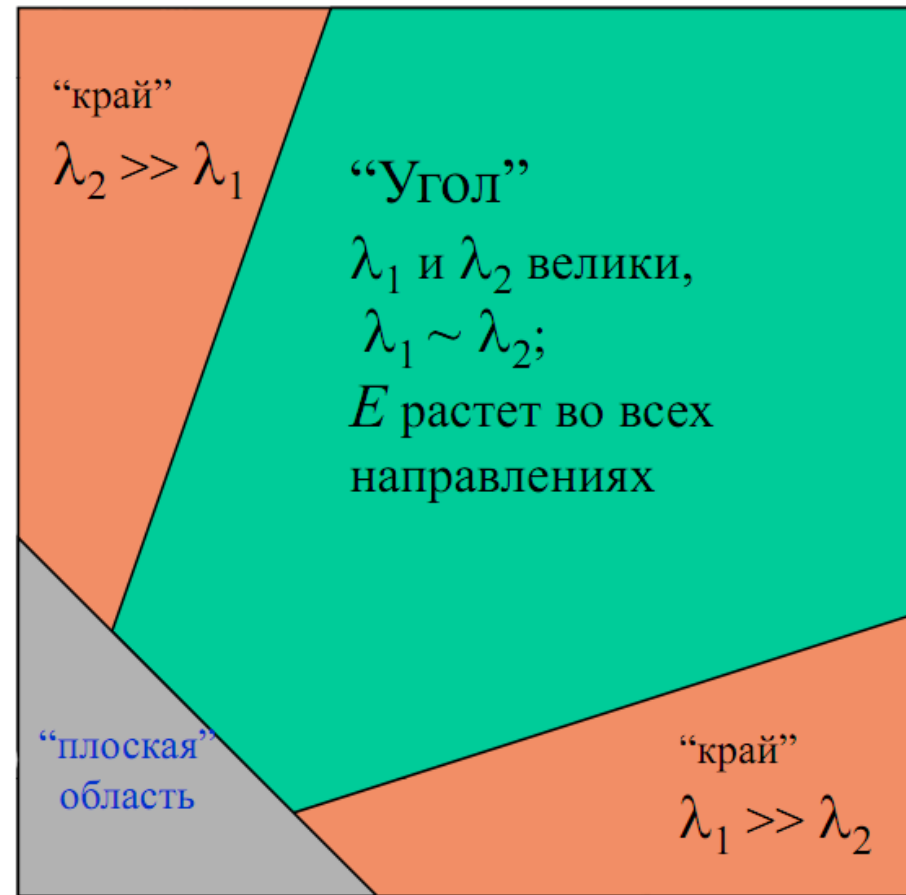
$$R_{\text{Shi}} = \min(\lambda_1, \lambda_2)$$

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

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

$$k = 0.04 \dots 0.06$$

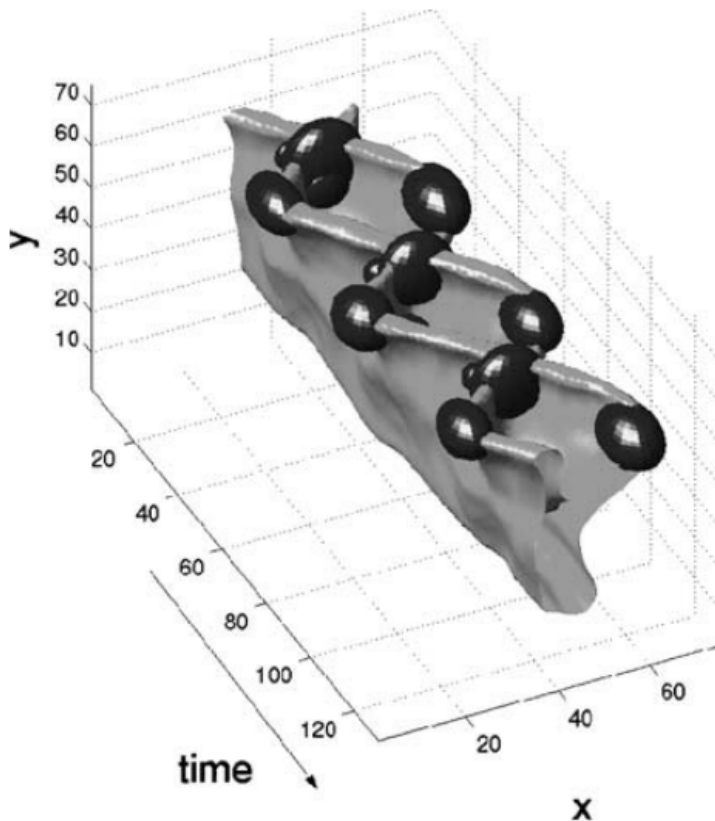
$\lambda_2$



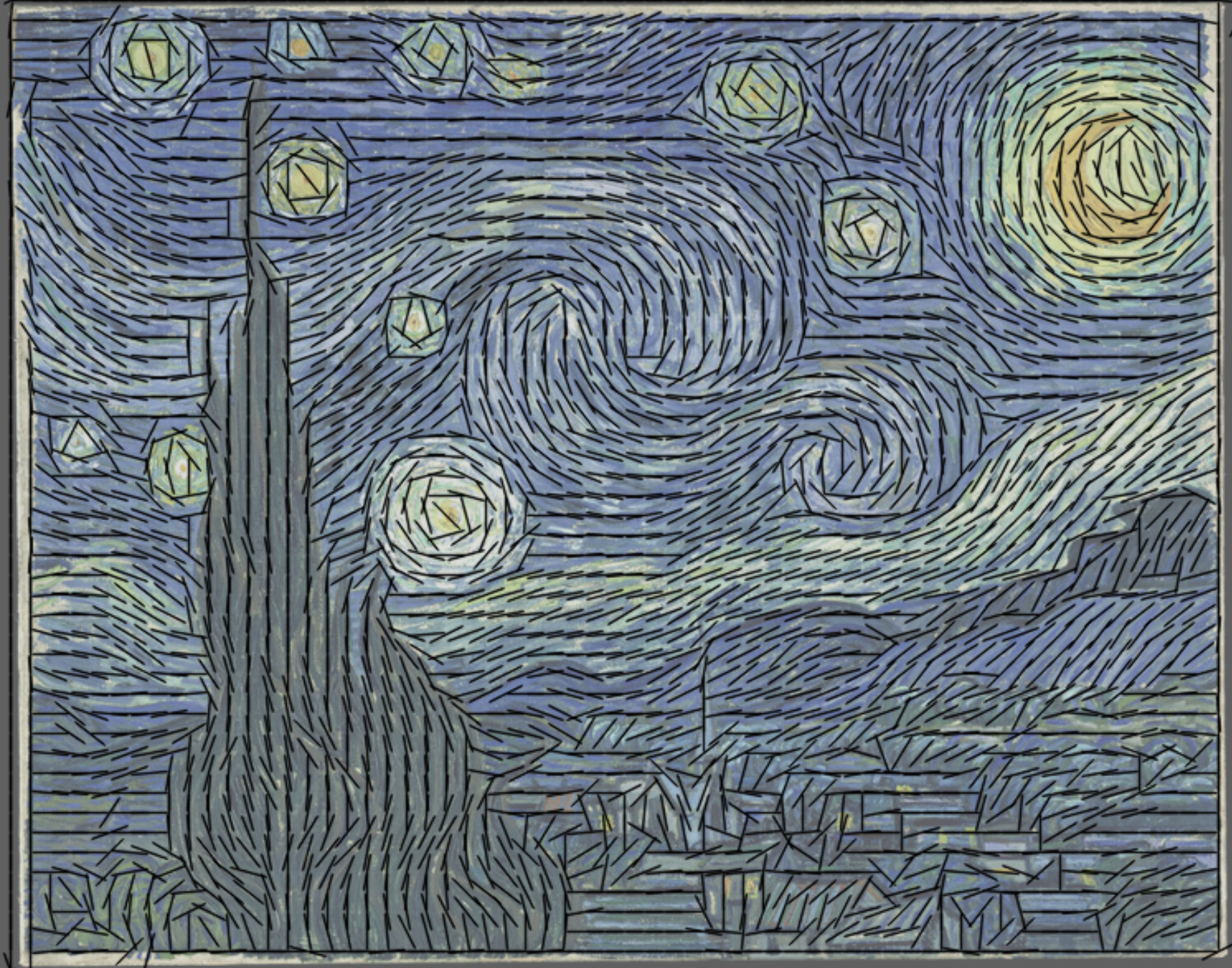
$\lambda_1$

# Структурный тензор

- Второй собственный вектор матрицы  $M$  указывает направление текстуры
- Обобщается на трехмерное пространство

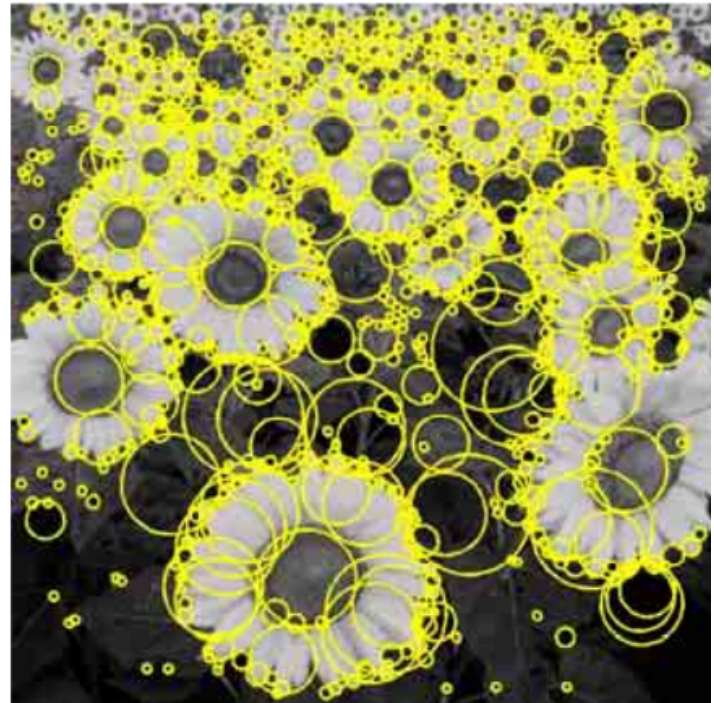
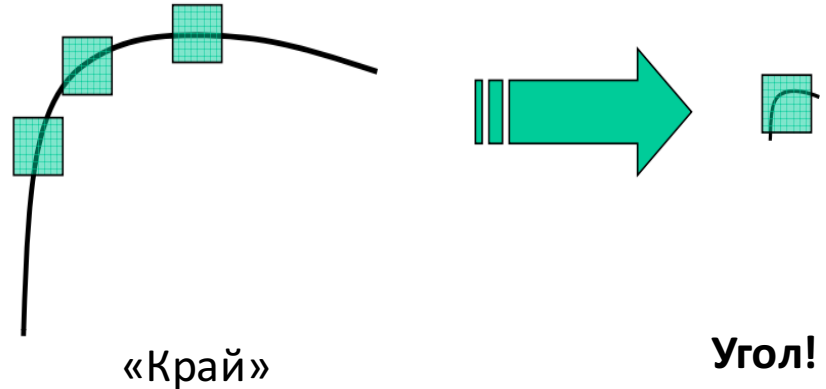






# Выбор масштаба

- Проблема: детектор Харриса не инвариантен к масштабу изображения
- Необходим выбор масштаба
- Локальная особенность: капля (blob)
  - Более яркое (или темное), чем фон, пятно на изображении

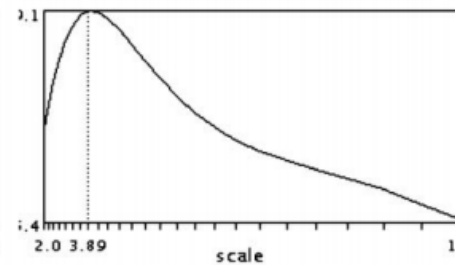
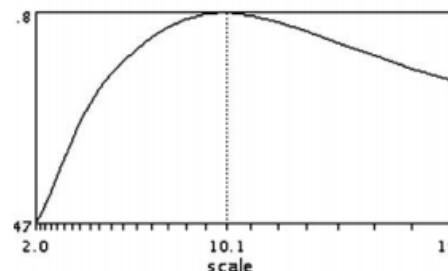
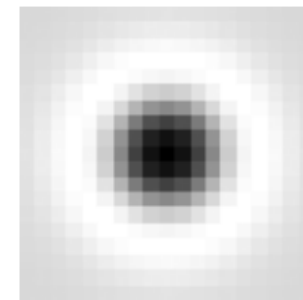
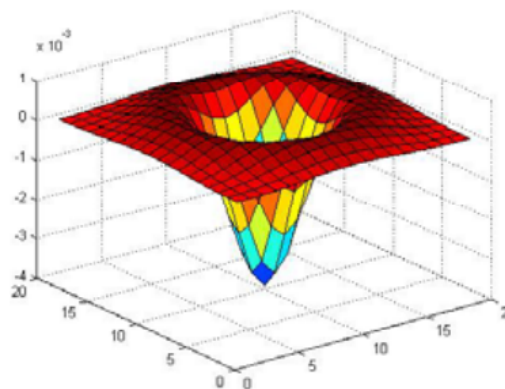




# Лапласиан Гауссиана

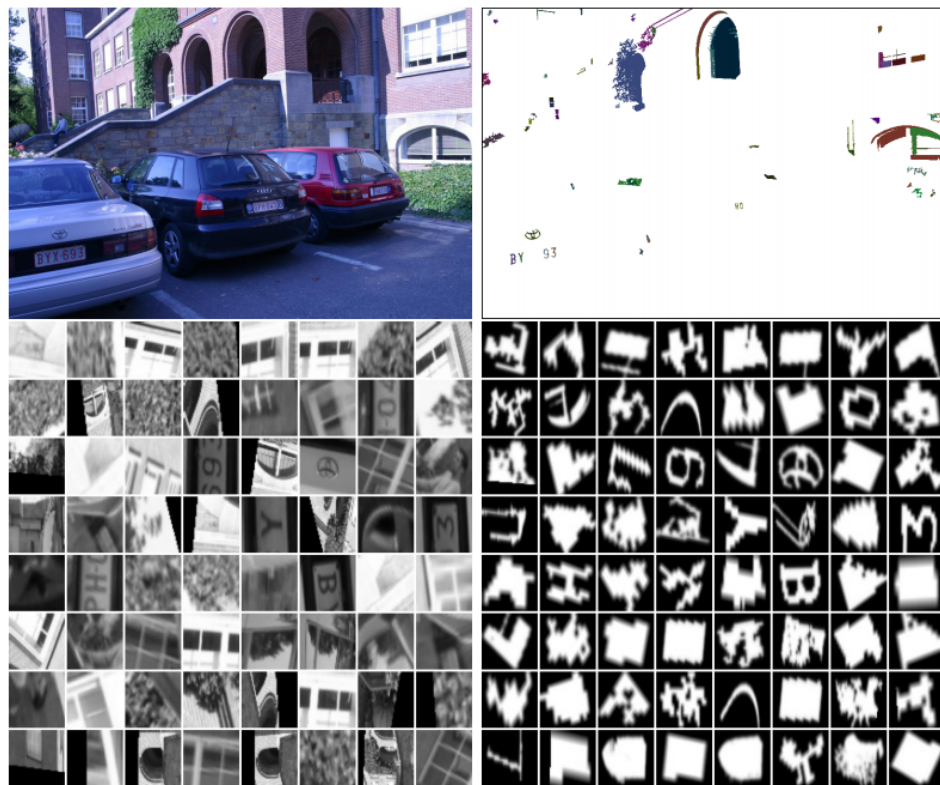
$$\text{LoG}(\mathbf{x}, \sigma) = \sigma^2 \left| \nabla^2 (g_\sigma * I(\mathbf{x})) \right|$$

- Положение локальных максимумов LoG определяет положение и масштаб особой точки
- Комбинируется с детектором Харриса для вычисления масштаба углов
- Аппроксимации используются в детекторах SIFT, SURF и многих других

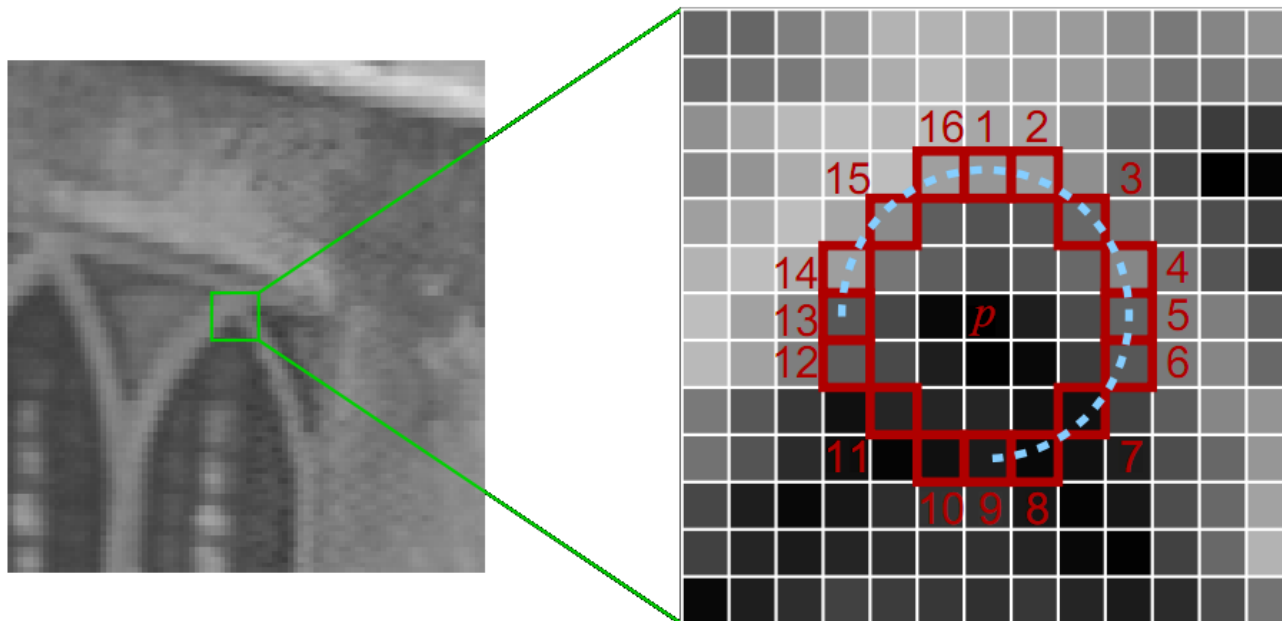


# Детектор областей MSER

- MSER - Maximally Stable Extremal Region
- Устойчивые к бинаризации с различным порогом участки изображения
- Эффективно реализуется при помощи системы непересекающихся множеств
- Инвариантен к аффинным преобразованиям



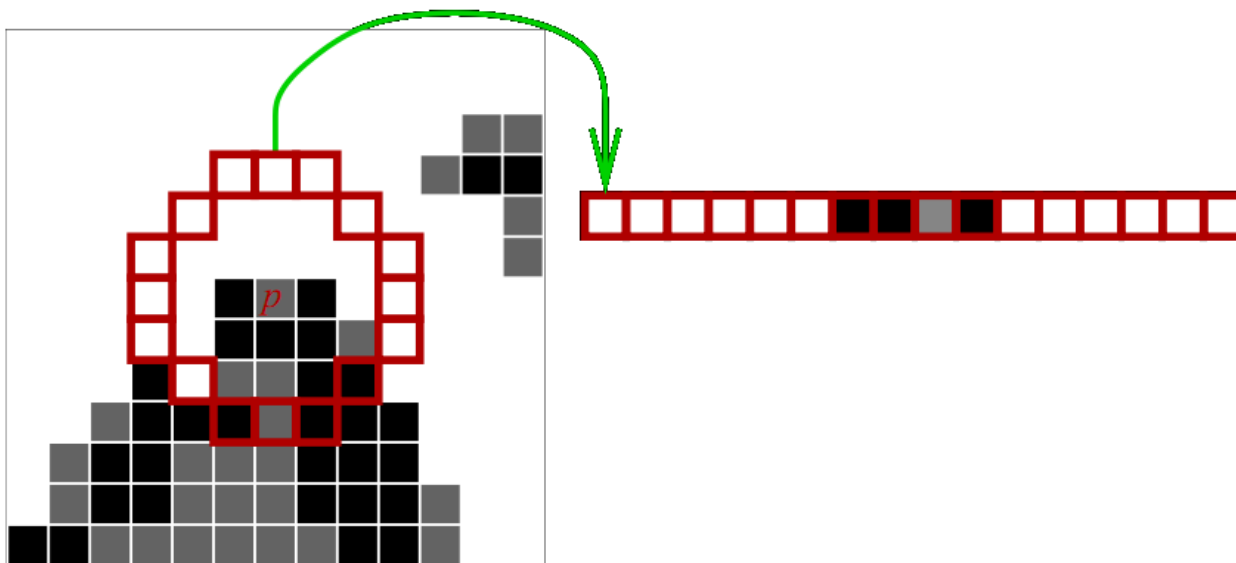
# Детектор FAST: идея



- Точка особенная, если в на кольце радиуса  $r$  есть дуга из  $N$  последовательных пикселей, все из которых
  - Значительно темнее  $p$
  - Значительно светлее  $p$

# Детектор FAST: реализация

- Классифицируем пиксели кольца по порогу  $t$  на светлые, темные и серые. Окрестность точки описывается тренируемым вектором
- Строим решающее дерево для классификации векторов на особые/не особые
- Результат - >4000 строк вложенных if-else
- Но в среднем всего **2.26** сравнений на пиксель для  $r = 3$ ,  $N=9$ . Работает очень быстро



# Дескрипторы особенностей

- Описывают окрестность каждой особой точки набором параметров
- Должны быть:
  - специфичны
  - локальны
  - устойчивы
  - просты в вычислении
  - иметь адекватную метрику



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

# Дескрипторы особенностей

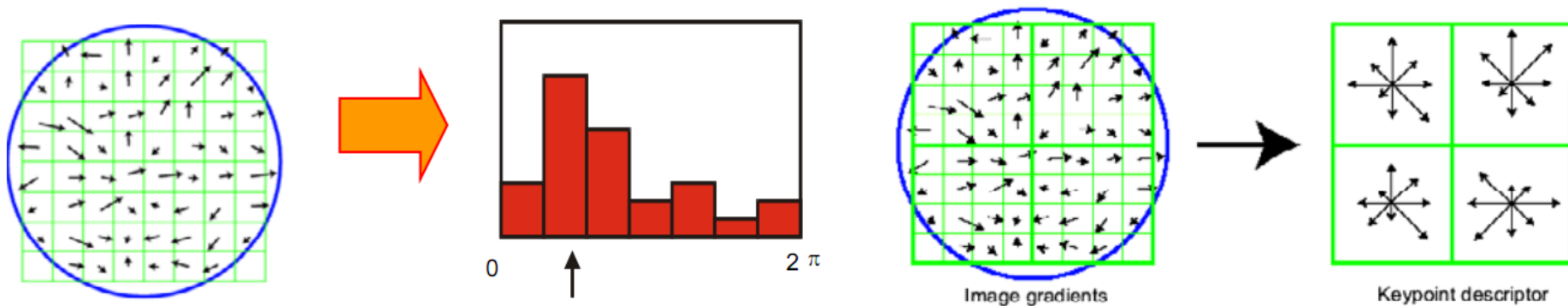
Простейшие

- (не)ориентированные окна



# Гистограммы градиентов (SIFT)

- Ориентация точки (поиск доминирующего градиента)
- Бьем ориентированную окрестность регулярной сеткой (обычно 4x4)
- Строим гистограмму градиентов, попавших в каждую ячейку (обычно 8 бинов)
- Получаем вектор из  $4*4*8=128$ . Нормализуем. Это и есть дескриптор SIFT

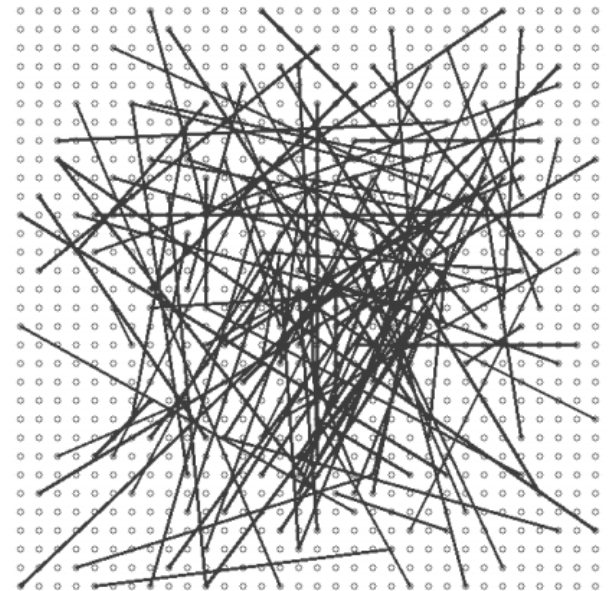


# Дескриптор BRIEF

- При разработке дескриптора генерируем случайный набор пар двухмерных векторов смещений  $(x_i, x'_i)$
- Дескриптор точки  $\mathbf{p}$   $b(\mathbf{p})$  - вектор из  $N=128...256$  булевых значений

$$b_i(\mathbf{p}) = \begin{cases} 0, & \text{если } I(\mathbf{p} + \mathbf{x}_i) > I(\mathbf{p} + \mathbf{x}'_i) \\ 1, & \text{если } I(\mathbf{p} + \mathbf{x}_i) \leq I(\mathbf{p} + \mathbf{x}'_i) \end{cases}$$

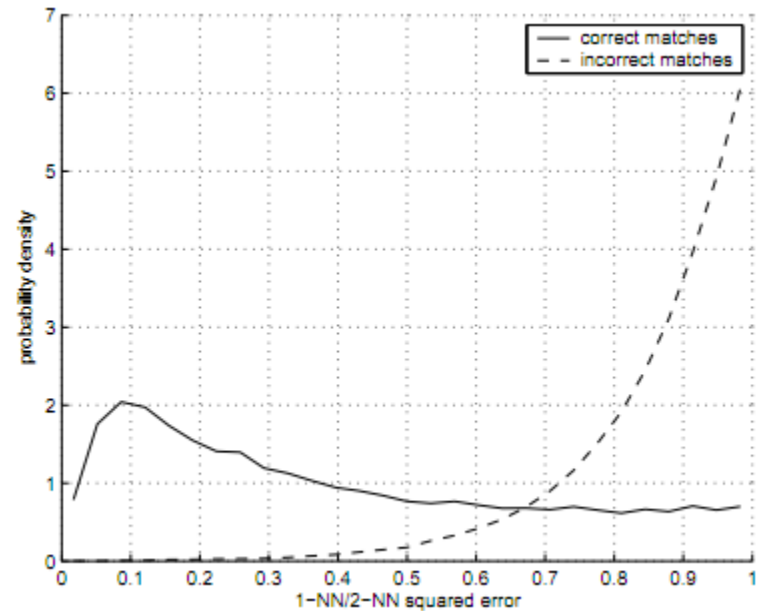
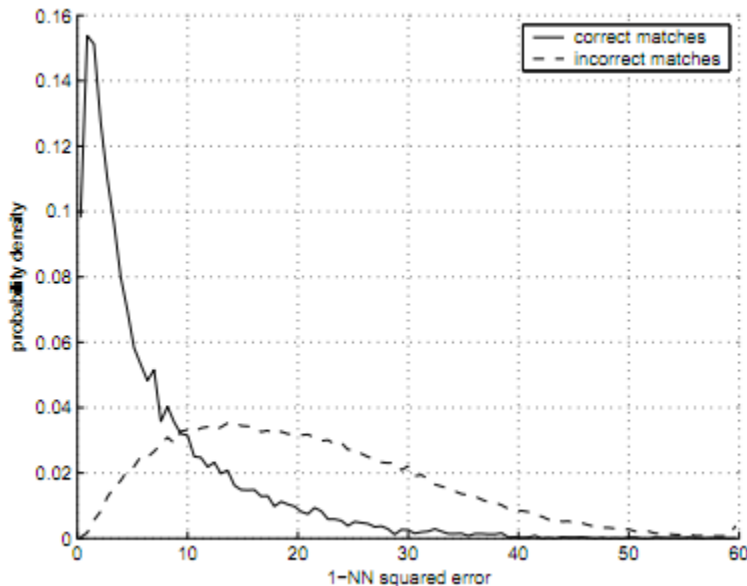
- Метрика Хемминга для сравнения дескрипторов



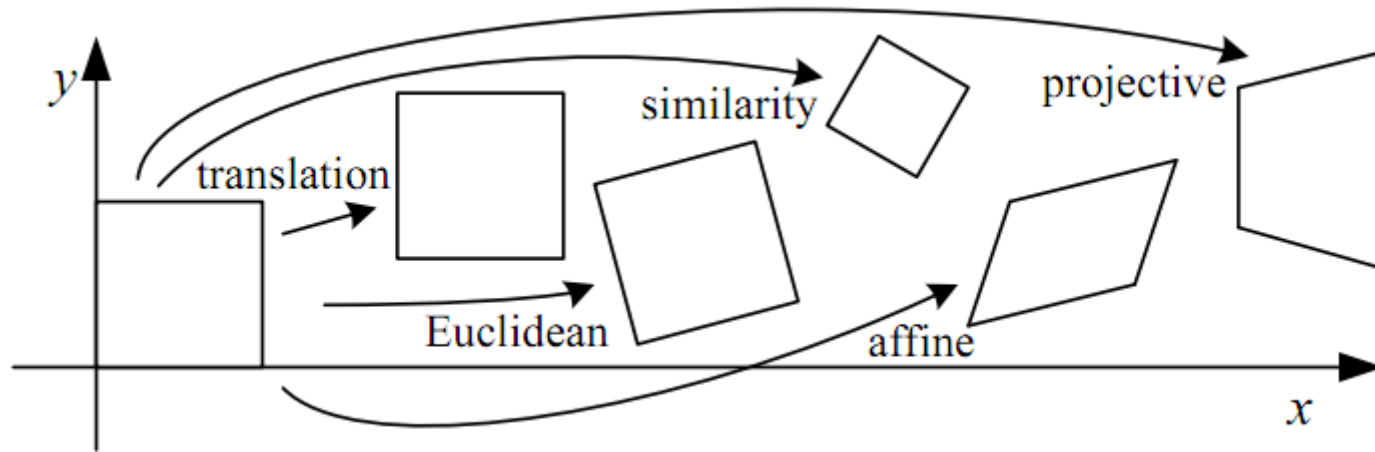


# Сравнение дескрипторов

- Имеем два набора дескрипторов, хотим найти соответствия
  - Критерий 1-NN distance
  - 1-NN / 2-NN distance
- Используем вероятностный индекс для поиска ближайших соседей в многомерном пространстве (FLANN)

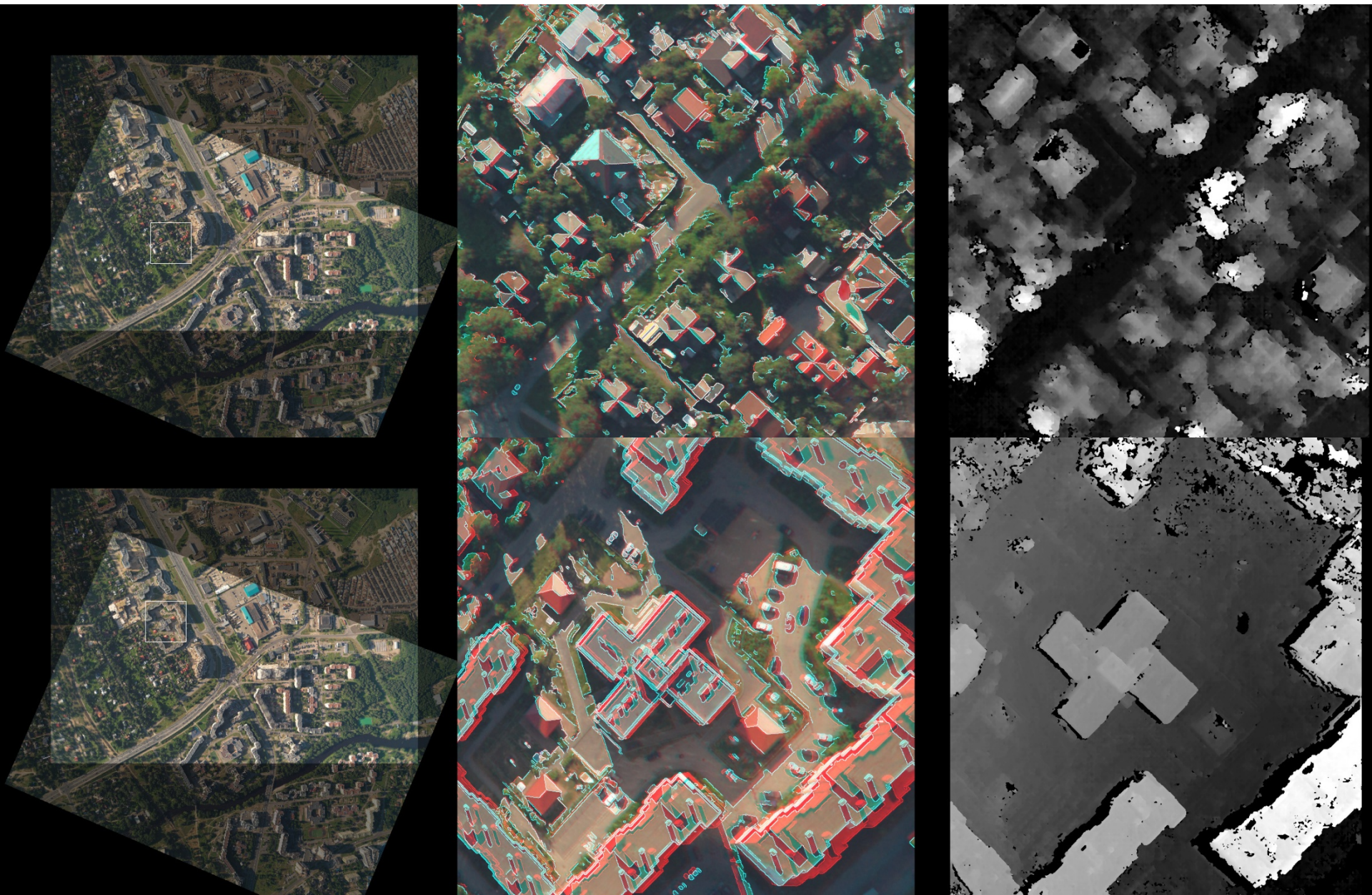


# Регистрация изображений



- Модель трансформации между изображениями
- RANSAC

# Приложения, демонстрации.



# ССЫЛКИ

- Richard Szeliski *Computer Vision: Algorithms and Applications* (<http://szeliski.org/Book/>)
- <http://courses.graphicon.ru/main/vision/2011/lectures>
- Feature detectors
  - <http://www.robots.ox.ac.uk/~vgg/research/affine>
  - <http://www.edwardrosten.com/work/fast.html>
  - [Space-time interest points](#)
- Descriptors
  - <http://www.cs.ubc.ca/~lowe/keypoints/>
  - <http://www.vision.ee.ethz.ch/~surf/>
  - (BRIEF) <http://cvlab.epfl.ch/~lepetit/>
  - <http://www.cmap.polytechnique.fr/~yu/research/ASIFT/demo.html>
  - <http://cvlab.epfl.ch/research/detect/ldahash/>
- Applications
  - [http://www.robots.ox.ac.uk/~bob/research/research\\_ptamm.html](http://www.robots.ox.ac.uk/~bob/research/research_ptamm.html)
  - <http://phototour.cs.washington.edu/bundler/>
  - <http://pix4d.com/>