

Компьютерное зрение и обработка изображений

Лекция 12

Детекторы и дескрипторы

Я.М.Демяненко
demayna@sfedu.ru

Южный федеральный университет
Институт математики, механики и компьютерных наук

2019

Содержание

- 1 Сопоставление изображений
 - Различные подходы
 - Что такое ключевая особенность?
- 2 Особые точки
 - Свойства особых точек
 - Детекторы углов
- 3 Детекторы и дескрипторы
 - SURF

Содержание

- 1 Сопоставление изображений
 - Различные подходы
 - Что такое ключевая особенность?
- 2 Особые точки
 - Свойства особых точек
 - Детекторы углов
- 3 Детекторы и дескрипторы
 - SURF

Контурный анализ

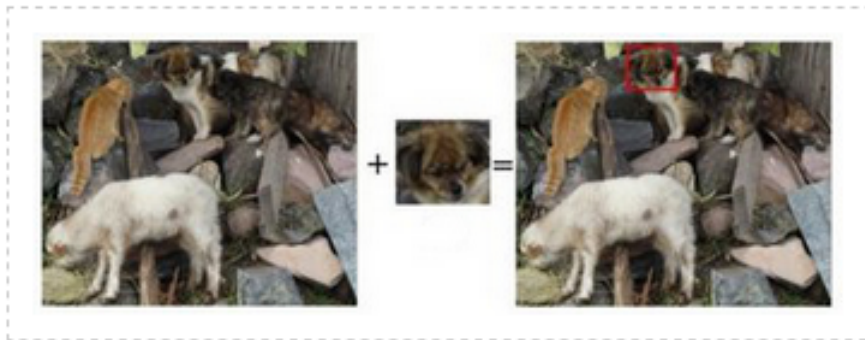
BRIDGESTONE
POTENZA



Контурный анализ: ограничения

- при одинаковой яркости с фоном объект может не иметь чёткой границы, или может быть зашумлён помехами, что приводит к невозможности выделения контура;
- перекрытие объектов или их группировка приводит к тому, что контур выделяется неправильно и не соответствует границе объекта.

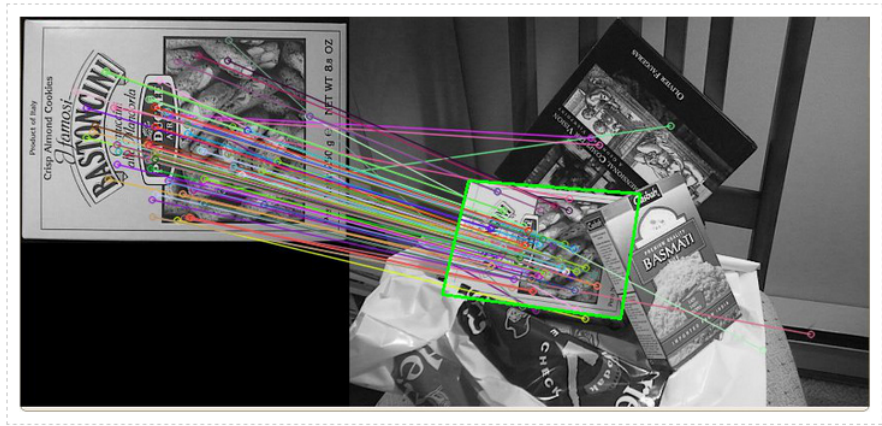
Template matching



Template matching: ограничения

- зависит от масштаба, углов обзора, поворотов картинки и наличия физических помех;
- возможны ложные срабатывания алгоритма, когда искомого объекта на самом деле нет, но имеются какие-то общие детали у шаблона и области на тестируемом изображении.

Feature Detection



Feature Detection

- нацелен на вычисление абстракций изображения и выделения на нем ключевых особенностей;
- не существует строго определения того, что такое ключевая особенность картинки.

Что такое ключевая особенность?

- изолированные точки
- кривые или некоторые связанные области
- грани объектов
- углы

Feature points

- Под ключевыми точками понимаются некоторые участки картинки, которые являются отличительными для данного изображения.
- Подобные точки каждый алгоритм определяет по своему.

Составляющие для нахождения ключевых точек

- Детектор (feature detector) — осуществляет поиск ключевых точек на изображении.
- Дескриптор (descriptor extractor) — производит описание найденных ключевых точек, оценивая их позиции через описание окружающих областей.
- Матчер (matcher) — осуществляет построение соответствий между двумя наборами точек изображений.

Содержание

- 1 Сопоставление изображений
 - Различные подходы
 - Что такое ключевая особенность?
- 2 **Особые точки**
 - Свойства особых точек
 - Детекторы углов
- 3 Детекторы и дескрипторы
 - SURF

Определение

- Особая точка m , или точечная особенность (англ. point feature, key point, feature), изображения – это точка изображения, окрестность которой $o(m)$ можно отличить от окрестности любой другой точки изображения $o(n)$ в некоторой другой окрестности особой точки $o_2(m)$.
- В качестве окрестности точки изображения для большинства алгоритмов берётся прямоугольное окно, составляющее размер 5×5 пикселей.

Haralick и Shapiro (1992)



В 1992 Haralick и Shapiro выделили требования к особым точкам

Требования к особым точкам

- 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.

Tuytelaars и Mikolajczyk (2006)



Tuytelaars и Mikolajczyk (2006) выделили свойства особых точек

Свойства особых точек

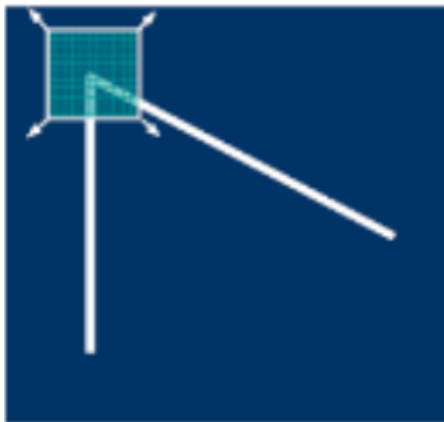
- Количество (quantity) – число обнаруженных особых точек должно быть достаточно большим. Однако оптимальное количество особых точек зависит от предметной области. В идеале количество обнаруженных особых точек должно адаптивно определяться с использованием простого и интуитивного порога. Плотность расположения 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.

Подходы к определению особых точек

- Основанные на интенсивности изображения.
- Использующие контуры изображения: ищут места с максимальным значением кривизны или делают полигональную аппроксимацию контуров и определяют пересечения. Эти методы чувствительны к окрестностям пересечений, поскольку извлечение часто может быть неправильным в тех местах, где пересекаются 3 или более краев.
- На основе использования модели: используются модели с интенсивностью в качестве параметров, которые подстраиваются к изображениям-шаблонам до субпиксельной точности.

Типы локальных особенностей

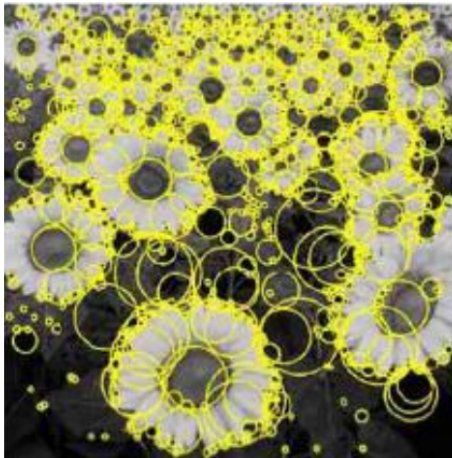
- Углы



- Края – множество точек на изображении, которые имеют высокую величину градиента

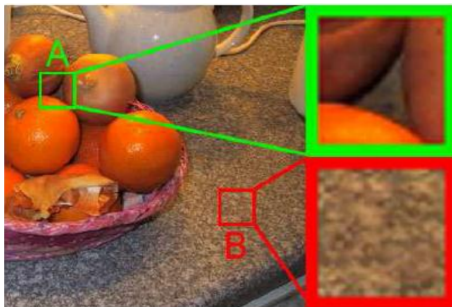
Типы локальных особенностей

- Пятна (Blob)

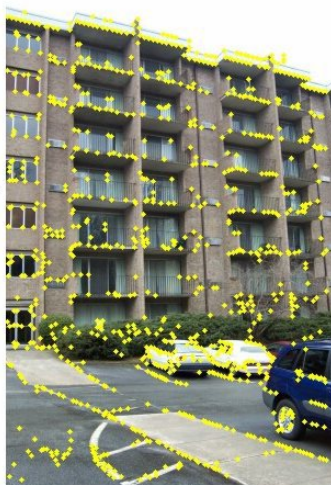
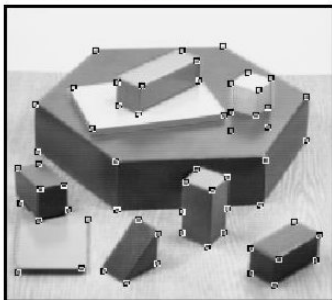


Типы локальных особенностей

- Пятна (Blob)



Особые точки



Виды углов



План

Moravec -> **Harris** -> **Shi-Tomashi**
Susan -> Trajkovic -> **FAST**

Moravec (1977)

- Изменение яркости квадратного окна $W(3 \times 3, 5 \times 5, 7 \times 7)$ относительно интересующей точки при сдвиге на 1 пиксель в 8-ми направлениях.
- Алгоритм:

- Для каждого пикселя (x, y) в изображении вычислить изменение интенсивности

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

- Построить карту вероятности нахождения углов в каждом пикселе (x, y) изображения посредством вычисления оценочной функции

$$C(x,y) = \min(V_{u,v}(x,y))$$

- Отсечь пиксели, в которых значения $C(x, y)$ ниже порогового значения T .
- Удалить повторяющиеся углы (non-maximal suppression).

Moravec (1977)

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

Window function

Shifted intensity

Intensity

Window function $w(x, y) =$



1 in window, 0 outside



flat



edge



corner
isolated point

Moravec (1977)

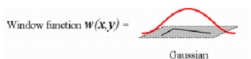
- Недостатки
 - Чувствителен к шуму
 - Рассматриваются сдвиги с шагом в 45 градусов => отсутствие полной инвариантности к преобразованию поворота
 - Возникновение ошибок детектирования при наличии большого количества диагональных ребер

Harris(1988)

- Наиболее оптимальным детектором L-связных углов является широко известный детектор Харриса
- его также называют оператором Плессея, детектором Харриса и Стефенса, Plessey operator, Harris and Stephens detector

Harris(1988)

$$w(x, y) = \exp\left(-\frac{(x^2 + y^2)}{2\sigma^2}\right)$$



$$E(x, y) = \sum_u \sum_v w(u, v) (I(u + x, v + y) - I(u, v))^2$$

$$I(u + x, v + y) \approx I(u, v) + I_x(u, v)x + I_y(u, v)y$$

$$E(x, y) \approx \sum_u \sum_v w(u, v) (I_x(u, v)x + I_y(u, v)y)^2,$$

$$E(x, y) \approx \begin{pmatrix} x & y \end{pmatrix} M \begin{pmatrix} x \\ y \end{pmatrix}, \quad M = \sum_u \sum_v w(u, v) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} = \begin{bmatrix} \langle I_x^2 \rangle & \langle I_x I_y \rangle \\ \langle I_x I_y \rangle & \langle I_y^2 \rangle \end{bmatrix}$$

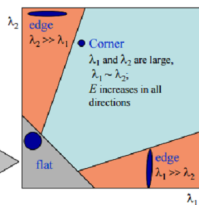
Harris - «лечение» основных недостатков Moravec

Недостатки:

- Чувствителен к шуму

Classification of image points using eigenvalues of M :

λ_1 and λ_2 are small, E is almost constant in all directions



Харрисом и Стефеном была предложена мера отклика

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

где k - эмпирическая константа, $k \in [0.04; 0.06]$

Shi-Tomasi (1993)

Аналогичен алгоритму Harris, за исключение финального шага. В Shi-Tomasi вычисляется значение

$$\min(\lambda_1, \lambda_2)$$

поскольку делается предположение, что поиск углов будет более стабильным.

Förstner (1987)

- Фёрстнер и Гёлч (Förstner and Gülch) первыми описали метод, который использует ту же самую меру угловатости, что и детектор Харриса.
- В отличие от детектора Харриса собственные значения вычисляются явно. Функция отклика угла Фёрстнера определяется следующим образом

$$R = \lambda_1 \lambda_2 / (\lambda_1 + \lambda_2) = \det M / \text{tr} M,$$

Förstner (1987)

- Также для правильности определения считается мера округлости угла, равная

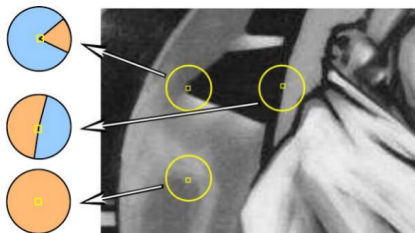
$$1 - \left(\frac{\lambda_1 - \lambda_2}{\lambda_1 + \lambda_2} \right)^2 = 4 \det M / (\text{tr} M)^2$$

- Детектор Фёрстнера на практике часто используется для расширения возможностей детектора Харриса – нахождению круговых особых точек вместе с углами.
- Также алгоритм обладает лучшим свойством локализации

Susan(1997)

- Алгоритм SUSAN (Smallest Univalued Segment Assimilation Nucleus) был предложен Смитом и Бреди (Smith и Brady)

Susan(1997)



Алгоритм:

1. Сравнить интенсивности пикселей внутри круговой маски с ядром
2. Вычислить относительную площадь непохожего участка
3. Угол = площадь < порога
4. Отфильтровать false positive (центроиды)
5. Удалить повторяющиеся углы (non-maximal suppression)

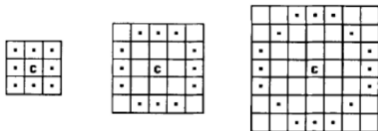
Недостатки:

- Неустойчив к размытию на изображениях

Trajkovic(1998)

- Тряковиц и Хедли (Miroslav Trajkovic and Mark Hedley) в статье «Fast corner detection» ввели новый тип детектора — оператор Тряковица
- Изначально к нему авторы предъявили требования стать самым популярным детектором углов и обладать минимальной вычислительной стоимостью. Сперва был разработан 4-соседний алгоритм Trajkovic4.

Trajkovic(1998)



Алгоритм:

1. Вычисляем CRF $R_N = \min_{P, P' \in S_N} ((I_P - I_N)^2 + (I_{P'} - I_N)^2)$,
для 4х точек на изображении низкого разрешения
 N – центральная точка;
 P и P' – две противоположные по диаметру точки
вокруг точки N ;
 S_N – дискретизированная окружность на
изображении диаметром 3, 5, 7 пикселей.
2. Области с большим CRN вероятно являются углами
(фильтрация по порогу)
3. Пересчитывается оценка угла на полноразмерной
картинке (фильтрация по порогу)
4. Удаляются повторяющиеся углы (non-maximal
suppression)

Недостатки:

- Ложно реагирует на диагональные края
- Чувствителен к шуму

Trajkovic(1998)

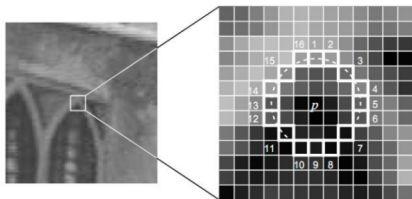
- В сравнение с детектором Харриса частота повторяемости алгоритма Trajkovic4 хуже, однако локализация сравнима с определением L-связных углов и превосходит на других видах углов.
- Однако Trajkovic8 всё еще находит ложные углы на некоторых диагональных гранях объекта (плохо проявляет себя на искусственных изображениях).

Edward Rosten and Tom Drummond



- Ростен и Драммонд ввели достаточно успешный алгоритм FAST (Features from Accelerated Segment Test)

Fast(2005)



Алгоритм:

1. Рассматривается окрестность из 16 пикселей
2. Точка является угловой, если для текущей рассматриваемой точки P существуют N смежных пикселей на окружности, интенсивности которых больше $I_P + t$ или интенсивности всех меньше $I_P - t$, где I_P – интенсивность точки P , t – пороговая величина.

Достоинства:

- Очень. Очень быстрый алгоритм

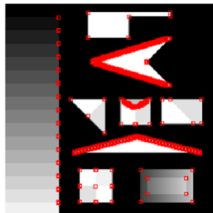
Недостатки:

- Повторяемость ключевых точек
(решается различными улучшениями)

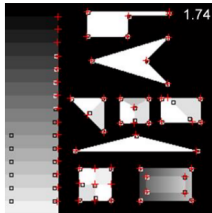
Оптимизация: проверять точки 1, 5, 9, 13

Вариации: FAST-9, FAST + ML, FAST-ER

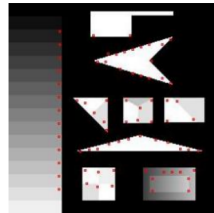
Поведение угловых детекторов на искусственном изображении Смита (The synthetic test pattern of Smith)



Moravec

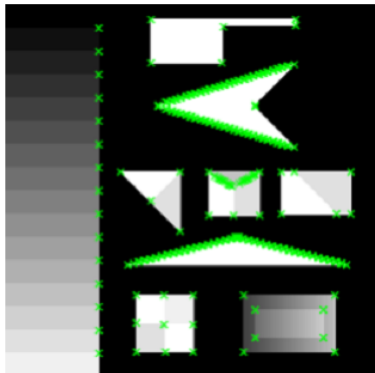


Harris

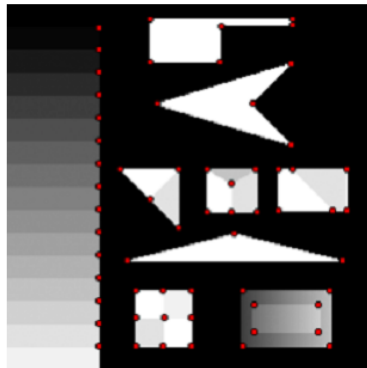


Shi-Tomasi

Поведение угловых детекторов на искусственном изображении Смита (The synthetic test pattern of Smith)

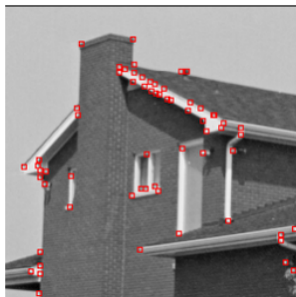


Trajkovic8

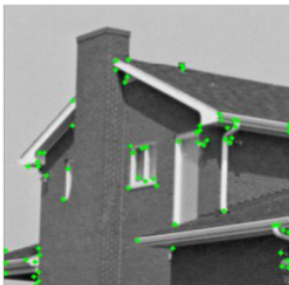


SUSAN

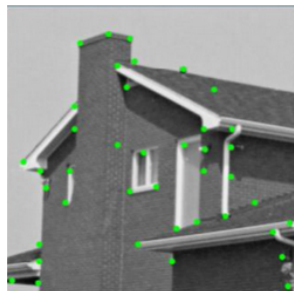
Поведение угловых детекторов на тестовом изображении дома (House Test Image)



Moravec

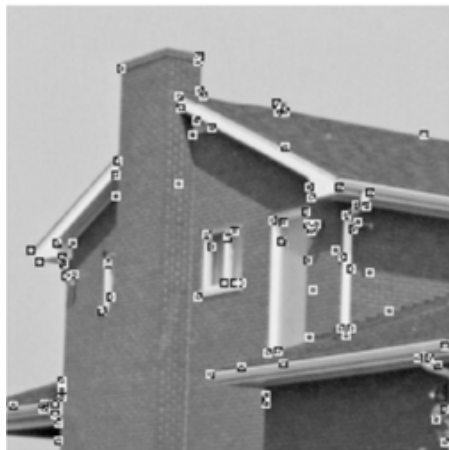


Harris

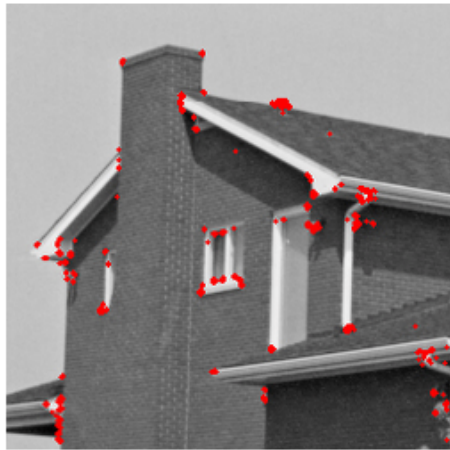


Förstner

Поведение угловых детекторов на тестовом изображении дома (House Test Image)



SUSAN



FAST

Содержание

- 1 Сопоставление изображений
 - Различные подходы
 - Что такое ключевая особенность?
- 2 Особые точки
 - Свойства особых точек
 - Детекторы углов
- 3 Детекторы и дескрипторы
 - SURF

Дескриптор

- Вектор, описывающие структуру окрестности точечной особенности.
- Как правило, эти вектора формируются на основе набора значений первых и вторых производных изображения в точке.

SURF



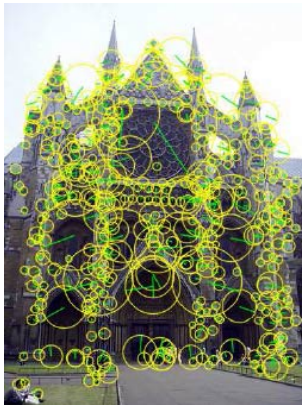
SURF

- Speeded Up Robust Features
- Решает две задачи – поиск особых точек изображения и создание их дескрипторов.
- Обнаружение особых точек в SURF основано на вычислении детерминанта матрицы Гессе (гессиана).

$$H(f(x, y)) = \begin{bmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial x \partial y} & \frac{\partial^2 f}{\partial y^2} \end{bmatrix}$$

$$\det(H) = \frac{\partial^2 f}{\partial x^2} \frac{\partial^2 f}{\partial y^2} - \left(\frac{\partial^2 f}{\partial x \partial y} \right)^2$$

Особые точки изображения здания, найденные с помощью матрицы Гессе



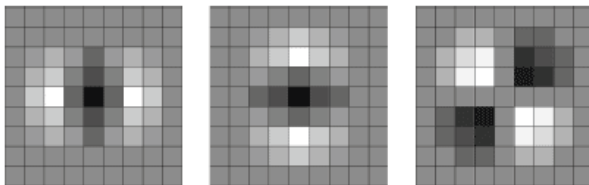
- Диаметр круга показывает масштаб особой точки.
- Зеленая линия – направление градиента яркости.

Дескрипторы

- Дескриптор представляет собой набор из 64(либо 128) чисел для каждой ключевой точки.
- Эти числа отображают флуктуации градиента вокруг ключевой точки

Дискретизированные фильтры для нахождения четырех элементов матрицы Гессе

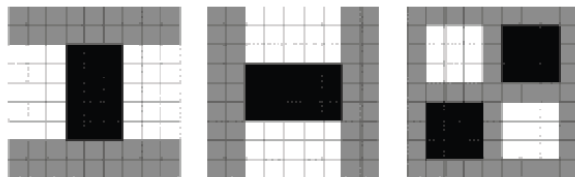
- Теоретически, вычисление матрицы Гессе сводится к нахождению Лапласиана Гауссиан (свёртка фильтра с изображением)



- Четвертый – совпадает с третьим, поскольку матрица Гессе симметрична

Фильтры, используемые для нахождения матрицы Гессе в SURF

- Использует бинаризованную аппроксимацию лапласиана гауссиан (Fast-Hessian).



- Белые области соответствуют значению $+1$, черные -2 (на третьем фильтре -1), серые – нулевые. Пространственный масштаб – 9×9 пикселей.

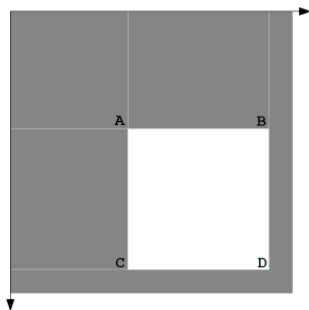
Интегральное представление

- Для эффективного вычисления фильтров Гессе и Хаара – используется интегральное представление изображений.
- Интегральное представление является матрицей, размерность которой совпадает с размерностью исходного изображения, а элементы считаются по формуле:

$$H(x, y) = \sum_{i=0, j=0}^{i \leq x, j \leq y} I(i, j)$$

Интегральное представление

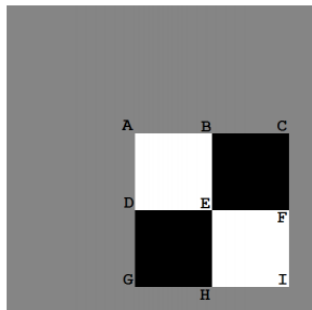
- Имея интегральную матрицу можно очень быстро вычислять сумму яркостей пикселей произвольных прямоугольных областей изображения, по формуле:



$$I_{ABCD} = L_D - L_B - L_C + L_A$$

$$L(x, y) = \sum_{i=0}^{x} \sum_{j=0}^{y} I(i, j)$$

Интегральное представление



$$S_{ACGI} = L_A - 2L_B + L_C - 2L_D + 4L_E - 2L_F + L_G - 2L_H + L_I$$

SURF

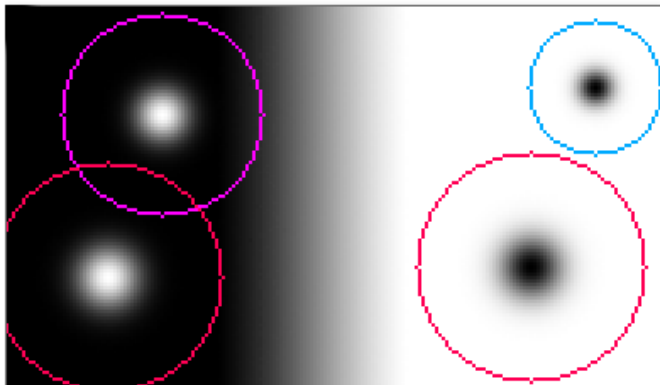
- гессиан вычисляется так:

$$\det(H_{approx}) = D_{xx}D_{yy} - (0.9D_{xy})^2$$

- где D_{xx} , D_{yy} , D_{xy} – свёртки по фильтрам, изображенным на рисунке
- Коэффициент 0.9 имеет теоретическое обоснование, и корректирует приближенный характер вычислений.
-
- Итак, для нахождения особых точек, SURF пробегается по пикселям изображения и ищет максимум гессиана.

Инвариантность к уровню освещения

- Гессиан зависит только от перепада яркости.



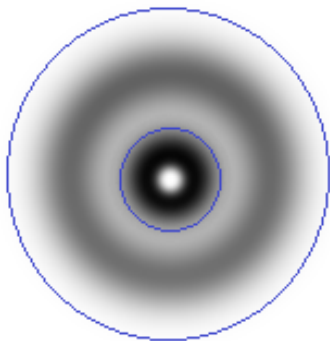
- Метод распознает как светлые точки на темном фоне, так и темные точки на светлом фоне

SURF

- Гессиан инвариантен относительно вращения.
- Но не инвариантен масштабу. Поэтому SURF использует разномасштабные фильтры для нахождения гессианов.
- Решение этой проблемы только одно – перебирать различные масштабы фильтров и поочередно их применять к данному пикселу.

Масштаб

- Две ключевые точки разного масштаба в одной точке изображения.

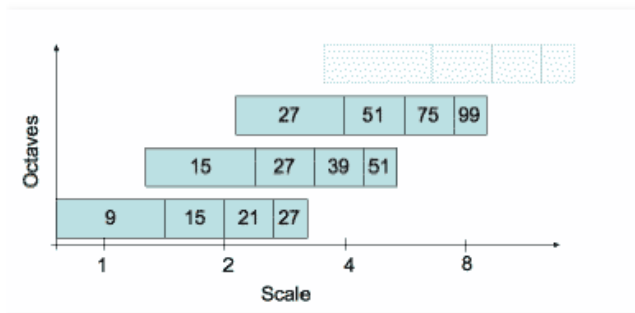


Масштаб

- Из соображений симметрии и дискретизации, размер фильтра Fast-Hessian не может принимать произвольные значения.
- Допустимые размеры этого фильтра: 9, 15, 21, 27 и так далее, с шагом 6.
- Однако, для крупных масштабов шаг 6 оказывается слишком мелким, а фильтры — избыточными.

Октавы

- SURF разбивает все множество масштабов на так называемые октавы.
- Каждая октава покрывает определенный интервал масштабов.
- Каждая октава содержит не один фильтр, а четыре фильтра, которые хорошо покрывают характерный масштаб октавы

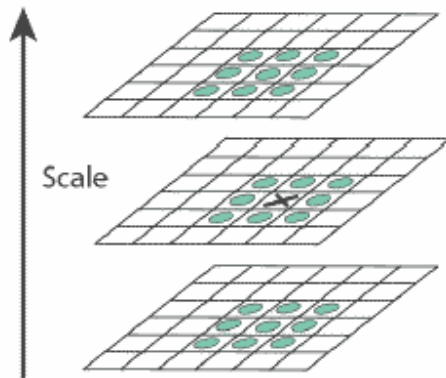


Для каких пикселов считать фильтры?

- Фильтры октавы считаются не для всех пикселов подряд. Первая октава считается для каждого второго пиксела изображения.
- Вторая – для каждого четвертого, третья – для каждого восьмого и так далее.
- Удвоение шага пикселов для октав позволяет экономить при расчёте фильтров.

Нахождение локального максимума гессиана

- метод соседних точек $3 \times 3 \times 3$



Нахождение точки истинного максимума

- Используется интерполирование найденных гессианов куба $3 \times 3 \times 3$ квадратичной функцией.
- Вычисляется производная (методом конечных разностей соседних точек). Если она близка к нулю – мы в точке истинного максимума.
- Если производная велика – сдвигаемся в сторону ее уменьшения, и повторяем итерацию, до тех пор пока производная не станет меньше заданного порога.
- Если в процессе итераций мы отходим от начальной точки слишком далеко, то это считается ложным максимумом, и точка больше не считается особой

Нахождение ориентации особой точки

- Для инвариантности вычисления дескрипторов особой точки требуется определить преобладающую ориентацию перепадов яркости в особой точке.
- Это понятие близко к понятию градиента, но SURF использует немного другой алгоритм нахождения вектора ориентации.

Нахождение ориентации особой точки

- Вычисляются точечные градиенты в пикселах, соседних с особой точкой.
- Для рассмотрения берутся пиксели в окружности радиуса $6s$ вокруг особой точки. Где s – масштаб особой точки.
- Для первой октавы берутся точки из окрестности радиусом 12 .

Вычисления градиентов

- Для вычисления градиента, используется фильтр Хаара. Размер фильтра берется равным $4s$. Вид фильтров Хаара:



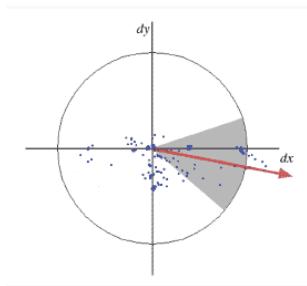
- Поскольку фильтры Хаара имеют прямоугольную форму, их значения легко считаются с помощью интегральной матрицы.
- Для расчёта одного фильтра произвольного размера требуется всего 6 операций.

Вычисления градиентов

- Значения вейвлета Хаара dX и dY для каждой точки умножаются на вес и запоминаются в массиве.
- Вес определяется как значение гауссианы с центром в особой точке и сигмой равной $2s$.
- Взвешивание на гауссиане необходимо для отсеечения случайных помех на далеких от особой точки расстояниях.

Приоритетное направление в области особой точки

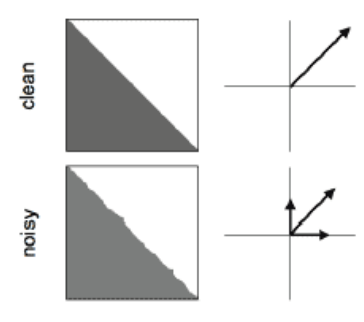
- Все найденные значения градиентов, условно наносятся в виде точек на плоскость



- Берётся угловое окно (60°), и вращается вокруг центра координат. Выбирается такое положение окна, при котором длина суммарного вектора для попавших в окно точек – максимальна. Вектор нормируется.

Уменьшения влияния шумовых точек

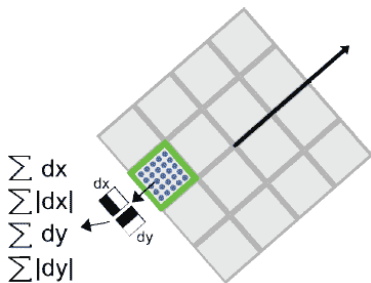
- Пример градиента при идеальном крае, и при крае с шумом



- Метод SURF имеет модификацию, в которой ориентация особых точек не рассчитывается. Такая модификация позволяет надежно идентифицировать точки, повернутые не более чем на ± 15 градусов.

Вычисление дескриптора особой точки

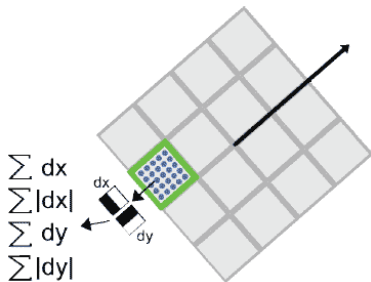
- Дескриптор представляют собой массив из 64 (в расширенной версии 128) чисел



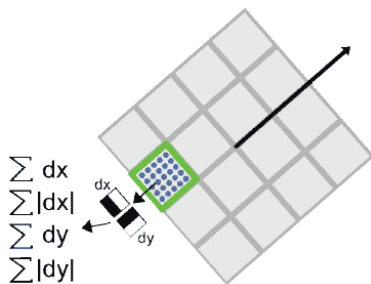
- Для вычисления дескриптора, вокруг особой точки формируется прямоугольная область, имеющая размер $20s$, где s – масштаб в котором была найдена особая точка. Для первой октавы, область имеет размер 40×40 пикселей. Квадрат ориентируется вдоль приоритетного направления, вычисленного для особой точки.

Дескриптор считается как описание градиента для 16 квадрантов вокруг особой точки

- Далее, квадрат разбивается на 16 более мелких квадрантов.
- В каждом квадранте берется регулярная сетка 5x5. Для каждой точки сетки ищется градиент с помощью фильтра Хаара. Размер фильтра Хаара берется равным $2s$, и для первой октавы составляет 4×4 .



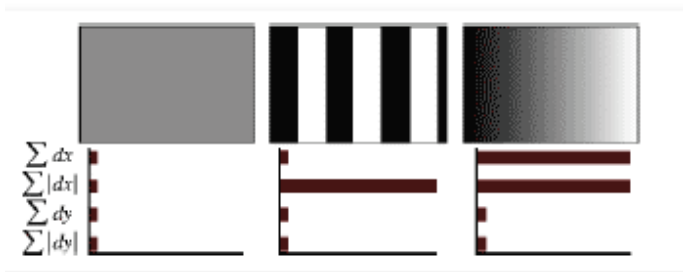
Дескриптор считается как описание градиента для 16 квадрантов вокруг особой точки



- При расчёте фильтра Хаара, изображение не поворачивается, фильтр считается в обычных координатах изображения.
- А вот полученные координаты градиента (dX, dY) поворачиваются на угол, соответствующий ориентации квадрата.

Компоненты дескриптора

- После нахождения 25 точечных градиента квадранта, вычисляются четыре величины, которые и являются компонентами дескриптора

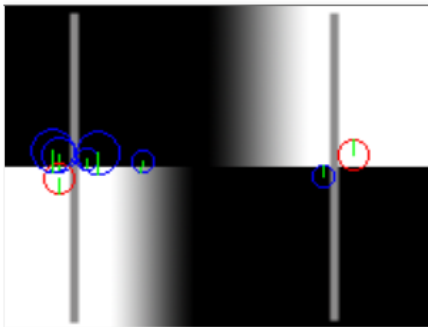


Компоненты дескриптора

- Четыре компонента на каждый квадрант и 16 квадрантов дают 64 компонента дескриптора для всей области особой точки.
- При занесении в массив значения дескрипторов взвешиваются на гауссиану с центром в особой точке и с сигмой $3s$.

Дескриптор плюс знак следа матрицы Гессе

- $\text{sign}(D_{xx}+D_{yy})$



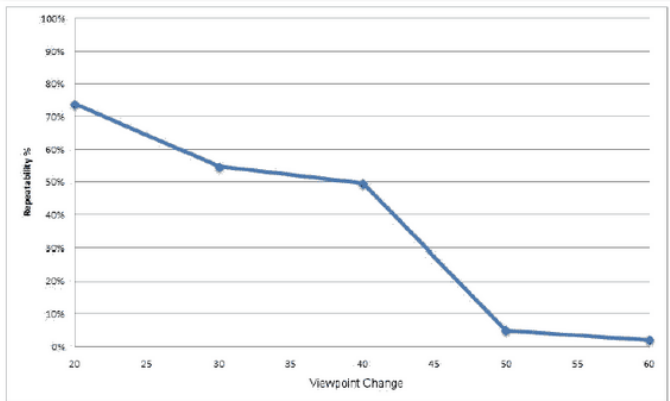
- На картинке показаны особые точки изображения. Зеленая линия показывает характерное направление для особой точки. Синий цвет окружности показывает положительный след матрицы Гессе, красный – отрицательный след

Тест

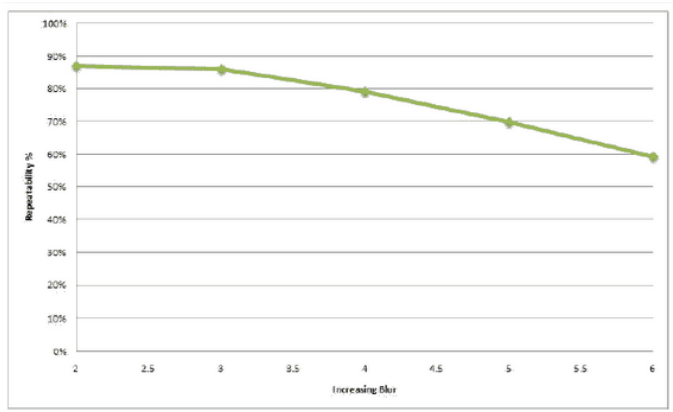


- 1) изображение граффити, снятое под разным углом зрения. 2) фотографии с размытием. 3) поворот и изменение масштаба. 4) разные уровни яркости. 5) идентичное изображение, но с разным качеством сжатия JPEG.

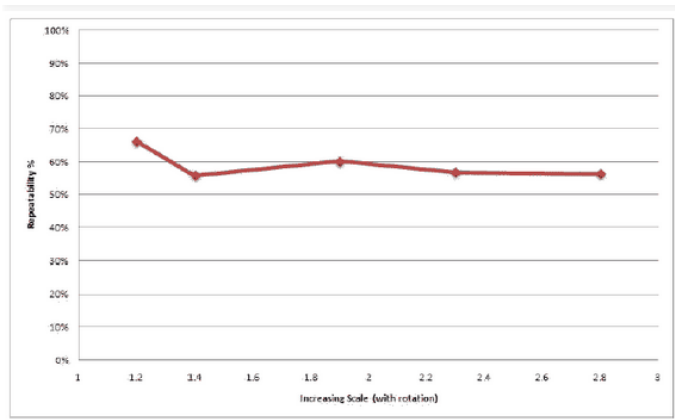
Изображение граффити, снятое под разным углом зрения



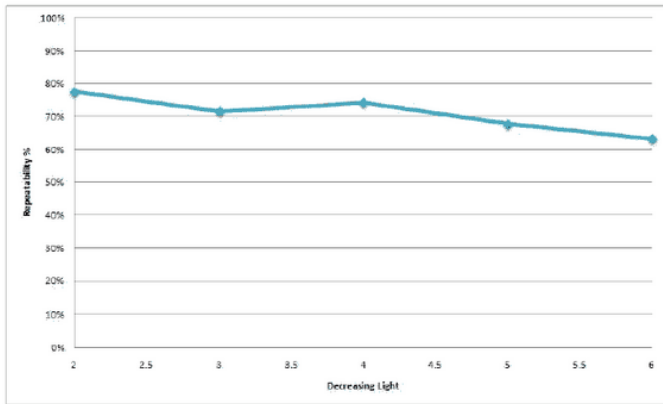
Фотографии с размытием



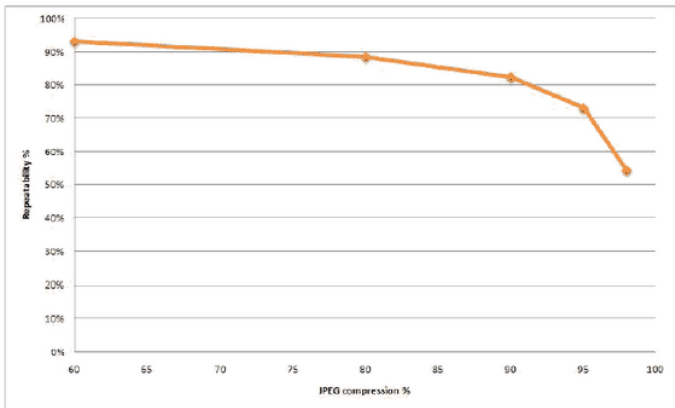
Поворот и изменение масштаба



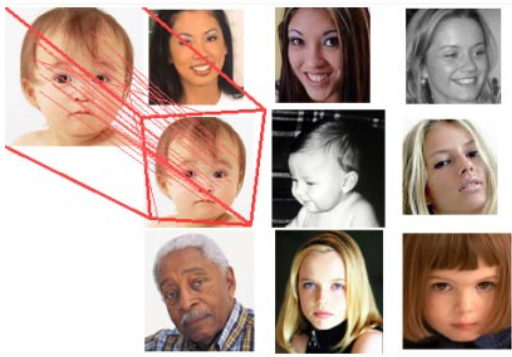
Разные уровни яркости



Идентичное изображение, но с разным качеством сжатия JPEG



Результаты

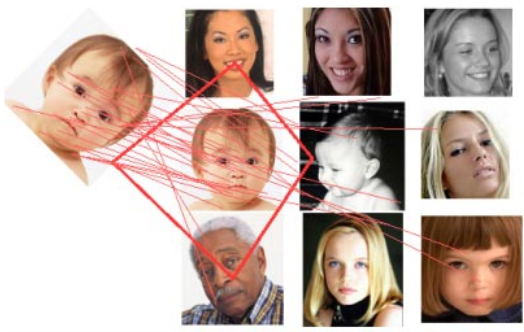


Результаты



- Образец имеет в два раза больший масштаб, чем на сцене. Успешно локализован.

Результаты



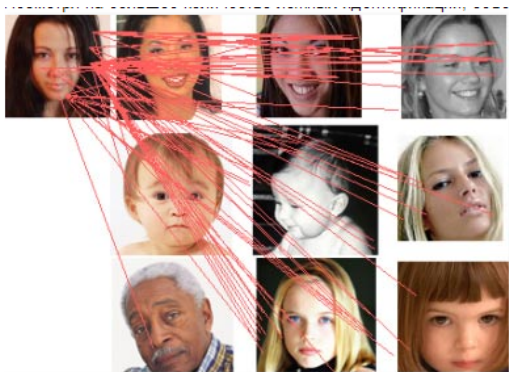
- Образец был повернут на 45 градусов. Локализован.

Результаты



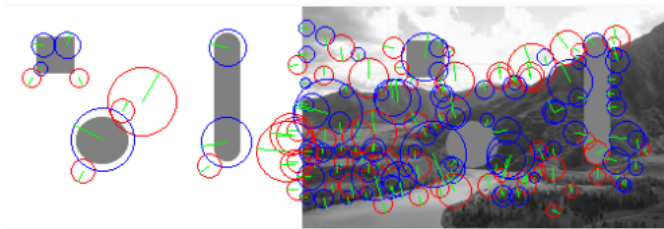
- Несмотря на большое количество ложных идентификаций, объект успешно локализован.

Результаты



- Образца нет на сцене. Алгоритм не нашел локализации.

Недостаток



- Недостаток – плохо работает для объектов простой формы и без ярко выраженной текстуры.