

# Компьютерное зрение '2014

Who? Александр Вахитов

When? October 21, 2014

# План лекции

Регистрация  
изображений

Варианты  
обобщения  
метода ЛК

Детекция  
особенностей

Точки

SIFT

Линии

Регионы

## Задача

$g$  - наблюдаемое изображение

$f$  - образец

$$g(x, y) = f(x + d_x, y + d_y) + n$$

$f$  - исходное изображение

$(d_x, d_y)^T$  - искомый сдвиг

$n$  - случайная помеха

Функция стоимости: сумма квадратов разностей (SSD)

$$C_{LK}(t_x, t_y) = \int_x \int_y (f(x, y) - g(x + t_x, y + t_y))^2 \rightarrow \min$$

Цель:

$$\begin{aligned} \nabla C_{LK}(t_x, t_y) &= 0 = \\ &= -2 \int_{x,y} (f(x, y) - g(x + t_x, y + t_y)) \nabla g(x + t_x, y + t_y). \end{aligned}$$

## Метод Ньютона

$$\hat{\theta}_{n+1} = \hat{\theta}_n - \alpha(\nabla^2 C(x))^{-1}\nabla C(x)$$

где  $\hat{\theta}_n$  - оценка минимума на итерации  $n$ ,  
 $\alpha$  - коэффициент,  
 $C(x)$  - минимизируемая функция стоимости.

## Метод Ньютона

$$\hat{\theta}_{n+1} = \hat{\theta}_n - \alpha(\nabla^2 C(x, y))^{-1} \nabla C(x, y)$$

Наша функция

$$C_{LK}(\theta_x, \theta_y) = \int_{x,y} (f(x, y) - g(x + \theta_x, y + \theta_y))^2 \rightarrow \min$$

Необходимо хорошее начальное приближение

## Параметризация деформации

- сдвиг
- сдвиг + вращение (ортогональное преобразование)
- аффинное преобразование
- нелинейная деформация

## Общий вид сопоставления

Введем функцию стоимости сопоставления  $C(p)$ , обозначая как  $\Omega$  множество сопоставляемых пикселей изображений  $I$  и  $J$ : Интерполяция

$$J(x, y) \in C(\mathbb{Z}^2 \rightarrow \mathbb{R}) \rightarrow J_c(s, t) \in C(\mathbb{R}^2 \rightarrow \mathbb{R})$$

Трансформация координат с параметром  $p$ :

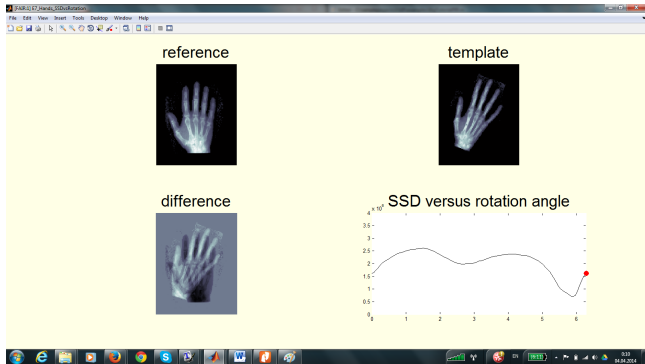
$$y = T_p(x)$$

Например, сдвиг:  $y = x + p$

$$C(p) = \int_{\Omega} \rho(I(x), J_c(T_p(x))) \rightarrow \min_p$$



# Пример: ищем вращение



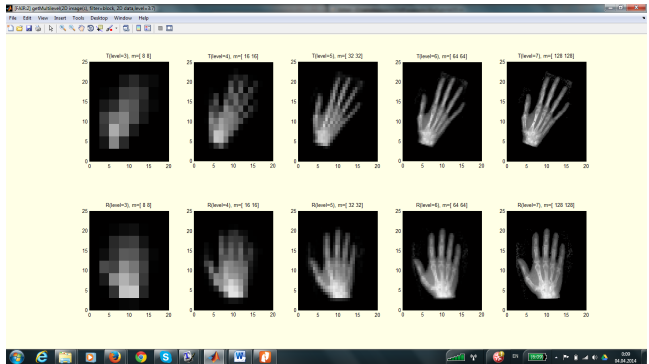
FAIR (J. Modersitzki)

## Интерполяция при регистрации изображений

$$T_p(J)(x) = J_p(x) * \delta(x),$$

где  $J_p(t)$  - непрерывная функция двумерного аргумента, представляющая деформированное изображение  $J$ . Деформация задана параметрами  $p$ . Например,  $J(t)$  - линейно интерполированное изображение,  $J_p(t) = J(t - p)$  - сдвинутое на  $p$  изображение.

# Пирамидальный подход



FAIR (J. Modersitzki)

## Функции качества

- нормализованная корреляция

$$C_{NC} = \frac{\int_{x,y} f(x,y)g(x+t_x, y+t_y)}{\|f(x,y)\|_{L_2}\|g(x,y)\|_{L_2}}$$

- взаимная информация (через интерпретацию изображений как случайных величин)

$$C_{MI} = H(g(x,y)) +$$

$$+ H(f(x,y)) - H(f(x+t_x, y+t_y), g(x,y))$$

где  $H$  - энтропия цвета изображений

## Какие бывают особенности



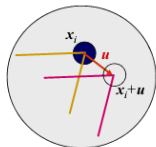
## Какие бывают особенности

- Точки
- Регионы
- Линии

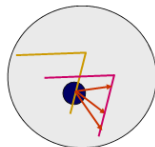
## Работа с особенностями

- Детекция
- Описание
- Сопоставление / отслеживание

# Какие бывают точки



(a)



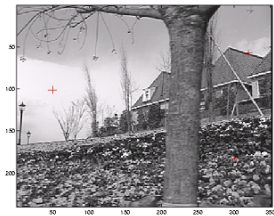
(b)



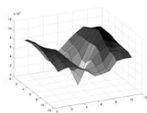
(c)



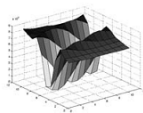
# Какие бывают точки



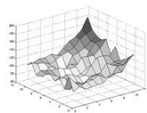
(a)



(b)



(c)



(d)

## Задача: отслеживание с помощью ЛК

$$C_{LK}(t_x, t_y) = \int_x \int_y (f(x, y) - g(x + t_x, y + t_y))^2 \rightarrow \min$$

Цель:

$$\begin{aligned} \nabla C_{LK}(t_x, t_y) = 0 = \\ = -2 \int_{x,y} (f(x, y) - g(x + t_x, y + t_y)) \nabla g(x + t_x, y + t_y). \end{aligned}$$

## Решение и его свойства

Разложение по Тейлору до первой степени:

$$g(x + t_x, y + t_y) = g(x, y) + (t_x, t_y)^T \nabla g(x, y)$$

$$\begin{aligned} 0 &= -2 \int_{x,y} (f(x, y) - g(x + t_x, y + t_y)) \nabla g(x + t_x, y + t_y) = \\ &= -2 \int_{x,y} (f(x, y) - g(x, y) - (t_x, t_y)^T \nabla g(x, y)) \nabla g(x, y) \end{aligned}$$

Линейная система:

$$\int_{x,y} \nabla g(x, y) \nabla^T g(x, y) dx dy (t_x, t_y)^T = b$$

$$Ax = b$$

## Обусловленность системы

$$Ax = b$$

$$A(x + \Delta x) = b + \Delta b$$

Насколько изменится ответ при изменении правой части?

$$\|\Delta x\| = f(\|\Delta b\|)$$

Требование:  $\min\{\lambda_1, \lambda_2\} > \lambda_{\min}$  (Детектор Shi-Tomasi)

## Детектор Харриса

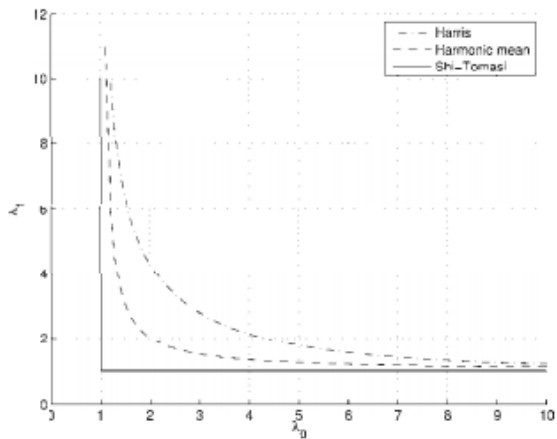
$$M_C = \lambda_1 \lambda_2 - \kappa(\lambda_1 + \lambda_2)^2$$

Уголок:  $M_C > \epsilon$

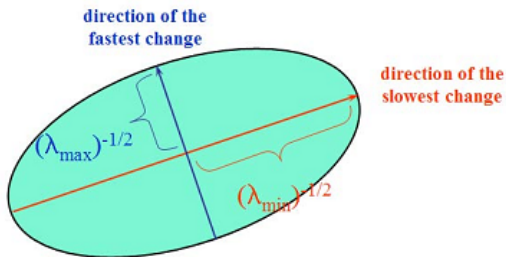
Другие функции:

$$\frac{\det A}{\text{Tr} A} = \frac{\lambda_1 - \alpha \lambda_2}{\lambda_1 + \lambda_2}$$

# Детекторы углов



## Структура изменения яркости в окрестности особой точки



## Kanade Lucas Tomasi (KLT)- трекинг

Необходимо найти положение точки с текущего кадра на следующем кадре видео

Детектор  
Shi-Tomasi

Детекция точек, оптимальных для регистрации при преобразовании сдвига:  $\min\{\lambda_1, \lambda_2\} > \lambda_{min}$

Трекинг  
Лукас-Канаде

Регистрация малого региона вокруг точки на следующем кадре видео. *Трансформация: сдвиг.*

Методы OpenCV: `cv2.goodFeaturesToTrack`,  
`cv2.calcPyrLK`

<http://www.youtube.com/watch?v=Jw1CCR0tAcY>

Проблема: накопление ошибки. Модель не подходит для поиска точки на всех кадрах видео по одному кадру



## DoG: Difference of Gaussians

$$DoG(I, \sigma, k) = G(k * \sigma) * I - G(\sigma) * I$$

$G(\sigma)$  - гауссовское ядро

$k$  - коэффициент

Детектор: экстремум DoG

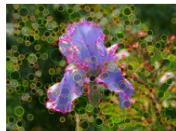
## Сравнение DoG и Harris



(a)



(b)



(c)

Вывод: взаимно дополняют друг друга

Размер круга: масштаб

Выбор масштаба?

# Разные масштабы важны



## Проблема: неравномерное распределение точек

Детектируемые точки распределены неравномерно, что вызывает проблемы в геометрической реконструкции и др. приложениях

## Non-maximal suppression



(a) Strongest 250



(b) Strongest 500



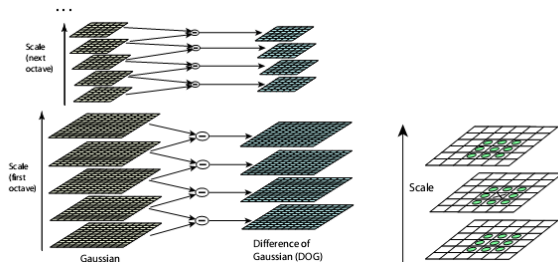
(c) ANMS 250,  $r = 24$



(d) ANMS 500,  $r = 16$

# Scale Invariant Feature Transform (SIFT)

## Построение пирамиды

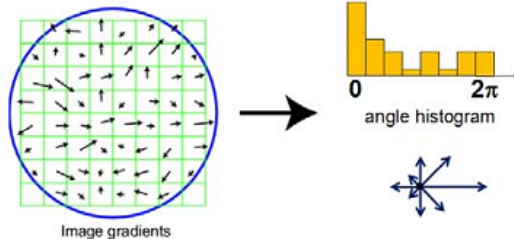


Идея: максимумы в 3D (изображение + масштаб)

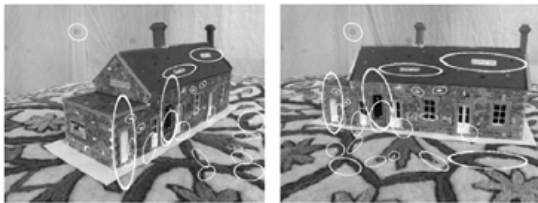
## SIFT: оценка ориентации

Алгоритм SIFT: детекция DoG, разбиение окрестности на квадранты, построение гистограммы градиентов (направлений и длин) для каждого квадранта.

Базовая реализация: 128 компонент = 4 квадранта на 8 направлений на 4 модуля



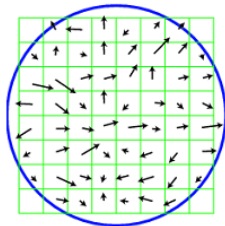
## Аффинные детекторы



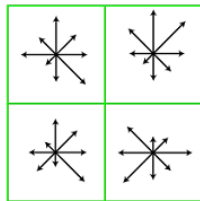
Идея: локальные координаты по "осям"  
автокорреляции



# Дескрипторы: SIFT

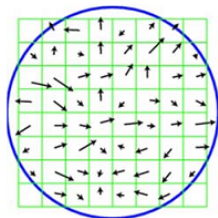


(a) image gradients

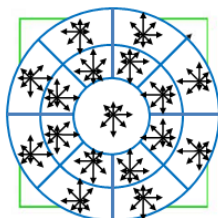


(b) keypoint descriptor

# Дескрипторы: GLOH

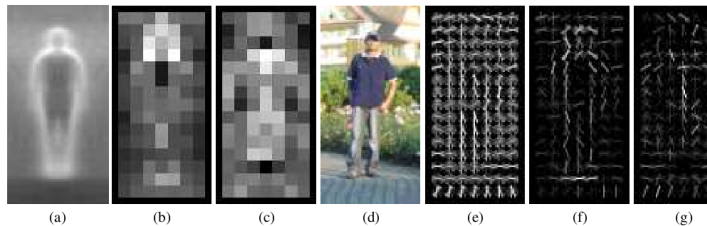


(a) image gradients



(b) keypoint descriptor

## Дескрипторы: HOG



(Dalal, Triggs 2005)

## Детекция краев

Максимумы мощности края в направлении градиента  
- производная градиента. Лапласиан:

$$S_{\sigma}(x) = \nabla J_{\sigma}(x) = \nabla^2 G_{\sigma}(x) * I(x)$$

Осталось найти точки перемены знака.  
Субпиксельно:

$$x_z = \frac{x_i S(x_j) - x_j S(x_i)}{S(x_j) - S(x_i)}$$

## Алгоритм Санны (1986)

Детекция края с двумя барьерами (в два прохода изображения): Изначально пиксели края  $E = \emptyset$

- проход 1: пиксель  $x \rightarrow E$ , если  $|\Delta I(x)| > \theta_0$
- проход 2: пиксель  $x \rightarrow E$ , если  $|\Delta I(x)| > \theta_1$  и  $\exists y \in E : x \in N_y$

Осмысленно, если  $\theta_0 < \theta_1$ ?

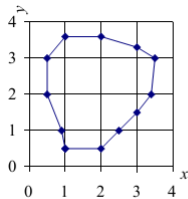
Эффект гистерезиса

## Объединение участков края

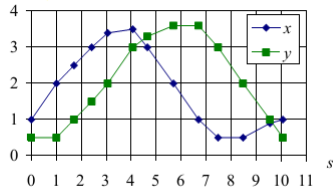
Цепной код: N, NW, W, SW, S, SE, E, NE  
(следующий пиксель по отношению к предыдущему)  
Натуральная параметризация  $x(s)$ :

$$s_0 = \int_0^{s_0} |\nabla x(s)|$$

# Натуральная параметризация



(a)



(b)

$$z(s) = x(s) + iy(s)$$

Поворот равносителен домножению  $z_2 = cz$ ,  $c \in \mathbb{C}$

## Преобразование Хафа (Hough)

Задача: объединить участки в линии

Идея:

- 1) минимальная параметризация прямой  $(\theta, d)$
- 2) голосование за каждое возможное значение  $(\theta, d)$



# Детекция прямых на базе RANSAC

RANSAC = Random Sample Consensus

- 1) случайная гипотеза ( $\theta, d$  - прямая)
- 2) голосование (сколько пикселей поддерживают гипотезу)

## Детекция vanishing point

$$v_{ij} = m_i \times m_j$$

$m_i, m_j$  - уравнения прямых:  $m_i = (\sin(\theta), \cos(\theta), d)$

координаты вр:  $x = v_{ij}^{(1)} / v_{ij}^{(3)}, y = v_{ij}^{(2)} / v_{ij}^{(3)}$

Голосование с весом  $\|v_{ij}\| l_i l_j$ , где  $l_i, l_j$  - длины участков  $m_i, m_j$

## Видео-трекинг объекта: mean shift

Найти объект с исходного кадра на имеющемся кадре

- строим гистограмму цветов объекта по исходному кадру. Гистограмма для каждого цвета содержит вероятность появления пикселя с данным цветом на объекте, т.е. гистограмма - распределение вероятностей  $P(I(x) = Color | x \in Object)$ .
- рассчитываем двумерное распределение вероятностей в имеющемся кадре того, что пиксель принадлежит объекту, на основе гистограммы.
- ищем локальный максимум построенной функции плотности распределения

<http://fr.wikipedia.org/wiki/Camshift>