

# О некоторых подходах к автоматической реконструкции трехмерных объектов и сцен

Михайличенко А.А.  
alexey.a.mikh@gmail.com

Южный федеральный университет, ИММиКН

2015

# Области применения технологии 3D-реконструкции

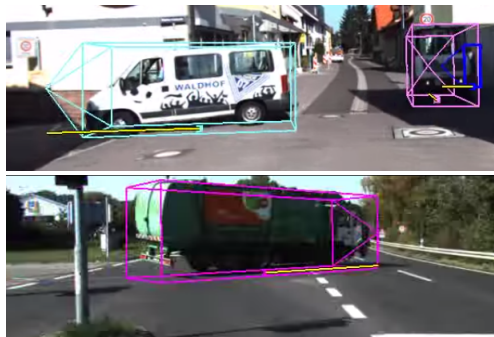


Рис. 1: Пример применения технологии 3D-реконструкции

# Области применения технологии 3D-реконструкции

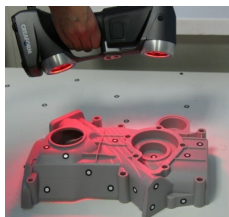
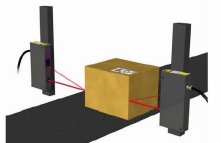
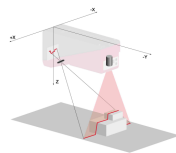


Рис. 2: Использование реконструкции в производстве, для бесконтактного определения дефектов и в логистике

# Существующие технологии 3D-реконструкции



Сканирующие лазерные  
дальномеры



Методы оптической  
триангуляции



Активное стереовидение



Пассивное стереовидение



# Постановка задачи



Левый снимок



Правый снимок



a)



b)

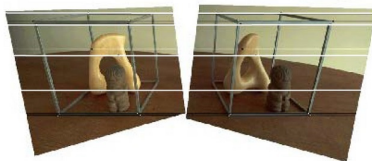
Рис. 3: Результат реконструкции. Вид с различных ракурсов: сверху (a), из позиции камеры (b)



# Корректировка стереопары



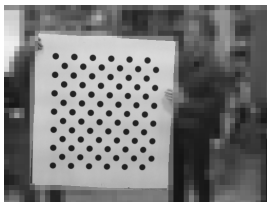
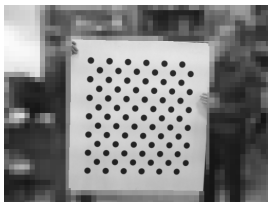
Исходная стереопара



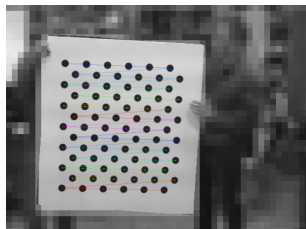
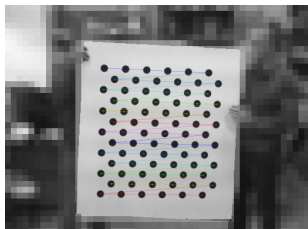
Откорректированная стереопара

OpenCV: `stereoRectify(..)`

# Предварительный этап. Калибровка камеры



Снимки калибровочного шаблона с левой и правой камер



Найденные на этих изображениях характерные элементы шаблона

OpenCV: `calibrateCamera(..)`

# Алгоритмы построения карты смещений

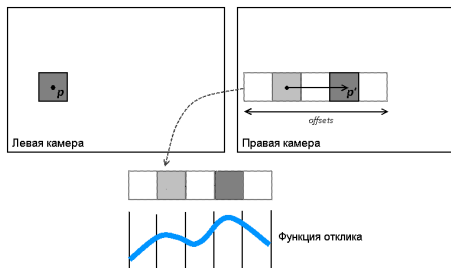


Рис. 5: Общая идея алгоритмов поиска соответствия

- Локальные (StereoBM, SGBM)
- Глобальные
  - Цепи Маркова
  - Графовые модели

OpenCV: StereoBM, StereoSGBM

# Примеры карт смещения



Рис. 6: Примеры работы различных алгоритмов получения карты смещений

# Построение карты глубины



Рис. 7: Скорректированная стереопара и соответствующая ей карта смещений (disparity map)

$$[X \ Y \ Z \ W]^T = Q * [x \ y \ disp(x, y) \ 1]^T \quad (1)$$

OpenCV: `reprojectImageTo3D(..)`

# Результат стерео-реконструкции



Левый снимок



Карта смещений



Результат  
реконструкции



Левый снимок



Карта смещений

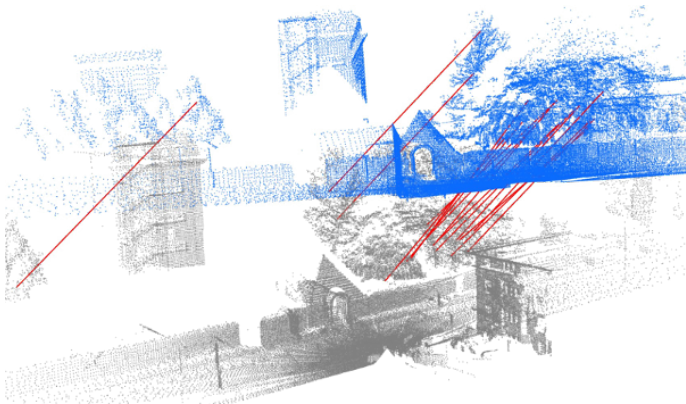


Результат  
реконструкции

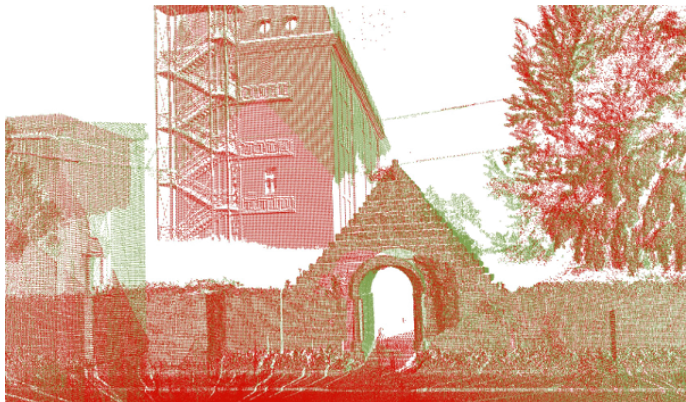


# Регистрация нескольких облаков точек

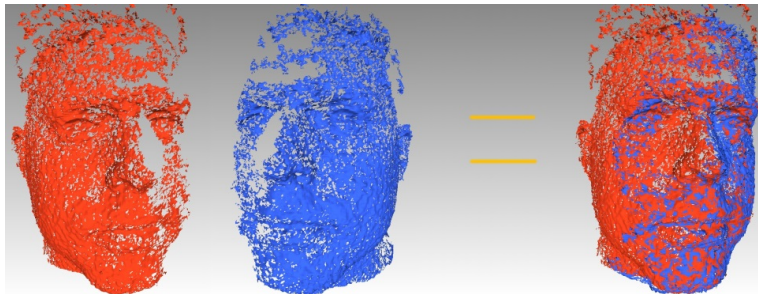
Point Cloud Library:  
`pcl::IterativeClosestPointNonLinear()`



# Результат регистрации пары облаков точек



# Еще один пример регистрации



- The KITTI Vision Benchmark Suite

[http://www.cvlibs.net/datasets/kitti/eval\\_stereo\\_flow.php?benchmark=stereo](http://www.cvlibs.net/datasets/kitti/eval_stereo_flow.php?benchmark=stereo)

- Point Clouds Library

<http://pointclouds.org/>

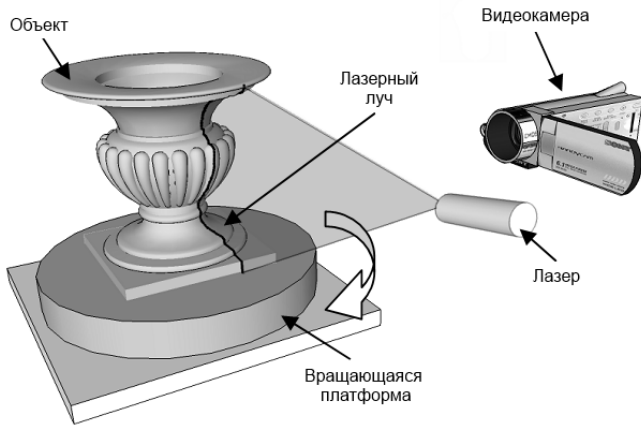
- OpenCV, модуль calib3d

<http://docs.opencv.org/modules/calib3d/doc/calib3d.html>

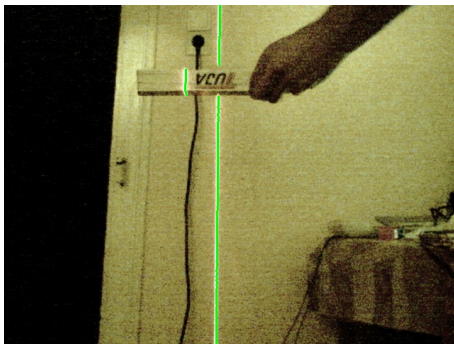
- insight3d

<http://insight3d.sourceforge.net/>

# Методы оптической триангуляции



# Пример отклонения луча



# Задача детектирования лазерного луча в виде линии на изображении



Рис. 8: Пример изображения с лазерным лучом в виде линии и детектирование этого луча

# Результат действия оператора подавления границ

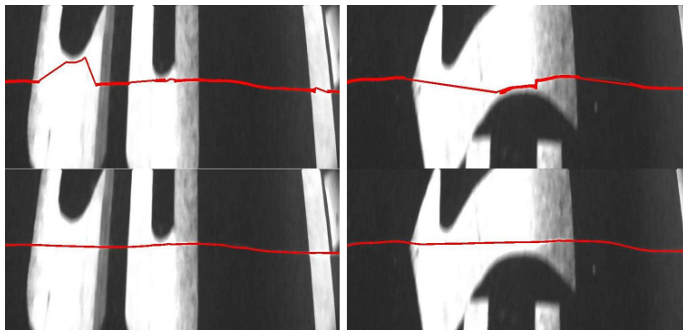


Рис. 9: Пример детектирования луча без использования оператора подавления границ (вверху) и с его использованием (внизу)



# Оператор подавления границ

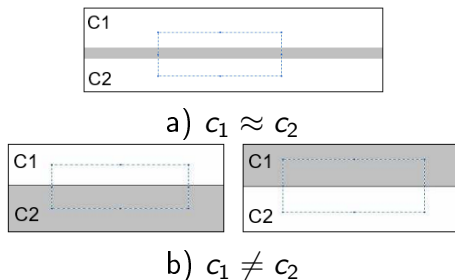


Рис. 10: Пример шаблонов границ образа лазерного луча (a) и границы объекта (b)

$$R = e^{-|c_1 - c_2|} \quad (2)$$

Вид оператора подавления границ

# Активные контуры

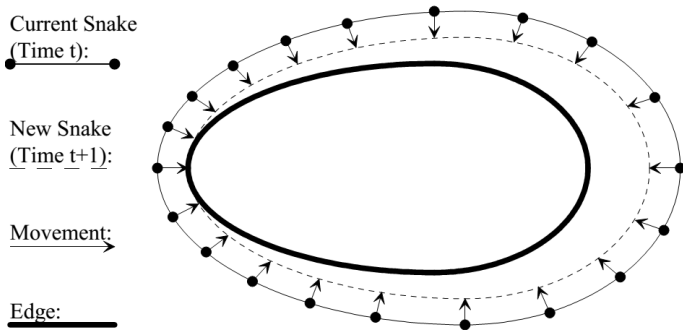


Рис. 11: Общая схема

# Модель эволюции кривой

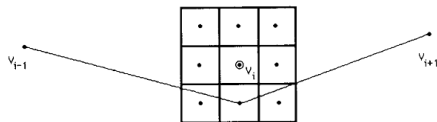


Рис. 12: Жадный алгоритм минимизации энергии контура

$$E(\mathbf{v}) = \alpha E_{img} + \beta E_{grad} + \gamma E_x + \delta E_y + \eta E_{dist} \rightarrow \min \quad (3)$$

Предлагаемый функционал энергии кривой

$$\mathbf{v} = \{v_i = (x_i, y_i)\}_{i=1}^n$$

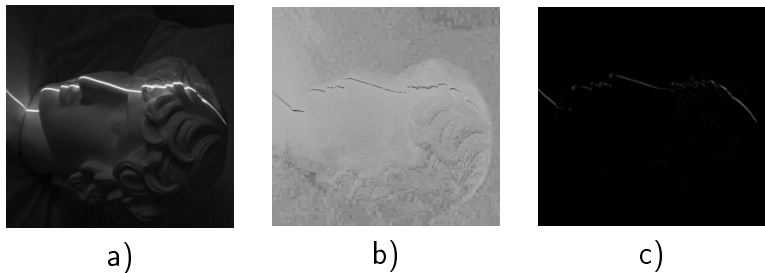
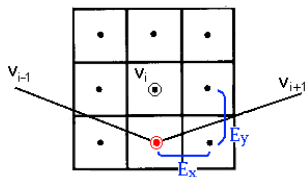


Рис. 13: Исходное в оттенках серого (a),  $E_{img}$  — компонента H формата HSV (b) и  $E_{grad}$  (c)

$$E_{img}(x_i, y_i) = 1 - I(x_i, y_i), \alpha = -1.25, \quad (4)$$

$$E_{grad}(x_i, y_i) = |\nabla I(x_i, y_i)|, \beta = -0.4. \quad (5)$$

Рис. 14:  $E_x$  и  $E_y$ 

$$E_x(x_i + k, y_i + j) = \frac{|k|}{2r + 1}, \quad -r \leq k, j \leq r, \gamma = 0.4,$$

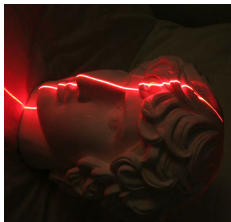
$$E_y(x_i + k, y_i + j) = \frac{-j}{2r + 1}, \quad -r \leq k, j \leq r, \delta = 0.31.$$

$$E_{dist}(v_{kj}) = |v_{i-1} - v_{kj}| + |v_{i+1} - v_{kj}|, \quad (6)$$

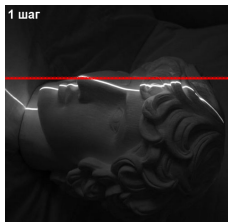
$$v_{kj} = (x_i + k, y_i + j), \eta = 0.05,$$

т.е.  $v_{kj}$  - потенциальное положение вершины  $v_i$

# Входные данные



a)



b)

Рис. 15: Входное изображение (a) и начальное положение кривой (b)

# Пример эволюции



Рис. 16: Поэтапный результат «эволюции» кривой



# Задача детектирования точечной лазерной подсветки на изображении

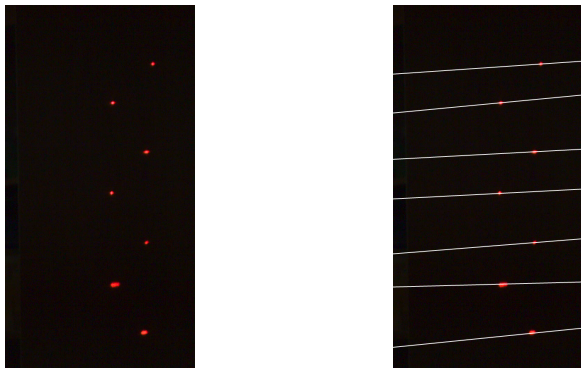


Рис. 17: Пример точечной подсветки и возможные траектории движения точек

# Наивный алгоритм

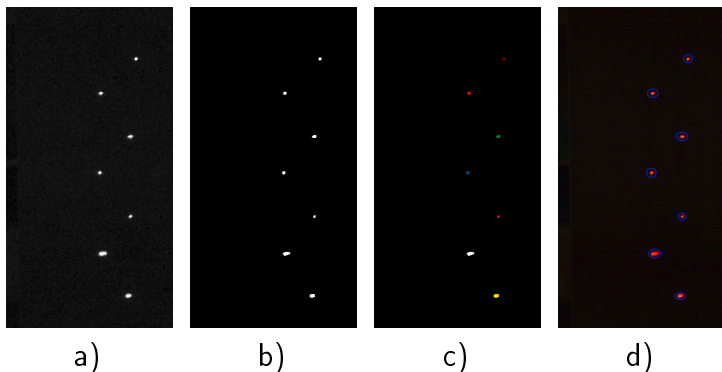


Рис. 18: R-компонента изображения (a), бинаризация (b), разметка найденных пятен (c) и итоговый результат детектирования (d)

# Разность двух гауссианов

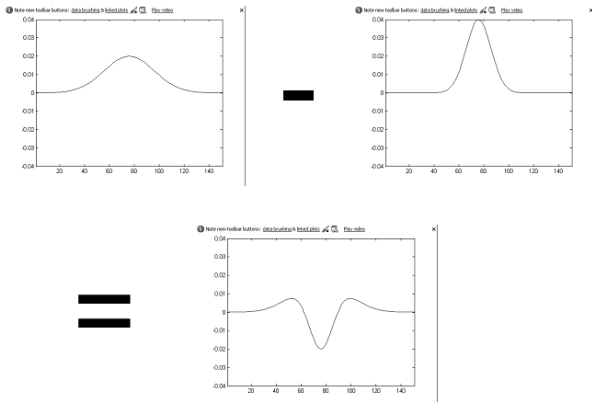
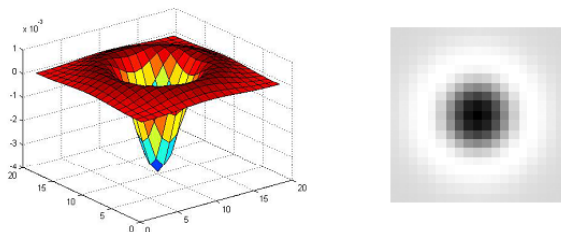


Рис. 19: Difference of Gaussians (DoG)

# Лапласиан гауссиана



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

Рис. 20: Laplacian of Gaussian (LoG)

# Эффект от нормализации масштаба

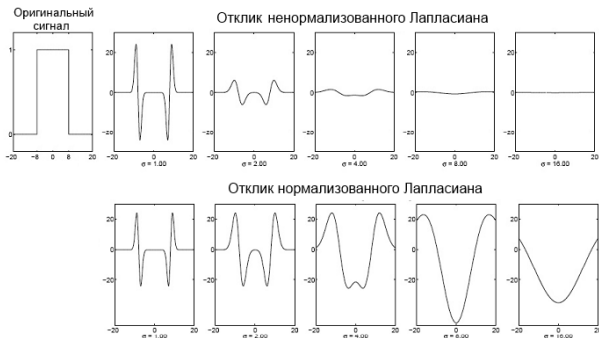
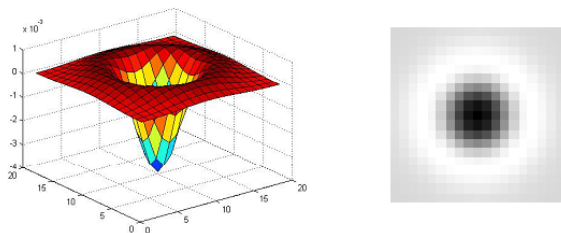


Рис. 21: Пример действия нормализации на отклик лапласиана

# Нормализованный лапласиан гауссиана



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

Рис. 22: Нормализованный LoG

# Эффективная реализация

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

(Laplacian)

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

(Difference of Gaussians)

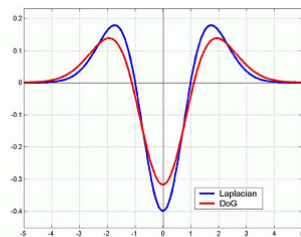


Рис. 23: Аппроксимация LoG и DoG

# Пример работы алгоритма DoG

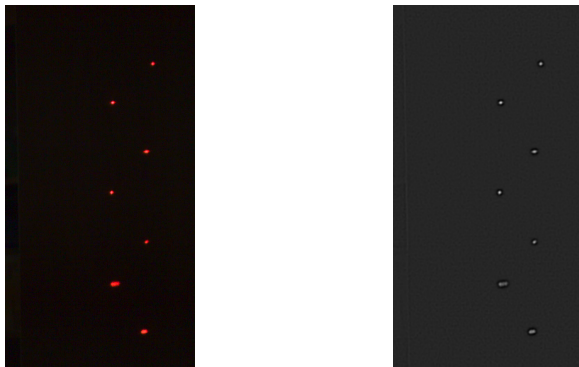


Рис. 24: Применение алгоритма DoG к исходному изображению



# Понятие фона

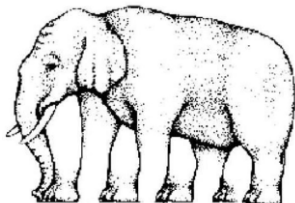


Рис. 25: Фон как неоднозначное понятие

# Подходы к определению фона

- Использование данных о глубине (стереокамеры, Kinect)

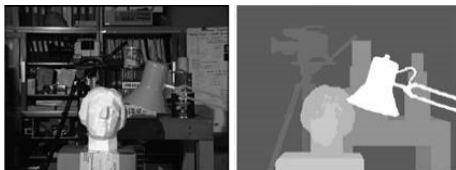


Рис. 26: Определение глубины с помощью стереовидения

- Запоминание картинки фона, затем нужно лишь определить разницу между входным изображением и запомненным

# Адаптивное обучение

```
const float k = 0.01;  
back = (1 - k) * back + k * current_frame;
```

Рис. 27: Усреднение последних N кадров



Рис. 28: Пример усреднения

# Оптический поток



Рис. 29: Пример оптического потока для пары последовательных кадров

# Оптический поток

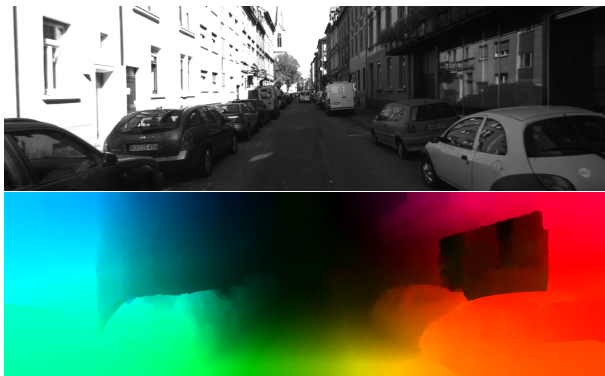


Рис. 30: Еще один пример визуализации оптического потока

# Методы вычисления оптического потока

- Блочные методы
- Дифференциальные методы
- На основе дискретной оптимизации

# Метод Лукаса-Канаде

$$I(x, y, t) = I(x + u_x, y + u_y, t + dt) \quad (7)$$

$$I(x + u_x, y + u_y, t + dt) \cong I(x, y, t) + u_x \frac{\partial I}{\partial x} + u_y \frac{\partial I}{\partial y} + \frac{\partial I}{\partial t} dt \quad (8)$$

$$I(x + u_x, y + u_y, t + dt) = I(x, y, t) \Rightarrow$$

$$u_x \frac{\partial I(x, y, t)}{\partial x} + u_y \frac{\partial I(x, y, t)}{\partial y} + \frac{\partial I(x, y, t)}{\partial t} = 0 \quad (9)$$

$$E(u_x, u_y) = \sum_{i,j} g(x_i, y_j) \left[ u_x \frac{\partial I(x_i, y_j, t)}{\partial x} + u_y \frac{\partial I(x_i, y_j, t)}{\partial y} + \frac{\partial I(x_i, y_j, t)}{\partial t} \right]^2$$

$$E(u_x, u_y) \rightarrow \min_{u_x, u_y} \quad (10)$$

# Недостатки метода Лукаса-Канаде



Рис. 31: Aperture problem



# Улучшение метода

OpenCV:  
buildOpticalFlowPyramid(..)  
calcOpticalFlowPyrLK(..)



Рис. 32: Использование multi-scaling'a

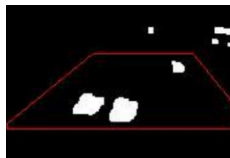
# Отслеживание объекта



a)



b)



c)

Рис. 33: Пример фона (a), результат вычитания фона из некоторого изображения с пороговым разделением (b) и применение морфологических операций (c)

OpenCV: `erode(..)`, `dilate(..)`

## Отслеживание объекта (2)



a)



b)

Рис. 34: Отбрасывание результатов обнаружения вне зоны наблюдения (a) и маркировка объектов (b)

# Метод детектирование ключевых точек

Альтернатива из OpenCV:  
`goodFeaturesToTrack(..)`

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

Рис. 35: Матрица Гессе изображения

$$Z e_i = \lambda_i e_i \quad (i=1,2)$$

Рис. 36: Поиск собственных значений матрицы

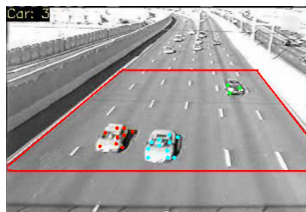
$$\min(\lambda_1, \lambda_2) > \lambda_{\text{Threshold}}$$

Рис. 37: Отбор собственных значений по заданному порогу

# Ключевые точки



a)



b)

Рис. 38: Обнаруженные с помощью гессиана ключевые точки (a) и разделение ключевых точек по соответствующим объектам (b)

# Определение скорости объекта



Рис. 39: Определение направления и величины сдвига ключевых точек с помощью оптического потока

$$v = \frac{1}{n} \sum_{i=1}^n v_i, \quad (11)$$

$$v_i = \frac{\Delta p_i}{\Delta t}, \quad (12)$$

где  $v$  — мгновенная скорость объекта,  $v_i$  — мгновенная скорость,  $\Delta p_i$  — сдвиг  $i$ -й ключевой точки,  $\Delta t$  — разность по времени между двумя кадрами

- Jim Ivins, John Porrill, «Everything You Always Wanted To Know About Snakes (but were afraid to ask», AIVRU Technical Memo #86, July 1993

<http://www.pagines.ma1.upc.edu/~toni/files/SnakesAivru86c.pdf>

- Chenyang Xu, Dzung L. Pham, Jerry L. Prince, «Image Segmentation Using Deformable Models»,  
<http://ee.sharif.edu/>

<http://ee.sharif.edu/~miap/Files/SegmentationUsingDeformableModel.pdf>

## Полезные ссылки

- H. J. Fan, L. P. Zhao, S. Y. He, Y. D. Tang, «A New Open Curve Detection Algorithm for Extracting the Laser Lines on the Road», Advanced Engineering Forum, Vols. 2–3, pp. 205–210, Dec. 2011  
<http://www.scientific.net/AEF.2-3.205>
- R. I. R. García, D. Shu, «Vision based Vehicle Tracking using a high angle camera», <http://www.ces.clemson.edu/>  
[http://www.ces.clemson.edu/~stb/ece847/projects/VISION\\_BASED\\_VEHICLE\\_TRACKING.pdf](http://www.ces.clemson.edu/~stb/ece847/projects/VISION_BASED_VEHICLE_TRACKING.pdf)
- S. Dogan, M. S. Temiz, S. Kulur, «Real Time Speed Estimation of Moving Vehicles from Side View Images from an Uncalibrated Video Camera», Sensors (Basel). 2010; 10(5): 4805–4824  
<http://www.mdpi.com/1424-8220/10/5/4805>



- H. J. Fan, L. P. Zhao, S. Y. He, Y. D. Tang, «A New Open Curve Detection Algorithm for Extracting the Laser Lines on the Road», Advanced Engineering Forum, Vols. 2-3, pp. 205-210, Dec. 2011

<http://www.scientific.net/AEF.2-3.205>

- OpenCV, модуль video

[http://docs.opencv.org/2.4.5/modules/video/doc/motion\\_analysis\\_and\\_object\\_tracking.html](http://docs.opencv.org/2.4.5/modules/video/doc/motion_analysis_and_object_tracking.html)

- `cv::buildOpticalFlowPyramid(..)`
- `cv::calcOpticalFlowPyrLK(..)`
- `cv::BackgroundSubtractorMOG2(..)`