Компьютерное зрение

Лекция № 9, осень 2023

Оптический видеопоток

ЛИТПМ

МФТИ

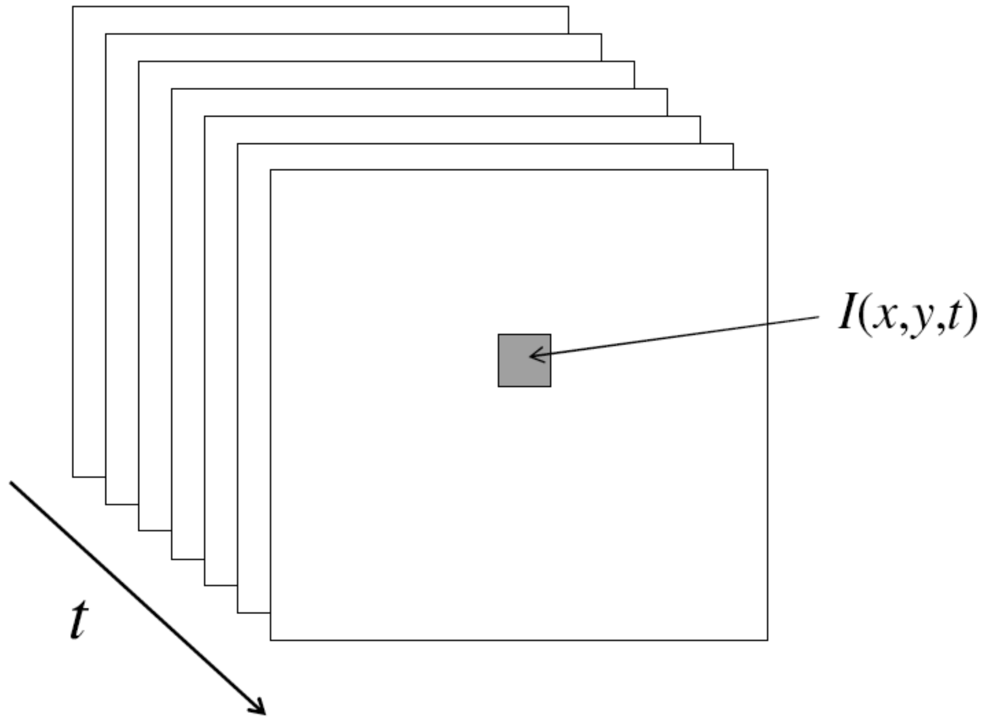
Кафедра технологий проектирования сложных технических систем

Что будем изучать сегодня

- Оптический поток
- Lucas-Kanade method
- Horn-Schunk method
- Пирамиды для большого движения
- Общий подход
- Применение

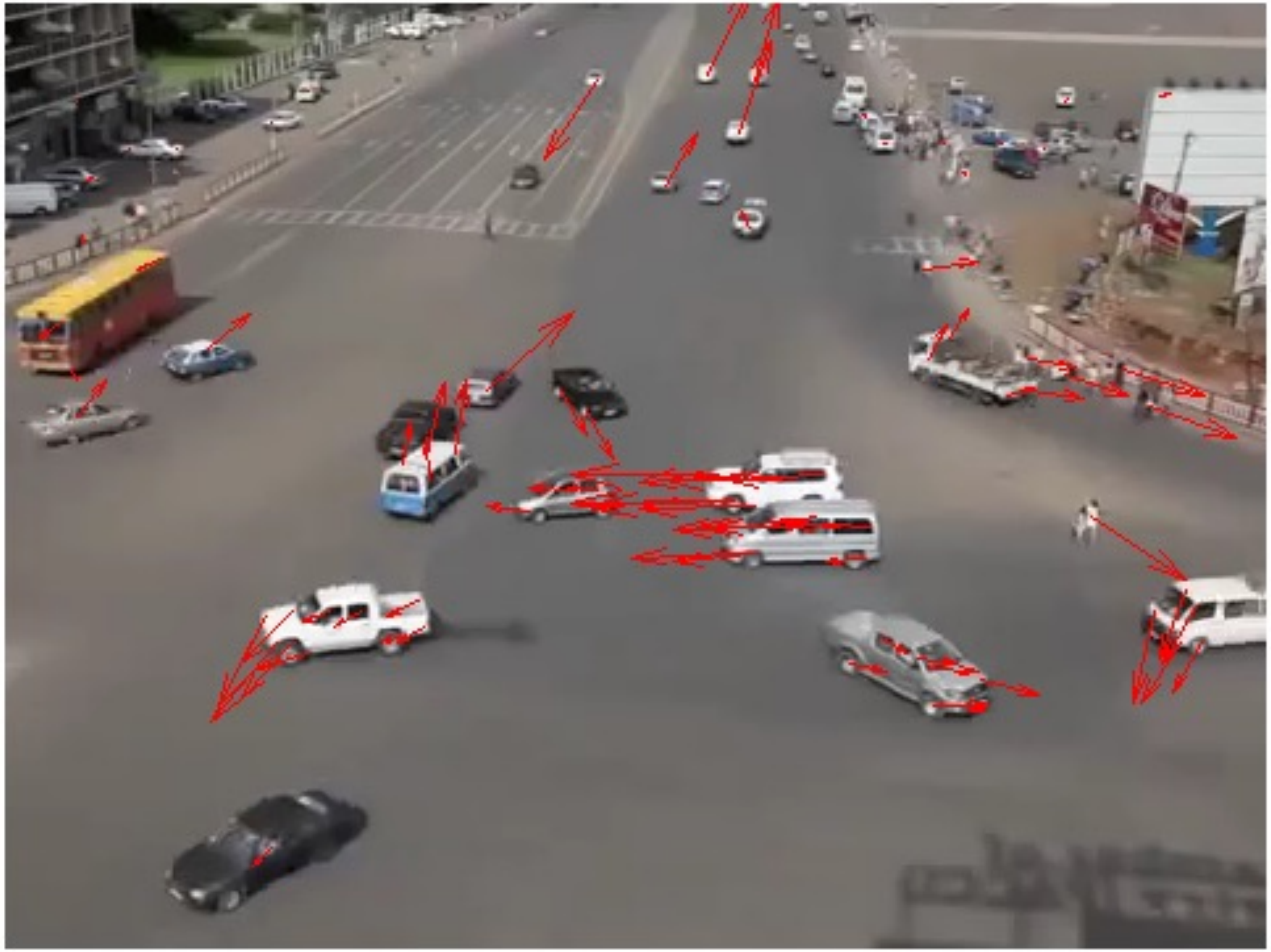
От изображений к видео

- Видео - это последовательность кадров, захваченных с течением времени.
- Теперь наши данные изображения являются функцией пространства (x, y) и времени (t).



Почему движение полезно?

Почему движение полезно?



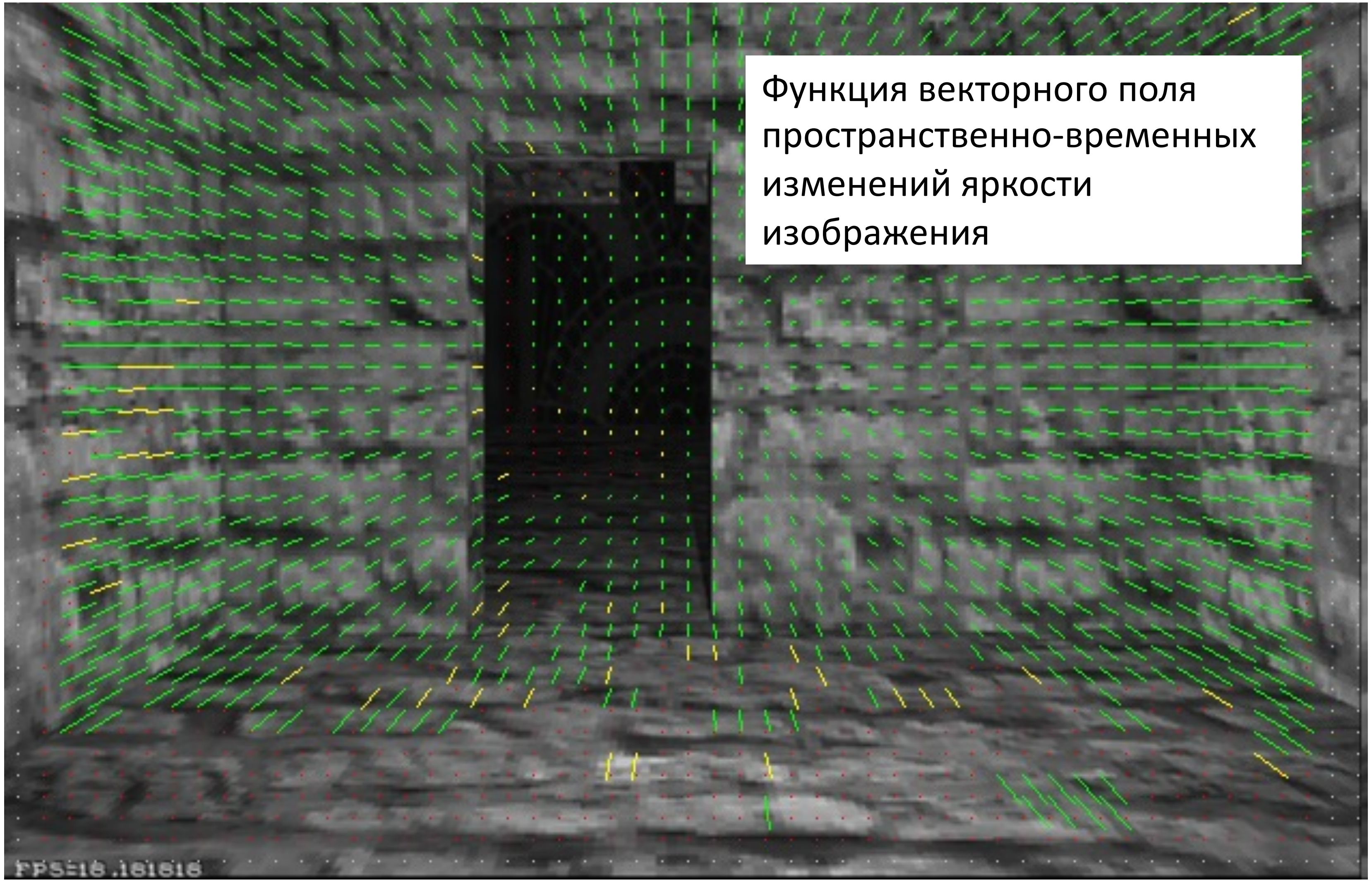
Оптический поток

- Определение: оптический поток - это видимое движение шаблонов яркости на изображении.

- Примечание: видимое движение может быть вызвано изменением освещения без какого-либо фактического движения.
  - Подумайте о равномерно вращающейся сфере при неподвижном освещении по сравнению со стационарной сферой при движущемся освещении.

Цель: восстановить движение изображения для каждого пикселя из оптического потока

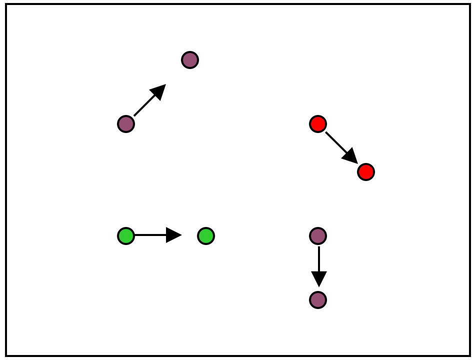
Оптический поток



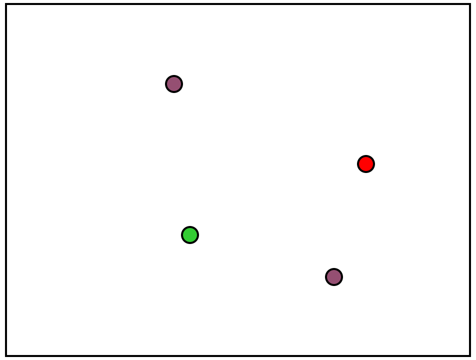
Функция векторного поля пространственно-временных изменений яркости изображения

Picture courtesy of Selim Temizer - Learning and Intelligent Systems (LIS) Group, MIT

Оценка оптического потока



$I(x,y,t-1)$

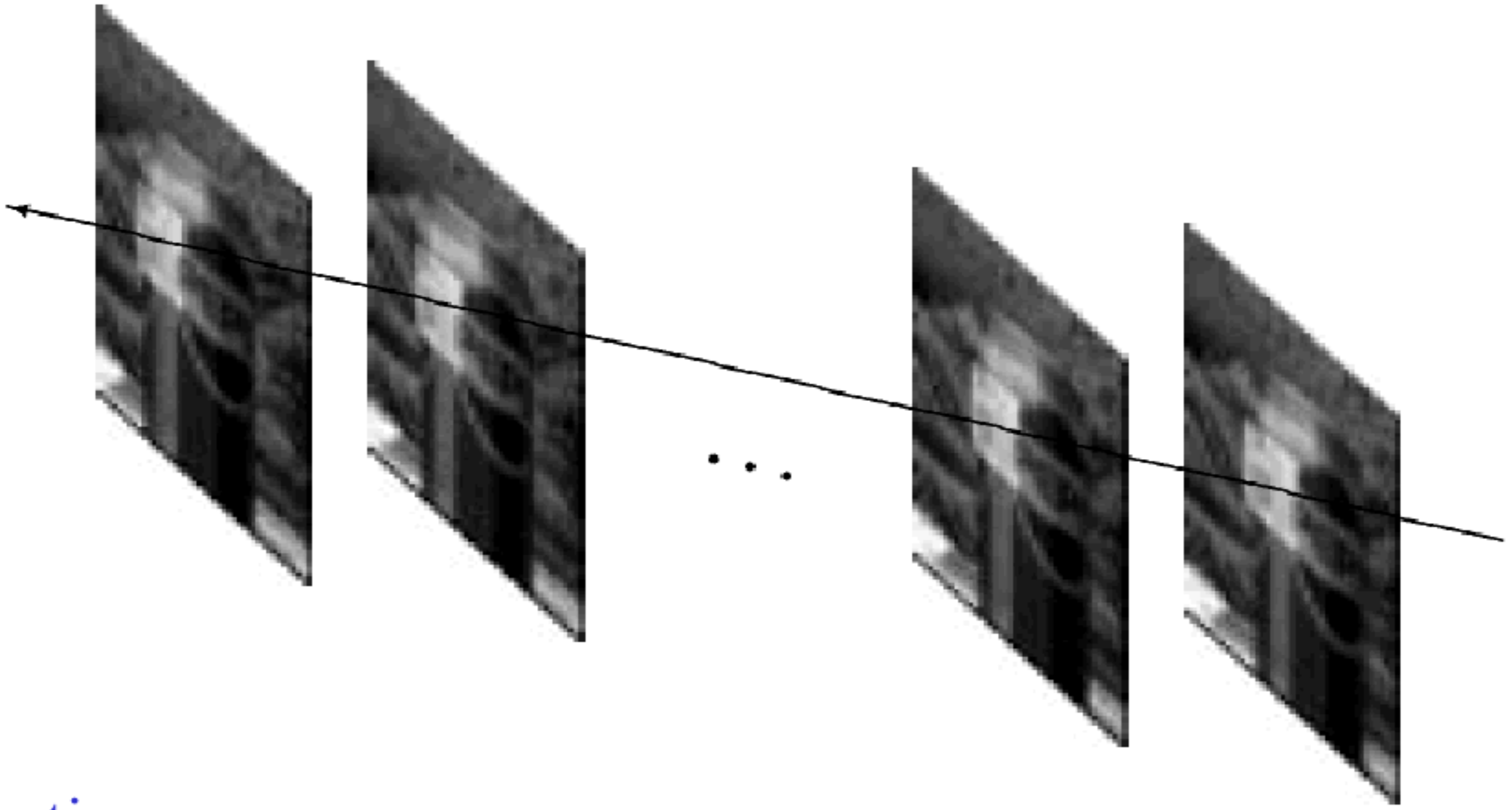


$I(x,y,t)$

- Учитывая два последующих кадра, оценим видимое поле движения u(x,y), v(x,y) между ними

- Ключевые допущения
  - **Небольшое движение:** точки уходят не очень далеко
  - **Постоянство яркости:** проекция одной и той же точки выглядит одинаково на каждом кадре.е
  - **Пространственная когерентность:** точки перемещаются, как их соседи

Ключевые допущения: небольшое изменение

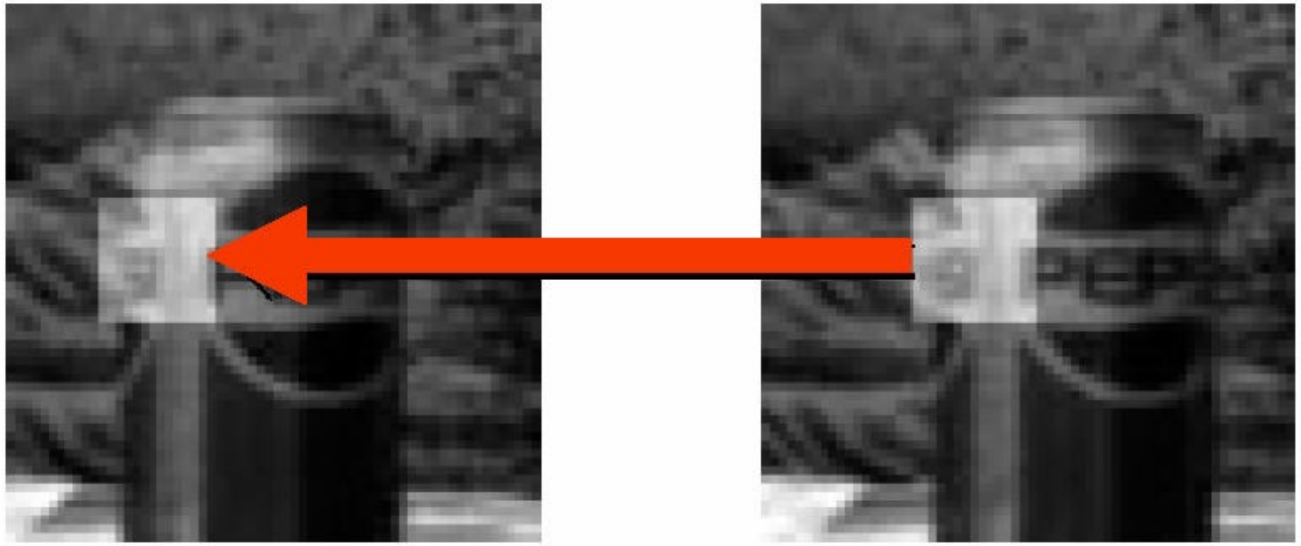


Assumption:

The image motion of a surface patch changes gradually over time.

* Slide from Michael Black, CS143 2003

Ключевые допущения: постоянство яркости



Assumption

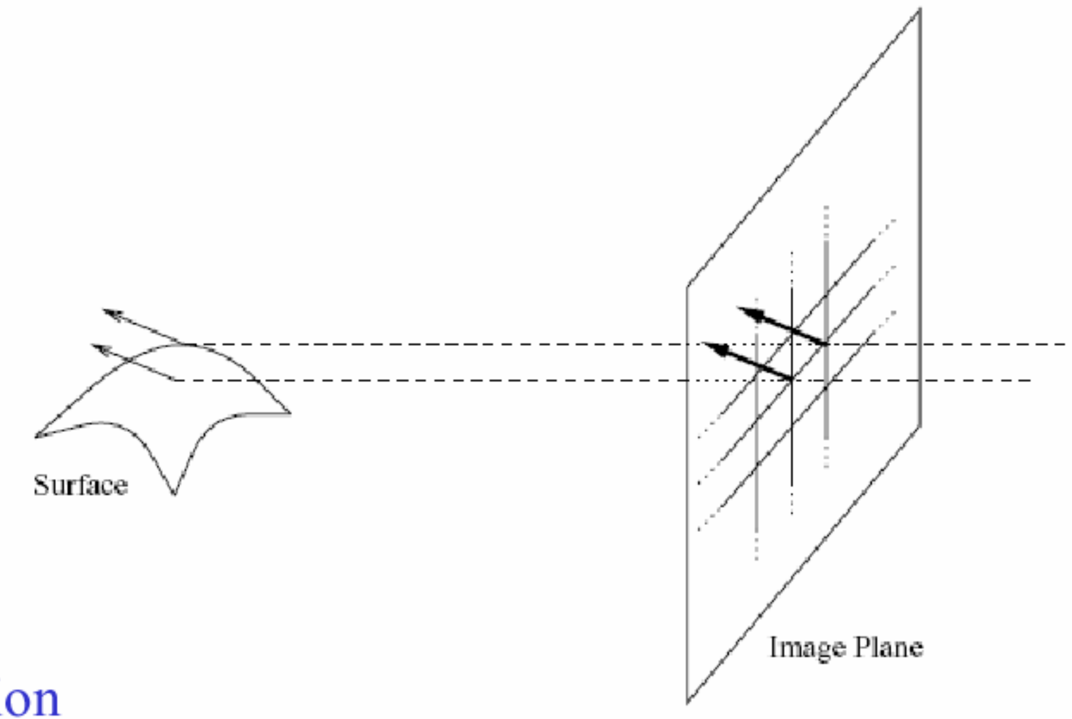
Image measurements (e.g. brightness) in a small region remain the same although their location may change.

$$I(x, y, t-1) = I(x + u(x, y), y + v(x, y), t)$$

(assumption)

* Slide from Michael Black, CS143 2003

Ключевые допущения: пространственная когерентность

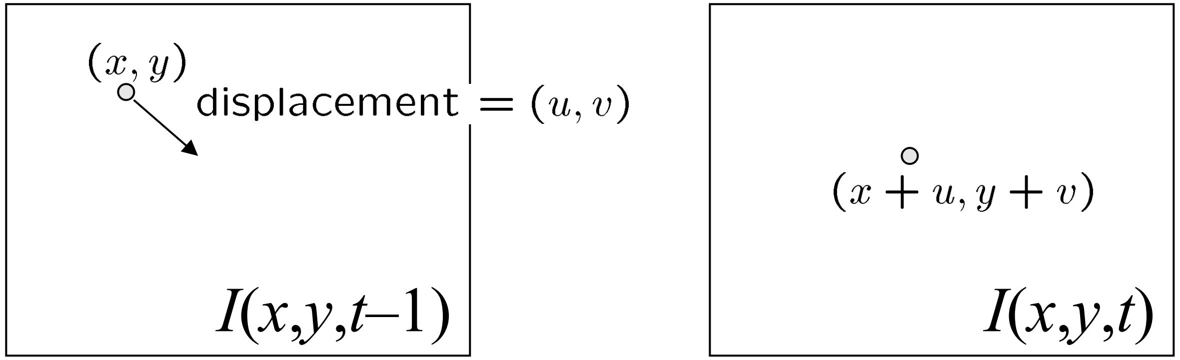


Assumption

* Neighboring points in the scene typically belong to the same surface and hence typically have similar motions.
* Since they also project to nearby points in the image, we expect spatial coherence in image flow.

* Slide from Michael Black, CS143 2003

Постоянство яркости



Уравнение Brightness Constancy:

$$I(x + u\delta t, y + v\delta t, t + \delta t) = I(x, y, t)$$

Линеаризация правой части уравнения:

$$I(x, y, t) + \frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t = I(x, y, t)$$

assuming small motion

$$\frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t = 0$$

divide by δt

take limit $\delta t \to 0$

$$\boxed{\frac{\partial I}{\partial x}\frac{dx}{dt} + \frac{\partial I}{\partial y}\frac{dy}{dt} + \frac{\partial I}{\partial t} = 0}$$

Brightness Constancy Equation

Представление оптического потока

flow velocities

$$I_x u + I_y v + I_t = 0$$

Image gradients
(at a point p)

temporal gradient

How do you compute …

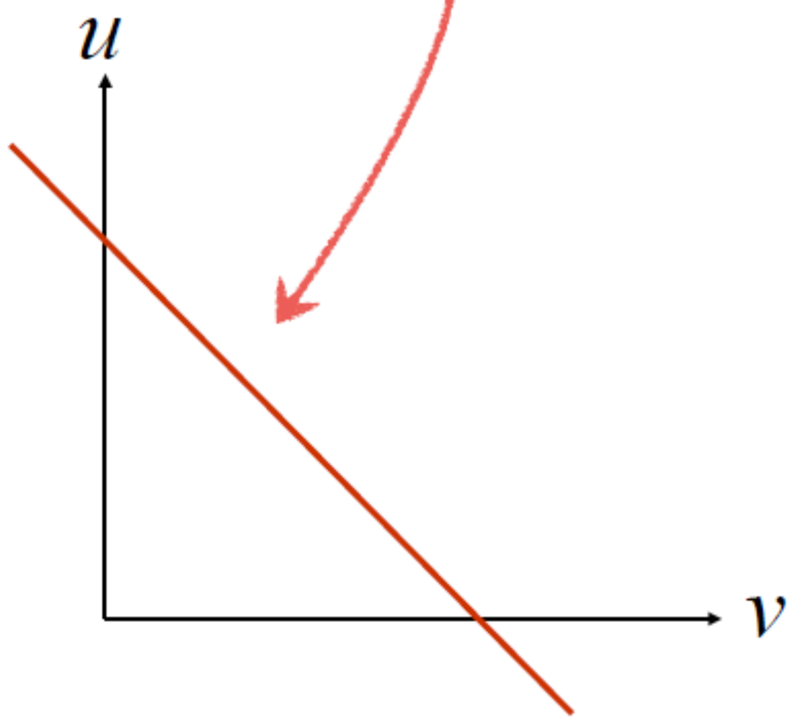
$I_x = \frac{\partial I}{\partial x} \quad I_y = \frac{\partial I}{\partial y}$	$u = \frac{dx}{dt} \quad v = \frac{dy}{dt}$	$I_t = \frac{\partial I}{\partial t}$
spatial derivative	**optical flow**	**temporal derivative**
Forward difference Sobel filter Scharr filter	**We need to solve for this!** (this is the unknown in the optical flow problem)	frame differencing

Представление оптического потока

Solution lies on a straight line

$$\boxed{I_x u + I_y v + I_t = 0}$$

many combinations of u and v will satisfy the equality



Что будем изучать сегодня

- Оптический поток
- Lucas-Kanade method
- Horn-Schunk method
- Пирамиды для большого движения
- Общая подход
- Применение

Solving the ambiguity…

- How to get more equations for a pixel?
- **Инвариант пространственной когерентности:**
- Соседние пиксели должны иметь схожие (u,v)
  - Если мы используем окно 5x5, это дает нам 25 уравнений от каждого пикселя

$$0 = I_t(\mathbf{p_i}) + \nabla I(\mathbf{p_i}) \cdot [u \ \ v]$$

$$\begin{bmatrix} I_x(\mathbf{p_1}) & I_y(\mathbf{p_1}) \\ I_x(\mathbf{p_2}) & I_y(\mathbf{p_2}) \\ \vdots & \vdots \\ I_x(\mathbf{p_{25}}) & I_y(\mathbf{p_{25}}) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} I_t(\mathbf{p_1}) \\ I_t(\mathbf{p_2}) \\ \vdots \\ I_t(\mathbf{p_{25}}) \end{bmatrix}$$

B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 674–679, 1981.

Lucas-Kanade flow

- Расширенная линейная система:

$$\begin{bmatrix} I_x(\mathbf{p_1}) & I_y(\mathbf{p_1}) \\ I_x(\mathbf{p_2}) & I_y(\mathbf{p_2}) \\ \vdots & \vdots \\ I_x(\mathbf{p_{25}}) & I_y(\mathbf{p_{25}}) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} I_t(\mathbf{p_1}) \\ I_t(\mathbf{p_2}) \\ \vdots \\ I_t(\mathbf{p_{25}}) \end{bmatrix} \qquad \underset{25\times 2}{A} \; \underset{2\times 1}{d} = \underset{25\times 1}{b}$$

Lucas-Kanade flow

- Расширенная линейная система

$$\begin{bmatrix} I_x(\mathbf{p_1}) & I_y(\mathbf{p_1}) \\ I_x(\mathbf{p_2}) & I_y(\mathbf{p_2}) \\ \vdots & \vdots \\ I_x(\mathbf{p_{25}}) & I_y(\mathbf{p_{25}}) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} I_t(\mathbf{p_1}) \\ I_t(\mathbf{p_2}) \\ \vdots \\ I_t(\mathbf{p_{25}}) \end{bmatrix} \qquad \underset{25\text{x}2}{A} \ \underset{2\text{x}1}{d} = \underset{25\text{x}1}{b}$$

Метод наименьших квадратов $\qquad (A^T A)\ d = A^T b$

$$\underbrace{\begin{bmatrix} \sum I_x I_x & \sum I_x I_y \\ \sum I_x I_y & \sum I_y I_y \end{bmatrix}}_{A^T A} \begin{bmatrix} u \\ v \end{bmatrix} = - \underbrace{\begin{bmatrix} \sum I_x I_t \\ \sum I_y I_t \end{bmatrix}}_{A^T b}$$

Обобщение всех пикселей в окне K x K

Условия для разрешения уравнения

- Решение (u, v) удовлетворяет уравнению Lucas-Kanade

$$\underbrace{\begin{bmatrix} \sum I_x I_x & \sum I_x I_y \\ \sum I_x I_y & \sum I_y I_y \end{bmatrix}}_{A^T A} \begin{bmatrix} u \\ v \end{bmatrix} = - \underbrace{\begin{bmatrix} \sum I_x I_t \\ \sum I_y I_t \end{bmatrix}}_{A^T b}$$

Когда эта система разрешима?

- **$A^T A$** должно быть обратима
- **$A^T A$** не должна быть маленькой из-за шума
  - собственные числа λ_1 и λ_2 матрицы **$A^T A$** не должны быть маленькими
- **$A^T A$** должно быть хорошо разрешимо
  - λ_1 / λ_2 должны быть небольшие (λ_1 = larger eigenvalue)

Ничего не напоминает?

M = A^TA это матрица вторых моментов!

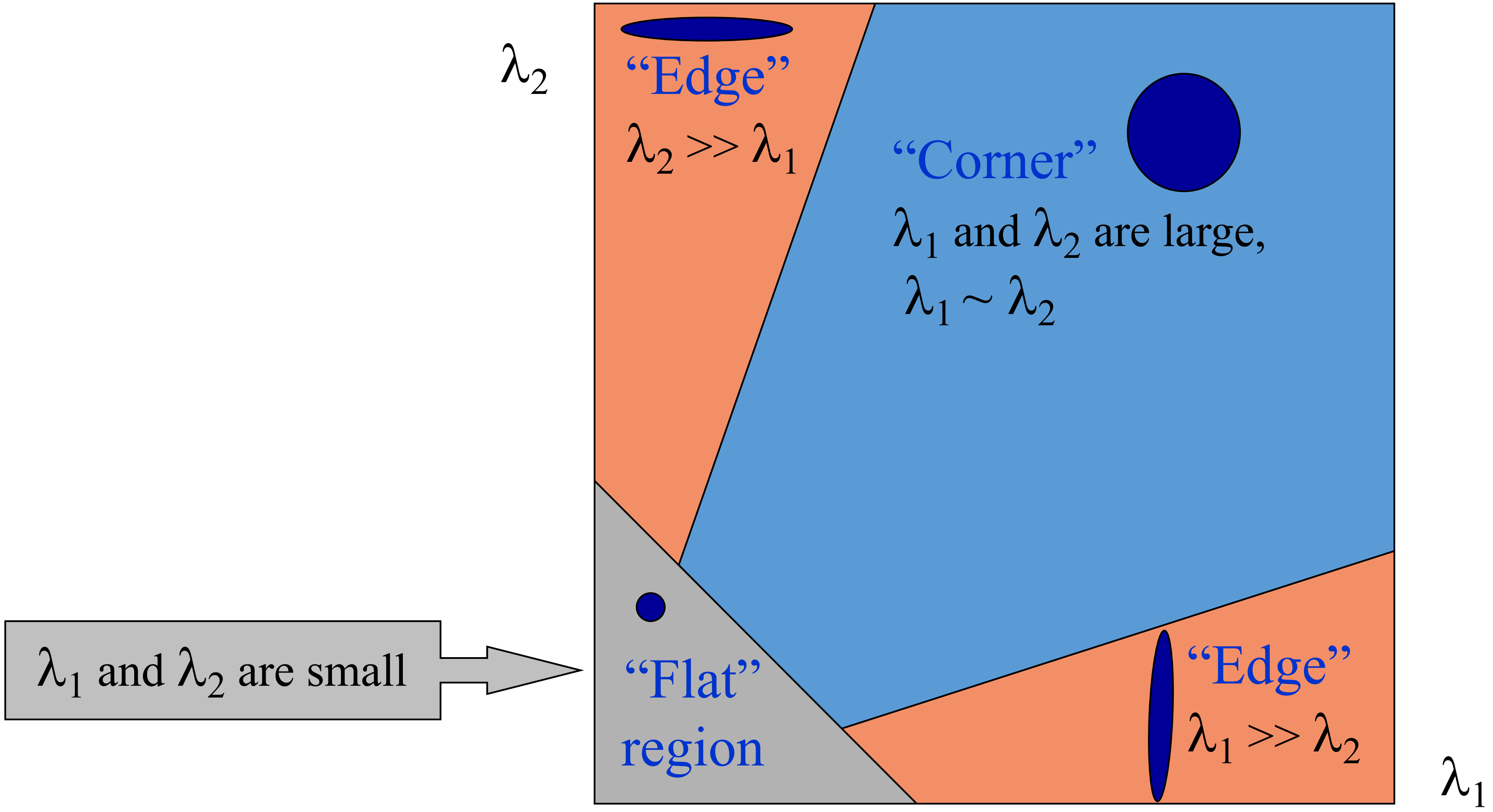
(Harris corner detector…)

$$A^T A = \left[\begin{array}{cc} \sum I_x I_x & \sum I_x I_y \\ \sum I_x I_y & \sum I_y I_y \end{array} \right] = \sum \left[\begin{array}{c} I_x \\ I_y \end{array} \right] [I_x \; I_y] = \sum \nabla I (\nabla I)^T$$

- Собственные вектора и значения матрицы A^TA определяют направление и амплитуду движения
  - Собственный вектор, связанный с большим собственным значением, указывает в направлении наиболее быстрого изменения интенсивности
  - Другой собственный вектор ортогонален ему

Интерпретация собственных чисел

Классификация точки по собственным значениям матрицы моментов:



Граница

$$\sum \nabla I (\nabla I)^T$$

– градиент очень большой или очень маленький

– большое λ_1, маленькое λ_2

Регион с слабой текстурой

$$\sum \nabla I (\nabla I)^T$$

– градиент имеет маленькую амплитуду

– маленькое λ_1, маленькое λ_2

Регион с сильной текстурой

$$\sum \nabla I (\nabla I)^T$$

– градиенты в разные стороны, большая амплитуда

– большое λ_1, большое λ_2

Ошибки в методе Lukas-Kanade

What are the potential causes of errors in this procedure?

- Suppose $A^T A$ is easily invertible
- Suppose there is not much noise in the image

- Когда нарушаются ограничения
  - Яркость **не** остается постоянной во времени
  - **Большое** изменение движения
  - Соседние точки ведут себя по-разному
    - окно слишком большое
    - какой оптимальный размер окна?

* From Khurram Hassan-Shafique CAP5415 Computer Vision 2003

Улучшение модели

- Разложение в ряд:

$$0 = I(x+u, y+v) - \boldsymbol{I_{t\text{-}1}(x,y)}$$

$$\approx I(x,y) + I_x u + I_y v - \boldsymbol{I_{t\text{-}1}(x,y)}$$

- Это не очень точно
  - Для повышения точности при разложении нужны члены высокого порядка:

$$= I(x,y) + I_x u + I_y v + \text{higher order terms} - \boldsymbol{I_{t\text{-}1}(x,y)}$$

- Теперь встает проблема поиска решения на (u, v) – нелинейная система:
  - Можно разрешить с помощью метода Ньютона
  - В методе Lukas-Kanade применяется одна итерация метода Ньютона:
    - Лучше результат, чем больше итераций

* From Khurram Hassan-Shafique CAP5415 Computer Vision 2003

Iterative Refinement

- Итеративный алгоритм Lukas-Kanade
    1. Оценить поток для каждого пикселя, решая уравнение Lucas-Kanade
    2. Деформация I(t-1) в направлении I(t) с использованием расчетного поля потока
        1. С помощью методов искажения изображений
    3. Решить до конца последовательности

* From Khurram Hassan-Shafique CAP5415 Computer Vision 2003

Что будем изучать сегодня

- Оптический поток
- Lucas-Kanade method
- Horn-Schunk method
- Пирамиды для большого движения
- Общая подход
- Применение

Horn-Schunk method

- Поток сформулирован как глобальная энергетическая функция, которая должна быть минимизирована:

$$E = \iint \left[(I_x u + I_y v + I_t)^2 + \alpha^2 (\|\nabla u\|^2 + \|\nabla v\|^2) \right] \mathrm{d}x\mathrm{d}y$$

Horn-Schunk method

- Поток сформулирован как глобальная энергетическая функция, которая должна быть минимизирована:
- Первая часть функции – изменение яркости.

$$E = \iint \left[\boxed{(I_x u + I_y v + I_t)}^2 + \alpha^2 (\|\nabla u\|^2 + \|\nabla v\|^2) \right] \mathrm{d}x \mathrm{d}y$$

Horn-Schunk method for optical flow

- Поток сформулирован как глобальная энергетическая функция, которая должна быть минимизирована :
- Вторая часть – регуляризация потока. Она пытается сделать так, чтобы изменения между пикселями были небольшими.

$$E = \iint \left[(I_x u + I_y v + I_t)^2 + \alpha^2 \boxed{\left(\|\nabla u\|^2 + \|\nabla v\|^2 \right)} \right] \mathrm{d}x \mathrm{d}y$$

Horn-Schunk method

- Поток сформулирован как глобальная энергетическая функция, которая должна быть минимизирована :
- α – масштаб регуляризации. Большие значения α делают более «гладким».

$$E = \iint \left[(I_x u + I_y v + I_t)^2 + \boxed{\alpha^2} (\|\nabla u\|^2 + \|\nabla v\|^2) \right] \mathrm{d}x \mathrm{d}y$$

Horn-Schunk method

- Поток сформулирован как глобальная энергетическая функция, которая должна быть минимизирована:

$$E = \iint \left[(I_x u + I_y v + I_t)^2 + \alpha^2 (\|\nabla u\|^2 + \|\nabla v\|^2) \right] \mathrm{d}x\mathrm{d}y$$

- Задачу минимизации можно решить, взяв производную по отношению к u и v. Получим следующие уравнения :

$$\frac{\partial L}{\partial u} - \frac{\partial}{\partial x}\frac{\partial L}{\partial u_x} - \frac{\partial}{\partial y}\frac{\partial L}{\partial u_y} = 0$$
$$\frac{\partial L}{\partial v} - \frac{\partial}{\partial x}\frac{\partial L}{\partial v_x} - \frac{\partial}{\partial y}\frac{\partial L}{\partial v_y} = 0$$

$$I_x(I_x u + I_y v + I_t) - \alpha^2 \Delta u = 0$$
$$I_y(I_x u + I_y v + I_t) - \alpha^2 \Delta v = 0$$

Horn-Schunk method

- Производные по отношению к u и v получаются:

$$I_x(I_x u + I_y v + I_t) - \alpha^2 \Delta u = 0$$
$$I_y(I_x u + I_y v + I_t) - \alpha^2 \Delta v = 0$$

- Где $\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$ is. На практике его считают так:

$$\Delta u(x, y) = \overline{u}(x, y) - u(x, y)$$

- Здесь $\overline{u}(x, y)$ едневзвешенное значение u, измеренное на (x,y).

Horn-Schunk method

- Теперь подставим $\Delta u(x,y) = \overline{u}(x,y) - u(x,y)$

$$I_x(I_x u + I_y v + I_t) - \alpha^2 \Delta u = 0$$
$$I_y(I_x u + I_y v + I_t) - \alpha^2 \Delta v = 0$$

- Получим:

$$(I_x^2 + \alpha^2)u + I_x I_y v = \alpha^2 \overline{u} - I_x I_t$$
$$I_x I_y u + (I_y^2 + \alpha^2)v = \alpha^2 \overline{v} - I_y I_t$$

- Система является линейной для u и v и может быть решена аналитически для каждого пикселя в отдельности.

Iterative Horn-Schunk

- Но так как решение зависит от соседних значений поля потока, то его необходимо повторить после обновления соседей.

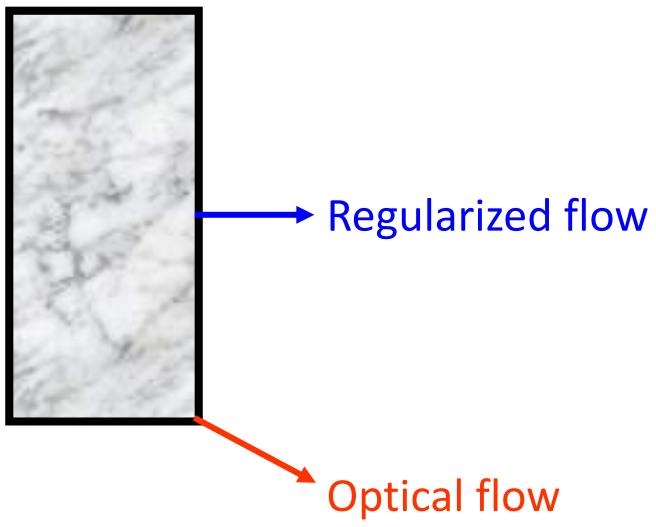
- Так что вместо этого, мы можем итеративно решать для u и v, используя:

$$u^{k+1} = \overline{u}^k - \frac{I_x(I_x\overline{u}^k + I_y\overline{v}^k + I_t)}{\alpha^2 + I_x^2 + I_y^2}$$

$$v^{k+1} = \overline{v}^k - \frac{I_y(I_x\overline{u}^k + I_y\overline{v}^k + I_t)}{\alpha^2 + I_x^2 + I_y^2}$$

Что вообще делает регуляризация потока?

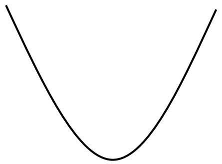
- Это сумма квадратов (евклидовая мера расстояния).
- Мы помещаем это в выражение, чтобы свести к минимуму.
- => В областях, свободных от текстуры, нет оптического потока.
- *=> По рёбрам точки будут стекаться к ближайшим точкам, решая aperture problem.*



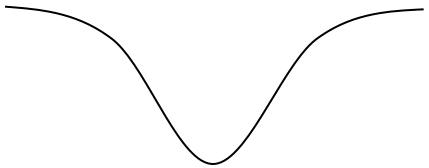
Плотный оптический поток по Michael Black’s method

- Майкл Блэк продвинул метод Хорн-Шанка на шаг дальше, начав с константы регуляризации. :
- Которая выглядит, как квадрат:

$$\|\nabla u\|^2 + \|\nabla v\|^2$$



- И заменил его этим:



- Почему эта регуляризация работает лучше?

Что будем изучать сегодня

- Оптический поток
- Lucas-Kanade method
- Horn-Schunk method
- Пирамиды для большого движения
- Общая подход
- Применение

Повторение

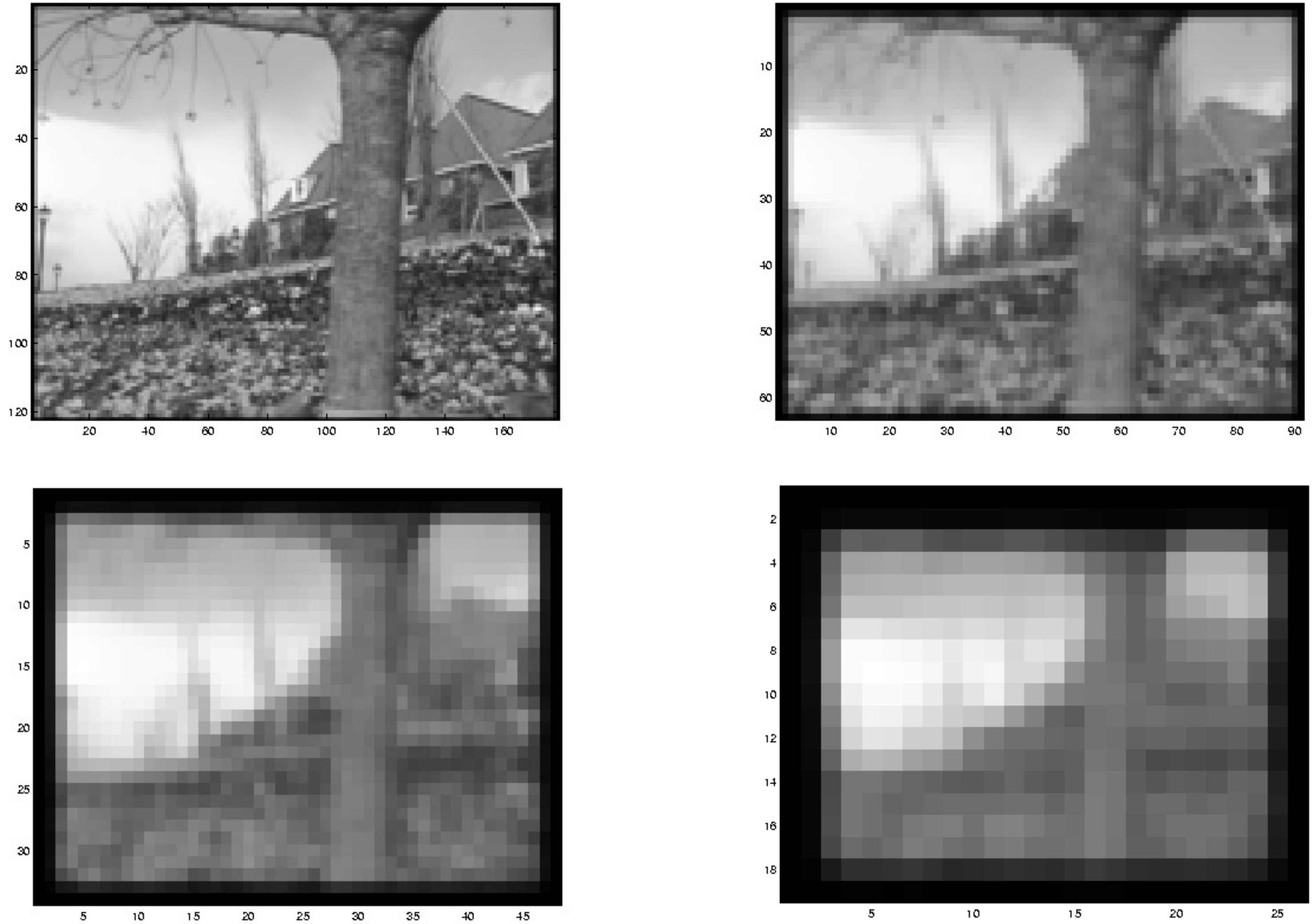
- Ключевые допущения (Ошибки в Lucas-Kanade)
  - **Небольшое движение:** точки уходят не очень далеко
  - **Постоянство яркости:** проекция одной и той же точки выглядит одинаково на каждом кадре.е
  - **Пространственная когерентность:** точки перемещаются, как их соседи

Пересмотр предположения о малом движении

- Это движение достаточно маленькое?
  - Наверное, не настолько. Это намного больше одного пикселя (доминируют термины 2-го порядка).
  - Как мы можем решить эту проблему?

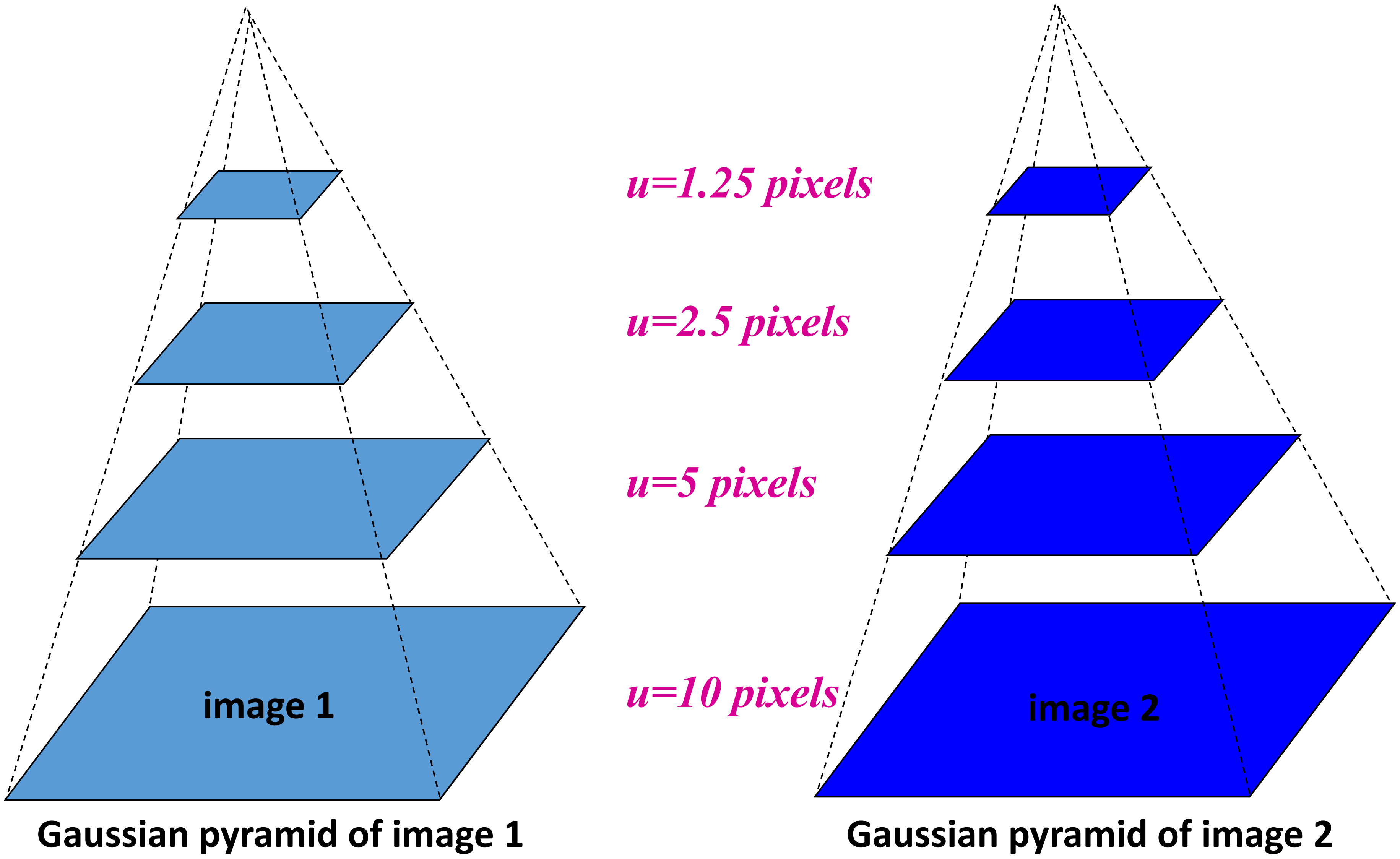
* From Khurram Hassan-Shafique CAP5415 Computer Vision 2003

Уменьшим разрешение

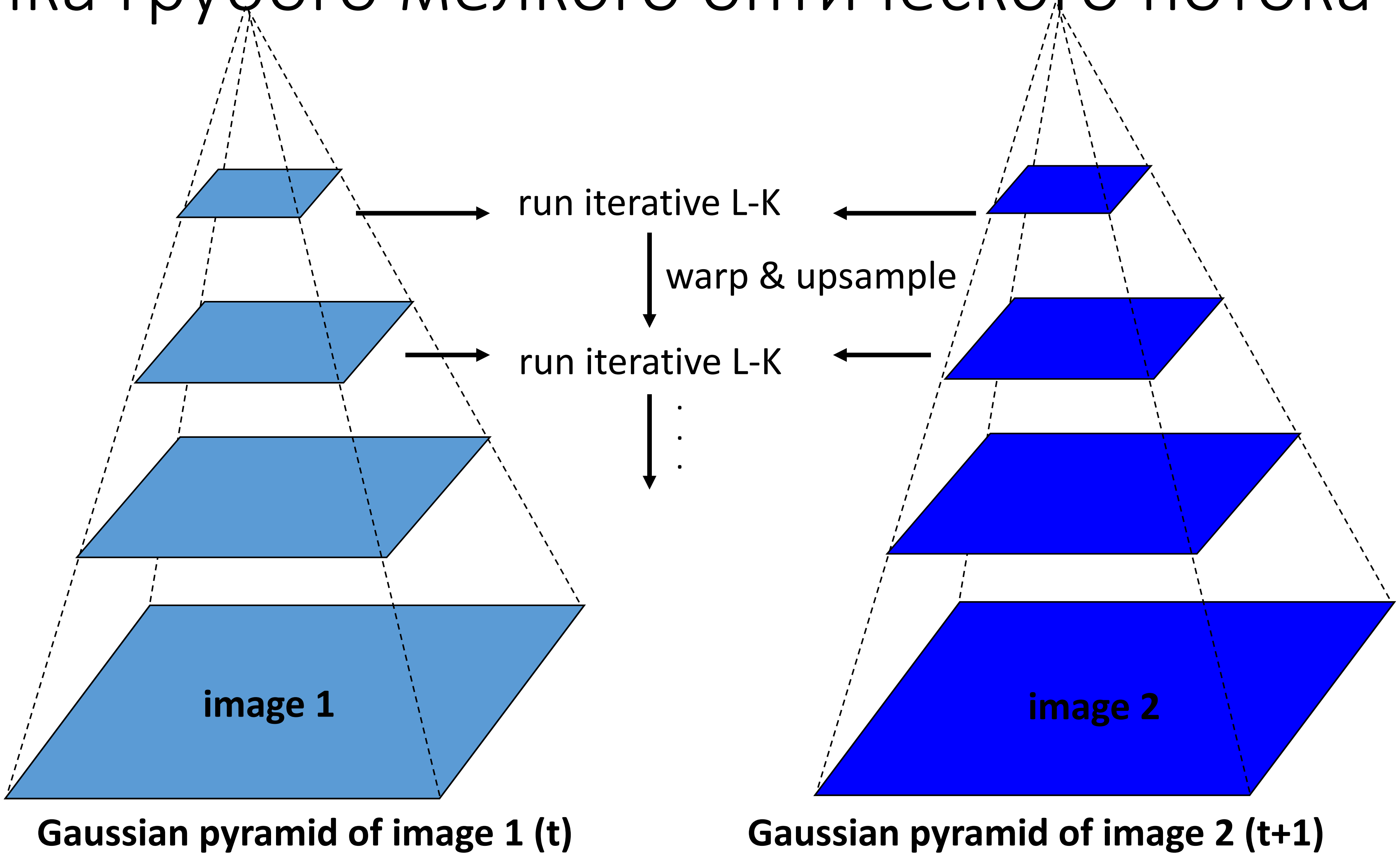


* From Khurram Hassan-Shafique CAP5415 Computer Vision 2003

Оценка грубого мелкого оптического потока

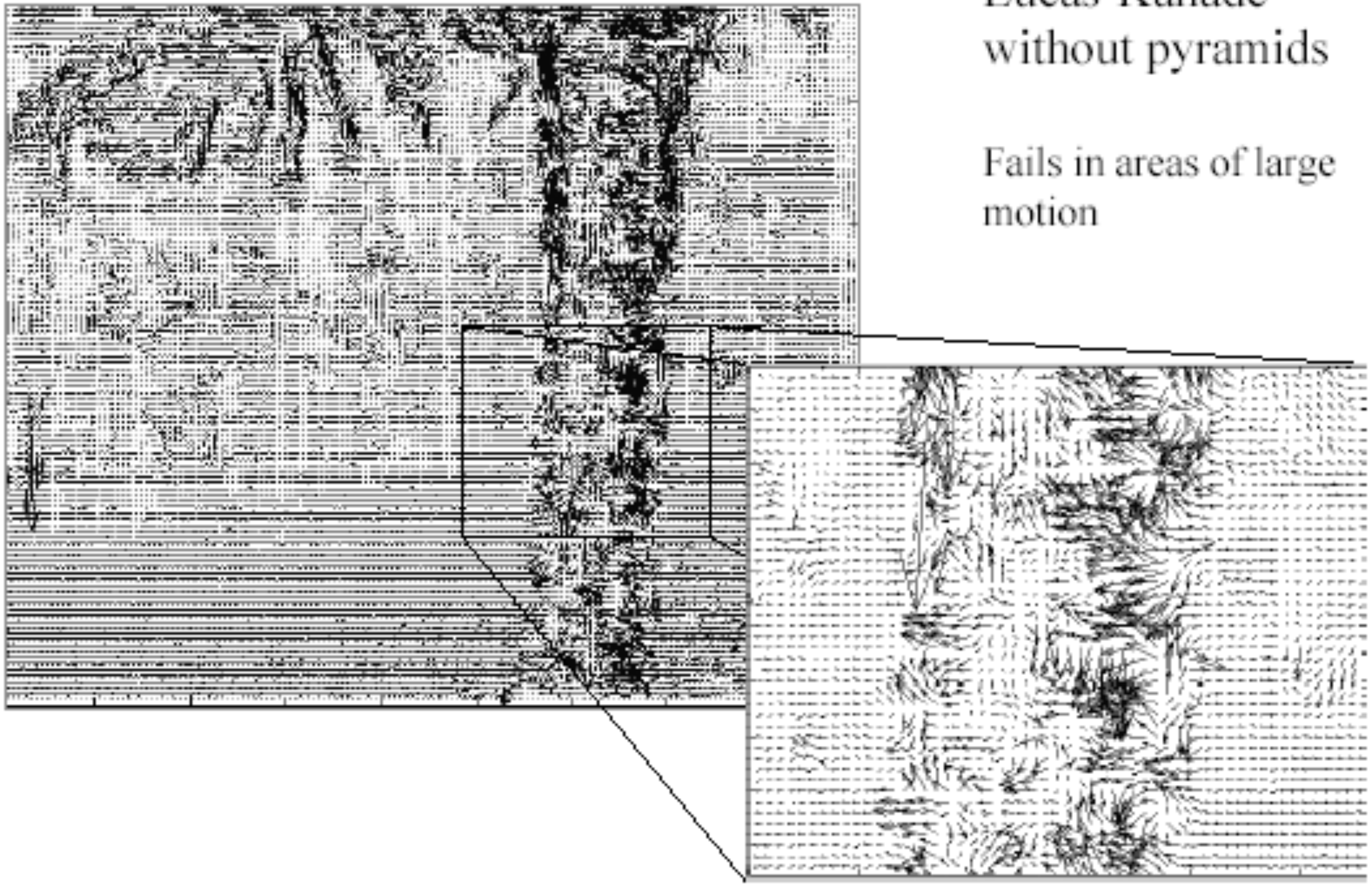


Оценка грубого мелкого оптического потока



Source: Silvio Savarese

Результаты без пирамид

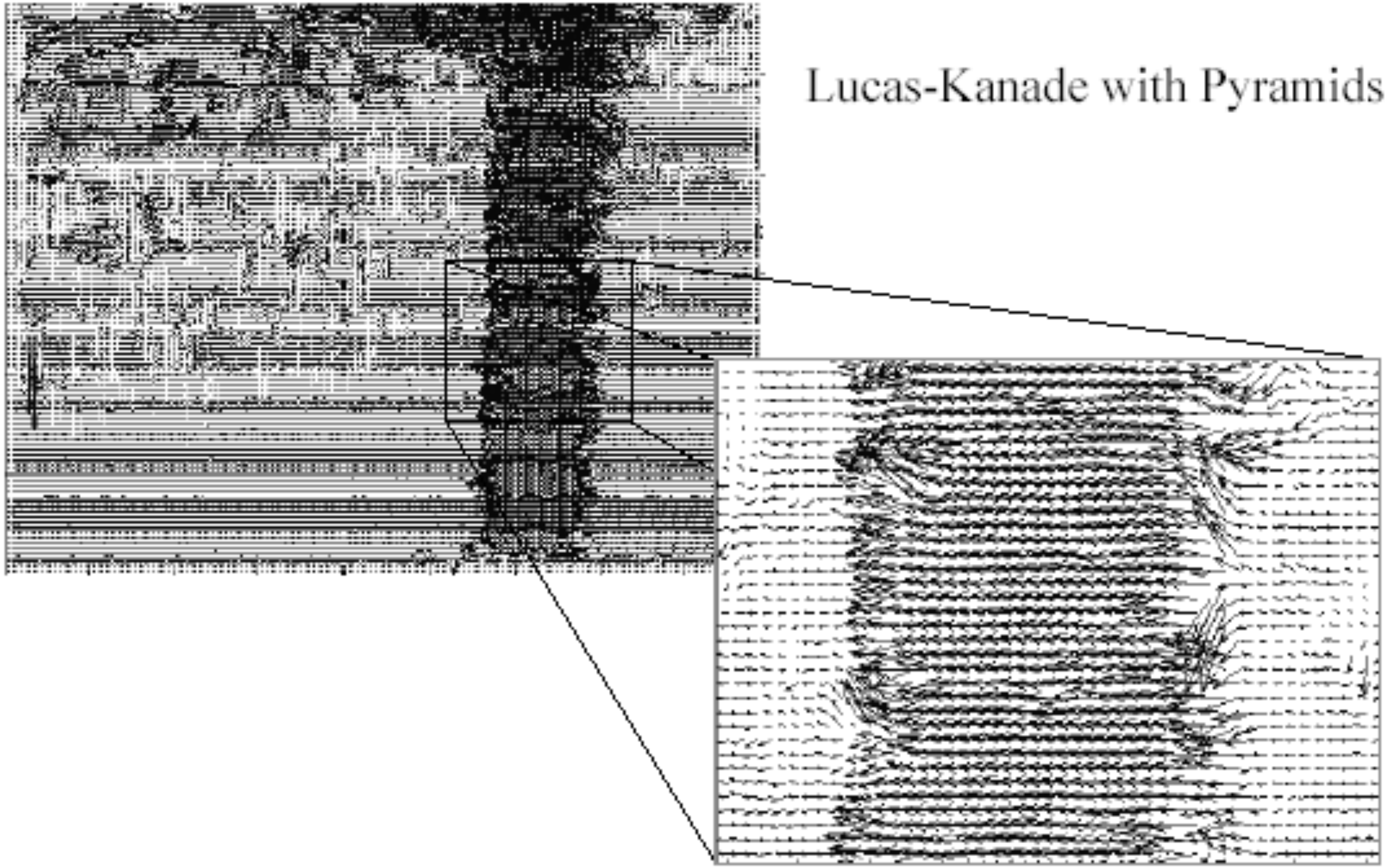


Lucas-Kanade
without pyramids

Fails in areas of large
motion

* From Khurram Hassan-Shafique CAP5415 Computer Vision 2003

Результаты с пирамидами



Что будем изучать сегодня

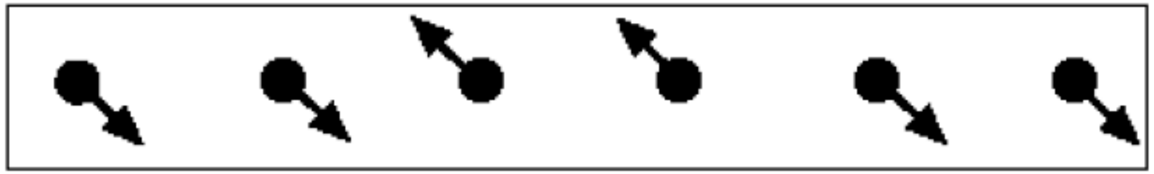
- Оптический поток
- Lucas-Kanade method
- Horn-Schunk method
- Пирамиды для большого движения
- Общая подход
- Применение

Повторение

- Ключевые допущения (Ошибки в Lucas-Kanade)

  - **Небольшое движение:** точки уходят не очень далеко
  - **Постоянство яркости:** проекция одной и той же точки выглядит одинаково на каждом кадре.е
  - **Пространственная когерентность:** точки перемещаются, как их соседи

Общее направление движения



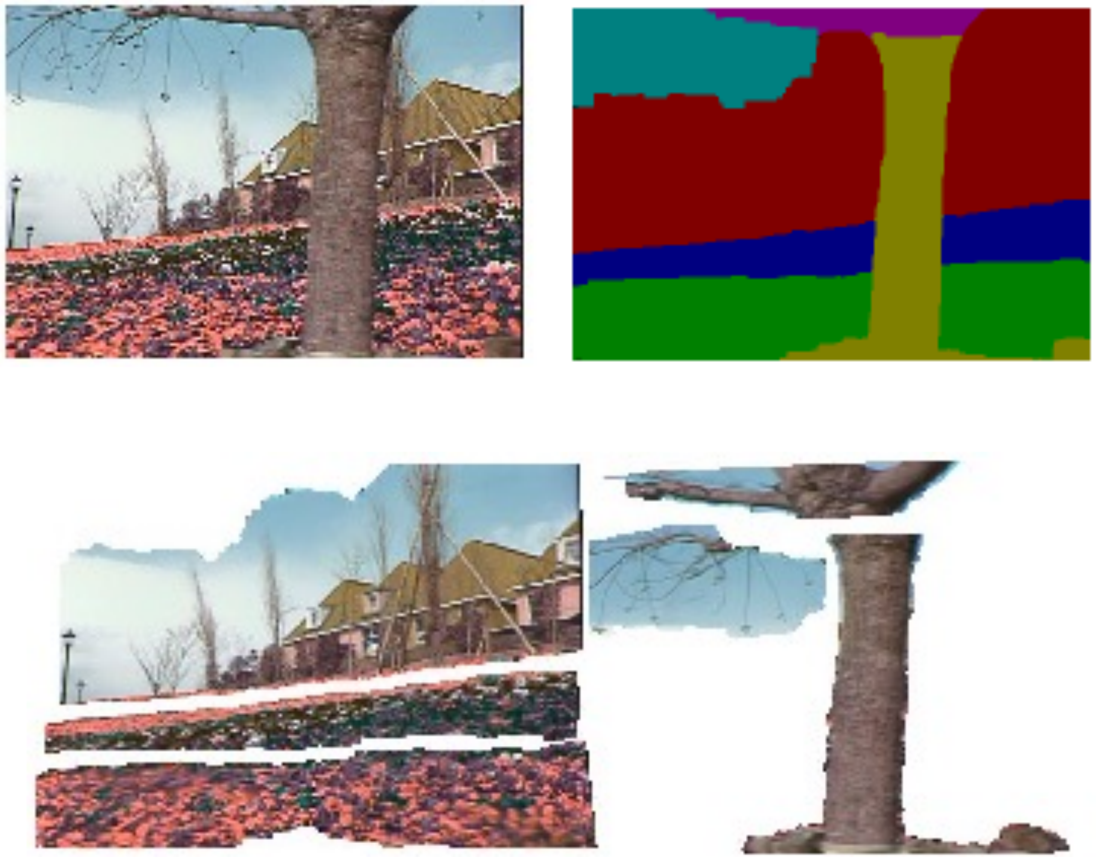
Люди склонны воспринимать элементы, движущиеся в одном направлении, как более связанные, чем стационарные или движущиеся в разных направлениях.

Сегментация движения

- Как мы представляем движение в этой сцене?

Сегментация движения

- Разбить последовательность изображений на "слои", каждый из которых имеет когерентное (аффинное) движение

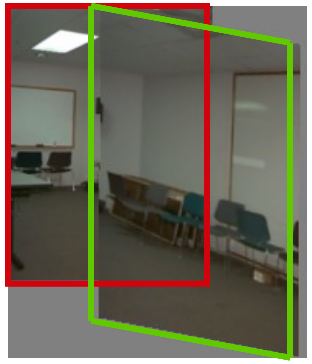


J. Wang and E. Adelson. Layered Representation for Motion Analysis. *CVPR 1993*.

Affine motion

$$u(x, y) = a_1 + a_2 x + a_3 y$$

$$v(x, y) = a_4 + a_5 x + a_6 y$$



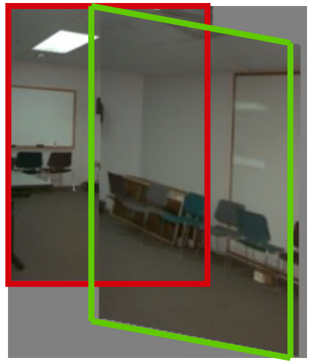
- Заменим в уравнении постоянства яркости:

$$\boxed{I_x \cdot u + I_y \cdot v + I_t \approx 0}$$

Affine motion

$$u(x,y) = a_1 + a_2 x + a_3 y$$

$$v(x,y) = a_4 + a_5 x + a_6 y$$



- Заменим в уравнении постоянства яркости :

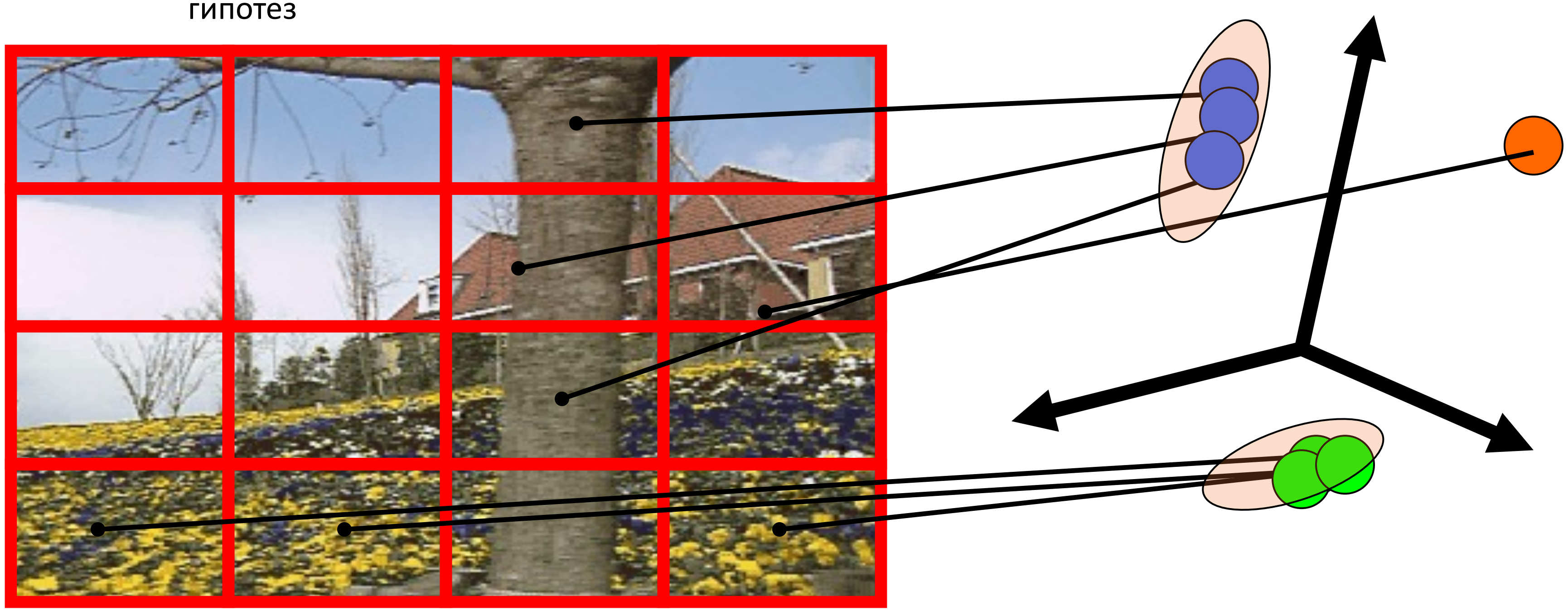
$$\boxed{I_x(a_1 + a_2 x + a_3 y) + I_y(a_4 + a_5 x + a_6 y) + I_t \approx 0}$$

- Каждый пиксель обеспечивает 1 линейное ограничение на 6 неизвестных
- Минимизация наименьших квадратов :

$$\boxed{Err(\vec{a}) = \sum \left[I_x(a_1 + a_2 x + a_3 y) + I_y(a_4 + a_5 x + a_6 y) + I_t \right]^2}$$

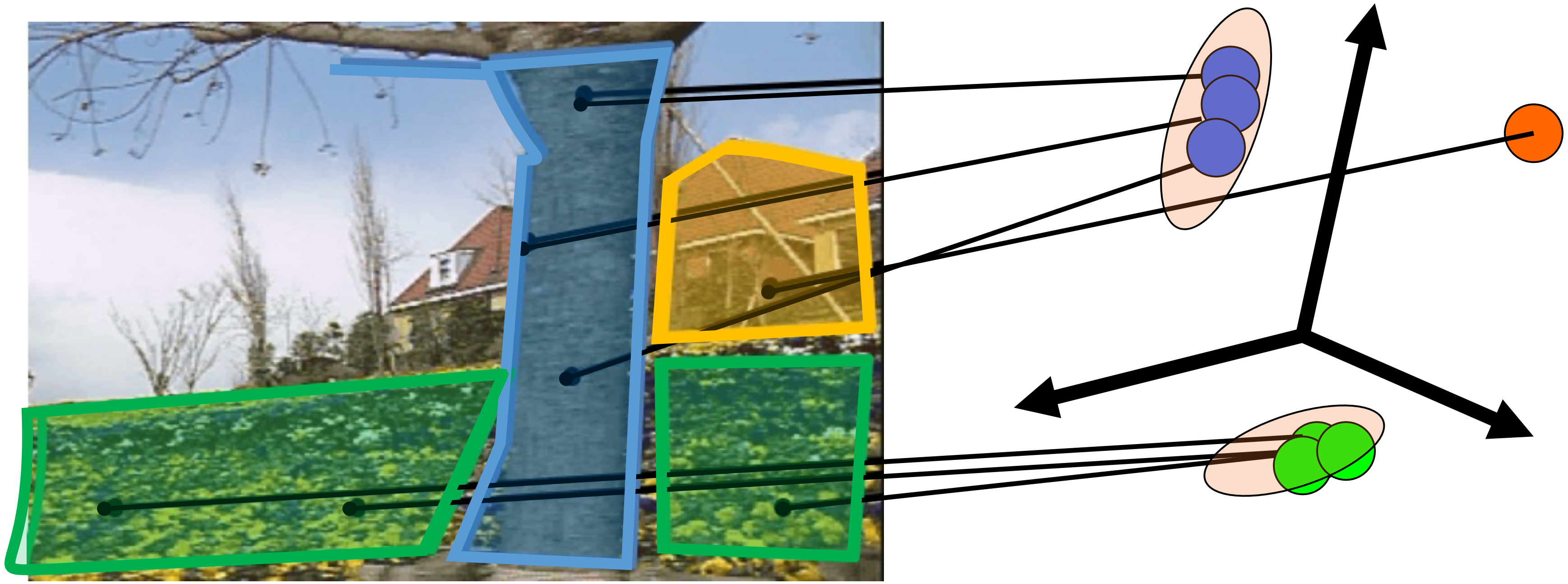
Как мы оцениваем слои?

- 1. Получим набор гипотез о affine motion
  - Разделить изображение на блоки и оценивать параметры affine motion в каждом из них по наименьшим квадратам
    - Исключить гипотезы с высокой ошибкой
  - Отобразить параметры движения в векторном пространстве
  - Сделать k-means кластеризацию на параметры affine motion
    - Соединяем кластеры, которые близки, чтобы добиться наименьшего количества гипотез



Как мы оцениваем слои?

- 1. Получим набор гипотез о affine motion
  - Разделить изображение на блоки и оценивать параметры affine motion в каждом из них по наименьшим квадратам
    - Исключить гипотезы с высокой ошибкой
  - Отобразить параметры движения в векторном пространстве
  - Сделать k-means кластеризацию на параметры affine motion
    - Соединяем кластеры, которые близки, чтобы добиться наименьшего количества гипотез



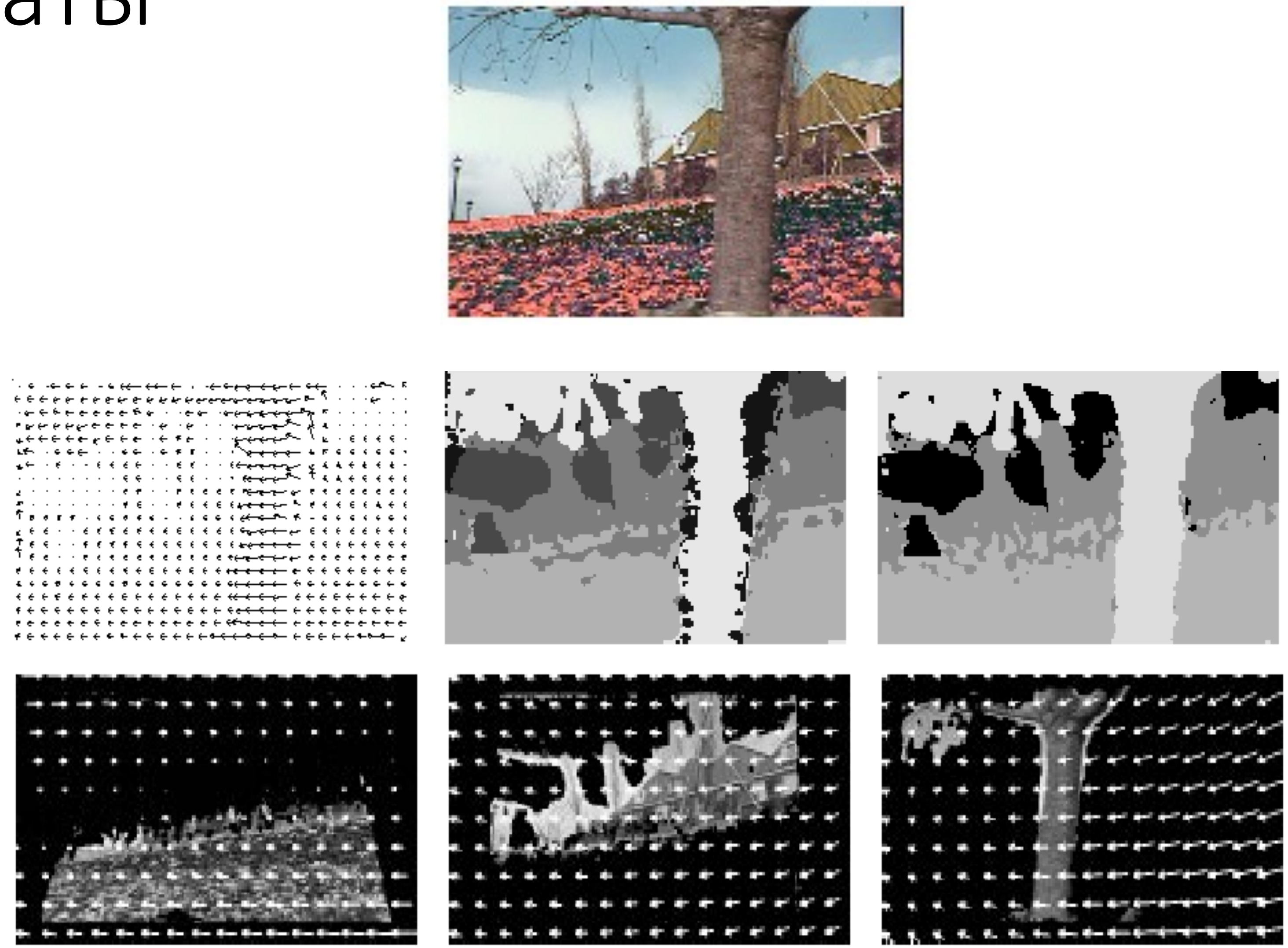
Как мы оцениваем слои?

- 1. Получим набор гипотез о affine motion
  - Разделить изображение на блоки и оценивать параметры affine motion в каждом из них по наименьшим квадратам
    - Исключить гипотезы с высокой ошибкой
  - Отобразить параметры движения в векторном пространстве
  - Сделать k-means кластеризацию на параметры affine motion
    - Соединяем кластеры, которые близки, чтобы добиться наименьшего количества гипотез

2. Повторить до сходимости:

- Отнести каждый пиксель к наилучшей гипотезе
  – Пиксели с высокой ошибкой остаются без гипотезы
- Фильтрация регионов для соблюдения пространственных ограничений
- Пересчитать оценку affine motions в каждом регионе

Результаты



J. Wang and E. Adelson. Layered Representation for Motion Analysis. *CVPR 1993*.

Sparse vs Dense Optical Flow

Разреженный оптический поток – отслеживание нескольких “характерных” пикселей

Плотный оптический поток – оценка потока всех пикселей в изображении

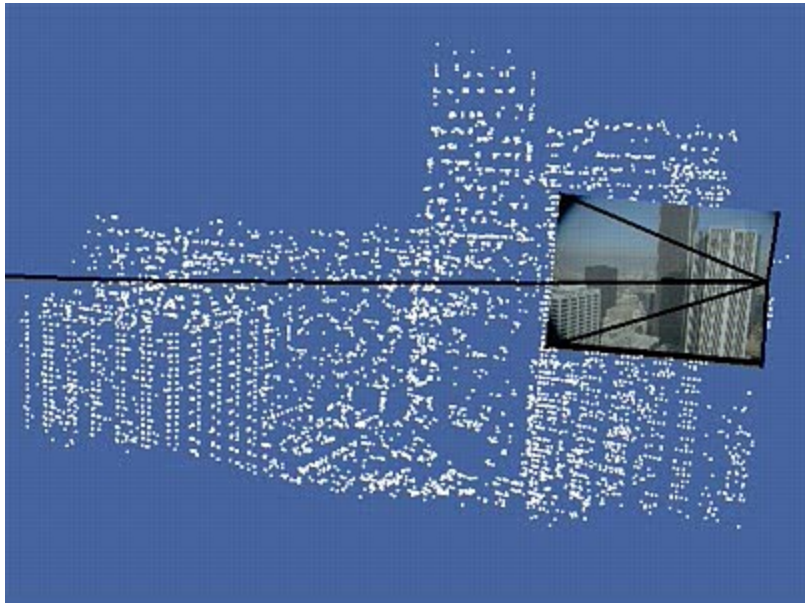
Что будем изучать сегодня

- Оптический поток
- Lucas-Kanade method
- Horn-Schunk method
- Пирамиды для большого движения
- Общая подход
- Применение

Применение

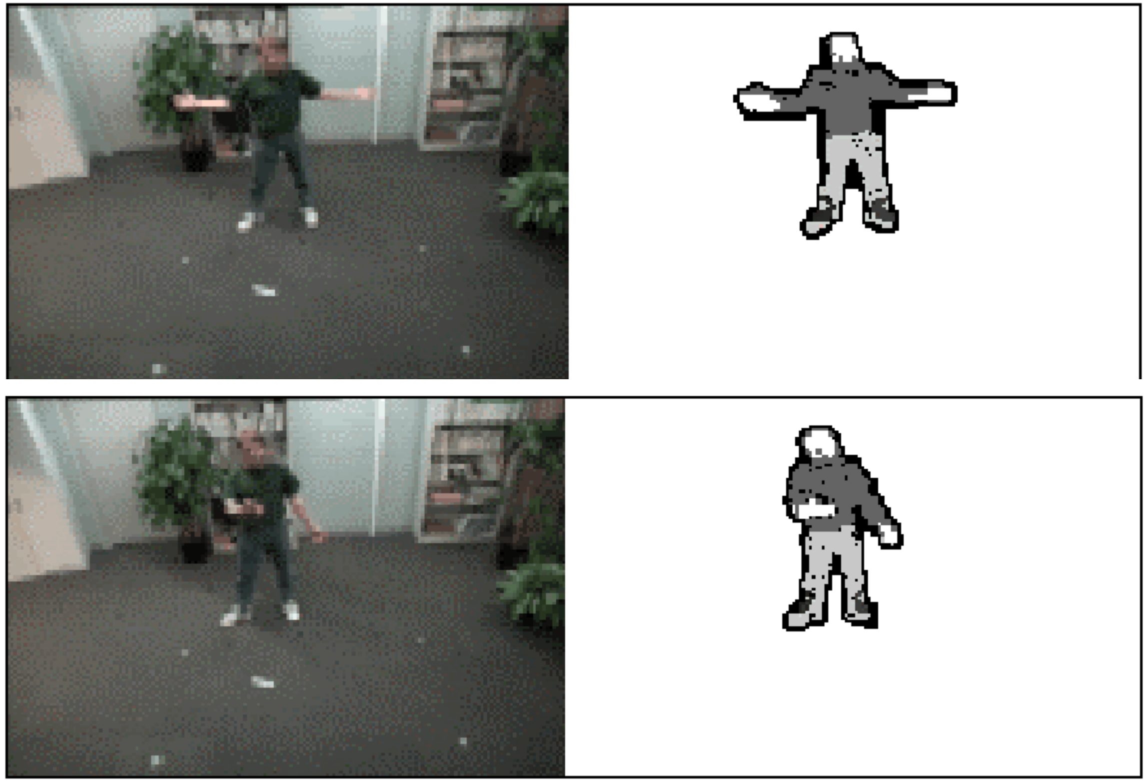
- Сегментация объектов на базе поиска движения
- Обучение динамических моделей
- Улучшение качество видео
    - Стабилизация потока
    - Улучшение разрашения
- Сопровождение образов
- Распознавание событий

Estimating 3D structure



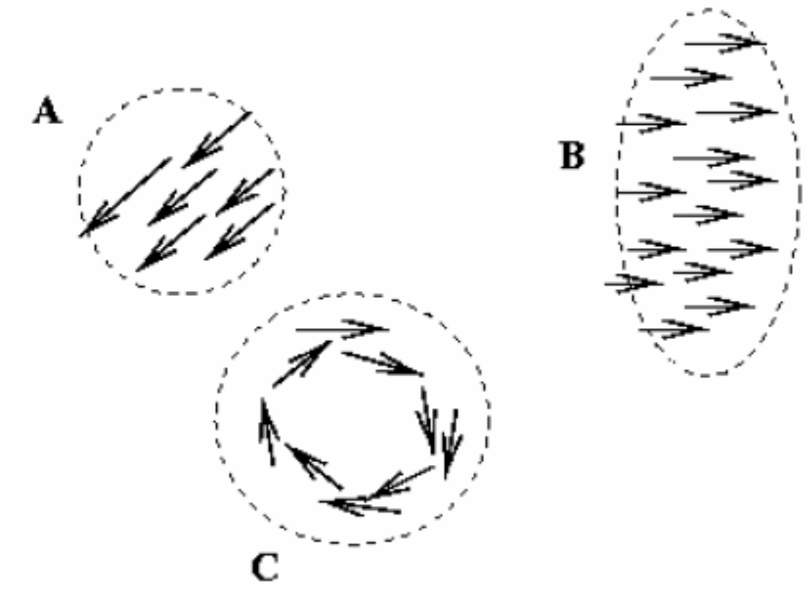
Segmenting objects based on motion cues

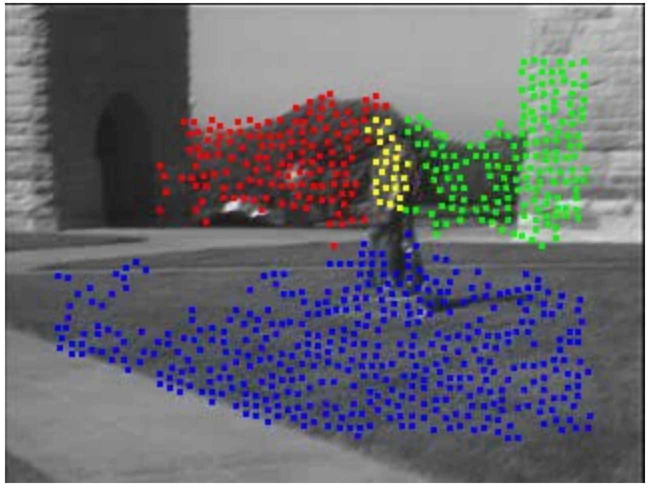
- Background subtraction
  - A static camera is observing a scene
  - Goal: separate the static *background* from the moving *foreground*



Segmenting objects based on motion cues

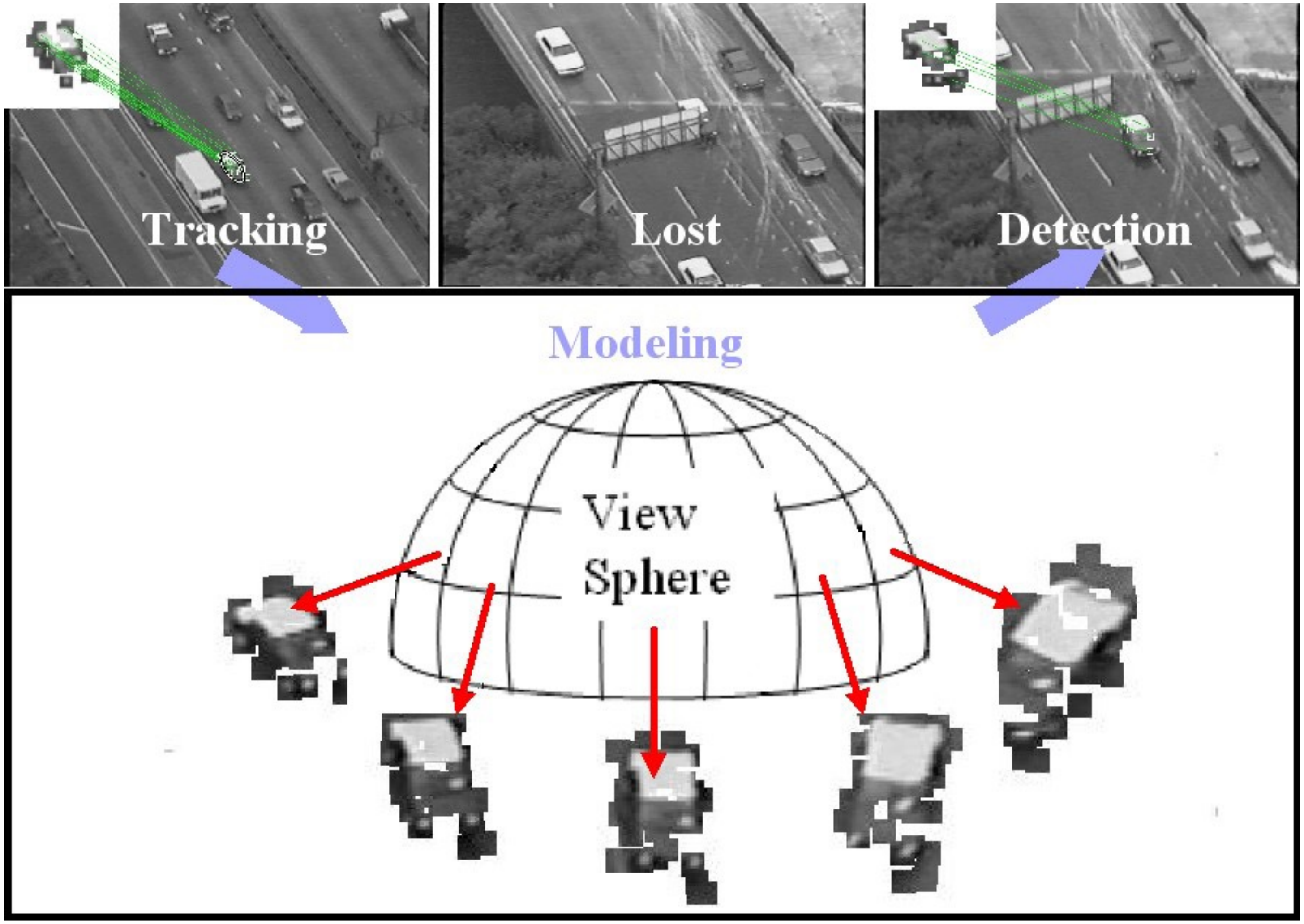
- Motion segmentation
  - Segment the video into multiple *coherently* moving objects





S. J. Pundlik and S. T. Birchfield, Motion Segmentation at Any Speed, Proceedings of the British Machine Vision Conference (BMVC) 2006

Tracking objects

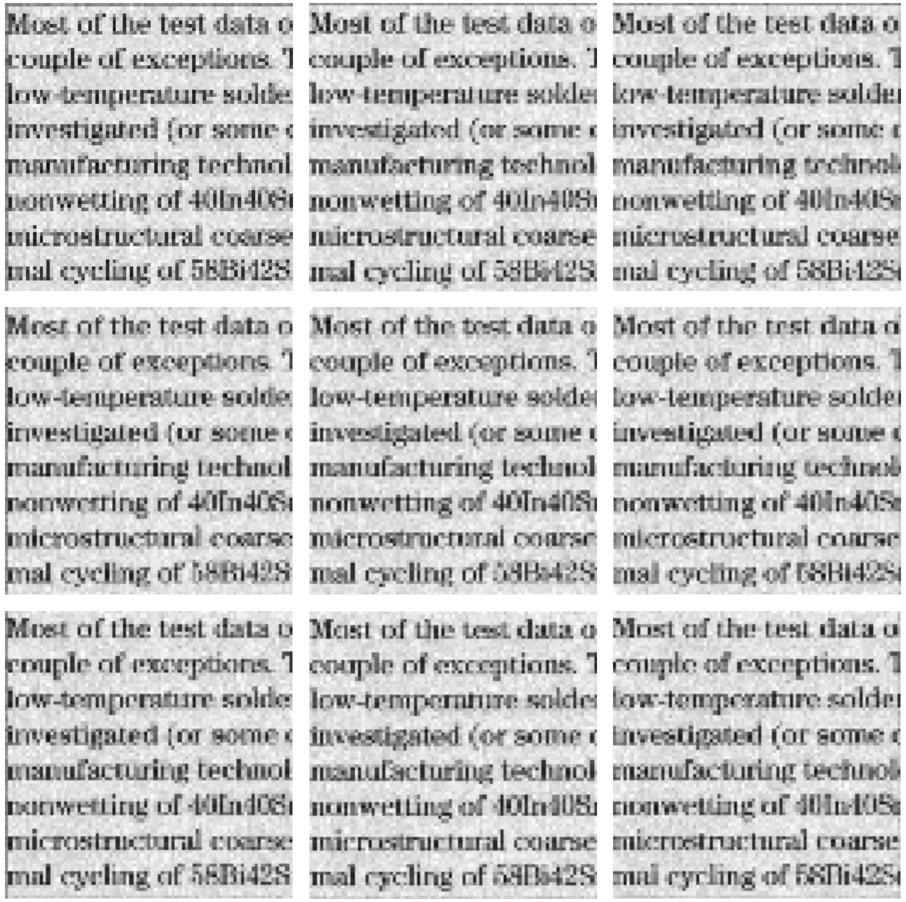


Z.Yin and R.Collins, "On-the-fly Object Modeling while Tracking," *IEEE Computer Vision and Pattern Recognition (CVPR '07),* Minneapolis, MN, June 2007.

Synthesizing dynamic textures

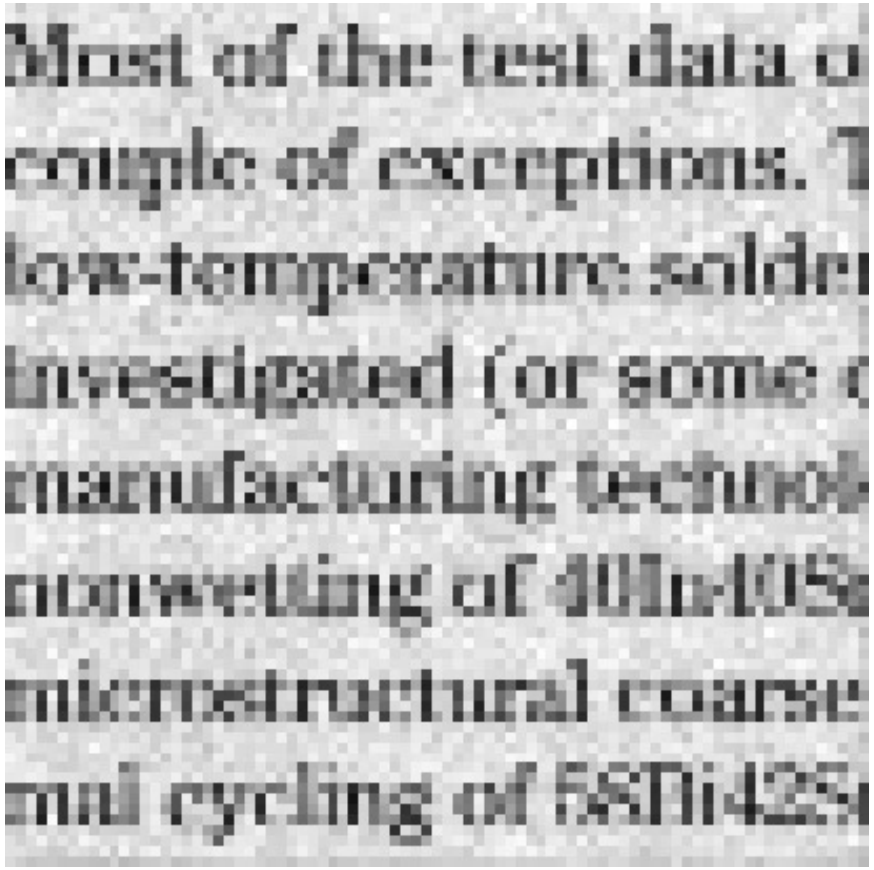
Super-resolution

Example: A set of low quality images



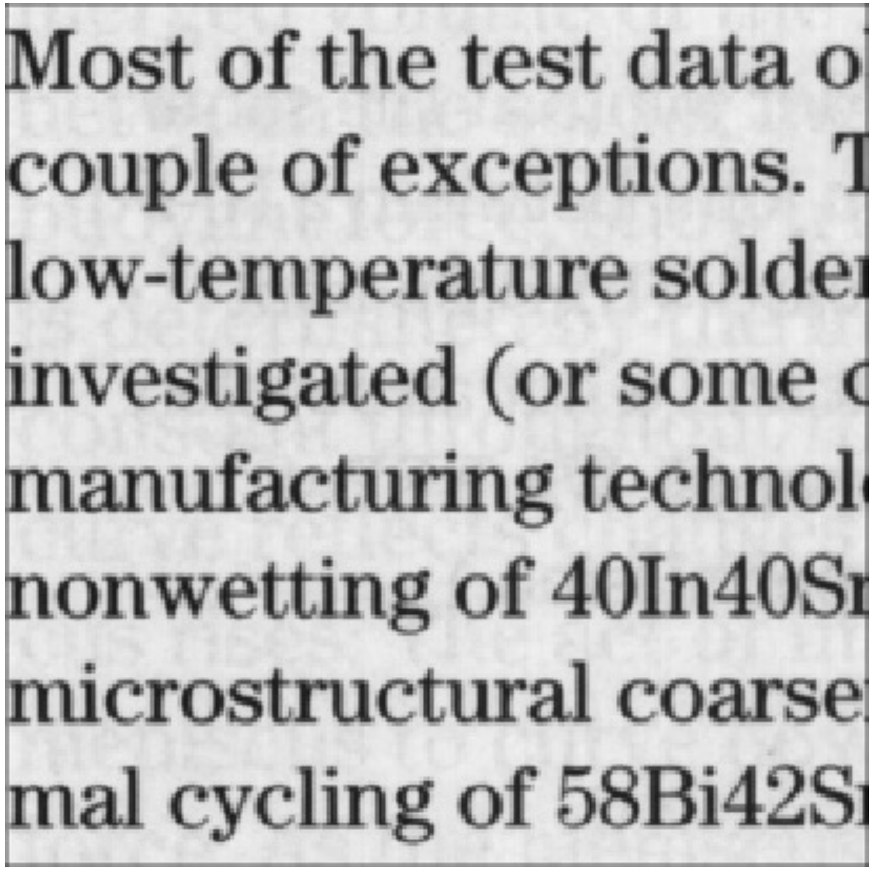
Super-resolution

Each of these images looks like this:

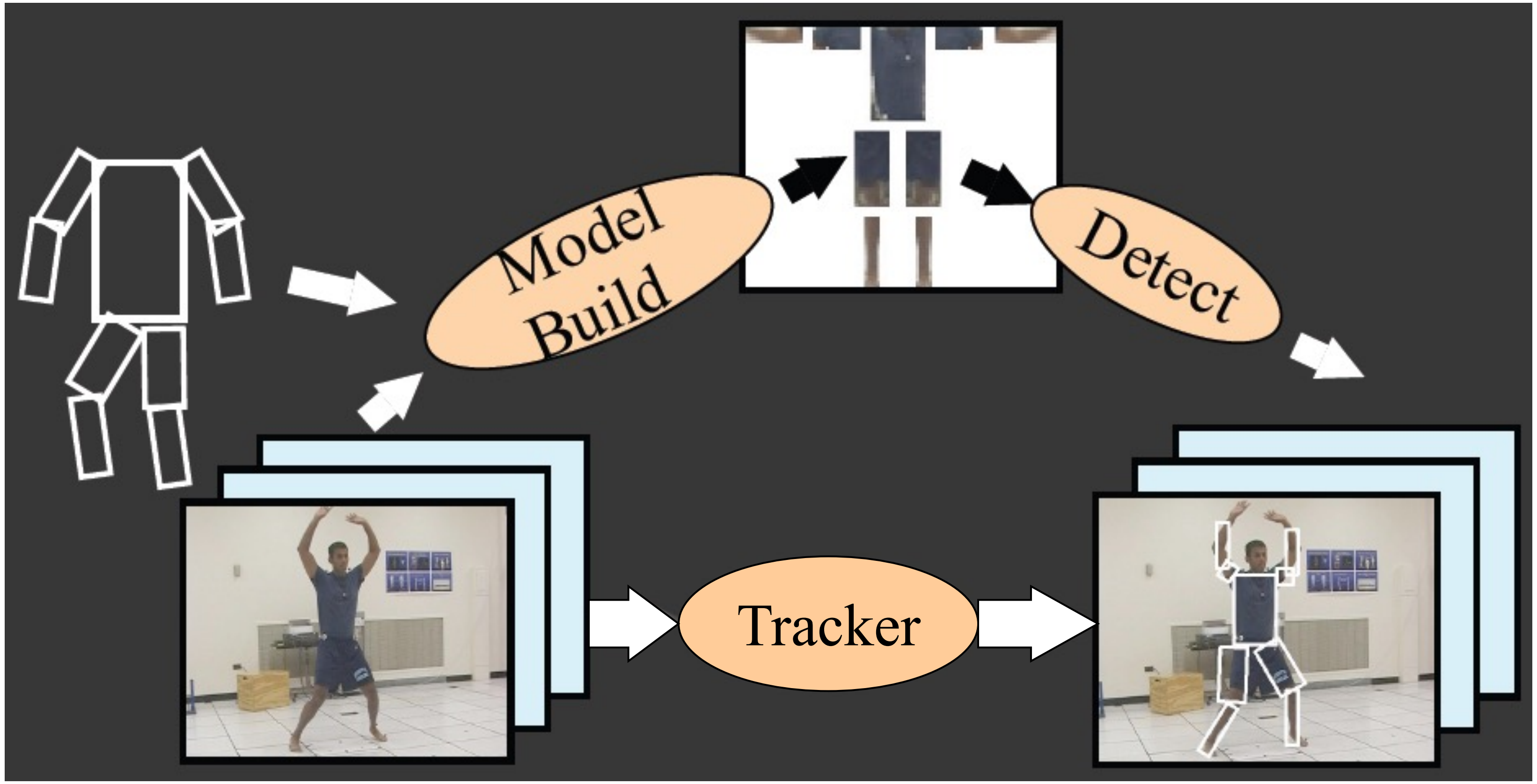

Most of the test data o
couple of exceptions. T
low-temperature solder
investigated (or some
manufacturing technol
nonwetting of 40In40
microstructural coarse
mal cycling of 58Bi42

Super-resolution

The recovery result:

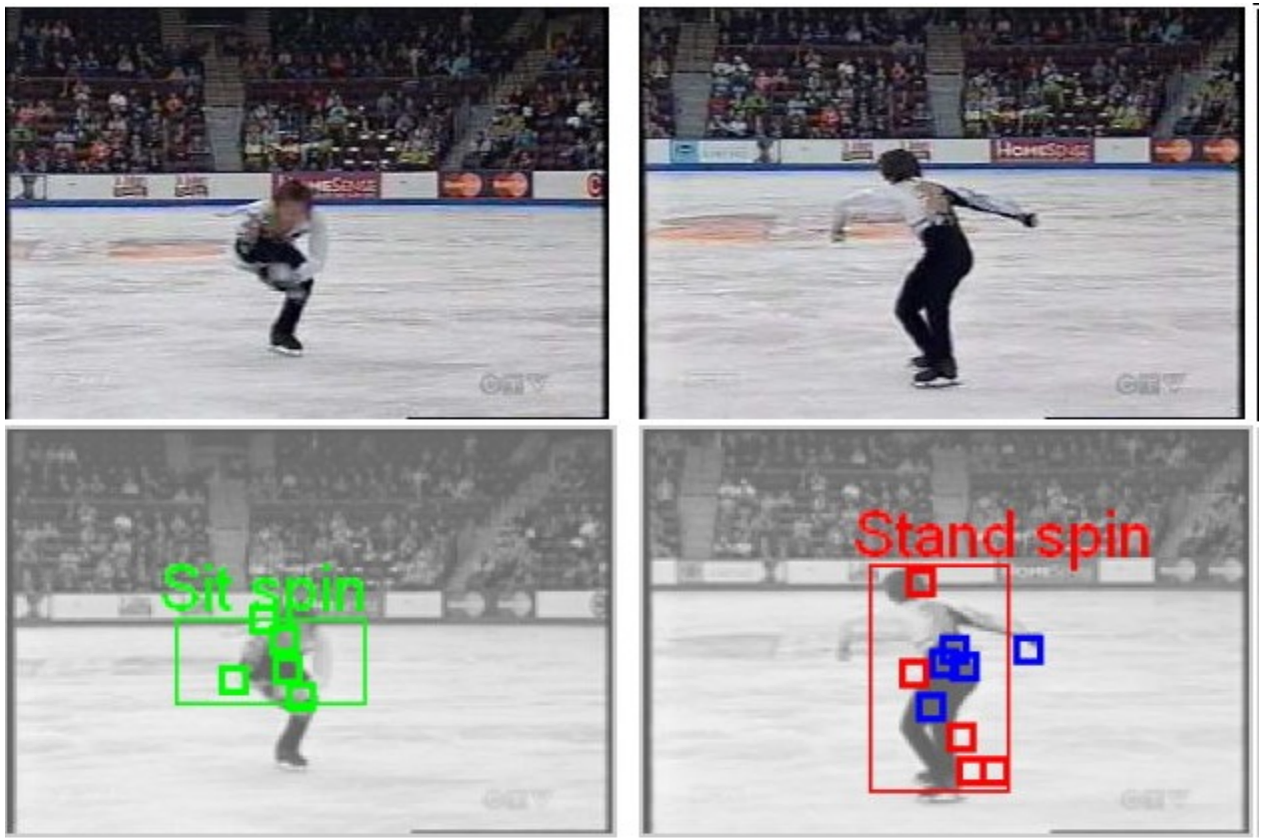

Most of the test data o
couple of exceptions. T
low-temperature solde
investigated (or some c
manufacturing technol
nonwetting of 40In40Sn
microstructural coarse
mal cycling of 58Bi42S

Recognizing events and activities



D. Ramanan, D. Forsyth, and A. Zisserman. [Tracking People by Learning their Appearance](). PAMI 2007.

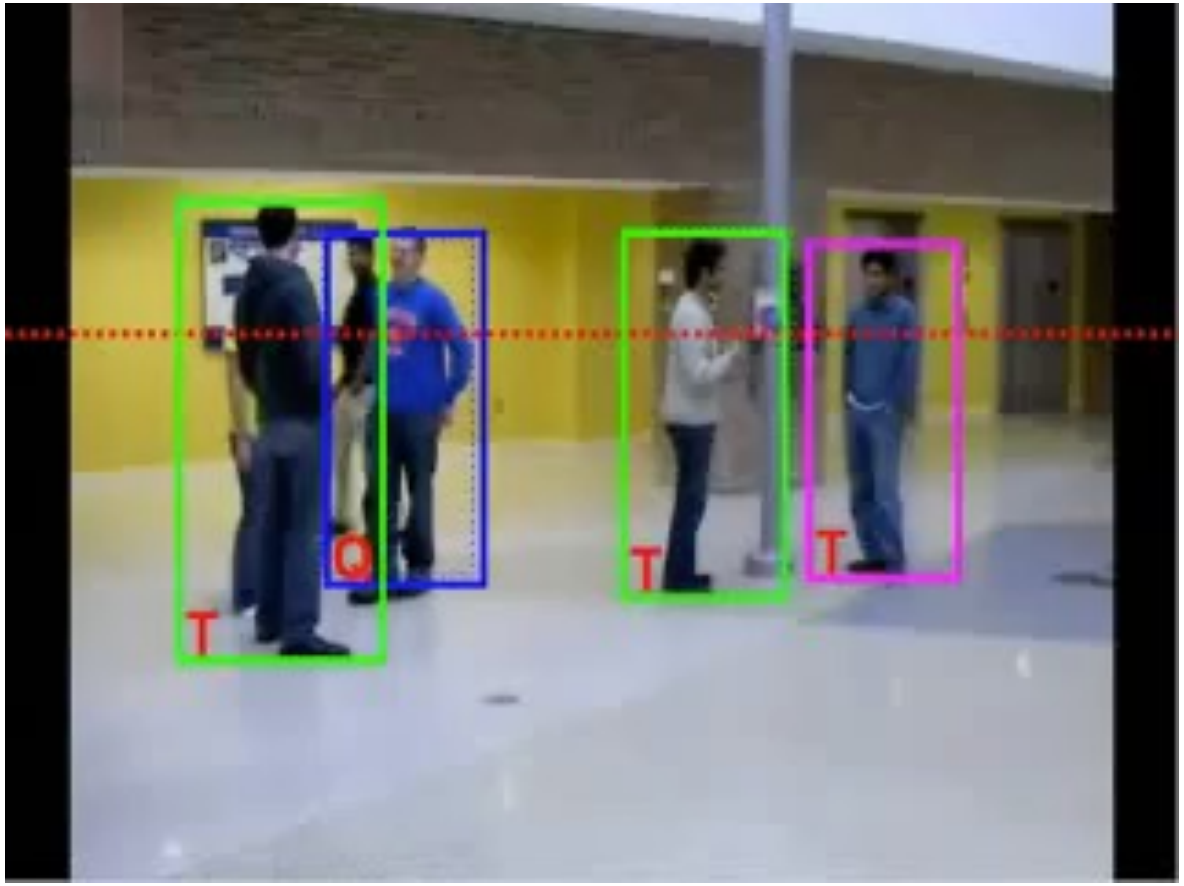
Recognizing events and activities



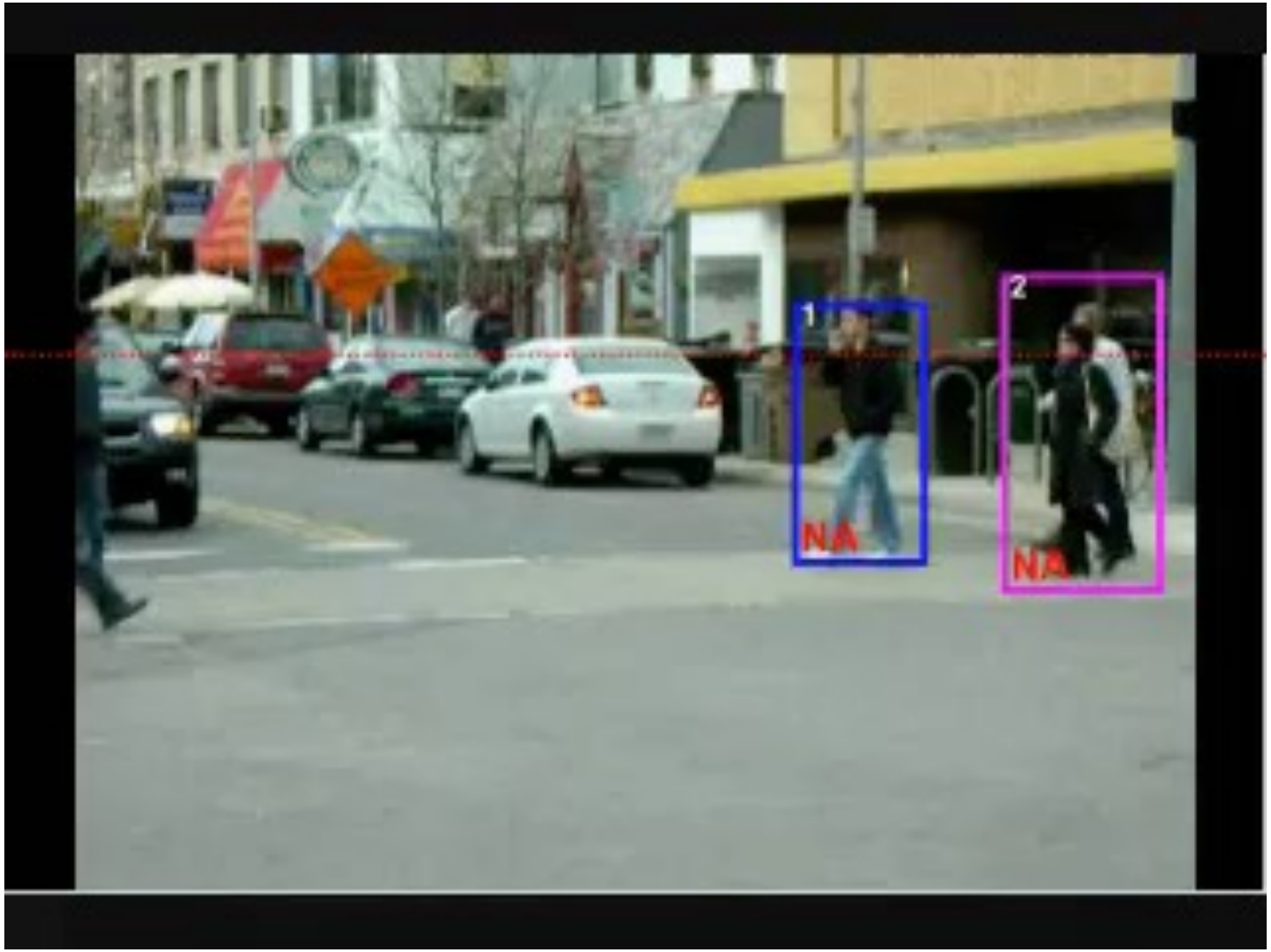
Juan Carlos Niebles, Hongcheng Wang and Li Fei-Fei, **Unsupervised Learning of Human Action Categories Using Spatial-Temporal Words**, *(BMVC)*, Edinburgh, 2006.

Recognizing events and activities

Crossing – Talking – Queuing – Dancing – jogging



W. Choi & K. Shahid & S. Savarese WMC 2010



W. Choi, K. Shahid, S. Savarese, "What are they doing? : Collective Activity Classification Using Spatio-Temporal Relationship Among People", 9th International Workshop on Visual Surveillance (VSWS09) in conjuction with ICCV 09

Заключение

- Оптический поток
- Lucas-Kanade method
- Horn-Schunk method
- Пирамиды для большого движения
- Общая подход
- Применение оптического потока

Литература: [Szeliski] Chapters: 8.4, 8.5

[Fleet & Weiss, 2005]
http://www.cs.toronto.edu/pub/jepson/teaching/vision/2503/opticalFlow.pdf