GRAPHICS & MEDIA LAB
VIDEO GROUP

Задачи оценки движения и вычисления оптического потока

Дмитрий Ватолин
Video Group
CS MSU Graphics&Media Lab

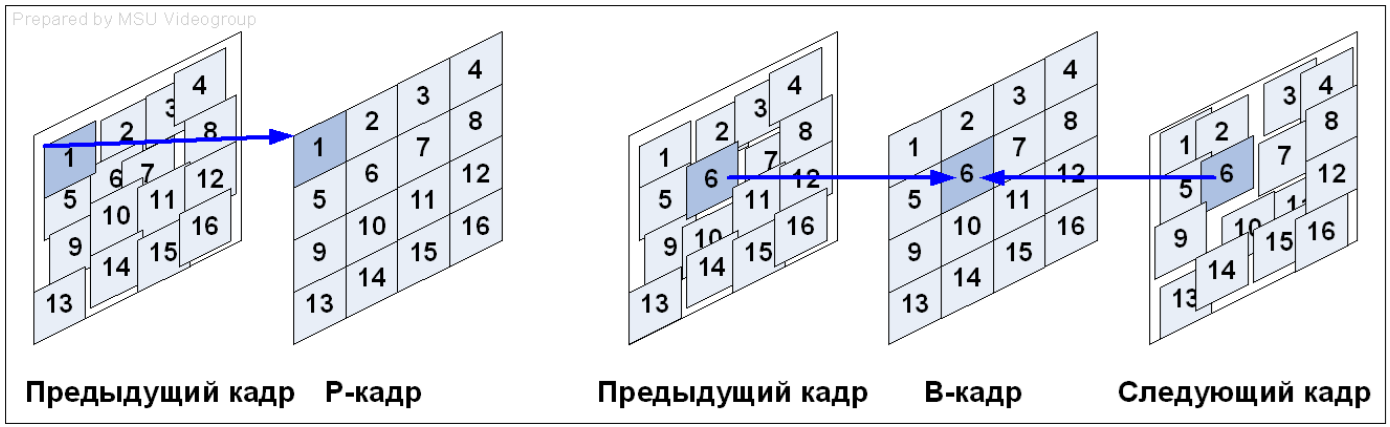
Компенсация движения

Постановка задачи

ME (Motion Estimation) — компенсация движения

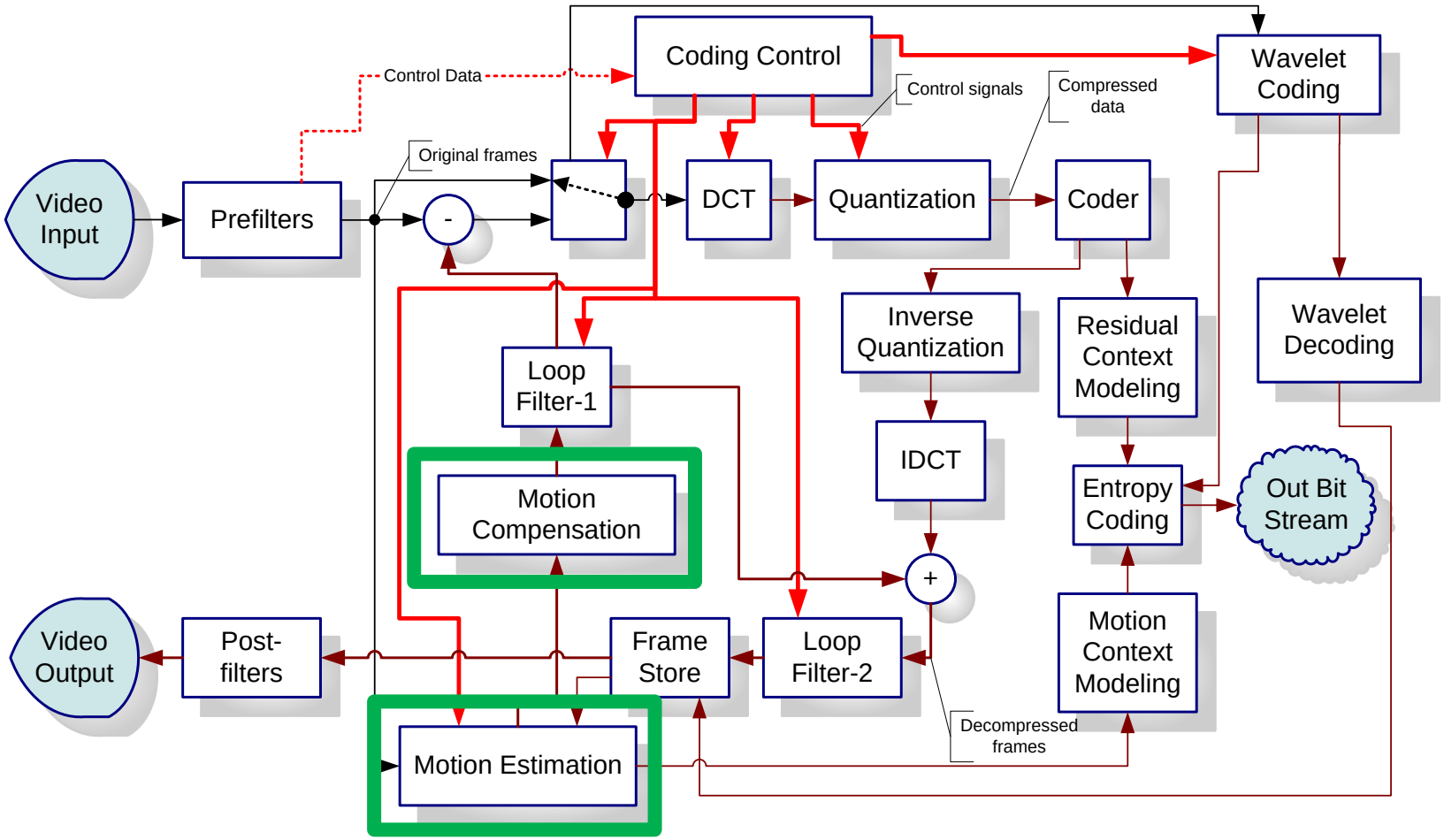
Кадры видео делятся на прямоугольные блоки одинакового размера

Задача: для каждого блока найти вектор смещения относительно другого кадра, минимизирующий попиксельную разницу



Области применения

Схема видеокодека



Области применения

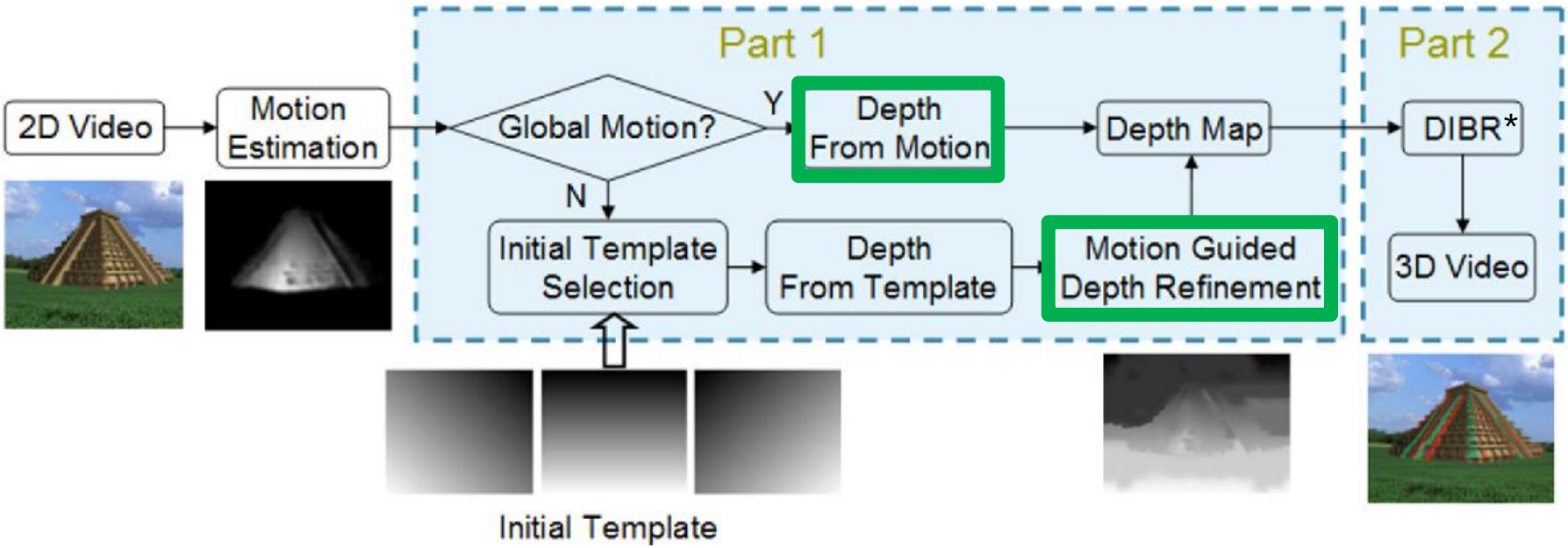
Вклад компенсации движения

Без компенсации

С компенсацией

Области применения

Конвертация из 2D в 3D



*depth image-based rendering

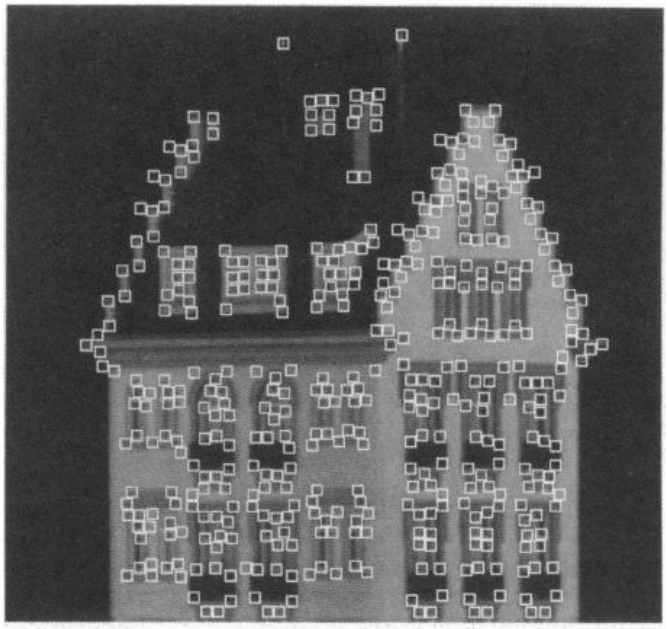
C. Jung et al, “2D to 3D conversion with motion-type adaptive depth estimation,” in *Multimedia Systems*, 2015

Области применения

Построение карты глубины из движения

Области применения

Отслеживание объектов и особых точек



Компенсация движения
Метрика блоков

Расстояние между блоками ищется как сумма модулей разности значений пикселей:

$$R(A,B) = \sum_{i,j=0}^{n,n} \left| a_{i,j} - b_{i,j} \right|$$

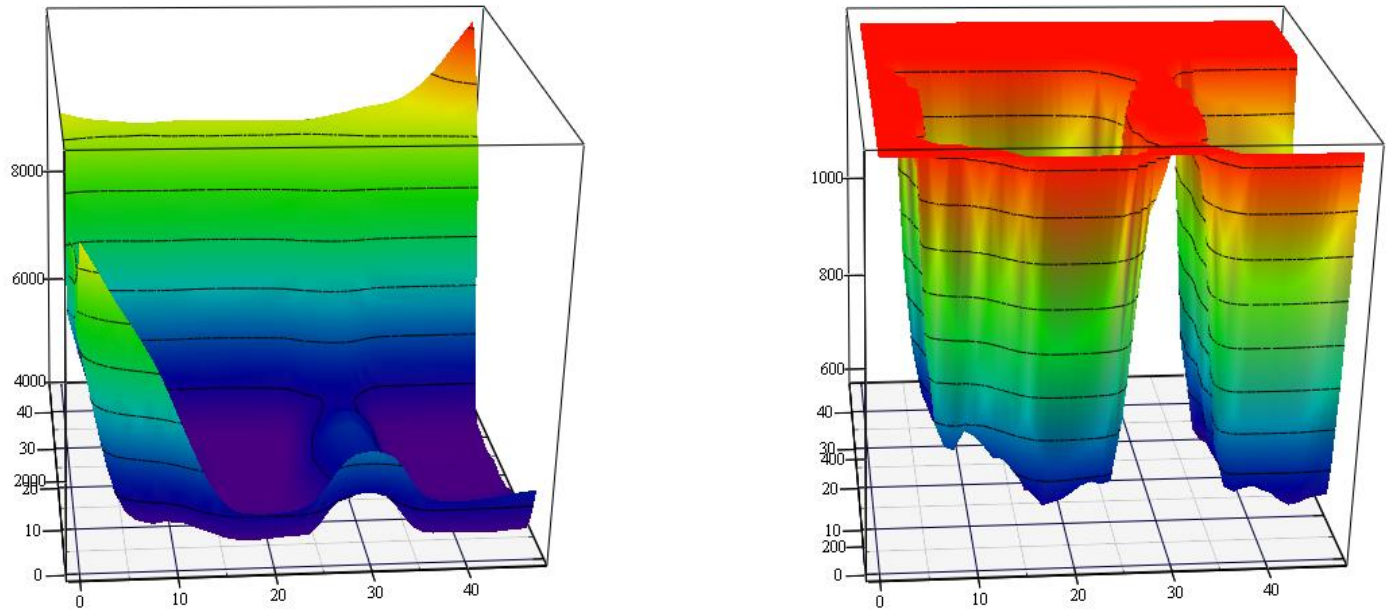
Достоинство:

- Хорошо оптимизируется (важно, поскольку ME — самая долгая операция при сжатии)

Недостатки:

- Неустойчива к шуму
- Не учитывает изменения яркости

Компенсация движения
Иллюстрация функции ошибки (1)

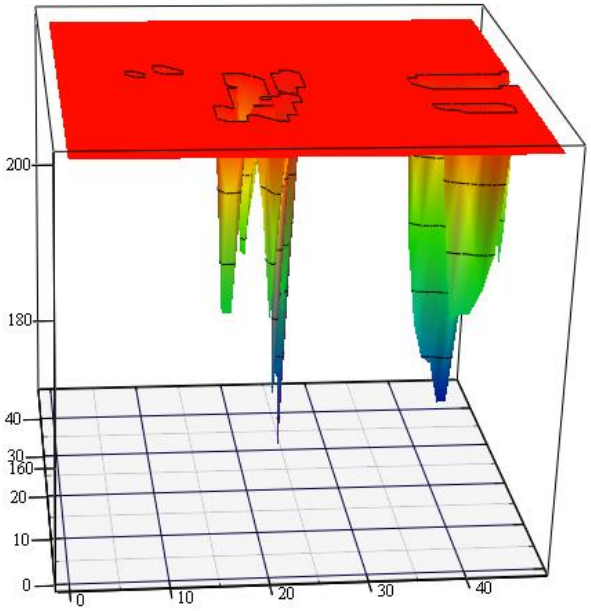


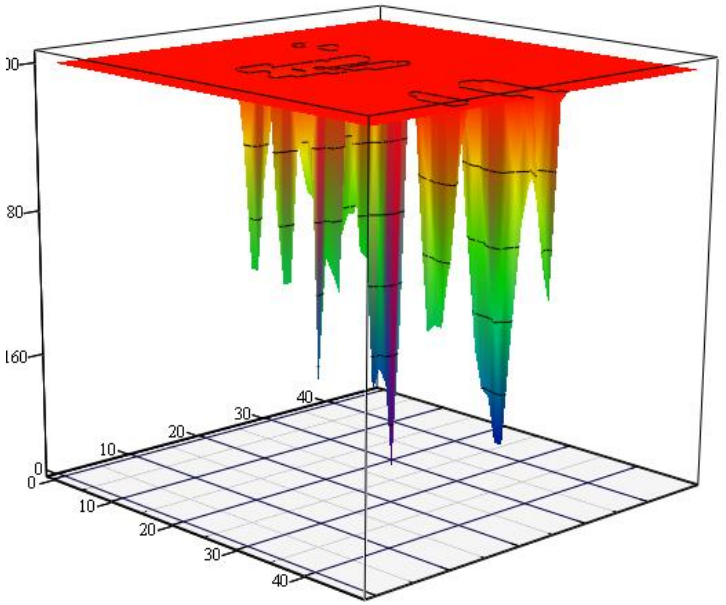
Величина функции ошибки в зависимости от вектора сдвига (x, y)

Взят блок в однородной области (стена), сдвинувшийся на 2 пикселя

Слева — все значения функции, справа — та же функция,
но значения обрезаны по порогу 1000

Компенсация движения
Иллюстрация функции ошибки (2)



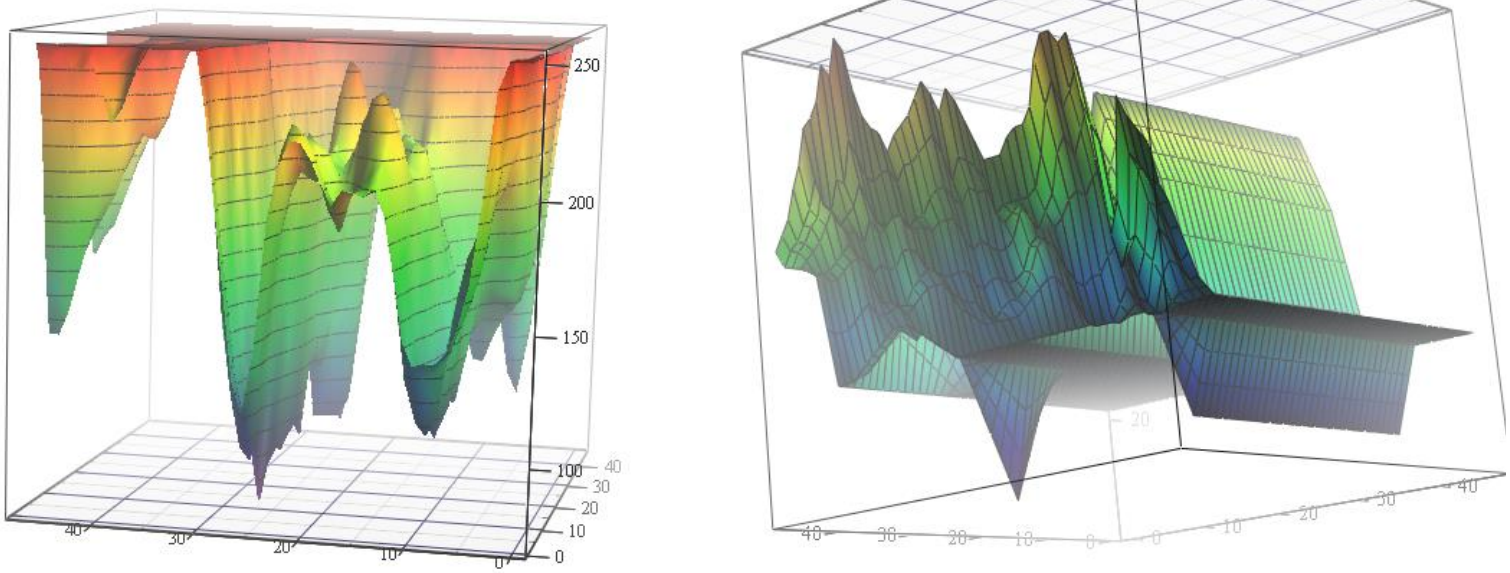


Та же функция с других ракурсов, значения обрезаны по порогу 200

Хорошо видно, как много у этой функции (казавшейся на первом рисунке гладкой) локальных минимумов

Если алгоритм будет выбирать их, то блоки начнут «гулять»

Компенсация движения
Иллюстрация функции ошибки (3)



Слева — блок, сдвинувшийся на 6 пикселей по одной оси
и на 11 по другой, однако локальных минимумов у графика много

Справа — также сильно сдвинувшийся блок, но минимум явно виден

«Ровный» график у дальней стенки — граница изображения

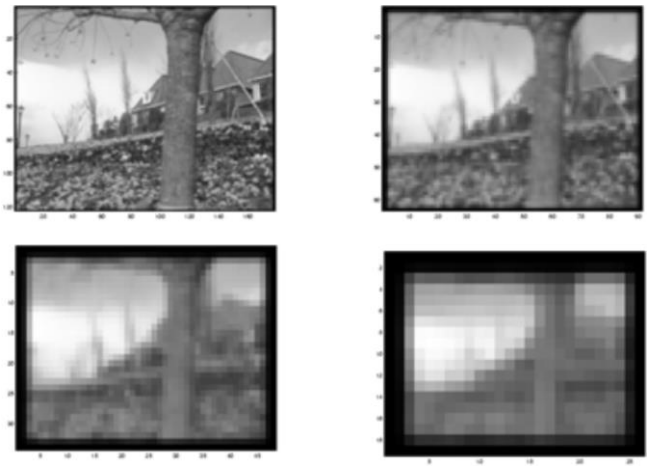
Компенсация движения
Трудности

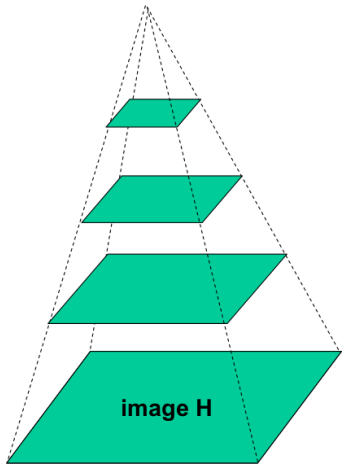
- Шумы: блоки не сопоставляются с нужными
- Однотонные поверхности или текстуры: блоки хорошо сопоставляются с ненужными
- Области открытия: нужного блока нет

Пирамида разрешений
Идея

Уменьшенные версии кадра легче сопоставлять, потому что в блок попадает больше крупных деталей

1. Сделаем несколько уменьшенных версий кадра
2. Сопоставим блоки на самом маленьком разрешении
3. Уточним найденные блоки на большем разрешении
4. Повторим шаг 3, пока не дойдём до исходного кадра



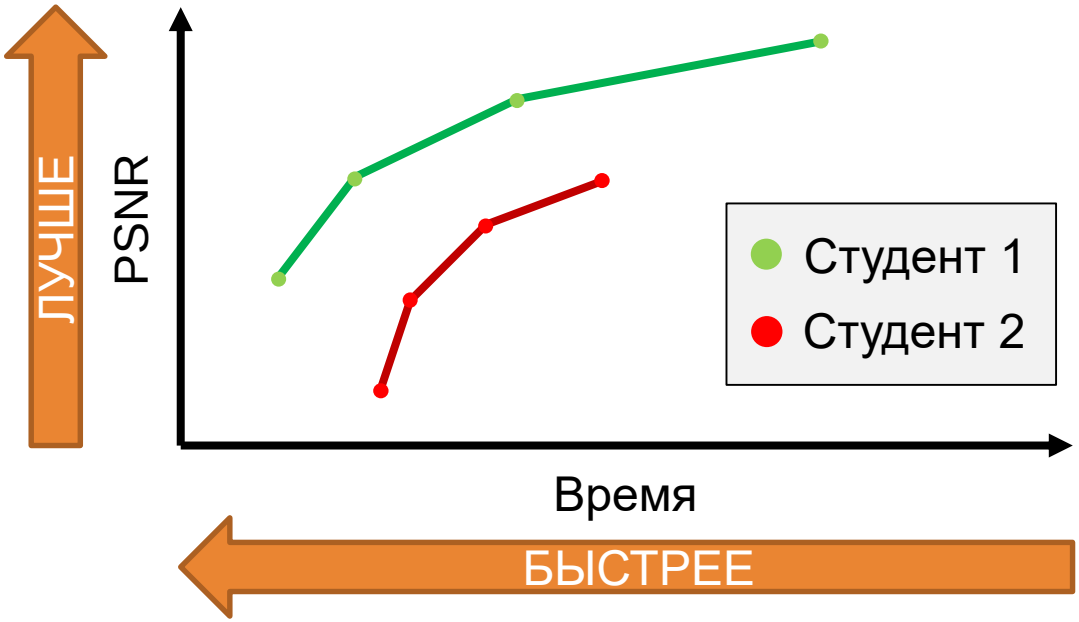


Компенсация движения
Задание

Цель: реализовать алгоритм компенсации движения

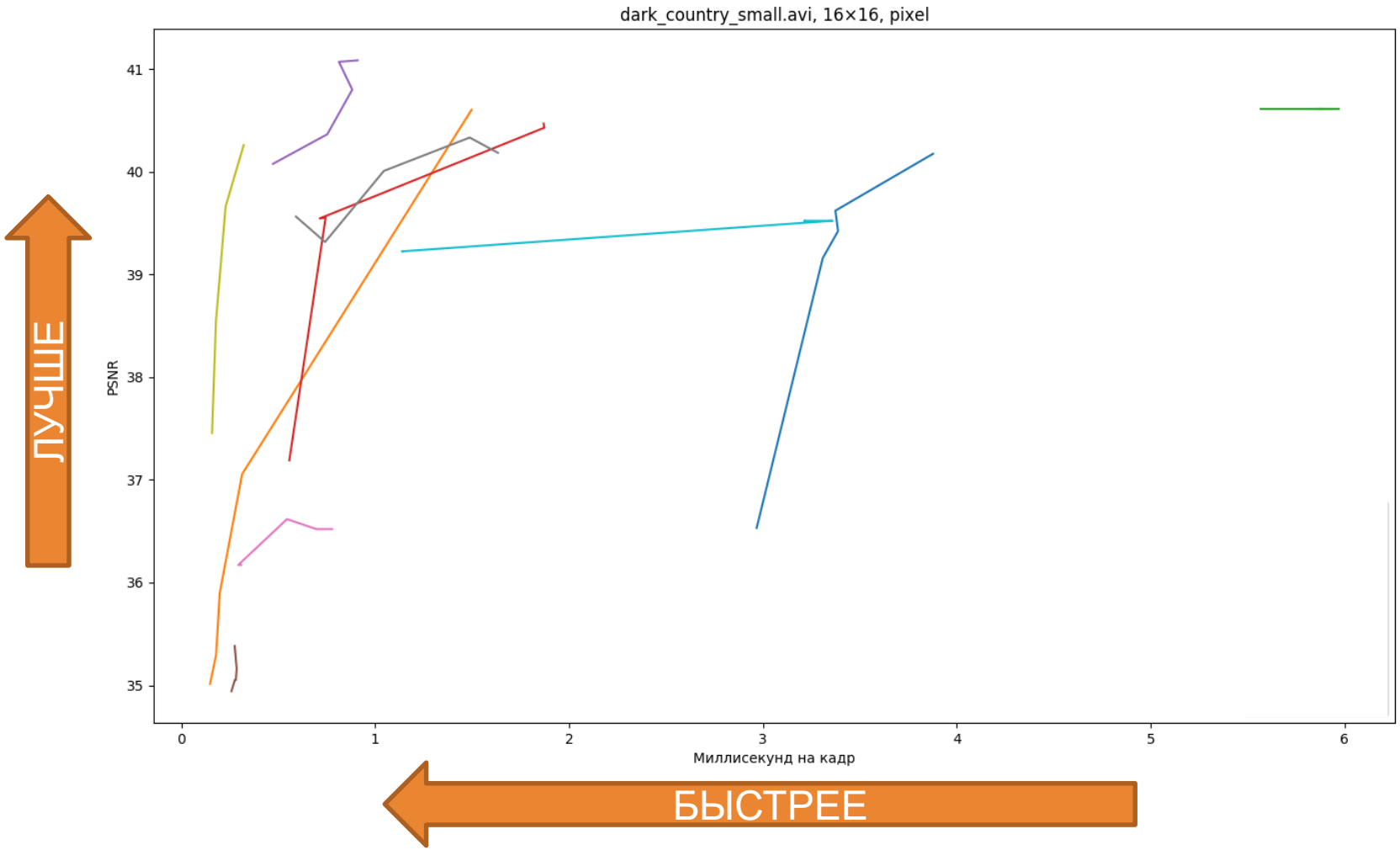
Измеряемые параметры:

- Качество (по метрике PSNR)
- Скорость работы



Компенсация движения

Пример графика в реальной жизни



Компенсация движения

Базовые стратегии поиска

Задача: за минимальное число замеров найти минимум

Используются комбинации стратегий:

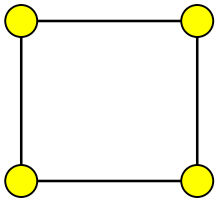
- Cross поиск
- Ортогональный поиск
- Трехшаговый поиск (TSS)
- Четырехшаговый поиск (FSS)

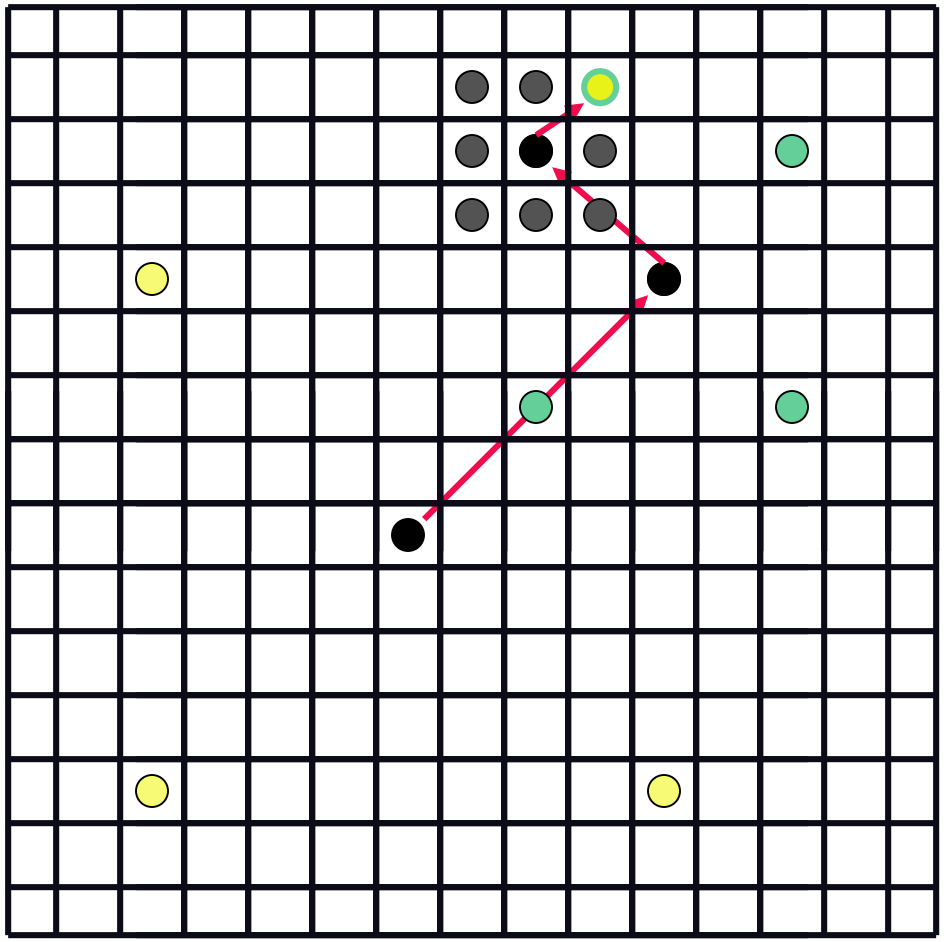
Стратегии поиска
Cross поиск

Cross поиск

Шаблон уменьшается на каждом шаге

Шаблон:



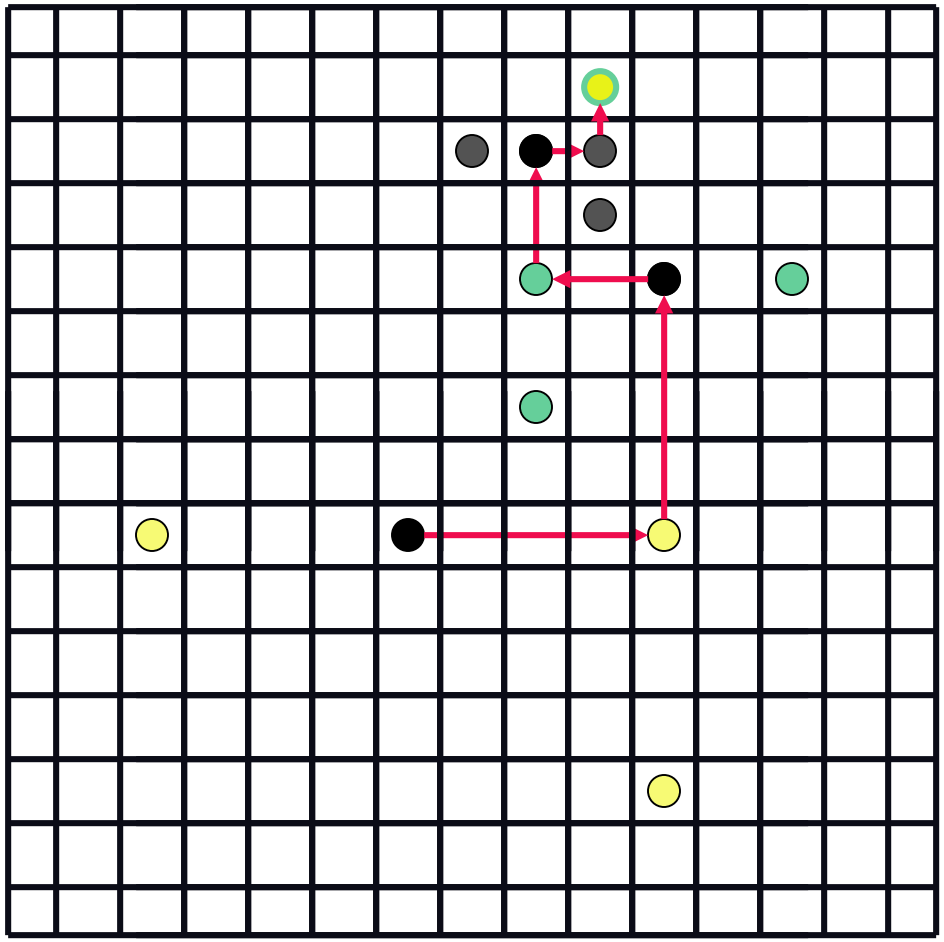


Стратегии поиска
Ортогональный поиск

Ортогональный поиск:

Пробуются попеременно вертикальное и горизонтальное направления

Шаблоны:

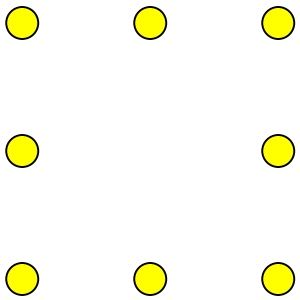


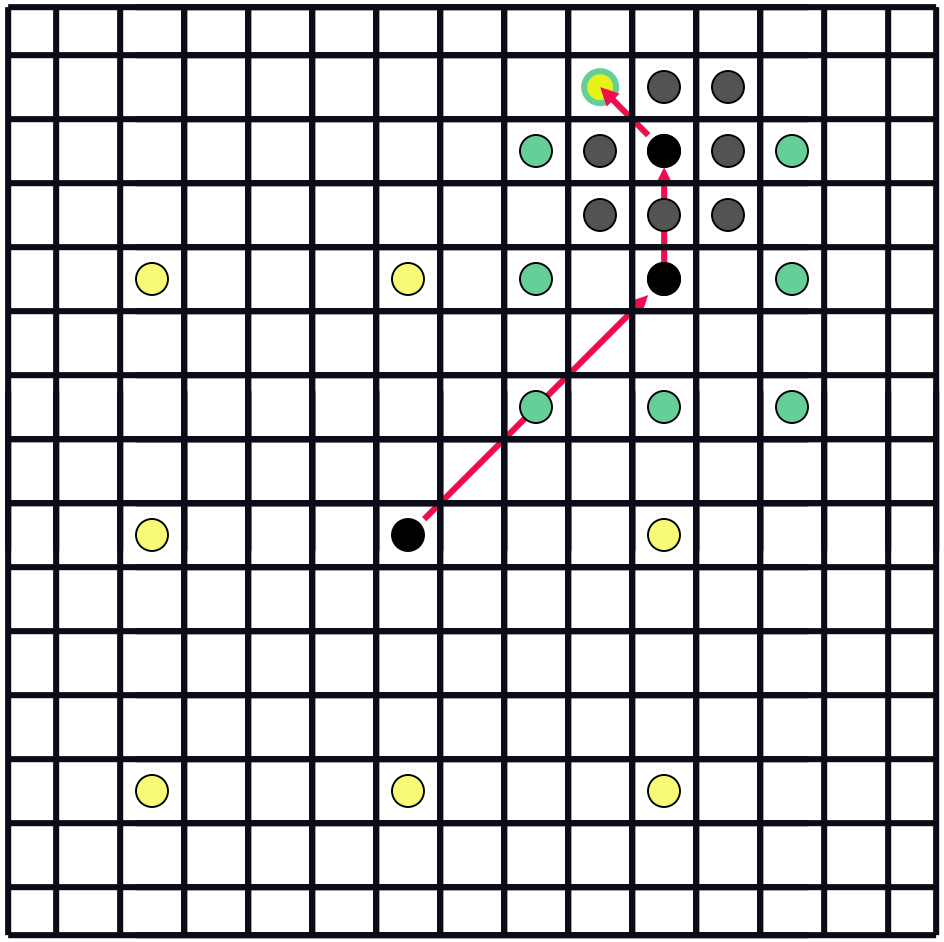
Стратегии поиска
Трехшаговый поиск (TSS)

Three Step Search (TSS)

На каждом шаге размер шаблона уменьшается

Шаблон:





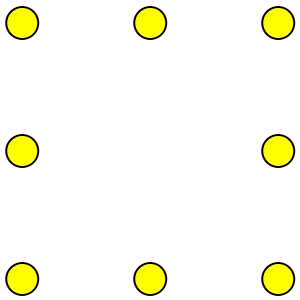
Стратегии поиска

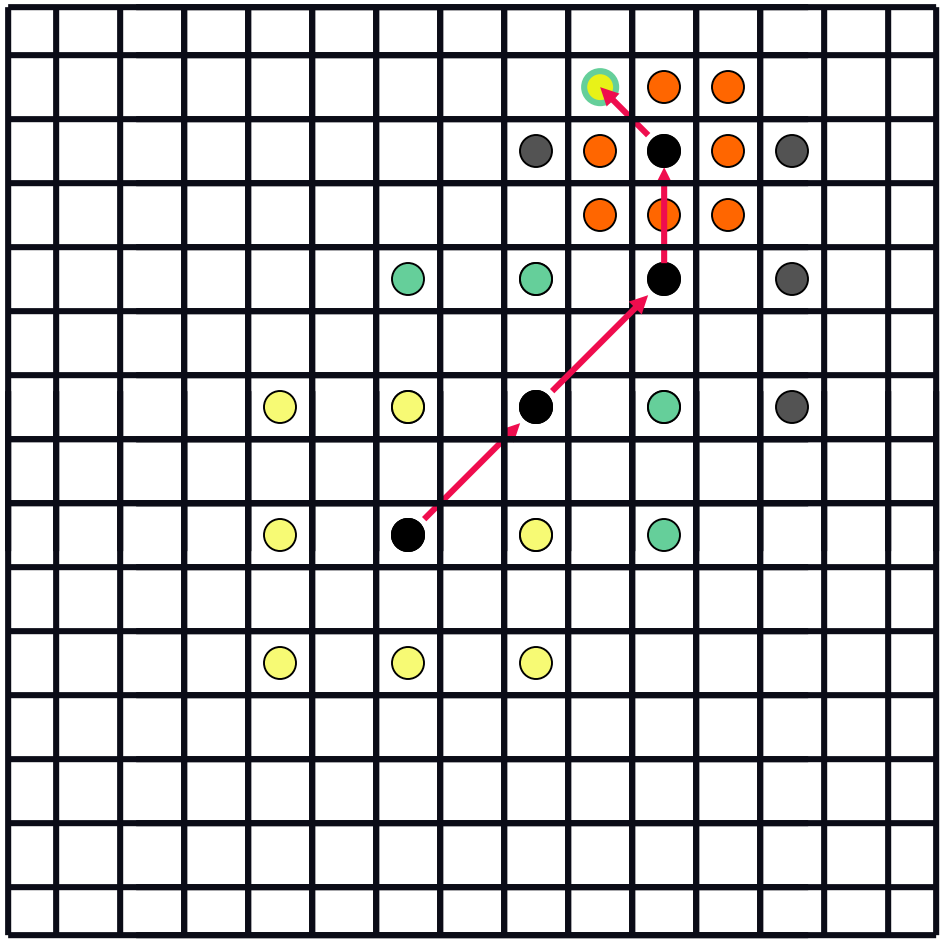
Четырехшаговый поиск (FSS)

Four Step Search (FSS)

Шаблон **не** уменьшается

Шаблон:

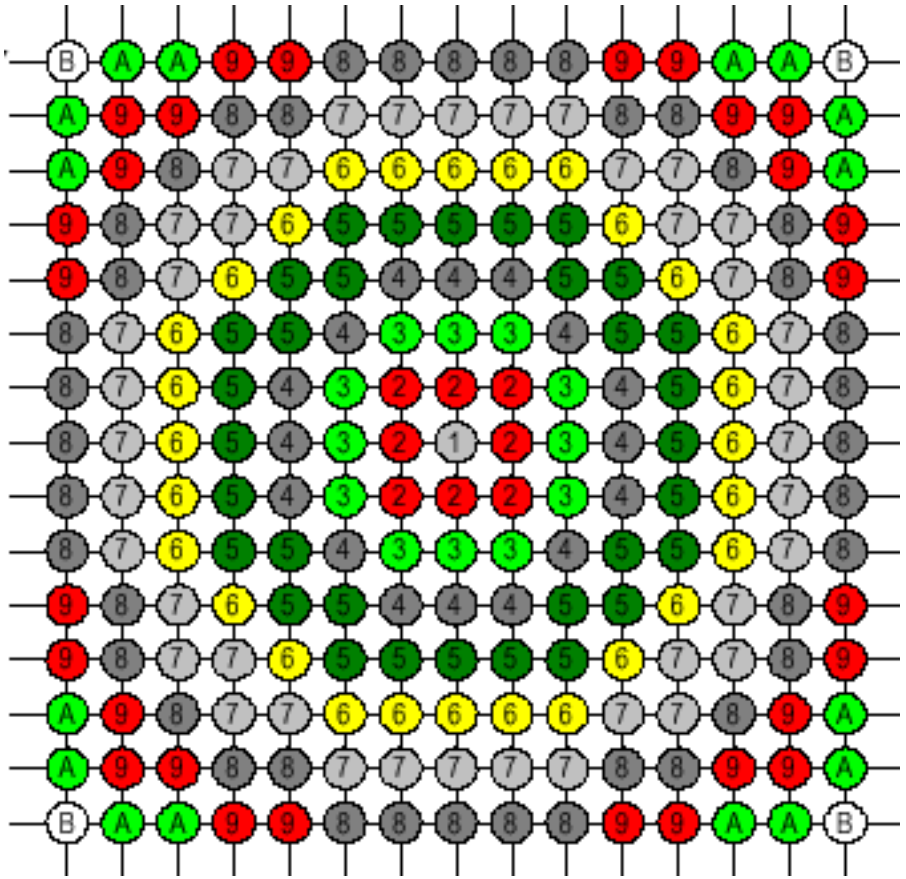




Стратегии поиска
Спиральный поиск

Спиральный поиск

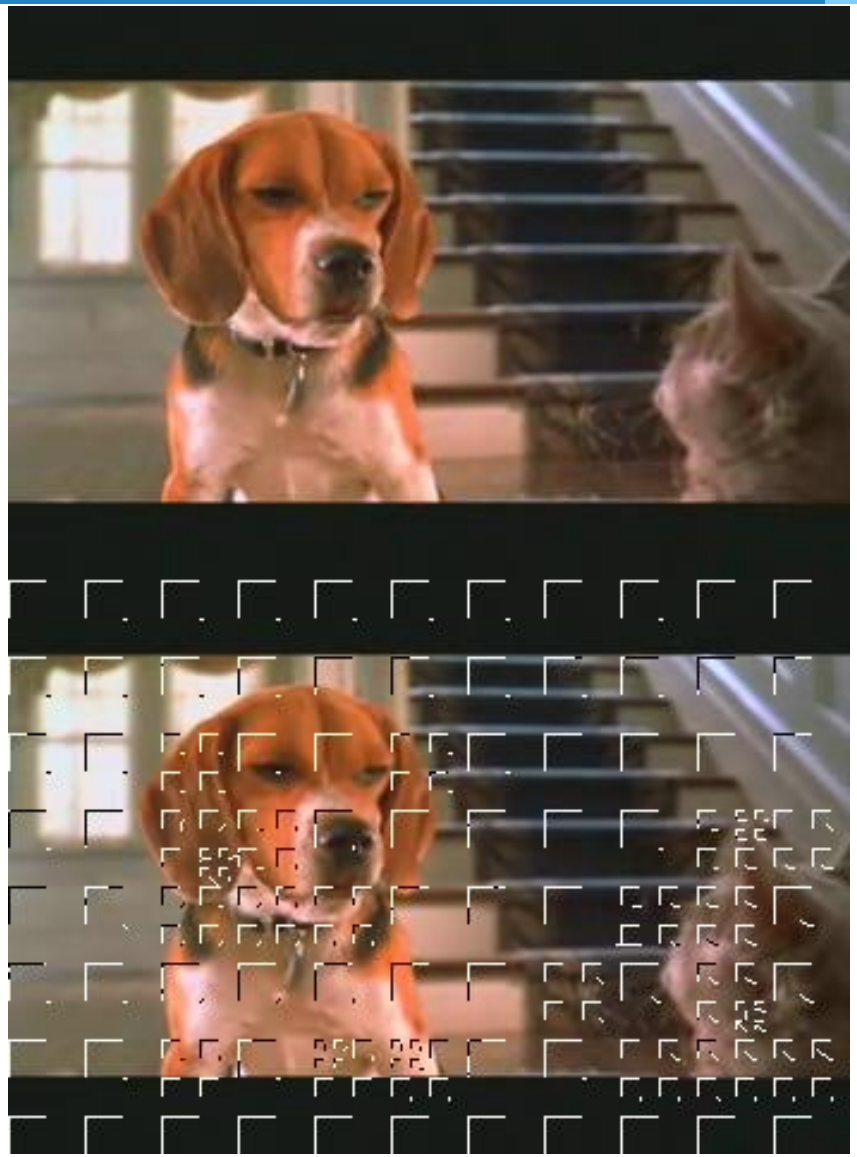
Контр-пример: Если мы будем все время искать с какого-то края, то у нас будут «ползти» ровные куски кадра



Примеры
Вектора движения (1)

Кадр из фильма
«Кошки против собак»

Векторами показаны
распознанные вектора
движения блоков

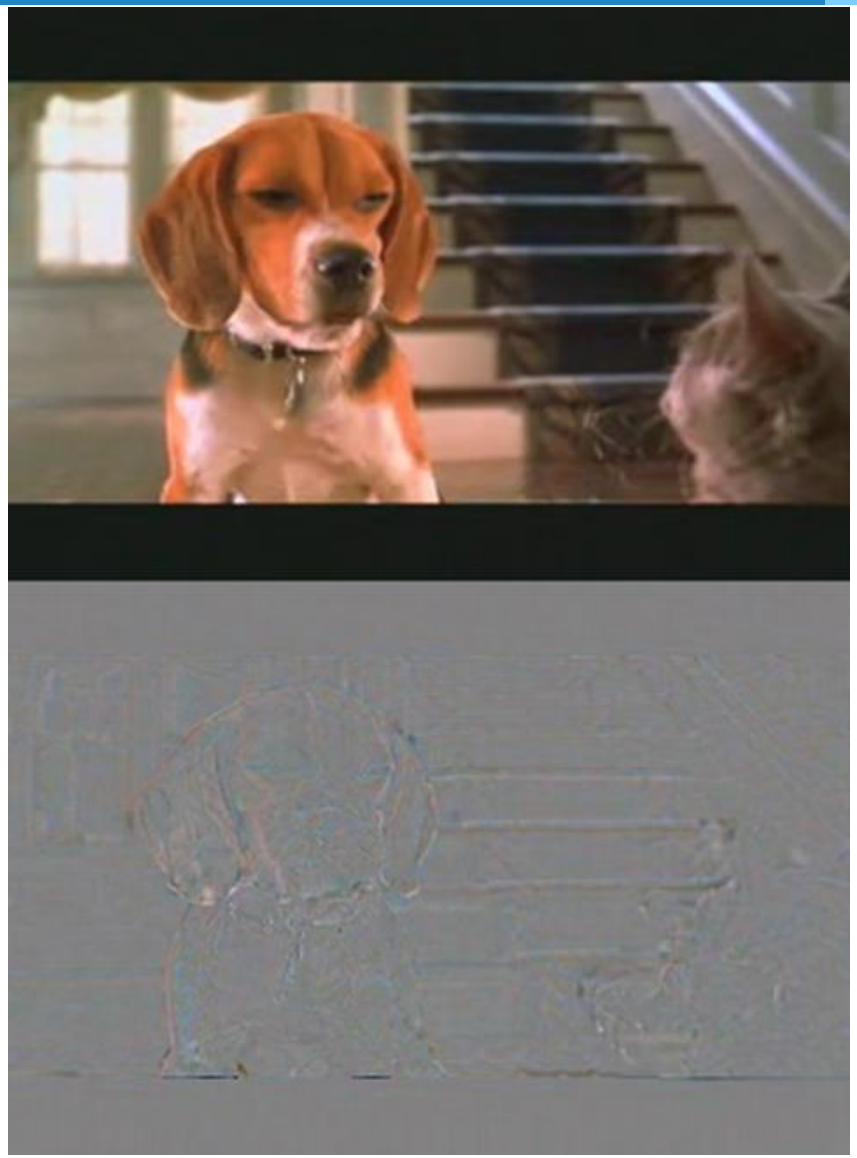


Примеры
Вектора движения (2)

Кадр из фильма «Кошки против собак»

Следующий кадр

Показана межкадровая разница, сравнительно небольшая, благодаря компенсации движения



Примеры
Вектора движения (3)

Кадр из фильма «Звездные войны, Эпизод 1»

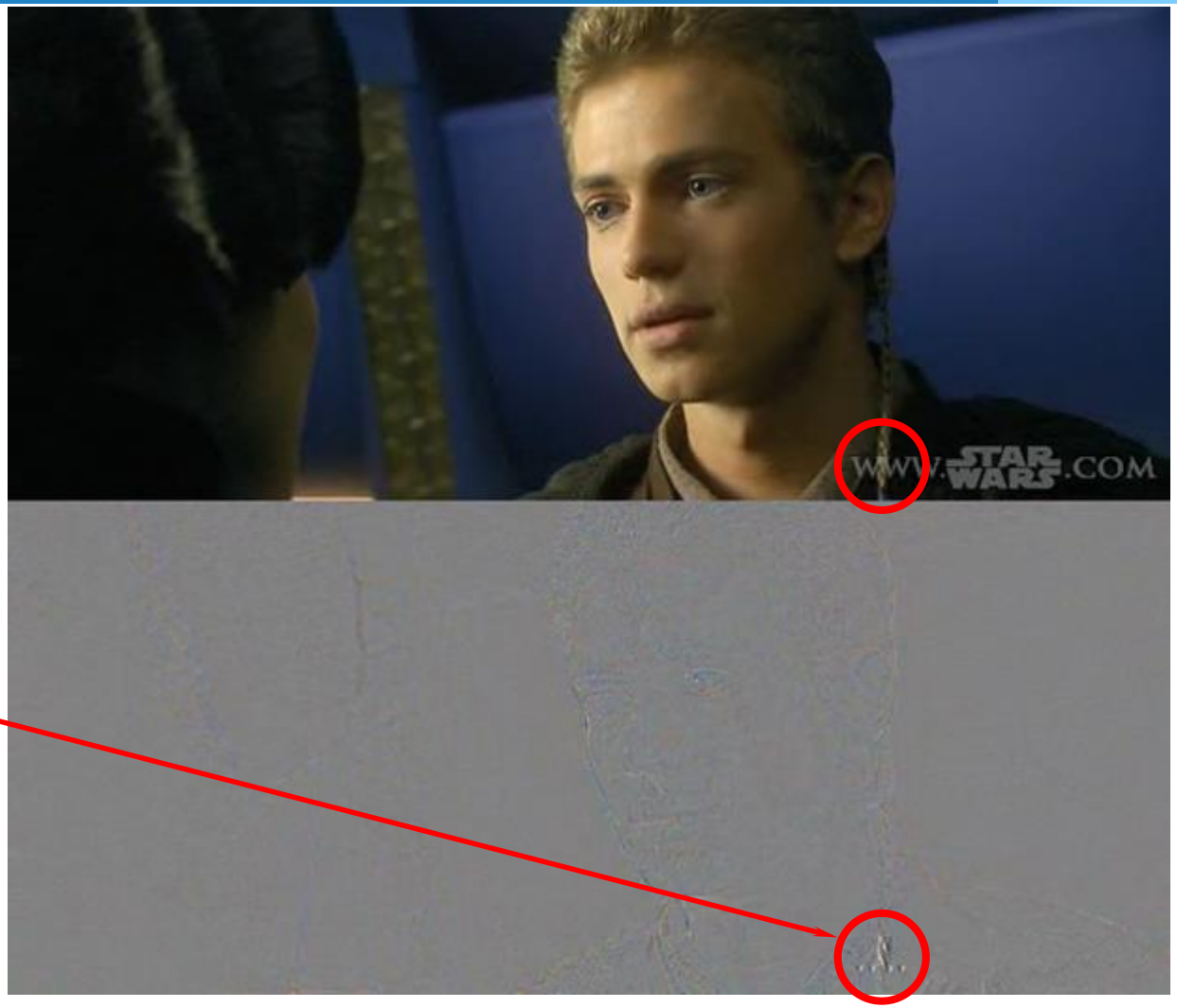
Векторами показаны распознанные вектора движения блоков

Примеры
Вектора движения (4)

Кадр из фильма «Звездные войны, Эпизод 1»

Следующий кадр

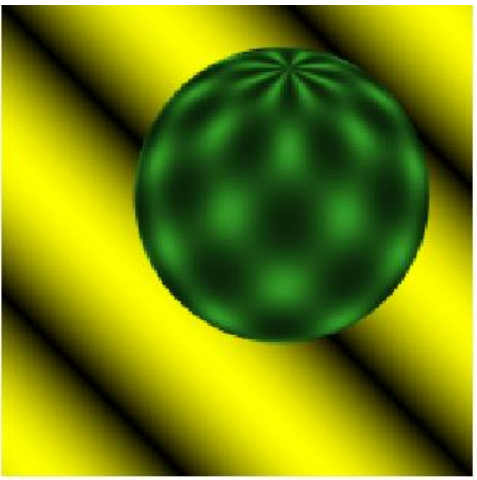
Хорошо видно, что хуже всего распознался «воротник за логотипом»



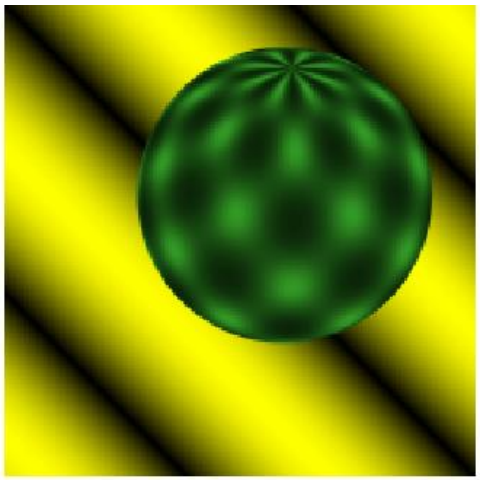
Оптический поток (optical flow)

Определение

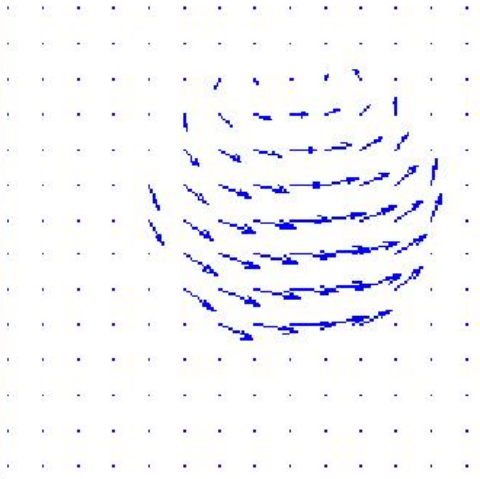
Векторное поле явного движения объектов (пикселей), поверхностей и ребер в визуальной сцене между кадрами, вызванное относительным движением между наблюдателем (глазом, камерой) и сценой



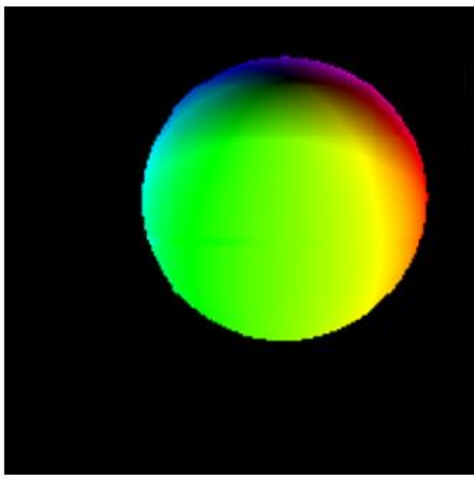
Первый кадр



Второй кадр

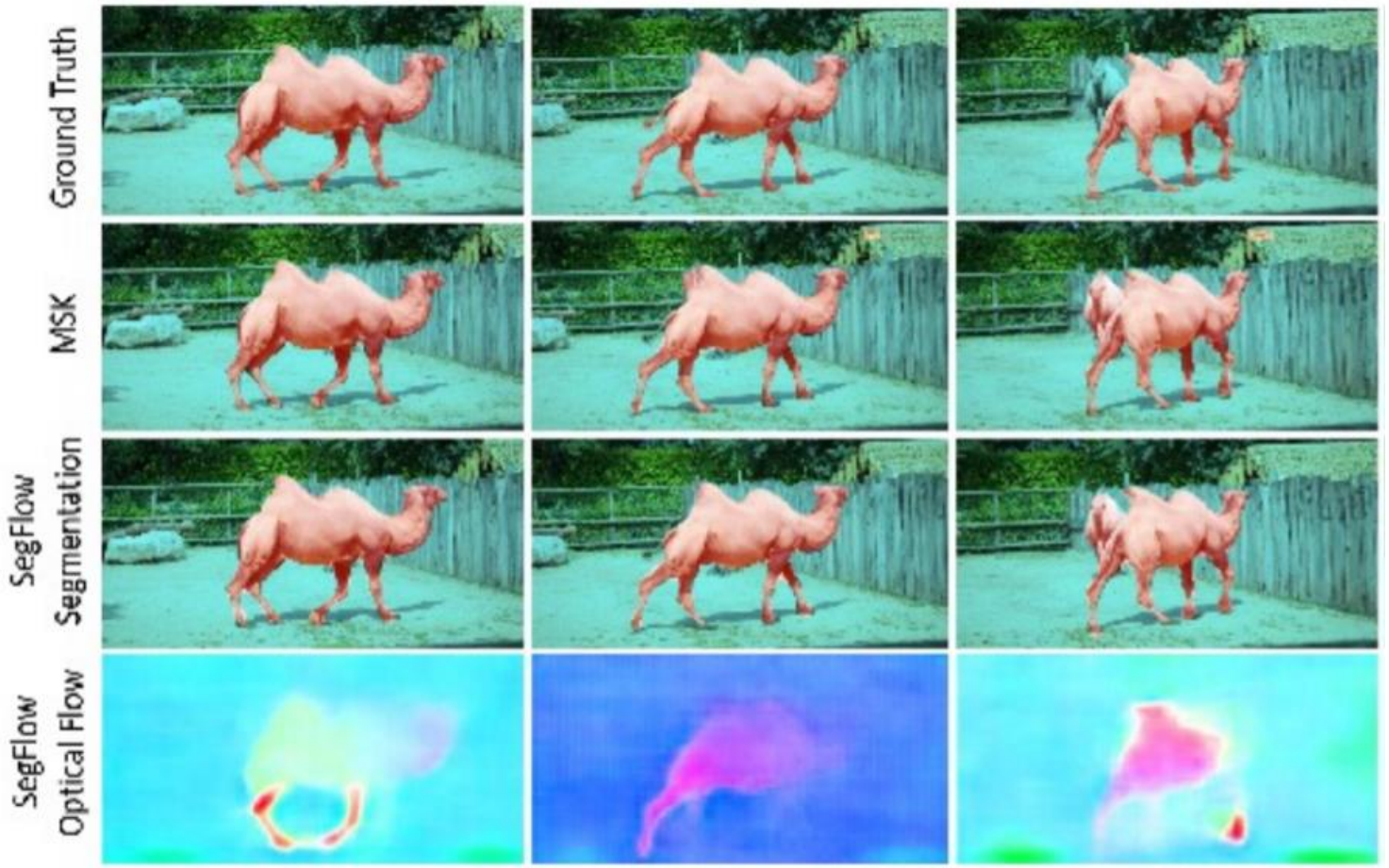


Оптический поток



Оптический поток (color coding)

Область применения
Сегментация

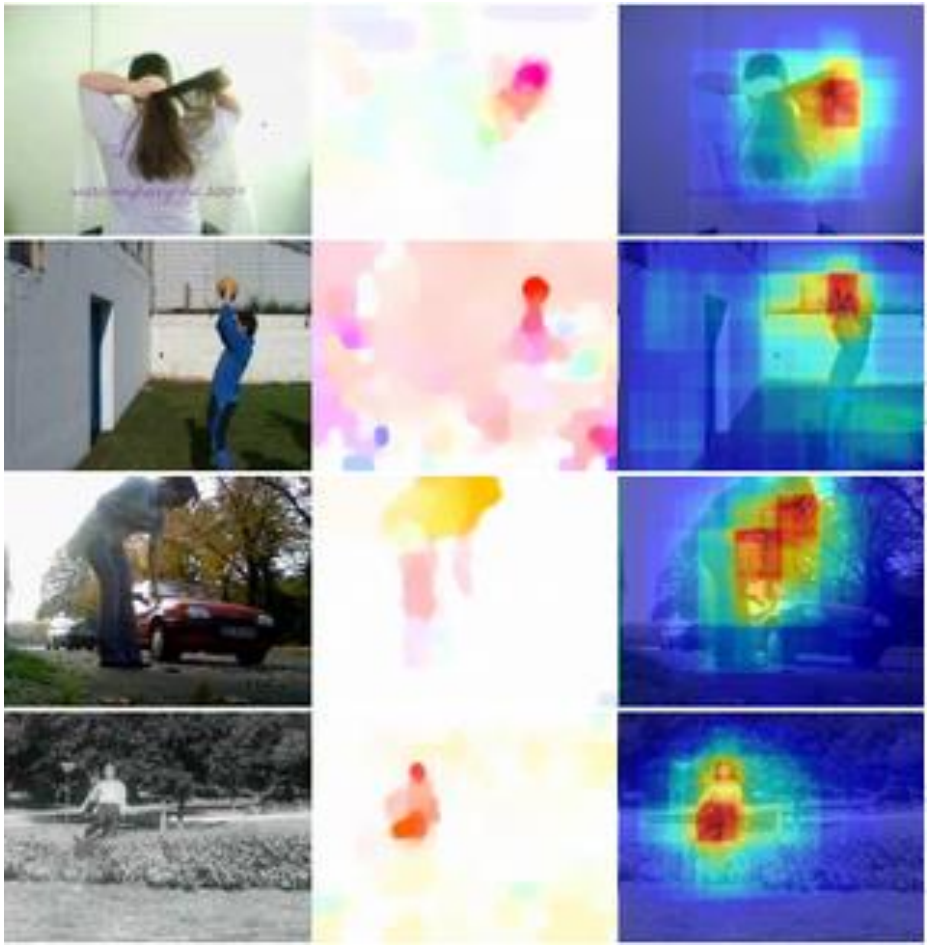


J. Cheng, et al., “Segflow: Joint learning for video object segmentation and optical flow,” in *ICIP*, 2017

Область применения

Saliency detection

Первый кадр Оптический поток Салиентность



Y. Zhou, et al., “Feature sampling strategies for action recognition,” in *ICIP*, 2017

Элементарное уравнение оптического потока

$$I(x+u, y+u) = I(x,y) + \frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v,$$

$$I(x+u, y+v) - H(x,y) = 0;$$

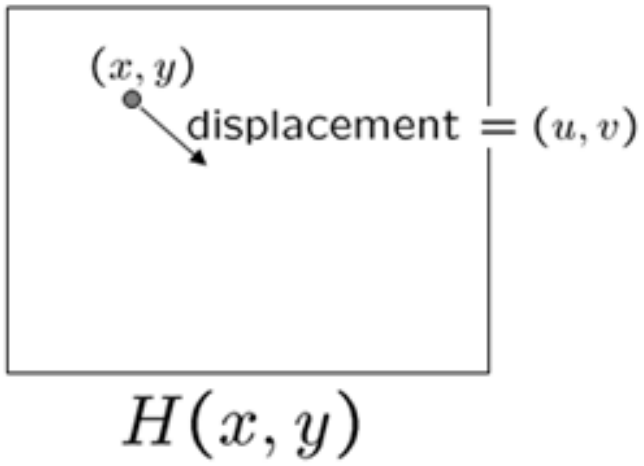
$$I(x,y) + I_x u + I_y v - H(x,y) = 0;$$

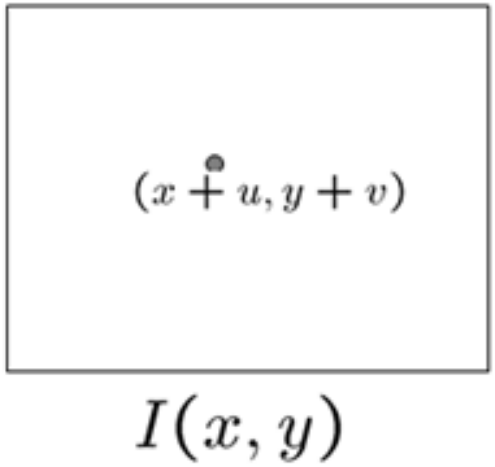
$$\left[I(x,y) - H(x,y)\right] + I_x u + I_y v = 0;$$

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

$$I_t + \nabla I(uv) = 0.$$

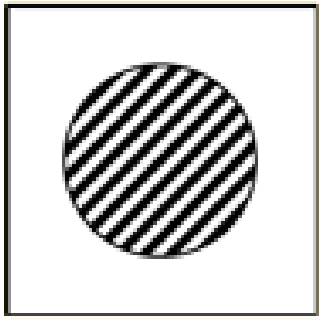
Получаем одно уравнение с двумя неизвестными, возникает неопределенность

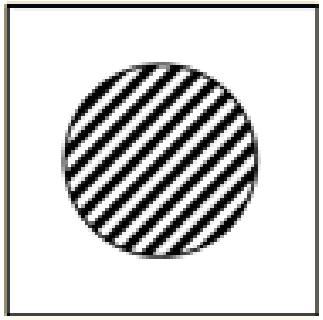


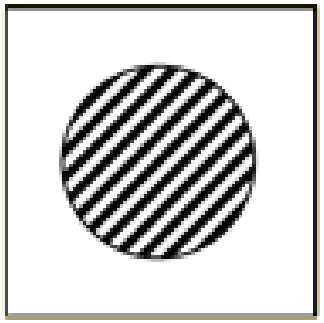


Aperture problem

Куда двигаются линии?

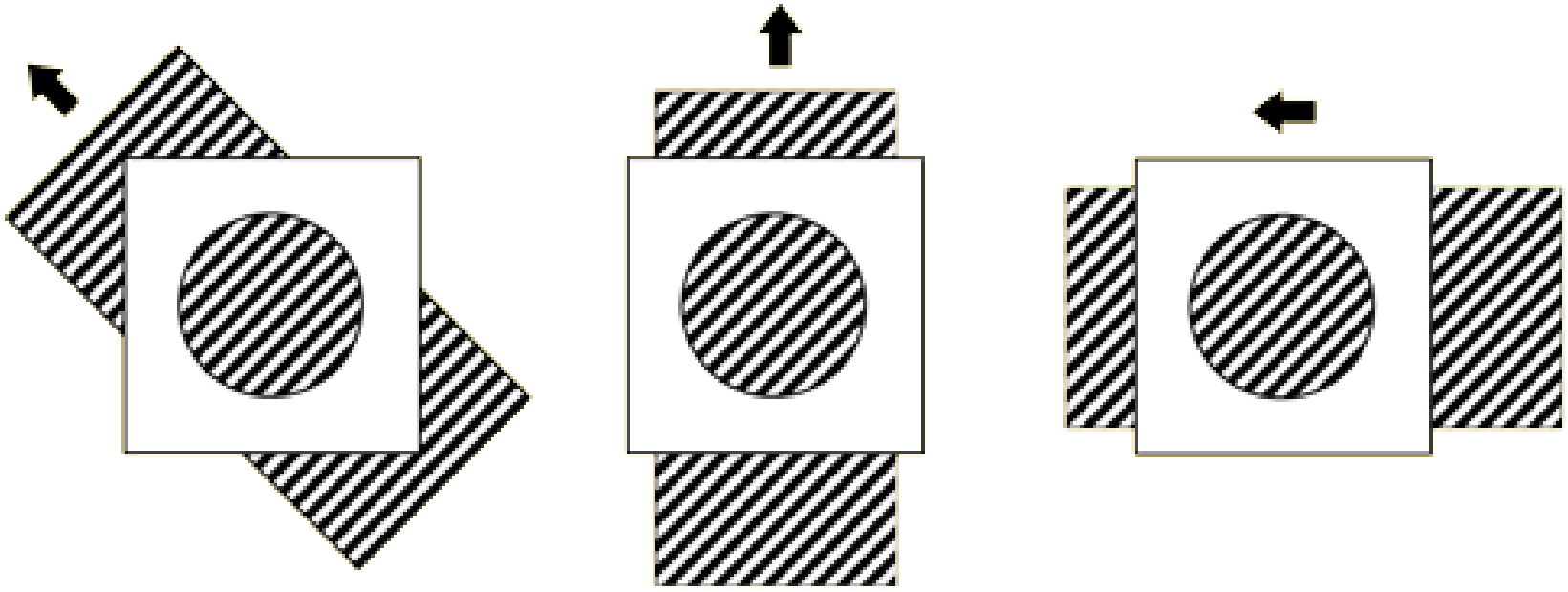






Aperture problem

Куда двигаются линии?



Ответ: невозможно определить

Вариационный подход и его модификации

Horn-Schunck:

$$E(u,v) = \int_\Omega (I_x u + I_y v + I_z)^2 + \alpha \left(|\nabla u|^2 + |\nabla v|^2 \right) \, dx \to \min$$

Robust smoothness term (Cohen 1993, Schnörr 1994)

Robust data term (Black-Anandan 1996, Mémin-Pérez 1996)

Gradient constancy (Brox et al. 2004)

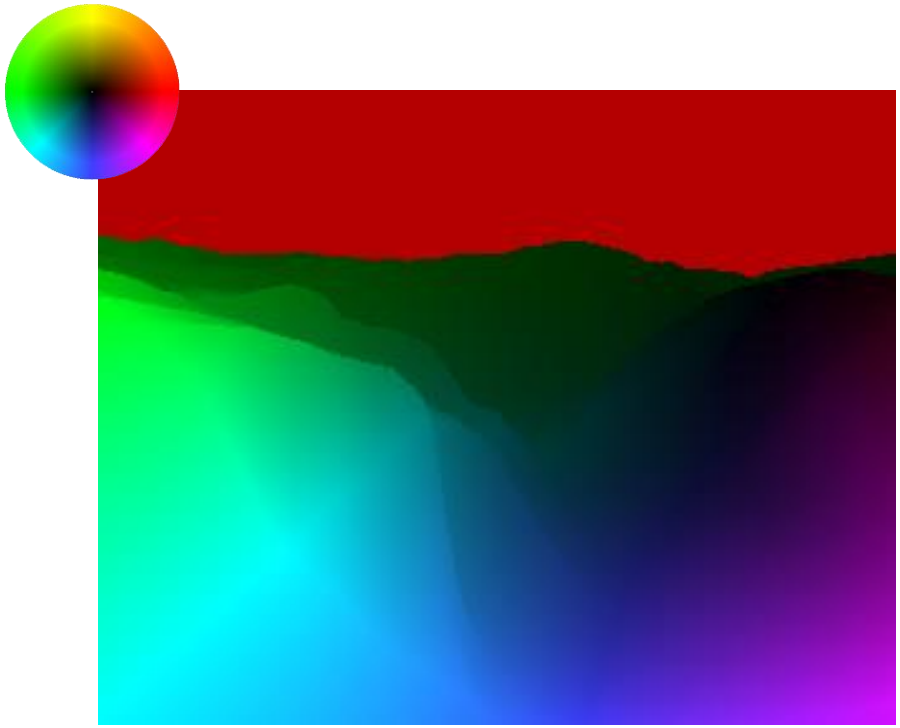
Non-linearized constancy (Nagel-Enkelmann 1986, Alvarez et al. 2000)

Spatiotemporal smoothness (Nagel 1990)

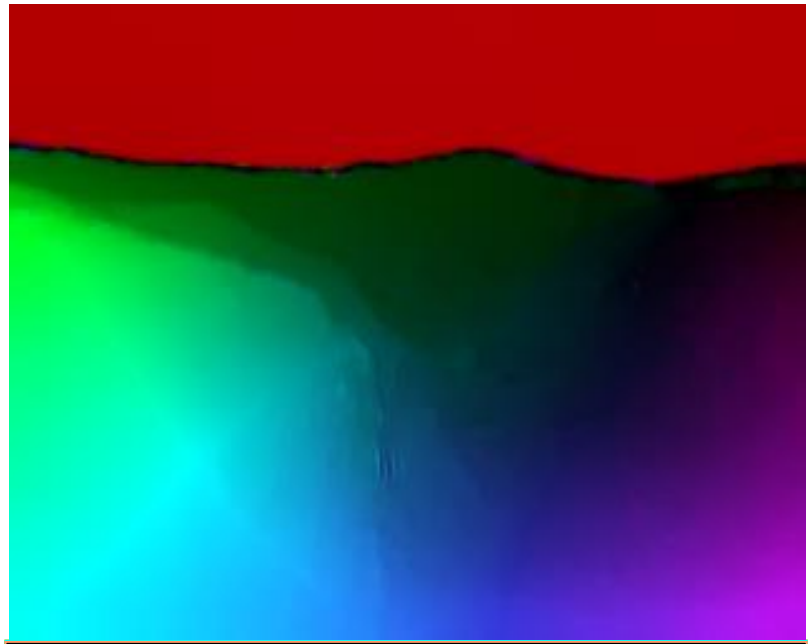
Финальная модель оптического потока:

$$\begin{aligned} E(u,v) = \int_\Omega & \Psi\Big((I(x+u,y+v,t+1) - I(x,y,t))^2 \\ & + \gamma |\nabla I(x+u,y+v,t+1) - \nabla I(x,y,t)|^2 \Big) \\ & + \alpha \Psi \left(|\nabla_3 u|^2 + |\nabla_3 v|^2 \right) \, dx \end{aligned}$$

Визуализация каждого улучшения

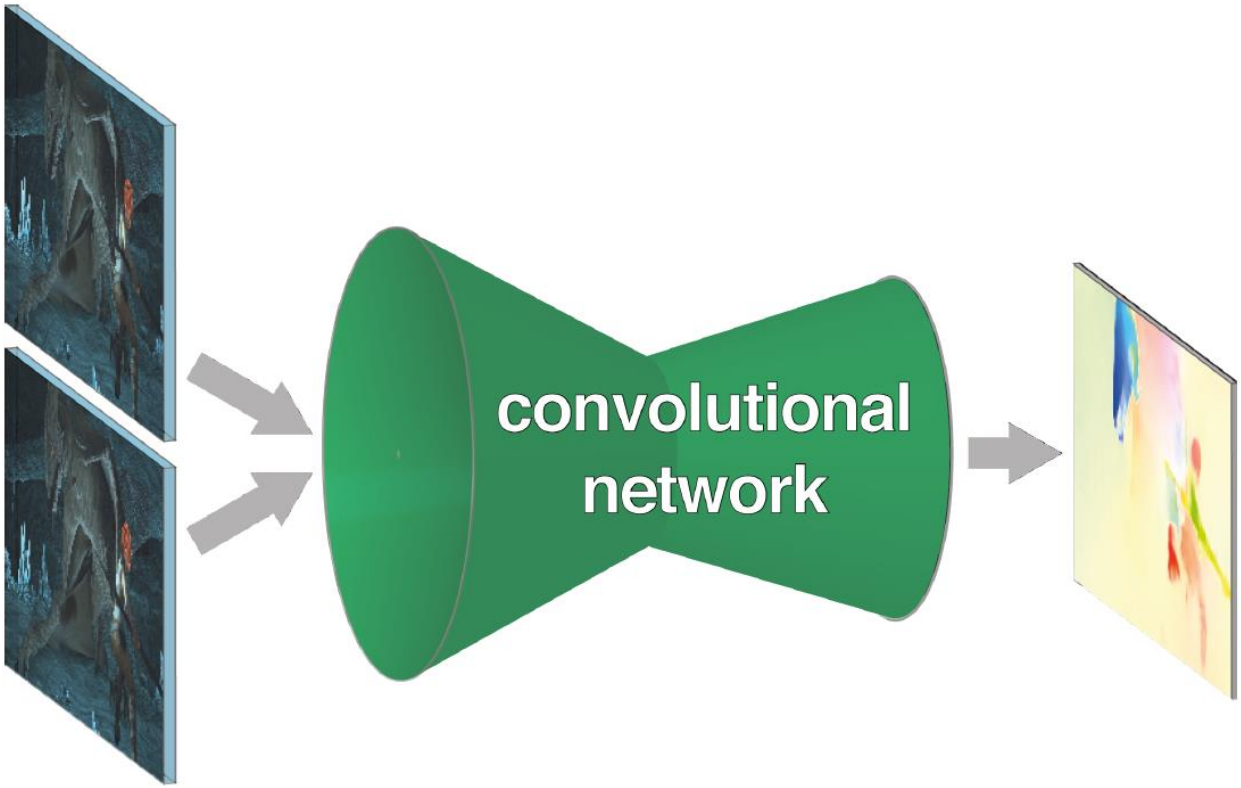


Ground truth



Spatiotemporal smoothness

FlowNet



Соседние кадры

Поле оптического потока

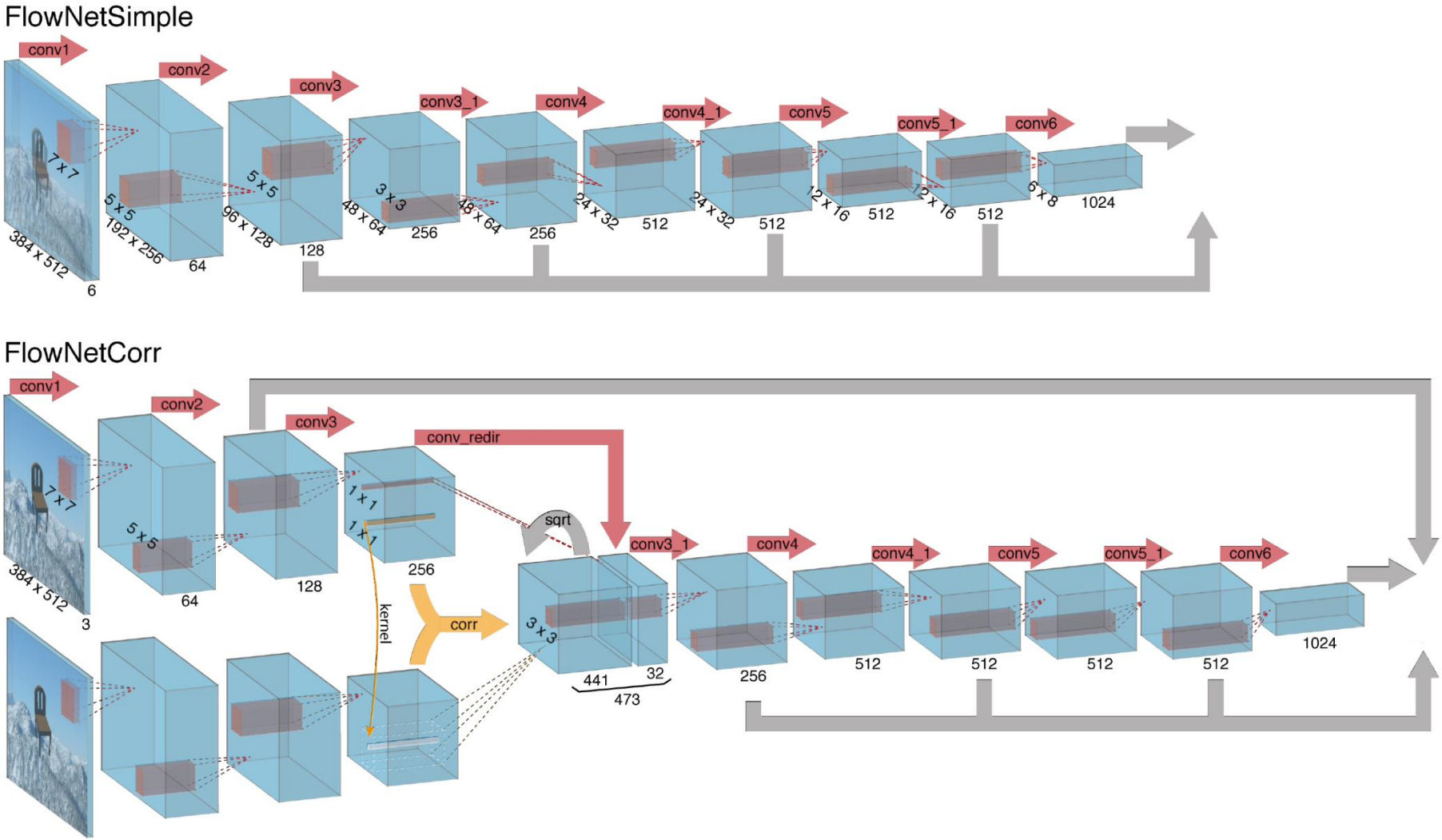
A. Dosovitskiy et al., “FlowNet: Learning Optical Flow With Convolutional Networks,” in *ICCV*, 2015

Обучающая выборка
Flying Chairs

- 964 изображения с Flickr в качестве фонов
- 809 3D-моделей стульев
- Движение: случайные аффинные преобразования фона и стульев
- Итого 22,872 пары изображений разрешения 512×384 с ground-truth OF

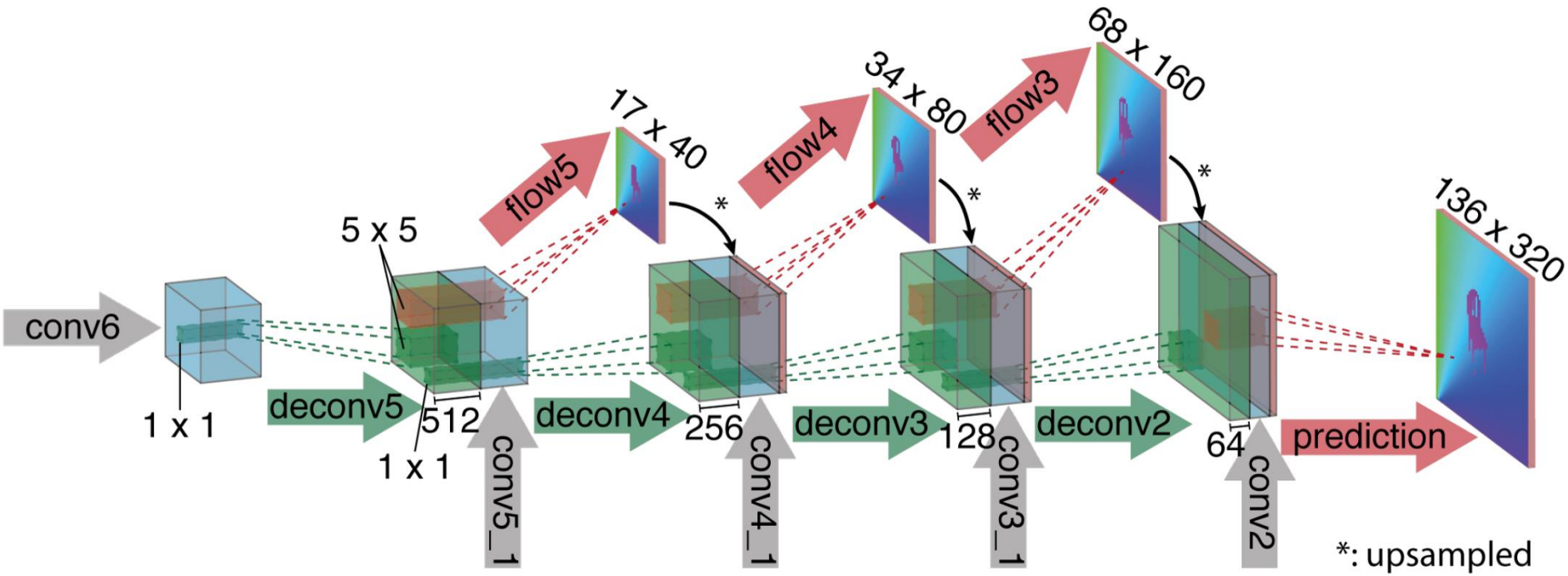
A. Dosovitskiy et al., “FlowNet: Learning Optical Flow With Convolutional Networks,” in *ICCV*, 2015

Архитектура нейросети
Сужающаяся часть



A. Dosovitskiy et al., “FlowNet: Learning Optical Flow With Convolutional Networks,” in *ICCV*, 2015

Архитектура нейросети
Расширяющаяся часть



A. Dosovitskiy et al., “FlowNet: Learning Optical Flow With Convolutional Networks,” in *ICCV*, 2015

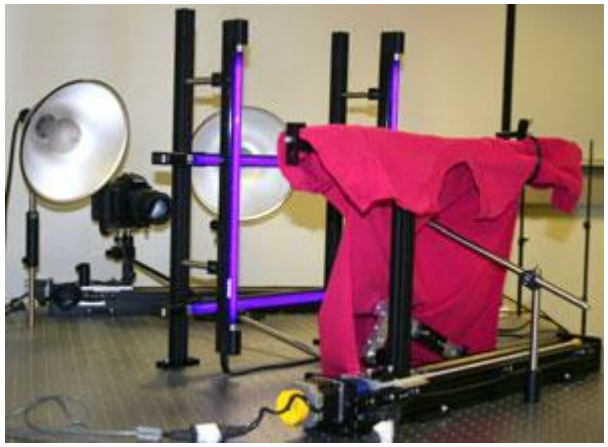
P. Fischer, A. Dosovitskiy, E. Ilg, P. Häusser, C. Hazırbas, V. Golkov
P. v.d. Smagt, D. Cremers, T. Brox

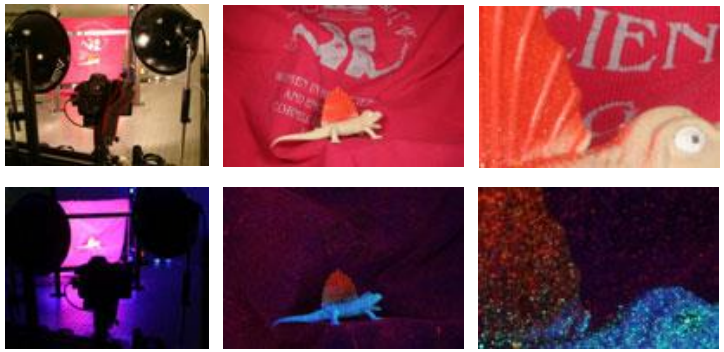
FlowNet: Learning Optical Flow with Convolutional Networks

Оптический поток
Экспериментальная оценка

Как сделать исходные данные для оценки алгоритмов оптического потока?

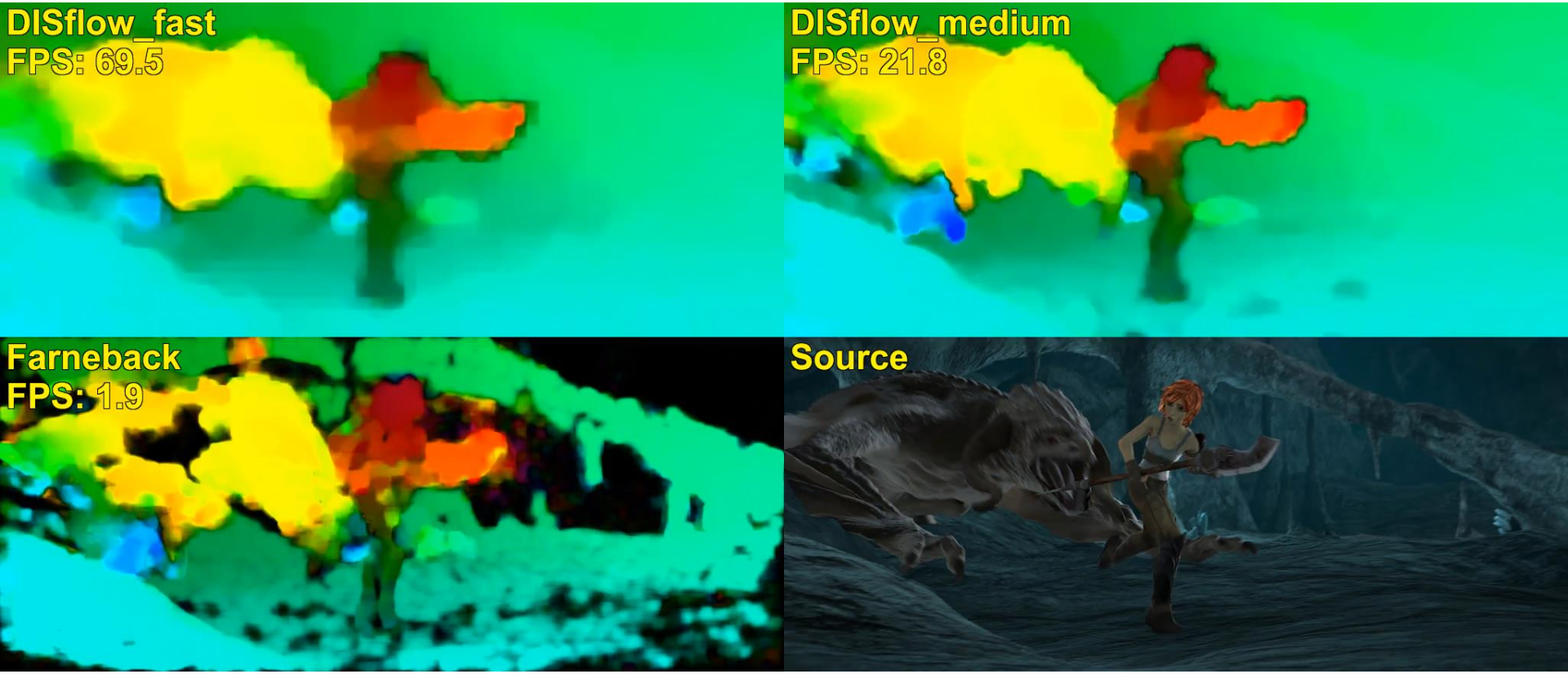
- Синтетика — Sintel, Flying Chairs, …
- Съёмка — датасет Middlebury



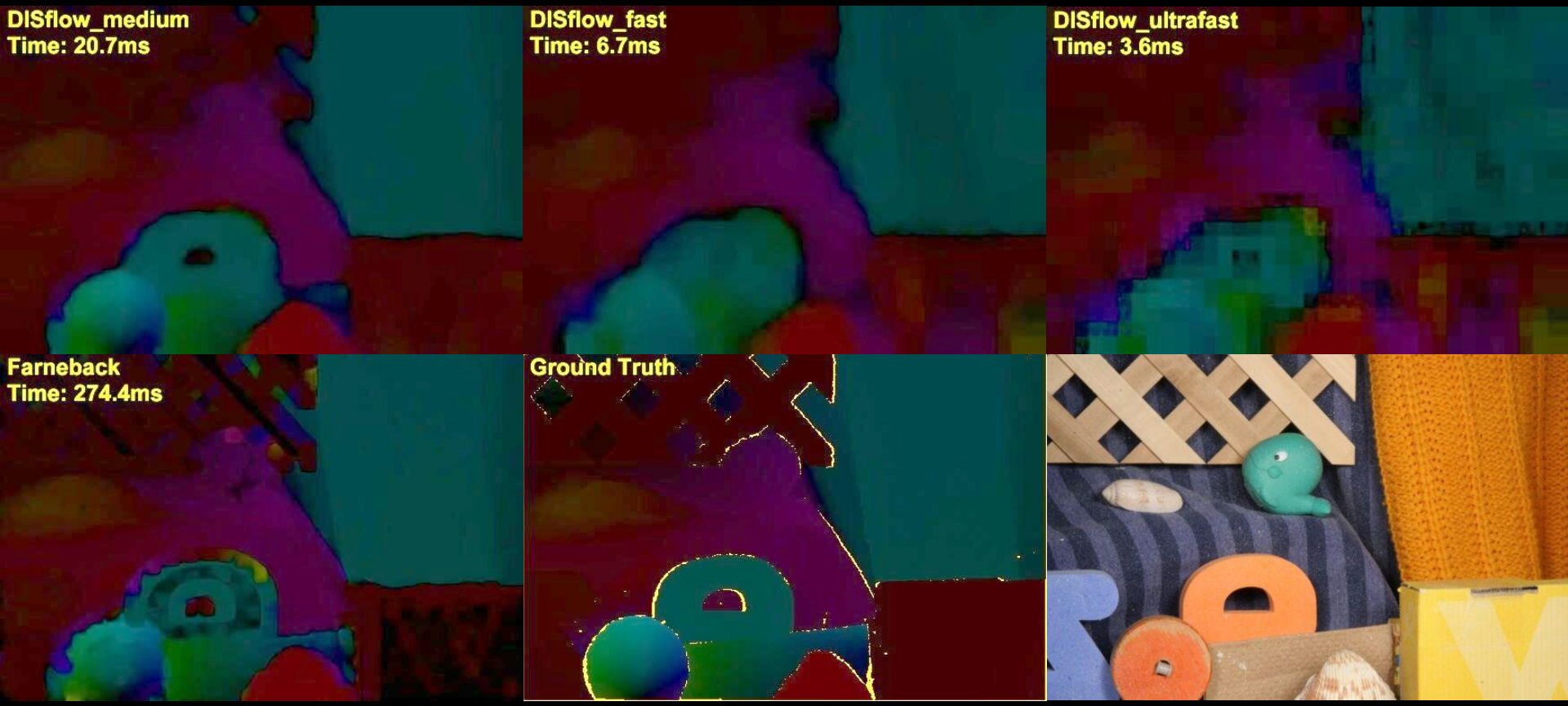


Экспериментальная оценка
Синтетический датасет на базе Sintel (1)

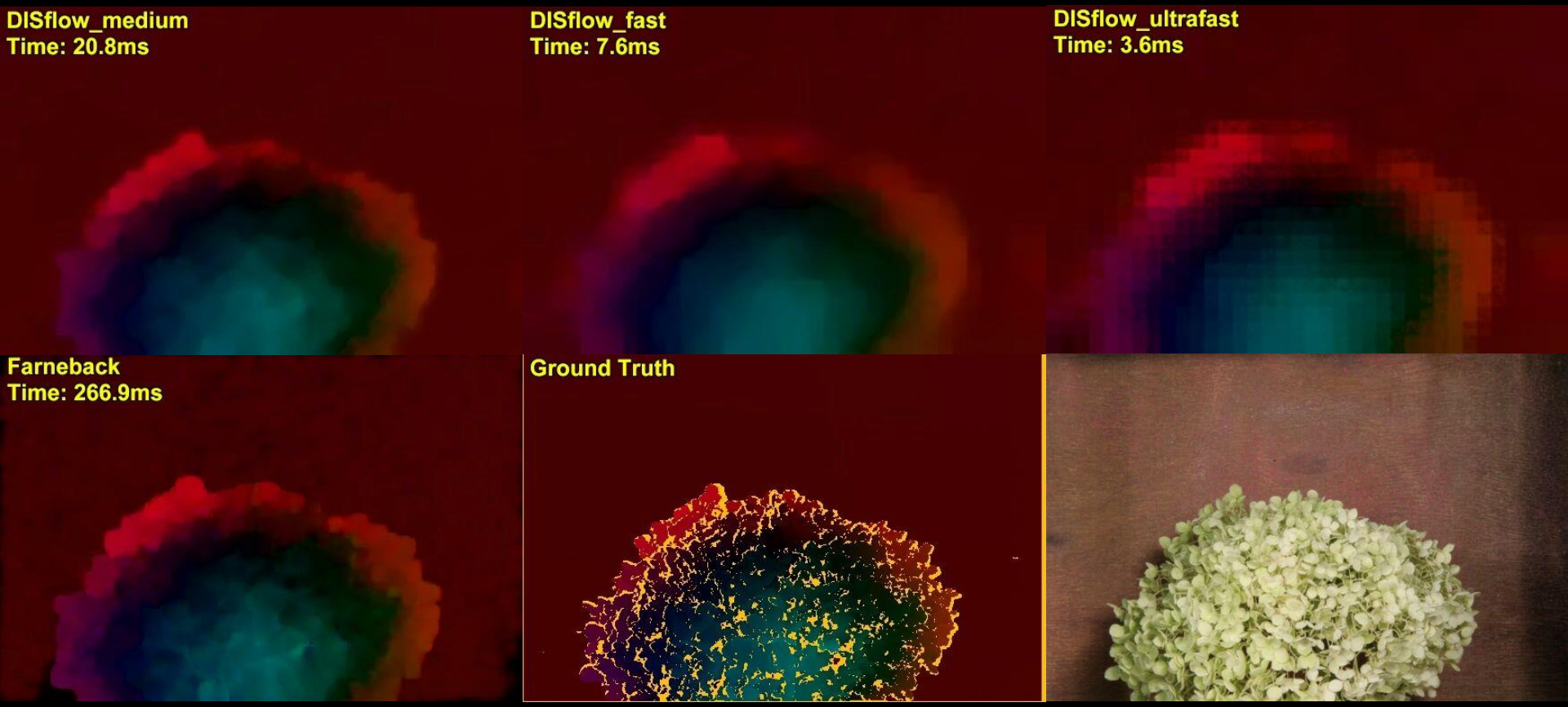
Экспериментальная оценка
Синтетический датасет на базе Sintel (2)



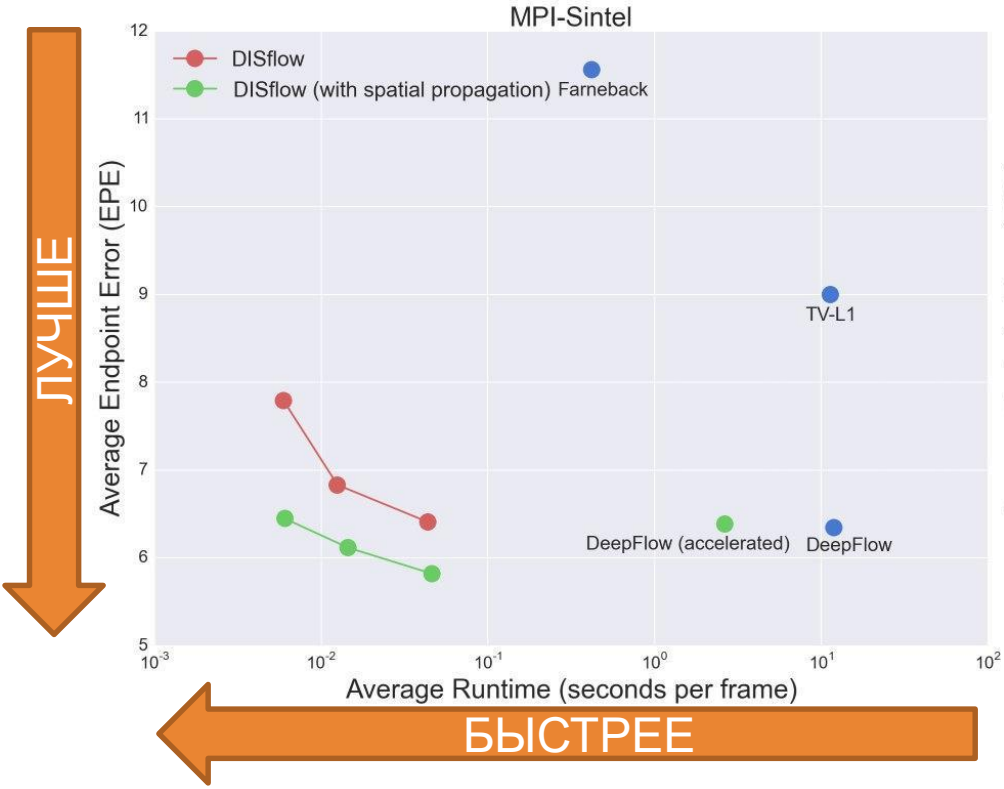
Экспериментальная оценка
Датасет Middlebury (1)

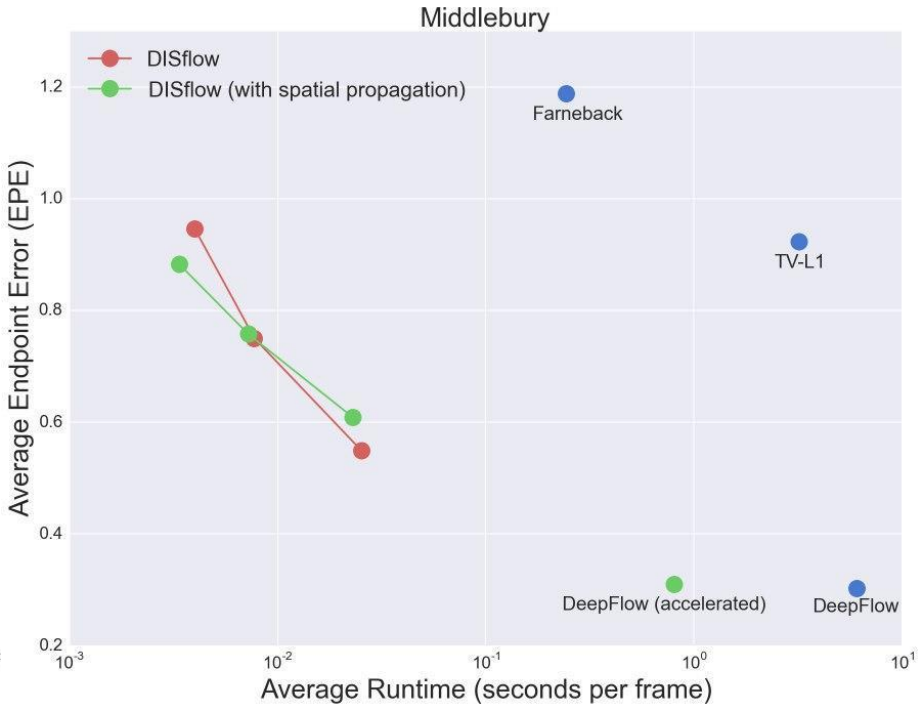


Экспериментальная оценка
Датасет Middlebury (2)



Экспериментальная оценка
Графики





Классические алгоритмы:

- Farneback (2003)
- TV-L1 (2008)

Нейросетевые алгоритмы:

- DeepFlow (2013)
- DISflow (2016)