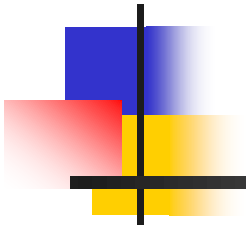
GRAPHICS & MEDIA LAB
VIDEO GROUP



Optical Flow на GPU

Илья Цветков

Video Group
CS MSU Graphics & Media Lab

Содержание

- Введение
- Gradient-based методы
  - Глобальная модель Horn & Schunck
  - Локальная модель Lucas & Kanade
- Energy-based метод
- Phase-based метод

Optical flow

Optical flow — векторное поле, определяющее скорость движения каждой точки изображения

- Задача компьютерного зрения
- Первые публикации — 1980 г.
- Возможные применения:
  - Сегментация
  - Изменение частоты кадров видео
  - Точная компенсация движения

Содержание

- Введение
- **Gradient-based методы**
  - Глобальная модель Horn & Schunck
  - Локальная модель Lucas & Kanade
- Energy-based метод
- Phase-based метод

Градиентные методы

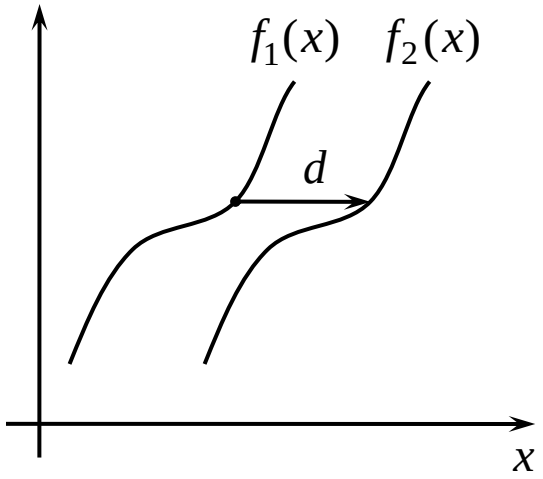
$f_1(x),\ f_2(x)$ — одномерные сигналы в различные моменты времени

$$f_2(x) = f_1(x-d)$$

$$f_1(x-d) = f_1(x) - d f_1'(x) + O(d^2)$$

$$f_1(x) - f_2(x) = d\, f_1'(x) + O(d^2)$$

$$\tilde{d} = \frac{f_1(x) - f_2(x)}{f_1'(x)}$$



Optical flow estimation. D. J. Fleet, Y. Weiss.
The Handbook in Mathematical models for Computer Vision.
N. Paragios, Y. Chen, and O. Faugeras, Springer, 2005

Градиентные методы

$I(\mathbf{x},t)$ — яркость изображения в точке $\mathbf{x}=(x,y)$ в момент времени t

$$I(\mathbf{x},t)=I(\mathbf{x}+\delta\mathbf{x},t+\delta t) \qquad \delta\mathbf{x}=(\delta x,\delta y)$$

$$I(\mathbf{x},t)=I(\mathbf{x},t)+\nabla I\cdot\delta\mathbf{x}+I_t\delta t+O\left(\delta x^2+\delta y^2+\delta t^2\right) \qquad \nabla I=(I_x,I_y)$$

$$\nabla I\cdot\mathbf{v}+I_t=0 \qquad \mathbf{v}=\frac{\delta\mathbf{x}}{\delta t}=(u,v)$$

Условие optical flow (optical flow constraint, gradient constraint equation):

$$\nabla I(\mathbf{x},t)\cdot\mathbf{v}(\mathbf{x},t)+I_t(\mathbf{x},t)=0$$

$$\mathbf{v}(\mathbf{x},t)=(u(\mathbf{x},t),v(\mathbf{x},t))$$

Optical flow estimation. D. J. Fleet, Y. Weiss.
The Handbook in Mathematical models for Computer Vision.
N. Paragios, Y. Chen, and O. Faugeras, Springer, 2005

Альтернативная постановка

$\mathbf{x}(t)$ — траектория движения точек изображения

$$I(\mathbf{x}(t), t) = const$$

$$\frac{d}{dt} I(\mathbf{x}(t), t) = 0$$

$$0 = \frac{d}{dt} I(\mathbf{x}(t), t) = \frac{\partial I}{\partial x}\frac{dx}{dt} + \frac{\partial I}{\partial y}\frac{dy}{dt} + \frac{\partial I}{\partial t} = \nabla I \cdot \mathbf{v} + I_t$$

В итоге получаем условие optical flow:

$$\nabla I(\mathbf{x}, t) \cdot \mathbf{v}(\mathbf{x}, t) + I_t(\mathbf{x}, t) = 0$$

Optical flow estimation. D. J. Fleet, Y. Weiss.
The Handbook in Mathematical models for Computer Vision.
N. Paragios, Y. Chen, and O. Faugeras, Springer, 2005

Aperture problem

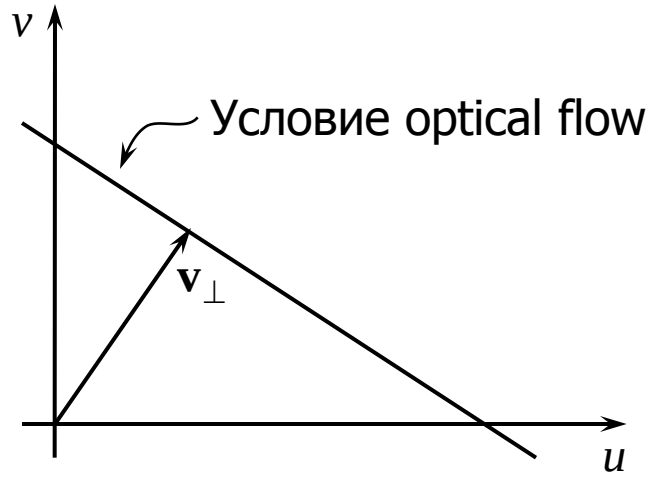
$$\nabla I(\mathbf{x},t)\cdot\mathbf{v}(\mathbf{x},t)+I_t(\mathbf{x},t)=0$$

$$\mathbf{v}(\mathbf{x},t)=(u(\mathbf{x},t),v(\mathbf{x},t))$$

Условие optical flow не является корректно поставленной задачей.

$\mathbf{v}_\perp$ — проекция $\mathbf{v}$ на вектор ∇I

$$\mathbf{v}_\perp=\frac{-I_t}{|\nabla I|}\frac{\nabla I}{|\nabla I|}$$



The Computation of Optical Flow. S. S. Beauchemin and J. L. Barron. ACM Computing Surveys, 1995

Ограничения модели

Модель:

$$I(\mathbf{x},t) = I(\mathbf{x}+\delta\mathbf{x},t+\delta t)$$

Сложности:

- Наложения
- Прозрачность, тени
- Отражения, блики

The Computation of Optical Flow. S. S. Beauchemin and J. L. Barron. ACM Computing Surveys, 1995

Сравнение optical flow

$$\mathbf{v} = (u, v) \quad \rightarrow \quad \vec{\mathbf{v}} = \frac{1}{\sqrt{u^2+v^2+1}}(u, v, 1)$$

Угловая ошибка (angular error) между векторами $\vec{\mathbf{v}}_c$ и $\vec{\mathbf{v}}_e$:

$$\psi_E = \arccos(\vec{\mathbf{v}}_c \cdot \vec{\mathbf{v}}_e)$$

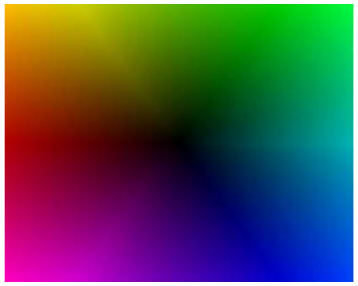
Средняя угловая ошибка (average angular error, AAE):

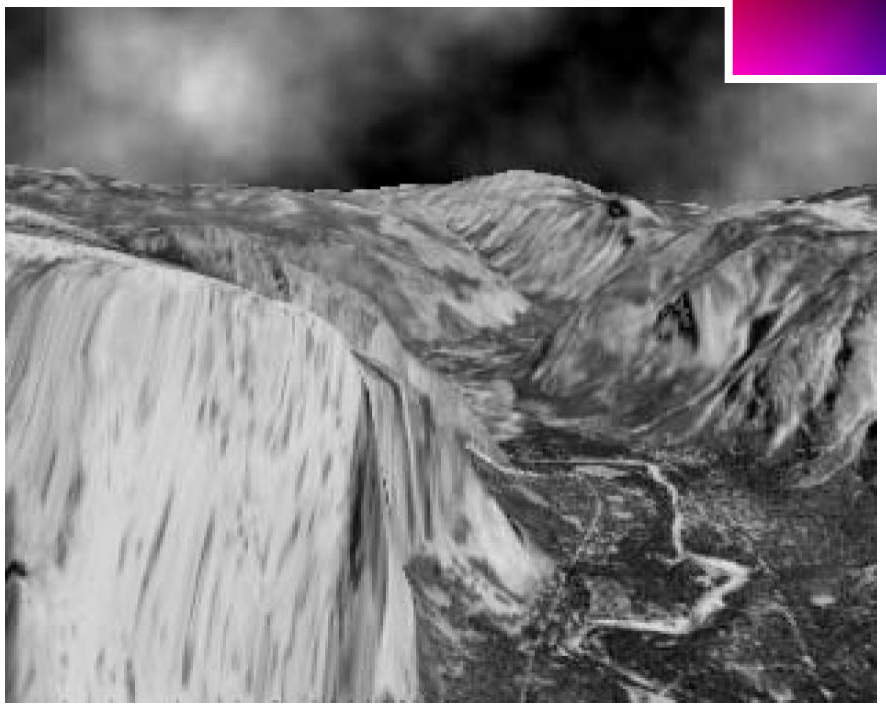
$$AAE = \frac{1}{|\Omega|}\sum_{\Omega}\psi_E$$

Performance of optical flow techniques. J. L. Barron, D. J. Fleet, and S. S. Beauchemin. International Journal of Computer Vision, 1994

Ground truth и визуализация

Последовательность Yosemite





Ground truth

Содержание

- Введение
- Gradient-based методы
  - **Глобальная модель Horn & Schunck**
  - Локальная модель Lucas & Kanade
- Energy-based метод
- Phase-based метод

Horn & Schunck

Цель — минимизация ошибки:

$$E(\mathbf{v}) = \int_{\Omega} \left[\underbrace{\left(\nabla I \cdot \mathbf{v} + I_t \right)^2}_{\text{Условие optical flow}} + \alpha^2 \underbrace{\left(\|\nabla u\|_2^2 + \|\nabla v\|_2^2 \right)}_{\text{Условие гладкости}} \right] d\mathbf{x}$$

α — параметр модели

$$\mathbf{v}(\mathbf{x}) = (u(\mathbf{x}), v(\mathbf{x}))$$

$$\nabla u = \left(\frac{\partial u}{\partial x}, \frac{\partial u}{\partial y} \right)$$

$$\nabla v = \left(\frac{\partial v}{\partial x}, \frac{\partial v}{\partial y} \right)$$

Сводим к системе дифференциальных уравнений:

$$I_x^2 u + I_x I_y v = \alpha^2 \nabla^2 u - I_x I_t$$

$$I_y^2 v + I_x I_y u = \alpha^2 \nabla^2 v - I_y I_t$$

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

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

Performance of Optical Flow Techniques on Graphics Hardware.
M. Durkovic, M. Zwick, F. Obermeier, K. Diepold.
IEEE International Conference on Multimedia and Expo, 2006

Horn & Schunck

$\nabla^2 a \approx \bar{a} - a$, где $\bar{a}$ — результат свёртки с ядром

$\frac{1}{12}$	$\frac{1}{6}$	$\frac{1}{12}$
$\frac{1}{6}$	0	$\frac{1}{6}$
$\frac{1}{12}$	$\frac{1}{6}$	$\frac{1}{12}$

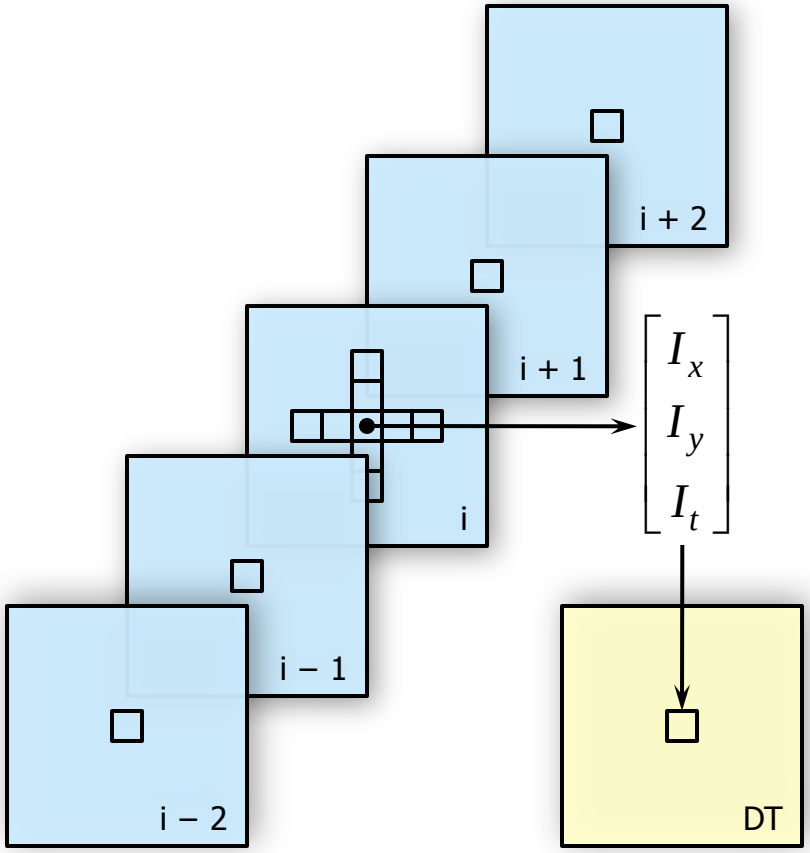
$$(\alpha^2 + I_x^2)u + I_x I_y v = (\alpha^2 \bar{u} - I_x I_t) \qquad (\alpha^2 + I_y^2)v + I_x I_y u = (\alpha^2 \bar{v} - I_y I_t)$$

Полученная СЛАУ решается итерационным методом:

$$u_{n+1} = \frac{\bar{u}_n - I_x \cdot \left(I_x \bar{u}_n + I_y \bar{v}_n + I_t\right)}{\alpha^2 + I_x^2 + I_y^2} \qquad v_{n+1} = \frac{\bar{v}_n - I_y \cdot \left(I_x \bar{u}_n + I_y \bar{v}_n + I_t\right)}{\alpha^2 + I_x^2 + I_y^2}$$

Performance of Optical Flow Techniques on Graphics Hardware.
M. Durkovic, M. Zwick, F. Obermeier, K. Diepold.
IEEE International Conference on Multimedia and Expo, 2006

Реализация на GPU



$$v_{n+1} = \frac{\bar{v}_n - I_y \cdot \left(I_x \bar{u}_n + I_y \bar{v}_n + I_t \right)}{\alpha^2 + I_x^2 + I_y^2}$$

$$u_{n+1} = \frac{\bar{u}_n - I_x \cdot \left(I_x \bar{u}_n + I_y \bar{v}_n + I_t \right)}{\alpha^2 + I_x^2 + I_y^2}$$

Производные вычисляются применением фильтра:

$$\frac{1}{12}\begin{bmatrix} -1 & 8 & 0 & -8 & 1 \end{bmatrix}$$

Performance of Optical Flow Techniques on Graphics Hardware.
M. Durkovic, M. Zwick, F. Obermeier, K. Diepold.
IEEE International Conference on Multimedia and Expo, 2006

Реализация на GPU

$$u_{n+1} = \frac{\bar{u}_n - I_x \cdot \left(I_x \bar{u}_n + I_y \bar{v}_n + I_t \right)}{\alpha^2 + I_x^2 + I_y^2} \qquad v_{n+1} = \frac{\bar{v}_n - I_y \cdot \left(I_x \bar{u}_n + I_y \bar{v}_n + I_t \right)}{\alpha^2 + I_x^2 + I_y^2}$$

Вычисление на GPU одной итерации:

$$\begin{bmatrix} u_{k+1} \\ v_{k+1} \\ \end{bmatrix} = \frac{\begin{bmatrix} \bar{u}_k \\ \bar{v}_k \\ 1 \end{bmatrix} - \left(\begin{bmatrix} \bar{u}_k & \bar{v}_k & 1 \end{bmatrix} \cdot \begin{bmatrix} I_x \\ I_y \\ I_t \end{bmatrix} \right) \begin{bmatrix} I_x \\ I_y \\ I_t \end{bmatrix}}{\begin{bmatrix} I_x & I_y & \alpha \end{bmatrix} \cdot \begin{bmatrix} I_x \\ I_y \\ \alpha \end{bmatrix}}$$

Performance of Optical Flow Techniques on Graphics Hardware.
M. Durkovic, M. Zwick, F. Obermeier, K. Diepold.
IEEE International Conference on Multimedia and Expo, 2006

Содержание

- Введение
- Gradient-based методы
  - Глобальная модель Horn & Schunck
  - **Локальная модель Lucas & Kanade**
- Energy-based метод
- Phase-based метод

Lucas & Kanade

Цель — минимизация ошибки:

$$E(\mathbf{v}) = \sum_{\mathbf{x}\in\Omega} w(\mathbf{x})\left[\nabla I(\mathbf{x},t)\cdot\mathbf{v} + I_t(\mathbf{x},t)\right]^2$$

$w(\mathbf{x}) \geq 0, \mathbf{x} \in \Omega$ — весовая функция

Точка минимума — критическая точка:

$$\frac{\partial E(u,v)}{\partial u} = \sum_{\mathbf{x}\in\Omega} w(\mathbf{x})\left(u\,I_x^2 + v\,I_xI_y + I_xI_t\right) = 0$$

$$\frac{\partial E(u,v)}{\partial v} = \sum_{\mathbf{x}\in\Omega} w(\mathbf{x})\left(v\,I_y^2 + u\,I_xI_y + I_yI_t\right) = 0$$

Performance of Optical Flow Techniques on Graphics Hardware.
M. Durkovic, M. Zwick, F. Obermeier, K. Diepold.
IEEE International Conference on Multimedia and Expo, 2006

Lucas & Kanade

$$\frac{\partial E(u,v)}{\partial u} = \sum_{\mathbf{x}\in\Omega} w(\mathbf{x})\left(u I_x^2 + v I_x I_y + I_x I_t\right) = 0$$

$$\frac{\partial E(u,v)}{\partial v} = \sum_{\mathbf{x}\in\Omega} w(\mathbf{x})\left(v I_y^2 + u I_x I_y + I_y I_t\right) = 0$$

В матричном виде:

$$\mathbf{M}\mathbf{v} = \mathbf{b}$$

$$\mathbf{M} = \begin{bmatrix} \sum w I_x^2 & \sum w I_x I_y \\ \sum w I_y I_x & \sum w I_y^2 \end{bmatrix} \qquad \mathbf{b} = -\begin{bmatrix} \sum w I_x I_t \\ \sum w I_y I_t \end{bmatrix}$$

Для каждой точки изображения находим вектор скорости:

$$\mathbf{v} = \mathbf{M}^{-1}\mathbf{b}$$

Performance of Optical Flow Techniques on Graphics Hardware.
M. Durkovic, M. Zwick, F. Obermeier, K. Diepold.
IEEE International Conference on Multimedia and Expo, 2006

Реализация на GPU

- Формат текстур:

$$\left(I_x^2, I_x I_y, I_y^2\right)$$
$$\left(I_x I_t, I_y I_t\right)$$

- Элементы $\mathbf{M}$ и $\mathbf{b}$ вычисляются применением фильтра с ядром $w(\mathbf{x})$: $\frac{1}{16}(1, 4, 6, 4, 1)$

$$\mathbf{M} = \begin{bmatrix} \sum w I_x^2 & \sum w I_x I_y \\ \sum w I_y I_x & \sum w I_y^2 \end{bmatrix} \qquad \mathbf{b} = -\begin{bmatrix} \sum w I_x I_t \\ \sum w I_y I_t \end{bmatrix}$$

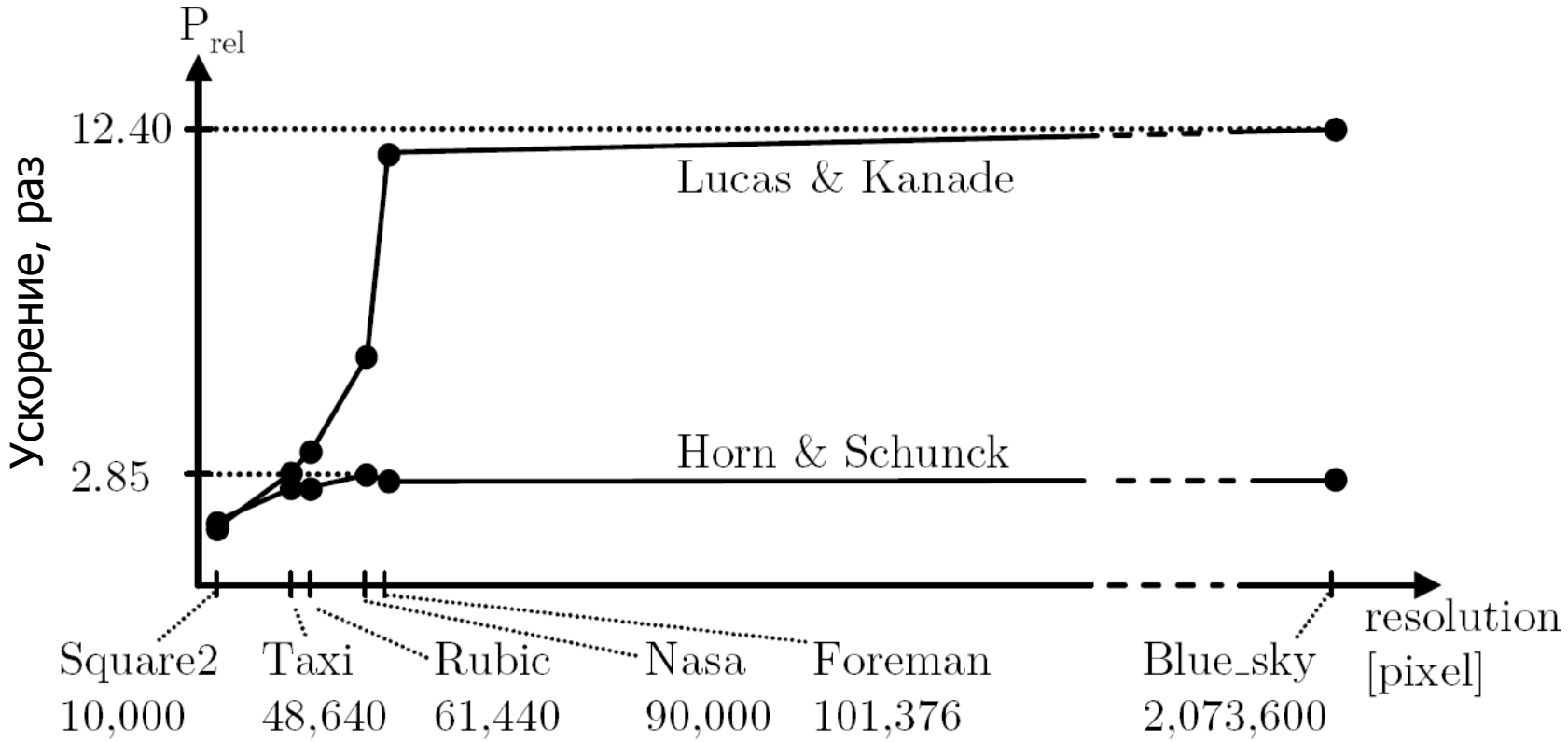
- Вычисление optical flow:

$$\mathbf{v} = \mathbf{M}^{-1}\mathbf{b}$$

Performance of Optical Flow Techniques on Graphics Hardware.
M. Durkovic, M. Zwick, F. Obermeier, K. Diepold.
IEEE International Conference on Multimedia and Expo, 2006

Результаты

Athlon64 3500+, 512 kB Cache, 2 GB RAM vs. GeForce 6800 Ultra

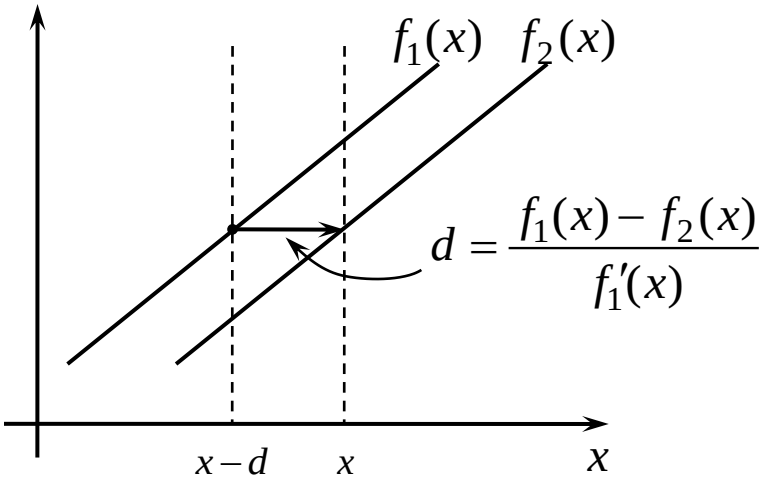


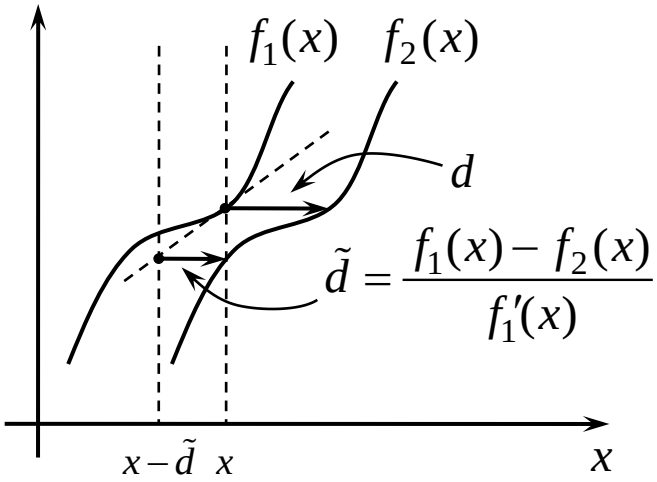
Performance of Optical Flow Techniques on Graphics Hardware.
M. Durkovic, M. Zwick, F. Obermeier, K. Diepold.
IEEE International Conference on Multimedia and Expo, 2006

Градиентные методы

- Достоинства
  - Простота
  - Высокая скорость работы
- Недостатки
  - Низкая точность

Уточнение метода





$$\left|\tilde{d} - d\right| \le \frac{d^2 \left| f_1''(x) \right|}{2 \left| f_1'(x) \right|} + O(d^3)$$

$f_1(x-d)$ — деформированный сигнал

Optical flow estimation. D. J. Fleet, Y. Weiss.
The Handbook in Mathematical models for Computer Vision.
N. Paragios, Y. Chen, and O. Faugeras, Springer, 2005

Уточнение метода

v^0 — приближение, полученное на первом шаге $\mathbf{v} = \mathbf{v}^0 + \delta\mathbf{v}^0$

$$I^0(\mathbf{x}, t+\delta t) = I(\mathbf{x} + \mathbf{v}^0 \delta t, t+\delta t)$$

$$I(\mathbf{x}, t) \equiv I^0(\mathbf{x}, t) = I^0(\mathbf{x} + \delta\mathbf{v}^0, t+1) \equiv I(\mathbf{x} + \mathbf{v}^0 + \delta\mathbf{v}^0, t+1)$$

$$\delta\tilde{\mathbf{v}}^0 = \mathbf{M}^{-1}\mathbf{b}^0$$

$\mathbf{v}^1 = \mathbf{v}^0 + \delta\tilde{\mathbf{v}}^0$ и т. д.

Итеративное приближение к точному решению:

$$\delta\tilde{\mathbf{v}}^j = \arg\min E^j(\delta\tilde{\mathbf{v}})$$

$$E^j(\delta\mathbf{v}) = \sum_{x\in\Omega} w(\mathbf{x}) \left[\nabla I^j(\mathbf{x}, t) \cdot \delta\mathbf{v} + I_t^j(\mathbf{x}, t) \right]^2$$

Optical flow estimation. D. J. Fleet, Y. Weiss.
The Handbook in Mathematical models for Computer Vision.
N. Paragios, Y. Chen, and O. Faugeras, Springer, 2005

Coarse-To-Fine Refinement

Сильное движение не может быть верно определено с использованием рассмотренных методов.

$$I^0(\mathbf{x}, t) = I(\mathbf{x}, t)$$

$$I^{k+1}(\mathbf{x}) = \left(S(g \cdot I^k)\right)(\mathbf{x})$$

S — оператор прореживания

g — фильтр Гаусса

$$E^k(\mathbf{v}) = \sum_{x \in \Omega} w(\mathbf{x}) \left[\nabla I^k(\mathbf{x}, t) \cdot \mathbf{v} + I_t^k(\mathbf{x}, t)\right]^2$$

$$k = \overline{n, 0}$$

Optical flow estimation. D. J. Fleet, Y. Weiss.
The Handbook in Mathematical models for Computer Vision.
N. Paragios, Y. Chen, and O. Faugeras, Springer, 2005

Robust estimator

$$E(\mathbf{v}) = \sum_{\mathbf{x}\in\Omega} w(\mathbf{x})\left[\nabla I(\mathbf{x},t)\cdot\mathbf{v} + I_t(\mathbf{x},t)\right]^2$$ — least-squares (LS) estimator

$$e(\mathbf{x}) = \nabla I(\mathbf{x},t)\cdot\mathbf{v} + I_t(\mathbf{x},t)$$

$$E(\mathbf{v}) = \sum_{x\in\Omega} w(\mathbf{x})\rho(e(\mathbf{x}),\sigma)$$

Например, $\rho(e,\sigma) = \dfrac{e^2}{e^2+\sigma^2}$

Optical flow estimation. D. J. Fleet, Y. Weiss.
The Handbook in Mathematical models for Computer Vision.
N. Paragios, Y. Chen, and O. Faugeras, Springer, 2005

Содержание

- Введение
- Gradient-based методы
  - Глобальная модель Horn & Schunck
  - Локальная модель Lucas & Kanade
- **Energy-based метод**
- Phase-based метод

Energy-based метод

Цель — минимизация энергии:

$$E(\mathbf{u}) = E_{D_1}(\mathbf{u}) + \alpha E_{D_2}(\mathbf{u}) + \beta E_S(\mathbf{u})$$

$$\alpha, \beta \geq 0$$

$$E_{D_1}(\mathbf{u}) = \int_\Omega \rho\left(\left|I(\mathbf{x}+\mathbf{u}) - I(\mathbf{x})\right|^2\right) d\mathbf{x}$$

$$\rho(s^2) = \sqrt{s^2 + \sigma^2}$$

$$E_{D_2}(\mathbf{u}) = \int_\Omega \rho\left(\left|\nabla I(\mathbf{x}+\mathbf{u}) - \nabla I(\mathbf{x})\right|^2\right) d\mathbf{x}$$

$$E_S(\mathbf{u}) = \int_\Omega \rho\left(\left|\nabla u\right|^2 + \left|\nabla v\right|^2\right) d\mathbf{x}$$

σ — регуляризационный параметр

Towards ultimate motion estimation: combining highest accuracy with real-time performance. A. Bruhn, J. Weickert. IEEE International Conference on Computer Vision, 2005

Численное решение

СЛАУ для u_i, v_i:

$$0 = \Psi_{D_1 i}(\mathbf{S}_{11i}\delta u_i + \mathbf{S}_{12i}\delta v_i + \mathbf{S}_{13i}) + \alpha\Psi_{D_2 i}(\mathbf{T}_{11i}\delta u_i + \mathbf{T}_{12i}\delta v_i + \mathbf{T}_{13i})$$
$$- \beta \sum_{j\in\Omega(i)} \frac{\Psi_{S i} + \Psi_{S j}}{2} \frac{u_j + \delta u_j - u_i - \delta u_i}{h^2}$$

$$0 = \Psi_{D_1 i}(\mathbf{S}_{21i}\delta u_i + \mathbf{S}_{22i}\delta v_i + \mathbf{S}_{23i}) + \alpha\Psi_{D_2 i}(\mathbf{T}_{21i}\delta u_i + \mathbf{T}_{22i}\delta v_i + \mathbf{T}_{23i})$$
$$- \beta \sum_{j\in\Omega(i)} \frac{\Psi_{S i} + \Psi_{S j}}{2} \frac{v_j + \delta v_j - v_i - \delta v_i}{h^2}$$

$\Omega(i)$ — множество пространственных соседей точки i

$$\Psi_{D_1} = \Psi((\delta u, \delta v, 1)^{\mathrm{T}} \mathbf{S} (\delta u, \delta v, 1))$$
$$\Psi_{D_2} = \Psi((\delta u, \delta v, 1)^{\mathrm{T}} \mathbf{T} (\delta u, \delta v, 1))$$
$$\Psi_S = \Psi\left(\left|\nabla(u+\delta u)\right|^2 + \left|\nabla(v+\delta v)\right|^2\right)$$

$$\mathbf{S} = \mathbf{I}_\nabla \mathbf{I}_\nabla^{\mathrm{T}}$$
$$\mathbf{T} = \mathbf{I}_{\nabla x}\mathbf{I}_{\nabla x}^{\mathrm{T}} + \mathbf{I}_{\nabla y}\mathbf{I}_{\nabla y}^{\mathrm{T}}$$
$$\mathbf{I}_{\nabla x} = (I_{xx}, I_{yx}, I_{zx})^{\mathrm{T}}$$
$$\mathbf{I}_{\nabla y} = (I_{xy}, I_{yy}, I_{zy})^{\mathrm{T}}$$
$$\mathbf{I}_\nabla = (I_x, I_y, I_z)^{\mathrm{T}}$$
$$I_z = I(\mathbf{x}+\mathbf{v}) - I(\mathbf{x})$$

GPU-Based Multigrid: Real-Time Performance in High Resolution Nonlinear Image Processing. H. Grossauer, P. Thoman. Computer Vision Systems, Springer Berlin, 2008

Итерационное решение

$$\begin{pmatrix} \delta u^{n+1} \\ \delta v^{n+1} \end{pmatrix} = \frac{1}{3}\begin{pmatrix} \delta u^{n} \\ \delta v^{n} \end{pmatrix} + \frac{2}{3}\left(\mathbf{M}^n\right)^{-1}\begin{pmatrix} r_u^n \\ r_v^n \end{pmatrix}$$

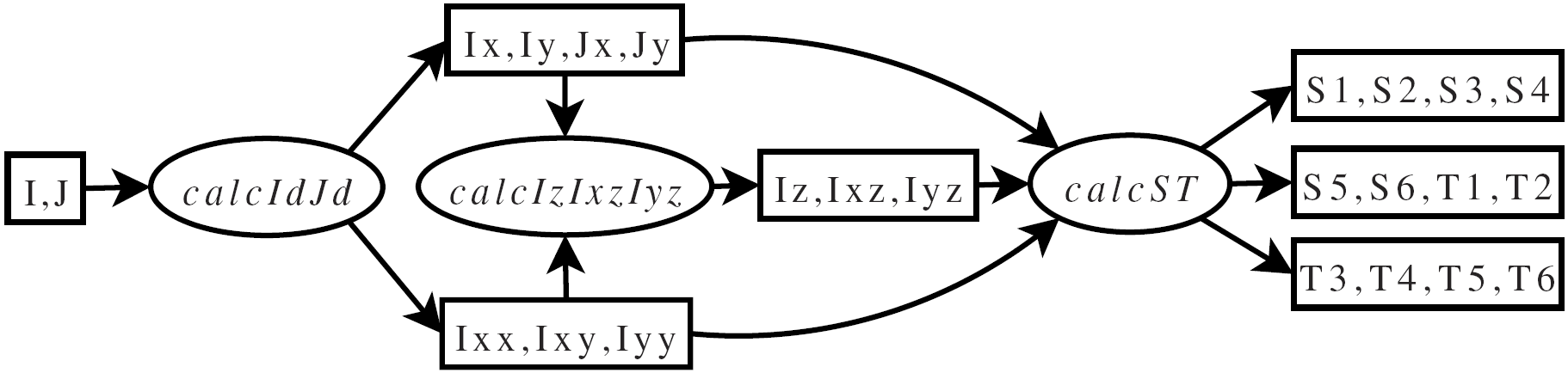
$$\mathbf{M}^n = \begin{bmatrix} \Psi^n_{D_1 i}\mathbf{S}^n_{11i} + \alpha\Psi^n_{D_2 i}\mathbf{T}^n_{11i} + \beta\sum\limits_{j\in\Omega(i)}\dfrac{\Psi^n_{S i}+\Psi^n_{S j}}{2h^2} & \Psi^n_{D_1 i}\mathbf{S}^n_{12i} + \alpha\Psi^n_{D_2}\mathbf{T}^n_{12i} \\ \Psi^n_{D_1 i}\mathbf{S}^n_{12i} + \alpha\Psi^n_{D_2}\mathbf{T}^n_{12i} & \Psi^n_{D_1 i}\mathbf{S}^n_{22i} + \alpha\Psi^n_{D_2 i}\mathbf{T}^n_{22i} + \beta\sum\limits_{j\in\Omega(i)}\dfrac{\Psi^n_{S i}+\Psi^n_{S j}}{2h^2} \end{bmatrix}$$

$$r_u^n = -\Psi^n_{D_1 i}\mathbf{S}^n_{13i} + \alpha\Psi^n_{D_2 i}\mathbf{T}^n_{13i} + \beta\sum_{j\in\Omega(i)}\frac{\Psi^n_{S i}+\Psi^n_{S j}}{2}\frac{u_j+\delta u_j^n-u_i}{h^2}$$

$$r_u^n = -\Psi^n_{D_1 i}\mathbf{S}^n_{23i} + \alpha\Psi^n_{D_2 i}\mathbf{T}^n_{23i} + \beta\sum_{j\in\Omega(i)}\frac{\Psi^n_{S i}+\Psi^n_{S j}}{2}\frac{v_j+\delta v_j^n-v_i}{h^2}$$

GPU-Based Multigrid: Real-Time Performance in High Resolution Nonlinear Image Processing. H. Grossauer, P. Thoman. Computer Vision Systems, Springer Berlin, 2008

Реализация на GPU



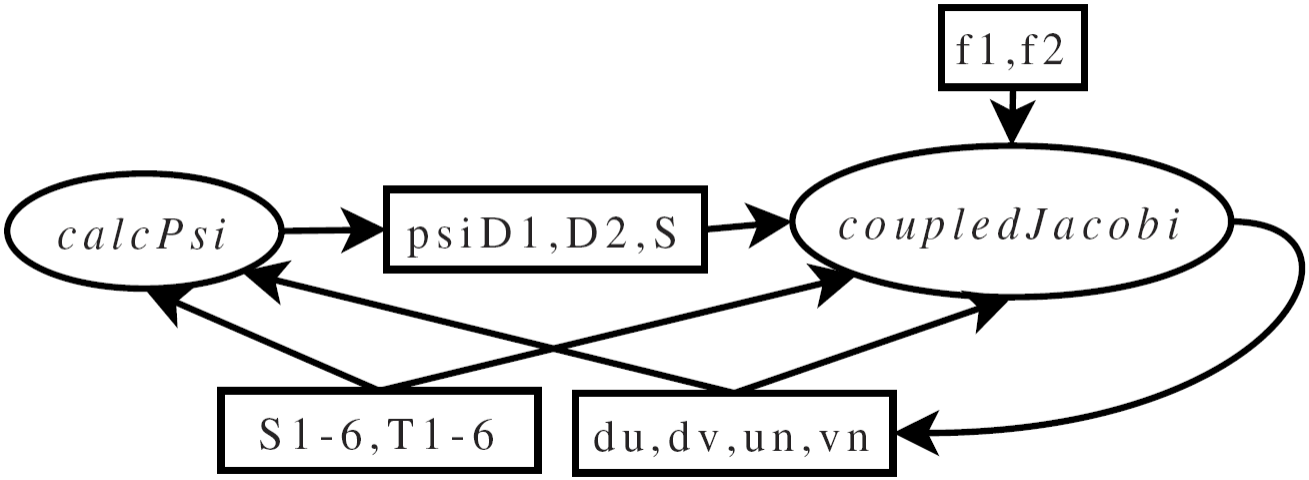
- calcIdJd — вычисление пространственных производных
- calcIzIxzIyz — вычисление производных по времени
- calcST — вычисление тензоров S и T:

$$\mathbf{S} = \mathbf{I}_{\nabla}\mathbf{I}_{\nabla}^{\mathrm{T}}$$

$$\mathbf{T} = \mathbf{I}_{\nabla x}\mathbf{I}_{\nabla x}^{\mathrm{T}} + \mathbf{I}_{\nabla y}\mathbf{I}_{\nabla y}^{\mathrm{T}}$$

GPU-Based Multigrid: Real-Time Performance in High Resolution Nonlinear Image Processing. H. Grossauer, P. Thoman. Computer Vision Systems, Springer Berlin, 2008

Реализация на GPU



$$vn = v + \delta v$$
$$un = u + \delta u$$

- calcPsi — предварительные вычисления
- coupledJacobi — вычисление одной итерации

GPU-Based Multigrid: Real-Time Performance in High Resolution Nonlinear Image Processing. H. Grossauer, P. Thoman. Computer Vision Systems, Springer Berlin, 2008

Результаты

Method	AAE	STD
Horn-Schunck, mod. [2]	9.78	16.19
Uras et al. [2]	8.94	15.61
Alvarez et al. [1]	5.53	7.40
Mémin-Pérez [14]	4.69	6.89
GPU-RT	**3.48**	**7.75**
GPU	**2.65**	**7.12**
Brox et al. [6]	2.46	7.31
Bruhn et al. [8]	2.42	6.70

- Последовательность Yosemite
- GPU и GPU-RT — предложенный метод, в GPU-RT применяется меньше итераций

GPU-Based Multigrid: Real-Time Performance in High Resolution Nonlinear Image Processing. H. Grossauer, P. Thoman. Computer Vision Systems, Springer Berlin, 2008

Результаты

Resolution (pixels)	Time (ms)	FPS
255^2	33.192	30.128
511^2	57.409	17.419
1023^2	206.107	4.852

- Производительность реализации GPU-RT
- GeForce 8800 GTX

GPU-Based Multigrid: Real-Time Performance in High Resolution Nonlinear Image Processing. H. Grossauer, P. Thoman. Computer Vision Systems, Springer Berlin, 2008

Содержание

- Введение
- Gradient-based методы
  - Глобальная модель Horn & Schunck
  - Локальная модель Lucas & Kanade
- Energy-based метод
- **Phase-based метод**

Phase-based метод

Основная идея — применить предварительную фильтрацию изображений.

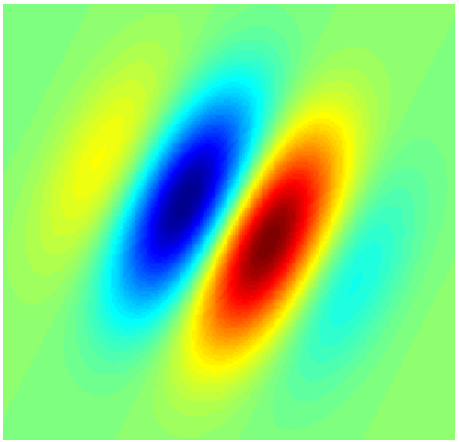
$$(I \cdot F)(\mathbf{x}, t) = (I \cdot F)(\mathbf{x} + \mathbf{v}\delta t, t + \delta t)$$

Используется фильтр Габора:

$$\mathbf{x} = (x, y)^{\mathrm{T}}$$

$$G(\mathbf{x}, \mathbf{f}_\theta) = e^{\frac{-\|\mathbf{x}\|^2}{\sigma^2}} e^{i\mathbf{x}\cdot(2\pi\mathbf{f}_\theta)}$$

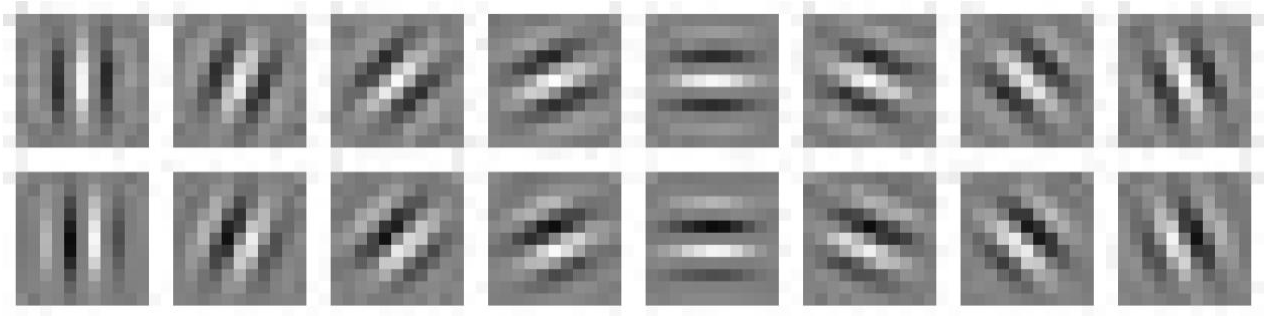
$\mathbf{f}_\theta = (f_{x\theta}, f_{y\theta})^{\mathrm{T}}$ — частота фильтра



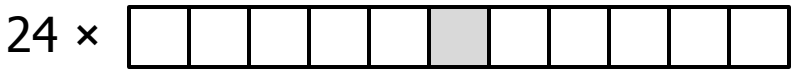
Realtime phase-based optical flow on the GPU. K. Pauwels, M. M. Van Hulle. IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2008

Фильтрация

- Проводится в различных направлениях:



- Используется приближение фильтров на основе одномерных свёрток:



Realtime phase-based optical flow on the GPU. K. Pauwels, M. M. Van Hulle. IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2008

Фильтрация

$R(\mathbf{x})$ — отфильтрованное изображение

$$R(\mathbf{x}) = (I \cdot G)(\mathbf{x}) = \rho(\mathbf{x}) e^{i\phi(\mathbf{x})}$$

$$\rho(\mathbf{x}) = \sqrt{\mathrm{Re}(R(\mathbf{x}))^2 + \mathrm{Im}(R(\mathbf{x}))^2}$$ — амплитуда

$$\phi(\mathbf{x}) = \arctan\left(\frac{\mathrm{Im}(R(\mathbf{x}))}{\mathrm{Re}(R(\mathbf{x}))}\right)$$ — фаза

Phase gradient constraint:

$$\nabla\phi \cdot \mathbf{v} + \psi = 0$$

$\mathbf{v}_\perp$ — проекция $\mathbf{v}$ на $\nabla\phi$

$$\mathbf{v}_\perp = \frac{-\psi}{|\nabla\phi|}\frac{\nabla\phi}{|\nabla\phi|}$$

$$\mathbf{v} = (u, v)$$

$$\nabla\phi = \left(\frac{\partial\phi}{\partial x}, \frac{\partial\phi}{\partial y}\right)^{\mathrm{T}}$$

$$\psi = \frac{\partial\phi}{\partial t}$$

Realtime phase-based optical flow on the GPU. K. Pauwels, M. M. Van Hulle. IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2008

Вычисление optical flow

$$\nabla \phi_\theta = 2\pi \mathbf{f}_\theta$$

$$\mathbf{v}_{\perp\theta}(\mathbf{x}) = \frac{\psi_\theta(\mathbf{x})}{2\pi|\mathbf{f}_\theta|}\frac{\mathbf{f}_\theta}{|\mathbf{f}_\theta|}$$

$$\left|\mathbf{v}_{\perp\theta}(\mathbf{x})\right| = \frac{\mathbf{v}(\mathbf{x})\cdot\mathbf{v}_{\perp\theta}(\mathbf{x})}{\left|\mathbf{v}_{\perp\theta}(\mathbf{x})\right|}$$

Цель — минимизация:

$$E(\mathbf{v}(\mathbf{x})) = \sum_{\theta\in O(\mathbf{x})}\left(\left|\mathbf{v}_{\perp\theta}(\mathbf{x})\right| - \frac{\mathbf{v}(\mathbf{x})\cdot\mathbf{v}_{\perp\theta}(\mathbf{x})}{\left|\mathbf{v}_{\perp\theta}(\mathbf{x})\right|}\right)^2$$

$O(\mathbf{x})$ — множество направлений θ, для которых имеется надёжное предсказание $\mathbf{v}_{\perp\theta}$

$\dfrac{\sum_{-\frac{n-1}{2}\le t\le\frac{n-1}{2}}\left(\left(a+\psi_\theta(\mathbf{x})t\right)-\phi_\theta(\mathbf{x},t)\right)^2}{n}$ — метрика доверия для $\psi_\theta(\mathbf{x})$ τ_l — порог доверия

n — количество рассматриваемых кадров, использовалось $n=5$

Realtime phase-based optical flow on the GPU. K. Pauwels, M. M. Van Hulle. IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2008

Coarse-To-Fine Refinement

- Метод может верно определять только вектора, которые меньше половины длины волны фильтра
- Пирамидальное вычисление:

$$I^{k+1}(\mathbf{x}) = \left(S(g \cdot I^k)\right)(\mathbf{x})$$

$$R^k(\mathbf{x}) = (I^k \cdot G)(\mathbf{x})$$

S — оператор прореживания

g — фильтр Гаусса

- Деформация изображения перед уточнением на более высоком разрежении:

$$p^{k-1}(\mathbf{x}, t) = \phi^{k-1}\left(x - 2\mathbf{v}^k(\mathbf{x})(t_c - t), t\right)$$

$$t_c = \frac{(n+1)}{2}$$

Realtime phase-based optical flow on the GPU. K. Pauwels, M. M. Van Hulle. IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2008

Реализация на CUDA

- Преимущества GPU:
  - Быстрая реализация свёртки
  - Высокая пропускная способность шины
  - Интерполяция «на лету»
- Основные идеи:
  - Банк фильтров реализован с помощью библиотеки CUDA
  - Вычисление optical flow происходит за одно применение ядра к каждому пикселю изображения
  - Расчёт деформированного изображения основан на интерполяции, предоставляемой GPU
  - Конвейерная обработка

Realtime phase-based optical flow on the GPU. K. Pauwels, M. M. Van Hulle. IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2008

Результаты

- GeForce 8800 GTX
- Измерения включают обмен данными между GPU и CPU

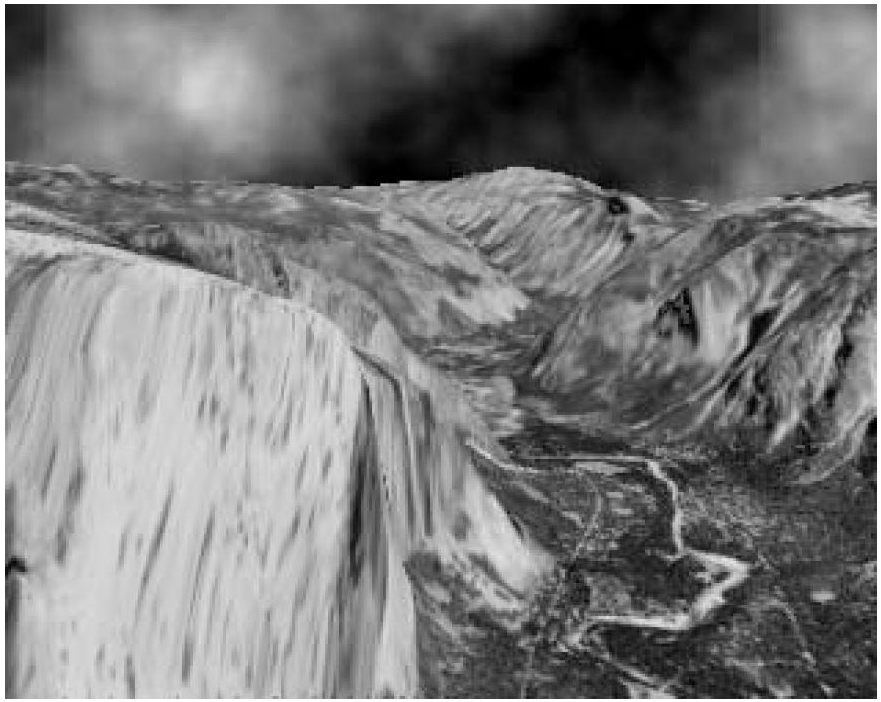
image resolution	320×256	640×512
Gabor pyramid	5.4	11.7
optical flow	2.4	8.9
total (**msec**)	7.8	20.6
frames per second	**127.6**	**48.5**

- Core 2 Quad 2.4 GHz
- Использовалось только одно ядро

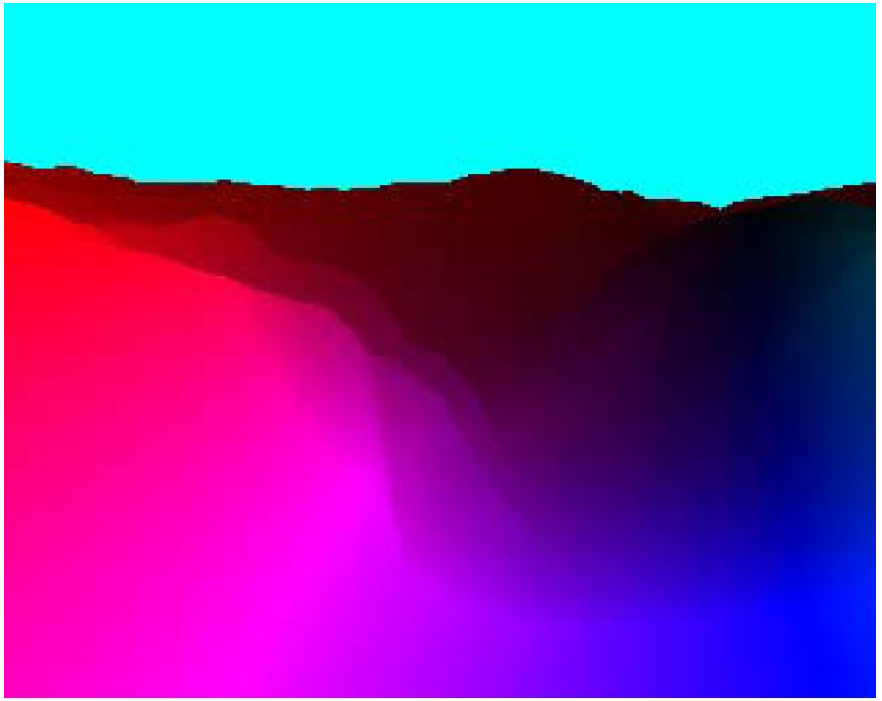
image resolution	320×256	640×512
Gabor pyramid	0.1	0.5
optical flow	1.1	4.8
total (**sec**)	1.2	5.3
frames per second	**0.8**	**0.2**

Realtime phase-based optical flow on the GPU. K. Pauwels, M. M. Van Hulle. IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2008

Результаты



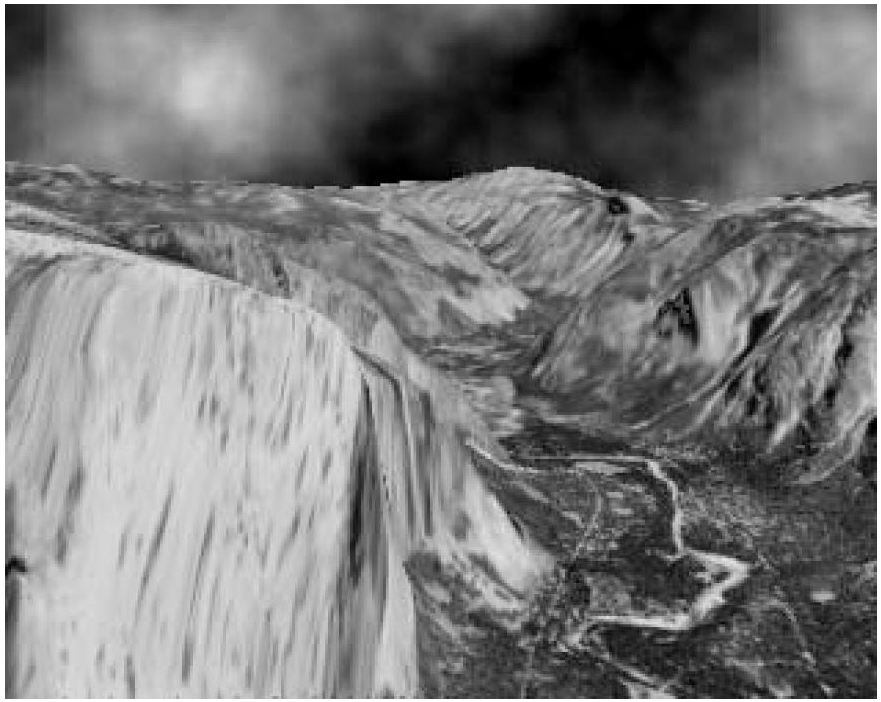
Исходный кадр



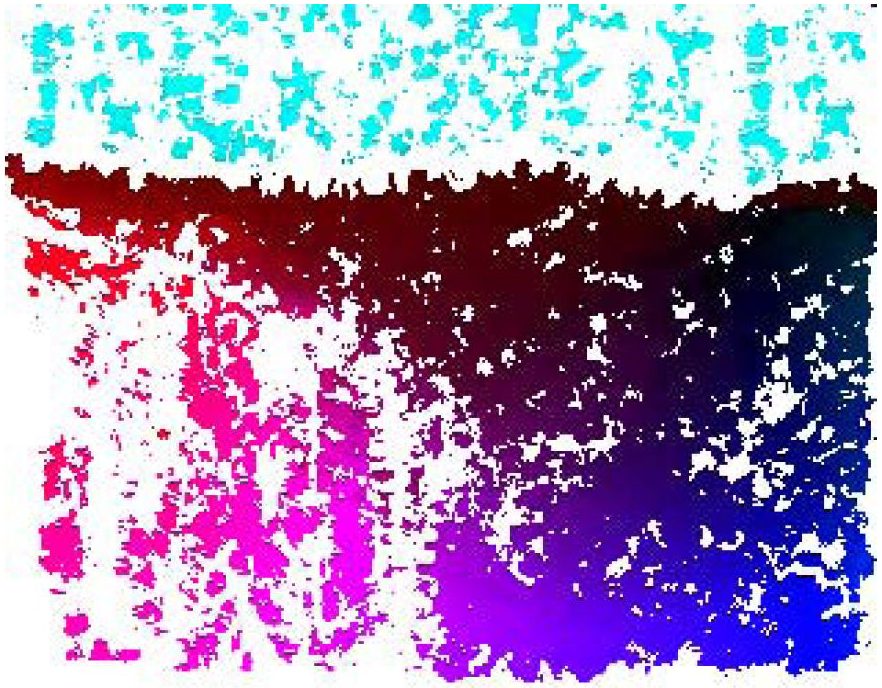
Ground truth

Realtime phase-based optical flow on the GPU. K. Pauwels, M. M. Van Hulle. IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2008

Результаты



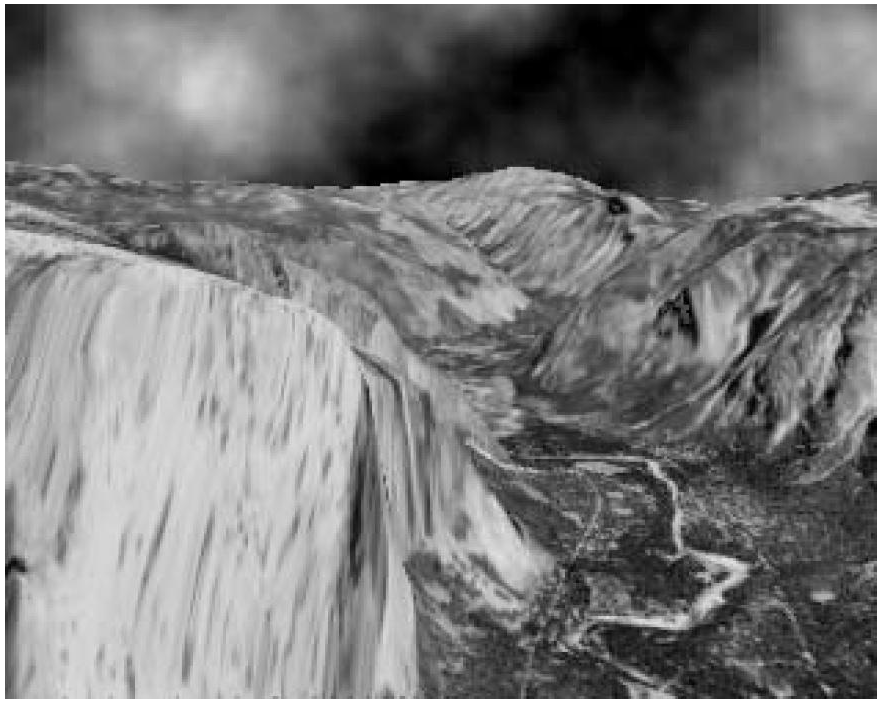
Исходный кадр



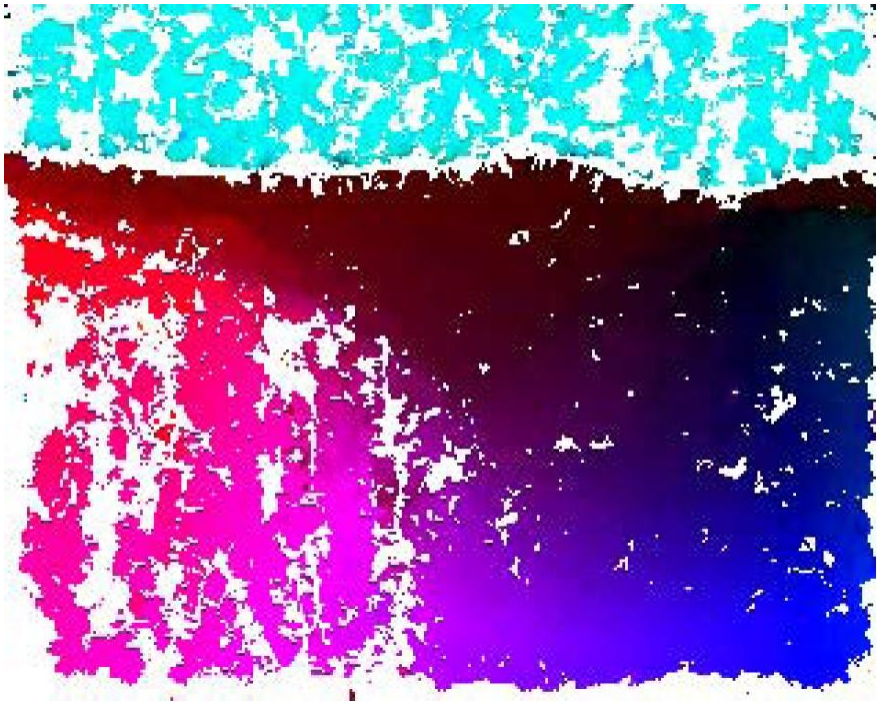
$\tau_l = 0.02$

Realtime phase-based optical flow on the GPU. K. Pauwels, M. M. Van Hulle. IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2008

Результаты



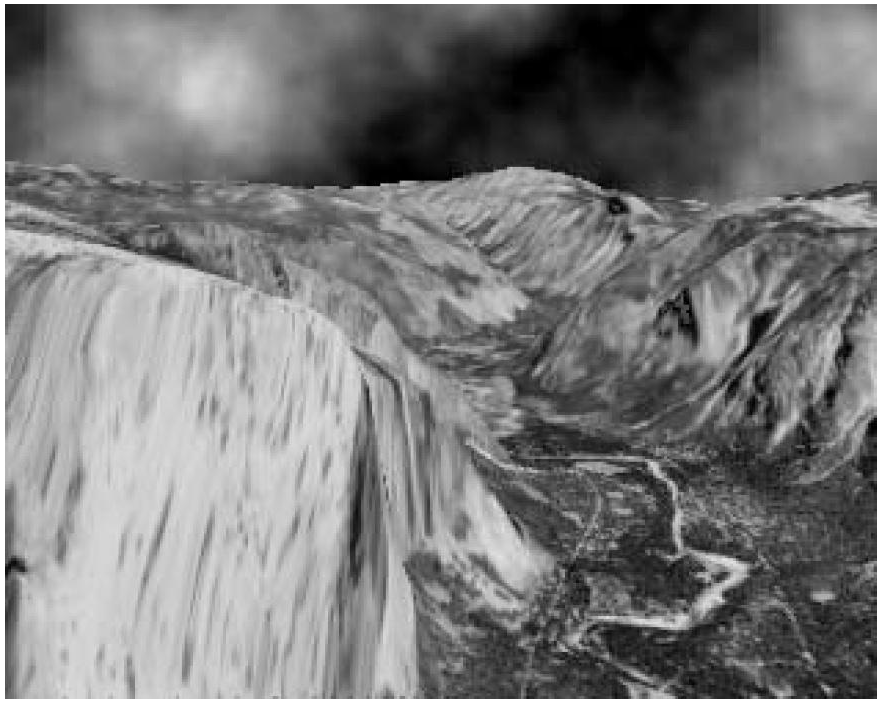
Исходный кадр



$\tau_l = 0.05$

Realtime phase-based optical flow on the GPU. K. Pauwels, M. M. Van Hulle. IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2008

Результаты

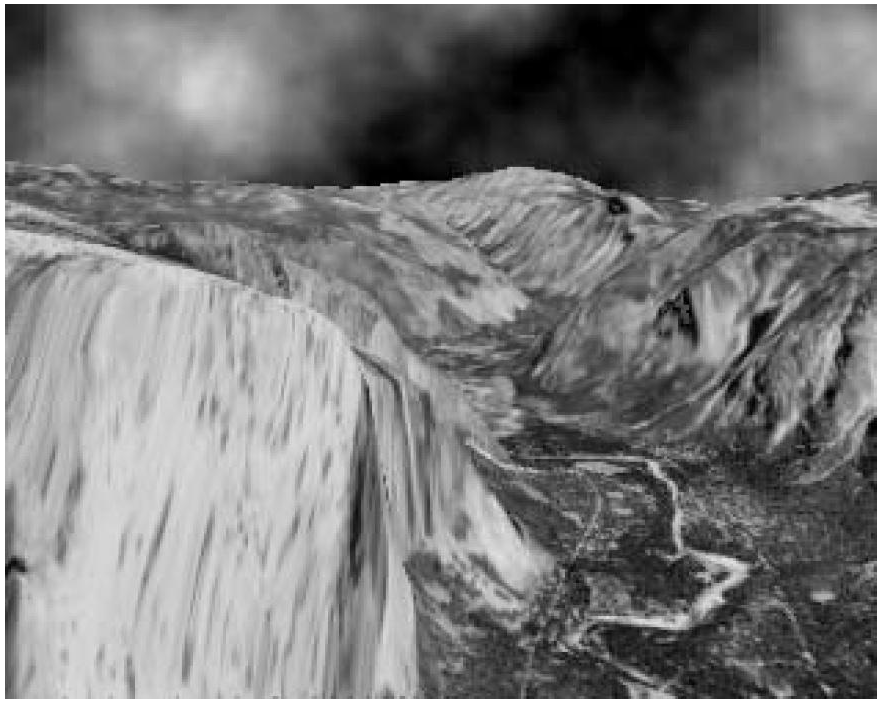


Исходный кадр

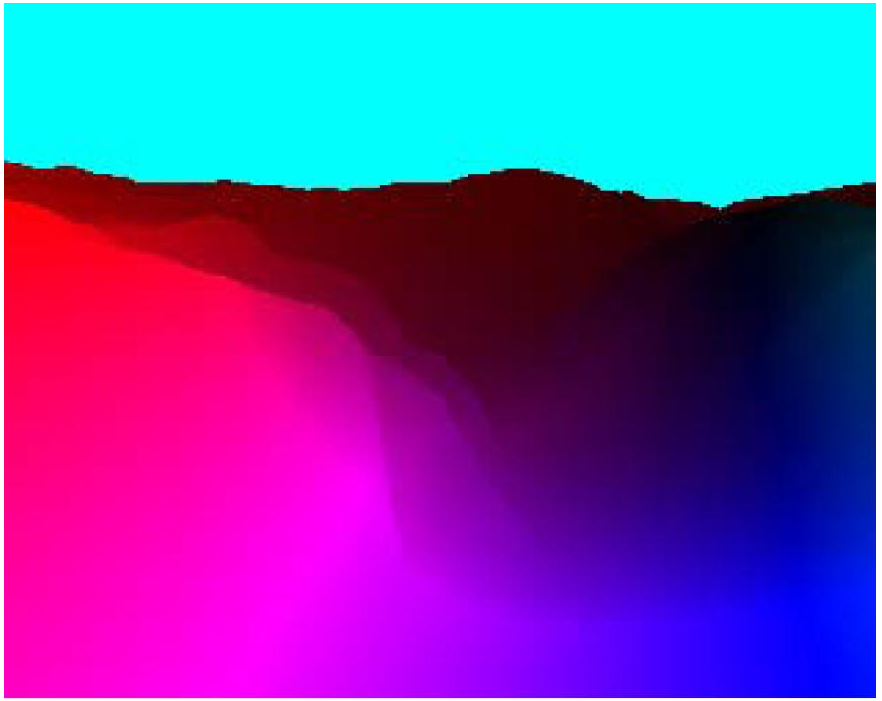
$\tau_l = 0.10$

Realtime phase-based optical flow on the GPU. K. Pauwels, M. M. Van Hulle. IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2008

Результаты



Исходный кадр



Ground truth

Realtime phase-based optical flow on the GPU. K. Pauwels, M. M. Van Hulle. IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2008

Результаты

- Phase-based

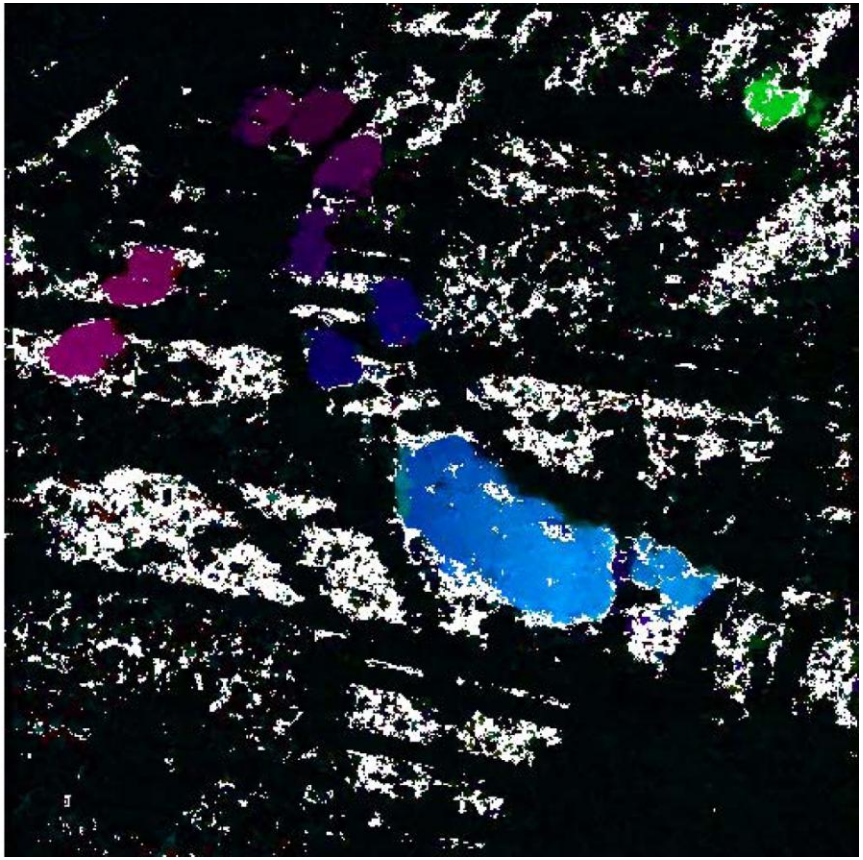
	without sky		with sky	
τ_l	AAE	density	AAE	density
0.02	2.09°	63 %	3.22°	54 %
0.05	2.35°	82 %	3.99°	76 %
0.10	2.67°	91 %	4.67°	88 %

- Другие методы

Method	AAE	STD
Horn-Schunck, mod. [2]	9.78	16.19
Uras et al. [2]	8.94	15.61
Alvarez et al. [1]	5.53	7.40
Mémin-Pérez [14]	4.69	6.89
GPU-RT	**3.48**	**7.75**
GPU	**2.65**	**7.12**
Brox et al. [6]	2.46	7.31
Bruhn et al. [8]	2.42	6.70

Realtime phase-based optical flow on the GPU. K. Pauwels, M. M. Van Hulle. IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2008

Результаты



Realtime phase-based optical flow on the GPU. K. Pauwels, M. M. Van Hulle. IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2008

Результаты

Realtime phase-based optical flow on the GPU. K. Pauwels, M. M. Van Hulle. IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2008

Список литературы

1. Optical flow estimation. Fleet, D. J. and Weiss, Y. The Handbook in Mathematical models for Computer Vision. N. Paragios, Y. Chen, and O. Faugeras, Springer, 2005
2. The Computation of Optical Flow. S. S. Beauchemin and J. L. Barron. ACM Computing Surveys, 1995
3. Performance of optical flow techniques. J. L. Barron, D. J. Fleet, and S. S. Beauchemin. International Journal of Computer Vision, 1994
4. Performance of Optical Flow Techniques on Graphics Hardware. M. Durkovic, M. Zwick, F. Obermeier, K. Diepold. IEEE International Conference on Multimedia and Expo, 2006
5. Towards ultimate motion estimation: combining highest accuracy with real-time performance. A. Bruhn, J. Weickert. IEEE International Conference on Computer Vision, 2005
6. GPU-Based Multigrid: Real-Time Performance in High Resolution Nonlinear Image Processing. H. Grossauer, P. Thoman. Computer Vision Systems, Springer Berlin, 2008
7. Realtime phase-based optical flow on the GPU. K. Pauwels, M. M. Van Hulle. IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2008

Вопросы

?

Лаборатория компьютерной графики и мультимедиа

Видеогруппа это:

- Выпускники в аспирантурах Англии, Франции, Швейцарии (в России в МГУ и ИПМ им. Келдыша)
- Выпускниками защищено 5 диссертаций
- Наиболее популярные в мире сравнения видеокодеков
- Более 3 миллионов скачанных фильтров обработки видео