

Введение в фильтрацию видеоданных (в рамках курса по CUDA)

Ватолин Дмитрий Сергеевич
К.ф.-м.н., с.н.с.

Видеогруппа лаборатории компьютерной
графики и мультимедиа

Демо – фильтрация видео

- Показ видео на 20 минут – примеры разных фильтров обработки видео с указанием, как конкретный фильтр подходит по архитектуре для реализации на GPU

См http://download.graphicon.ru/video/MSU/MSU_video_filters_man.wmv

- Далее для примера – рассказ об одной из технологий фильтрации видео (есть более 300 слайдов с рассказом о разных технологиях, для курса CUDA был выбран Deringing)

Deringing: Содержание

- **Эффект Гиббса и причины возникновения**
- Алгоритмы дерингинга
 - Simple Adaptive Filter
 - Fuzzy Post-Filtering
 - Bilateral Filter
 - Linear Post-Filtering

Понятие Ringing`а

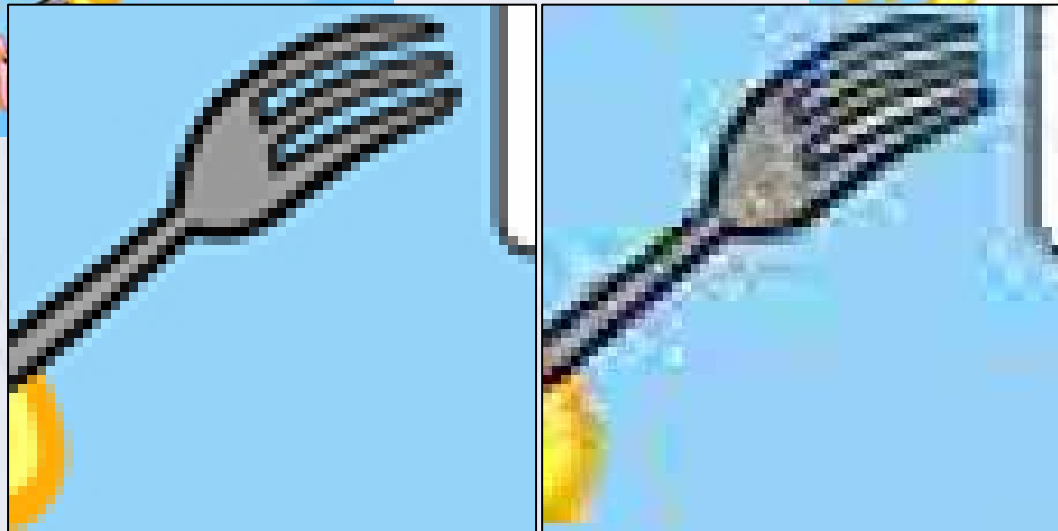
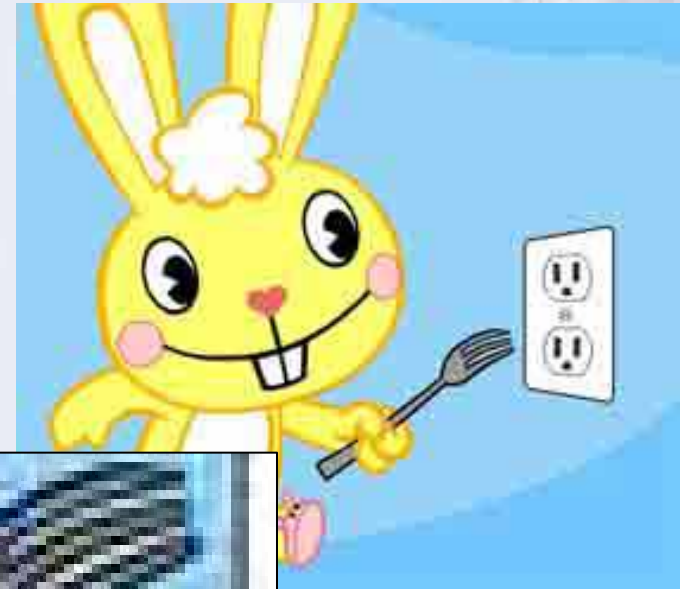
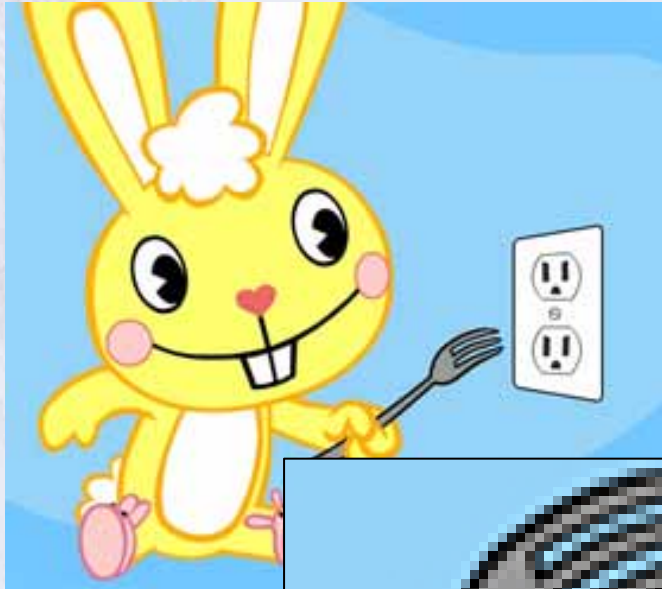
Причины

В алгоритмах сжатия, основанных на DCT, в результате квантования происходит потеря информации.

При квантовании низкочастотных коэффициентов появляется эффект блочности, а при квантовании высокочастотных – эффект Гиббса (ringing)

Понятие Ringing`а

Примеры



Зачем?

- Современные кодеки ориентированы на сжатие фотореалистичного видео
- Существует большое количество анимированного видео, сжатого такими кодеками

Понятие Ringing`а

Возможные стратегии дерингинга

- Фильтрация
- Синтез
- Векторизация

Deringing: Содержание

- Эффект Гиббса и причины возникновения
- Алгоритмы дерингинга
 - **Simple Adaptive Filter**
 - Fuzzy Post-Filtering
 - Bilateral Filter
 - Linear Post-Filtering

Simple Adaptive Filter

Описание алгоритма



- Яркость каждого пикселя сравнивается с соседними
- Если различие не превышает некий порог, происходит усреднение цвета

Simple Adaptive Filter

Реализация

$$\tilde{g}_5 = (\lambda g_5 + \sum_{\substack{i=1 \\ i \neq 5}}^9 \delta_i g_i) / (\lambda + \sum_{\substack{i=1 \\ i \neq 5}}^9 \delta_i)$$

g_1	g_2	g_3
g_4	g_5	g_6
g_7	g_8	g_9

$$\delta_i = \begin{cases} 1, & \text{if } |g_5 - g_i| < S \\ 0, & \text{else} \end{cases}$$

- δ – Весовая функция
- λ – Сила фильтрации
- S – Порог

Simple Adaptive Filter

Пример работы



Simple Adaptive Filter

- Достоинства
 - Высокая скорость работы
- Недостатки
 - Не может полностью удалить артефакты
 - Не учитывает характер области фильтрации

Deringing: Содержание

- Эффект Гиббса и причины возникновения
- Алгоритмы дерингинга
 - Simple Adaptive Filter
 - **Fuzzy Post-Filtering**
 - Bilateral Filter
 - Linear Post-Filtering

Fuzzy Post Filtering

Схема алгоритма



Fuzzy Post-Filtering

Классификация блоков

- Для каждого пикселя определяется его класс

$$pixel(x, y) = \begin{cases} flat & \text{if } \sigma^2(x, y) \leq flatThreshold \\ edge & \text{if } \sigma^2(x, y) > edgeThreshold \\ texture & \text{else} \end{cases}$$

где $\sigma^2(x, y)$ - дисперсия пикселя с центром в (x,y)

Fuzzy Post-Filtering

Классификация блоков



Fuzzy Post Filtering

Фильтрация блоков



- Класс блока определяется как максимальный класс принадлежащего ему пикселя ($\text{flat} < \text{texture} < \text{edge}$)
- Фильтруются только flat- и edge-блоки
- Фильтрация flat-блоков необходима для удаления шумов на сплошных участках
- Фильтрация edge-блоков исправляет ошибки квантования

Fuzzy Post Filtering Фильтрация

Используется фильтр 5x5

$$\tilde{x}_c = \frac{\sum_{j=1}^N x_j \mu(x_c, x_j)}{\sum_{j=1}^N \mu(x_c, x_j)}$$

$$\mu(a, b) = e^{-(a-b)^2 / 2\xi^2}$$

ξ^2 – сила фильтра,

зависит от типа блока

\tilde{x}_c – *новое значение пикселя*

Fuzzy Post Filtering

Пример работы

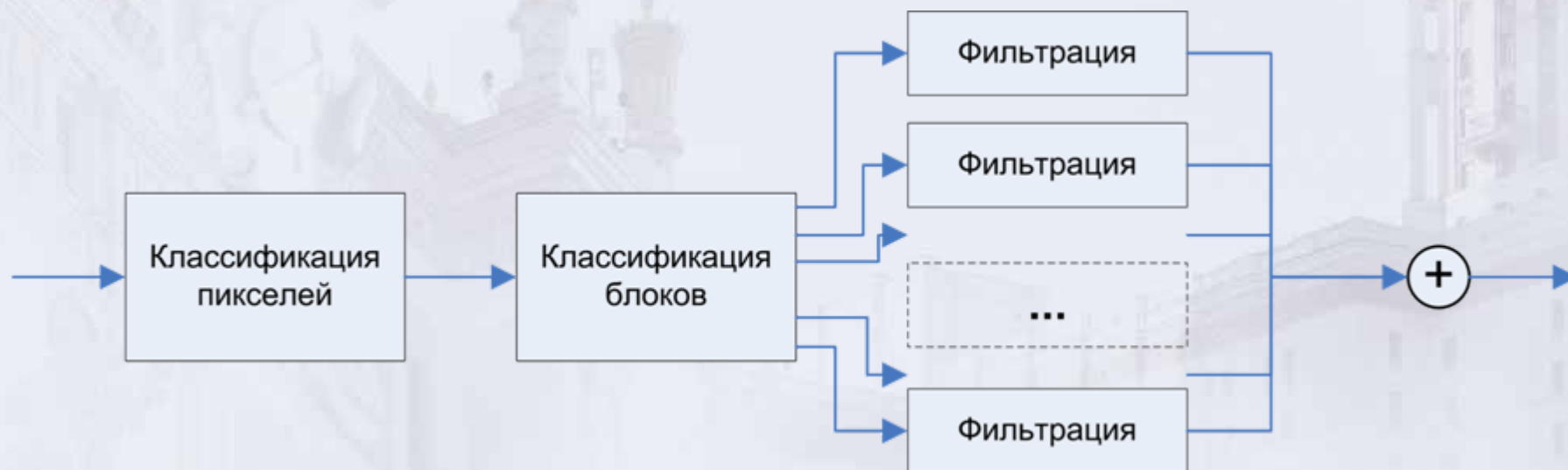


Fuzzy Post Filtering

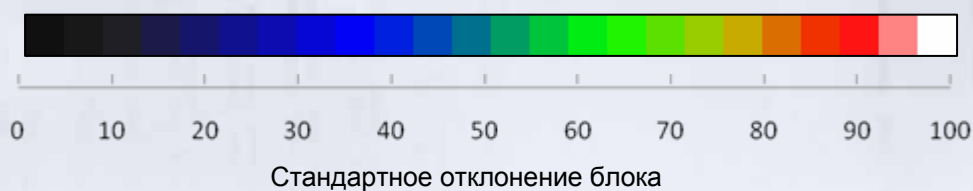
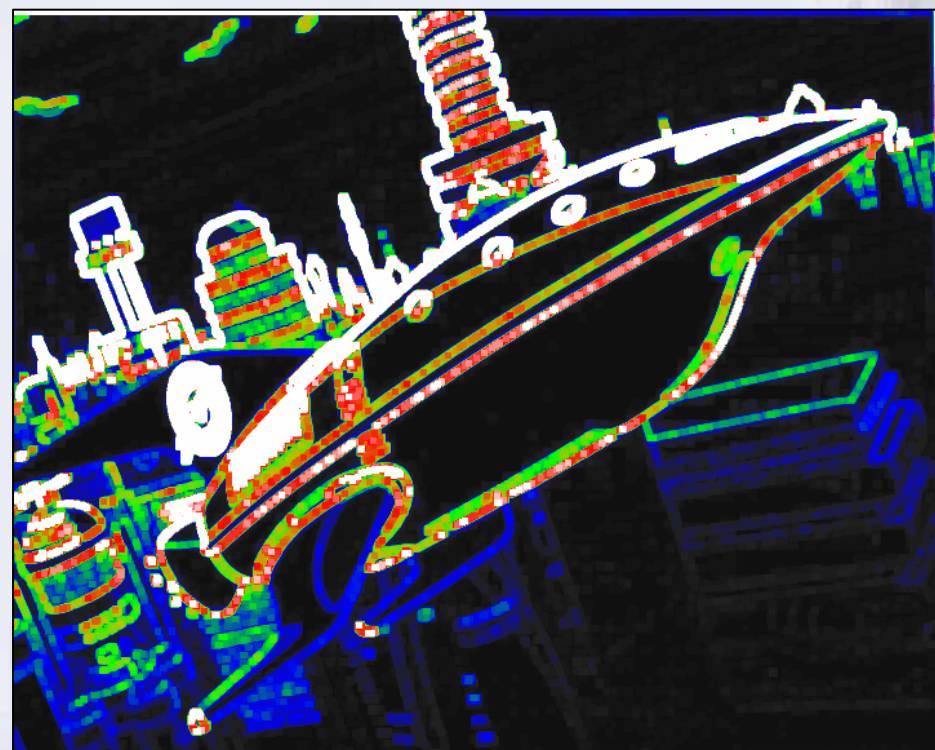
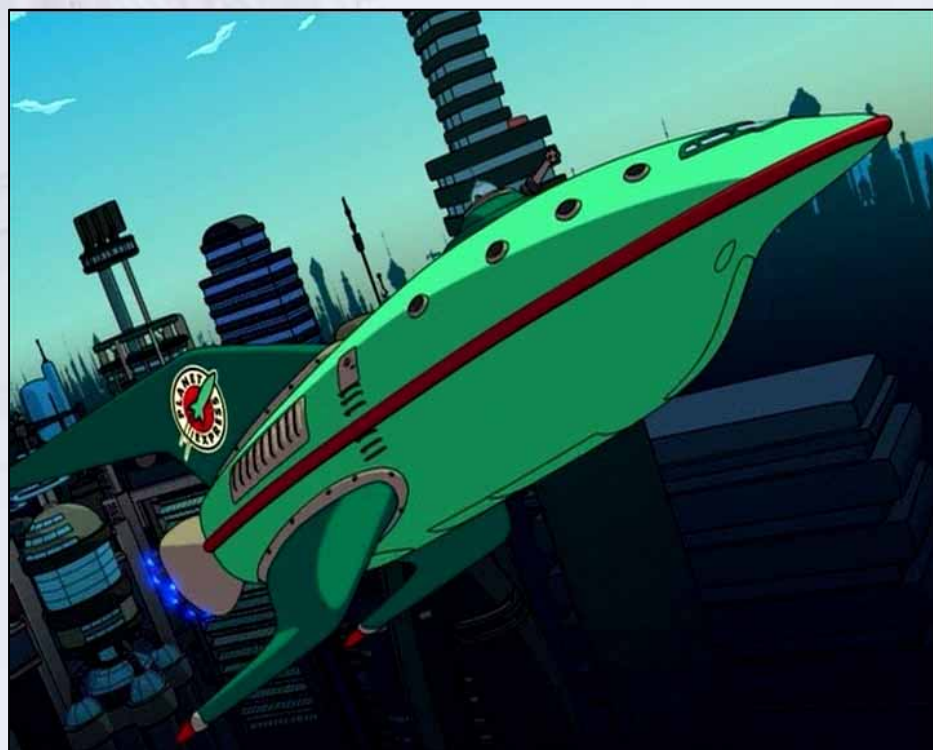
- Достоинства
 - Высокая скорость работы
 - Исправляет многие артефакты
- Недостатки
 - Может размывать картину
 - Необходим точный подбор параметров

Fuzzy Adaptive Filter

- Непрерывная зависимость силы фильтрации от дисперсии блока

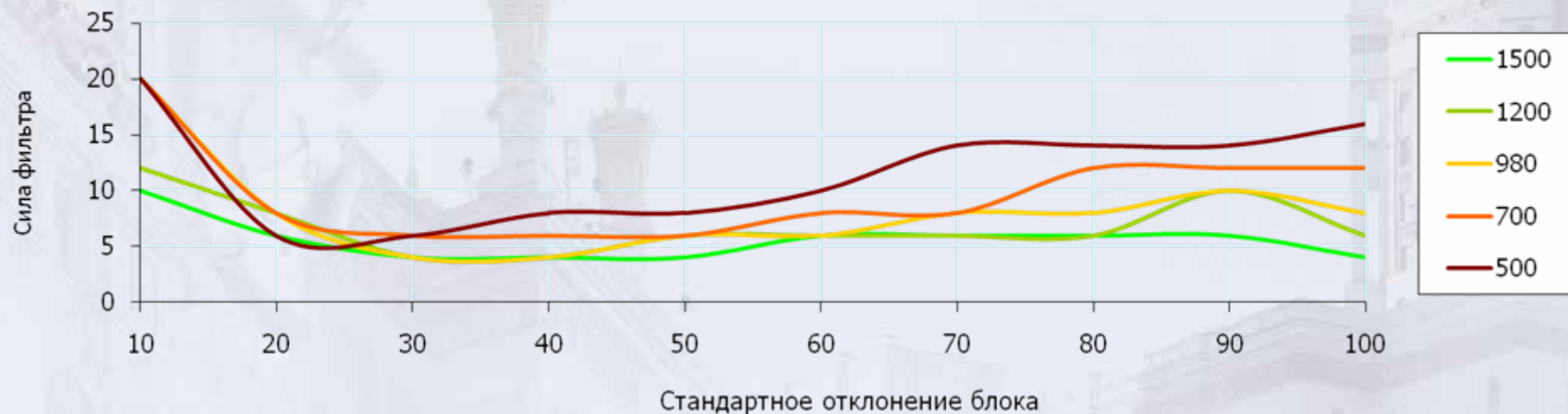


Fuzzy Adaptive Filter

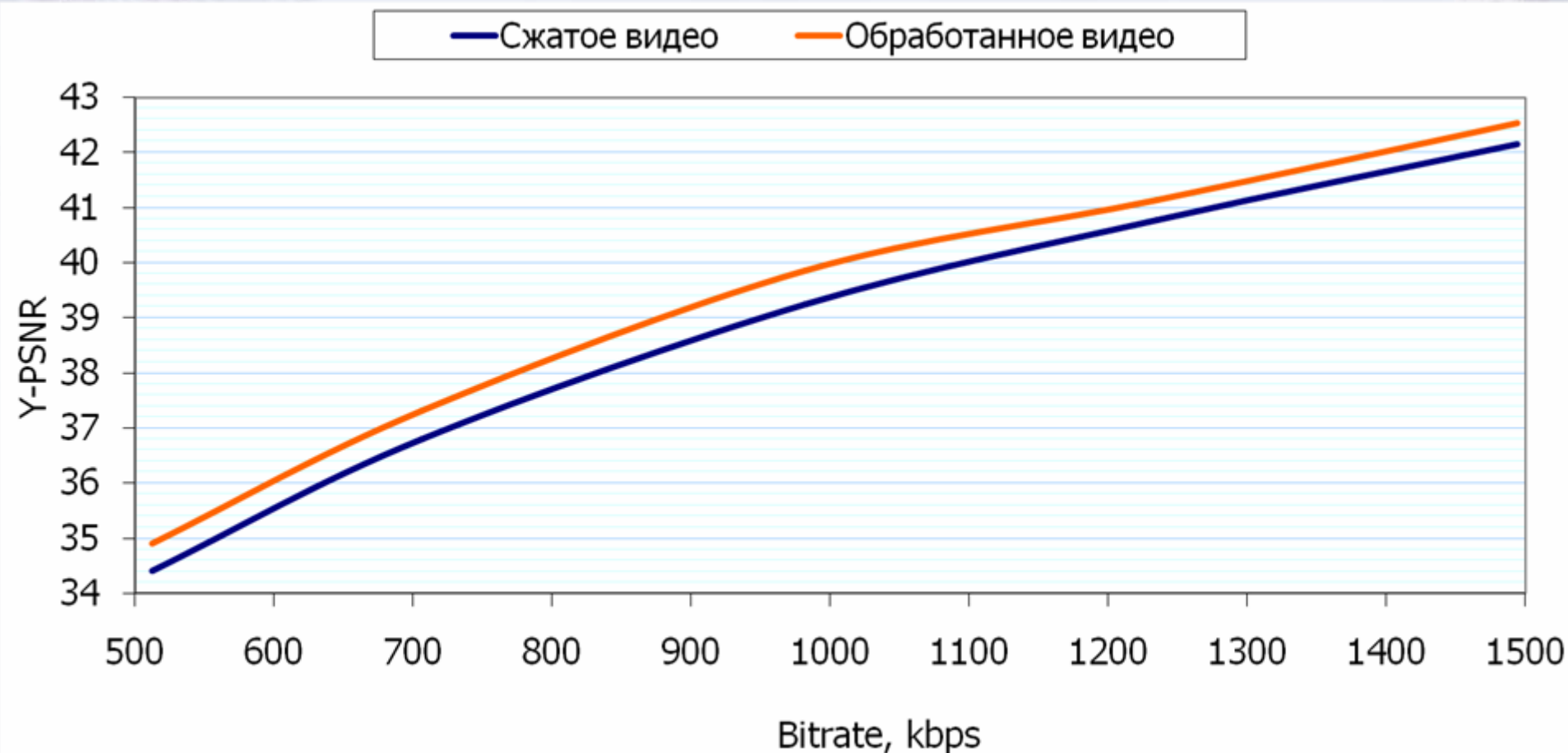


Fuzzy Adaptive Filter

Зависимость силы фильтрации от дисперсии блока при наилучшем качестве (PSNR)



Fuzzy Adaptive Filter



Fuzzy Adaptive Filter

- Достоинства
 - Высокая скорость работы
 - Учитывается характер блока
- Недостатки
 - Необходим точный подбор параметров

Содержание

- Эффект Гиббса и причины возникновения
- Алгоритмы дерингинга
 - Simple Adaptive Filter
 - Fuzzy Post-Filtering
 - **Bilateral Filter**
 - Linear Post-Filtering
 - Библиотека Артефактов
- Результаты

Bilateral Filter

- Идея:
 - Использовать фильтр, аналогичный Fuzzy Post Filtering
 - Учитывать не только фотометрическое расстояние, но и геометрическое между пикселями

Bilateral Filter

- Геометрическая составляющая:

$$h(x) = k_d^{-1}(x) \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\xi) c(\xi, x) d\xi, \quad k_d(x) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} c(\xi, x) d\xi$$

- Фотометрическая составляющая:

$$h(x) = k_r^{-1}(x) \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\xi) s(f(\xi), f(x)) d\xi$$

$$k_r(x) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} s(f(\xi), f(x)) d\xi$$

f – исходное изображение

$c(\xi, x)$ – геометрическое близость точек ξ и x

$s(f(\xi), f(x))$ – фотометрическая близость точек ξ и x

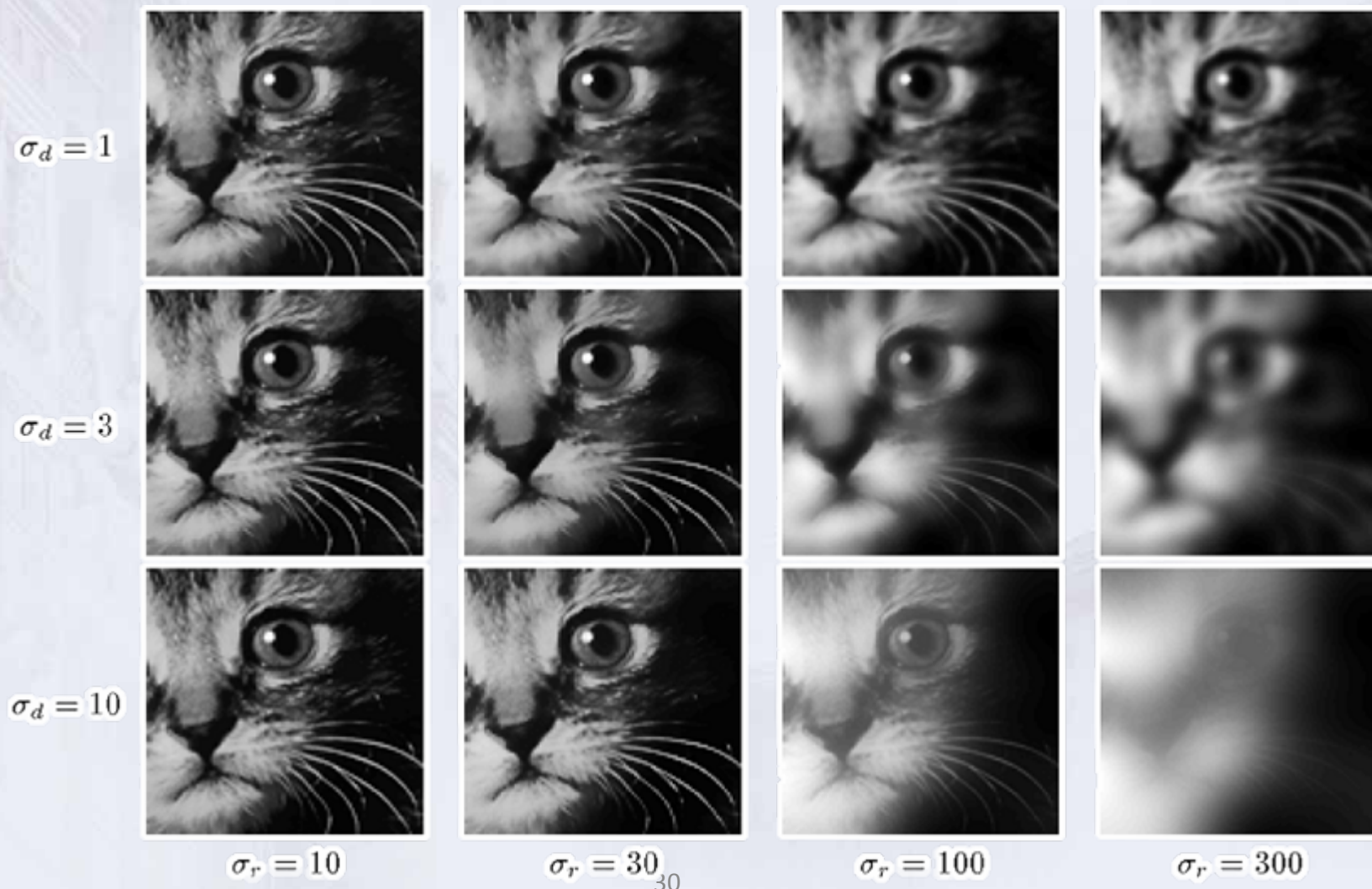
Bilateral Filter

$$h(x) = k^{-1}(x) \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\xi) c(\xi, x) s(f(\xi), f(x)) d\xi$$

$$c(\xi, x) = e^{-\frac{1}{2} \left(\frac{\|\xi - x\|}{\sigma_d} \right)^2}$$

$$s(\xi, x) = e^{-\frac{1}{2} \left(\frac{\|f(\xi) - f(x)\|}{\sigma_r} \right)^2}$$

Bilateral Filter



Multipass Bilateral Filter

Несколько проходов фильтра приводят к "картунизации" изображения



исходное изображение

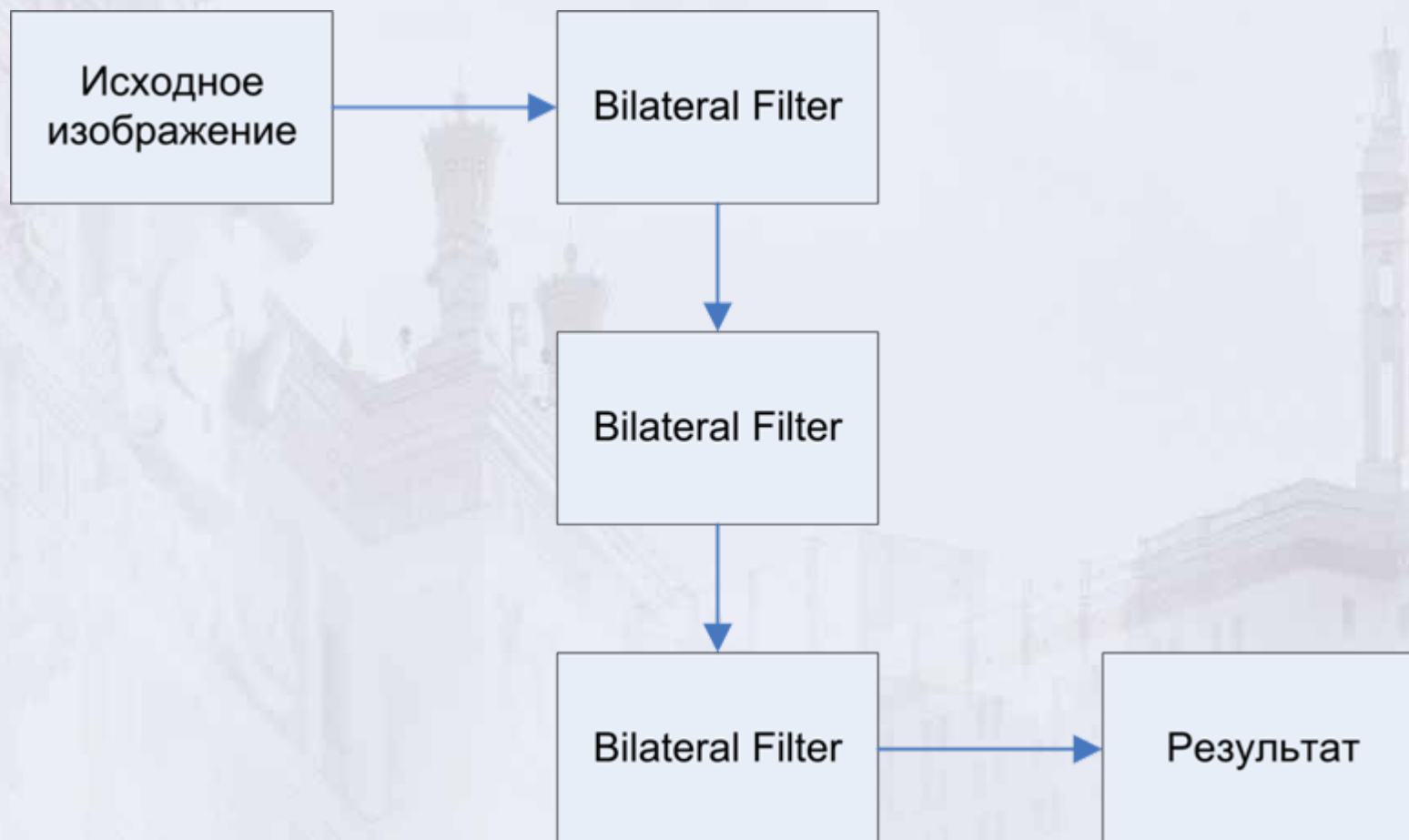


одна итерация



пять итераций

Multipass Bilateral Filter

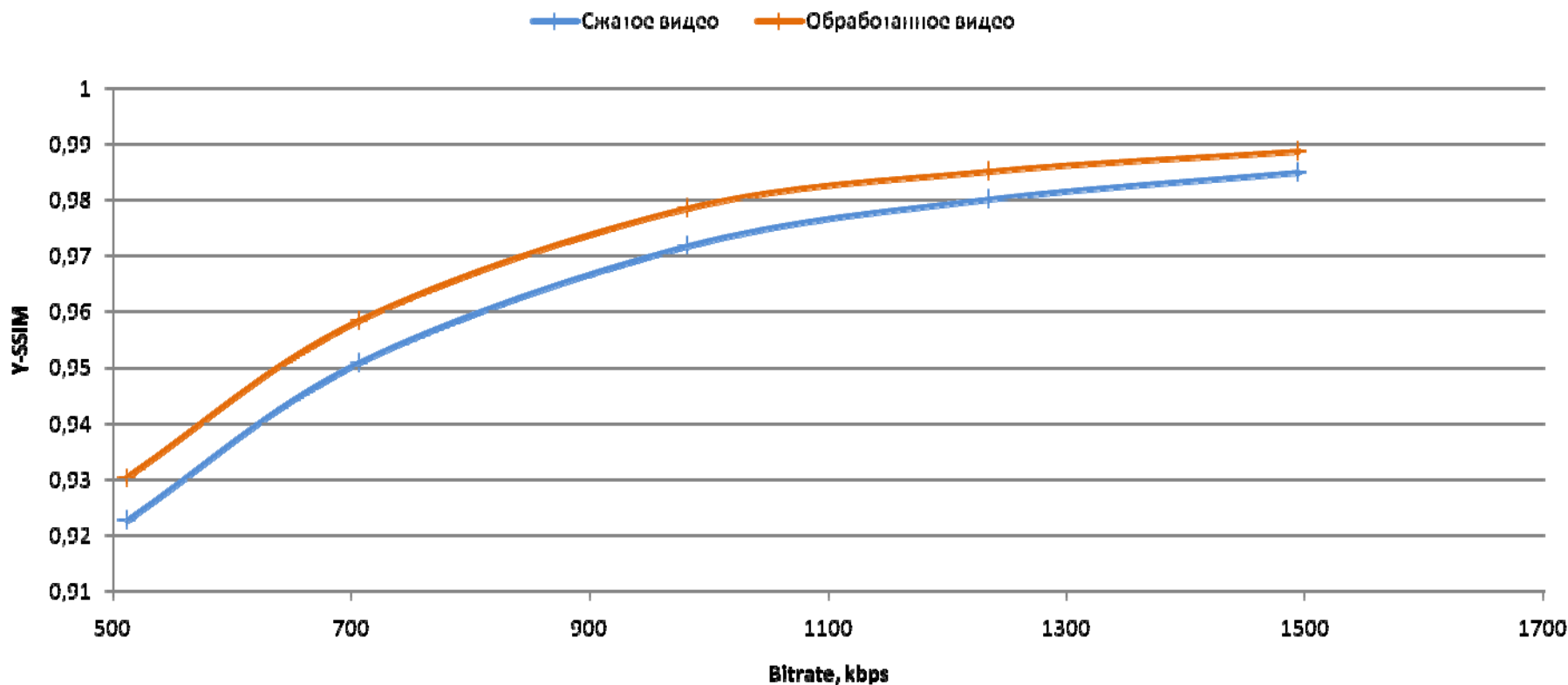


Multipass Bilateral Filter



Результаты

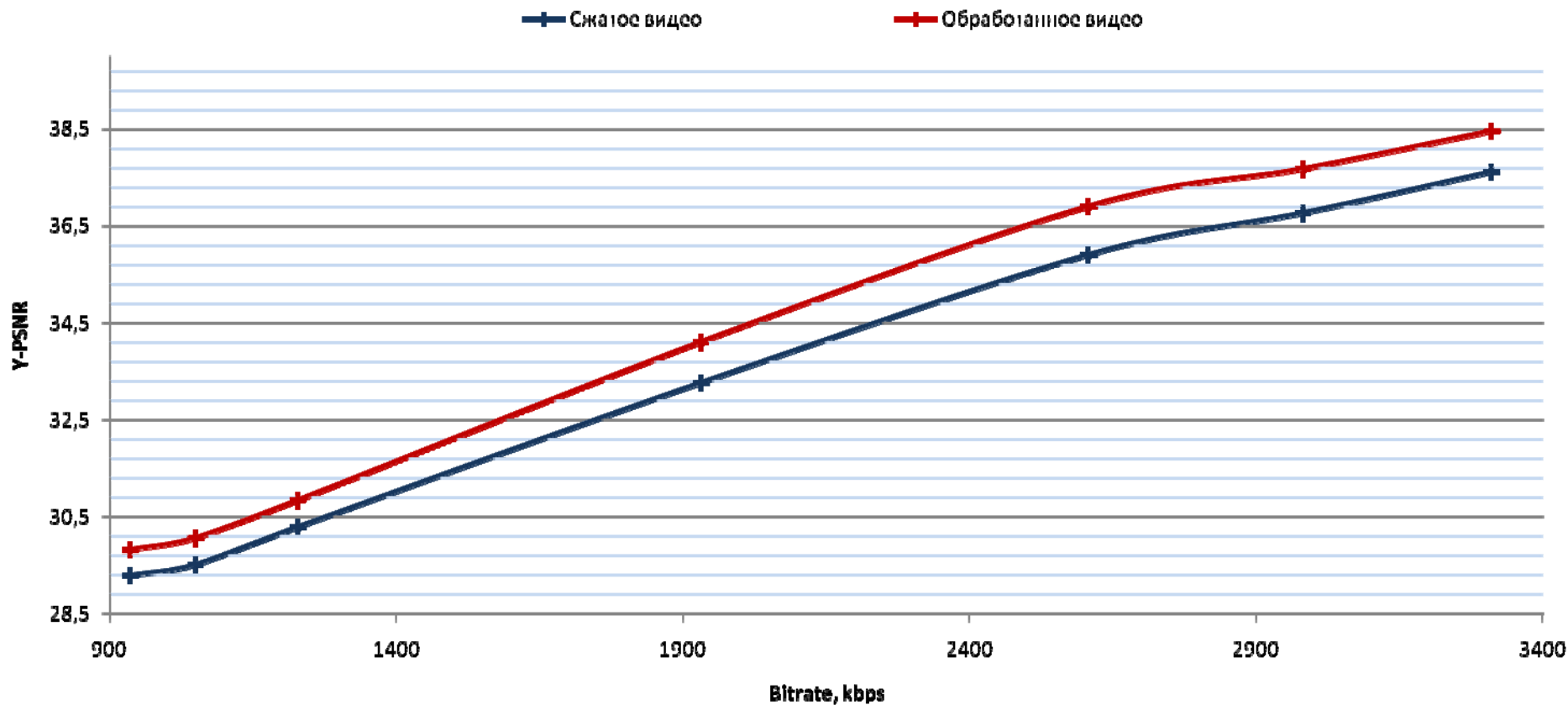
Зависимость Y-SSIM от битрейта



Multipass Bilateral Filter



Результаты



Multipass Bilateral Filter



Пример работы



Multipass Bilateral Filter

- Достоинства
 - Хорошее качество
- Недостатки
 - Эффект алиасинга
 - Закрашивает текстуры
 - Довольно медленный

Deringing: Содержание

- Эффект Гиббса и причины возникновения
- Алгоритмы дерингинга
 - Simple Adaptive Filter
 - Fuzzy Post-Filtering
 - Bilateral Filter
 - **Linear Post-Filtering**

Linear Post-Filtering

Схема алгоритма



Linear Post-Filtering

Классификация блоков

- Intra-блоки

$$S = \sum_{u=0}^7 \sum_{v=0}^7 |F_{u,v}| \quad (u,v) \neq (0,0)$$

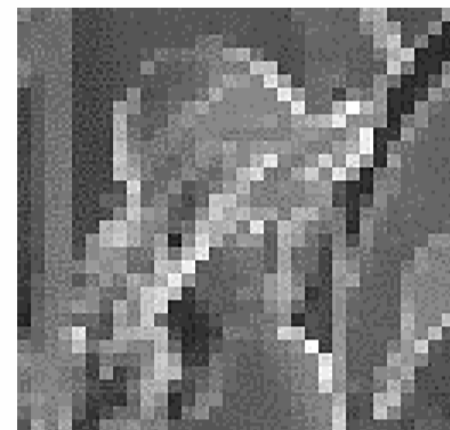
If ($S < \text{Threshold}$)

Block \leftarrow Smooth

else

Block \leftarrow Complex

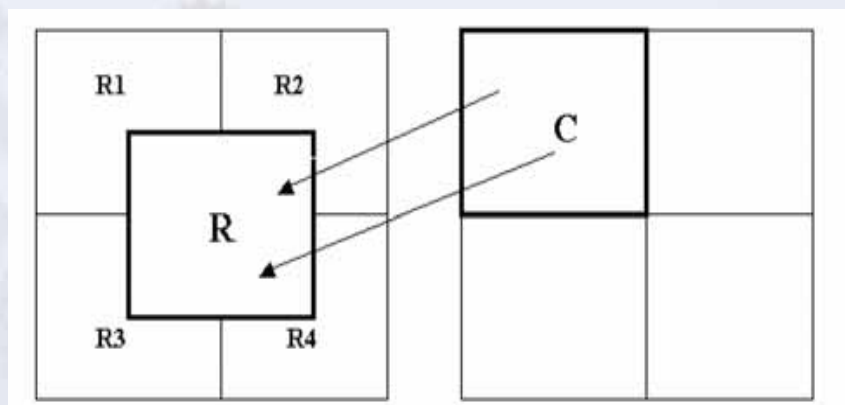
$F_{u,v}$ - коэффициенты BDCT



Linear Post-Filtering

Классификация блоков

- Inter-блоки



If (R1, R2, R3 or R4 is Complex)

 C ← Complex

else

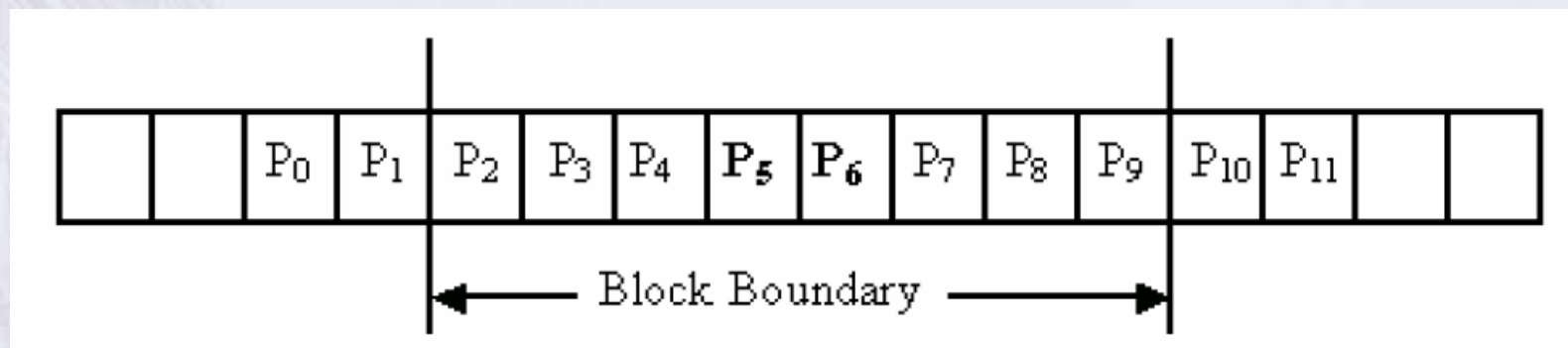
 C ← Smooth

Linear Post-Filtering

Классификация блоков

- Только Complex блоки подлежат фильтрации
- Smooth блоки копируются без изменений

Linear Post-Filtering

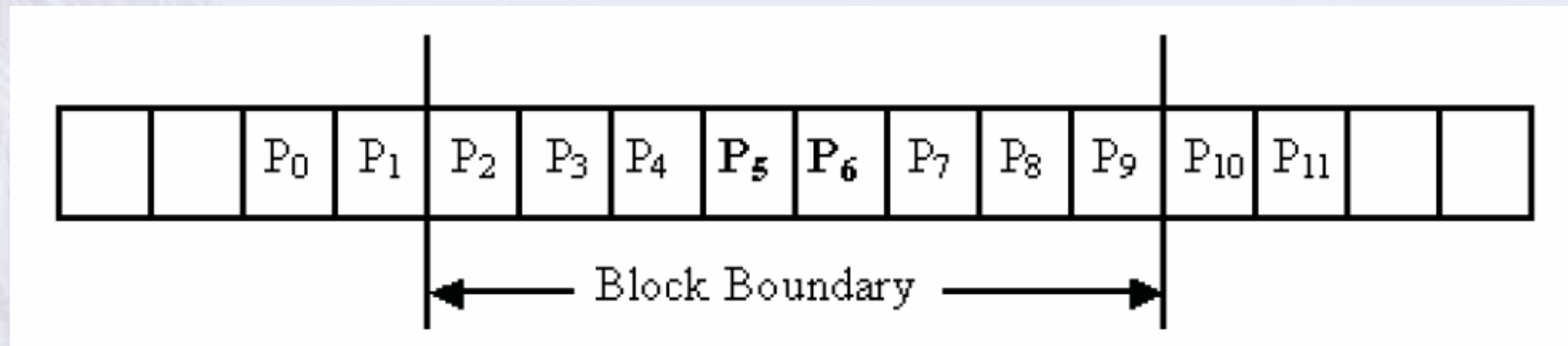


if($|P_{i+1} - P_i| \geq QP$)
 $P_{i+1}, P_i \leftarrow \text{edge pixels}$

Если найдены граничные пиксели, то они копируются без фильтрации.

QP – коэффициент квантования

Linear Post-Filtering



if($|P_1 - P_2| < (0.5QP)$ && P_1 lies in a smooth block)

$$P_2' = (P_0 + P_1 + 2P_2) / 4$$

$$P_3' = (P_1 + P_2' + 2P_3) / 4$$

$$P_4' = (P_2' + P_3' + 2P_4) / 4$$

else

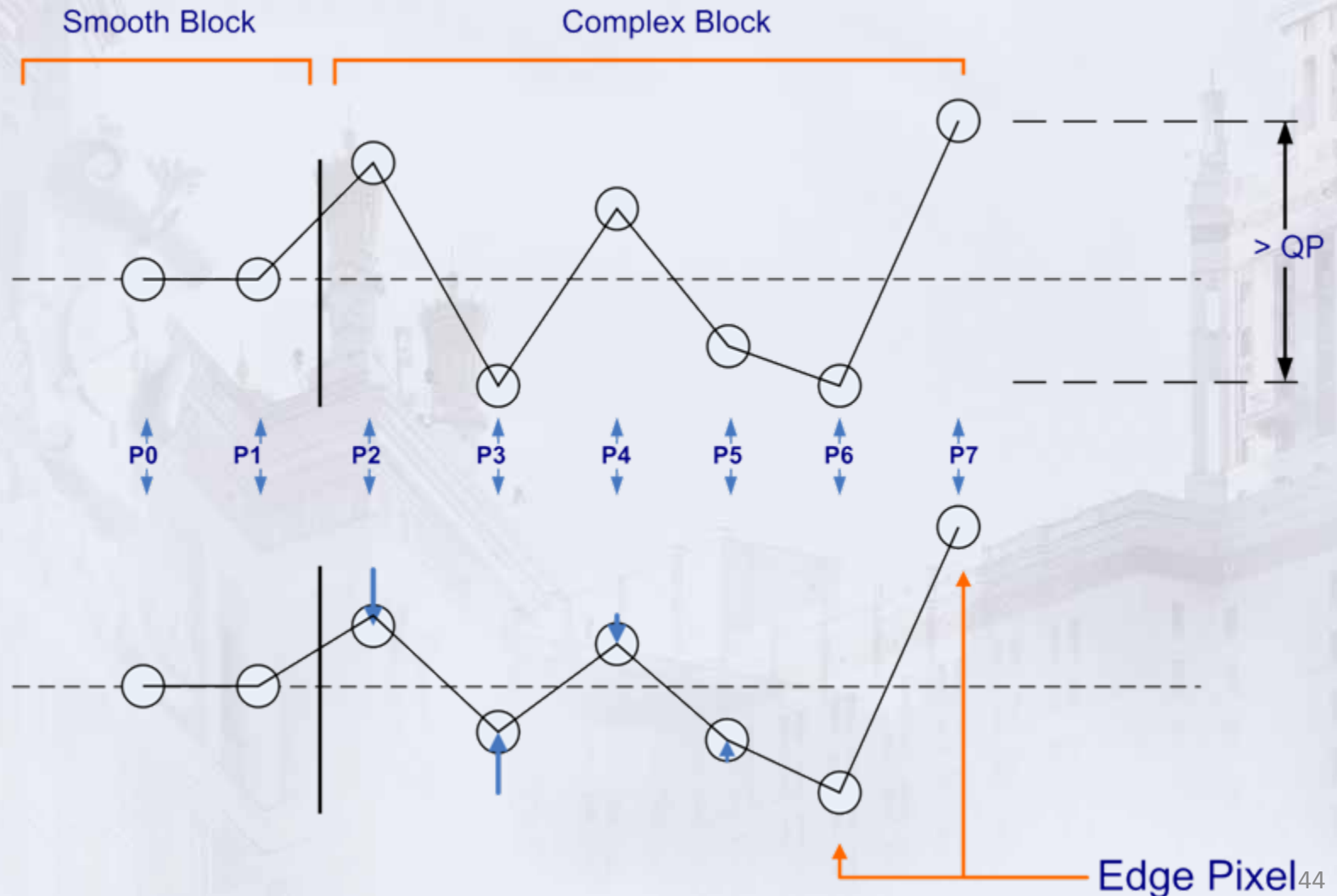
$$P_3' = (P_2 + 2P_3 + P_4) / 4$$

$$P_4' = (P_3' + 2P_4 + P_5) / 4$$

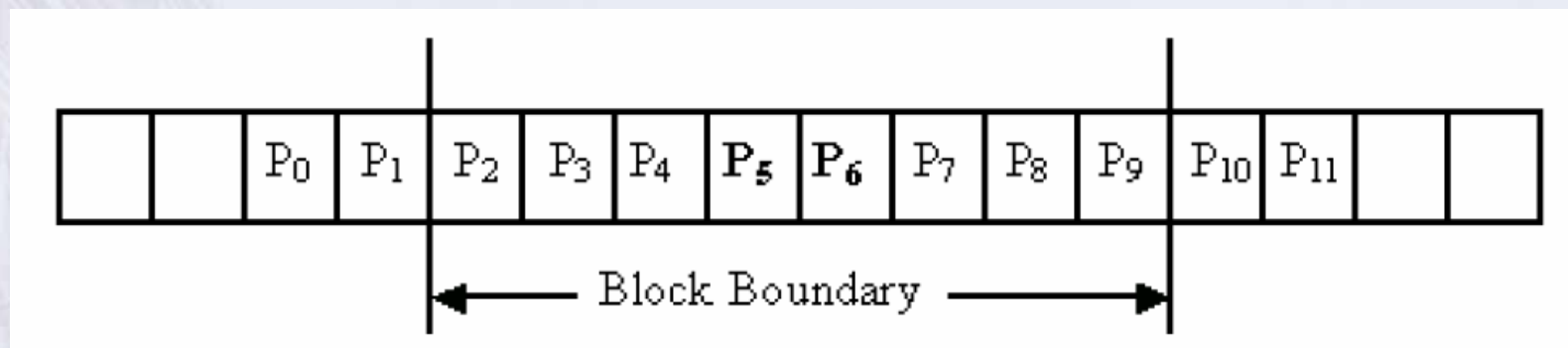
$$\frac{1}{4} \begin{bmatrix} 1 & 1 & 2 & 0 & 0 \end{bmatrix}$$

$$\frac{1}{4} \begin{bmatrix} 1 & 2 & 1 \end{bmatrix}$$

Linear Post-Filtering



Linear Post-Filtering



- Если не найдено ни одного граничного пикселя

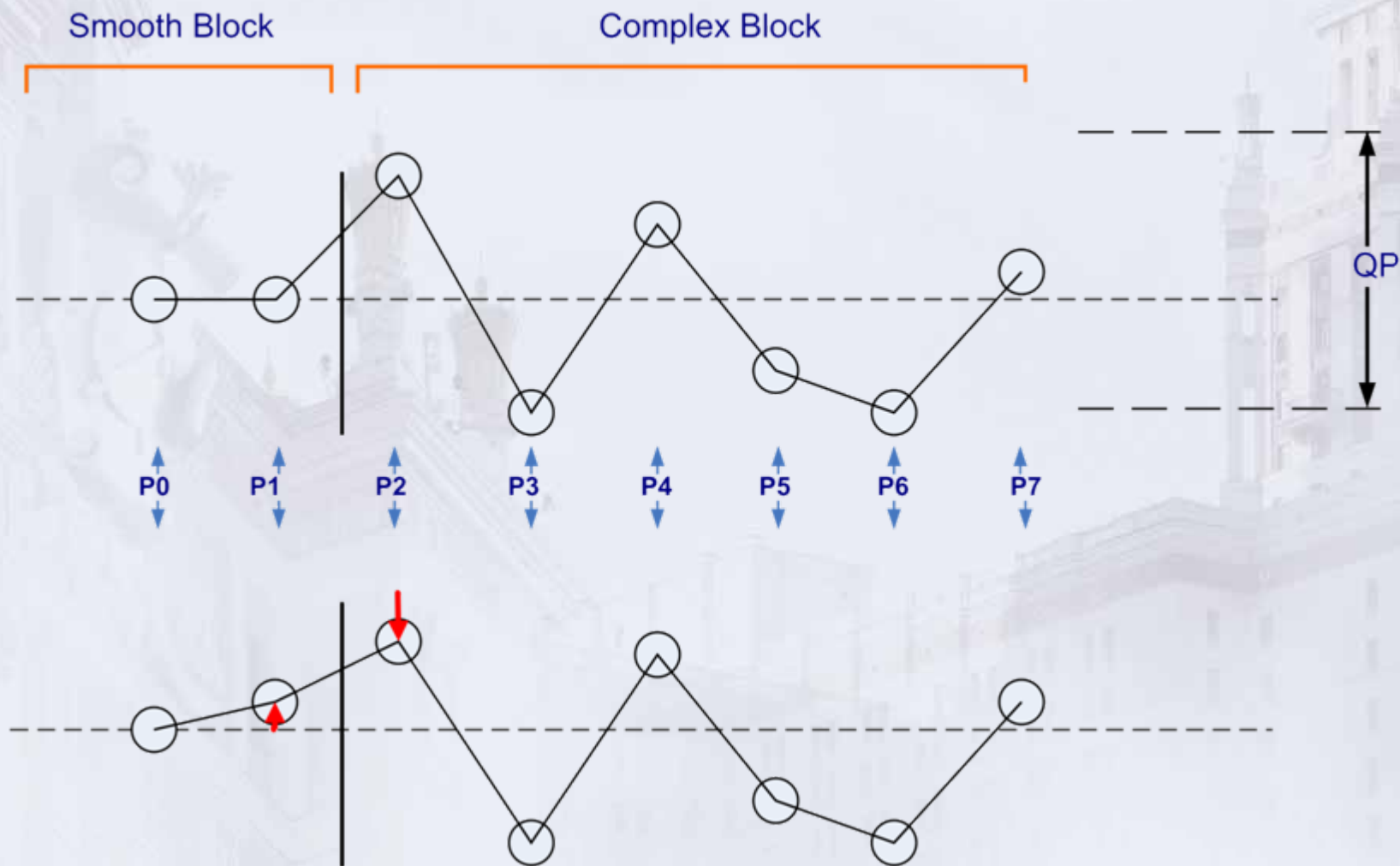
$$d = (P_1 - P_2)$$

$$\text{if } (| d | < 2QP)$$

$$P_1' = P_1 - d/4$$

$$P_2' = P_2 + d/4$$

Linear Post-Filtering



Linear Post-Filtering

Пример



Видеогруппа лаборатории машграфики

- Постоянная работа с крупными компаниями (Intel – 5 лет, Samsung – 3 года, RealNetworks – 6 лет и многими другими) в области фильтрации видео
- Наши метрики измерения качества видео используются в Intel, NVIDIA, Apple, Dolby, Cisco, Skype, Sun, NASA и т.д.
- В компаниях весьма известны наши сравнения видеокодеков (сравните с доступными):
http://en.wikipedia.org/wiki/Comparison_of_video_codecs#Freely_available_codecs_comparisons
- Во многих проектах используется GPU (включая Fermi)

Для работ в проектах, заинтересованы в студентах, знающих CUDA (см. http://groups.google.com/group/video_processing_course)