



# Оценка уровня шума

Борис Кумок

*Video Group  
CS MSU Graphics & Media Lab*

# Содержание доклада

- ◆ Введение
- ◆ Wavelet
  - MAD
  - MBE
  - Trained moments
  - CDF
- ◆ Block-based
  - Classic
  - Structure-oriented
  - Temporal SO
- ◆ Ссылки

# Введение



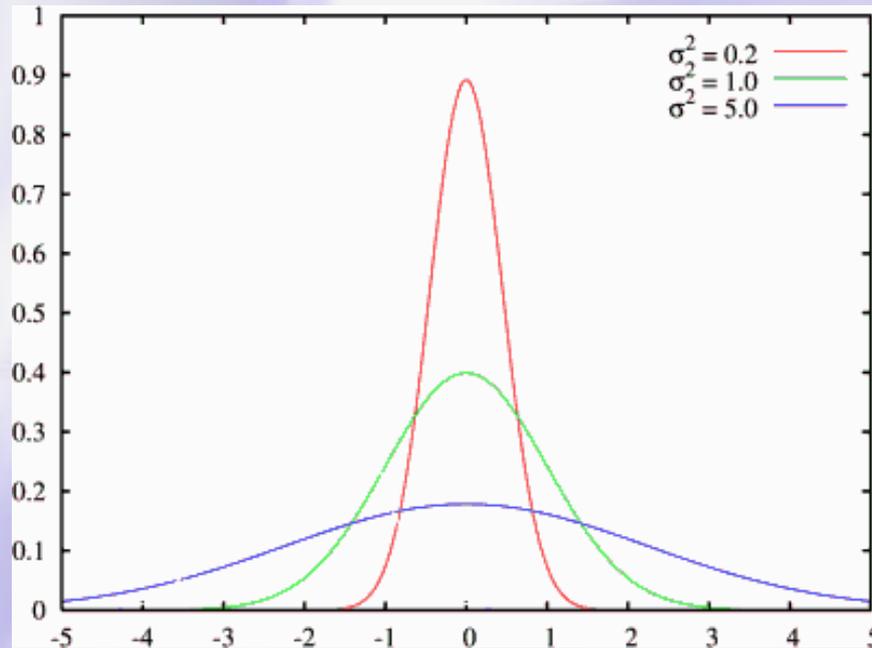
Определение уровня шума требуется для:

- Подавление шума
- Сжатие видео
- Сегментация
- Определение границ

Для большинства приложений критична точность оценки, а для видео еще важна скорость.

# Постановка задачи

По входному изображению определить уровень шума.  
Полагая, что шум является белым с нормальным распределением (AWGN), сведем задачу к определению  $\sigma$ .



$$p(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{x^2}{2\sigma^2}}$$

# Содержание доклада

- ◆ Введение
- ◆ Wavelet
  - MAD
  - MBE
  - Trained moments
  - CDF
  - Сравнение
- ◆ Block-based
- ◆ Ссылки

# MAD



Изображение



MAD блоки

$$\text{MAD} = \text{median}(|x_i - \text{median}(x_i)|)$$

Для всех блоков изображения посчитаем MAD и положим  $\sigma$  пропорциональной их медиане.

$$\sigma = 1.483 * \text{median}(\text{MAD}(i))$$

# MAD



Т.к. необходимо вычислять медиану, метод работает медленно. Кроме того, он требует дополнительную память.

Один из самых часто применяемых методов.

Существует несколько модификаций для работы в разных пространствах.

# Содержание доклада

- ◆ Введение
- ◆ Wavelet
  - MAD
  - MBE
    - ◆ Идея
    - ◆ Построение модели
    - ◆ Метод максимального правдоподобия
    - ◆ Метод соответствия моментов
  - Trained moments
  - CDF
  - Сравнение
- ◆ Block-based
- ◆ Ссылки

# МВЕ

Идея метода



Построим модель зашумленного изображения.  
Используем ее для оценки уровня шума.

- ◆ Построение модели
- ◆ Оценка шума
  - Метод максимального правдоподобия
  - Метод моментов

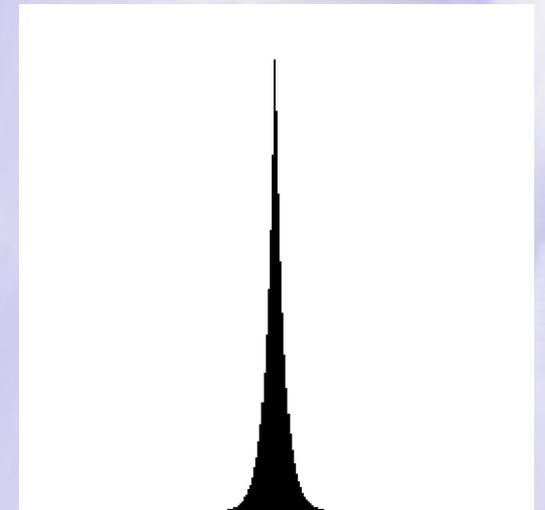
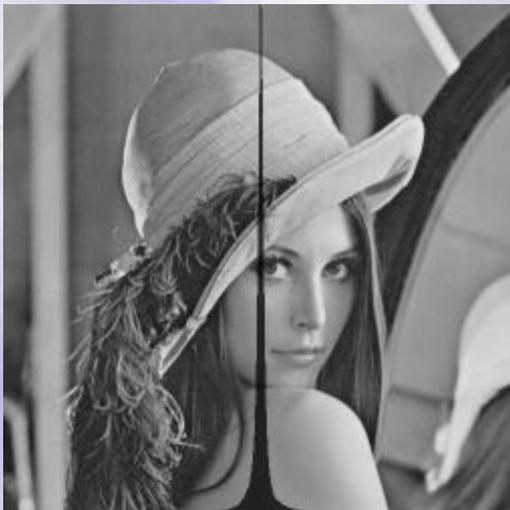
# МВЕ

## Построение модели

Гистограмма вейвлет коэффициентов чистого изображения похожа на лапласиан.

Предполагаемое распределение

$$p_{image}(x) = \frac{1}{\sigma_u \sqrt{2}} e^{-(\sqrt{2}/\sigma_u)|x|}$$



# МВЕ

## Построение модели

Построим модель зашумленного изображения.

Модель распределения вейвлет коэффициентов чистого изображения

$$p_{image}(x) = \frac{1}{\sigma_u \sqrt{2}} e^{-(\sqrt{2}/\sigma_u)|x|}$$

Плотность шума

$$p_{noise}(x) = \frac{1}{\sigma_v \sqrt{2\pi}} e^{-x^2/(2\sigma_v^2)}$$

$$p_{\xi_1+\xi_2}(y) = \int_{-\infty}^{\infty} p_{\xi_1}(u) \cdot p_{\xi_2}(y-u) du$$

Модель распределения коэффициентов зашумленного изображения

$$P_{image+noise}(x) = \frac{e^{(\sigma_v/\sigma_u)^2}}{\sigma_v \sqrt{2}} \left( e^{(\sqrt{2}/\sigma_u)x} \operatorname{erfc}\left(\frac{\sigma_v}{\sigma_u} + \frac{x}{\sigma_v \sqrt{2}}\right) + e^{-(\sqrt{2}/\sigma_u)x} \operatorname{erfc}\left(\frac{\sigma_v}{\sigma_u} - \frac{x}{\sigma_v \sqrt{2}}\right) \right)$$

# МВЕ

Метод максимального правдоподобия

Оптимальное решение можно получить методом максимального правдоподобия.

Функция правдоподобия

$$L(x, \sigma) = \prod P_{image+noise}(x_i, \sigma)$$

$$\sigma(x) = \arg \max_{\sigma} L(x, \sigma)$$

Аналитического решения данная задача не имеет. Численное решение – очень медленно.

# МВЕ

## Метод моментов

$$P_{image+noise}(x)$$

$$E(x^2) = m_2 = \sigma_u^2 + \sigma_v^2$$

$$E(x^4) = m_4 = 6\sigma_u^2 + 3\sigma_v^2 + 6\sigma_u^2\sigma_v^2$$

Не оптимальный, но более быстрый метод моментов позволяет получить результат за один проход.

Решим систему и заменим теоретические моменты на выборочные.

$$\hat{m}_k = \frac{1}{N} \sum x(i)^k$$

$$\sigma_v^2 = \hat{m}_2 \left( 1 - \sqrt{\frac{1}{3} \frac{\hat{m}_4}{\hat{m}_2^2} - 1} \right)$$

# Содержание доклада

- ◆ Введение
- ◆ Wavelet
  - MAD
  - MBE
  - Trained moments
    - ◆ Описание
    - ◆ Обучение
  - CDF
  - Сравнение
- ◆ Block-based
- ◆ Ссылки

# Обученные моменты

Развитием предыдущего метода может быть учет иных статистик кроме 2 и 4 моментов.

$$\sigma = \alpha_1 M_1 + \alpha_2 M_2 + \alpha_4 M_4$$

$$M_1 = \hat{m}_1$$

$$M_2 = \hat{m}_2 / \hat{m}_1$$

$$M_4 = \hat{m}_4 / \hat{m}_2 \hat{m}_1$$

$$\hat{m}_k = \frac{1}{N} \sum x(i)^k$$

Выбор подходящих коэффициентов  $\alpha_k$  осуществим путем обучения алгоритма.

# Обучение алгоритма

## Метод наименьших квадратов

$$\sigma(x) = \sum \alpha_k M_k(x)$$

Для нахождения хороших коэффициентов алгоритм необходимо предварительно обучить.

Возьмем набор  $n$  изображений  $x_i$  с известным уровнем шума  $y_i$  и применим метод наименьших квадратов.

# Обучение алгоритма

## Метод наименьших квадратов

$$F(x) = \sum_{k=0}^m \alpha_k \varphi_k(x)$$

$$J = \sum_{i=0}^n (y_i - F(x_i))^2$$

Найдем коэффициенты  $\alpha_k$ , при которых суммарная квадратичная погрешность  $J$  минимальна.

Получим СЛАУ с  $m$  неизвестными, которую можно решить, например, методом Гаусса.

$$\sum_{k=0}^m \gamma_{lk} \alpha_k = b_l, \quad l = 0, 1, \dots, m,$$

$$\gamma_{lk} = \sum_{i=0}^n \varphi_l(x_i) \varphi_k(x_i),$$

$$b_l = \sum_{i=0}^n \varphi_l(x_i) y_i$$

# Содержание доклада

- ◆ Введение
- ◆ Wavelet
  - MAD
  - MBE
  - Trained moments
  - CDF
    - ◆ Введение
    - ◆ Алгоритм
    - ◆ Обучение
  - Сравнение
- ◆ Block-based
- ◆ Ссылки

# CDF

## Введение

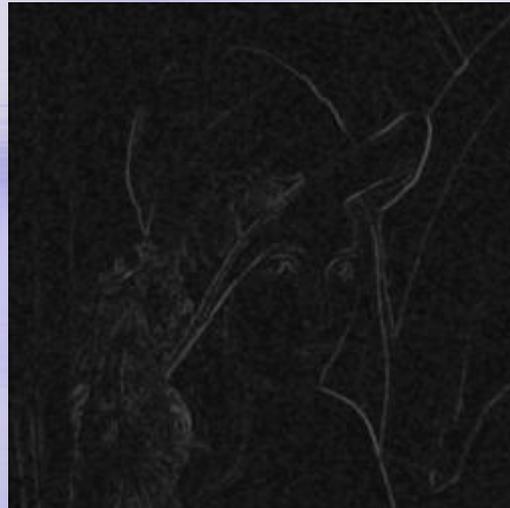
$cdf(x)$  - количество пикселей с дисперсией меньшей  $x$

Будем использовать  $cdf(x)$  для определения уровня шума.

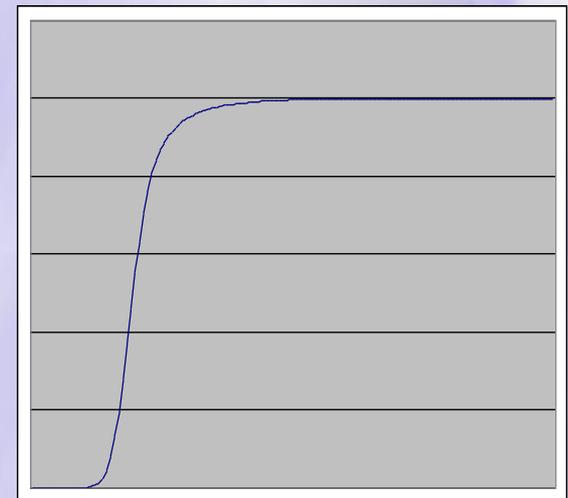
Изображение



Локальная дисперсия

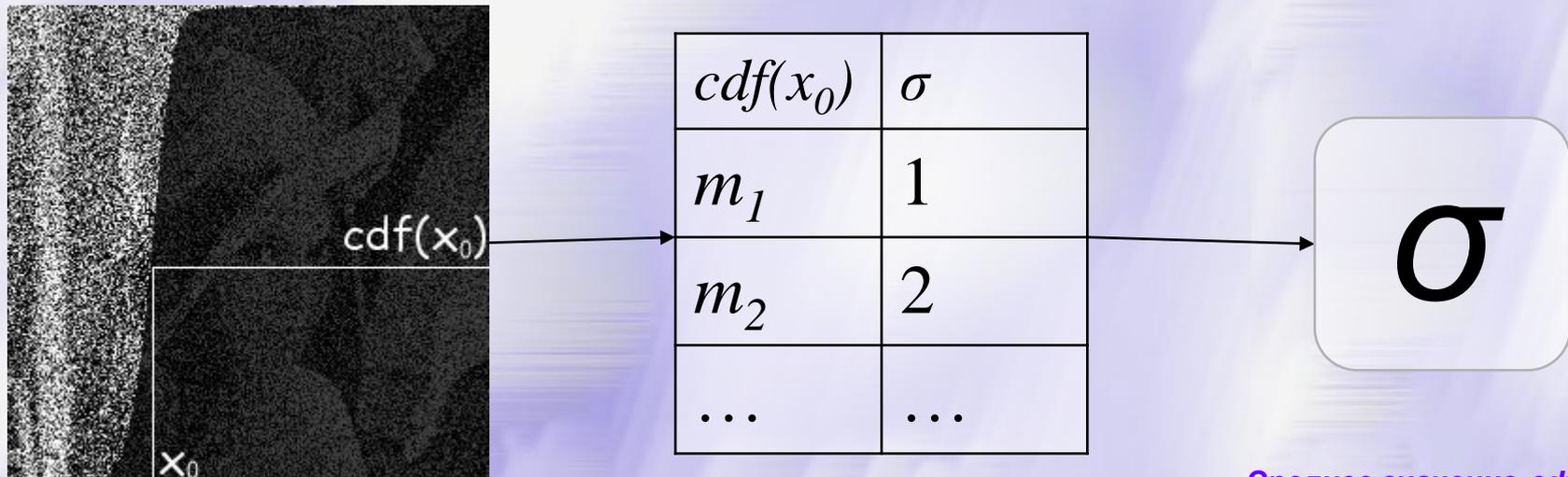


$cdf(x)$



# CDF Алгоритм

Вычислим  $cdf(x_0)$  и по таблице найдем соответствующее значение  $\sigma$ .



Создание таблицы и выбор точки  $x_0$  происходят на этапе обучения алгоритма.

Среднее значение  $cdf(x)$  среди изображений с  $\sigma = l$ .

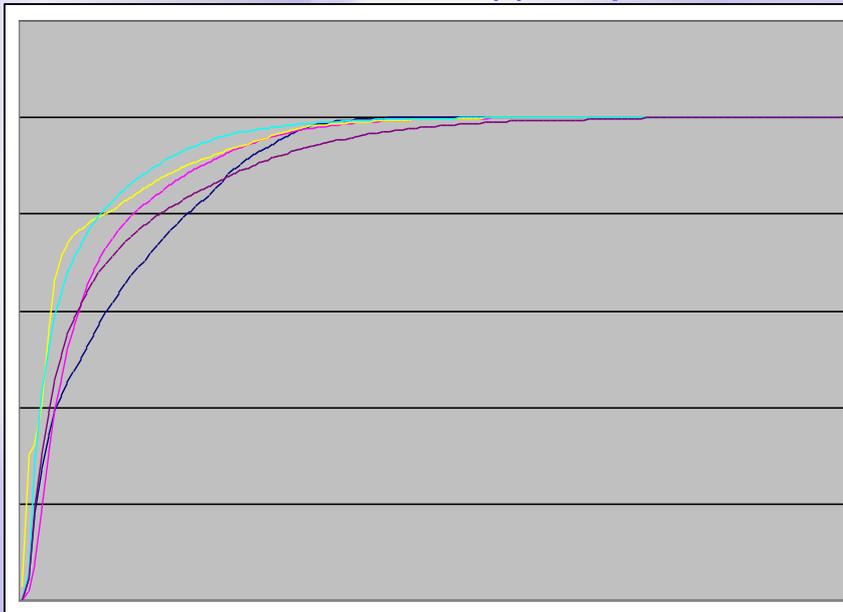
$$m_l(x) = \frac{1}{N} \sum_{k=1}^N cdf_l^k(x)$$

# CDF

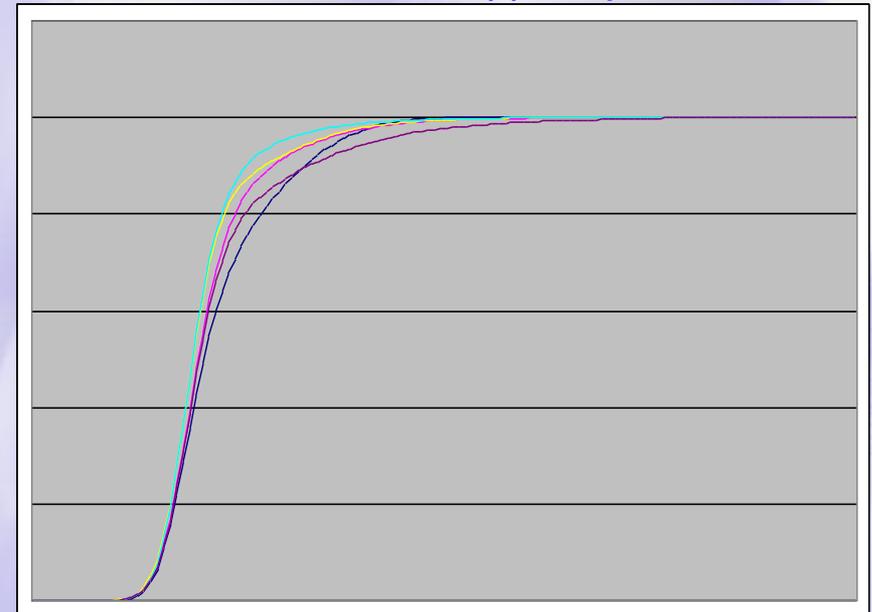
## Обучение

Не все точки графика дают надежную оценку.  
Найдем наилучшую.

$\text{cdf}(x)$  изображений с  $\sigma = 0$



$\text{cdf}(x)$  изображений с  $\sigma = 25$



# CDF

## Обучение

На обучающем наборе изображений с известной  $\sigma$  шума выберем точку  $x_0$ , наиболее надежную для классификации.

Степень различимости  $cdf(x)$   
между уровнями  $l_1$  и  $l_2$ .

$$f_{l_1, l_2}(x) = \frac{|m_{l_1}(x) - m_{l_2}(x)|}{\sqrt{\sigma_{l_1}(x) + \sigma_{l_2}(x)}}$$

$$x_0 = \arg \max_x \sum_{i,j} f_{i,j}(x)$$

Среднее значение  $cdf(x)$   
среди изображений с  $\sigma = l$ .

$$m_l(x) = \frac{1}{N} \sum_{k=1}^N cdf_l^k(x)$$

Дисперсия  $cdf(x)$  среди  
изображений с  $\sigma = l$ .

$$\sigma_l^2(x) = \frac{1}{N} \sum_{k=1}^N (cdf_l^k(x) - m_l(x))^2$$

# Сравнение

TABLE 1: Mean squared errors for noise standard deviation estimates computed over 5 images.

Standard deviation of synthetic noise	MAD	Moment matching	Trained moments	cdf
5	3.49	2.49	2.35	2.73
7	3.01	2.36	1.91	1.86
9	2.63	2.13	1.59	1.55
11	2.33	1.97	1.38	1.40
13	2.08	1.76	1.37	0.46
15	1.84	1.62	1.31	0.43
Overall mean	2.56	2.06	1.65	1.41

Методы основанные на обучении дают более хороший результат.

# Содержание доклада

- ◆ Введение
- ◆ Wavelet
- ◆ Block-based
  - Classic
  - Structure-oriented
  - Temporal SO
- ◆ Ссылки

# Блочный метод

Разобьем изображение на блоки (7x7) и выберем из них  $N$  штук с наименьшей дисперсией. Положим  $\sigma$  равной их усреднению.

$$\sigma^2 = \frac{1}{N} \sum_{k=1}^N \min^k \sigma_{B_k}^2$$

$\sigma_{B_k}^2$  – дисперсия  $k$ -ого блока

$N$  – фиксированная часть общего числа блоков.

# Блочный метод

Метод является одним из самых быстрых и требует мало дополнительной памяти.

Для оценки уровня шума используются «гладкие» области. Если таких областей нет, точность резко падает.



Дисперсия

# Содержание доклада

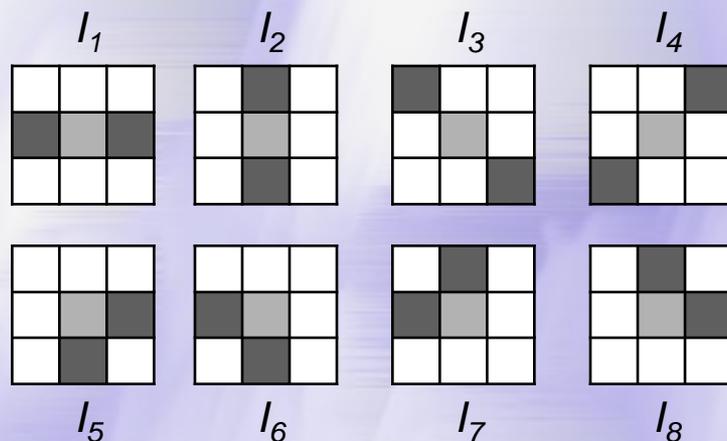
- ◆ Введение
- ◆ Wavelet
- ◆ Block-based
  - Classic
  - Structure-oriented
  - Temporal SO
- ◆ Ссылки

# Структурный

Дисперсия – не лучший способ определить гладкость блока.

Воспользуемся детектором границ.

Свертка  $\{-1 -1 \dots (w-1) \dots -1 -1\}$



$$I_k(i) = -I(i-1) + 2I(i) - I(i+1)$$

Мера гладкости блока.  
Сумма сверток по всем направлениям для каждого пикселя блока.

$$\xi_B = \sum_i \sum_{k=1}^8 |I_k(i)|$$

# Структурный

$B_h$  –  $h$ -ый гладкий блок

$$\sigma^2 = \frac{1}{N} \sum_{h=1}^N \sigma_{B_h}^2$$

Усредним дисперсию  $N$   
наиболее гладких блоков.

Дополнительно наложим ограничение на  
слагаемые.

$$|\sigma_{B_h}^2 - \sigma_{B_R}^2| < T$$

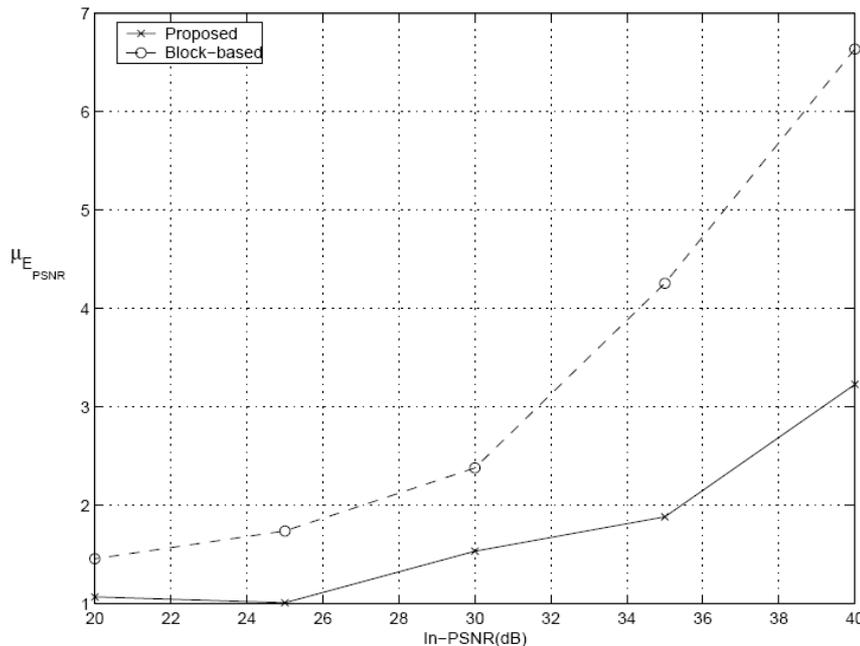
$$\sigma_{B_R}^2 = \text{median}\{\sigma_{B_h}^2 : h = 1, 2, 3\}$$

$$T = \text{const}$$

# Структурный

При вычислении  $\xi_B$  можно учитывать не все пиксели, а только каждый  $s$ -ый.

$$\xi_B = \sum_i \sum_{k=1}^8 |I_k(i)|$$



Метод работает в 4 раза быстрее, чем блочный и дает лучший результат.

# Содержание доклада

- ◆ Введение
- ◆ Wavelet
- ◆ Block-based
  - Classic
  - Structure-oriented
  - Temporal SO
- ◆ Ссылки

# Структурный метод

## Расширение

Применим структурный метод к видео.

Будем хранить несколько кадров. Из них образуем 3D блок, который разобьем на набор кубов.

Детектор границ теперь работает еще и как детектор движения.

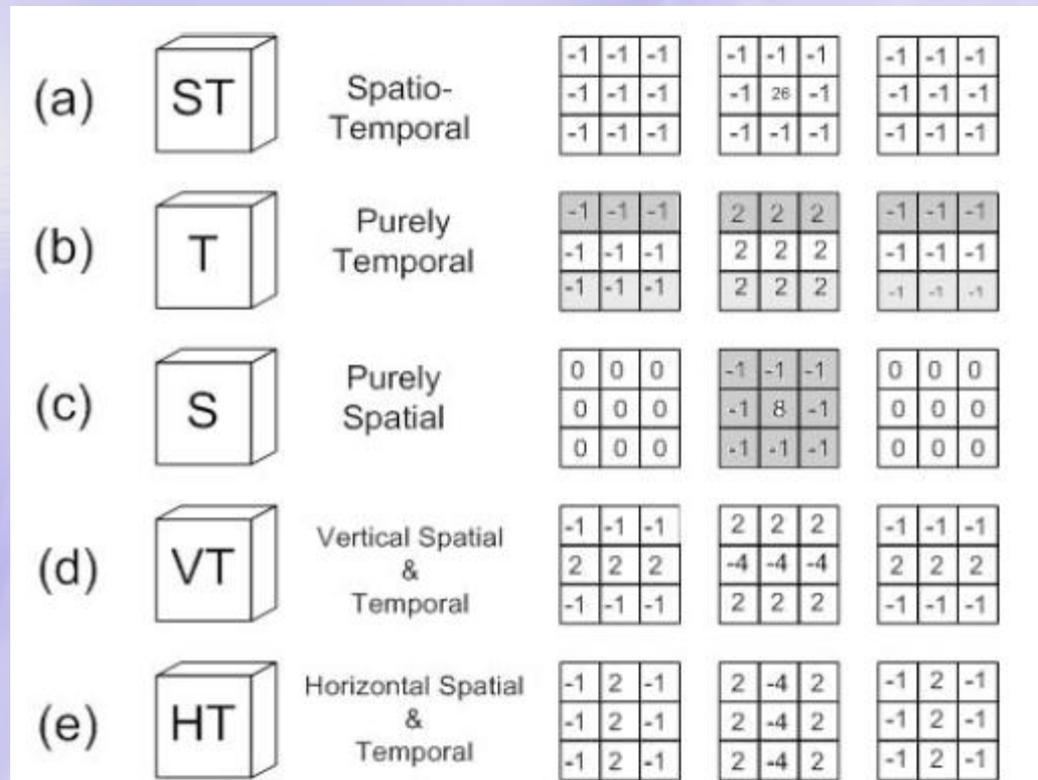


Fig. 1. Homogeneity analyzer cubical masks where pixels in the same gray level belong to one plain.

# Структурный метод

## Расширение

Блоки могут быть гладкими во времени (low-motion), но совсем не гладкими пространственно (текстура). Поэтому меры гладкости не смешиваются, для каждого из направлений есть своя.

$$\zeta_{ST} = \left| \frac{\partial^2 V}{\partial i^2} + \frac{\partial^2 V}{\partial j^2} + \frac{\partial^2 V}{\partial n^2} \right|$$

$$\zeta_T = \left| \frac{\partial^2 V}{\partial n^2} \right|$$

$$\zeta_S = \left| \frac{\partial^2 V}{\partial i^2} + \frac{\partial^2 V}{\partial j^2} \right|$$

$$\zeta_{VT} = \left| \frac{\partial^2 V}{\partial j^2} + \frac{\partial^2 V}{\partial n^2} \right|$$

$$\zeta_{ZT} = \left| \frac{\partial^2 V}{\partial i^2} + \frac{\partial^2 V}{\partial n^2} \right|$$

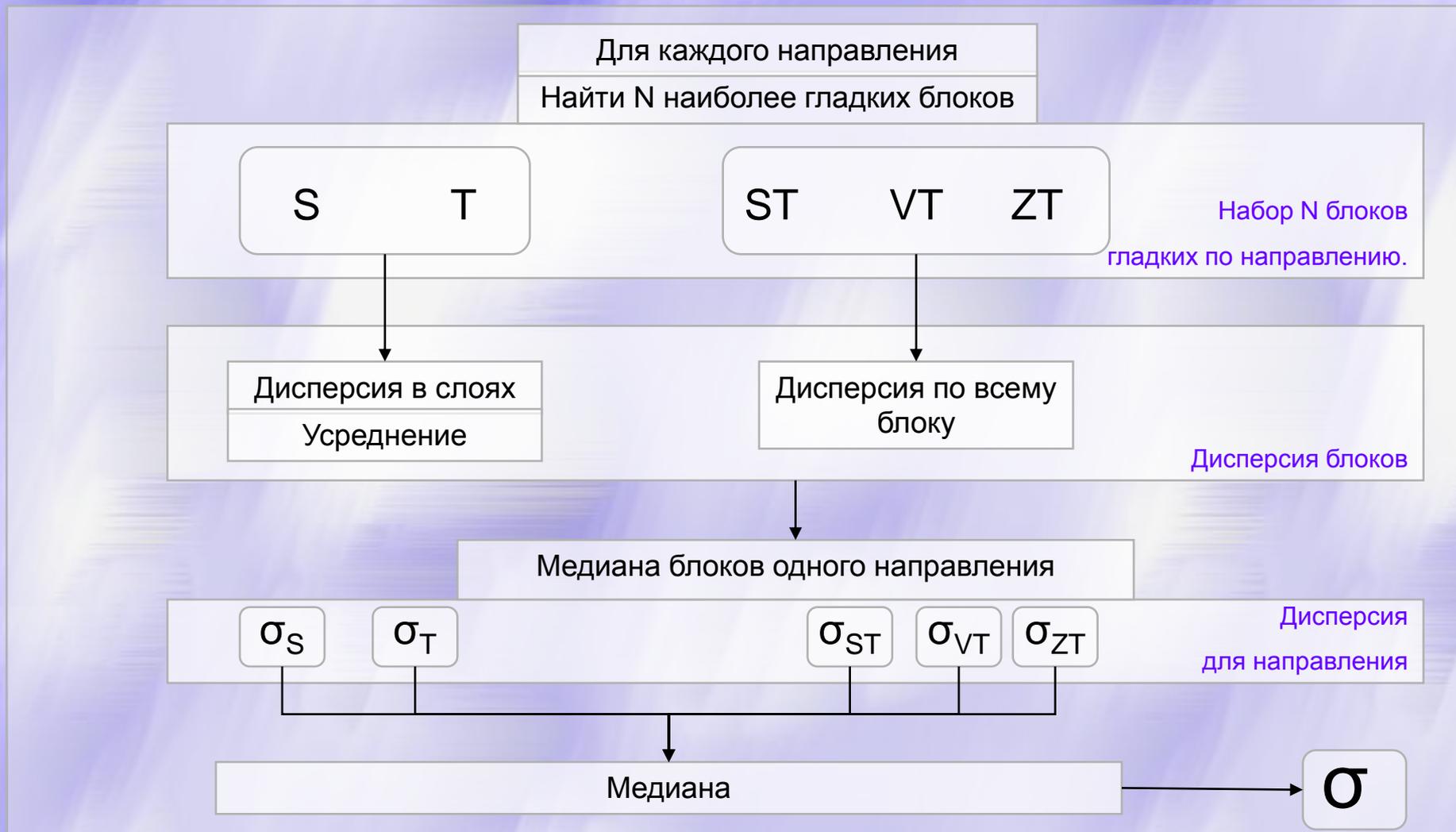
# Структурный метод

## Расширение. Алгоритм.

1. Для каждого направления:
  - ◆ Выбрать  $N$  наиболее гладких по этому направлению блоков.
  - ◆ Для каждого выбранного блока
    - ◆ Если блок гладок в  $S$  ( $T$ ), то дисперсия в нем равна среднему дисперсий по слоям  $S$  ( $T$ ).
    - ◆ Если блок гладок в  $ST$ ,  $VT$  или  $ZT$ , то дисперсия считается по всему блоку обычным образом.
  - ◆ Дисперсия направления равна среднему (или медиане) дисперсий блоков.
2. Сигма шума равна среднему (медиане) дисперсий по направлениям.

# Структурный метод

## Расширение. Алгоритм.

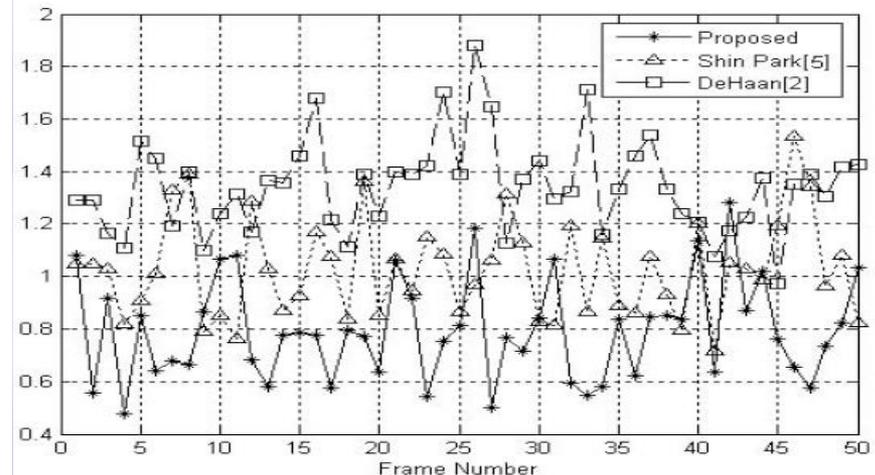


# Структурный метод

## Расширение. Сравнение.

Метод требует много дополнительной памяти, но его точность значительно выше, чем в других методах.

	20 dB		30 dB		40 dB		
Alg.	$\mu_{E_k}$	$\sigma_{E_k}$	$\mu_{E_k}$	$\sigma_{E_k}$	$\mu_{E_k}$	$\sigma_{E_k}$	TR
Inter-frame							
Ours	0.61	0.83	0.87	0.91	0.98	1.08	1.0
deHaan	1.30	1.77	1.53	1.79	5.54	5.78	1.1
Intra-frame							
AVG	1.99	1.20	3.21	1.42	4.34	1.70	4.7
MAD	1.75	1.26	2.12	1.81	3.36	2.70	2.4
G-Block	0.79	1.13	1.01	1.20	1.10	1.24	2.5
Struct	1.60	1.55	2.39	1.25	1.91	1.16	0.8



# Содержание доклада

- ◆ Введение
- ◆ Wavelet
  - MAD
  - MBE
  - Trained moments
  - CDF
- ◆ Block-based
  - Classic
  - Structure-oriented
  - Temporal SO
- ◆ Ссылки

# Список литературы



1. S.I. Olsen. Noise variance estimation in images. In In 8th Scandinavian Conference on Image Analysis, Norway, 1993.
2. D.L. Donoho, "De-noising by soft-thresholding," IEEE Transactions on Information Theory, vol. 41, no. 3, pp. 613–627, 1995
3. De Stefano, A., White, P.R. and Collis, W.B. (2004) Training methods for image noise level estimation on wavelet components. EURASIP Journal on Applied Signal Processing, 2004, (16), 2400-2407.
4. D.-H. Shin, R.-H. Park, S. Yang, J.-H. Jung, "Block-based noise estimation using adaptive Gaussian filtering", IEEE Trans. Consumer Electronics, vol. 51, no. 1, pp. 218–226, Feb. 2005
5. A. Amer and E. Dubois, "Fast and reliable structure-oriented video noise estimation," IEEE Transactions on Circuits and Systems for Video Technology, vol.15, pp. 113–118, January 2005.
6. M. Ghazal, A. Amer, and A. Ghrayeb, "Structure-Oriented Spatio-Temporal Video Noise Estimation", in Proc. IEEE Int. Conference on Acoustics, Speech, and Signal Processing (ICASSP), Toulouse, France, May 2006, pp. 845-848.

# Вопросы



?