

Быстрое межкадровое преобразование с использованием локальных бинарных паттернов.

Егоров А.И. Горбацевич В. С., Вишняков Б. В.

gvs@gosniias.ru

ФГУП ГосНИИАС

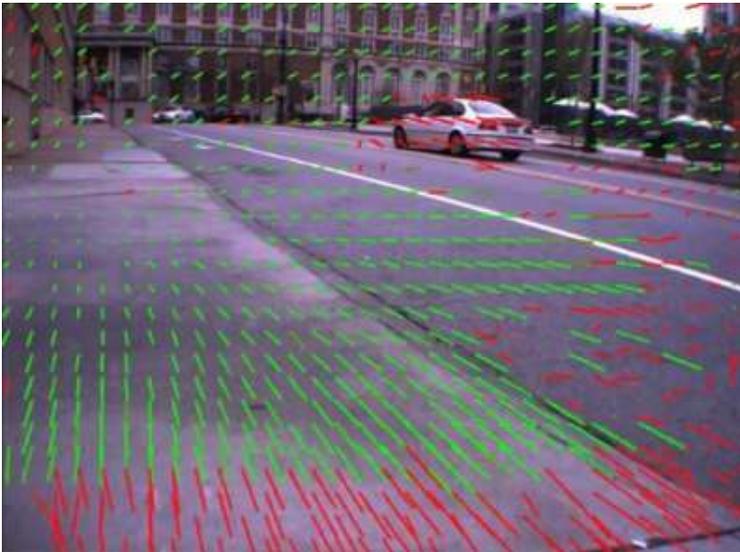
Проблема



- **низкая производительность системы**
- **жесткие требования по быстродействию**
- **высокое разрешение видеопотока**
- **“сложная” модель преобразований**

Существующие методы

- Optical flows



- Feature points



- другие методы

Допущения

- Площадь движущихся в одном направлении объектов менее половины кадра
- Отсутствуют значительные изменения в условиях съемки между соседними кадрами
- Меж кадровые изменения относительно невелики

Этапы

1. Обнаружение особых точек.
2. Нахождение связей между особыми точками на текущем и предыдущем кадрах, формирование из них пар.
3. Формирование гипотезы о преобразовании предыдущего кадра в текущий.

Детекторы особых точек

- Детектор Харриса/Форстера
- DOG(SIFT)
- SURF
- Детектор FAST
- SUSAN
- ...

Почему DoG

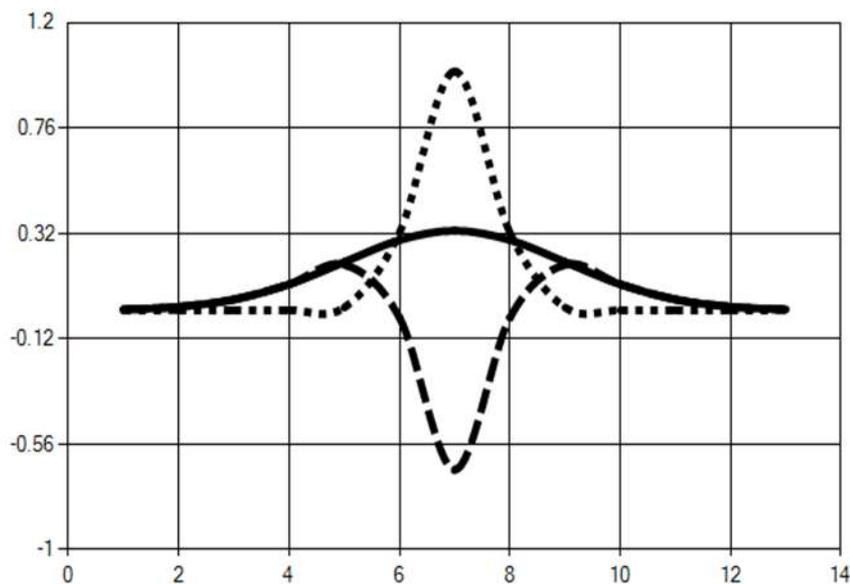
- Аффинно-инвариантность
- Легкая реализация
- Высокая производительность
- Хорошо себя зарекомендовал и широко используется например в SIFT

1. Обнаружение особых точек - DoG

► Разница Гауссиан (DoG):

$$DoG(x, y) = (G(x, y, \sigma) - G(x, y, K \cdot \sigma))$$

$$G(x, y, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2+y^2}{2\sigma^2}}$$



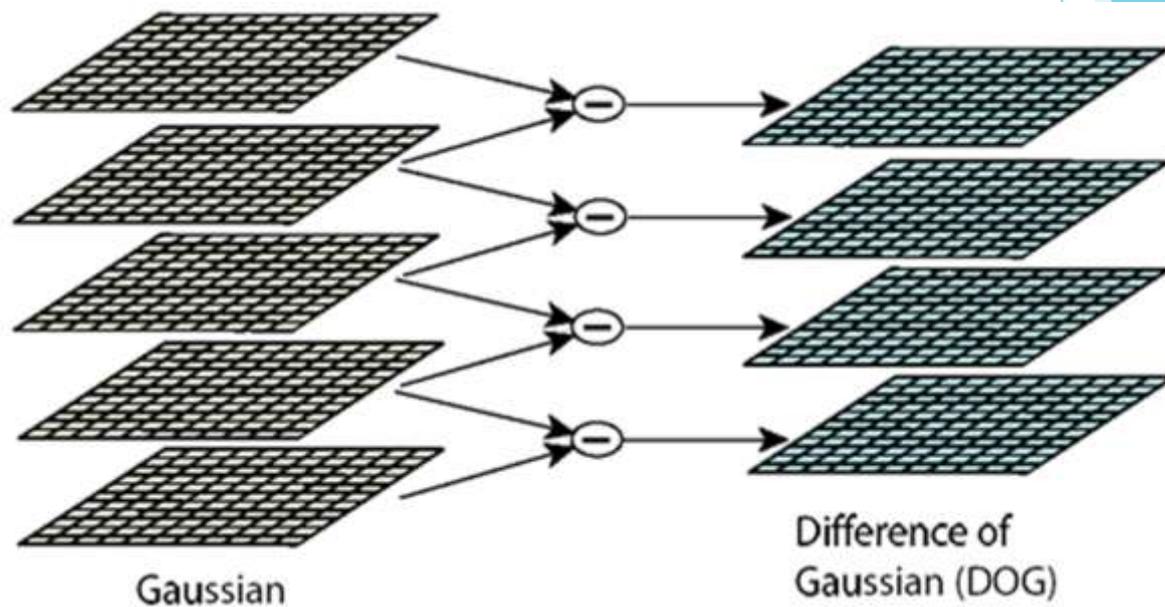
Если $K \sim 1.6$ LoG отлично аппроксимируется DoG

Пирамида DoG

- ▶ Многомасштабность:

$$IDog^i(x, y) = I * (G(x, y, K_{i-1} \cdot \sigma) - G(x, y, K_i \cdot \sigma))$$
$$K_i \in \{1.6, 2.56, 4.096, \dots 1.6^N\}$$

Предлагается
рассчитывать DoG
на каждом уровне
для обнаружения
как малых, так и
больших
особенностей

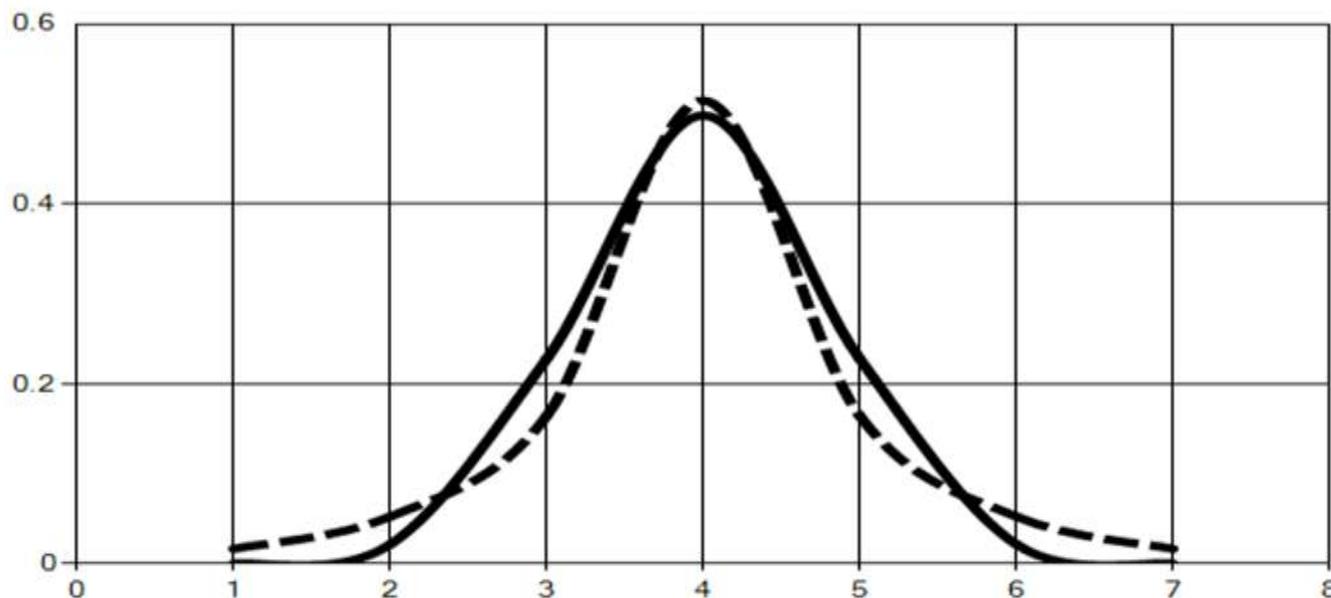


Рекурсивный DoG

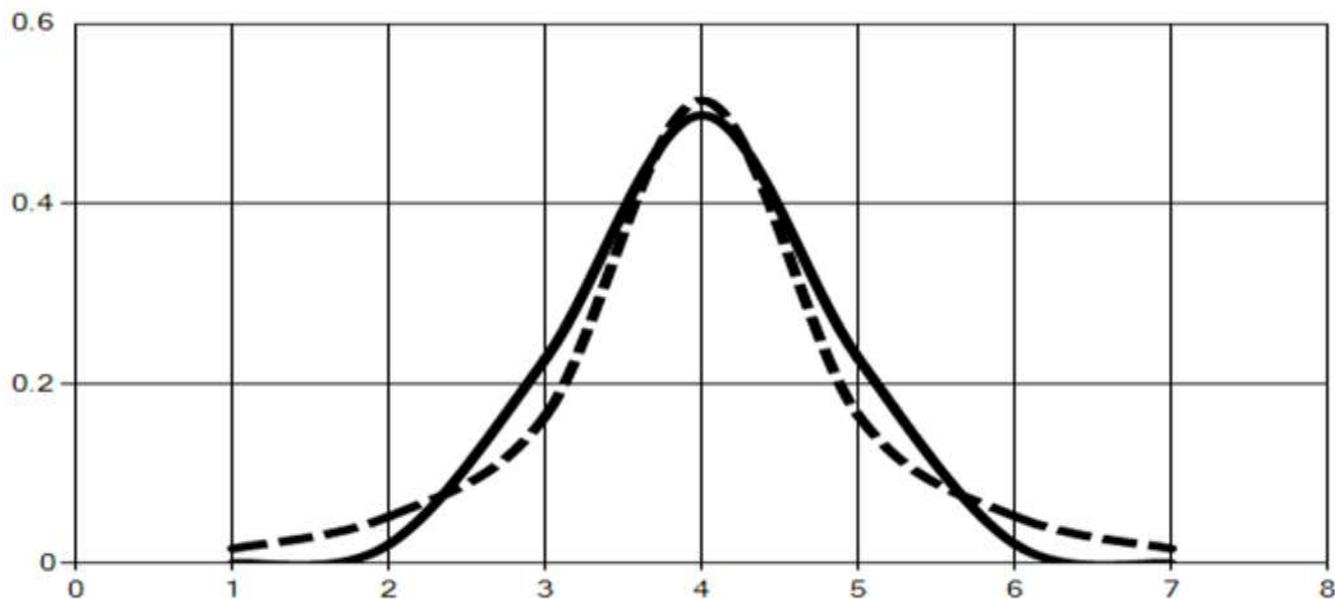
С практической точки зрения можно использовать простой рекурсивный фильтр первого порядка, для ускорения вычислений. Форма такого фильтра:

$$H(z) = H_R * H_L,$$
$$H_R = \frac{A}{b_0 - b_1 z^1}, H_L = \frac{A}{b_0 - b_1 z^{-1}},$$

где b_0, b_1 – коэффициенты фильтра.



Рекурсивный DoG



Прямой проход:

$$\text{tmp}[x] = B \text{ in}[x] + b_1 \text{ tmp}[x-1] \quad B = 1 - b_1$$

Обратный проход:

$$\text{out}[x] = B \text{ tmp}[x] + b_1 \text{ out}[x+1]$$

Сопоставление особенностей

- SURF (G-SURF, uSURF ...)
- SIFT
- ORB

...

- Низкое время работы

Хотелось бы использовать те
данные которые уже
подсчитаны...

2. Сопоставление особенностей: Пороговый LBP - дескриптор



200	155	40
75	143	47
23	47	25

1	1	0
0	x	0
0	0	0

1
1
0
0
0
0
0
0

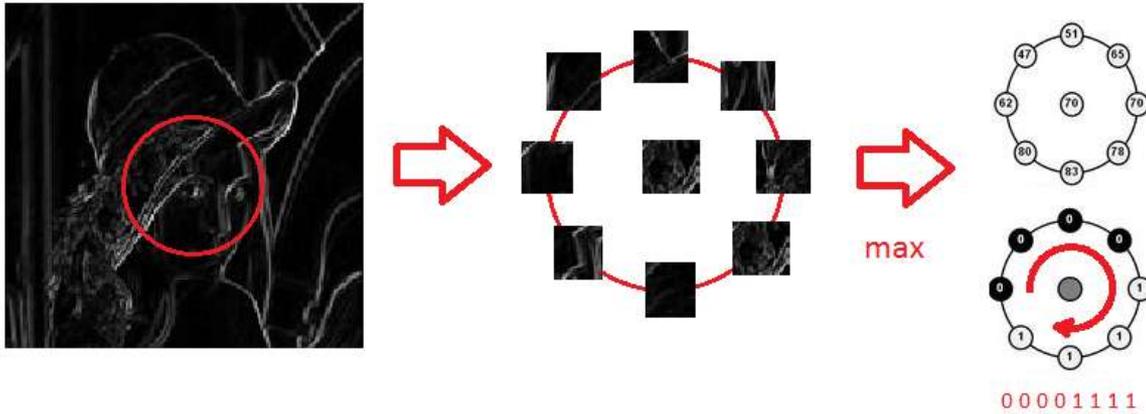
$$C_{LBP}(x, y) = \begin{cases} 1: I(x, y) - I(x + dx, y + dy) \geq 0 \\ 0: I(x, y) - I(x + dx, y + dy) < 0 \end{cases}$$

dx, dy - относительные координаты точки окрестности

Пороговый LBP:

$$C_{LBP}(x, y, thr) = \begin{cases} 1: I(x, y) - I(x + dx, y + dy) \geq thr \\ 0: I(x, y) - I(x + dx, y + dy) < thr \end{cases}$$

MaxLBP

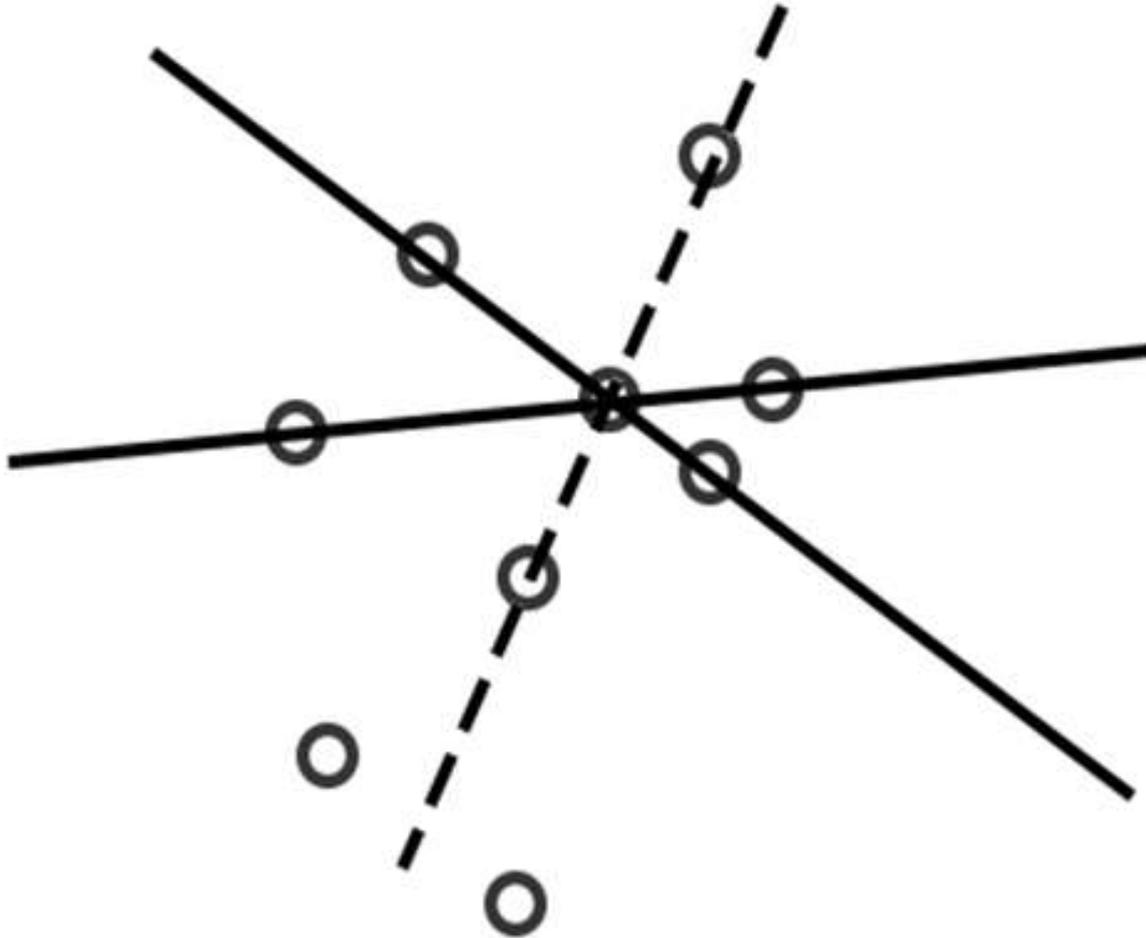


$$D(a, b) = \sum_{k=1}^K D(\text{MaxLBP}_k(a), \text{MaxLBP}_k(b)) + C \cdot (\text{DOG}(a) - \text{DOG}(b))^2$$

где a, b – сравниваемые точки;
 D – расстояние Хемминга;
 C – весовой коэффициент.

Расчёт преобразования

Проблема равного числа голосов:



Расчёт преобразования

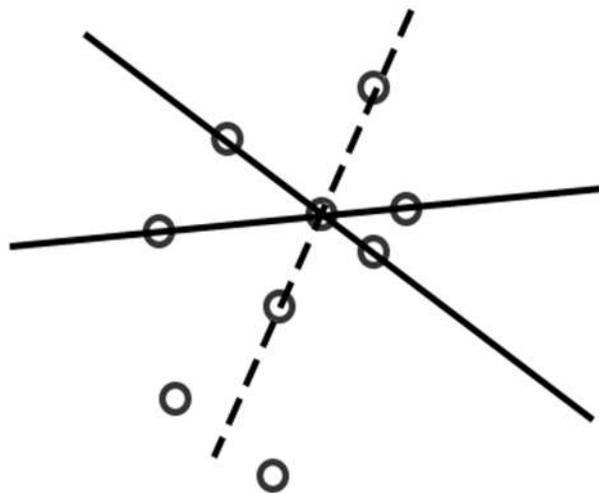
Функционал качества для RANSAC:

$$Q^*(A) = \sum_{(x_1, x_2) \in X} q(x_1, x_2) + K \sum_{(x_1, x_2) \in X} (A \times x_1 - x_2)^2,$$

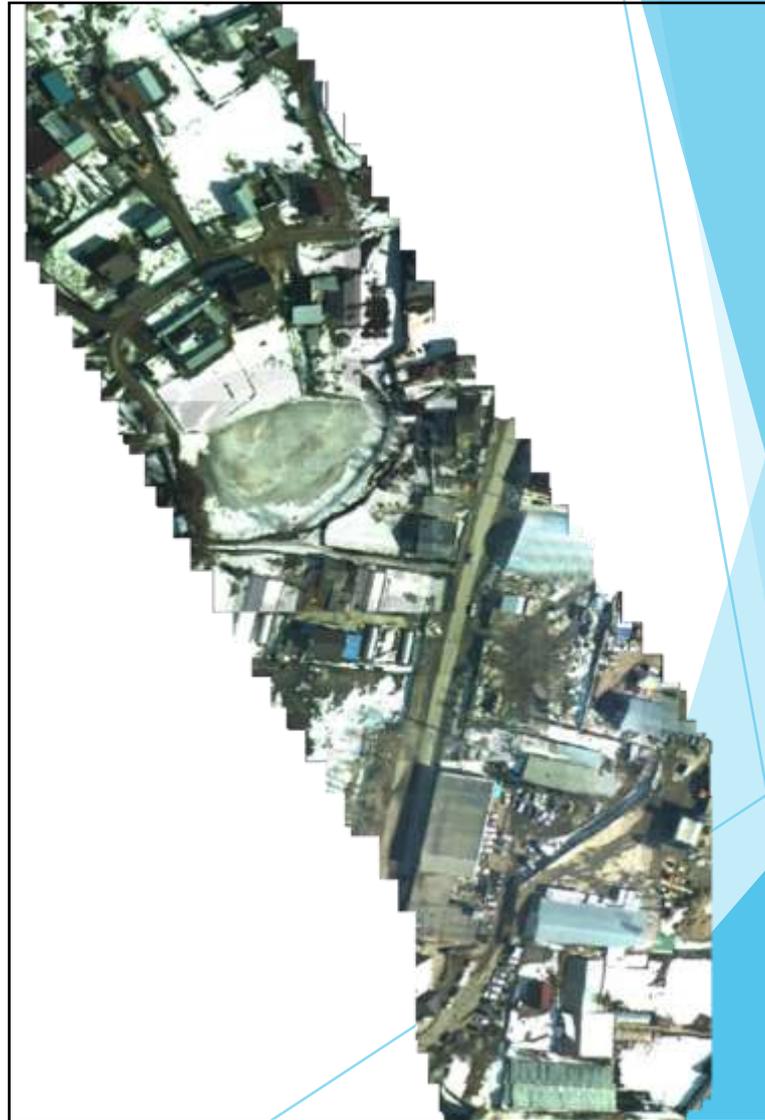
$$q(x_1, x_2) = \begin{cases} 1: \|A \times x_1 - x_2\|^2 > d \\ 0: \|A \times x_1 - x_2\|^2 \leq d \end{cases}$$

A - матрица преобразования

X - множество пар точек на соседних кадрах



Результаты



Результаты

1. Предлагаемый подход позволяет выполнять сшивку в реальном времени видеопотока: 2048 x 1520@20 Hz сru core i7-3517UE(2C/4Mb 17W)
2. Предлагаемый подход позволяет эффективно работать с преобразованиями вплоть до проективных.
3. Предлагаемый подход позволяет получать высококачественную сшивку аэрофотоснимков.

