

# Выделение объекта в видеопотоке в различных условиях освещения

Тарбаев Б.Б.<sup>1</sup>, Калайда В.Т.<sup>1</sup>  
tarbaevbb@gmail.com|kvt@iao.ru

<sup>1</sup>Национальный исследовательский Томский государственный университет, Томск, Россия

*Целью данной работы является выделение объекта в видеопотоке в режиме реального времени. Для достижения данной цели предлагается модификация SIFT, в которой уменьшается вычислительная сложность гауссового размытия. Предлагается аппроксимировать этот алгоритм с помощью Box linear filter.*

**Ключевые слова:** видеопоток, выделение объекта, фильтр Гаусса.

## Selecting an object in the video stream under different lighting conditions

Tarbaev B.B.<sup>1</sup>, Kalayda V.T.<sup>1</sup>  
tarbaevbb@gmail.com|kvt@iao.ru

<sup>1</sup>the National Research Tomsk State University, Tomsk, Russia

*The purpose of this work is to select an object in the video stream in real time. To achieve this goal, a modification of SIFT is proposed, in which the computational complexity of the Gaussian blur is reduced. It is suggested to approximate this algorithm using the Box linear filter.*

**Keywords:** video stream, object tracking, Gaussian filter.

### 1. Введение

В связи с широким применением систем видеонаблюдения возникает проблема повышения эффективности обработки и использования «огромных» объёмов информации предоставляемыми ими. Полученные данные также включают различные шумы (освещённость, резкость, контрастность и т.д.). Существующие в настоящее время подходы к решению задачи выделения полезной информации в видеопотоке обладают двумя недостатками: невысокое быстродействие и невысокая для практического применения точность [14].

Среди всех задач выделения информации из видеопотока наибольшее практическое значение имеет задача поиска заданного объекта в кадре в условиях отмеченных помех, которая естественным образом возникает в самых разных областях: от компьютерного зрения до задач обеспечения безопасности.

При поиске объекта в видеопотоке необходимо решать две задачи:

1. быстрый поиск областей, претендующих на дальнейшее сравнение, схожих с образцом;
2. оценка мера близости между выделенной областью и образцом.

Исходя из вышесказанного, целью данной работы является разработка алгоритма и программная реализация системы поиска объекта в видеопотоке.

### 2. Метод определения ключевых точек

На первом этапе определяются инвариантные характеристики двух соседних изображений путём выделения особых точек и их дескрипторов. Ключевая точка (особая точка) – это точка изображения, окрестность которой можно отличить от окрестности любой другой точки изображения в некоторой другой окрестности ключевой точки.

Дескриптор – это идентификатор ключевой точки, выделяющий её из остального множества ключевых точек.

Для обнаружения особых точек используется модифицированный алгоритм преобразования масштабно-инвариантных характеристик (англ. Scale Invariant Feature Transform, SIFT). Алгоритм SIFT показывает лучшие

результаты обнаружения особых точек по качеству чёткости и контрастности цвета и освещения изображения, чем другие алгоритмы обнаружения точек [6,16]. Недостатком SIFT является его вычислительная сложность, что ограничивает его применение в режиме постобработки [3].

Одним из способов преодоления указанных недостатков является разработка эффективных численных методов поиска точечных особенностей, расчета дескрипторов, вычисления расстояний между дескрипторами, поиска ближайших дескрипторов. К таким методам относятся методы приближенного анализа дифференциальной структуры изображений на основе вейвлетов и интегральных изображений [1], методы индексирования дескрипторов [2,10,19], методы понижения размерности дескрипторов [7].

В работах, посвященных сравнительному анализу методов-детекторов [5,11,12], отмечается, что используемые детекторы имеют относительно невысокую степень воспроизводимости, т.е. точечные особенности, найденные в эталонном изображении, не всегда могут быть найдены в преобразованном изображении, хотя качество сопоставления изображений остается высоким. При этом число невоспроизводимых точечных особенностей может достигать 40% от общего числа найденных точечных особенностей. Это наблюдение говорит о том, что, с точки зрения задачи сопоставления изображений, невоспроизводимые точечные особенности являются избыточными данными. В то же время в известных алгоритмах сопоставления изображений все точечные особенности обрабатываются единообразно. Таким образом, обработка избыточных данных представляет собой источник дополнительной вычислительной сложности, исключение которого не повлияет на результат сопоставления изображений.

В данной работе алгоритм SIFT будет рассмотрен с точки зрения теории алгоритмов.

Алгоритм SIFT [4,6,8,15,18] состоит из четырех главных этапов:

- 1) Определение максимума масштабируемого пространства: первый этап вычислений. Он реализуется посредством вычисления функции разности Гауссиана для определения потенциальных ключевых точек

инвариантных к масштабированию [13] и изменению ориентации.

Инвариантность относительно масштаба достигается за счет нахождения ключевых точек для исходного изображения, взятого в разных масштабах. Для этого строится пирамида гауссианов: все масштабируемое пространство разбивается на некоторые участки — октавы, причем часть масштабируемого пространства, занимаемого следующей октавой, в два раза больше части, занимаемой предыдущей. К тому же, при переходе от одной октавы к другой размеры изображения уменьшаются вдвое. Естественно, что каждая октава охватывает бесконечное множество гауссианов изображения, поэтому строится только некоторое их количество  $N$ , с определенным шагом по радиусу размытия. С тем же шагом достраиваются два дополнительных гауссиана (всего получается  $N+2$ ), выходящие за пределы октавы.

Гауссианом является изображение (рис. 1, слева):

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y),$$

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

где  $L$  – значение гауссиана в точке с координатами  $(x, y)$ ,

$\sigma$  – радиус размытия,

$G$  – гауссово ядро,

$I$  – значение яркости исходного изображения,

$*$  – операция свертки.

Разностью гауссианов называют изображение, полученное путем попиксельного вычитания одного гауссиана исходного изображения из гауссиана с другим радиусом размытия (рис. 1, справа):

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) =$$

$$L(x, y, k\sigma) - L(x, y, \sigma),$$

где  $k = 2^{\frac{1}{s}}$  обычно имеет значение  $\sqrt{2}$ ,

$s \geq 1$  и является целым числом.

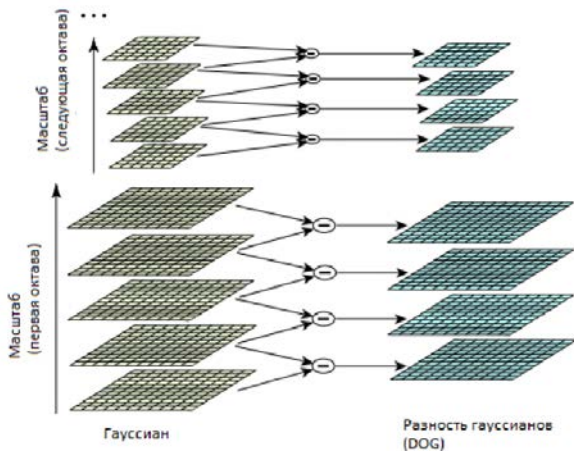


Рис. 1. Пирамида Гауссианов

2) Определение ключевых точек. Ключевые точки определяются на основе меры их стабильности.

Будем считать точку ключевой, если она является локальным экстремумом разности гауссианов.

Для поиска экстремумов используется метод, схематично изображенный на рис. 2. Если значение разности гауссианов в точке, помеченной X, больше (меньше) всех значений в точках, помеченных окружностями, то эта точка считается точкой экстремума. Для того, чтобы обнаружить локальные максимумы и минимумы  $D(x, y, \sigma)$  каждая точка сравнивается с ее ближайшими соседями в текущем изображении и с девятью соседями в масштабах выше и ниже.

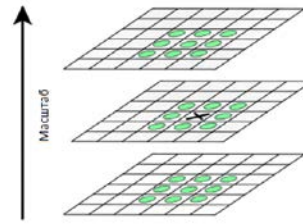


Рис. 2. Поиск экстремумов разности Гауссианов

Следующий шаг по поиску ключевых точек – проверка пригодности точки экстремума на роль ключевой. Сначала определяются координаты ключевой точки с субпиксельной точностью. Потом происходит удаление точек с малым контрастом и точек, находящихся на границе объектов.

3) Определение ориентации. Для каждой ключевой точки может быть определено одно или несколько направлений, основанных на локальных направлениях градиентов. Все последующие операции выполняются на изображении, которое было преобразовано в соответствии с ориентацией и масштабом каждой ключевой точки, обеспечивая тем самым инвариантность к этим преобразованиям.

Направление, соответствующее характерной точке, вычисляется исходя из направлений градиентов точек, соседних с характерной. Все вычисления градиентов приводятся на изображении в пирамиде гауссианов, с масштабом наиболее близким к масштабу характерной точки. Величина и направление градиента в точке  $(x, y)$  вычисляются по формулам.

$$m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2}$$

$$\theta(x, y) = \tan^{-1}((L(x, y+1) - L(x, y-1)) / (L(x+1, y) - L(x-1, y))),$$

где  $m$  – величина градиента,  
 $\theta$  – его направление.

4) Построение дескрипторов.

Как и направление ключевой точки, дескриптор вычисляется на гауссиане, ближайшей по масштабу к характерной точке, и исходя из градиентов некотором окне характерной точки. Перед вычислением дескриптора это окно поворачивают на угол направления ключевой точки, чем достигается инвариантность относительно поворота.

### 3. Оценка сложности алгоритма

Пусть размер изображения  $N \times M$ . Обозначим за  $\alpha$  долю пикселей, которая окажется экстремумами, выразим ее как дробь  $0 < \alpha < 1$ . Таким образом  $\alpha NM$  – количество экстремумов. Только небольшая часть из этих экстремумов будет квалифицирована как ключевые точки. Обозначим их как  $0 < \beta < 1$ . Итого имеем  $\alpha\beta NM$  ключевых точек. Далее проверяем не были ли пропущены ключевые точки, обозначим эту долю пикселей  $\gamma$ . Если такие точки нашлись, то они становятся  $\gamma NM$  новых ключевых точек, таким образом всего у нас  $(\alpha\beta + \gamma)NM$  ключевых точек. Далее эти точки будут преобразованы в дескрипторы. Их число будет  $(\alpha\beta + \gamma)NM$ .  $\alpha, \beta, \gamma$  зависят от рассматриваемого изображения [17].

В таблице 1 представлена вычислительная сложность и число операций каждого из этапов алгоритма.

Таблица 1. Вычислительная сложность операций

Этап	Сложность	Число операций
Гауссовое размытие	$O(NMw^2s)$	$4NMw^2s$
Разница гауссиан	$O(sNM)$	$4NM_s$
Обнаружение экстремумов	$O(sNM)$	$104sNM$
Обнаружение ключевых точек	$O(\alpha sNM)$	$100\alpha sNM$
Назначение ориентации	$O(sNM(1 - \alpha\beta))$	$48sNM$
Генерация дескриптора	$O(x^2NM(\alpha\beta + \gamma))$	$1520x^2(\alpha\beta + \gamma)NM$

Как видно из таблицы, наибольшей вычислительной сложностью обладает операция получения гауссового размытия. Таким образом следующим шагом является уменьшение вычислительной сложности алгоритма.

#### 4. Уменьшение сложности гауссового размытия

Пусть  $f$  – изображение,  $g$  – ядро гаусса, тогда фильтр Гаусса можно представить, как:

$$Gb(i, j) = \sum_{y=j-r}^{j+r} \sum_{x=i-r}^{i+r} f(x, y) * g(x, y).$$

Алгоритм Vox linear filter (BLF) позволяет аппроксимировать размытие по Гауссу [9]. Обозначим половину размера поля как  $r$  – радиус окна, тогда постоянное значение веса равно  $\frac{1}{(2r)^2}$ . Здесь вес (ядро) постоянный и лежит в квадрате (поле). Особенностью алгоритма является то, что повторное применение BLF будет аппроксимировать гауссовое размытие. Определим BLF как:

$$Bb(i, j) = \sum_{y=j-r}^{j+r} \sum_{x=i-r}^{i+r} f(x, y) / (2r)^2.$$

Сложность этого алгоритма по-прежнему является квадратичной  $O(NMr^2)$ . Но в данном случае нам не нужно вычислять новые веса. Для аппроксимации гауссового размытия, данный алгоритм обычно применяют 3 раза.

Определим горизонтальное размытие и глобальное размытие:

$$b_h(i, j) = \sum_{x=i-br}^{i+br} f(i, y) / (2r),$$

$$b_g(i, j) = \sum_{y=j-br}^{j+br} b_h(x, j) / (2r),$$

$$b_g(i, j) = \sum_{y=j-r}^{j+r} (\sum_{x=i-r}^{i+r} f(x, y) / (2r)) / (2r) = \sum_{y=j-r}^{j+r} \sum_{x=i-r}^{i+r} f(x, y) / (2r)^2.$$

Отсюда следует, что глобальный фильтр является фильтром BLF. Полное размытие как и горизонтальное имеют сложность  $O(NMr)$ , поэтому фильтр BLF тоже имеет сложность  $O(NMr)$ .

Попытаемся вычислить одномерное размытие еще быстрее. Если мы хотим вычислить горизонтальное размытие, то мы вычисляем:  $b_h(i, j)$ ,  $b_h(i + 1, j)$ ,  $b_h(i + 2, j)$  и т.д. Но эти значения почти одинаковы. Единственное различие в том, что они равны в самом левом и правом значении:

$$b_h(i + 1, j) = b_h(i, j) + f(i + r + 1, j) - f(i - r, j).$$

Таким образом, мы можем создать буфер, в который мы поместим значение самой левой ячейки. Затем мы вычислим следующие значения, просто изменив предыдущее. Сложность этого действия будет  $O(NM)$ . Но его надо выполнить дважды, чтобы получить размытие ящика, которое в свою очередь надо выполнить 3 раза, чтобы получить гауссовое размытие. Таким образом, сложность алгоритма равна  $6 \cdot O(NM)$ .

#### 5. Результаты

В таблице 2 приведены результаты оценки времени выполнения и вычислительная сложность алгоритмов для изображения 800x200 пикселей.

Таблица 2. Время выполнения и вычислительная сложность алгоритмов

Алгоритм	Время: мс, r=5	Время: мс, r=10	Вычисл-ая сложность
Фильтр Гаусса	7077	27021	$O(NMr^2)$
Vox filter	586	2437	$O(NMr^2)$
Vox filter оптимизация 1	230	435	$O(NMr)$
Vox filter оптимизация 2	32	34	$O(NM)$

#### 6. Заключение

В статье представлен метод для оптимального вычисления размытия по Гауссу для использования в алгоритме SIFT. Проведённые вычислительные эксперименты подтверждают эффективность и устойчивость работы алгоритма. Различия изображений, полученных стандартным фильтром Гаусса и его аппроксимацией, не превышает 0,2%.

В дальнейшем планируется: оптимизация работы программы для ускорения обработки данных, проведения тестирования модифицированного SIFT.

#### 7. Литература

- [1] Кудряшов А. П. Извлечение и сопоставление точечных особенностей. Электронный научный журнал Исследовано в России, т.10, 2007. <http://masters.donntu.org/2012/iii/chigarev/library/article1.htm>
- [2] Пименов В. Ю. Метод поиска нечетких дубликатов изображений на основе выявления точечных особенностей. Труды РОМИП2007-2008, с. 145-158. СПб.:НУ ЦСИ, 2008.
- [3] Со С.А. Реализация цифровых технологий обработки изображений в адаптивных системах управления сварочными процессами: диссертация кандидата техн. наук. МГТУ им. Н.Э. Баумана, 2015, 145 с.
- [4] Chu E., Hsu E., Yu S. Image-Guided Tours: Fast-Approximated SIFT with U-SURF Features, Stanford University. 2007. pp. 6-12.
- [5] Frintop S. The High Repeatability of Salient Regions. Proc. Of ECCV Workshop «Vision in Action: Efficient Strategies for Cognitive Agents in Complex Environment». 2008.
- [6] Juan L., Gwun O. A Comparison of SIFT, PCA-SIFT and SURF // International Journal of Image Processing (IJIP). 2009. Vol. 3. No. 4. PP. 143-15.
- [7] Ke Y., Suthanakar R. PCA-SIFT: A More Distinctive Representation for Local Image Descriptors. In Computer Vision and Pattern Recognition, volume 2, pages 506-513, 2004.
- [8] Khan N., McCane B., Wyvill G. SIFT and SURF performance evaluation against various image deformations on benchmark dataset // International conference on digital image computing: techniques and applications. 2011. pp. 109-114.
- [9] Kovesi P. Fast Almost-Gaussian Filtering // Centre for Exploration Targeting School of Earth and Environment.
- [10] Lowe D. Distinctive Image Features from Scale-Invariant Keypoints. In International Journal of Computer Vision, volume 60, pages 91-110, 2004.

- [11] Mikolajczyk K., Schmidt C. A Performance Evaluation of Local Descriptors. *IEEE Trans. Pattern Anal. Mach. Intell.*, volume 27 (10), pages 1615-1630. 2005.
- [12] Mikolajczyk K. et al. A Comparison of Affine Region Detectors. *Int. J. of Comput. Vis.* volume 65 (1-2), pages 43-72, 2005.
- [13] Mikolajczyk K., Tuytelaars T., Schmid C., Zisserman A., Matas J., Schaffalitzky F., Kadir T., Van Gool L. A comparison of affine region detectors // *Intelligent journal of computer vision*, Vol. 65 (1-2), 2005. pp. 43-72.
- [14] Prajapati D., Galiyawala H. J. A Review on Moving Object Detection and Tracking // *International Journal of Computer Application (2250-1797) Volume 5– No. 3, April 2015 PP. 168-175.*
- [15] Tang C.Y., Wu Y.L., Hor M.K. Modified sift descriptor for image matching under interference // *Machine learning and cybernetics, international conference on*, Vol. 6, July 2008. pp. 3294 – 3300.
- [16] Valgren C., Lilienthal A.J. SIFT, SURF & Seasons: Appearance-based Long-term Localization in Outdoor Environments // *Robotics and Autonomous Systems*. 2010. Vol. 58. No. 2. PP. 149–156. DOI: <http://dx.doi.org/10.1016/j.robot.2009.09.010>.
- [17] Vinukonda P. A study of the scale-invariant feature transform on a parallel pipeline // *B.Tech., JNTU University*, 2007 May 2011.
- [18] Zhang Z. A Flexible new technique for camera calibration. technical report MSRTR // *Microsoft Research*. December 1998. pp. 98-71.
- [19] Zhao W.-L., Ngo C.-W., Tan H.-K., Wu X. Near-Duplicate Keypoint Identification with Interest Point Matching and Pattern Learning. In *IEEE Transactions on Multimedia*, volume 9(5), pages 1037-1048, 2007.

## Об авторах

Тарбаев Баир Баясхаланович, аспирант кафедры оптико-электронных систем и дистанционного зондирования радиофизического факультета Томского государственного университета. Его e-mail [tarbaevbb@gmail.com](mailto:tarbaevbb@gmail.com).

Калайда Владимир Тимофеевич, д.т.н., профессор кафедры оптико-электронных систем и дистанционного зондирования радиофизического факультета Томского государственного университета. Его e-mail [kvt@iao.ru](mailto:kvt@iao.ru).