

УДК 004.932.2

АНАЛИЗ МЕТОДОВ ДЕТЕКТИРОВАНИЯ, ОПИСАНИЯ И СОПОСТАВЛЕНИЯ КЛЮЧЕВЫХ ТОЧЕК ИЗОБРАЖЕНИЙ

А. А. Денисов, аспирант РГРТУ, Рязань, Россия;

orcid.org/0009-0006-6765-5054, e-mail: denal777@gmail.com

А. И. Новиков, д.т.н., доцент, профессор кафедры ВМ РГРТУ, Рязань, Россия;

orcid.org/0000-0002-8166-8234, e-mail: novikovanatoly@yandex.ru

***Цель работы** – описание и проведение сравнительных испытаний известных методов и алгоритмов обнаружения и описания ключевых точек изображения, а также установления попарного соответствия между ключевыми точками двух изображений. Задача работы заключается в выявлении сильных и слабых сторон каждого из методов. В экспериментах сравнивались следующие методы детектирования и описания ключевых точек: SIFT, SURF, FAST/FREAK, BRISK, KAZE и ORB. Сравнение детекторов ключевых точек проводилось по критериям быстродействия и количества найденных ключевых точек. Deskрипторы сравнивались по показателю точности установления соответствия между ключевыми точками двух изображений и по времени, затраченному на описание ключевых точек и их сопоставление.*

***Ключевые слова:** ключевая точка, детектор, дескриптор, время обработки, корректность соответствия, SIFT, SURF, FAST, FREAK, BRISK, KAZE, ORB.*

DOI: 10.21667/1995-4565-2024-89-104-116

Введение

Сопоставление изображений лежит в основе многих задач компьютерного зрения, включая восстановление 3D-структур, одновременную локализацию и картографирование (SLAM) и обнаружение изменений [1], совмещение изображений [2, 3], робототехнику, системы автоматического пилотирования и системы безопасности, системы распознавания лиц и т. д. Задачу сопоставления изображений можно решить с помощью так называемых ключевых точек.

Ключевые точки или точки интереса – это точки (пиксели), отображающие некую уникальность, особенность изображения, по которым можно классифицировать изображение, распознать его [4]. Каждая ключевая точка имеет окрестность, обладающую определенными отличиями от окрестностей других точек. Такая особенность ключевой точки позволяет формировать ее описание в виде некоторого вектора признаков (дескрипторов). Элементами дескриптора могут быть, например, модули и направления градиентов в каждом пикселе из окрестности ключевой точки [5].

После этапа обнаружения и описания ключевых точек эти данные могут быть использованы для сравнения с ключевыми точками на других изображениях. Используя дескрипторы точек, можно найти один и тот же объект (или похожий) на различных изображениях. Особенно эффективен данный подход в случаях ограничения вычислительных возможностей и необходимости производить поиск объектов и совмещение изображений в реальном времени [6].

Существует множество алгоритмов обнаружения и описания ключевых точек, использующих различные методы обнаружения и описания ключевых точек. Как следствие, отличается их эффективность как с точки зрения времени, затраченного на вычисления и обработку изображений, так и с позиций корректности установления соответствия между парами точек на различных изображениях, подверженных различным искажениям, таким как поворот, масштабирование, изменение угла съемки, размытие, изменение освещения и наличие шумов. Учитывая множественность отличительных особенностей методов и особенностей ана-

лизируемых сцен, представляется целесообразным проведение сравнительного анализа по определенным параметрам наиболее известных и широко применяемых в цифровой обработке изображений методов и алгоритмов детектирования и описания ключевых точек.

В данной работе рассматриваются следующие методы детектирования ключевых точек и формирования дескрипторов: SIFT, SURF, FAST/FREAK, BRISK, KAZE и ORB. Все они доступны в расширении Computer Vision Toolbox для MATLAB, которое предоставляет алгоритмы, функции и приложения для проектирования и тестирования систем компьютерного зрения, 3D-зрения и обработки видео. Рассмотрим краткое описание каждого из них, но предварительно сделаем важное замечание относительно терминологии.

В цифровой обработке изображений понятия «метод» и «алгоритм» часто отождествляются. Происходит это отчасти потому, что авторы не следят за корректностью используемых терминов. Однако есть и важная причина для такого отождествления. Многие методы в цифровой обработке изображений излагаются в виде четкой последовательности действий. Это относится и к методам детектирования ключевых точек. Авторы настоящей работы в теоретической части старались использовать термин «метод» при его кратком описании. Однако в экспериментальной части работы чаще используется термин «алгоритм» поскольку в этом разделе производятся сравнения конкретных программных реализаций того или иного метода.

Теоретическая часть

Метод поиска ключевых точек **FAST (Features from Accelerated Segment Test** – авторы Эдвард Ростен, Т. Драммонд, 2006 г. [7]) предполагает выполнение следующих шагов.

1. Определение порога интенсивности. На первом шаге определяется порог интенсивности пикселей изображения, который используется для определения принадлежности пикселя к классу ключевых точек.

2. Определение кандидатов на ключевые точки. Алгоритм оценивает интенсивность его соседних пикселей (обычно 16 пикселей по окружности с центром в анализируемом пикселе (рисунок 1, заимствован из [7]), чтобы определить, является ли данный пиксель потенциальной ключевой точкой.

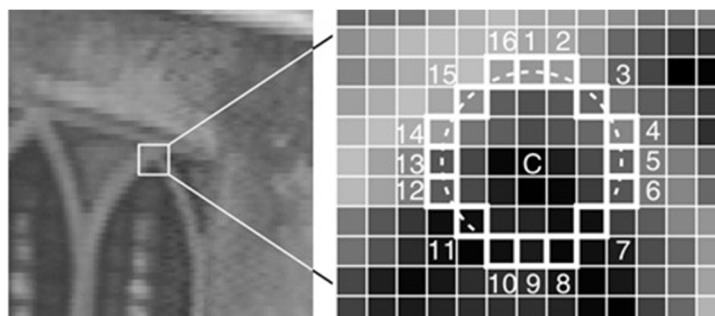


Рисунок 1 – Иллюстрация к детектору FAST
Figure 1 – Illustration for FAST detector

3. Для ускорения процесса обнаружения ключевых точек разработчики метода предложили сначала проверить только четыре пикселя окружности, расположенные на осях Ox и Oy с центром в потенциальной ключевой точке [7]. Если хотя бы три пикселя из них превышают заданный порог интенсивности, то происходит полная проверка по 16 точкам. В противном случае точка отмечается как «не ключевая».

4. Определение ключевых точек. Если центральный пиксель и все его 16 соседних пикселей либо ярче порога интенсивности, либо темнее порога интенсивности на заданную величину, то пиксель считается ключевой точкой [4, 7].

Метод FAST является одним из наиболее простых в вычислительном отношении. Поэтому его часто применяют в системах технического зрения реального времени.

Также существуют доработанные версии алгоритма FAST. Рассмотрим некоторые из них.

1. FAST-9 и FAST-12. Эти версии метода FAST используют не 16, как в оригинальной версии, а 9 или 12 точек окружности соответственно. Это позволяет увеличить скорость выполнения алгоритма.

2. FAST-AGAST. Это улучшенная версия метода FAST, которая использует алгоритм AGAST (Adaptive and Generic Accelerated Segment Test). AGAST позволяет адаптировать пороговое значение для определения точек в зависимости от уровня яркости вокруг них, что делает его более устойчивым к изменениям освещения.

3. FAST-ER (Enhanced Repeatability). Эта версия метода FAST специально разработана для повышения устойчивости к шумам и улучшения повторяемости детектированных ключевых точек на изображении. FAST-ER достигает этого путем улучшения процесса детектирования ключевых точек с использованием дополнительных шагов, таких как фильтрация и подбор лучших точек.

Поскольку метод FAST является только детектором ключевых точек, то для их описания необходимо применять другие алгоритмы. Одним из часто используемых дескрипторов является **FREAK (Fast Retina Keypoint)**. Этот дескриптор ключевой точки разработан Александром Алахи, Рафаэлем Ортиз и Пьером Вандергейнст [8]. FREAK разработан для быстрого и компактного описания ключевых точек, что делает его подходящим для реального времени и больших по объему данных.

Данный метод описания ключевой точки основывается на специфике зрительной системы человека, а точнее, сетчатки глаза. FREAK использует сетчаточную сетку выборки, которая имеет круглую форму, в которой плотность точек более высокая вблизи центра. Чтобы соответствовать модели сетчатки, алгоритм использует разные размеры ядер (радиусы) для каждой точки выборки. Особенность метода заключается в экспоненциальном изменении размера и перекрытия рецептивных полей. Рисунок 2 (заимствован из [8]) иллюстрирует топологию рецептивных полей. Каждый круг представляет собой стандартные отклонения гауссовских ядер, примененных к соответствующим точкам выборки [8].

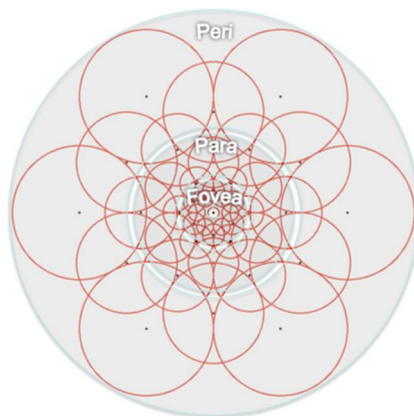


Рисунок 2 – Схема выборки фрагментов ключевой точки, имитирующая рецептивные структуры сетчатки

Figure 2 – Keypoint fragment sampling scheme simulating receptive structures of the retina

Дескриптор FREAK является бинарным и имеет длину в 512 бит [5, 8].

Метод **SIFT (Scale Invariant Feature Transform)** предложен Дэвидом Г. Лоу для решения проблем вращения изображения, аффинных преобразований, интенсивности и изменения точки обзора в сравниваемых изображениях [9].

Ключевые точки обнаруживаются с помощью каскадной фильтрации. Претендентами на ключевые точки являются такие пиксели, которые должны регистрироваться при разных масштабах и при съемке объекта под разными углами. Нахождение участков, устойчивых к изменению масштаба изображения, достигается с применением непрерывной функции масштабирования $L(x, y, \sigma)$ известной как масштабное пространство. Эта функция является

сверткой гауссовой функции переменного масштаба $G(x, y, \sigma)$ с входным изображением $I(x, y)$:

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

Для эффективного выявления устойчивых местоположений ключевых точек в масштабном пространстве находятся экстремумы функции:

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

При этом каждая точка выборки сравнивается с восьмью своими соседями на текущем изображении и девятью соседями в шкале сверху и снизу (рисунок 3, заимствован из [9]). Ключевой точка считается только в том случае, если ее значение больше всех этих соседей или меньше их всех.

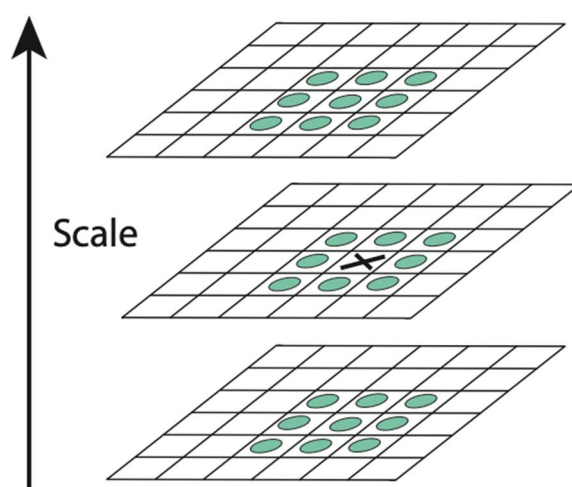


Рисунок 3 – Визуализация сравнения значений ключевой точки с соседними точками в областях 3×3 в текущем и соседних масштабах

Figure 3 – Visualization of keypoint values comparison with neighboring points in 3×3 areas at current and neighboring scales

Гистограмма ориентации с 36 интервалами, охватывающими диапазон 360 градусов, формируется на основе ориентации градиента в области вокруг точки. Наконец, локальная область ключевой точки разбивается на $4 \times 4 = 16$ подобластей по 8 значений гистограммы ориентации в каждой. Следовательно, вектор дескриптора ключевой точки состоит из $16 \times 8 = 128$ значений признаков [9].

Метод **SURF (Speeded-Up Robust Features)**, предложенный Гербертом Бейем, Андреасом Эссем, Тинне Туйтelaarс и Люком Ван Гулом, созданный как улучшенная версия метода SIFT, предлагает более быстрый и эффективный способ поиска ключевых точек с помощью использования интегрального представления изображений для ускорения вычислений [10].

Для обнаружения ключевых точек в методе SURF используется сравнение гессииана $\det H(I)$ функции яркостей $I(x, y), \dots, x = \overline{1, M}; y = \overline{1, N}$:

$$\det H(I) = I_{xx} \cdot I_{yy} - (I_{xy})^2. \quad (1)$$

Здесь I_{xx}, I_{yy} – частные производные второго порядка по переменным x и y соответственно, I_{xy} – смешанная частная производная второго порядка функции яркости изображения $I(x, y)$. Для получения оценок названных частных производных как один из возможных вариантов используются матричные маски A_{xx}, A_{yy}, A_{xy} :

$$A_{xx} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & -2 & -2 & -2 & 1 & 1 & 1 \\ 1 & 1 & 1 & -2 & -2 & -2 & 1 & 1 & 1 \\ 1 & 1 & 1 & -2 & -2 & -2 & 1 & 1 & 1 \\ 1 & 1 & 1 & -2 & -2 & -2 & 1 & 1 & 1 \\ 1 & 1 & 1 & -2 & -2 & -2 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad A_{yy} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & -1 & -1 & -1 & 0 \\ 0 & 1 & 1 & 1 & 0 & -1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & 0 & -1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & -1 & -1 & 0 & 1 & 1 & 1 & 1 \\ 0 & -1 & -1 & -1 & 0 & 1 & 1 & 1 & 1 \\ 0 & -1 & -1 & -1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Маска A_{yy} получается транспонированием матрицы A_{xx} . Значение гессиана $\det H(I)$ с использованием приведенных масок находится как свертка

$$\det H(I) = (A_{xx} * I) \cdot (A_{yy} * I) - (0,9 \cdot A_{xy} * I)^2. \quad (2)$$

Используемая на практике формула (2) отличается от (1) наличием корректировочного коэффициента в виде множителя 0,9, уменьшающего вклад в оценку гессиана смешанной частной производной второго порядка.

Гессиан инвариантен к изменениям яркости изображения и повороту, но не инвариантен к масштабу [10–12]. Для учета эффекта масштаба используются фильтры (маски) различных размеров для оценки гессиана. Для уменьшения числа используемых фильтров вводятся октавы. В каждой октаве размер фильтра изменяется с определенным шагом. При этом размеры фильтров в соседних октавах перекрываются. Пиксель считается ключевым, если гессиана имеет в нем экстремум как по отношению к соседним пикселям данной октавы, так и в соседних с ним нижней и верхней октавах того же размера (3×3). Для ускорения вычисления сумм яркостей пикселей в соответствующей прямоугольной области изображения используется интегральное представление изображений [9].

На втором этапе метода SURF происходит вычисление дескрипторов. Для формирования дескриптора используются анализируемое изображение и набор ключевых точек, которые были определены на исходном изображении на первом этапе работы алгоритма. Результатом работы дескриптора является набор векторов признаков. Вектор признаков ключевой точки (пикселя) (i_0, j_0) представляет собой набор функций (признаков) $f_k(w(i_0, j_0))$, $k = \overline{1, K}$, описывающих характерные особенности данной ключевой точки в ее окрестности $w(i_0, j_0)$. Эти признаки формируются на основе информации об интенсивности, цвете и текстуре данной точки.

Дескриптор каждой ключевой точки в методе SURF содержит 64 (в других вариантах 128 чисел) [10]. Основная задача дескриптора – определить направление, в котором модуль гра-

диента $|\nabla I| = \left(\left(\frac{\partial I}{\partial x} \right)^2 + \left(\frac{\partial I}{\partial y} \right)^2 \right)^{1/2}$ функции яркости принимает максимальное значение.

Метод **KAZE** также является улучшенной версией SIFT. Он разработан Пабло Фернандесом Алькантариллом и его коллегами [13] для улучшения точности и устойчивости к различным видам временных и геометрических трансформаций (поворот, масштабирование, освещение, шум и размытие), а также для работы с изображениями различных структур и фонов. KAZE использует адаптивный детектор порога для поиска локальных максимумов значения гессиана по пространству и масштабу. Чтобы избежать размытия краев и потери деталей в пространстве линейного масштаба, построенном с помощью фильтра Гаусса, создается пространство нелинейного масштаба с помощью нелинейного диффузионного фильтра. Пространство нелинейного масштаба строится с использованием методов AOS (аддитивное операторное расщепление) и диффузии переменной проводимости. Для выявления важных точек вычисляется нормализованное значение определителя гессиана на различных уровнях масштаба:

$$\det H(I) = \sigma^2(I_{xx} \cdot I_{yy} - I_{xy}^2).$$

Здесь (I_{xx}, I_{yy}) – производные второго порядка по горизонтали и вертикали соответственно, а I_{xy} – смешанная производная второго порядка. Каждый экстремум ищется в прямоугольном окне размером $\sigma_i \times \sigma_i$ для текущего i , верхнего $i+1$ и нижнего $i-1$ отфильтрованных изображений. Для ускорения поиска экстремумов результаты вычислений проверяются сначала в окне размером 3×3 пикселя, чтобы быстрее отбросить немаксимумные ответы. Наконец, положение ключевой точки оценивается с субпиксельной точностью с использованием метода, предложенного в [14].

Набор производных первого и второго порядка аппроксимируется с помощью фильтров Шарпа 3×3 с различными размерами шага производной σ_i . Производные второго порядка аппроксимируются с помощью последовательных фильтров Шарпа [13].

Использование двоичного дескриптора позволяет значительно ускорить описание признаков в нелинейном масштабном пространстве [15]. Дескриптор KAZE инвариантен к масштабу и вращению и требует мало места для хранения.

Метод **BRISK: Binary Robust Invariant Scalable Keypoints** объединяет методы поиска и описания ключевых точек. Для поиска он использует продвинутый алгоритм FAST и производит расчет максимумов не только в плоскости изображения, но и в масштабном пространстве [16].

Пирамида масштабного пространства строится путем понижающей дискретизации исходного изображения c_0 с использованием 4 октав c_i и 4 внутриоктав d_i , каждая из которых располагается между слоями c_i и c_{i+1} (показано на рисунке 4, заимствован из [16]). Детекторы FAST 9 – 16 применяются на каждой октаве и внутриоктаве отдельно для выявления потенциальных ключевых точек. Для достижения инвариантности к масштабу предлагается выбирать точку с максимальным значением интенсивности.

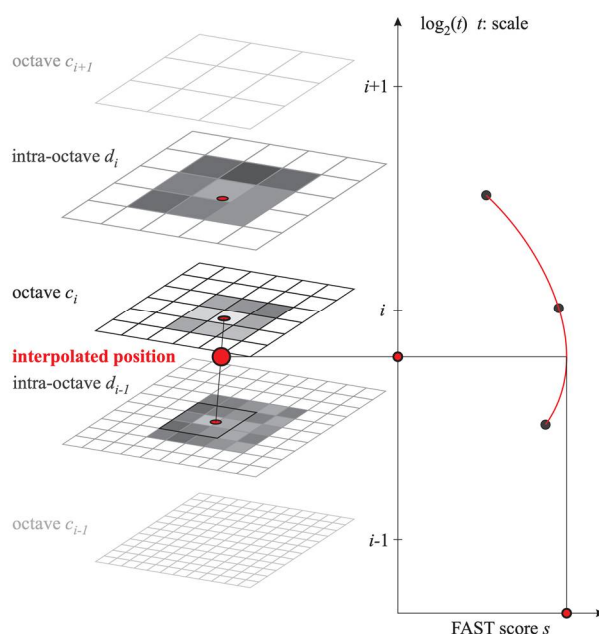


Рисунок 4 – Ключевая точка идентифицируется в октаве путем сравнения 8 пикселей окрестности c_i , а также соответствующих участков соседних слоев выше и ниже
Figure 4 – Keypoint is identified in the octave by comparing 8 pixels of neighborhood c_i as well as corresponding areas of neighboring layers above and below

Метод **ORB (Oriented FAST and Rotated BRIEF Features)** является комбинацией детектора FAST и улучшенного дескриптора BRIEF. Имеет более высокую производительность за счет использования бинарного дескриптора и потому может использоваться в системах технического зрения реального времени [17]. В модифицированной версии детектора FAST, используемой в ORB, алгоритм оценивает центральный пиксель плюс 12 пикселей внутри кольца радиусом 3 пикселя. Точка считается ключевой, если найдется набор 9 или больше пикселей, все яркостные различия между которыми больше или меньше определенного порога. Улучшение BRIEF заключается в добавлении инвариантности к ориентации в дескрипторе.

Время сопоставления ключевых точек в методе ORB существенно выше по сравнению с другими методами, что делает его малоприменимым для обработки изображений в реальном времени.

Экспериментальные исследования

Сравнение исследуемых методов проводилось на основе данных, полученных экспериментальным путем с использованием платформы программирования и числовых вычислений MATLAB (версия 23.2.0.2485118 (R2023b) Update 6) и расширения Computer Vision Toolbox. Вычисления проводились на компьютере Apple MacBook Pro с процессором Apple M1 Max и 32ГБ общей оперативной и видеопамати. Операционная система – macOS Sonoma версии 14.4.1 (23E224).

В исследовании использовался открытый набор данных Oxford [18], в состав которого входит 8 групп изображений, каждая из которых подвержена одному из типов преобразований, среди которых: поворот, масштабирование, изменение угла съемки, размытие, изменение освещения и сжатие. Каждая группа содержит по 6 изображений с разной степенью искажений. В таблице 1 приведены соответствия названий групп и типов преобразований, которым подвержены изображения в них.

Таблица 1 – Группы набора данных Oxford

Table 1 – Oxford dataset groups

Название группы	Тип преобразования
graff	Изменение угла съемки
wall	Изменение угла съемки
boat	Масштабирование и вращение
bark	Масштабирование и вращение
bikes	Размытие
trees	Размытие
ubc	Сжатие (наличие шумов)
leuven	Изменение освещения

Все сравниваемые алгоритмы входят в состав расширения Computer Vision Toolbox в виде отдельных функций, и их вызов происходил с настройками по умолчанию.

Таблица 2 – Среднее количество обнаруженных ключевых точек

Table 2 – Average number of detected keypoints

Название группы	SIFT	SURF	FAST	BRISK	KAZE	ORB
graff	3247	1775	1123	3249	8084	13868
wall	8961	2308	3051	4742	13682	47618
boat	5360	1907	2768	5389	8593	27397
bark	5708	1817	1322	2580	8660	26150
bikes	1197	610	150	545	4962	4543
trees	7922	3355	4563	7873	14486	44549
ubc	4848	1610	2220	3937	8460	25946
leuven	1605	639	756	1335	4486	7276

Оценка результатов работы алгоритмов обнаружения ключевых точек проводилась по показателю количества найденных точек и времени обработки одного изображения. В таблице 2 представлены усредненные результаты по количеству обнаруженных ключевых точек с разбивкой по группам изображений.

Поскольку скорость обработки изображений не зависит от типа изображения, то результаты по этому показателю можно сравнивать по среднему значению без разбивки на группы (рисунок 5).

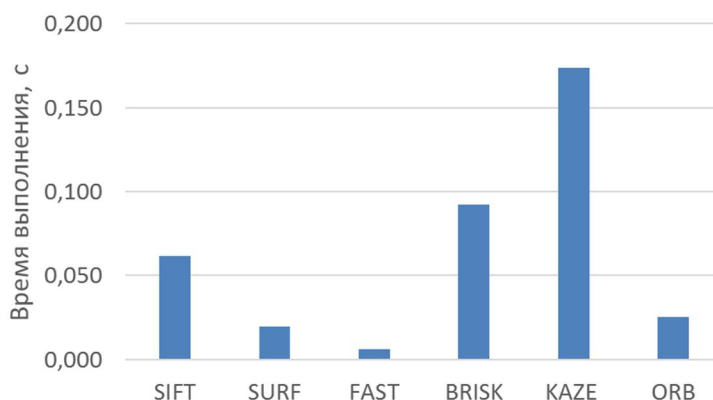


Рисунок 5 – Среднее время работы алгоритмов обнаружения ключевых точек для обработки одного изображения

Figure 5 – Average operating time of keypoint detection algorithms for processing one image

Из таблицы 2 видно, что алгоритмы SURF и FAST в среднем обнаружили меньше ключевых точек, чем остальные алгоритмы. При этом алгоритмы SIFT и BRISK выделили примерно в два раза больше ключевых точек. Заметно большее количество (в среднем в 3–10 раз) ключевых точек обнаружено алгоритмом ORB.

Эффективность метода нельзя однозначно оценивать исключительно по количеству обнаруженных алгоритмом ключевых точек. Однако эти данные позволяют понять, насколько по-разному работают алгоритмы. Также стоит отметить, что алгоритм FAST хуже всего справился с группой *bikes* и для нескольких изображений с наибольшим размытием в группе не обнаружил ключевых точек вовсе. В то же время остальные алгоритмы, хотя и обнаружили значительно меньше ключевых точек, но все-таки дали достаточный результат для сопоставления изображений. Аналогичное соотношение обнаруженных ключевых точек и уровня размытия изображения наблюдается и в группе *trees*, однако в этом случае все алгоритмы обнаружили некоторое количество точек.

При этом время работы алгоритмов значительно отличается, и здесь можно объективно выделить лидера. Метод FAST показал наилучшие результаты, в среднем потратив на обработку одного изображения 6 миллисекунд. Хорошие результаты также продемонстрировали алгоритмы SURF и ORB со средним временем 20 и 25 миллисекунд соответственно. Остальные алгоритмы затратили значительно больше времени на вычисления, и наихудшее время показал KAZE – 174 миллисекунды.

Для оценки эффективности дескрипторов использовалось время работы алгоритма (рисунок 6), время сопоставления ключевых точек двух изображений (рисунок 7) и показатель корректности сопоставления «Inlier Ratio» (рисунок 8).

Из рисунка 6 следует, что затраты среднего времени работы алгоритмов-дескрипторов аналогичны изменениям времени работы детекторов. KAZE снова выполняется дольше остальных алгоритмов, а FREAK (дескриптор для точек, обнаруженных методом FAST), SURF и ORB демонстрируют максимальную скорость выполнения.

Однако время, затраченное на сопоставление, нивелирует преимущества в скорости обнаружения и описания ключевых точек метода ORB. По итогам эксперимента, включавшего попарное сопоставление каждого из изображений внутри группы, среднее

время сопоставления этим методом составило 1,33 секунды, что в десятки раз дольше по сравнению с остальными методами. Вероятнее всего, это связано с большим количеством сопоставляемых точек.

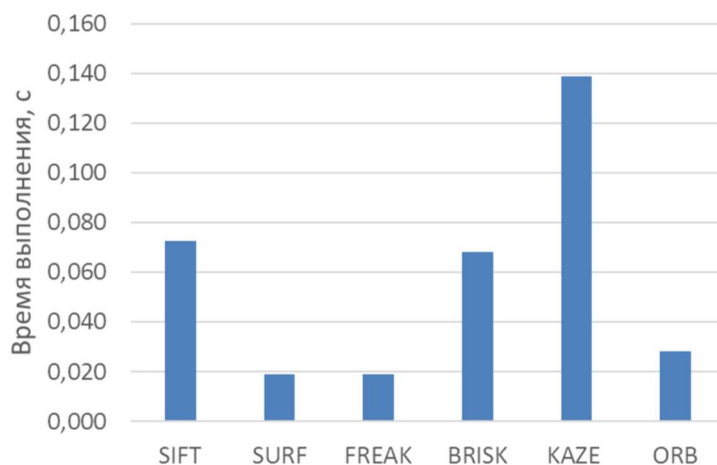


Рисунок 6 – Среднее время работы алгоритмов описания ключевых точек для обработки одного изображения

Figure 6 – Average operating time of keypoints describing algorithms for processing one image

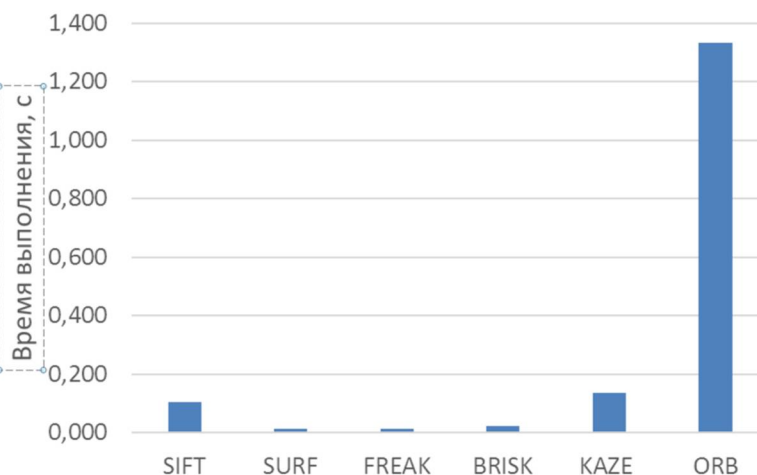


Рисунок 7 – Среднее время сопоставления ключевых точек двух изображений

Figure 7 – Average time of matching keypoints of two images

С точки зрения времени сопоставления лучшие результаты показали SURF, FREAK и BRISK. Их время сопоставления заняло 12, 13 и 24 миллисекунды соответственно.

Для определения качества обнаруженных и описанных ключевых точек использовался параметр Inlier Ratio, алгоритм определения которого заключается в следующих шагах.

1. Вычисление фундаментальной матрицы с использованием сопоставленных точек.
2. Вычисление эпполярных линий для каждого из двух изображений.
3. Поиск ближайших точек на эпполярных линиях к точкам изображения.
4. Оценка расстояния между рассчитанными ближайшими точками и изначальными парами точек на изображениях.
5. Определение порогового расстояния, меньше которого можно считать сопоставление точек верным.
6. Подсчет количества «inliers» – пар точек, расстояние между которыми меньше порога.
7. Вычисление соотношения количества «inliers» и общего числа пар точек.

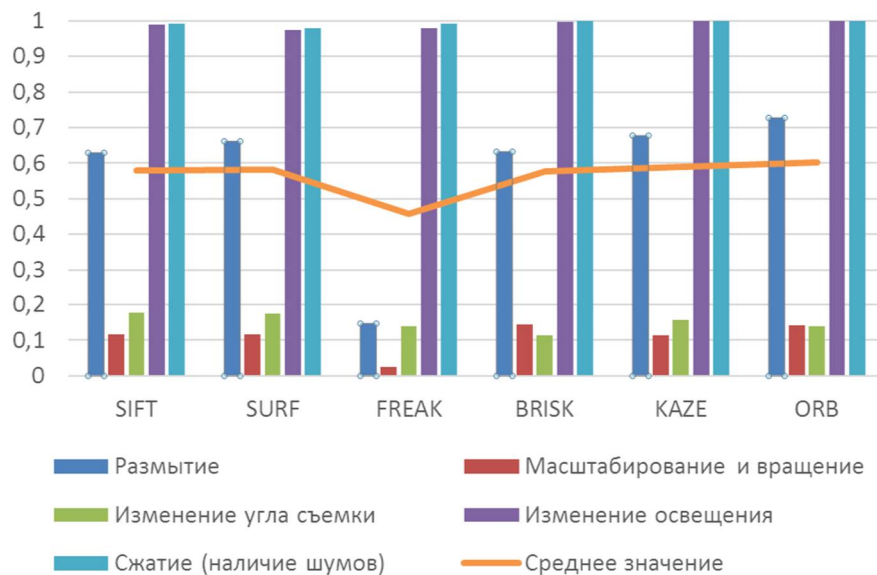


Рисунок 8 – Средние значения параметра Inlier Ratio, сгруппированные по типам преобразований изображений

Figure 8 – Average Inlier Ratio values grouped by image transformation type

Чем ближе значение Inlier Ratio к единице, тем более точным является сопоставление.

Из рисунка 8 следует, что все алгоритмы корректно справились с задачами обработки изображений, подверженных сжатию и изменению угла освещения. Также наблюдается высокое значение корректности обработки изображений с размытием. Причем стоит заметить, что все алгоритмы, кроме FREAK справились с размытыми изображениями на высоком уровне, испытывая сложности только с максимальным уровнем размытия. Тогда как FREAK смог корректно сопоставить только изображения с минимальным размытием.

Аналогичное отставание алгоритма FREAK наблюдается и при работе с изображениями, подверженными масштабированию и вращению. Остальные алгоритмы продемонстрировали результат на стабильно равном уровне.

По результатам эксперимента видно, что обработка изображений, подверженных аффинным преобразованиям, является более сложной задачей для всех исследуемых алгоритмов. Это говорит о том, что метод сопоставления изображений с применением ключевых точек имеет более высокую эффективность при незначительных преобразованиях данного типа, а именно при обработке кадров видеопотока или аэрофотосъемки.

Заключение

Выполнены описание, анализ и обобщение преимуществ и недостатков методов обнаружения и описания ключевых точек. Проведенный анализ позволяет учитывать полученные результаты при выборе метода для реальных приложений. Рассмотренные алгоритмы обнаружения и описания ключевых точек являются стандартом в области компьютерного зрения при решении задач сопоставления изображений. Проведенный эксперимент позволил выявить ряд особенностей каждого из алгоритмов.

1. SIFT. Показывает средний результат по показателям времени работы детектора и дескриптора, а также по времени сопоставления изображений. При этом демонстрирует хорошие результаты при обработке всех типов изображений.

2. SURF. При схожих показателях точности при сопоставлении изображений демонстрирует лучшие результаты по скорости работы.

3. FAST/FREAK. Алгоритм обнаружения ключевых точек FAST является самым быстрым алгоритмом среди исследуемых. Дескриптор FREAK также демонстрирует высокие показатели в скорости описания и сопоставления. Однако в задачах обработки размытых изображений или подверженных масштабированию и вращению, данная связка показала значи-

тельно меньшую эффективность. Следовательно данный метод имеет ряд ограничений, но за счет высокой скорости обработки изображений может применяться при необходимости обработки изображений в реальном времени.

4. BRISK. При стабильно высокой точности сопоставления, время работы алгоритма на всех этапах выше, чем у метода SURF.

5. KAZE. Данный метод аналогично BRISK показал еще более высокое время обработки изображений при незначительно более высоком уровне точности для всех групп изображений.

6. ORB. Является наиболее стабильным и точным алгоритмом в среднем. Время обнаружения и описания ключевых точек также находится на высоком уровне. Однако время сопоставления ключевых точек многократно выше, чем у конкурентов, что исключает возможность обработки изображений в реальном времени с использованием этого алгоритма.

Библиографический список

1. **Nancy Xu, Giannis Nikolentzos, Michalis Vazirgiannis, and Henrik Boström**, «Image Keypoint Matching using Graph Neural Networks», *Complex Networks & Their Applications X* (pp.441-451). DOI:10.1007/978-3-030-93413-2_37.

2. **Novikov A.I., Sablina V.A., Efimov A.I., Nikiforov M.B.** Contour Analysis in the tasks of real and virtual images superimposition // *Journal Coupled Systems and Multiscale Dynamics*, vol 4(4), 2016. p p. 251-259. (Doi: 10.1166/jcsmd. 2016.1112 J. Coupled Syst. Multiscale Dyn. vol 4(4)/ 2330-152X/2016/251/009).

3. **Новиков А.И., Пронькин А.В.** Методы цифровой обработки изображений подстилающей поверхности. – М.: Горячая линия- Телеком, 2023. – 224 с.

4. **Ивашечкин А.П., Василенко А.Ю., Гончаров Б.Д.**, Методы нахождения особых точек изображения и их дескрипторов, *Международный научный журнал «Молодой учёный»* № 15 (119) / 2016.

5. **Mikhaylov A.V., Kovalenko P.O., Kataev A.V., Rozaliev V.L.**, «Methods for Searching Key Points in Images», IV International research conference «Information technologies in Science, Management, Social sphere and Medicine». DOI:10.2991/itsmssm-17.2017.47.

6. **Новиков А.И., Ефимов А.И., Колчаев Д.А.** Алгоритмы автоматической идентификации объектов на изображениях и совмещения изображений // IV Международная конференция и молодёжная школа «Информационные технологии и нанотехнологии» (ИТНТ-2018), Самара. С. 651-654.

7. **Rosten, Edward; Drummond, Tom** (2006). «Machine Learning for High-speed Corner Detection». *Computer Vision – ECCV 2006. Lecture Notes in Computer Science*. vol. 3951. pp. 430-443. doi:10.1007/11744023_34. ISBN 978-3-540-33832-1. S2CID 1388140.

8. **Alahi A, Raphael O., Pierre V.** «FREAK: fast Retina keypoint», in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 510-517, IEEE, Providence, RI, USA, June 2012. DOI:10.1155/2021/8509164.

9. **Lowe D.G.** Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision* 60, 91–110 (2004). <https://doi.org/10.1023/B:VISI.0000029664.99615.94>.

10. **Bay H., Tinne T., Luc V. G.**, «SURF: speeded up robust features», *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008. DOI:10.1007/11744023_32.

11. **Mikolajczyk K., Schmid C.** A performance evaluation of local descriptors. *IEEE transactions on pattern analysis and machine intelligence*, pages 1615-1630, 2005.

12. **Дышлюк В.О.** Исследование показателей качества и быстродействия поиска опорных точек на изображениях методом SURF. *Молодой ученый*, 2018. № 27, с. 23-26.

13. **Alcantarilla P.F., Bartoli A., Davison A.J.**, KAZE Features, *ECCV 2012*, Springer, vol. 7577, London, UK, 2012. DOI:10.1007/978-3-642-33783-3_16.

14. **Brown M., Lowe D.**: Invariant features from interest point groups. In: *British Machine Vision Conf., BMVC*, Cardiff, UK (2002). DOI:10.5244/C.16.23.

15. **Alcantarilla P.F., Nuevo J, Bartoli A.A.**, «Fast Explicit Diffusion for Accelerated Features in Non-linear Scale Spaces», *British Machine Vision Conference (BMVC)At: Bristol, UK* (2013). DOI:10.5244/C.27.13.

16. **Leutenegger S., Chli M., Siegwart R.Y.**, «BRISK: binary robust invariant scalable keypoints», in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 2548-2555, IEEE, Barcelona, Spain, November 2011. DOI:10.1109/ICCV.2011.6126542.

17. Rublee E., Rabaud V., Konolige K., Bradski G., «ORB: an efficient alternative to SIFT or SURF», in Proceedings of the 2011 International Conference on Computer Vision, pp. 2564-2571, IEEE, Barcelona, Spain, November 2011. DOI:10.1109/ICCV.2011.6126544.

18. Oxford Image Matching Dataset [Электронный ресурс]. URL: https://cvssp.org/featurespace/web/related_papers/graffiti.html (дата обращения: 22.04.2024).

UDC 004.932.2

ANALYSIS OF IMAGE KEY POINT DETECTION, DESCRIPTION AND MATCHING METHODS

A. A. Denisov, post-graduate student, RSREU, Ryazan, Russia;
orcid.org/0009-0006-6765-5054, e-mail: denal777@gmail.com

A. I. Novikov, Dr. Sc.(Tech.), full professor, RSREU, Ryazan, Russia;
orcid.org/0000-0002-8166-8234, e-mail: novikovanatoly@yandex.ru

The aim of the work is to describe and conduct comparative tests of known detecting and describing image key points methods and algorithms, as well as to establish pairwise correspondence between key points of two images. The objective of the work is to identify the strengths and weaknesses of each of the algorithms. The following algorithms for detecting and describing keypoints were compared in the experiments: SIFT, SURF, FAST/FREAK, BRISK, KAZE and ORB. The comparison of keypoint detectors was carried out according to the criteria of performance and the number of keypoints found. The descriptors were compared in terms of accuracy to establish a correspondence between key points of two images and computational time for key points describing and matching them.

Keywords: keypoint, descriptor, detector, SIFT, SURF, FAST, FREAK, BRISK, KAZE, ORB ORB.

DOI: 10.21667/1995-4565-2024-89-104-116

References

1. **Nancy Xu, Giannis Nikolentzos, Michalis Vazirgiannis, and Henrik Boström**, «Image Keypoint Matching using Graph Neural Networks», Complex Networks & Their Applications X (pp.441-451). DOI:10.1007/978-3-030-93413-2_37.

2. **Anatoly I. Novikov, Victoria A. Sablina, Aleksey I. Efimov, Michael B. Nikiforov**. Contour Analysis in the tasks of real and virtual images superimposition. *Journal Coupled Systems and Multiscale Dynamics*, vol. 4(4), 2016, pp. 251-259. (Doi: 10.1166/jcsmd. 2016.1112 J. Coupled Syst. Multiscale Dyn. vol 4(4)/2330-152X/2016/251/009).

3. **Novikov A.I., Pron'kin A.V.** Metody cifrovoj obrabotki izobrazhenij podstilayushchej poverhnosti. M.: Goryachaya liniya- Telekom, 2023. - 224 p. (in Russian).

4. **Ivashechkin A.P., Vasilenko A.Yu., Goncharov B.D.**, Metody nahozhdeniya osobyh toчек izobrazheniya i ih deskriptorov. *Mezhdunarodnyj nauchnyj zhurnal "Molodoj uchyonyj"*, no. 15 (119), 2016. (in Russian).

5. **Mikhaylov A.V., Kovalenko P.O., Kataev A.V., Rozaliev V.L.**, «Methods for Searching Key Points in Images», *IV International research conference «Information technologies in Science, Management, Social sphere and Medicine»*. DOI:10.2991/itsmssm-17.2017.47.

6. **Novikov A.I., Efimov A.I., Kolchaev D.A.** Algoritmy avtomaticheskoy identifikacii ob"ektov na izobrazheniyah i sovmeshcheniya izobrazhenij. *IV Mezhdunarodnaya konferenciya i molodyozhnaya shkola «Informacionnye tekhnologii i nanotekhnologii» (ITNT-2018)*, Samara, pp. 651-654. (in Russian).

7. **Rosten, Edward; Drummond, Tom** (2006). «Machine Learning for High-speed Corner Detection». *Computer Vision – ECCV 2006. Lecture Notes in Computer Science*, vol. 3951, pp. 430-443. doi:10.1007/11744023_34. ISBN 978-3-540-33832-1. S2CID 1388140.

8. **Alahi A., Raphael, O., Pierre V.**, «FREAK: fast Retina keypoint» in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 510-517, IEEE, Providence, RI, USA, June 2012. DOI:10.1155/2021/8509164.

9. **Lowe, D.G.** Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision* 60, 91–110 (2004). <https://doi.org/10.1023/B:VISI.0000029664.99615.94>.
10. **Bay H., Tinne T., Luc V.G.**, «SURF: speeded up robust features», *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346-359, 2008. DOI:10.1007/11744023_32.
11. **Mikolajczyk K., Schmid C.** A performance evaluation of local descriptors. *IEEE transactions on pattern analysis and machine intelligence*, pages 1615–1630, 2005.
12. **Dyslyuhk V.O.** Issledovanie pokazatelej kachestva i bystrodejstviya poiska opornyh toчек na izobrazheniyah metodom SURF. *Molodoj uchenyj*, 2018, no. 27, pp. 23-26. (in Russian).
13. **Alcantarilla P.F., Bartoli A.A., Davison J.**, KAZE Features, *ECCV 2012*, Springer, vol. 7577, London, UK, 2012. DOI:10.1007/978-3-642-33783-3_16.
14. **Brown M., Lowe D.**: Invariant features from interest point groups. In: *British Machine Vision Conf., BMVC*, Cardiff, UK (2002). DOI:10.5244/C.16.23.
15. **Alcantarilla P.F., Nuevo J., Bartoli A.A.**, «Fast Explicit Diffusion for Accelerated Features in Non-linear Scale Spaces», *British Machine Vision Conference (BMVC)* At: Bristol, UK (2013). DOI:10.5244/C.27.13.
16. **Leutenegge S., Chli M., Siegwart R.Y.**. «BRISK: binary robust invariant scalable keypoints» in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 2548-2555, IEEE, Barcelona, Spain, November 2011. DOI:10.1109/ICCV.2011.6126542.
17. **Rublee E., Rabaud V., Konolige K., Bradski G.**, «ORB: an efficient alternative to SIFT or SURF» in *Proceedings of the 2011 International Conference on Computer Vision*, pp. 2564-2571, IEEE, Barcelona, Spain, November 2011. DOI:10.1109/ICCV.2011.6126544.
18. Oxford Image Matching Dataset [Elektronnyj resurs]. URL: https://cvssp.org/featurespace/web/related_papers/graffiti.html (data obrashcheniya: 22.04.2024).