

Сравнительный анализ методов поиска ключевых точек для распознавания лица

А.П. Семёнова, А.В. Левкина, Е.В. Радевич, Е.О. Сухорукова

Донецкий национальный технический университет

nastena-semenova19@rambler.ru, lewkina.anastasya@yandex.ru,
radevich_katerina@mail.ru

Аннотация

В данной статье проводится анализ методов сопоставления изображений, основанных на ключевых точках, таких как SIFT, SURF, FAST/FREAK, BRISK, KAZE и ORB. Рассматриваются их преимущества и недостатки, а также эффективность в различных условиях, включая обработку изображений. Эксперименты показывают, что алгоритмы успешно справляются с размытием и аффинными преобразованиями. Выбор метода критически важен для достижения высоких результатов в задачах компьютерного зрения, что подчеркивает необходимость учета полученных результатов при выборе алгоритмов для реальных приложений.

Введение

Сопоставление изображений является ключевым аспектом множества задач в области компьютерного зрения, включая восстановление трехмерных структур, SLAM, обнаружение изменений, совмещение изображений, робототехнику, распознавание лиц и эмоций [1-4]. Для решения этой задачи часто применяются ключевые точки, которые представляют собой пиксели с уникальными характеристиками, позволяющими идентифицировать и классифицировать изображения [5-7].

Ключевые точки окружены областями с уникальными отличиями от соседних пикселей, что позволяет создавать их описания в виде векторов признаков или дескрипторов. Элементы дескриптора могут включать модули и направления градиентов, вычисляемых в окрестности ключевой точки. После обнаружения и описания ключевых точек данные могут быть использованы для сопоставления с ключевыми точками на других изображениях [7].

Существует множество алгоритмов для обнаружения и описания ключевых точек, каждый из которых имеет свои особенности и эффективность.

В данной работе рассматриваются следующие методы: SIFT (Scale Invariant Feature Transform), SURF (Speeded-Up Robust Features), FAST/FREAK (Features from Accelerated Segment Test / Fast Retina Keypoint), BRISK (Binary Robust Invariant Scalable Keypoints), KAZE и ORB (Oriented FAST and Rotated BRIEF Features). Все они доступны в расширении Computer Vision Toolbox для MATLAB.

Метод обнаружения ключевых точек FAST/FREAK

Метод FAST [8] разработанный Эдвардом Ростеном и Т. Драммондом в 2006 году, включает несколько этапов:

1) Определение порога интенсивности. На первом этапе устанавливается пороговое значение интенсивности пикселей изображения, которое используется для классификации пикселей как ключевых точек;

2) Выявление кандидатов на ключевые точки. Алгоритм анализирует интенсивность соседних пикселей (обычно 16 пикселей, расположенных по окружности вокруг анализируемого пикселя), чтобы определить, может ли данный пиксель стать потенциальной ключевой точкой;

3) Ускорение процесса обнаружения. Для повышения скорости обнаружения разработчики метода предложили сначала проверять только четыре пикселя, расположенные на осях O_x и O_y относительно потенциальной ключевой точки. Если хотя бы три из этих пикселей превышают установленный порог интенсивности, выполняется полная проверка по всем 16 точкам. В противном случае пиксель помечается как «не ключевой»;

4) Определение ключевых точек. Пиксель считается ключевой точкой, если он и все 16 его соседей либо ярче порога интенсивности, либо темнее его на заданное значение.

FAST отличается высокой вычислительной эффективностью и подходит для систем реального времени. Существуют усовершенствованные версии, такие как FAST-9, FAST-12, FAST-AGAST и FAST-ER.

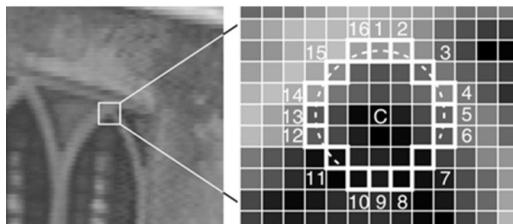


Рисунок 1 – Иллюстрация к детектору Fast

1) FAST-9 и FAST-12. Эти модификации используют 9 или 12 точек окружности вместо 16, что позволяет увеличить скорость выполнения алгоритма;

2) FAST-AGAST – это улучшенная версия метода, использующая алгоритм AGAST (Adaptive and Generic Accelerated Segment Test), который адаптирует пороговое значение для определения ключевых точек в зависимости от уровня яркости окружающих пикселей, что повышает устойчивость к изменениям освещения;

3) FAST-ER (Enhanced Repeatability). Данная версия разработана для повышения устойчивости к шумам и улучшения повторяемости обнаруженных ключевых точек. FAST-ER достигает этого за счет улучшения процесса детектирования с использованием дополнительных шагов, таких как фильтрация и выбор наилучших точек.

Поскольку метод FAST является исключительно детектором ключевых точек, для их описания необходимо использовать другие алгоритмы. Одним из часто применяемых дескрипторов является FREAK (Fast Retina Keypoint), разработанный Александром Алехи, Рафаэлем Ортизом и Пьером Вандергейном. FREAK предназначен для быстрого и компактного описания ключевых точек, что делает его подходящим для работы в реальном времени и с большими объемами данных.

Одним из часто применяемых дескрипторов является FREAK (Fast Retina Keypoint), разработанный Александром Алехи, Рафаэлем Ортизом и Пьером Вандергейном. FREAK предназначен для быстрого и компактного описания ключевых точек, что делает его подходящим для работы в реальном времени и с большими объемами данных. Дескриптор FREAK является бинарным и имеет длину 512 бит.

Данный метод описания ключевой точки основывается на особенностях зрительной системы человека, в частности, на структуре сетчатки глаза. FREAK использует сетчаточную сетку выборки с круглой формой (см. рис. 2), где плотность точек выше в центре. Для соответствия модели сетчатки алгоритм применяет различные размеры ядер (радиусы) для каждой точки выборки.

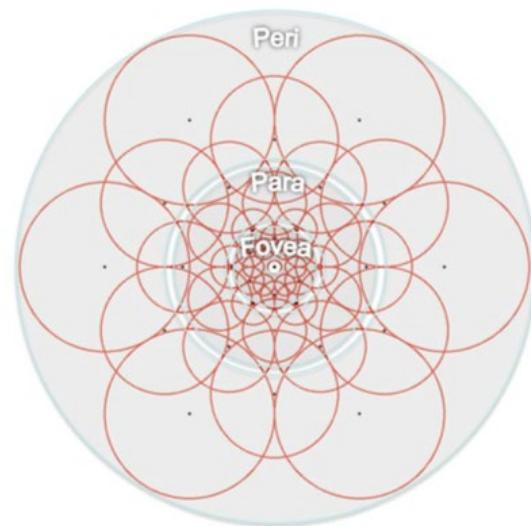


Рисунок 2 – Схема выборки фрагментов ключевой точки, имитирующая рецептивные структуры сетчатки

Уникальность метода заключается в экспоненциальном изменении размеров и перекрытии рецептивных полей. Каждый круг на иллюстрации представляет собой стандартные отклонения гауссовых ядер, примененных к соответствующим точкам выборки.

Метод обнаружения ключевых точек SIFT

Метод SIFT [9] решает проблемы, связанные с вращением, аффинными преобразованиями и изменением интенсивности. Ключевые точки обнаруживаются с помощью каскадной фильтрации и масштабирования пространства, что позволяет находить устойчивые местоположения ключевых точек.

Функция масштабирования является сверткой гауссовой функции переменного масштаба $L(x, y, \sigma)$ с входным изображением $I(x, y)$:

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

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

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) = L(x, y, \sigma) - L(x, y, \sigma). \quad (2)$$

При этом каждая точка выборки сравнивается с восьмью своими соседями на текущем изображении и девятью соседями в шкале сверху и снизу (рис. 3, заимствован из [9]).

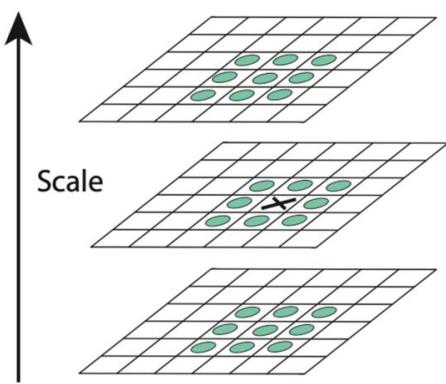


Рисунок 3 – Визуализация сравнения значений ключевой точки с соседними точками в областях 3×3 в текущем и соседних масштабах.

Ключевой точка считается только в том случае, если ее значение больше всех этих соседей или меньше их всех.

Гистограмма ориентации, состоящая из 36 интервалов, охватывающих полный круг в 360 градусов, формируется на основе направления градиента в окрестности ключевой точки. В итоге локальная область вокруг ключевой точки делится на 16 подрегионов размером 4×4 , при этом в каждом подрегионе содержится по 8 значений гистограммы ориентации. Таким образом, вектор дескриптора ключевой точки включает 128 значений признаков (16×8).

Метод обнаружения ключевых точек SURF

Метод SURF [10], разработанный Гербертом Бейем, Андреасом Эссем, Тинне Туйтелаарс и Люком Ван Гулом, является усовершенствованной версией SIFT, предлагая более быстрый и эффективный подход к обнаружению ключевых точек с использованием интегрального представления изображений.

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

$$\det H(I) = I_{xx} \cdot I_{yy} - (I_{xy})^2, \quad (3)$$

где 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} :

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

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

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

$$\det H(I) = (A_{xx} * I) \cdot (A_{yy} * I) - (0,9 \cdot A_{xy} * I)^2. \quad (4)$$

Формула, применяемая на практике (4), отличается от (1) наличием корректировочного коэффициента в виде множителя 0,9, который снижает вклад в оценку гессиана смешанной второй производной. Гессиан сохраняет инвариантность к изменениям яркости изображения и поворотам, однако не является инвариантным к изменениям масштаба. Для учета масштабного эффекта применяются фильтры (маски) различных размеров для оценки гессиана. Чтобы сократить количество используемых фильтров, вводятся октавы, в каждой из которых размер фильтра изменяется с определенным шагом, при этом размеры фильтров в соседних октавах перекрываются. Пиксель считается ключевым, если гессиан в нем имеет экстремум как по отношению к соседним пикселям в данной октаве, так и в соседних нижней и верхней октавах того же размера (3×3). Для ускорения вычислений сумм яркостей пикселей в соответствующей прямоугольной области изображения используется интегральное представление изображений.

На втором этапе метода SURF происходит вычисление дескрипторов. Для формирования дескриптора используются анализируемое изображение и набор ключевых точек, определенных на исходном изображении в 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) $f_k(w(i_0, j_0))$, $k = 1, K$, описывающих характерные особенности данной ключевой точки в её 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 градиента $|\nabla I| = \left(\left(\frac{\partial I}{\partial x} \right)^2 + \left(\frac{\partial I}{\partial y} \right)^2 \right)^{1/2}$ функция яркости принимает максимальное значение.

Метод обнаружения ключевых точек KAZE

Метод KAZE [11] представляет собой усовершенствованную версию SIFT, разработанную Пабло Фернандесом Алькантариллом и его коллегами. Этот метод направлен на повышение точности и устойчивости к различным видам временных и геометрических трансформаций, таким как поворот, масштабирование, изменения освещения, шум и размытие, а также на работу с изображениями, имеющими различные структуры и фоны. KAZE применяет аддитивный детектор порога для поиска локальных максимумов значений гессиана в пространстве и масштабе.

Чтобы избежать размытия краев и потери деталей в пространстве линейного масштаба, созданного с использованием гауссовского фильтра, формируется пространство нелинейного масштаба с помощью нелинейного диффузионного фильтра. Это пространство строится с применением методов аддитивного операторного расщепления (AOS) и диффузии с переменной проводимостью. Для выявления значимых точек вычисляется нормализованное значение определителя гессиана на различных уровнях масштаба.

$$\det H(I) = \sigma^2 (I_{xx} \cdot I_{yy} - I_{xy}^2), \quad (5)$$

где 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 $i+1$, нижнего $i-1$, отфильтрованных изображений.

Для ускорения поиска экстремумов результаты вычислений проверяются сначала в окне размером 3×3 пикселя, чтобы быстрее

отбросить минимальные ответы.

В конечном итоге положение ключевой точки оценивается с субпиксельной точностью с использованием метода, предложенного в [14]. Набор производных первого и второго порядка аппроксимируется с помощью фильтров Шарра размером 3×3 с различными шагами производной σ_i . Производные второго порядка аппроксимируются с использованием последовательных фильтров Шарра [13]. Применение двоичного дескриптора значительно ускоряет процесс описания признаков в нелинейном масштабном пространстве. Дескриптор KAZE инвариантен к изменениям масштаба и вращению, а также требует минимального объема для хранения.

Метод обнаружения ключевых точек BRISK

Метод BRISK объединяет методы поиска и описания ключевых точек. Для поиска он использует усовершенствованный алгоритм FAST и вычисляет максимумы не только в плоскости изображения, но и в масштабном пространстве [12].

Пирамида масштабного пространства создается 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} (см. рис. 4, заимствованном из [12]). 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.

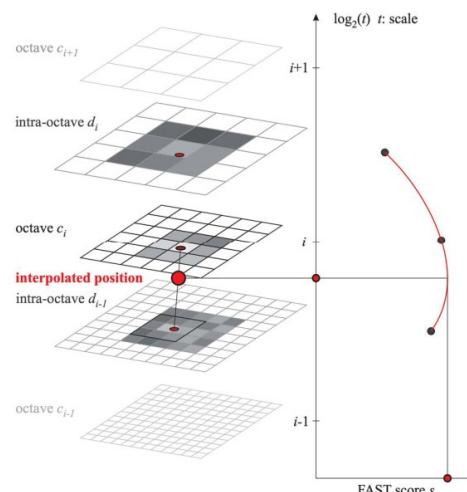


Рисунок 4 – Ключевая точка идентифицируется в октаве путем сравнения 8 пикселей окрестности c_i , а также соответствующих участков соседних слоев выше и ниже

Метод обнаружения ключевых точек ORB

Метод ORB [13] представляет собой комбинацию детектора FAST и усовершенствованного дескриптора BRIEF. Он демонстрирует высокую производительность благодаря использованию бинарного дескриптора, что делает его подходящим для применения в системах компьютерного зрения в реальном времени. В модифицированной версии детектора FAST, используемой в ORB, алгоритм анализирует центральный пиксель и 12 пикселей, расположенных в кольце радиусом 3 пикселя. Пиксель считается ключевым, если существует набор из 9 или более пикселей, все яркостные различия между которыми превышают или не достигают определенного порога. Улучшение BRIEF заключается в добавлении инвариантности к ориентации в дескрипторе. Однако время сопоставления ключевых точек в методе ORB значительно выше по сравнению с другими методами, что ограничивает его применение 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 выполнялись на компьютере Apple MacBook Pro с процессором Apple M1 Max и 32 ГБ общей оперативной и видеопамяти. Операционная система – macOS Sonoma версии 14.4.1 (23E224).

В исследовании использовался открытый набор данных Oxford, который включает 8 групп изображений, каждая из которых подвергнута одному из типов преобразований, таких как поворот, масштабирование, изменение угла съемки, размытие, изменение освещения и сжатие. Каждая группа содержит по 6 изображений с различной степенью искажений (см. табл. 1).

Таблица 1 - Группы набора данных Oxford.

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.

Название группы	SIRF	SURF	FAST	BRISK	KAZE	ORB
graff	3247	1775	1123	3249	8084	13868
wall	8961	2308	3051	4742	13652	47618
boat	5360	1907	2768	5389	8593	27397
bark	5708	1817	1322	2580	5660	26150
bikes	1197	610	150	545	4965	4543
trees	7922	3355	4563	7873	14486	44549
ubc	4848	1610	2220	3937	6460	25946
leuven	1605	639	756	13335	4486	7276

Оценка эффективности работы алгоритмов обнаружения ключевых точек проводилась на основе двух показателей: количества найденных ключевых точек и времени обработки одного изображения. В таблице 2 представлены усредненные результаты по количеству обнаруженных ключевых точек, сгруппированные по категориям изображений.

Поскольку скорость обработки изображений не зависит от их типа, результаты по этому показателю можно сравнивать в общем, без разбивки на группы (см. рис. 5).

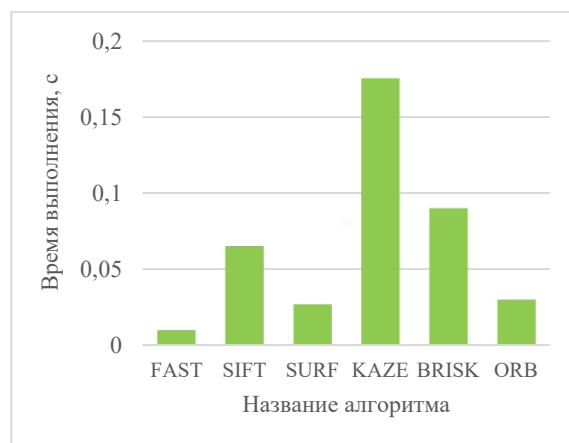


Рисунок 5 – Среднее время работы алгоритмов обнаружения ключевых точек для обработки одного изображения

Из таблицы 2 видно, что алгоритмы SURF и FAST в среднем обнаружили меньше ключевых точек по сравнению с другими методами. Алгоритмы SIFT и BRISK выявили примерно в два раза больше ключевых точек, а алгоритм ORB продемонстрировал значительно большее количество (в среднем в 3–10 раз) обнаруженных ключевых точек.

Однако эффективность метода не может быть оценена исключительно по количеству найденных ключевых точек. Эти данные помогают понять, как по-разному работают

алгоритмы. Следует отметить, что алгоритм FAST показал наихудшие результаты в группе "bikes" и не обнаружил ключевых точек на нескольких изображениях с сильным размытием. В то же время остальные алгоритмы, хотя и выявили значительно меньше ключевых точек, все же обеспечили достаточные результаты для сопоставления изображений. Похожая зависимость между количеством обнаруженных ключевых точек и уровнем размытия наблюдается и в группе "trees", хотя в этом случае все алгоритмы нашли некоторое количество точек.



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

Время работы алгоритмов значительно варьируется, и здесь можно выделить явного лидера. Метод FAST показал наилучшие результаты, в среднем затрачивая 6 миллисекунд на обработку одного изображения. Алгоритмы SURF и ORB также продемонстрировали хорошие результаты со средним временем 20 и 25 миллисекунд соответственно. Остальные алгоритмы потребовали значительно больше времени на вычисления, причем наихудший результат показал KAZE – 174 миллисекунды.

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

Из рисунка 6 видно, что среднее время работы алгоритмов-дескрипторов аналогично изменениям времени работы детекторов. KAZE снова выполняется дольше всех остальных алгоритмов, в то время как FREAK (дескриптор для точек, обнаруженных методом FAST), SURF и ORB показывают максимальную скорость выполнения.

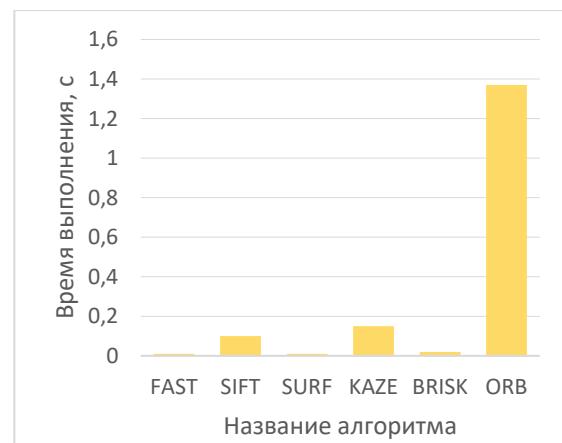


Рисунок 7 – Среднее время сопоставления ключевых точек для двух изображений

Тем не менее, время, затраченное на сопоставление, нивелирует преимущества в скорости обнаружения и описания ключевых точек метода ORB. В результате эксперимента, включавшего попарное сопоставление каждого изображения внутри группы, среднее время сопоставления с использованием этого метода составило 1,33 секунды, что в десятки раз больше по сравнению с другими методами. Это, вероятно, связано с большим количеством сопоставляемых точек.

С точки зрения времени сопоставления наилучшие результаты продемонстрировали алгоритмы SURF, FREAK и BRISK, затратившие 12, 13 и 24 миллисекунды соответственно.

Для оценки качества обнаруженных и описанных ключевых точек использовался параметр Inlier Ratio, алгоритм вычисления которого включает следующие шаги:

1. Вычисление фундаментальной матрицы на основе сопоставленных точек.
2. Определение эпиполярных линий для каждого из двух изображений.
3. Поиск ближайших точек на эпиполярных линиях к точкам изображения.
4. Оценка расстояния между найденными ближайшими точками и исходными парами точек на изображениях.
5. Установление порогового расстояния, ниже которого сопоставление точек считается корректным.
6. Подсчет количества «inliers» – пар точек, расстояние между которыми меньше установленного порога.
7. Вычисление соотношения количества «inliers» к общему числу пар точек.



Рисунок 8 – Качество работы алгоритма в зависимости от типа преобразований изображений губы



Рисунок 9 – Усредненный показатель качества работы алгоритма при искажениях на изображении

Чем ближе значение Inlier Ratio к единице, тем более точным считается сопоставление. Из рисунка 8 видно, что все алгоритмы успешно справились с задачами обработки изображений, подвергшихся сжатию и изменению угла освещения. Также наблюдается высокая корректность обработки изображений с размытием. При этом стоит отметить, что все алгоритмы, кроме FREAK, продемонстрировали высокий уровень работы с размытыми изображениями, испытывая трудности только с максимальным уровнем размытия. В то время как FREAK смог корректно сопоставить только изображения с минимальным размытием.

Аналогичное отставание алгоритма FREAK наблюдается и при работе с изображениями, подвергшимися масштабированию и вращению. Остальные алгоритмы показали стабильные результаты на равном уровне.

По итогам эксперимента видно, что обработка изображений, подвергнутых аффинным преобразованиям, представляет собой более сложную задачу для всех исследуемых алгоритмов. Это указывает на то, что метод сопоставления изображений с использованием

ключевых точек более эффективен при незначительных преобразованиях данного типа, например, при обработке кадров видеопотока или аэрофотосъемки.

Заключение

Подводя итог, можно заключить:

- алгоритмы SURF и FAST в среднем обнаружили меньше ключевых точек, чем SIFT и BRISK;

- ORB продемонстрировал значительно большее количество обнаруженных ключевых точек;

- метод FAST показал наилучшие результаты по времени обработки (6 мс), в то время как KAZE потребовал 174 мс;

- время сопоставления наилучших результатов показали SURF, FREAK и BRISK.

Проведенный анализ позволяет учитывать полученные результаты при выборе метода для реальных приложений. Рассмотренные алгоритмы являются стандартом в области компьютерного зрения. Эксперимент выявил особенности каждого алгоритма, что поможет в дальнейшем выборе подходящих методов для различных задач компьютерного зрения.

Литература

1. Семенова, А. П. Область применения алгоритма распознавания эмоций в информационных технологиях / А. П. Семенова, А. С. Миненко, Т. В. Ванжа // Цифровой регион: опыт, компетенции, проекты: сборник статей Международной научно-практической конференции (г. Брянск, 30 ноября 2018 г.) [Электронный ресурс]. – Брянск: Брян. гос. инженерно-технол. ун-т, 2018. – С. 443-446.

2. Миненко, А. С. Анализ эмоционального состояния человека по фотографическим изображениям / А. С. Миненко, А. П. Семенова // Информатика, управляющие системы, математическое и компьютерное моделирование (ИУСМКМ – 2019). Материалы X Международной научно-технической конференции в рамках V Международного Научного форума Донецкой Народной Республики «Инновационные перспективы Донбасса». – Донецк: ДонНТУ, 2019. – С. 123-126.

3. Левкина, А. В. Область применения систем распознавания эмоций / А. В. Левкина, Е. В. Радевич, А. П. Семёнова // Донбасс будущего глазами молодых ученых: сб. материалов науч.-техн. конф. для студ., асп. и мол. уч. – Донецк: ДонНТУ, 2024. – С. 158-162.

4. Семёнова, А. П. Анализ мимических выражений для задачи распознавания эмоций / А. П. Семёнова, В. Н. Павлыш // Проблемы искусственного интеллекта, 2020. - № 4 (19). - С. 69-79.

5. Миненко, А. С. Формальная модель эмоций / А. С. Миненко, А. П. Семенова // Проблемы искусственного интеллекта, 2018. - №3(10). - С. 84-93.
6. Семенова, А. П. Математическая модель эмоций // Международная научно-техническая конференция молодых ученых БГТУ им. В.Г. Шухова. Материалы национальной конференции с международным участием. – Белгород, 2019. – С. 4584-4587.
7. Семёнова, А. П. Поиск ключевых точек лица для задачи распознавания эмоций/ А. П. Семёнова, В. Н. Павлыши // Информатика и кибернетика, 2021. - № 1-2 (23-24). - С. 59-64.
8. Rosten, E. Machine learning for high-speed corner detection / E. Rosten, T. Drummond // Computer Vision – ECCV 2006. – PP. 430-443.
9. Lowe, D.G. Distinctive Image Features from Scale-Invariant Keypoints / D.G. Lowe // International Journal of Computer Vision, 2004. – № 60. – PP. 91-110.
10. Bay, H. SURF: speeded up robust features / H. Bay, T. Tinn, V.G. Luc // Computer Vision and Image Understanding, 2008. – № 3(110). – PP. 346-359.
11. Alcantarilla, P. F. KAZE Features / P. F. Alcantarilla, A. Bartoli, A. J. Davison // Computer Vision – ECCV 2012. – PP. 214-227.
12. Leutenegger, S. BRISK: binary robust invariant scalable keypoints / S. Leutenegger, M. Chli, R. Y. Siegwart // The IEEE International Conference on Computer Vision (ICCV). – November 2011, Barcelona, Spain. – PP. 2548-2555.
13. Rublee, E. ORB: an efficient alternative to SIFT or SURF / E. Rublee [et al.] // The IEEE International Conference on Computer Vision (ICCV). – November 2011, Barcelona, Spain. – PP. 2564-2571.
14. Brown, M. Invariant features from interest point groups / M. Brown, D. Lowe // In: British Machine Vision Conf., BMVC, Cardiff, UK (2002).

Семёнова А.П., Левкина А.В., Радевич Е.В., Сухорукова Е.О. Сравнительный анализ методов поиска ключевых точек для распознавания лица. В данной статье проводится анализ методов сопоставления изображений, основанных на ключевых точках, таких как SIFT, SURF, FAST/FREAK, BRISK, KAZE и ORB. Рассматриваются их преимущества и недостатки, а также эффективность в различных условиях, включая обработку изображений. Эксперименты показывают, что алгоритмы успешно справляются с размытием и аффинными преобразованиями. Выбор метода критически важен для достижения высоких результатов в задачах компьютерного зрения, что подчеркивает необходимость учета полученных результатов при выборе алгоритмов для реальных приложений.

Ключевые слова: ключевые точки лица, контурная модель, распознавание образов.

Semenova A.P., Levkina A.V., Radevich E.V., Sukhorukova E.O. Comparative analysis of key point search methods for face recognition. This article analyzes image matching methods based on key points such as SIFT, SURF, FAST/FREAK, BRISK, KAZE and ORB. Their advantages and disadvantages, as well as their effectiveness in various conditions, including image processing, are considered. Experiments show that algorithms successfully cope with blurring and affine transformations. The choice of method is critically important for achieving high results in computer vision tasks, which emphasizes the need to take into account the results obtained when choosing algorithms for real-world applications.

Key words: key points of the face, contour model, image recognition.

Статья поступила в редакцию 18.05.2025
Рекомендована к публикации профессором Павлышиом В. Н.