

УДК 004.932

Г.А. Буткин, И.А. Емельянов, А.В. Тузиков

АЛГОРИТМЫ ПОСТРОЕНИЯ ДЕСКРИПТОРОВ ЛОКАЛЬНЫХ ОСОБЕННОСТЕЙ ИЗОБРАЖЕНИЙ НА БАЗЕ МНОГОКОЛЬЦЕВЫХ НЕПАРАМЕТРИЧЕСКИХ ПРЕОБРАЗОВАНИЙ

Рассматривается применение непараметрических преобразований при построении дескрипторов локальных особенностей изображений. Предлагаются подходы, в основу которых положены алгоритмы построения масштабируемых дискретных окружностей по Брезенхему, для многокольцевых непараметрических преобразований. Приводятся результаты моделирования предложенного подхода к построению локальных дескрипторов на тестовой базе по сравнению с известными дескрипторами SURF, ORB, BRISK, FREAK.

Введение

При решении целого ряда задач обработки изображений используются дескрипторы локальных особенностей изображений, которые должны быть инвариантны к различного рода искажениям и обладать высоким быстродействием [1–6]. В этой связи представляет интерес рассмотрение дескрипторов и детекторов углов, использующих непараметрические преобразования, которые по своей природе являются инвариантными к изменениям уровня освещенности [7] и при некоторых конструкциях – робастными к шумам [8], а также характеризуются низкой трудоемкостью вычисления и простотой сопоставления. Однако традиционные непараметрические преобразования [7] являются неинвариантными к повороту. Впервые идея непараметрических преобразований, инвариантных к вращению, была предложена в работах [9, 10] при рассмотрении дескрипторов типа LBP (Local Binary Patterns) для описания текстур. В этом случае непараметрические преобразования локальных особенностей изображения рассматривались на окружностях различных радиусов с заданным числом фиксируемых с помощью билинейной аппроксимации точек. Предлагаемый подход состоит в построении дескриптора локальных особенностей изображений с использованием непараметрических преобразований на множестве колец, формируемых по алгоритму Брезенхема [11] с учетом дискретного пространства пикселей. В рассматриваемом дескрипторе сначала на двух кольцах с малыми радиусами осуществляется поиск характерных точек (углов) объектов изображения. Затем добавляются дополнительные кольца с возрастающими радиусами и на них путем использования непараметрических преобразований на задаваемых точках колец строится дескриптор найденных характерных точек в виде бинарного вектора.

1. Кольцевые непараметрические преобразования

Известно [7], что непараметрические преобразования оперируют не параметрами изображения, например интенсивностью (яркостью), а отношениями между параметрами. Предположим, что p является пикселем изображения, а $I(p)$ – его интенсивностью. Обозначим через $N(p^*)$ набор пикселей на некотором кольце Брезенхема радиусом r пикселей, окружающем пиксел p^* . Число пикселей в наборе $N(p^*)$ в принципе может быть любым и произвольно расположенным по кольцу, но не может превышать число пикселей в кольце Брезенхема заданного радиуса. Непараметрическое кольцевое преобразование основывается на установлении результатов сравнения интенсивности центрального пикселя p^* с интенсивностями пикселей из его окружения $N(p^*)$.

Если $\xi(p^*, p)$ является результатом сравнения пары пикселей, то можно принять

$$\xi(p^*, p) = \begin{cases} 1, & \text{если } I(p) > I(p^*); \\ 0, & \text{если } I(p) \leq I(p^*). \end{cases} \quad (1)$$

Представление в виде набора результатов сравнения пар пикселей по кольцу, начиная с первого пикселя p_1 (например, с координатами $x = r, y = 0$) и заканчивая последним пикселем в кольце p_n , будем считать классическим кольцевым непараметрическим преобразованием локальной окрестности пикселя p^* :

$$C(p^*) = U\{p, \xi(p^*, p)\}, \tag{2}$$

где $p = p_1, p_2, p_3, \dots, p_n$.

Последовательная запись непараметрического преобразования формирует бинарную последовательность результатов сравнений пикселей кольцевого окружения $N(p^*)$ с пикселем p^* . На множестве колец различных радиусов можно сформировать обобщенный бинарный вектор (дескриптор), характеризующий рассматриваемую локальную особенность.

На рис. 1 показаны примеры дискретных аппроксимаций по алгоритму Брезенхема окружностей с радиусами $r = 1, 3, 6$ и 9 пикселей и их координаты (для $r = 6$ и 9 пикселей см. табл. 1) в первом квадранте, которые в целом образуют кольца соответственно с 8, 16, 32 и 52 пикселями. Значения координат для пикселей в других квадрантах можно легко получить, используя симметрию и отражение.

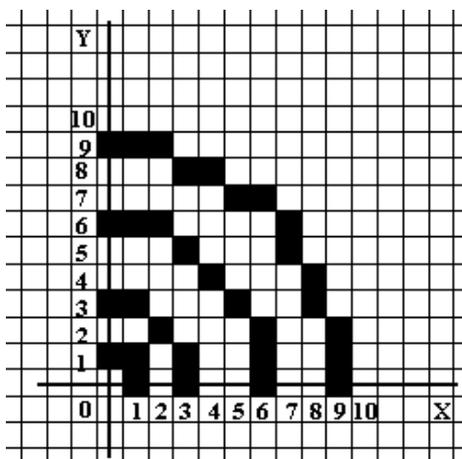


Рис. 1. Фрагменты дискретных аппроксимаций по алгоритму Брезенхема окружностей в первом квадранте с радиусами $r = 1, 3, 6$ и 9 пикселей

Таблица 1

Последовательности пикселей и их координаты при дискретной аппроксимации окружностей с различными радиусами по алгоритму Брезенхема

	$r = 6$ пикселей									$r = 9$ пикселей													
Номер пиксела	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Координата по X	6	6	6	5	4	3	2	1	0	9	9	9	8	8	7	7	6	5	4	3	2	1	0
Координата по Y	0	1	2	3	4	5	6	6	6	0	1	2	3	4	5	6	7	7	8	8	9	9	9

В качестве описаний кольцевых непараметрических преобразований в соответствии с [7] могут рассматриваться ранговая форма (rank transform) и форма в виде последовательной записи (census transform).

Первая традиционная форма (rank transform) непараметрического преобразования определяет число пикселей в области окружения, интенсивность которых больше интенсивности центрального пиксела:

$$R(p^*) = |\{p \in N(p^*), I(p) > I(p^*)\}|. \tag{3}$$

Вторая форма непараметрического преобразования (census transform) отображается в виде записи, представляющей бинарную последовательность результатов сравнений из набора пикселей кольцевого окружения $N(p^*)$ с центральным пикселем p^* . Примеры формирования такой записи при $n=16$ показаны на рис. 2 и для разных характерных локальных особенностей изображений могут быть реализованы в виде отличающихся по содержанию строк размерностью в 16 битов.

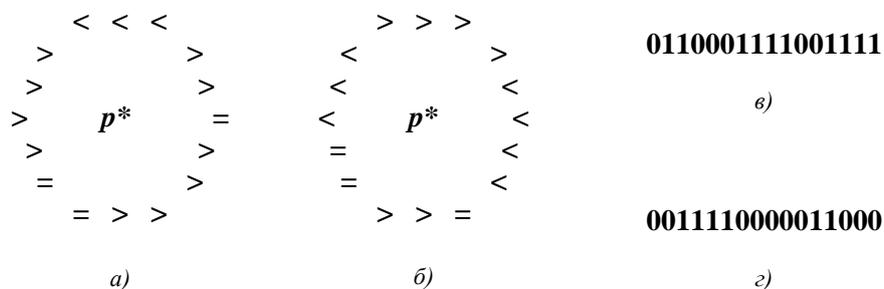


Рис. 2. Примеры формирования кольцевого непараметрического преобразования в виде census transform: а), б) результаты сравнения пикселей кольцевого окружения с центральным пикселем; в), г) сформированные соответствующие census-векторы

Форма census transform является удобным средством для сопоставления локальных особенностей характерных точек изображений по заданному порогу расстояния Хемминга:

$$D(a, b) = \sum_{i=1}^N a_i \text{XOR} b_i, \quad (4)$$

где a_i и b_i – элементы сравниваемых бинарных векторов a и b длины N ; XOR – логическая операция «исключающее ИЛИ».

2. Алгоритм поиска характерных точек в виде углов для дескриптора на базе непараметрических преобразований на кольцах

Алгоритм поиска углов на изображении представляется в виде двух этапов, первый из которых включает анализ присутствия признаков угла в анализируемой точке, а второй – анализ наличия самого угла [12]. На первом этапе производится анализ наличия вершины темного или светлого углов. Такой анализ в окне размером 3×3 пиксела позволяет выделить точку – претендента на угол, а если такой точки нет, то перейти к анализу следующей точки-претендента, не анализируя наличие самого угла. Если точка – претендент на вершину угла – обнаружена, то осуществляется переход на второй этап поиска угла со сторонами длиной не менее трех пикселей. Второй этап аналогичен алгоритму, реализуемому в известном детекторе углов FAST (Features from Accelerated Segment Test) [13]. Однако в предлагаемом детекторе углов имеются особенности, связанные с применением хеш-таблиц. Кроме того, применение двухэтапного алгоритма позволяет исключать из рассмотрения лежащие вблизи границы точки как ложные претенденты на углы.

Процедура работы *первого этапа* рассматриваемого алгоритма может быть представлена в виде следующих шагов:

Шаг 1. Поиск вершины темного угла на изображении. Под вершиной темного угла подразумеваются различные сочетания между центральным пикселем и пикселями окружения в окне размером 3×3 пиксела. На рис. 3, а и б показан ряд возможных сочетаний соответственно при трех и четырех пикселях, составляющих вершину темного угла.

В окне 3×3 осуществляются непараметрические преобразования путем сравнения с учетом заданного порога σ_1 интенсивности i -го пиксела окружения с интенсивностью центрального пиксела I_c и формирование бинарного вектора вершины темного угла VT1:

$$\Delta I_i = I_i - (I_c + \sigma_1), \text{ где } i = 1, 2, 3, \dots, 8. \quad (5)$$

Если $\Delta_i > 0$, то в бинарном векторе разряд $VT1(i) = 1$, в противном случае значение $VT1(i) = 0$.

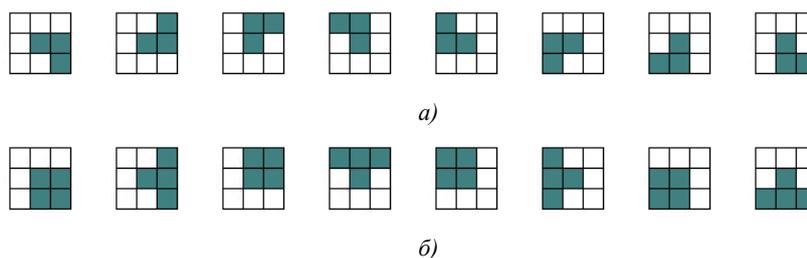


Рис. 3. Возможные сочетания начал темного угла относительно центрального пиксела в окне 3×3

Проделав цикл по $i = 1, 2, 3, \dots, 8$, получают восьмиразрядный вектор $VT1$. Он представляется двоичным числом, старший разряд которого соответствует первому сравнению. При этом считается, что вершины темных углов описываются сочетаниями следующих подряд нулей (единиц) вектора $VT1$ в соответствии с числами 63, 126, 159, 207, 231, 243, 249, 252 (если число темных пикселей окружения равно 2) и 31, 62, 124, 143, 199, 227, 241, 248 (если число темных пикселей окружения равно 3).

Шаг 2. Бинарный вектор $VT1$ используется в качестве входного адреса хэш-таблицы, в которую занесены значения 2 и 3 в соответствии с числом темных пикселей окружения в вершине угла и вышеприведенными числами двоичного кода вектора $VT1$.

Шаг 3. Если на выходе хэш-таблицы $p = 2$ или 3, то осуществляется переход ко второму этапу алгоритма – поиску темного угла, в противном случае – к поиску наличия вершины светлого угла для анализируемой точки.

Поиск вершины светлого угла. Под вершиной светлого угла подразумеваются сочетания между центральным пикселем и пикселями окружения, которые являются инверсными по отношению к представленным на рис. 3 сочетаниям. Поэтому вычисление разности между интенсивностью центрального пиксела I_c и интенсивностью i -го пиксела окружения осуществляется по формуле

$$\Delta_i = I_c - (I_i + \sigma_1), \quad (6)$$

где $i = 1, 2, 3, \dots, 8$.

В остальном поиск вершины светлого угла аналогичен поиску вершины темного угла.

Второй этап алгоритма поиска углов базируется на анализе отношений между найденным на первом этапе пикселем вершины угла и пикселями окружения на кольце радиусом $r = 3$ пиксела, которое содержит 16 пикселей окружения. Для получения восьмиразрядного входного вектора и хэш-таблицы меньшей размерности проведено «прореживание» пикселей на кольце путем перехода от 16 к 8 пикселям. Углами будем считать число следующих подряд прореженных пикселей с большей (меньшей) интенсивностью по отношению к центральному пикселу, равное 1, 2, 3 или 4 из 8 рассматриваемых пикселей. Пошаговая процедура алгоритма поиска угла на втором этапе может быть представлена в следующем виде:

Шаг 1. Поиск темного угла. Сравняется по интенсивности i -й пиксел на кольце с центральным пикселем I_c с учетом заданного порога σ_2 и формируется бинарный вектор $VT2$ по следующему правилу:

$$\Delta_i = I_i - (I_c + \sigma_2), \quad (7)$$

где $i = 1, 2, 3, \dots, 8$.

Если $\Delta_i > 0$, то в бинарном векторе i -й разряд определяют как $VT2(i) = 1$ и значение I_i суммируется в $ST1$, в противном случае – как $VT2(i) = 0$ и значение I_i суммируется в $ST2$.

Проделав цикл по $i = 1, 2, 3, \dots, 8$, получают бинарный вектор $VT2$ и две суммы ($ST1$ и $ST2$) соответствующих значений I_i .

Шаг 2. Использование бинарного вектора VT2 в качестве входного адреса хеш-таблицы, в которую занесены числа 1, 2, 3 и 4 в соответствии со следующими значениями двоичного кода вектора VT2:

число 1: 127, 191, 223, 239, 247, 251, 253, 254;

число 2: 63, 126, 159, 207, 231, 243, 249, 252;

число 3: 31, 62, 124, 143, 199, 227, 241, 248;

число 4: 15, 30, 60, 120, 135, 195, 225, 240.

Шаг 3. Обращение к хэш-таблице и получение на выходе числа p .

Если $p = 0$, то темный угол отсутствует для анализируемой точки изображения и осуществляется переход на конец алгоритма.

Если $p \neq 0$, то вычисляется величина «значимости» темного угла, которая равна модулю разности нормированных ST1 и ST2:

$$ST = \{|[ST1 / (8 - p)] - (ST2 / p)|\}. \quad (8)$$

Шаг 4. Фильтрация найденных темных углов в заданном окне: выбирается только один угол с максимальным значением значимости ST, затем осуществляется переход в конец алгоритма.

Поиск светлого угла аналогичен рассмотренному выше поиску темного угла, однако сравнение интенсивностей осуществляется по правилу

$$\Delta I_i = I_c - (I_i + \sigma_2), \quad (9)$$

где $i = 1, 2, 3, \dots, 8$.

Фильтрация светлых углов также осуществляется в заданном окне путем выбора угла с максимальным значением значимости светлого угла.

3. Алгоритм построения многокольцевого дескриптора локальных особенностей

Как отмечалось выше, структура формируемого вокруг характерной точки дескриптора локальных особенностей изображения описывается набором колец, которые строятся по алгоритму Брезенхема описания окружности с различными радиусами на дискретно заданных точках (пикселах) плоскости изображения. В данном алгоритме на каждом шаге точка формируемого кольца шириной в один пиксел определяется путем рассмотрения трех пикселов (горизонтального, диагонального и вертикального) и выбора из них пиксела с минимальным квадратом разности между его расстоянием до точки и радиусом задаваемой окружности. Задавая на дискретном поле растрового изображения различные радиусы в целых числах (пикселах), можно получать аппроксимации окружностей (колец) с различным числом пикселов. В качестве примера локального дескриптора может быть приведен дескриптор со структурой, состоящей из четырех колец (см. рис. 1) с максимально возможным числом пикселов 8, 16, 32 и 52 соответственно. В общем случае и число колец в дескрипторе, и значения соответствующих радиусов могут быть различными. Эти параметры устанавливаются как компромиссное решение между точностью и трудоемкостью вычисления дескриптора.

Для обеспечения инвариантности описанного алгоритма к изменению масштаба необходимо увеличить радиусы анализируемых колец на величину масштаба и осуществить смещение координат точек на кольцо.

Алгоритм построения локального дескриптора с изменяемым масштабом M показан на примере построения кольца с радиусом $r = 3$ пиксела вокруг характерной точки с координатами $X = i, Y = j$. В табл. 2 представлены изменения координат пикселов окружения при изменении масштаба в M раз.

Таблица 2

Значения координат для кольца радиусом $r = 3$ пиксела при изменении масштаба дескриптора в M раз

Номер точки	Координата по оси X	Координата по оси Y	Номер точки	Координата по оси X	Координата по оси Y
1	$i+3 \times M$	$j+0 \times M$	9	$i-3 \times M$	$j+0 \times M$
2	$i+3 \times M$	$j+1 \times M$	10	$i-3 \times M$	$j-1 \times M$
3	$i+2 \times M$	$j+2 \times M$	11	$i-2 \times M$	$j-2 \times M$
4	$i+1 \times M$	$j+3 \times M$	12	$i-1 \times M$	$j-3 \times M$
5	$i+0 \times M$	$j+3 \times M$	13	$i+0 \times M$	$j-3 \times M$
6	$i-1 \times M$	$j+3 \times M$	14	$i+1 \times M$	$j-3 \times M$
7	$i-2 \times M$	$j+2 \times M$	15	$i+2 \times M$	$j-2 \times M$
8	$i-3 \times M$	$j+1 \times M$	16	$i+3 \times M$	$j-1 \times M$

Как следует из табл. 2, при изменении масштаба в M раз радиус кольца также увеличивается в M раз и координаты пикселей окружения получают смещение на величину M по сравнению с единичным масштабом. Несложно показать, что аналогичные изменения происходят с координатами пикселей окружения для колец с другими радиусами.

При изменении масштаба также может производиться замена отдельных пикселей на масштабируемые окна (суперпиксели) размером $M \times M$ пикселей и осуществляться усреднение в них значений интенсивности (яркости) пикселей. Средняя интенсивность масштабируемого пиксела I_M с координатами i, j может быть вычислена следующим образом:

если M – нечетное число, то

$$I_M = \left(\sum_{k=-(M-1)/2}^{(M-1)/2} \sum_{l=-(M-1)/2}^{(M-1)/2} I_{(i+k),(j+l)} \right) / M^2, \quad (10)$$

где $k = -(M-1)/2, \dots, 0, \dots, (M-1)/2$; $l = -(M-1)/2, \dots, 0, \dots, (M-1)/2$;

если M – четное число, то

$$I_M = \left(\sum_{k=-\frac{M}{2}}^{\frac{M}{2}-1} \sum_{l=-[\frac{M}{2}-1]}^{\frac{M}{2}} I_{(i+k),(j+l)} \right) / M^2, \quad (11)$$

где $k = -M/2, \dots, 0, \dots, [(M/2)-1]$; $l = -[(M/2)-1], \dots, 0, \dots, (M-1)/2$.

Из выражений (10), (11) следует:

$$I_1 = I_{(i),(j)} \text{ при } M = 1, k = 0 \text{ и } l = 0;$$

$$I_2 = \left(\sum_{k=-1}^0 \sum_{l=0}^1 I_{(i+k),(j+l)} \right) / 4 \text{ при } M = 2, k = -1, 0 \text{ и } l = 0, 1;$$

$$I_3 = \left(\sum_{k=-1}^1 \sum_{l=-1}^1 I_{(i+k),(j+l)} \right) / 9 \text{ при } M = 3, k = -1, 0, 1 \text{ и } l = -1, 0, 1;$$

$$I_4 = \left(\sum_{k=-2}^1 \sum_{l=-1}^2 I_{(i+k),(j+l)} \right) / 16 \text{ при } M = 4, k = -2, -1, 0, 1 \text{ и } l = -1, 0, 1, 2;$$

$$I_5 = \left(\sum_{k=-2}^2 \sum_{l=-2}^2 I_{(i+k),(j+l)} \right) / 25 \text{ при } M = 5, k = -2, -1, 0, 1, 2 \text{ и } l = -2, -1, 0, 1, 2;$$

$$I_6 = \left(\sum_{k=-3}^2 \sum_{l=-2}^3 I_{(i+k),(j+l)} \right) / 36 \text{ при } M = 6, k = -3, -2, -1, 0, 1, 2 \text{ и } l = -2, -1, 0, 1, 2, 3;$$

$$I_7 = \left(\sum_{k=-3}^3 \sum_{l=-3}^3 I_{(i+k),(j+l)} \right) / 49 \text{ при } M = 7, k = -3, -2, -1, 0, 1, 2, 3 \text{ и } l = -3, -2, -1, 0, 1, 2, 3$$

и т. д.

Таким образом, алгоритм изменения масштаба дескриптора связан с изменением радиуса его колец в M раз по сравнению с единичным масштабом и смещением на M единиц координат соответствующих пикселей окружения в кольцах.

Рассмотрим структуру локального дескриптора, заданного четырьмя кольцами: радиусами в 1 и 3 пиксела, а также последующими кольцами с масштабами $M = 3$ и $M = 5$ по отношению ко второму кольцу, т. е. с радиусами в 9 и 15 пикселей соответственно. Таким образом, представленный дескриптор будет содержать 56 ($8+3 \times 16$) пикселей окружения.

Затем алгоритм построения дескриптора сводится к выбору на множестве пикселей колец пар сравнения, на основании результатов которых формируется бинарный вектор. Вариантов построения пар пикселей сравнения может быть несколько. Например, это может быть традиционное сравнение центрального пиксела (характерной точки) с пикселями окружения, расположенными на кольцах. Однако такой подход обладает низкой помехоустойчивостью при изменении интенсивности центрального пиксела [14]. Вторым вариантом является выбор пар сравнения интенсивностей пикселей, расположенных диаметрально симметрично на кольцах или лежащих на концах их хорд. Может быть осуществлен случайный выбор пар сравнения пикселей, заданных на кольцах, как это делается в алгоритме BRIEF (Binary robust independent elementary features) [3]. При построении рассматриваемого дескриптора, состоящего из 56 пикселей окружения, возможно построение 1540 пар сравнения. Поэтому, задавая размерность бинарного вектора дескриптора, используем аналогично [4, 6] выбор пар сравниваемых пикселей, которые характеризуются наибольшим значением разности и наименьшей корреляцией.

Сопоставление сформированных локальных дескрипторов проводится путем вычисления расстояния Хемминга рассматриваемых бинарных векторов (реализации логической процедуры «исключающее ИЛИ» (XOR) и сравнения числа единиц несовпадения с установленными порогом). Современные архитектуры имеют только одну инструкцию (POPCNT), чтобы сравнить по XOR и подсчитать число бит в бинарном векторе [3]. Это обеспечивает высокое быстродействие при вычислении расстояния Хемминга.

4. Экспериментальные результаты работы дескрипторов на базе многокольцевых непараметрических преобразований

Оценка предложенного подхода к построению локальных дескрипторов изображений на базе многокольцевых непараметрических преобразований базировалась на проверке результатов работы предложенных дескрипторов на широко используемой базе тестовых изображений [15, 16], состоящей из восьми видов изображений (graf, ubc, wall, bikes, trees, bark, leuven, boat). Каждый вид содержит по шесть изображений с различными типами и пятью степенями искажений. В качестве примера рассматривались дескрипторы размерностью 256 битов (New256), 128 битов (New128 и New128* – произвольный выбор из 256 пар сравнения) и 64 бита (New64), построенные на четырех масштабируемых кольцах Брезенхема с общим окружением в 56 пикселей. Проводилось сравнение работы предложенных дескрипторов с известными дескрипторами SURF [2], ORB [4], BRISK [5], FREAK[6], представленными в открытой библиотеке OpenCV 2.4. В качестве критериев оценки работы дескрипторов использовались ставшие традиционными показатели точности (precision) и полноты (recall) [15, 16].

В табл. 3 и 4 и на рис. 4 и 5 приведены результаты экспериментальной проверки работы дескрипторов на отдельных тестовых изображениях в виде усредненных показателей точности и полноты соответственно.

Таблица 3

Среднее значение точности работы дескрипторов на отдельных тестовых изображениях

Изображение	SURF	ORB	BRISK	BRISK+ +FREAK	New256	New128	New128*	New64
graf	0,34	0,46	0,37	0,34	0,43	0,41	0,35	0,31
ubc	0,93	0,98	0,82	0,67	0,97	0,96	0,97	0,95
wall	0,71	0,68	0,52	0,28	0,66	0,57	0,56	0,41
bikes	0,84	0,89	0,32	0,23	0,85	0,84	0,84	0,79
trees	0,58	0,80	0,46	0,67	0,75	0,65	0,68	0,55
bark	0,63	0,68	0,44	0,18	0,55	0,49	0,54	0,34
leuven	0,87	0,86	0,83	0,73	0,85	0,80	0,81	0,77
boat	0,70	0,90	0,30	0,15	0,82	0,78	0,78	0,67

Таблица 4

Среднее значение полноты работы дескрипторов для различных тестовых изображений

Изображение	SURF	ORB	BRISK	BRISK+ +FREAK	New256	New128	New128*	New64
graf	0,22	0,18	0,11	0,15	0,18	0,17	0,15	0,15
ubc	0,82	0,82	0,32	0,37	0,82	0,80	0,81	0,77
wall	0,47	0,25	0,10	0,11	0,25	0,23	0,22	0,19
bikes	0,64	0,57	0,09	0,11	0,54	0,53	0,52	0,50
trees	0,25	0,37	0,08	0,37	0,35	0,33	0,34	0,31
bark	0,61	0,38	0,06	0,06	0,37	0,34	0,35	0,32
leuven	0,68	0,60	0,33	0,35	0,59	0,56	0,55	0,52
boat	0,53	0,41	0,06	0,07	0,42	0,41	0,41	0,35

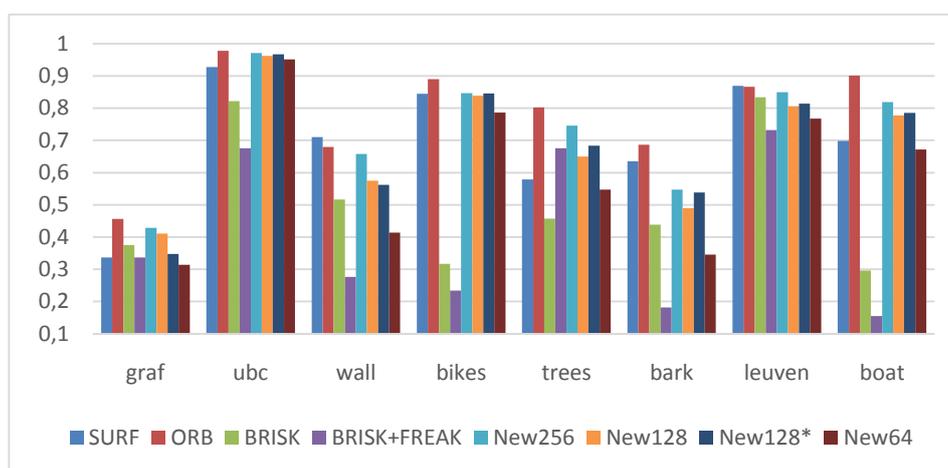


Рис. 4. Диаграммы средней точности работы дескрипторов на отдельных тестовых изображениях

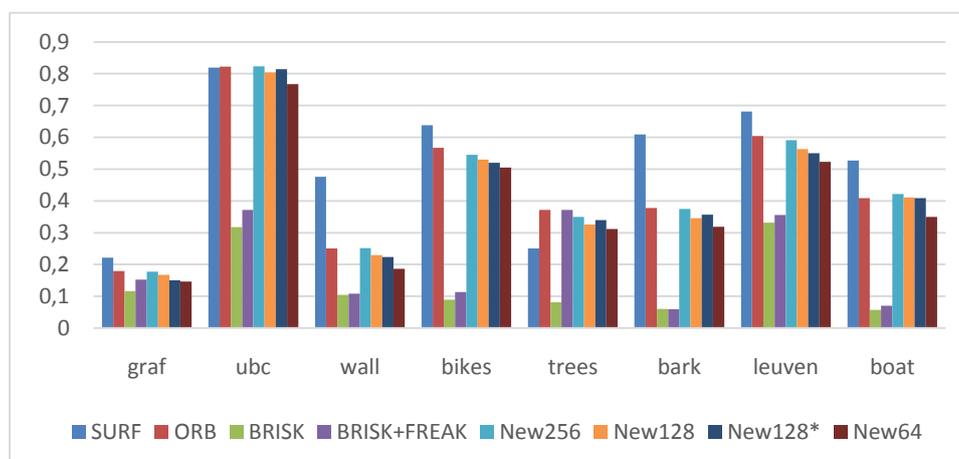


Рис. 5. Диаграммы средней полноты работы дескрипторов на различных тестовых изображениях

В табл. 5 и на рис. 6 представлены усредненные в целом по тестовой базе изображений результаты работы дескрипторов в виде показателей точности и полноты.

Таблица 5

Средние значения точности и полноты работы дескрипторов на тестовой базе изображений

Показатели	SURF	ORB	BRISK	BRISK+ +FREAK	New256	New128	New128*	New64
precision	0,70	0,78	0,51	0,41	0,73	0,69	0,69	0,60
recall	0,53	0,45	0,14	0,20	0,44	0,42	0,42	0,39

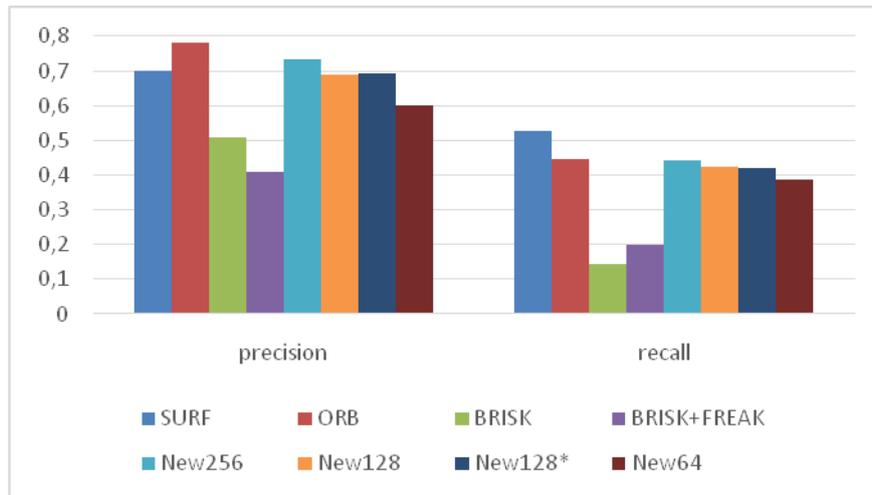


Рис. 6. Диаграммы средней точности и полноты работы дескрипторов на тестовой базе изображений

Как следует из приведенных результатов, предложенный подход к построению локальных дескрипторов на базе многокольцевых непараметрических преобразований на масштабируемых кольцах Брезенхема позволяет создавать конструкции дескрипторов, которые могут быть конкурентоспособными известным дескрипторам. Результаты экспериментальной проверки работы предложенного подхода к построению локальных дескрипторов показали, что качество работы дескриптора по критериям точности и полноты зависит от структуры обрабатываемого изображения. На рис. 7 и 8 приведены примеры графиков точности для тестовых изображений graf и ubc при различных уровнях искажений.

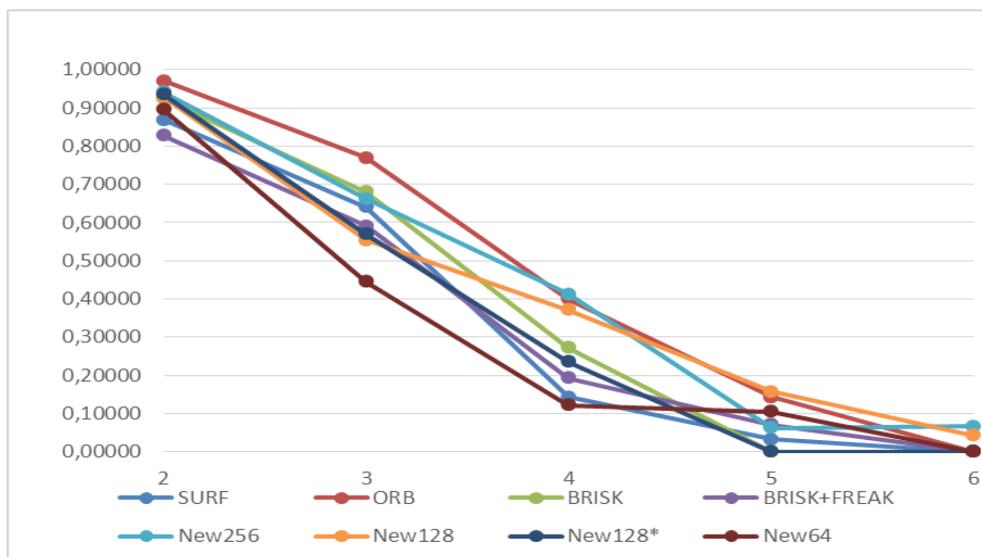


Рис. 7. Графики точности работы дескрипторов на тестовом изображении graf при пяти различных изменениях угла съемки

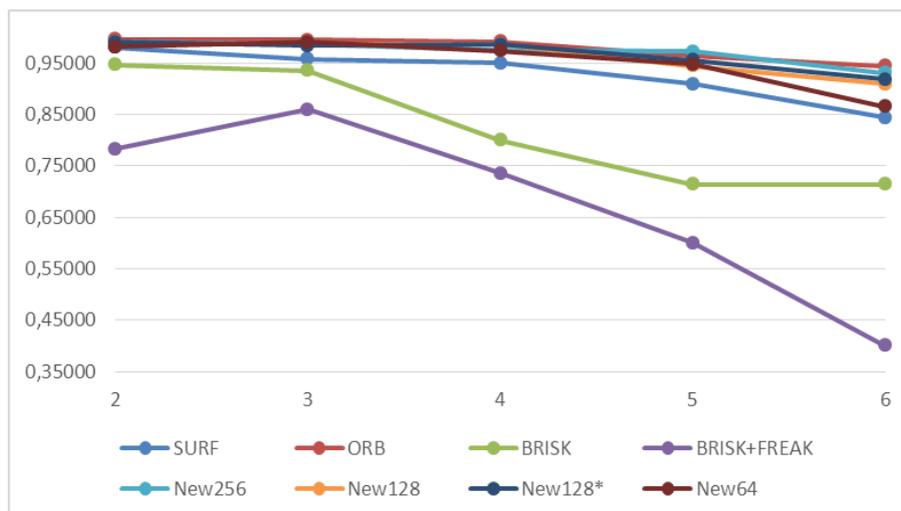


Рис. 8. Графики точности работы дескрипторов на тестовом изображении ubc при пяти различных степенях сжатия

Следует также отметить, что в общем случае значения показателей зависят от размера формируемого бинарного вектора. Однако, например, для набора изображений ubc (искажение типа «различные степени JPEG-сжатия») дескрипторы размером 256, 128 и 64 бита дают по точности примерно равные хорошие результаты. Поэтому представляет интерес исследование выбора эффективных параметров дескрипторов для работы на разнообразных структурах изображений.

Заключение

Экспериментальная проверка работы дескрипторов локальных особенностей изображений на базе многокольцевых непараметрических преобразований показала, что такие дескрипторы в зависимости от задаваемого размера бинарного вектора обладают различными значениями показателей точности и полноты, являются конкурентоспособными известным дескрипторам и могут быть использованы в качестве детекторов углов и описателей локальных особенностей изображений.

Список литературы

1. Lowe, D. Distinctive image features from scale-invariant key points / D. Lowe // Intern. J. of Computer Vision. – 2004. – Vol. 2(60). – P. 91–110.
2. Bay, H. SURF: Speeded Up Robust Features / H. Bay, T. Tuytelaars, L. van Gool // Proc. of the 9th European Conf. on Computer Vision. – Graz, Austria : Springer LNCS, 2006. – Vol. 3951, pt. 1. – P. 404–417.
3. Brief: Binary robust independent elementary features / M. Calonder [et al.] // Proc. of the 11th European conf. on Computer vision (ECCV 2010), sept. 5–11, 2010. – Heraklion, Crete, Greece, 2010. – P. 778–792.
4. ORB: an efficient alternative to SIFT or SURF / E. Rublee [et al.] // IEEE Intern. Conf. on Computer Vision (ICCV 2011), nov. 6–13, 2011. – California, 2011. – P. 2564–2571.
5. Leutenegger, S. BRISK: Binary robust invariant scalable keypoints / S. Leutenegger, M. Chli, R.Y. Siegwart // IEEE Intern. Conf. on Computer Vision (ICCV 2011), nov. 6–13, 2011. – California, 2011. – P. 2548–2555.
6. Alahi, A. Freak: Fast retina keypoint / A. Alahi, R. Ortiz, P. Vandergheynst // Computer Vision and Pattern Recognition (CVPR), 16–21 June 2012. – Switzerland, 2012. – P. 510–517.
7. Zabih, R. Non-parametric Local Transforms for Computing Visual Correspondence / R. Zabih, J. Woodfill // Proc. of 3rd European Conf. on Computer Vision. – Sweden, 1994. – P. 150–158.

8. Butkin, G. Nonparametric transforms for describing local image features / G. Butkin, D. Zhuk, A. Tuzikov // Conf. Pattern Recognition and Information Proc. (PRIP'2011), 18–20 May 2011. – Minsk : BSUIR, 2011. – P. 209–212.
9. Ojala, T. A comparative study of texture measures with classification based on feature distributions / T. Ojala, M. Pietikainen, D. Harwood // Pattern Recognition. – 1996. – Vol. 29(1). – P. 51–59.
10. Ojala, T. Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns / T. Ojala, M. Pietikainen, T. Mäenpää // IEEE Transactions on Pattern Analysis and Machine Intelligence. – 2002. – Vol. 24, no. 7. – P. 971–987.
11. Роджерс, Д. Алгоритмические основы машинной графики / Д. Роджерс. – М. : Мир, 1989. – 512 с.
12. Буткин, Г.А. Построение детекторов углов на базе анализа непараметрических преобразований на кольцах / Г.А. Буткин, И.А. Емельянов, А.В. Тузиков // Материалы VI Белорусского космического конгресса, 28–30 октября 2014 г. – Минск : ОИПИ НАН Беларуси, 2014. – Т. 1. – С. 257–260.
13. Rosten, E. Machine Learning for High Speed Corner Detection / E. Rosten, T. Drummond // Proc. Ninth European Conf. Computer Vision. – 2006. – Vol. 1. – P. 430–443.
14. Буткин, Г.А. Об устойчивости непараметрических преобразований в задачах описания локальных особенностей изображений / Г.А. Буткин, А.В. Тузиков // Информатика. – 2011. – № 1. – С. 15–24.
15. Mikolajczyk, K. A performance evaluation of local descriptors / K. Mikolajczyk, C. Schmid // IEEE Transactions on Pattern Analysis and Machine Intelligence. – 2005. – Vol. 27, no. 10. – P. 1615–1630.
16. Miksik, O. Evaluation of Local Detectors and Descriptors for Fast Feature Matching / O. Miksik, K. Mikolajczyk // Proc. of the Intern. Conf. of Pattern Recognition (ICPR), 11–15 nov. 2012. – Tsukuba Science City, Japan, 2012. – P. 2681–2684.

Поступила 04.10.2016

*Объединенный институт проблем
информатики НАН Беларуси,
Минск, Сурганова, 6
e-mail: butkin@newman.bas-net.by,
bujhm24@gmail.com,
tuzikov@newman.bas-net.by*

G.A. Butkin, I.A. Emelianov, A.V. Tuzikov

ALGORITHMS FOR CONSTRUCTING THE DESCRIPTORS OF LOCAL IMAGE FEATURES BASED ON MULTIRING NONPARAMETRIC TRANSFORMATION

Application of nonparametric transformations for construction of local image features descriptors is considered. An approach based on scalable Bresenham circles for multiring nonparametric transformations is proposed. A problem of finding characteristic points (corners) in images is solved using two rings of small radius. Then, considering rings of greater radius a final decision is made concerning local features under investigation. Testing results of proposed local descriptors have demonstrated their competitiveness with known descriptors.