

## МЕТОД БЫСТРОГО ВЫЧИСЛЕНИЯ LBP-ПРИЗНАКОВ ОБЛАСТИ ИНТЕРЕСА ВИДЕОИЗОБРАЖЕНИЯ

*Толочко А.М., аспирант кафедры СиУТ Белорусского государственного университета информатики и радиоэлектроники, e-mail: Aljaksandr@gmail.com*

*Борискевич А.А., к.т.н., доцент кафедры СиУТ Белорусского государственного университета информатики и радиоэлектроники, e-mail: anbor@bsuir.by*

**Ключевые слова:** видеоизображение, текстура, LBP-признаки, быстрые вычисления, примитивы, локальная окрестность.

### Введение

Одной из важных характеристик визуальной информации, используемой зрительной системой человека и современными системами машинного зрения, является текстура. Она является характеристикой пространственной организации изменения интенсивности поверхности объекта, количественно определяющей такие свойства, как гладкость, грубость и регулярность. Основными подходами к описанию текстуры являются статистические, спектральные и структурные методы [1].

Среди множества известных методов статистического описания текстуры локальной области изображения следует выделить группу методов, основанных на вычислении LBP-признаков (Local Binary Pattern) [2, 3]. Устойчивость к изменениям условий освещения, простота реализации и компактность представления базового LBP-признака определили появление большого количества его модификаций, обладающих новыми свойствами: инвариантность к повороту ( $LBP_{8,1}^{riu2}$  [4]), устойчивость к аддитивному шуму (TLBP [5]) и др.

Различные модификации LBP-признака, используемые для описания текстуры области изображения, находят широкое применение в различных прикладных задачах: обнаружение и распознавание лиц [6]; анализ, сегментация и классификация текстур аэрокосмических, промышленных и биомедицинских изображений [7]; моделирование заднего плана и обнаружение динамических объектов [8]; сопровождение динамических объектов в видеопотоке [9].

Для эффективного использования вычислительных ресурсов в современных системах обработки и анализа видеоизображения в режиме реального времени актуальной становится задача уменьшения времени вычисления текстурных признаков. В связи с этим, целью работы является разработка метода быстрого вычисления LBP-признаков.

### Базовые методы формирования LBP-признаков

Базовый вариант  $LBP_{8,1}$  [4] представляет собой 8-битное кодовое слово, бинарное представление которого основано на анализе круговой окрестности единич-

*Предложен метод быстрого вычисления LBP-признаков изображения, основанный на выделении примитива регулярной вычислительной структуры, установлении свойства взаимности пороговой и модифицированной пороговой функций, определяющих правила сравнения соседних пикселей локальной окрестности, и правила обновления LBP-признаков. Он позволяет получить выигрыш в скорости вычисления на уровне 66-72% для инвариантного к повороту LBP-признака с размером примитива 3x3 по отношению к традиционному методу за счёт сокращения удельного количества операций сравнения и пикселей, к которым осуществляется доступ при вычислении LBP-признака.*

ного радиуса кодируемого пикселя изображения.  $LBP_{8,1}$  - признак вычисляется в соответствии с выражением:

$$LBP_{8,1} = \sum_{i=0}^8 g(I_p, I_c) 2^i, \quad (1)$$

где  $I_c$  - значение интенсивности центрального пикселя в окрестности, для которого рассчитывается значение  $LBP_{8,1}$ -признака;  $I_p$  - значение интенсивности  $p$ -го элемента локальной круговой окрестности,  $p = 0, 1, \dots, 7$ .

Пороговая функция  $g(I_p, I_c)$  значений интенсивностей элемента окрестности  $I_p$  и центрального пикселя  $I_c$  определяется выражением:

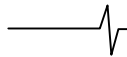
$$g(I_p, I_c) = \begin{cases} 1, & \text{при } I_p \geq I_c, \\ 0, & \text{при } I_p < I_c. \end{cases} \quad (2)$$

Локальная круговая окрестность центрального пикселя  $I_c$  представлена на рис. 1. Пусть  $I_c$  определяет значение интенсивности элемента изображения с координатами  $(x_c, y_c)$  ( $I_c = \text{Img}(x_c, y_c)$ ), тогда координаты  $(x_p, y_p)$   $p$ -го элемента круговой окрестности для формирования  $LBP_{8,1}$ -признака задаются формулой:

$$\begin{aligned} x_p &= x_c + \sin(\pi(p-1)/4), \\ y_p &= y_c - \cos(\pi(p-1)/4). \end{aligned} \quad (3)$$

Значения интенсивностей элементов круговой окрестности, имеющих нецелочисленные координаты, рассчитываются посредством билинейной интерполяции. В соответствии с формулой (1),  $LBP_{8,1}$ -признак может принимать одно из 256 значений из диапазона  $[0, 255]$ .

$LBP_{8,1}^{u2}$ -признак уменьшает пространство возможных значений  $LBP_{8,1}$ -признака, используя свойство равно-



мерности (uniformity) локального паттерна. Степень равномерности  $U(LBP_{8,1})$  [4], характеризующая количество переходов 0/1, 1/0 в циклическом представлении  $LBP_{8,1}$ -признака, вычисляется с помощью соотношения:

$$U(LBP_{8,1}) = |g(I_7, I_c) - g(I_0, I_c)| + \sum_{p=1}^7 |g(I_{p-1}, I_c) - g(I_p, I_c)| \quad (4)$$

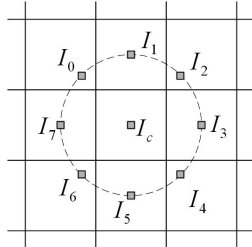


Рис. 1. Локальная круговая окрестность для формирования  $LBP_{8,1}$ -признака центрального пикселя  $I_c$

Равномерными считаются паттерны, имеющие не более двух бинарных переходов в циклическом представлении  $LBP_{8,1}$ -признака. В случае невыполнения данного требования паттерн считается неравномерным. Замечено, что для реальных изображений почти 90%  $LBP_{8,1}$ -признаков обладают свойством равномерности.

Выражение для вычисления  $LBP_{8,1}^{u2}$  признака можно представить в виде

$$LBP_{8,1}^{u2} = \begin{cases} 0, & \text{при } LBP_{8,1} = 0; \\ 57, & \text{при } LBP_{8,1} = 255; \\ 8 \cdot \sum_{p=0}^7 g(I_p, I_c) - 7 + \lg_2(\text{idx}), & \text{при } LBP_{8,1} \neq 0, 255; \\ 58, & \text{при } U(LBP_{8,1}) > 2, \end{cases} \quad \text{при } U(LBP_{8,1}) \leq 2, \quad (5)$$

Здесь  $\text{idx}$  является решением модулярного уравнения вида:

$$\left( 2^{\sum_{p=0}^7 g(I_p, I_c)} - 1 \right) \text{idx} \equiv LBP_{8,1} \pmod{255} \quad (6)$$

$LBP_{8,1}^{u2}$ -признак может принимать одно из 59 возможных значений. Кодирование группы равномерных  $LBP_{8,1}^{u2}$ -признаков с одинаковой мощностью (количество единиц в бинарном представлении) только одним кодовым словом позволяет добиться инвариантности к повороту  $LBP_{8,1}$ -признака. Выражение для расчёта инвариантного к повороту  $LBP_{8,1}^{riu2}$ -признака имеет вид:

$$LBP_{8,1}^{riu2} = \begin{cases} \sum_{p=0}^7 g(I_p, I_c), & \text{при } U(LBP_{8,1}) \leq 2 \\ 9, & \text{при } U(LBP_{8,1}) > 2 \end{cases} \quad (7)$$

$LBP_{8,1}^{u2}$  и  $LBP_{8,1}^{riu2}$  признаки естественным образом выявляют типовые особенности текстуры: точки, окончания линий, углы, края.

Круговой характер топологии анализируемой окрестности является условием обеспечения инвариантности  $LBP_{8,1}^{riu2}$ -признака к повороту [4]. Вместе с тем широко применяется базовая версия  $LBP$ -признака [2], использующая квадратную топологию окрестности (рис. 2).

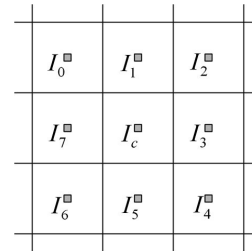


Рис. 2. Локальная квадратная окрестность для формирования  $LBP$ -признака центрального пикселя  $I_c$

Использование квадратной топологии окрестности позволяет избежать вычислительно ёмкой процедуры расчёта интерполированных значений её элементов с нецелочисленными координатами. Таким образом, значения элементов квадратной окрестности представляют собой значения пикселей соседей центрального пикселя  $I_c$  в окрестности 3x3 пикселя. Выражения для расчёта  $LBP$ -признака аналогичны выражениям для расчёта  $LBP_{8,1}$ -признака (1, 2).

Изменение пороговой функции (2) посредством сдвига порога на величину  $c$  позволяет обеспечить устойчивость признака к шуму. Пороговая функция  $g_t(I_p, I_c)$  для модифицированного  $LBP$ -признака – Threshold LBP ( $TLBP$ ) [4] – определяется выражением:

$$g_t(I_p, I_c) = \begin{cases} 1, & \text{при } I_p + c \geq I_c, \\ 0, & \text{при } I_p + c < I_c. \end{cases} \quad (8)$$

Метод повышения устойчивости признака к шуму посредством изменения пороговой функции (8) применим к представленным модификациям  $LBP$ -признака, образуя соответственно  $TLBP_{8,1}$ ,  $TLBP_{8,1}^{u2}$ ,  $TLBP_{8,1}^{riu2}$  признаки.

### Метод быстрого вычисления $LBP$ -признаков

$LBP$ -признаки с круговой топологией локальной окрестности, несмотря на ряд достоинств, обладают существенным недостатком, препятствующим реализации его быстрого вычисления: имеет место необходимость вычисления интерполированных значений элементов окрестности кодируемого пикселя, имеющих нецелочисленные координаты. Напротив,  $LBP$ -признаки с квадратной топологией локальной окрестности не требуют осуществления интерполяции значений элементов окрестности. При этом для базового варианта  $LBP$ -признака также применимо понятие равномерности паттерна, что позволяет сформировать  $LBP^{u2}$  и  $LBP^{riu2}$ -признаки.

Квадратная топология локальной окрестности позволяет реализовать метод быстрого вычисления  $LBP$ -признака. Метод быстрого вычисления  $LBP$ -признаков ( $F-LBP$ , Fast Local Binary Pattern) использует свойство взаимности пороговой функции: при известном результате операции сравнения аргументов  $a, b$  для пороговой функции  $g(a, b)$  значение функции  $g(b, a)$  также известно. Для расширенного представления пороговой функции (2) свойство взаимности может быть представ-

лено в форме табл. 1. Из табл. 1 видно, что каждому результату операции сравнения аргументов  $a, b$  из трёх возможных исходов ( $a > b$ ,  $a = b$ ,  $a < b$ ) соответствует пара бинарных значений пороговых функций  $g(a, b)$ ,  $g(b, a)$ . Таким образом, операция сравнения для получения значений пороговых функций  $g(a, b)$ ,  $g(b, a)$  при заданных аргументах  $a, b$  осуществляется единожды.

Таблица 1  
Свойство взаимности пороговой функции  $g(a, b)$

Результат операции сравнения аргументов $a, b$	Значение пороговой функции	
	$g(a, b)$	$g(b, a)$
$a > b$	1	0
$a = b$	1	1
$a < b$	0	1

Свойство взаимности для расширенного представления пороговой функции (8)  $g_i(a, b)$ , используемой для формирования модифицированных вариантов  $F$ -LBP-признаков ( $F$ -TLBP,  $F$ -TLBP<sup>u2</sup>,  $F$ -TLBP<sup>riu2</sup>), при положительных значениях порога  $c$  представлено в табл. 2. Следует отметить, что при нулевом значении порога  $c = 0$  пороговые функции  $g_i(a, b)$  и  $g(a, b)$  идентичны.

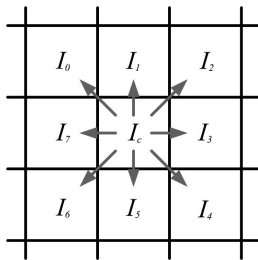


Рис. 3. Направление векторов вычисления операций сравнения в локальной окрестности

Использование свойства взаимности пороговой функции позволяет сократить удельное количество операций сравнения на один пиксель при вычислении LBP-признаков изображения (или его фрагмента). На рис. 3 представлена локальная окрестность размером 3x3 пикселей с указанием векторов вычислений операций сравнения в соответствии с пороговой функцией (от кодируемого центрального пикселя к элементам его окрестности). В соответствии с рис. 3 при вычислении LBP-признака центрального пикселя  $I_c$  требуется выполнение  $N_{comp}^{LBP} = 8$  процедур сравнения и осуществление доступа к  $N_{access}^{LBP} = 9$  элементам изображения (8 элементов локальной окрестности и, собственно, сам кодируемый пиксель). Несложно заметить, что выполнение операции сравнения пороговой функции дублируется: каждая из 8 операций сравнения осуществляется при расчёте LBP-признака центрального элемента анализируемого паттерна, а также при расчёте LBP-признака соответствующего элемента локальной окрестности.

Повышение скорости вычисления LBP-признаков изображения (или его фрагмента) достигается посред-

ством выделения примитива регулярной вычислительной структуры за счёт двух влияющих факторов:

- снижение удельного количества процедур сравнения на один пиксель;
- снижение удельного количества элементов изображения, к которым осуществляется доступ для формирования  $F$ -LBP-признака.

Таблица 2  
Свойство взаимности модифицированной пороговой функции  $g_i(a, b)$

Результат операции сравнения аргументов $a, b$	Значение пороговой функции	
	$g(a, b)$	$g(b, a)$
$ b - a  \leq c$	1	1
$b - a > c$	0	1
$b - a < -c$	1	0

Вычислительный примитив представляет собой повторяющийся в пределах изображения фрагмент вычислительной структуры, определяющий правила доступа к пикселям изображения и правила осуществления операции сравнения при формировании  $F_{x,m}$ -LBP-признака. Параметры  $l, m$  характеризуют размерность вычислительного примитива.

На рис. 4 представлены три варианта вычислительного примитива различной размерности: 1x1 пиксель, 2x2 пикселей, 3x3 пикселей. Направление стрелок на рисунке определяет направление от элемента со значением интенсивности  $b$  к элементу с интенсивностью  $a$  пороговой функции  $g(a, b)$  (или  $g_i(a, b)$ ), осуществляющей сравнение двух соседних пикселей. Элементы изображения, к которым направлены стрелки, в совокупности с опорными элементами изображения, отмеченными символом  $h$ , образуют множество пикселей изображения  $P_{pe}$ , к которым осуществляется доступ при обработке примитивного элемента. На рис. 5 представлена регулярная вычислительная структура для фрагмента изображения размером 5x5 пикселей с примитивом размером 2x2 пикселей.

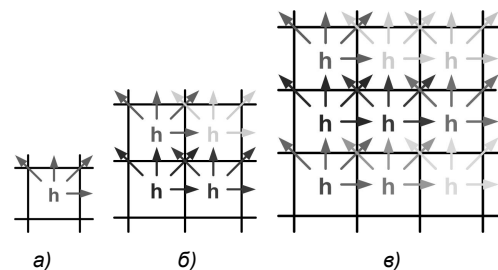


Рис. 4. Примитивы регулярной вычислительной структуры: а) 1x1 пиксель; б) 2x2 пикселей; в) 3x3 пикселей

Представленные на рис. 4 базовые вычислительные примитивы отличаются удельным количеством элементов изображения, к которым осуществляется доступ для формирования  $F$ -LBP-признака. Увеличение размерности примитива регулярной вычислительной структуры приводит к уменьшению удельного количества пикселей, к которым осуществляется доступ. При этом удельное количество процедур сравнения в пересчёте на один пиксель  $N_{comp}^{F-LBP}$  является одинаковым для всех размер-

ностей примитивов. Характеристики вычислительных примитивов представлены в табл. 3.

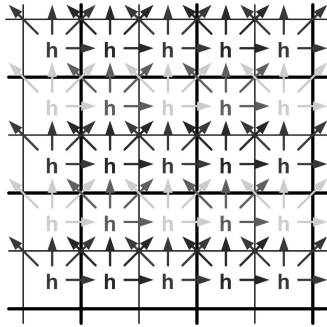


Рис.5. Регулярная вычислительная структура для фрагмента изображения

Вычислительная процедура обработки примитива регулярной вычислительной структуры включает следующие основные операции:

1. Осуществляется доступ к элементам изображения  $P_{pe}$ , необходимым для обработки вычислительного примитива.
2. Для каждого опорного компонента вычислительно-го примитива (отмеченного символом **h**) осуществляется сравнение пикселей по четырём направлениям в соответствии с пороговой функцией  $g(a,b)$  (или  $g_i(a,b)$ ), задаваемой табл. 1 (или табл. 2). По бинарным результатам  $i$ -й ( $i=1..4$ ) операции сравнения осуществляется обновление  $F-LBP$ -признаков соседних элементов в соответствии с побитовыми функциями  $f_i^a(F-LBP_a)$  и  $f_i^b(F-LBP_b)$  (табл. 4) по формулам:

$$F-LBP_a = f_i^a(F-LBP_a), \tag{9}$$

$$F-LBP_b = f_i^b(F-LBP_b). \tag{10}$$

Таблица 3

Характеристики базовых вычислительных примитивов

Размерность вычислительного примитива, $l \times m$	$N_{comp}^{F-LBP}$	$N_{access}^{F-LBP}$
1x1	4	5
2x2	4	11/4
3x3	4	19/9

При осуществлении обновления  $F-LBP$ -признаков в соответствии с формулами (9), (10) функция  $f_i^a(F-LBP_a)$  воздействует на один из четырёх младших битов бинарного представления  $LBP$ -признака пикселя с интенсивностью  $a$ , в то время, как функция  $f_i^b(F-LBP_b)$  воздействует на один из четырёх старших битов бинарного представления  $LBP$ -признака пикселя с интенсивностью  $b$ . Вычисление признаков  $F-LBP^{iu2}$ ,  $F-LBP^{riu2}$  ( $F-TLBP^{u2}$ ,  $F-TLBP^{riu2}$ ), реализованное на основании базового признака  $F-LBP$  ( $F-TLBP$ ) в соответствии с формулами (4)-(6), может быть эффективно осуществлено посредством таблиц поиска  $LUT_{u2}(LBP)$ ,  $LUT_{riu2}(LBP)$ .

**Результаты моделирования**

Для оценки эффективности предложенного метода быстрого вычисления  $LBP$ -признаков осуществлена программная реализация на языке программирования C++  $LBP$ -признаков:  $LBP^{u2}$ ,  $LBP^{riu2}$ ,  $TLBP^{u2}$ ,  $TLBP^{riu2}$ ,  $F_{l \times m} - LBP^{u2}$ ,  $F_{l \times m} - LBP^{riu2}$ ,  $F_{l \times m} - TLBP^{u2}$ ,  $F_{l \times m} - TLBP^{riu2}$ . Размерность  $l \times m$  вычислительного примитива в экспериментах представлена набором из значений: 1x1, 2x2, 3x3 пикселей. Реализованные функции вычисления  $LBP$ -признаков скомпилированы в тех-функции для использования в среде Matlab.

Представленные  $LBP$ -признаки рассчитывались для полутоновых изображений «Tbilisi», «Ankara», «Kopenhagen» размером  $n_x \times n_y = 800 \times 600$  пикселей (рис. 6). Моделирование осуществлялось на стандартном ПК Celeron 1.86GHz, 2Gb. В серии из  $N = 20$  запусков фиксировалось время  $T$  вычисления соответствующих типов  $LBP$ -признаков всех пикселей изображения. По серии измерений вычислялась средняя скорость  $Rate$  расчёта  $LBP$ -признаков:

$$Rate = \frac{n_x \times n_y}{N} \sum_{i=1}^N T_i \text{ [пиксель/сек]}. \tag{11}$$

Таблица 4

Функции  $f_i^a(F-LBP_a)$ ,  $f_i^b(F-LBP_b)$  обновления  $LBP$ -признака

Вектор вычисления	$g(a,b), g_i(a,b)$	0	1	1
	$g(b,a), g_i(b,a)$	1	1	0
↖	$f_1^a(F-LBP_a)$	$F-LBP_a$ or 1		$F-LBP_a$ and 254
	$f_1^b(F-LBP_b)$	$F-LBP_b$ and 239	$F-LBP_b$ or 16	
↑	$f_2^a(F-LBP_a)$	$F-LBP_a$ or 2		$F-LBP_a$ and 253
	$f_2^b(F-LBP_b)$	$F-LBP_b$ and 223	$F-LBP_b$ or 32	
↗	$f_3^a(F-LBP_a)$	$F-LBP_a$ or 4		$F-LBP_a$ and 251
	$f_3^b(F-LBP_b)$	$F-LBP_b$ and 191	$F-LBP_b$ or 64	
→	$f_4^a(F-LBP_a)$	$F-LBP_a$ or 8		$F-LBP_a$ and 247
	$f_4^b(F-LBP_b)$	$F-LBP_b$ and 127	$F-LBP_b$ or 128	

Для каждого варианта быстрого вычисления  $LBP$ -признака оценивался выигрыш в скорости  $\delta Rate$  по отношению к соответствующему базовому варианту  $LBP$ -признака по формуле:

$$\delta Rate = \frac{Rate_F - Rate_{Base}}{Rate_{Base}} \cdot 100\%, \quad (12)$$

где  $Rate_F$  – средняя скорость вычисления  $LBP$ -признаков, соответствующая быстрому методу вычисления;  $Rate_{Base}$  – средняя скорость вычисления  $LBP$ -признаков, соответствующая базовому варианту вычисления.

Величина сдвига порога функции  $g_i(a,b)$  (8) в процессе моделирования составляла  $c = 4$ . Табл. 5, 6 содержат результаты оценки скорости  $Rate$  и выигрыша в скорости  $\delta Rate$  относительно базового варианта расчёта  $LBP$ -признаков для  $LBP^{riu2}$ ,  $F-LBP^{riu2}$ ,  $TLBP^{riu2}$ ,  $F-TLBP^{riu2}$  признаков при разных размерностях примитива регулярной вычислительной структуры. На рис. 7 представлено графическое изображение  $LBP^{riu2}$ ,  $TLBP^{riu2}$ -признаков для изображения «Копенгаген».

Таблица 5  
Скорость вычисления  $LBP^{riu2}$ ,  $F-LBP^{riu2}$ -признаков

Наименование $LBP$ -признака	«Tbilisi»		«Ankara»		«Kopenhagen»	
	$Rate$ , пиксели/сек	$\delta Rate$ , %	$Rate$ , пиксели/сек	$\delta Rate$ , %	$Rate$ , пиксели/сек	$\delta Rate$ , %
$LBP^{riu2}$	4570864	0	4681408	0	4577011	0
$F_{1 \times 1} - LBP^{riu2}$	5627498	23	6263555	34	6002274	31
$F_{2 \times 2} - LBP^{riu2}$	7035291	54	7433466	59	7131787	56
$F_{3 \times 3} - LBP^{riu2}$	7620634	67	8050563	72	7708744	68

Таблица 6  
Скорость вычисления  $TLBP^{riu2}$ ,  $F-TLBP^{riu2}$ -признаков

Наименование $LBP$ -признака	«Tbilisi»		«Ankara»		«Kopenhagen»	
	$Rate$ , пиксели/сек	$\delta Rate$ , %	$Rate$ , пиксели/сек	$\delta Rate$ , %	$Rate$ , пиксели/сек	$\delta Rate$ , %
$TLBP^{riu2}$	4764211	0	4685137	0	4756369	0
$F_{1 \times 1} - TLBP^{riu2}$	6101520	28	6009571	28	6234694	31
$F_{2 \times 2} - TLBP^{riu2}$	7379653	55	7253018	55	7559939	59
$F_{3 \times 3} - TLBP^{riu2}$	7936973	67	7799633	66	8187392	72



а)

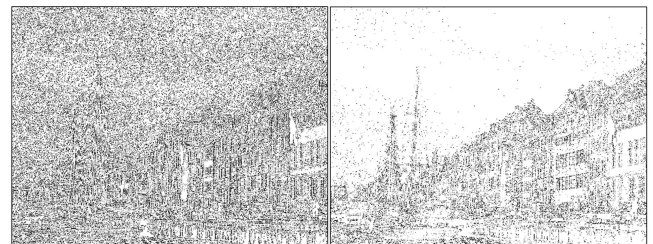


е)

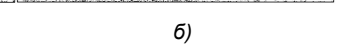


б)

Рис. 6. Тестовые изображения:  
а) «Tbilisi», б) «Ankara», е) «Kopenhagen»



а)



б)

Рис. 7. Графическое представление  $LBP$ -признаков для изображения «Копенгаген» а)  $LBP^{riu2}$ ; б)  $TLBP^{riu2}$

Из табл. 5, 6 видно, что применение метода быстрого вычисления  $LBP$ -признаков с размером вычислительного примитива  $3 \times 3$  позволяет обеспечить относительный выигрыш в скорости на уровне 66-72% для вариантов  $LBP$ -признака с обычной и модифицированной пороговой функцией. Для признаков  $F_{l \times l} - LBP^{riu2}$ ,  $F_{l \times l} - TLBP^{riu2}$  выигрыш в скорости вычисления составляет 23-34% и обеспечивается, с одной стороны, сокращением удельного количества операций сравнения с  $N_{comp}^{LBP} = 8$  до  $N_{comp}^{F-LBP} = 4$ , с другой стороны, сокращением удельного количества пикселей, к которым осуществляется доступ при расчёте  $LBP$ -признака, с  $N_{access}^{LBP} = 9$  до  $N_{access}^{F-LBP} = 5$ . Дальнейший выигрыш в скорости при увеличении размерности вычислительного примитива достигается за счёт сокращения удельного количества пикселей, к которым осуществляется доступ при расчёте  $LBP$ -признака.

Увеличение размерности примитива регулярной вычислительной структуры для обеспечения увеличения выигрыша в скорости вычисления повышает требования к размеру кэш-памяти процессора.

### Заключение

Предложен метод быстрого вычисления  $F_{l \times m} - LBP$ -признаков изображения, основанный на выделении примитива регулярной вычислительной структуры размером  $l \times m$  пикселей, установлении свойства взаимности пороговой  $g(a, b)$  и модифицированной пороговой  $g_t(a, b)$  функций, определяющих правила сравнения соседних пикселей локальной окрестности, и правила обновления  $LBP$ -признаков. Примитив регулярной вычислительной структуры определяет правила доступа к пикселям изображения и правила осуществления операции сравнения в соответствии с пороговой функцией.

Установлено, что предложенный метод с размером примитива  $3 \times 3$  для быстрого инвариантного к повороту  $LBP$ -признака ( $F_{3 \times 3} - LBP^{riu2}$ ,  $F_{3 \times 3} - TLBP^{riu2}$ ) позволяет получить выигрыш в скорости вычисления на уровне 66-72% по отношению к варианту без ускорения ( $LBP^{riu2}$ ,  $TLBP^{riu2}$ ) за счёт сокращения удельного количества операций сравнения с  $N_{comp}^{LBP} = 8$  до  $N_{comp}^{F-LBP} = 4$  и сокращения удельного количества пикселей, к которым осуществляется доступ при расчёте  $LBP$ -признака, с  $N_{access}^{LBP} = 9$  до  $N_{access}^{F-LBP} = 19/9$ .

Данный метод может быть использован для решения задач извлечения текстурной информации из видеоизображения в режиме реального времени.

### Литература

1. F. Tomita, S. Tsuji: Computer Analysis of Visual Textures, Kulwer Academic, Boston, 192 p. (1990)
2. Ojala, T., Pietikäinen, M., Harwood, D.: A comparative study of texture measures with classification based on feature distributions. Pattern Recognit. 29(1), 51–59 (1996)
3. Matti Pietikäinen, Abdenour Hadid, Guoying Zhao, Timo Ahonen: Computer Vision Using Local Binary Patterns. Springer, London (2011)
4. Ojala, T., Pietikäinen, M., Maenpää, M.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Transactions on Pattern Analysis and Machine Intelligence 24(7), 971–987 (2002)
5. Jun Meng, Yumao Gao, Xiukun Wang, Tsauyoung Lin, Jianying Zhang, Face Recognition Based on Local Binary Patterns with Threshold, IEEE International Conference on Granular Computing, San Jose, California, August 14–August 16, (2010)
6. Di Huang, Caifeng Shan, Ardabilian, M., Yunhong Wang, Liming Chen, «Local Binary Patterns and Its Application to Facial Image Analysis: A Survey», Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, On page(s): 765 - 781 Volume: 41, Issue: 6, Nov. (2011).
7. Loris Nanni, Alessandra Lumini, Sheryl Brahnam: Local binary patterns variants as texture descriptors for medical image analysis. Artificial Intelligence in Medicine 49(2): 117-125 (2010).
8. Heikkilä, M., Pietikäinen, M.: A texture-based method for modeling the background and detecting moving objects. IEEE Trans. Pattern Anal. Mach. Intell. 28(4), 657–662 (2006).
9. Ning, J., Zhang, L., Zhang, D., Wu, C.: Robust object tracking using joint color-texture histogram. Int. J. Pattern Recognit. Artif. Intell. 23(7), 1245–1263 (2009).

### FAST LBP-FEATURES COMPUTATION METHOD FOR VIDEO IMAGE ROI

*Talochka A.M., Boriskevich A.A*

A fast LBP-features computation method for the image based on forming a regular computation structure primitive, defining a mutuality property of the threshold- and the modified threshold-function determining pixel comparison rules in the local neighborhood, and update LBP-features rules is proposed. It allows us to obtain a computation rate gain equal about 66-72% for the Rotation Invariant LBP-feature in using  $3 \times 3$  primitive in contrast with the conventional one due to specific amount reduction of the comparison operations and pixels that are used for the LBP-feature computation.