

# АЛГОРИТМ ОБНАРУЖЕНИЯ ДУБЛИКАТОВ НА ОСНОВЕ ЛОКАЛЬНЫХ ДИФФЕРЕНЦИРУЮЩИХ ШАБЛОНОВ

Н.И. Евдокимова, А.В. Кузнецов

Самарский государственный аэрокосмический университет имени академика С.П. Королёва (национальный исследовательский университет) (СГАУ), Самара, Россия

Встраивание дубликатов является одним из часто применяемых способов подделки изображений, при котором некоторый фрагмент изображения копируется в другую его часть с целью сокрытия какой-либо информативной части. Целью алгоритма обнаружения дубликатов является выявление областей дубликатов на изображении. В основе алгоритма обнаружения дубликатов лежит вычисление признаков в скользящем окне или окне с перекрытиями. Данная работа посвящена описанию алгоритма обнаружения дубликатов с использованием признаков на основе локальных дифференцирующих шаблонов. В части исследований приводятся результаты анализа точности обнаружения дубликатов при помощи предложенного метода. Отличительной особенностью используемых признаков является устойчивость к вносимым в дубликат искажениям и низкая вычислительная сложность.

**Ключевые слова:** дубликат, искажение, локальный дифференцирующий шаблон, локальный бинарный шаблон.

## Введение

В настоящее время цифровые изображения и видео являются одними из основных средств передачи информации. Например, они используются для подтверждения каких-либо событий в СМИ. Широкая доступность программного обеспечения привела к тому, что осуществить подделку изображения можно очень просто. Для обеспечения безопасности данных исследователи на протяжении последних 10-15 лет разрабатываются алгоритмы обнаружения искусственных искажений на цифровых изображениях [1-3]. С помощью таких методов можно определять области изображения, повергавшиеся изменениям, а также характеристики вносимых злоумышленниками искажений.

Встраивание дубликатов является одним из самых часто используемых методов подделки изображений в силу простоты его использования. Процесс встраивания состоит из трёх последовательных этапов: копирование некоторой области изображения, искажение этой области и встраивание её в той части этого же изображения, которую необходимо скрыть. Несмотря на большое количество существующих решений, многие из них не являются вычислительно эффективными. Именно поэтому на сегодняшний день разработка методов, позволяющих быстро и с высокой точностью обнаруживать изменённые области, является актуальной.

Работа построена следующим образом. В первой части рассмотрены основные сведения о локальном бинарном шаблоне (ЛБШ) и локальном дифференцирующем шаблоне (ЛДШ). Вторая часть посвящена описанию алгоритма обнаружения дубликатов. Третья часть содержит в себе результаты экспериментальных исследований на наборе изображений с дубликатами.

## 1. Локальные шаблоны

### Локальный бинарный шаблон

Идея ЛБШ принадлежит Wang [4]. В базовой версии для построения ЛБШ используется окно  $3 \times 3$ . Восемь соседних значений сравниваются со значением центрального пикселя, и в зависимости от результата сравнения центральному пикселю ставится в соответствие восьмибитное число – код ЛБШ (рис.1).

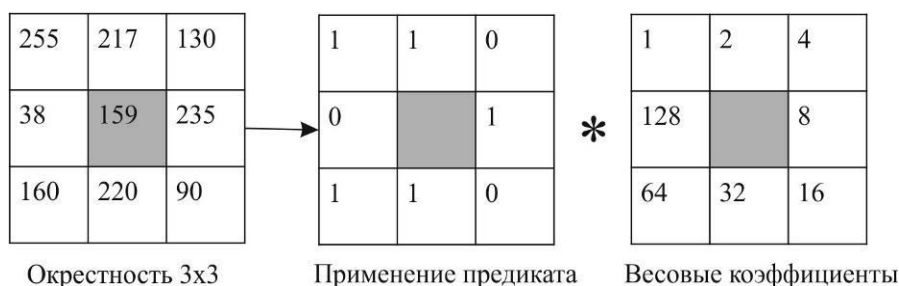


Рис.1. Базовая версия оператора ЛБШ

Определим изображение как функцию яркости пикселя  $f(x, y)$ . Код ЛБШ вычисляется в каждом пикселе с использованием значений пикселей в круговой окрестности  $f_i = f(x_i, y_i), i = \overline{1, P}$ , где  $P$  – количество равноудаленных от центрального пикселя  $f_0$  соседей, расположенных на окружности радиуса  $R$  (см. рис. 2) следующим образом:

$$LBP(N, R) = \sum_{i=0}^{N-1} I_1(f_i - f_0) \cdot 2^i, \quad (1)$$

где  $I_1$  – индикатор события, определяемый следующим образом:

$$I_1(m) = \begin{cases} 0, & m < 0 \\ 1, & m \geq 0 \end{cases} \quad (2)$$

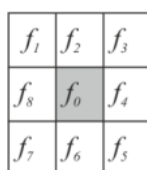


Рис.2. Пример окрестности  $3 \times 3$  вокруг пикселя  $f_0$  при  $R=1$

Следует также отметить, что метод ЛБШ очень эффективен благодаря своей низкой вычислительной сложности. Главным недостатком данного вида шаблонов является чувствительность к шумовым искажениям [5].

### Локальный дифференцирующий шаблон

Код ЛБШ представляет из себя результат применения производной первого порядка к окрестности центрального пикселя, как было показано в (1). В целях более информативного описания изображения в [6] был предложен оператор ЛДШ, который позволяет кодировать информацию, получаемую в результате вычисления производных более высо-

кого порядка. Используемая в данном методе производная  $n$ -го порядка по различным направлениям вычисляется на основе функции бинарного кодирования.

Производную первого порядка по направлениям  $0^\circ, 45^\circ, 90^\circ$  и  $135^\circ$  будем обозначать как  $f'_\alpha(x_0, y_0)$ , где  $\alpha = 0^\circ, 45^\circ, 90^\circ, 135^\circ$ .

Производные первого порядка в точке  $(x_0, y_0)$  могут быть записаны в следующем виде:

$$\begin{aligned} f'_0(x_0, y_0) &= f_0 - f_4, & f'_{45}(x_0, y_0) &= f_0 - f_3, \\ f'_{90}(x_0, y_0) &= f_0 - f_2, & f'_{135}(x_0, y_0) &= f_0 - f_1. \end{aligned} \quad (3)$$

ЛДШ второго порядка по направлениям  $\alpha$  в точке  $(x_0, y_0)$  определяется следующим выражением:

$$LDP^2_\alpha(x_0, y_0) = \left\{ \begin{array}{l} I_2(f'_\alpha(x_0, y_0), f'_\alpha(x_1, y_1)) \\ I_2(f'_\alpha(x_0, y_0), f'_\alpha(x_2, y_2)) \\ \dots \\ I_2(f'_\alpha(x_0, y_0), f'_\alpha(x_8, y_8)) \end{array} \right\} \quad (4)$$

где  $I_2$  – индикатор события, определяемый следующим образом:

$$I_2(m, n) = \begin{cases} 0, & m \cdot n < 0 \\ 1, & m \cdot n \geq 0 \end{cases} \quad (5)$$

Окончательно ЛДШ второго порядка определяется как объединение четырёх восьмибитных ЛДШ, вычисленных по заданным направлениям:

$$LDP^2(x, y) = \{LDP^2_\alpha(x, y) \mid \alpha = 0^\circ, 45^\circ, 90^\circ, 135^\circ\} \quad (6)$$

Как видно из приведённых выше уравнений, оператор ЛДШ присваивает код каждому пикселю изображения путём сравнения двух производных по направлению в двух соседних пикселях. Далее следует объединение полученных кодов в единую последовательность из 32 бит. Сравнения производных по направлению выполняются с помощью 16 примитивных шаблонов [6] (рис. 3), характеризующих отличительные пространственные отношения в локальной области. В отличие от ЛБШ, ЛДШ второго порядка позволяет кодировать изменение значения производной по направлению, которое является характеристикой локальной области изображения.

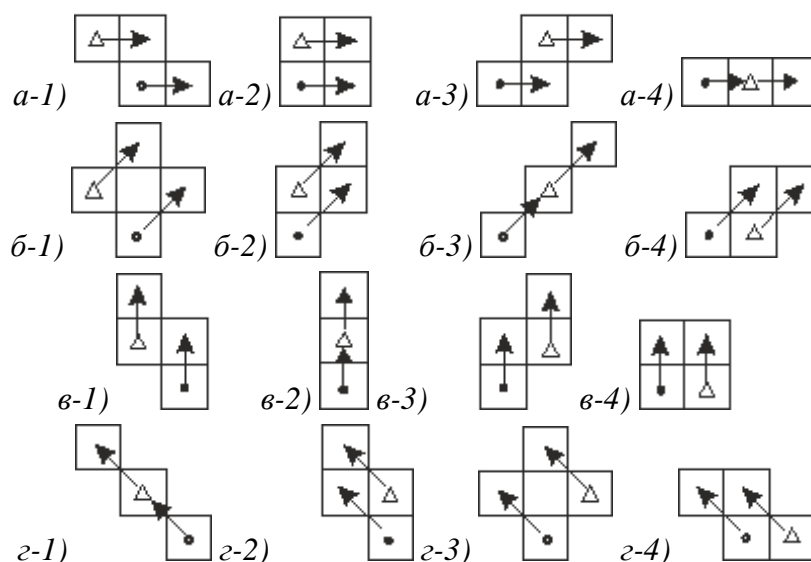


Рис.3. Прimitives шаблоны, используемые в ЛДШ

Рис. 4 иллюстрирует типы локальных переходов в примитивных шаблонах, используемых в ЛДШ. Переходы представляют из себя бинарные значения. Каждый из 16 примитивных шаблонов на рис. 3 принадлежит одной из двух групп: трёхпиксельные (3-point) или четырёхпиксельные (4-point). Трёхпиксельному шаблону в соответствии с (9) присваивается значение 0, если он является монотонно возрастающим или убывающим (рис. 4в), в противном случае шаблону присваивается значение 1 (рис. 4а). Аналогично, четырёхпиксельному шаблону присваивается значение 1, если он представляет из себя «поворот градиента» (рис. 4б), и значение 0, если он монотонно возрастает или убывает (рис. 4г). Таким образом, при помощи локальных переходов на основе примитивных шаблонов формируется бинарный код области изображения.

В качестве примера рассмотрим некоторую область изображения, представленную на рис. 5.

Для вычисления ЛДШ второго порядка по направлению  $\alpha = 0^\circ$  в точке  $f_0$ , четыре шаблона из рис. 3а применяются к изображению путём совмещения \* и  $\Delta$  с положением  $f_0$  последовательно. При применении шаблона а-1 (рис. 3а) путём совмещения \* с  $f_0$ , обе производные по направлению монотонно возрастают, как это показано на рис. 4г-1. Таким образом, примитивному шаблону соответствует значение 0. Аналогично, применение шаблонов из а-2, а-3 и а-4 (рис. 3а) путём совмещения \* с  $f_0$  соответствует случаям на рис. 4б-1, рис. 4г-1 и рис. 4а-2 соответственно. В результате вычисляются следующие три бита ЛДШ 101. Повторяя описанную выше процедуру с теми же четырьмя шаблонами, но на этот раз совмещая  $\Delta$  с  $f_0$ , мы получаем последовательность бит 0100 для последних четырёх битов. В результате формируется код 01010100 для направления  $\alpha = 0^\circ$ . Очевидным образом формируются коды примитивных шаблонов по направлениям  $\alpha = 45^\circ, 90^\circ, 135^\circ$ , представленных на рис. 3б-3г соответственно. В итоге получаем 32-битный код:

$$LDP^2(f_0) =$$

$$0101010000\ 1011111101\ 0000110001\ 10,$$

который генерируется путём объединения четырёх 8-битных ЛДШ, вычисленных по направлениям  $\alpha = 0^\circ, 45^\circ, 90^\circ, 135^\circ$  в соответствии с выражением (6).

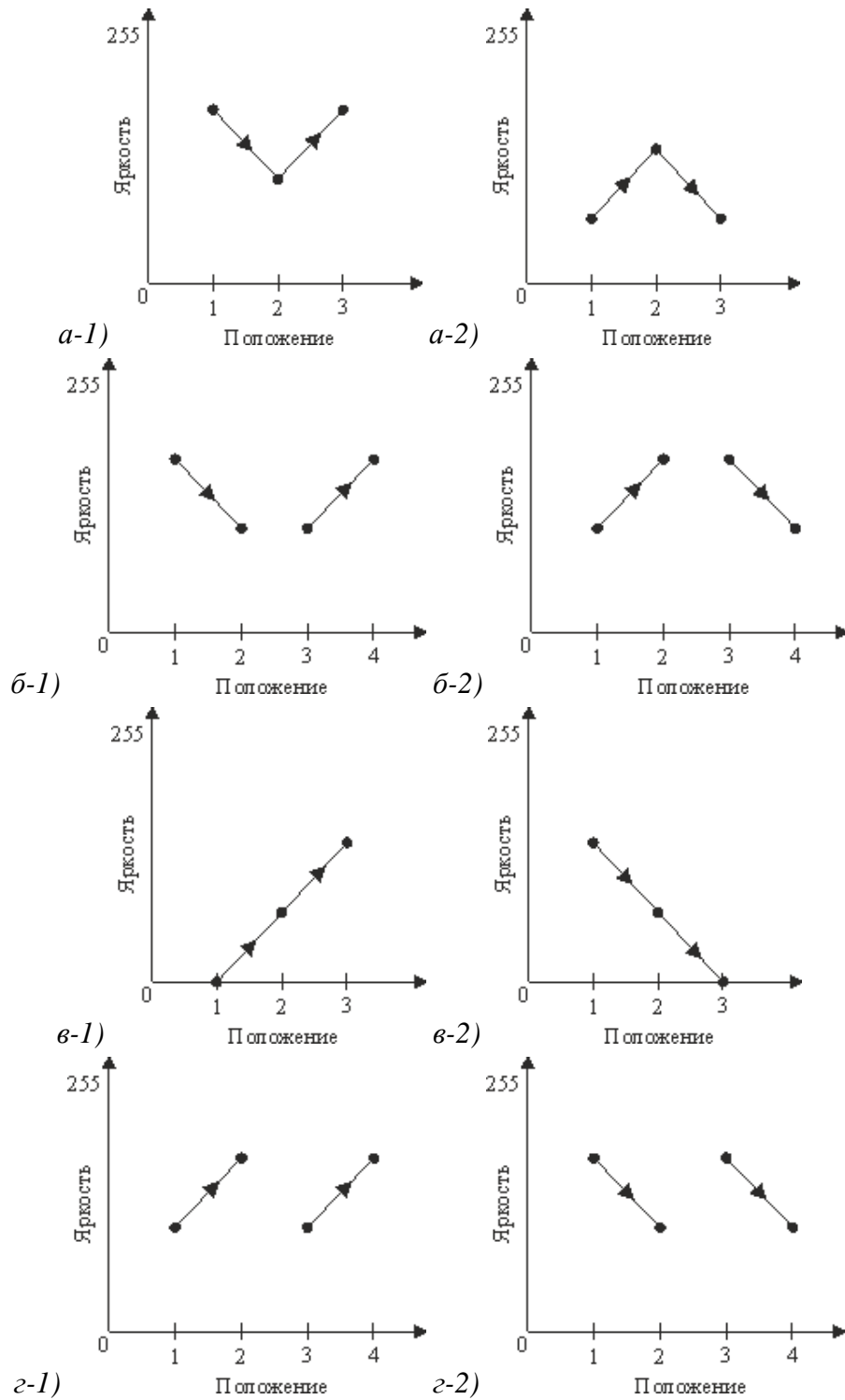


Рис.4. Типы локальных переходов, используемых для кодирования примитивных шаблонов ЛДШ

122	125	123	125	121
126	127	129	121	125
122	123	124	128	122
123	122	123	122	129
121	122	123	122	121

Рис.5. Область изображения,  $f_0 = 124$ *Локальный дифференцирующий шаблон n-го порядка*

Вычисляя производные  $n$ -го порядка по направлениям  $\alpha = 0^\circ, 45^\circ, 90^\circ, 135^\circ$  для пикселя  $f_0$ , можно получить ЛДШ  $n$ -го порядка, который определяется следующим образом:

$$LDP_\alpha^n(x_0, y_0) = \left\{ \begin{array}{l} I_2(f_\alpha^{n-1}(x_0, y_0), f_\alpha^{n-1}(x_1, y_1)) \\ I_2(f_\alpha^{n-1}(x_0, y_0), f_\alpha^{n-1}(x_2, y_2)) \\ \dots \\ I_2(f_\alpha^{n-1}(x_0, y_0), f_\alpha^{n-1}(x_8, y_8)) \end{array} \right\}, \quad (7)$$

где  $f_\alpha^{n-1}(x_0, y_0)$  – производная  $(n-1)$ -го порядка по направлению  $\alpha$  в точке  $(x_0, y_0)$ , которая описывает изменения градиента  $(n-1)$ -го порядка, предоставляя дополнительную информацию о локальной окрестности. Окончательно ЛДШ  $n$ -го порядка определяется следующим выражением:

$$LDP^n(x, y) = \{LDP_\alpha^n(x, y) \mid \alpha = 0^\circ, 45^\circ, 90^\circ, 135^\circ\} \quad (8)$$

Таким образом, ЛДШ также строится на основе производных по четырём направлениям.

Локальные шаблоны высокого порядка позволяют более детально описывать локальную окрестность изображения, чем локальный шаблон первого порядка, который используется в ЛБШ. Тем не менее, они являются более чувствительными к шуму в случае роста порядка производной. В работе [6] авторы экспериментально получили, что чувствительность к шуму у ЛДШ повышается при  $n \geq 4$ . Учитывая это ограничение, в дальнейшем будем использовать только ЛДШ второго и третьего порядков (ЛДШ-2 и ЛДШ-3).

**2. Алгоритм обнаружения дубликатов**

В рамках данной работы будем использовать алгоритм, предложенный в работах Кузнецова А.В., Мясникова В.В. и Глумова Н.И. [7-8]. Алгоритм состоит из следующих основных этапов:

1. В режиме скользящего окна с размером  $5 \times 5$  производится вычисление значений кодов локальных шаблонов (ЛБШ или ЛДШ). Значения помещаются в матрицу кодов.
2. В режиме скользящего окна производится вычисление значений хэш-функции (отображение значений матрицы кодов в некоторую целочисленную неотрицательную величину или хэш-значение) [8].
3. Помещение хэш-значений в матрицу хэш-значений.

4. Построение гистограммы хэш-значений.
5. Анализ гистограммы, заключающийся в поиске таких хэш-значений, для которых значение в гистограмме превышает единицу.
6. Формирование выходного изображения, содержащего выделенные области дубликаты.

### 3. Экспериментальные исследования

Для проведения экспериментальных исследований авторами используется стандартный ПК (Intel Core i5-3470 3.2. ГГц, 8 Гб ОЗУ). В качестве объектов исследования были выбраны 10 полутоновых цифровых изображений с размерами  $512 \times 512$ . Авторами была разработана процедура генерации изображений, содержащих дубликаты, подвергавшиеся искажениям (линейное контрастирование и аддитивный белый шум). Сравнение проводилось между признаками на основе ЛБШ и на основе ЛДШ в рамках алгоритма обнаружения дубликатов.

Для оценки качества разработанного алгоритма мы использовали величину  $F1\_Score$ , которая связывает точность обнаружения и ошибки первого и второго рода:

$$F1 = \frac{2 \cdot tp}{2 \cdot tp + fp + fn}. \quad (9)$$

Результаты проведённых экспериментов на наборе изображений показаны в Таблицах 1 и 2.

Табл 1. Исследование алгоритмов обнаружения искажений при размерах встраивания  $128 \times 128$

Тип искажения	ЛБШ	ЛДШ-2	ЛДШ-3
<b>Контрастирование</b>	0.87	0.93	0.96
<b>Белый шум</b>	0.84	0.91	0.95

Табл 2. Исследование алгоритмов обнаружения искажений при размерах встраивания  $64 \times 64$

Тип искажения	ЛБШ	ЛДШ-2	ЛДШ-3
<b>Контрастирование</b>	0.82	0.91	0.96
<b>Белый шум</b>	0.80	0.89	0.94

### Выводы

В рамках проведённых исследований был разработан алгоритм на основе ЛДШ, позволяющий осуществлять обнаружение искажённых дубликатов на цифровых изображениях. Ввиду специфики локальных шаблонов, алгоритм имеет низкую вычислительную сложность, что позволяет применять его для анализа больших изображений (например, космических снимков). За рамками работы остались исследования обнаружения дубликатов, подвергавшихся геометрическим искажениям и компрессии. В дальнейшем также планируется сравнить качество обнаружения дубликатов с существующими решениями на основе признакового описания.

## Благодарности

Работы по созданию алгоритма обнаружения дубликатов на основе ЛДШ получены в рамках государственного задания вузу №2014/198 (код проекта 2298) и гранта РФФИ 16-37-00056 мол\_а.

## Литература

1. Christlein, V. An Evaluation of Popular Copy-Move Forgery Detection Approaches / V. Christlein, C. Riess, J. Jordan, E. Angelopoulou // *IEEE Transactions on information forensics and security*. – 2012. – Vol. 7(6), P. 1841-1854.
2. Popescu, A. Exposing digital forgeries by detecting duplicated image regions / A. Popescu, H. Farid [Электронный ресурс]. – 2004. – URL: <http://www.ists.dartmouth.edu/library/102>.
3. Fridrich J. Detection of copy-move forgery in digital images / J. Fridrich, D. Soukal, J. Lukas. // *Proceedings of Digital Forensic Research Workshop*, August 2003.
4. Wang, L. Texture classification using texture spectrum / L. Wang, D.-C. He // *Pattern Recognition*. – 1990. – Vol. 23(8), P. 905-910.
5. Ren, J. Noise-Resistant Local Binary Pattern With an Embedded Error-Correction Mechanism / J. Ren, X. Jiang, J. Yuan // *IEEE Transactions on Image Processing*. – 2013. – Vol. 22(10), P. 4049-4060.
6. Zhang, B. Local derivative pattern versus local binary pattern: face recognition with high-order local pattern descriptor / B. Zhang, Y. Gao, S. Zhao, J. Liu // *IEEE Transactions on Image Processing*. – 2010. – Vol. 19(2), P. 533-544.
7. Глумов, Н.И. Поиск дубликатов на цифровых изображениях / Н.И. Глумов, А.В. Кузнецов, В.В. Мясников // *Компьютерная оптика*. –2013. – Т. 37, № 3. – С. 360-367.
8. Kuznetsov, A. A Fast Plain Copy-Move Detection Algorithm Based on Structural Pattern and 2D Rabin-Karp Rolling Hash / A. Kuznetsov, V. Myasnikov // *Lecture Notes in Computer Science*. – 2014. – Vol. 8814(1), P. 461-468.