

УДК 004.928

Е.А. Альтман, Е.И. Захаренко

## Быстрый алгоритм вычисления двумерной корреляции для видеообработки

Предложен быстрый алгоритм вычисления двумерной корреляции сигналов размерами  $N \times N$  и  $(N + M - 1) \times (N + M - 1)$ , где  $N = 2^i$ ,  $i$  и  $M$  – целые числа. Метод построен на принципе рекурсивного разбиения корреляции на более короткие длины и применении на каждом шаге разбиения наиболее эффективного алгоритма. Вычислительная эффективность различных методов корреляции выявлена в результате представленного в статье исследования. Также авторами разработан еще один быстрый алгоритм вычисления двумерной корреляции, разбивающий ее на 12. Этот метод эффективен при длине сигнала  $4 \times 4$  и применяется на одном из шагов рекурсивного разбиения предложенного алгоритма. В результате исследования выявлено, что предложенный алгоритм рекурсивного разбиения двумерной корреляции эффективен для коротких сигналов (менее  $32 \times 32$ ). Применение этого метода для решения задач видеообработки позволяет снизить вычислительную сложность блочной оценки движения полным перебором вдвое по сравнению с вычислением по прямой формуле для блока размером  $8 \times 8$  точек и на 3% по сравнению с наиболее быстрыми из известных методов двумерной корреляции.

**Ключевые слова:** двумерная корреляция, оценка движения, суммарная квадратичная разность, видеокодирование.

Популярность графических средств представления и передачи информации возрастает. В связи с этим повышаются требования к качеству видеоизображения и размеру видеофайла. Задача получения видеоизображения высокого визуального качества при высокой степени его компрессии решается кодированием. Одним из основных этапов кодирования, оказывающих наибольшее влияние на качество и степень сжатия видео, является оценка движения, которая была предложена как часть видеокодека в 1960-х гг. Современные стандарты видеокодирования MPEG-4 Visual и H.265 предлагают применять блочную оценку движения (block matching algorithm – *BMA*). При этом кадр разбивается на непересекающиеся прямоугольные (квадратные) блоки, и осуществляется сравнение каждого блока текущего кадра с несколькими блоками внутри области поиска предыдущего/последующего (ссылочного) кадра. Чем меньше ошибка в определении схожего блока, тем точнее оценка движения, следовательно, выше визуальное качество изображения и степень компрессии видеофайла [1, 2].

Эталонным или стандартным и наиболее точным методом *BMA* является полный перебор (full search – *FS*) [1–3]. В период 1981–2000-х гг. зарубежными учеными Т. Koga, J.R. Jain, M. Ghanbari, R. Li, L.M. Po, S. Zhu были предложены быстрые алгоритмы оценки движения, получившие реализацию в нескольких видеокодеках, одним из которых является трехшаговый поиск (three step search – *TSS*). Недостатками этих методов является поиск локального, а не глобального экстремума функции сравнения двух блоков, что приводит к снижению точности оценки движения по сравнению с полным перебором и, следовательно, к деградации визуального качества видео [4]. Обладая наибольшей точностью, *FS* требует высоких вычислительных ресурсов, что затрудняет его применение для решения практических задач.

Метрика или функция сопоставления блоков не регламентированы видеостандартами, что делает ее предметом научного исследования. Наиболее популярные метрики сравнения блоков появились в 1981 г. вместе с быстрыми методами оценки движения в работах Т. Koga и J.R. Jain и сохраняют свою актуальность на сегодняшний день. Исследования Y. Naito, H.G. Musmann и других ученых посвящены проблеме неэффективного вычисления этих функций [3, 5]. Эффективность и сложность математических алгоритмов можно измерять различными способами. В статье, как и в некоторых литературных источниках [6–8], для этих целей выбрано количество арифметических операций. Такой подход к измерению сложности алгоритма легко осуществим и связан со скоростью его работы при аппаратной реализации [8].

Текущий блок и совокупность блоков в ссылочной области поиска являются двумерными дискретными сигналами. Наиболее популярный способ сравнения таких сигналов – это их корреляция, или свертка. Корреляция предполагает сравнение одного сигнала со всеми возможными позициями второго внутри определенной области. Таким условиям удовлетворяет только оценка движения полным перебором. Применение эффективного алгоритма двумерной корреляции для сопоставления блоков повысит быстродействие метода *FS*.

Один из эффективных методов вычисления корреляции состоит в использовании теоремы о свертке и дискретного преобразования Фурье. Быстрые алгоритмы преобразования Фурье (БПФ) описаны в работах J. Cooley, S. Winograd, R. Blahut, Г. Нуссбаумера, Л.М. Гольденберга, А. Оппенгейма и других ученых. Еще один класс быстрых методов вычисления корреляции состоит в ее разложении на более короткие [5, 9, 10]. Каждый из существующих методов может быть эффективнее других для определенных задач и условий.

Для оценки движения размер блоков, на которые разбивается кадр, не превышает  $16 \times 16$  точек. В стандарте видеокодирования MPEG-4 Visual регламентирован размер блока, равный  $8 \times 8$  точек. Стандарт H.264 предполагает разбиение макроблока  $16 \times 16$  точек на блоки разного размера –  $4 \times 4$ ,  $8 \times 8$ . Они сравниваются по яркостным  $Y$ -компонентам точек (пикселей) изображения в цветовом пространстве  $YCbCr$ .  $Y$ -компонента представляется целым числом в интервале от 0 до 255 лм. Для всех форматов сэмплирования цветового пространства  $YCbCr$  размер матрицы  $Y$  равен разрешению изображения (кадра, блока). Поэтому для оценки движения интерес представляют алгоритмы двумерной корреляции для сигналов короткой и четной длины, состоящих из целых положительных чисел.

**Блочная оценка движения.** Для сопоставления блоков при оценке движения предложено несколько функций, описанных в работах [11, 12]. Одной из популярных метрик является суммарная квадратичная разность (sum of square difference – *SSD*):

$$SSD(i, j) = \sum_{y=0}^{N_h-1} \sum_{x=0}^{N_w-1} (B(x, y) - S(x+i, y+j))^2, \quad (1)$$

где  $i, j$  – координаты вектора движения относительно текущего блока,  $i$  изменяется в интервале от  $-S_w/2$  до  $S_w/2$ ,  $j$  изменяется в интервале от  $-S_h/2$  до  $S_h/2$ ;  $x, y$  – координаты точки блока;  $N_w \times N_h$  – размер блока;  $B$  – текущий блок;  $S$  – ссылочный блок размером  $S_w \times S_h$ .

Выражение (1) может быть разложено на три слагаемых по формуле квадрата разности. Для оценки движения методом полного перебора одно из слагаемых *SSD* является корреляцией [3, 5].

Корреляция может применяться для сравнения двух блоков как самостоятельная метрика [11]:

$$COR(i, j) = \sum_{y=0}^{N_h-1} \sum_{x=0}^{N_w-1} B(x, y)S(x+i, y+j), \quad (2)$$

где  $i, j$  – координаты вектора движения относительно текущего блока,  $i$  изменяется в интервале от  $-S_w/2$  до  $S_w/2$ ,  $j$  изменяется в интервале от  $-S_h/2$  до  $S_h/2$ ;  $x, y$  – координаты точки блока;  $N_w \times N_h$  – размер блока;  $B$  – текущий блок;  $S$  – ссылочный блок размером  $S_w \times S_h$ .

Для практического применения оценки движения полным перебором необходимо сократить время его работы, которое определяется вычислительной сложностью. Одним из способов решения такой задачи является применение эффективного метода двумерной корреляции для вычисления значений функции сопоставления блоков.

В качестве критерия эффективности и сложности алгоритма выбрано количество арифметических операций. Для вычисления корреляции арифметические операции можно свести к двум: сложение и умножение [6–8]. В некоторых процессорах помимо этих операций в одном цикле выполняется умножение с накоплением (multiply and accumulate – *MAC*). В работе для исследования алгоритмов вычислительная сложность рассчитывается для двух случаев: с учетом и без учета *MAC*-операций.

**Быстрые методы вычисления двумерной корреляции.** Двумерную корреляцию можно вычислить по прямой формуле (2). Еще один способ нахождения двумерной корреляции через одномерную: вычисление каждой строки результата через сумму столбцов матрицы, полученной в результате одномерной корреляции каждой строки одного сигнала со всеми строками второго сигнала. В этом случае для двумерной корреляции сигнала размером  $N \times N$  необходимо  $N^2$  раз вычислить одномерную корреляцию и  $N$  раз – их сумму.

В 1987 г. Z. J. Mou и P. Duhamel был предложен быстрый метод поиска значений одномерной корреляции, который состоит в разбиении одномерных последовательностей  $x$  и  $y$  на четные и нечетные отсчеты, предварительном сложении полученных сигналов, вычислении коротких корреляций и их сложении. Вычислительная сложность этого способа на 25% меньше прямого метода [9].

На основании одномерного алгоритма Y. Naito, T. Miyazaki и I. Kuroda разработали метод вычисления корреляции для двумерных сигналов  $x$  размером  $N \times N$  и  $y$   $(N + M - 1) \times (N + M - 1)$  [5]. Аналогично одномерному случаю двумерные последовательности  $x$  и  $y$  делятся по четным и нечетным строкам и столбцам:

$$X_{i,j} = [\hat{x}_{i,j}, \hat{x}_{i,j+2}, \dots, \hat{x}_{i,j+N-2}]^T; Y_{i,j} = [\hat{y}_{i,j}, \hat{y}_{i,j+2}, \dots, \hat{y}_{i,j+N-2}], \quad (3)$$

где  $\hat{x}_{i,j} = [x_{i,j}, x_{i,j+2}, \dots, x_{i,j+N-2}]$ ,  $\hat{y}_{i,j} = [y_{i,j}, y_{i,j+2}, \dots, y_{i,j+N-2}]$ ,  $i, j \in [0; N/2]$

Каждый элемент векторов  $X_{i,j}$  и  $Y_{i,j}$  является матрицей элементов в позиции  $(i, j)$  текущего и ссылочного блока соответственно.

Данный метод требует по пять поэлементных сложений для  $X_{i,j}$  и  $Y_{i,j}$  в различных позициях  $i, j$  сигналов  $x$  и  $y$ , последующих девяти корреляций результатов сложения. Корреляция является рекурсивной функцией и вызывается, пока входной сигнал не будет размером  $2 \times 2$ . Затем выполняются умножения и сложения результатов корреляции и формирование результата двумерной корреляции  $x$  и  $y$  путем заполнения четных и нечетных отсчетов полученными значениями. Более подробное описание алгоритма приведено в работе [5].

Авторами статьи предложен еще один быстрый метод поиска значений двумерной корреляции. Предлагаемый способ аналогичен описанному выше алгоритму. Входной сигнал разделяется по четным и нечетным строкам и столбцам, выполняется по два сложения для  $X_{i,j}$  и  $Y_{i,j}$  и рекурсивно вызываются 12 корреляций:

$$\begin{aligned} C0_{i,j} &= (Y_{i+1,j+1} - Y_{i,j+1})A_0; & C1_{i,j} &= (Y_{i+1,j+1} - Y_{i,j})A_1; & C2_{i,j} &= (Y_{i+1,j+1} - Y_{i,j+1})A_1; \\ C3_{i,j} &= (Y_{i+1,j+1} - Y_{i,j})A_0; & C4_{i,j} &= Y_{i+1,j}X_{i,j}; & C5_{i,j} &= Y_{i+1,j+1}X_{i,j+1}; & C6_{i,j} &= Y_{i+1,j+1}X_{i,j}; \\ C7_{i,j} &= Y_{i+1,j+2}X_{i,j+1}; & C8_{i,j} &= Y_{i+2,j}X_{i+1,j}; & C9_{i,j} &= Y_{i+2,j+1}X_{i+1,j+1}; & C10_{i,j} &= Y_{i+2,j+1}X_{i+1,j}; \\ C11_{i,j} &= Y_{i+2,j+2}X_{i+1,j+1}, \end{aligned} \quad (4)$$

где  $A_0 = X_{0,0} - X_{1,0}$ ,  $A_1 = X_{0,1} - X_{1,1}$ , подстрочный индекс 0 сигнала  $X$  означает четную позицию сигнала  $x$ , 1 – нечетную.

По сравнению с алгоритмом через девять корреляций этот способ вычисления двумерной корреляции требует меньше предварительных сложений сигналов и объединений коротких корреляций, поэтому может быть вычислительно более эффективным для некоторых  $N$ .

Для исследования вычислительной эффективности (сложности/скорости) алгоритмов двумерной корреляции рассматриваются четыре алгоритма: рекурсивное разложение на девять (2D Fast 9 convol) и 12 корреляций короткой длины (2D Fast 12 convol), вычисление двумерной корреляции через одномерную (2D Full – 1D Full), прямое вычисление по формуле (2); и два сигнала:  $x$  (текущий блок  $B$ ) размером  $N \times N$  и фильтр  $y$  (область поиска  $S$  на ссылочном кадре) размерами  $(2N - 1) \times (2N - 1)$  и  $(3N - 1) \times (3N - 1)$ , где  $N = 2^i$ ,  $i$  – целое число от 1 до 6. В условиях рассматриваемой области применения исследования сигналы  $x$  и  $y$  являются яркостной компонентой пикселя изображения, поэтому заполняются случайными целыми числами от 0 до 255.

В качестве критерия вычислительной эффективности алгоритма используется количество сложений и умножений.

В результате исследования выявлены наиболее быстрые алгоритмы для исследуемых длин фильтра  $N = 2^i$ ,  $i$  – целое число от 1 до 6 (см. табл. 1). При  $N \geq 8$  наиболее быстрым из исследуемых четырех алгоритмов оказался метод разложения на девять корреляций. Для  $N = 2$  и  $N = 4$  эффективны алгоритм вычисления двумерной корреляции через одномерную и метод разложения на 12 корреляций соответственно.

**Быстрый алгоритм вычисления двумерной корреляции рекурсивным разложением.** Алгоритмы рекурсивного разложения двумерной корреляции на короткие – наименее сложные из исследуемых методов при длине фильтра более 2 (табл. 1). Каждый рекурсивный шаг сокращает длину фильтра и сигнала вдвое, так как делит их по четным и нечетным отсчетам. Это позволяет на каждом шаге разбиения применять наиболее быстрый алгоритм, что повышает эффективность вычисления двумерной корреляции.

На основе этого предположения авторами предложен быстрый алгоритм вычисления двумерной корреляции сигналами размерами  $N \times N$  и  $(N + M - 1) \times (N + M - 1)$ , где  $N = 2^i$ ,  $i, M$  – целые положительные числа. В этом случае метод состоит из  $\log_2 N$  шагов, на каждом из которых выбирается наиболее эффективный алгоритм, выявленный в результате приведенного выше исследования. Если  $N \geq 8$ , применяется метод разложения на девять корреляций. Когда путем рекурсивного разбиения корреляции достигается длина сигнала  $N = 4$ , то используется метод разложения на 12 корреляций, при  $N = 2$  – алгоритм вычисления двумерной корреляции через одномерную.

Вычислительная эффективность этого метода исследовалась для блока длиной  $N \times N$  и ссылочной области размерами  $(2N - 1) \times (2N - 1)$  и  $(3N - 1) \times (3N - 1)$ , где  $N = 2^i$ ,  $i$  – целое число от 1 до 6. С помощью предложенного алгоритма вычислена корреляция, которая является слагаемым метрики сравнения двух блоков *SSD*. Сравнение поиска значений *SSD* полным перебором *FS* через предложенный авторами метод с вычислением через наиболее быстрый из известных алгоритмов для сигнала размером  $(3N - 1) \times (3N - 1)$ , где  $N = 2^i$ ,  $i$  – целое число от 1 до 6, представлено в табл. 1.

Таблица 1

**Вычислительная сложность *SSD* полным перебором *FS* через предложенный алгоритм двумерной корреляции блока *B* размером  $N \times N$  и области поиска *S* размером  $(3N - 1) \times (3N - 1)$**

$N$	Количество арифметических операций <i>SSD</i> через предложенный алгоритм	Самый быстрый алгоритм вычисления двумерной корреляции	Количество арифметических операций <i>SSD</i> через самый быстрый алгоритм двумерной корреляции	Вычислительная сложность <i>SSD</i> через предложенный алгоритм относительно самого быстрого, %
2	207	2D Full – 1D Full	207	100,0
4	2 106	2D Fast 12 convol	2 178	96,7
8	20 355	2D Fast 9 convol	21 003	96,9
16	195 696	2D Fast 9 convol	201 528	97,1
32	1 874 109	2D Fast 9 convol	1 926 597	97,3
64	17 837 154	2D Fast 9 convol	18 309 546	97,4

Разработанный способ является более эффективным по сравнению с самыми быстрыми методами при длине сигнала более  $2 \times 2$ . При этом наибольшая эффективность вычисления *SSD* через предложенный алгоритм достигается при коротких  $N$ : 4 и 8. Когда размер области поиска  $(2N - 1) \times (2N - 1)$ , применение описанного алгоритма для вычисления *SSD* дает результаты, аналогичные приведенным в табл. 1, т.е. позволяет сократить вычислительную сложность метода полного перебора *FS* на 3% относительно применения наиболее эффективных алгоритмов вычисления двумерной корреляции.

В современной вычислительной технике реализована операция совмещенного умножения и сложения *MAC*. В этом случае при определении вычислительной сложности алгоритма учитываются сложения, умножения и *MAC*-операции. Тогда наибольшую эффективность дает метод, в котором для длин фильтра  $2 \times 2$  и  $4 \times 4$  двумерная корреляция вычисляется через одномерную (2D Full – 1D Full), для  $N = 8, 16$  и  $32$  применяется алгоритм преобразования в девять корреляций (2D Fast 9 convol). При учете *MAC*-операций вычислительная сложность полного перебора с использованием предложенного алгоритма также сокращается не более чем на 3% по сравнению с использованием самого эффективного алгоритма вычисления двумерной корреляции.

Одним из популярных способов вычисления корреляции является использование теоремы о свертке. В этом случае необходимо применять алгоритм дискретного преобразования Фурье (ДПФ). Один из самых быстрых ДПФ – алгоритм Винограда, что подтверждается в работе [1].

Авторами было произведено сравнение вычисления корреляции предложенным методом и алгоритмом Винограда по количеству арифметических операций (умножений и сложений). Результаты представлены в табл. 2.

Новый алгоритм показал большую вычислительную эффективность по сравнению преобразованием Винограда, когда размер сигнала меньше  $32 \times 32$ . В соответствии с популярными видеостандартами размер блока не превосходит  $16 \times 16$  точек, поэтому предложенный алгоритм может применяться для сокращения вычислительной сложности оценки движения полным перебором.

В видеокодеках и видеосистемах, работающих в реальном времени, применяются быстрые алгоритмы оценки движения по причине низкой вычислительной сложности. Исследование, приве-

денное в статье, посвящено выявлению вычислительно более эффективного алгоритма двумерной корреляции для повышения быстродействия оценки движения полным перебором. Это требует его сравнения с одним из быстрых методов – трехшаговым поиском *TSS* (табл. 3).

Таблица 2

**Вычислительная сложность двумерной корреляции через предложенный алгоритм и через теорему о свертке с использованием БПФ Винограда для двух сигналов длинами  $N \times N$  и  $(2N - 1) \times (2N - 1)$**

N	Количество арифметических операций (сложений и умножений)		Вычислительная сложность нового алгоритма относительно алгоритма Винограда, %
	Предложенный алгоритм	БПФ Винограда	
1	2	3	4
2	26	216	12
4	460	560	82
8	5 737	9 135	63
16	62 310	82 980	75
32	636 667	344 547	185
64	6 296 472	2 486 272	253

Таблица 3

**Сравнение вычислительной сложности оценки движения методом полного перебора по прямой формуле *SSD* с полным перебором *FS* через предложенный алгоритм и трехшаговым поиском *TSS***

N	Быстродействие относительно полного перебора <i>FS</i> по прямой формуле <i>SSD</i> (1), %			
	Область поиска $S(2N - 1) \times (2N - 1)$		Область поиска $S(3N - 1) \times (3N - 1)$	
	<i>FS</i> через предложенный алгоритм корреляции	быстрый алгоритм <i>TSS</i>	<i>FS</i> через предложенный алгоритм корреляции	быстрый алгоритм <i>TSS</i>
2	120,5	50,0	117,6	25,0
4	83,6	25,0	70,0	9,4
8	53,6	9,4	41,6	3,1
16	33,5	3,1	24,9	1,0
32	20,7	1,0	14,9	0,3
64	12,6	0,3	8,9	0,1

Оценка движения полным перебором с использованием предложенного метода вычисления метрики *SSD* выполняется быстрее, чем прямым вычислением по формуле (1), но медленнее, чем трехшаговым поиском *TSS*.

**Заключение.** В работе представлен новый способ разбиения двумерной корреляции на 12 меньшей длины, при применении которого построен быстрый алгоритм вычисления двумерной корреляции с различными рекурсивными способами разбиения при разных размерах сигнала. Предложенный алгоритм обладает большей вычислительной эффективностью при коротких длинах сигнала (менее  $32 \times 32$ ) по сравнению с другими методами. Это позволяет использовать его для блочной оценки движения на видео. При сопоставлении блоков размером  $8 \times 8$  точек применение разработанного алгоритма двумерной корреляции сокращает вычислительную сложность полного перебора *FS* вдвое по сравнению с вычислением по прямой формуле суммарной квадратичной разности *SSD*. Однако количество арифметических операций *FS* остается в несколько раз больше быстрого метода оценки движения трехшаговым поиском *TSS*. Полученный выигрыш в быстродействии дает возможность использовать приведенный метод в условиях, когда предъявляются высокие требования только к качеству видеоизображения, а не к вычислительной сложности алгоритма. Также новый алгоритм может применяться для решения задач цифровой фильтрации изображений и видеоданных, например, для реализации метода повышения визуального качества объекта на видеопоследовательности [13].

*Литература*

1. Обухова Н.А. Векторы оптического потока в задачах сегментации и сопровождения подвижных объектов // Изв. вузов России. Радиоэлектроника. – 2006. – Вып. 2. – С. 42–51.

2. Lee C.-H. A fast motion estimation algorithm based on the block sum pyramid / C.-H. Lee, L.-H. Chen // IEEE Trans. Image Processing. – 1997. – Vol. 6. – P. 1587–1591.
3. Зубарев Ю.Б. Методы анализа и компенсации движения в динамических изображениях / Ю.Б. Зубарев, В.П. Дворкович и др. // Электросвязь. – 1998. – № 11. – С. 15–21.
4. Cheung C.H. A novel cross-diamond search algorithm for fast block motion estimation / C.H. Cheung, L.M. Po // IEEE Trans. Circuits Syst. Video Technol. – 2002. – Vol. 12, № 12. – P. 1168–1177.
5. Naito Y. A Fast full-search motion estimation method for programmable processors with a multiply-accumulator / Y. Naito, T. Miyazaki, I. Kuroda // IEEE International Conference on Acoustics, Speech and Signal Processing. – 1996. – P. 3221–3224.
6. Блейхут Р. Быстрые алгоритмы цифровой обработки сигналов. – М.: Мир, 1989. – 448 с.
7. Нуссбаумер Г. Быстрое преобразование Фурье и алгоритмы вычисления свёрток. – М.: Радио и связь, 1985. – 248 с.
8. Оппенгейм А.В. Цифровая обработка сигналов / А.В. Оппенгейм, Р.В. Шафер. – М.: Техносфера, 2006. – 848 с.
9. Mou Z.J. Fast FIR filtering: algorithms and implementations / Z.J. Mou, P. Duhamel // Signal Processing. – 1987. – Vol. 13, № 4. – P. 377–384.
10. Рабинер Л. Теория и применение цифровой обработки сигналов / Л. Рабинер, Б. Гоулд. – М.: Мир, 1978. – 848 с.
11. Toivonen T. Number theoretic transform – based block motion estimation. – 2002. – 85 p.
12. Boltz S. A minimum-entropy procedure for robust motion estimation / S. Boltz, E. Wolsztynski, E. Debreuve et al. // IEEE International Conference Image Processing. – 2006. – P. 1249–1252.
13. Альтман Е.А. Высокопроизводительный метод повышения визуального качества изображения объекта на видеопоследовательности / Е.А. Альтман, Е.И. Захаренко // Омский научный вестник. Сер.: Приборы, машины и технологии. – 2013. – № 3 (123). – С. 247–250.

---

**Альтман Евгений Анатольевич**

Канд. техн. наук, доцент каф. автоматизации и систем управления (АиСУ)  
Омского государственного университета путей сообщения  
Тел.: 8 (381-2) 31-05-89  
Эл. почта: altmanea@gmail.com

**Захаренко Елена Игоревна**

Аспирант каф. АиСУ  
Тел.: 8 (381-2) 31-05-89  
Эл. почта: zaharenko.elena@gmail.com

Altman E.A., Zakharenko E.I.

**Fast algorithm for computing the two-dimensional correlation for video processing**

In this paper we propose a fast algorithm for calculating the two-dimensional correlation for a signal sizes  $N \times N$  and  $(N + M - 1) \times (N + M - 1)$ , where  $N = 2^i$ ,  $i$  and  $M$  – an integer. The method is based on the principle of the correlation recursive partitioning into shorter lengths and applying at each stage the most efficient partitioning algorithm. As a result the presented research revealed the effectiveness of various methods of computing the correlation. The authors have also developed another fast algorithm for calculating the two-dimensional correlation, partitioning it to 12. This method is effective when the length of a signal is  $4 \times 4$  and it is applied on one of the steps of the recursive partitioning of the proposed algorithm. The study found that the proposed algorithm for a recursive partitioning of a two-dimensional correlation is effective for short signals (less than  $32 \times 32$ ). The application of this approach for video processing reduces the computational complexity of full search block motion estimation by half in comparison with the straightforward approach for the block size of  $8 \times 8$  points and 3% compared to the fastest known method of two-dimensional correlation.

**Keywords:** two-dimensional correlation, motion estimation, sum of squared difference, video encoding.