

УДК 681.32

**Э. А. Мельник***(Кишинев, Молдова)***ПАРАЛЛЕЛЬНЫЕ АЛГОРИТМЫ ДВУМЕРНОЙ СВЕРТКИ**

Рассматриваются вопросы организации векторизованного процесса вычислений при реализации операции двумерной свертки изображений.

К операции свертки двумерных массивов сводятся многие типовые задачи обработки видеоинформации: согласованной фильтрации, подавления шумов, контрастирования, выделения контуров, корреляции и т. д. [1].

Однако эта операция, будучи массовой, одновременно и наиболее трудоемка с точки зрения вычислительных затрат. По этой причине одним из направлений в области повышения эффективности процедуры двумерной свертки является поиск алгоритмов, снижающих общее число требуемых для ее выполнения арифметических операций.

Существующие подходы, как правило, предполагают известным оператор свертки, имеющий вид матрицы специального типа и заданных размеров [2, 3]. Учет особенностей оператора свертки позволяет в конечном счете снизить необходимое число арифметических операций тем или иным образом. При этом неизбежно с помощью реорганизации управления вычислительным процессом приходится решать вопросы структурной перестройки обрабатываемых данных. По этой причине реализация такого подхода универсальными компьютерными средствами приводит к возрастанию вычислительных затрат на обмены и переупорядочение промежуточных массивов данных, что снижает ожидаемую от такого подхода эффективность.

Этого можно избежать с помощью аппаратно-микропрограммных средств, использующих быстродействующую микроэлектронную базу. При этом перестройка структуры данных организуется естественным распараллеливанием процессов их обработки. Наиболее эффективная реализация такого подхода – концепция систолической обработки информации [4–8].

Рассмотрим вопросы организации векторизованного процесса вычислений свертки изображений.

Пусть заданы два массива: массив исходных данных

$$X_N = \|x_{k,l}\|, \quad k, l = \overline{0, N-1},$$

и массив весовых коэффициентов

$$\Phi_M = \|\phi_{i,j}\|, \quad i, j = \overline{0, M-1}, \quad M < N.$$

Необходимо выполнить следующую операцию:

$$y_{k,l} = \sum_i \sum_j x_{k+i, l+j} \otimes \phi_{i,j}, \quad (1)$$

далее называемую двумерной скользящей сверткой массивов  $X_N$  и  $\Phi_M$  ( $\otimes$  – знак тензорного произведения матриц).

Матричная запись выражения (1), открывающая путь к прямой скалярно-конвейерной обработке, возможна лишь в том случае, когда оператор свертки линейно разделим. Если же этого нет, то приходится прибегать к искусственной векторизации двумерной свертки, для чего вводятся операции тензорного (кронекеровского, прямого) произведения матриц и операции коммутации, определяющие в совокупности маршрутизацию потоков входных и промежуточных данных в процессе параллельных вычислений [9].

Процесс векторизации, превращая двумерный массив в одномерный, позволяет впоследствии в полной мере использовать преимущества мощных скалярно-конвейерных процессов, не прерываемых программными контекстными переходами. Одно из решений данной задачи может быть представлено следующим образом.

Введем в рассмотрение вектор

$$X^{(0)} = [X^{(1)}, X^{(2)}, \dots, X^{(N-1)}]^T,$$

где  $X^{(k)} = [x_{k,0}, x_{k,1}, \dots, x_{k,N-1}]^T$  – подвектор размером  $N \times 1$ , описывающий элементы  $k$ -й строки массива  $X_N$ , а также вектор

$$Y^{(0)} = [Y^{(1)}, Y^{(2)}, \dots, Y^{(N-M+1)}]^T,$$

где  $Y^{(r)} = [Y_{r,0}, Y_{r,1}, \dots, Y_{r,N-M+1}]^T$  – подвектор размером  $N - M + 1$ , описывающий элементы  $r$ -й строки массива  $Y$  (далее для краткости  $N - M + 1 = Q$ );  $T$  – знак транспонирования матриц.

Процедура, описывающая модификации вычислительного процесса, может быть записана с помощью следующего векторно-матричного выражения:

$$Y_{Q^2} = U_{Q^2 N^2} X_{N^2}, \quad (2)$$

где

$$U_{Q^2 N^2} = A_{Q^2 M Q^2} A_{M Q^2 (M Q)^2} \Phi_{(M Q)^2 (M Q)^2} P_{(M Q)^2 (M N Q)} P_{(M N Q) N^2}. \quad (2a)$$

При этом матричные конструкции, которые входят в выражение (2a), определяются следующим образом:

$$P_{(M N Q) N^2} = [P_{M N N^2}^{(0)}, P_{M N N^2}^{(1)}, \dots, P_{M N N^2}^{(N-M+1)}]^T,$$

$P_{(MNQ)N^2}$  – матрица «глобального» мультиплексирования первой ступени с подматрицами  $P_{MNN^2}^{(\alpha)}$ , формируемыми в виде

$$P_{MNN^2}^{(\alpha)} = V_{MNN^2} I_N^{(\alpha N \rightarrow)}, \quad \alpha = \overline{0, N - M},$$

где  $I$  – единичная матрица, верхний индекс которой определяет число операций циклического сдвига ее столбцов в направлении, указанном стрелкой;

$$V_{MNN^2} = \|\theta_{m,n}\|, \quad \theta_{m,n} = \begin{cases} 1, & m = n, \quad m = \overline{0, M - 1}, \\ 0, & m \neq n, \quad n = \overline{0, N - 1}. \end{cases}$$

Матрица глобального мультиплексирования «придерживает» строки изображения таким образом, что в формате действия оператора свертки оказывается  $M$  строк с инкрементно меняющимися начальными номерами.

Матрица мультиплексирования второй ступени с «подматрицами локального» мультиплексирования определяется выражением

$$P_{(MQ)^2(MNQ)} = I_Q \otimes P_{(M^2Q)(MN)},$$

где

$$P_{(MQ)N} = [P_{MN}^{(0)}, P_{MN}^{(1)}, \dots, P_{MN}^{(N - M + 1)}]^T, \quad P_{MN}^{(i)} = V_{MN} I_N^{(i \rightarrow)}, \quad i = \overline{0, N - M}$$

( $V, I$  определены выше).

Матрица локального мультиплексирования обеспечивает «скольжение» оператора свертки внутри текущего формата обработки, подготовленного описанным выше глобальным мультиплексором.

$\Phi_{(MN)^2} = I_Q \otimes \Phi_{M^2Q}$  – блочно-диагональная матрица весовых коэффициентов;  $A_{(MQ^2)(MQ)^2} = I_Q \otimes (I_M \otimes (I_Q \otimes L_M))$  – матрица суммирования первой ступени (где  $L_M = [1, 1, \dots, 1]$  – вектор-строка, содержащая  $M$  единиц), обеспечивающая сборку частичных сумм свертки отдельно по строкам;  $A_{Q^2(MQ)^2} = I_Q \otimes (I_Q \otimes L_M)$  – матрица суммирования второй ступени, обеспечивающая сложение строчных сумм по столбцам и завершающая вычисления.

Алгоритм, реализующий выражение (2) со структурой и способами суммирования матричных конструкций, описанными выше, позволяет производить обработку исходного массива данных по  $Q$  независимо работающим каналам. При этом данные обрабатываются «полосами» по  $M$  строк исходного массива в каждой. На рис. 1 представлена графоструктурная модель, иллюстрирующая организацию вычислительного процесса обработки данных в соответствии с этим алгоритмом.

При альтернативном подходе базовые матричные конструкции принимают следующий вид:

$$P_{(MNQ)N^2} = [P_{(NQ)N^2}^{(0)}, P_{(NQ)N^2}^{(1)}, \dots, P_{(NQ)N^2}^{(M - 1)}]^T,$$

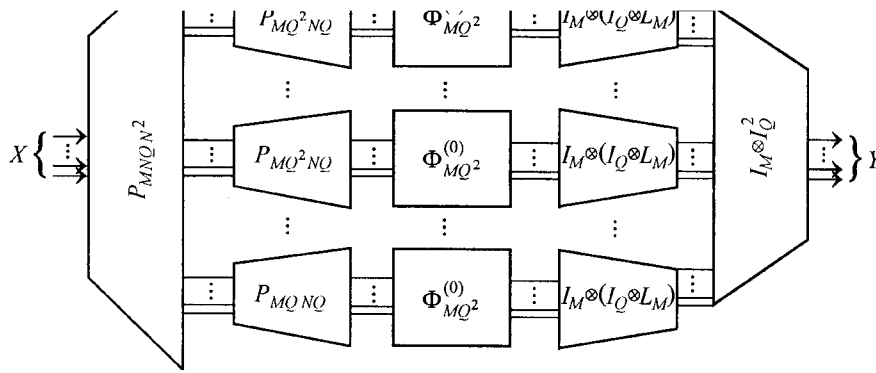


Рис. 1

где подматрицы  $P_{(NQ)N^2}^{(i)}$  формируются описанным выше способом. Матрица весовых коэффициентов  $\Phi_{(MQ)^2} = \text{diag}(\Phi_{MQ^2}^{(0)}, \Phi_{MQ^2}^{(1)}, \dots, \Phi_{MQ^2}^{(M-1)})$ , где  $\Phi_{NQ^2}^{(i)} = I_Q \otimes \Phi_M$ ,  $i = 0, M-1$ , а матрицы  $\Phi_M$  определены выше. Матрица суммирования первой ступени  $A_{(MQ^2)(MQ)^2}$  в этом случае совпадает с соответствующей матрицей для предыдущего случая, а матрица суммирования второй ступени будет формироваться таким образом:

$$A_{Q^2MQ^2} = L_M \otimes I_{Q^2}.$$

Алгоритм, реализующий выражение (2), со структурой и способами суммирования матриц (входящих в обобщенную векторно-матричную вычислительную процедуру), определенными для второго случая, позволяет производить обработку данных по  $M$  независимо работающим каналам. При этом каждый канал обрабатывает «полосу» исходного массива данных, состоящую из  $Q$  строк. На рис. 2 представлена графоструктурная модель, иллюстрирующая

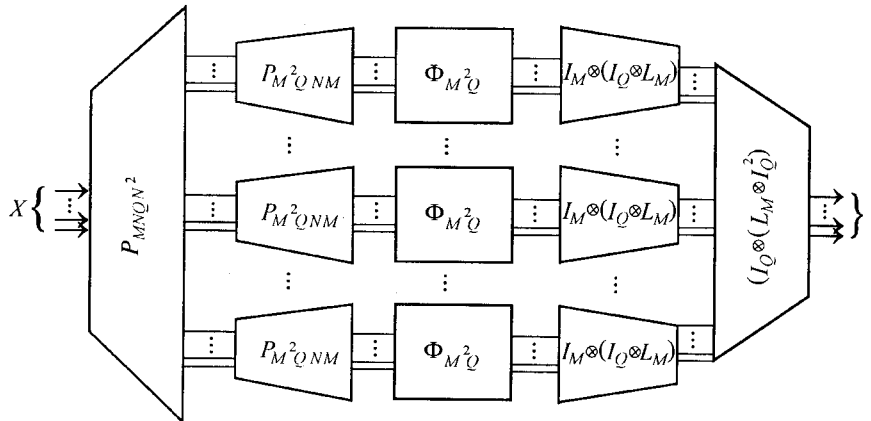


Рис. 2

рирующая организацию вычислительного процесса обработки данных в соответствии со вторым алгоритмом.

Предложенные алгоритмы могут быть использованы при организации вычисления двумерной свертки массивов в параллельных универсальных и специализированных вычислительных системах и средах. Приведенные графоструктурные модели позволяют проанализировать различные способы организации вычислений в рамках описанного подхода, оценить исходя из требований конкретной задачи количество и быстродействие процессорных модулей, возможности аппаратной реализации операции двумерной свертки.

Очевидно, что данные алгоритмы могут быть реализованы последовательно в каждом канале, последовательно-параллельно и полностью параллельно. Видоизменяя способ «накачки» данных на вход реализующего эти алгоритмы устройства, можно синтезировать целый класс приведенных вычислителей, отличающихся лучшей производительностью и более эффективных по аппаратным затратам.

#### СПИСОК ЛИТЕРАТУРЫ

1. Прэтт У. Цифровая обработка изображений: Пер с англ. /Под ред. Д. Лебедева. М.: Мир, 1982.
2. Вайрадян А. С., Пчелинев И. П., Чельшев М. М. Алгоритмы вычислений цифровых сверток // Зарубеж. радиоэлектрон. 1982. № 3. С. 3.
3. Нуссбаунмер Г. Быстрое преобразование Фурье и алгоритмы вычислений сверток: Пер. с англ. М.: Радио и связь, 1985.
4. Молдаван Д. О разработке алгоритмов для систолических матриц СБИС // ТИИЭР. 1983. № 1. С. 130.
5. Кунг С. Матричные процессоры на СБИС: Пер с англ. М.: Мир, 1991.
6. Дончеко С. Е., Матвеев Ю. Н., Очин Е. Ф. Принципы организации параллельных процессов цифровой свертки изображения // Зарубеж. радиоэлектрон. 1987. № 7. С. 84.
7. Аксенов В. П., Красинский П. Я., Спиридонов Г. В. Систолические алгоритмы и процессоры // Там же. С. 7.
8. Чиконов В. В., Кравцов С. Г., Самошин В. Систолическая обработка информации: элементная база и алгоритмы // Там же. С. 34.
9. Царев А. П. Адаптация параллельного алгоритма ускоренного вычисления ДПФ для его реализации на векторных ЭВМ // Изв. вузов. Сер. Радиоэлектроника. 1985. № 11. С. 98.

*Университет иностранных языков  
и международного бизнеса*

*Поступило в редакцию  
15 марта 1999 г.*