РЕКУРСИВНЫЙ АЛГОРИТМ ВЫЧИСЛЕНИЯ СВЕРТКИ ИЗОБРАЖЕНИЯ С НЕРАЗДЕЛИМЫМ ДВУМЕРНЫМ ПОЛИНОМИАЛЬНЫМ КИХ-ФИЛЬТРОМ

Мясников В.В.
Самарский государственный аэрокосмический университет
Институт систем обработки изображений РАН

Аннотация

В статье рассматривается рекурсивный алгоритм вычисления свертки изображения с двумерным неразделимым полиномиальным КИХ-фильтром. Существенным моментом является отказ от использовании декомпозиции 2-*D* фильтра набором разделимых звеньев. Приведены оценки вычислительной сложности предложенного рекурсивного алгоритма и дано его сравнение с известным алгоритмом, использующим указанную декомпозицию.

Введение

Вычисление свертки – одна из базовых операций в теории сигналов и цифровой обработки изображений. Для двумерного (2-*D*) случая вычисление свертки изображения $x(n_1,n_2)$ с линейным фильтром, заданным ИХ $h(n_1,n_2)$, может быть представлено в виде:

$$y(n_1,n_2)=\sum_{(m_1,m_2)\in D} h(m_1,m_2)x(n_1-m_1,n_2-m_2). \qquad (1)$$

Здесь $y(n_1,n_2)$ - выходного изображение, D – область ненулевых значений ИХ фильтра. В дальнейшем предполагается, что область D финитна, то есть рассматриваются фильтры с конечными импульсными характеристиками – КИХ-фильтры. Для определенности положим

$$D=\left[-M_1^-,M_1^+\right]\times\left[-M_2^-,M_2^+\right].$$

Одним из существующих подходов к быстрому вычислению 1-*D* и 2-*D* сверток, помимо быстрых ортогональных преобразований, является их рекурсивная реализация. Существующая теория и результаты в этой области существенным образом ориентированы на 1-*D* случай [1-3]. В частности, известный метод перехода к двумерным рекурсивным КИХ-фильтрам использует декомпозицию ИХ фильтра с помощью разделимых функций $p_k(m_1)p_l(m_2)$, каждая из которых реализуется рекурсивно [1,3]:

$$h(m_1,m_2)=\sum_{k=0}^{K}\sum_{l=0}^{L}\beta_{kl}\,p_k(m_1)p_l(m_2) \qquad (2)$$

Как показано в работах [1,2] удобными для рекурсивной реализации свертки являются полиномиальные функции, в частности, базис МВС. Этот базис позволяет реализовать свертку рекурсивно и с минимальным (для полиномиального базиса общего вида) числом арифметических операций.

В настоящей работе предлагается метод построения 2-*D* рекурсивного алгоритма вычисления свертки изображения с полиномиальной ИХ. В методе не требуется разложение ИХ на разделимые звенья. Показано, что рекурсивный алгоритм, получаемый при использовании указанного метода, позволяет снизить теоретическую сложность вычисления свертки (1), оцениваемую числом арифметических операций, на 20% по сравнению с известным (использует разделимые звенья).

Статья организована следующим образом. В первом разделе описан известный алгоритм рекурсивного вычисления свертки с 2-*D* полиномиальной ИХ, использующий декомпозицию ИХ набором разделимых звеньев. Также в этом разделе приведены известные результаты по 1-*D* рекурсивной фильтрации на основе МВС - базисов. Во втором разделе дано построение предлагаемого 2-*D* рекурсивного алгоритма. В третьем разделе приведен сам алгоритм, представлены оценки его вычислительной сложности и дано сравнение с известным алгоритмом по сложности реализации свертки.

1. Рекурсивное вычисление 2-D свертки на основе декомпозиции КИХ фильтра с помощью разделимых звеньев

Пусть ИХ фильтра представима в виде разложения по 2-*D* разделимым функциям (2). Тогда выражение (1) расчета свертки можно переписать следующим образом:

$$y(n_1,n_2)=\sum_{k=0}^{K}\sum_{l=0}^{L}\beta_{kl}\mu_{kl}(n_1,n_2), \qquad (3)$$

где

$$\mu_{kl}(n_1,n_2)= = \sum_{m_1=-M_1^-}^{M_1^+} p_k(m_1)\sum_{m_2=-M_2^-}^{M_2^+} p_l(m_2)x(n_1-m_1,n_2-m_2) \qquad (4)$$

По аналогии с работами [1-3] величины $\mu_{kl}(n_1,n_2)$ назовем *обобщенными моментами*. Очевидно, что в соответствии с выражением (4) вычисление 2-*D* свертки можно производить в построчно-постолбцовой манере, а именно:

- производится вычисление 1-*D* сверток вдоль строк (формируем полуспектр одномерных моментов по горизонтали):

$$\tilde{\mu}_l(n_1-m_1,n_2)=\sum_{m_2=-M_2^-}^{M_2^+} p_l(m_2)x(n_1-m_1,n_2-m_2)\,; \qquad (5)$$

- вдоль столбцов полуспектра вычисляются обобщенные моменты:

$$\mu_{kl}(n_1,n_2)=\sum_{m_1=-M_1^-}^{M_1^+} p_k(m_1)\tilde{\mu}_l(n_1-m_1,n_2)\,; \qquad (6)$$

- рассчитывается скалярное произведение значений обобщенных моментов (6) на коэффициенты разложения ИХ (3).

Сложность вычисления 2-D свертки с использованием такого построчно-постолбцового алгоритма складывается из сложностей каждой из трех указанных групп операций.

При использовании рекурсивного подхода вычисление 1-D сверток (5) и (6) производится рекурсивно. При использовании полиномиального базиса функции $p_k(m_1)$ и $p_l(m_2)$ являются полиномами:

$$p_k(m) = \sum_{i=0}^{k} a_{ki} m^i, \quad k = \overline{0,K}. \tag{7}$$

В работах [1,3] показано, что коэффициенты $\{a_{ki}\}_{i=0}^{k}$ $(a_{kk} \neq 0)$ могут быть подобраны таким образом, чтобы произвольная 1-D свертка с набором функций (7) выполнялась с минимальной вычислительной сложностью (МВС) для полиномиального базиса общего вида. Рекурсивный алгоритм формирования 1-D моментов и его вычислительная сложность в сложениях $U_+(K)$ и умножениях $U_*(K)$ в пересчете на один отсчет выходного изображения составляют [1,3]:

$$\begin{cases} \mu_0(n) = \mu_0(n-1) + x(n+M^-) - x(n-M^+-1), \\ \mu_k(n) = \mu_k(n-1) + \mu_{k-1}(n-1) + \\ + p_k^{mcc}(-M^-)x(n+M^-), \quad 1 \le k \le K. \end{cases} \tag{8}$$

$$U_+(K) = 2(K+1) \quad U_*(K) = K.$$

При использовании базиса МВС $\{p_k^{mcc}\}_{k=0}^{K}$ вычислительная сложность расчета двумерной свертки в пересчете на один отсчет выходного изображения равна:

$$\begin{aligned} &U_+(K,L) = 3(K+1)(L+5/3), \\ &U_*(K,L) = 2(K+1)(L+1) - 1. \end{aligned} \tag{9}$$

Суммарная сложность, таким образом, пропорциональна величине:

$$U_+(K,L) + U_*(K,L) \cong 5KL. \tag{10}$$

2. Построение 2-D рекурсивного алгоритма вычисления свертки

Для построения 2-D рекурсивного алгоритма вычисления свертки воспользуемся тем фактом, что любая горизонтальная строка полиномиальной импульсной характеристики степени (K,L) также является полиномиальной функцией степени не выше чем L. Действительно, если импульсная характеристика фильтра представима в виде степенного ряда

$$h(m_1,m_2) = \sum_{k=0}^{K} \sum_{l=0}^{L} \alpha_{kl} m_1^k m_2^l, \tag{11}$$

тогда m_1-ая строка ИХ имеет вид:

$$h_{m_1}(m_2) = \sum_{l=0}^{L} m_2^l \left(\sum_{k=0}^{K} \alpha_{kl} m_1^k \right) = \sum_{l=0}^{L} \alpha_l^{m1} m_2^l.$$

Введем обозначение:

$$p_L^{m_1}(m_2) \equiv h_{m_1}(m_2) = \sum_{l=0}^{L} \alpha_l^{m1} m_2^l. \tag{12}$$

Тогда свертка (1) представима:

$$y(n_1,n_2) = \sum_{(m_1,m_2)\in D} p_L^{m_1}(m_2) x(n_1-m_1, n_2-m_2) \tag{13}$$

По аналогии с (4) обозначим:

$$\mu_l^{m_1}(n_1,n_2) = \sum_{m_2=-M_2^-}^{M_2^+} p_l^{m_1}(m_2) x(n_1-m_1, n_2-m_2) \tag{14}$$

$$\mu_l(n_1,n_2) = \sum_{m_1=-M_1^-}^{M_1^+} \mu_l^{m_1}(n_1,n_2) \tag{15}$$

Далее, для каждой полиномиальной функции $p_L^{m_1}(m_2)$ (вид функции зависит от m_1) построим последовательность полиномиальных функций $\{p_l^{m_1}\}_{l=0}^{L}$ следующим образом $l = \overline{L,0}$:

$$p_{l-1}^{m_1}(m_2-1) = p_l^{m_1}(m_2) - p_l^{m_1}(m_2-1).$$

Как показано в работах [1,3], для последовательности таких 1-D полиномов существует эффективный алгоритм рекурсивного вычисления свертки, который в 1-D случае выглядит следующим образом (индексы m_1 для удобства опущены):

$$\begin{cases} \mu_l(n) = \mu_l(n-1) + \mu_{l-1}(n-1) + \\ \quad + p_l(-M^-)x(n+M^-) - \\ \quad - p_l(M^++1)x(n-M^+-1), \; l = \overline{1,L}. \\ \mu_0(n) = \mu_0(n-1) + \\ \quad + p_0(0)\cdot\left(x(n+M^-) - x(n-M^+-1)\right). \end{cases} \tag{16}$$

Здесь

$$\mu_l(n) = \sum_{m=-M^-}^{M^+} p_l(m) x(n-m). \tag{17}$$

Используя этот результат, можно переписать выражение (14) следующим образом:

$$\begin{aligned} &\mu_l^{m_1}(n_1,n_2) = \mu_l^{m_1}(n_1,n_2-1) + \mu_{l-1}^{m_1}(n_1,n_2-1) + \\ &+ p_l^{m_1}(-M_2^-)x(n_1-m_1, n_2+M_2^-) - \\ &- p_l^{m_1}(M_2^++1)x(n_1-m_1, n_2-M_2^+-1), \; l = \overline{1,L} \end{aligned}$$

Подставив это рекуррентное соотношение в равенство (15), можно получить следующее выражение:

$$\mu_l(n_1,n_2) = \sum_{m_1=-M_1^-}^{M_1^+} \mu_l^{m_1}(n_1,n_2) =$$

$$= \sum_{m_1=-M_1^-}^{M_1^+} \mu_l^{m_1}(n_1,n_2-1) + \sum_{m_1=-M_1^-}^{M_1^+} \mu_{l-1}^{m_1}(n_1,n_2-1) +$$

$$+\sum_{m_1=-M_1^-}^{M_1^+} p_l^{m_1}(-M_2^-)x(n_1-m_1,n_2+M_2^-)-$$

$$-\sum_{m_1=-M_1^-}^{M_1^+} p_l^{m_1}(M_2^+ +1)x(n_1-m_1,n_2-M_2^+ -1).$$

Упрощая первые два слагаемых по формуле (15), получим:

$$\mu_l(n_1,n_2)=\mu_l(n_1,n_2-1)+\mu_{l-1}(n_1,n_2-1)+$$

$$+\sum_{m_1=-M_1^-}^{M_1^+} p_l^{m_1}(-M_2^-)x(n_1-m_1,n_2+M_2^-)-$$

$$-\sum_{m_1=-M_1^-}^{M_1^+} p_l^{m_1}(M_2^+ +1)x(n_1-m_1,n_2-M_2^+ -1). \quad (18)$$

Очевидно, вычислительная сложность полученного 2-*D* рекурсивного алгоритма вычисления свертки с полиномиальным КИХ-фильтром зависит от вертикального размера ИХ. Для снижения сложности операций скалярного произведения в (18) их можно реализовать также рекурсивным образом.

Действительно, любая величина $p_l^{m_1}(M)$ может быть представлена как полиномиальная функция степени *K* (или ниже) от m_1. Например, для случая $l=L$:

$$p_L^{m_1}(M)=\sum_{l=0}^{L}M^l\left(\sum_{k=0}^{K}\alpha_{kl}m_1^k\right)=$$

$$=\sum_{k=0}^{K}m_1^k\left(\sum_{l=0}^{L}\alpha_{kl}M^l\right)=\sum_{k=0}^{K}m_1^k\alpha_k^{(L)M}=p_K^{(L)M}(m_1)$$

В общем случае выполняется:

$$p_l^{m_1}(M)=p_K^{(l)M}(m_1),$$

$$l=\overline{0,L}, \qquad M=M_2^+ +1 \text{ или } M=-M_2^- .$$

Каждую из этих $2(L+1)$ одномерных полиномиальных функций можно представить в виде разложения по 1-*D* полиномиальному МВС-базису (8) $\left\{p_k^{mcc}(m_1)\right\}_{k=0}^{K}$:

$$p_K^{(l)M}(m_1)=\sum_{k=0}^{K}\beta_k^{(l)M}p_k^{mcc}(m_1), \quad l=\overline{0,L}, \qquad (19)$$

$$l=\overline{0,L}, \qquad M=M_2^+ +1 \text{ или } M=-M_2^- ..$$

Свертка с функциями МВС-базиса, как указывалось ранее, реализуется рекурсивно (по столбцам) в соответствии с выражением (8). Таким образом, предлагаемый алгоритм рекурсивного вычисления 2-*D* свертки изображения с неразделимым КИХ-фильтром представим следующим образом.

3. Алгоритм

Шаг 1. Рекурсивное вычисление 1-*D* обобщенных моментов за счет свертки столбцов изображения с полиномами МВС по выражению (8).

Шаг 2. Вычисление $2(L+1)$ скалярных произведений значений 1-*D* обобщенных моментов на коэффициенты (19).

Шаг 3. Подстановка полученных $2(L+1)$ значений в общее рекуррентной 2-*D* соотношение (18) и получение значения величины $\mu_L(n_1,n_2)\equiv y(n_1,n_2)$ на основании предшествующих значений 2-*D* свертки. Этот шаг не требует ни одной операции умножения.

Вычислительная сложность предложенного алгоритма в пересчете на один отсчет выходного изображения составляет:

$$U_+(K,L)=2(K+1)+2(L+1)K+(3L+2),$$
$$U_*(K,L)=K+2(L+1)(K+1).$$

или, окончательно:

$$U_+(K,L)=2(K+1.5)(L+2)-2,$$
$$U_*(K,L)=2(K+1)(L+1.5)-1.$$

Итоговая сложность пропорциональна:

$$U_+(K,L)+U_*(K,L)\cong 4KL,$$

что на 25% ниже, чем у традиционного построчно-постолбцового алгоритма.

В заключении следует отметить, что предложенный подход может быть естественным образом обобщен на ситуацию представления ИХ фильтра произвольными рекуррентными последовательностями.

Благодарности

Работа выполнена при поддержке Министерства образования РФ, Администрации Самарской области и Американского фонда гражданских исследований и развития (CRDF Project SA-014-02) в рамках российско-американской программы "Фундаментальные исследования и высшее образование" (BRHE).

Литература

1. N.I.Glumov, V.V.Myasnikov, V.V.Sergeyev Parallel-Recursive Local Image Processing and Polynomial Bases // Proceedings of the Third IEEE International Conference on Electronics, Circuits, and Systems ICECS'96, 1996. Vol.2. Rodos, Greece. P.696-699.
2. B.C.Li and J.Shen Two-dimensional local moment, surface fitting and their fast computation // Pattern Recognition, 1994. Vol.27. No.6. P.785-790.
3. V.V.Myasnikov Construction of Integer-Value Polynomials for Recursive Calculation of the Convolution with FIR-Filter // Proceedings of the 7-th International Conference on Pattern Recognition and Image Analysis - PRIA'2004, 2004. P.331-334.

A recursive algorithm for computing the convolution of an image with a two-dimensional inseparable polynomial FIR filter

V.V. Myasnikov [1]
[1] *Samara State Aerospace University*
[2] *Image Processing Systems Institute of RAS*

Abstract

The article discusses a recursive algorithm for computing the image convolution with a two-dimensional nonseparable polynomial FIR filter. The essential point is the refusal to use the 2-D filter decomposition by a set of separable links. The proposed recursive algorithm is evaluated from the point of view of its computational complexity and is compared to the well-known algorithm using the above decomposition.

Keywords: FIR filter, two-dimensional polynomial, 2-D filter.

Citation: Myasnikov VV. A recursive algorithm for computing the convolution of an image with a two-dimensional inseparable polynomial FIR filter. Computer Optics 2004; 26: 81-83.

References

[1] Glumov NI, Myasnikov VV, Sergeyev VV. Parallel recursive local image processing and polynomial bases. Proc Third IEEE Int Conf on Electronics, Circuits, and Systems (ICECS'96) 1996; 2: 696-699.
[2] Li BC, Shen J. Two-dimensional local moment, surface fitting and their fast computation. Patt Recogn 1994; 27(6): 785-790.
[3] Myasnikov VV. Construction of integer-value polynomials for recursive calculation of the convolution with FIR-filter. Proc 7-th Int Conf on Pattern Recognition and Image Analysis (PRIA'2004) 2004: 331-334.