

Лабораторная работа 4

Алгоритмы свертки в OpenMP

Цель работы. Изучение различных алгоритмов выполнения линейной фильтрации на примере среднеарифметического фильтра и фильтра Гаусса с его различными аппроксимациями. Получение навыков разработки параллельных алгоритмов, реализующих линейные фильтры, принимая во внимания подходы к оптимизации обработки.

ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. Ознакомиться с теоретическими сведениями.
2. Ознакомиться с алгоритмами линейной фильтрации изображения: линейный усредняющий фильтр, фильтр Гаусса.
3. Выполнить задания 1–2.
4. Подготовить отчет по лабораторной работе.
5. Защитить лабораторную работу перед преподавателем.

ЗАДАНИЯ

Задание 4.1. Разработайте консольное приложение, реализующее выполнение линейной фильтрации изображения. Необходимо разработать и реализовать последовательные и параллельные варианты среднеарифметического фильтра и фильтра Гаусса. Для среднеарифметического фильтра также необходимо реализовать быстрый вариант.

В приложение должен загружаться файл `input.bmp` и формироваться выходные файлы `output_alg_var.bmp`, где `alg` означает имя используемого алгоритма обработки (или его номер), а `var` – варианта реализации.

Задание 4.2. Проведите экспериментальное исследование по обработке данных с разными реализациями алгоритмов фильтрации (не менее **20 запусков** для каждой реализации). При применении алгоритмов получить показатели для разных размеров окрестности $K_{size} \times K_{size}$ для четырех изображений с разрешением от 1280×720 до 3840×2160 ¹. В рамках лабораторной работы $K_{size} = (RH \times 2 + 1) = (RW \times 2 + 1)$. Результаты исследования оформить в виде табл. 4.1.

¹ В зависимости от оборудования (на котором будут выполняться эксперименты) и согласованию с преподавателем возможно изменения наборов данных в сторону уменьшения.

Таблица 4.1

Функция (реализация)	Поток	K _{size}	НД 1		НД 2		НД 3		НД 4	
			Время	S _p (n)	Время	S _p (n)	Время	S _p (n)	Время	S _p (n)
Алгоритм 1		7								
Вариант реализации 1		11								
(посл.)		15								
Алгоритм 1 Вариант реализации 1 (парал.)	2	7								
		11								
		15								
	3	7								
		11								
		15								
	4	...								
Алгоритм 1		7								
Вариант реализации 2		11								
(посл.)		15								
Иные варианты реализации.										
Алгоритм 2		7								
Вариант реализации 1		11								
(посл.)		15								
Алгоритм 2 Вариант реализации 1 (парал.)	2	7								
		11								
		15								
	3	...								
	4	...								
Иные варианты реализации.										

ОТЧЕТ

Отчет сдается преподавателю в электронном виде и должен содержать:

- титульный лист;
- цель работы;
- постановку задачи;
- исходный код программы задания 4.1;
- примеры исходных и обработанных данных (картинки);
- конфигурация компьютера и параметры операционной системы, на которой производится выполнение задания 4.2;
- результаты выполнения задания 4.2 (экспериментального исследования) в табличной форме;
- выводы по результатам экспериментального исследования;
- выводы по лабораторной работе в целом.

КОНТРОЛЬНЫЕ ВОПРОСЫ И ЗАДАНИЯ

1. Дайте определение понятию «линейный фильтр», приведите последовательный алгоритм его реализации.
2. Перечислите и охарактеризуйте известные вам виды распараллеливания обработки изображений линейными фильтрами.
3. Приведите рекурсивный алгоритм линейной фильтрации (вычисление среднеарифметического значения).
4. Охарактеризуйте способы распараллеливания рекурсивного алгоритма линейной фильтрации.
5. Приведите последовательный алгоритм фильтра Гаусса.
6. Покажите способы повышения скорости обработки при реализации фильтра Гаусса.
7. Охарактеризуйте особенности параллельной реализации фильтра Гаусса.
8. Как используются секции для распараллеливания алгоритма фильтра Гаусса?
9. В чем заключаются особенности балансировки нагрузки при реализации обработки изображений линейными фильтрами.

ИНФОРМАЦИЯ ПО ЛИНЕЙНЫМ ФИЛЬТРАМ

Как следует из названия, отклик (результат применения фильтра) линейного фильтра линейным образом зависит от обрабатываемого изображения. Значение нового пиксела вычисляется следующим образом:

$$C_{\text{new}}[y, x] = \sum_{dy=-RH}^{RH} \sum_{dx=-RW}^{RW} K[dy + RH, dx + RW] \times C_{\text{old}}[y + dy, x + dx],$$

$$RH = (M - 1) / 2; \quad RW = (N - 1) / 2,$$

где C_{old} – матрица значений пикселей изображения до обработки; C_{new} – матрица значений пикселей изображения после обработки; K – матрица коэффициентов определяющая эффект, который накладывает фильтр; RH , RW – размеры рангов скользящего окна; M , N – размеры скользящего окна по вертикали и горизонтали.

Константы M и N задают размер фильтра (обладает двумерной апертурой, в случае одномерной обработки M или N считается единичным размером). Коэффициент фильтрации в случаях одномерной фильтрации представляет собой матрицу «столбец» или «строка». Условная схема, демонстрирующая принцип выполнения линейной фильтрации изображения, представлена на рис. 4.1.

Среднеарифметический сглаживающий фильтр основывается на следующем принципе: находится среднее арифметическое значение всех элементов скользящего окна изображения, для каждого из каналов цветовой

модели (например, RGB). В случае для размерности скользящего окна 3×3 матрица коэффициентов (ядро) будет иметь следующий вид:

$$K = \begin{vmatrix} 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \end{vmatrix}.$$

Фильтр Гаусса представляет собой разновидность линейного фильтра, где элементы матрицы коэффициентов рассчитываются, исходя из функции распределения. Функция распределения коэффициентов фильтра Гаусса в одномерном пространстве принимает вид

$$G(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}},$$

а в двумерном пространстве рассчитывается по формуле

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}.$$

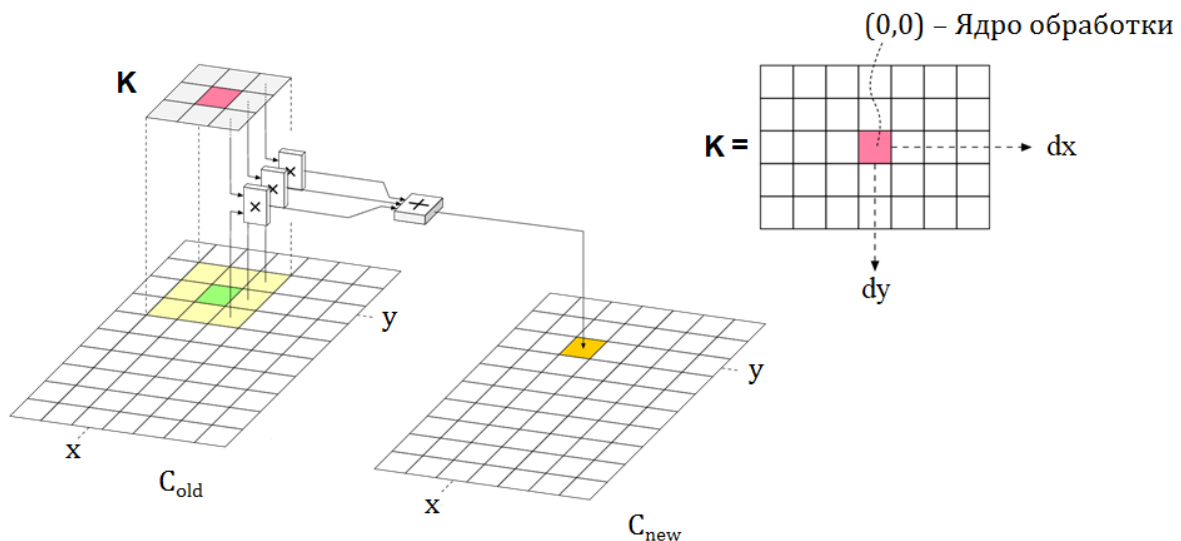


Рис. 4.1. Принцип линейных фильтров (Свертка)

В случае квадратных матриц возможно использование свойств сепарабельности получения результата. Так ядро фильтра (матрица коэффициентов) называется сепарабельным, если его можно представить в виде $K_{yx} = A_x B_y$, где A и B – некоторые одномерные массивы.

$$K = \begin{vmatrix} 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \end{vmatrix} = \begin{vmatrix} 1/3 \\ 1/3 \\ 1/3 \end{vmatrix} \begin{vmatrix} 1/3 & 1/3 & 1/3 \end{vmatrix}$$

Вычисление элемента для фильтра Гаусса можно представить:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} = \left(\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{y^2}{2\sigma^2}} \right) \left(\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \right)$$

Однако, в таком случае придется применять две одномерные обработки к изображению. Поскольку интерпретировать изображение можно по-разному, то можно сначала выполнить обработку по горизонтали, а потом по вертикали. Для большего ускорения обработки целесообразно выполнять две обработки по горизонтали, при этом результат будет записываться в транспонированной форме: $\text{Filtr}(\text{Im}_{yx}) \rightarrow \text{Im}_{xy}$, $\text{Filtr}(\text{Im}_{xy}) \rightarrow \text{Im}_{yx}$. При этом понадобится дополнительный массив для хранения промежуточного результата (транспонированное изображение).

Пример возможной реализации обобщенного линейного фильтра показан с использованием псевдокода².

Используемые обозначения:

IMAGE[][]	– массив элементов входного изображения в формате адресации [Y][X]
IMAGE2[][]	– массив элементов выходного изображения в формате адресации [Y][X]
Height, Width	– размеры изображения по высоте и ширине
RH, RW	– размеры рангов скользящего окна $M \times N = (RH \times 2 + 1) \times (RW \times 2 + 1)$
MATR_coef[][]	– массив коэффициентов фильтра [0..M][0..N]

Линейный фильтр

Начало

Цикл 1 по Y изображению (Y от 0 до Height)

Цикл 2 по X изображению (X от 0 до Width)

Начало 2

LinF_Value = 0;

Цикл A1 (DY от -RH до RH включительно)

Начало A1

KY = Y + DY

Если KY < 0 **то** KY = 0

Если KY > Height - 1 **то** KY = Height - 1

Цикл A2 (DX от -RW до RW включительно)

Начало A2

KX = X + DX

Если KX < 0 **то** KX = 0

Если KX > Width - 1 **то** KX = Width - 1

LinF_Value += IMAGE[KY][KX] ×

² Для обработки RGB изображения понадобятся соответствующие переменные для каждого канала, например: LinF_Value_R, LinF_Value_G, LinF_Value_B.

MATR_coef[DY + RH][DX + RW]

Конец A2

Конец A1

Если LinF_Value < 0 **то** LinF_Value = 0

Если LinF_Value > 255 **то** LinF_Value = 255

Запись в выходное изображение IMAGE2(Y,X) значения отклика LinF_Value

Конец 2

Конец

Для реализации фильтра Гаусса необходимо заполнить матрицу коэффициентов для этого можно использовать следующую реализацию:

SUM = 0

Цикл A1 (Y от -RH до RH **включительно**)

Цикл A2 (X от -RW до RW **включительно**)

Начало A2

YK = Y + RH;

XK = X + RW;

CF = (1 / (2 × PI × σ^2)) × exp(-1 × (X² + Y²) / (2 × σ^2);

MATR_coef[YK][XK] = CF

SUM += MATR_coef[YK][XK]

Конец A2

Цикл B1 (Y от -RH до RH **включительно**)

Цикл B2 (X от -RW до RW **включительно**)

Начало B2

YK = Y + RH;

XK = X + RW;

MATR_coef[YK][XK] = MATR_coef[YK][XK] × (1 / SUM)

Конец B2

В случае реализации линейного среднеарифметического фильтра можно не использовать матрицу коэффициентов, а осуществлять вычисления следующим образом в цикле A2 :

Summ_Value += IMAGE(KY, KX)

По завершения цикла A1, необходимо рассчитать среднеарифметическое значение:

LinF_Value = Summ_Value / Size

где Size – площадь ядра (скользящего окна) = M×N=(RH×2+1)×(RW×2+1)

Работу среднеарифметического фильтра можно дополнительно ускорить путем модификации расчета суммы в цикле 2 линейного фильтра (используя рекурсивные математические вычисления)³.

³ На сервере дистанционного образования СибГУ им. М.Ф. Решетнева доступен презентационный слайд поясняющий механизм вычисления.

Так в начале каждой строки (координата $X = 0$) будет выполняться расчет суммы общим образом:

```
Если  $X = 0$  то  
    Summ_Value = 0;  
    Начало А  
    Цикл А1 (DY от -RH до RH включительно)  
        Начало А1  
        KY = Y + DY  
        Если KY < 0 то KY=0  
        Если KY > Height - 1 то KY = Height -1  
        Цикл А2 (DX от -RW до RW включительно)  
            Начало А2  
            KX = X + DX  
            Если KX < 0 то KX = 0  
            Если KX > Width - 1 то KX = Width - 1  
            Summ_Value += IMAGE[KY][KX]  
            Конец А2  
        Конец А1  
    LinF_Value = Summ_Value / Size  
    Запись в выходное изображение IMAGE2(Y,X) значения  
отклика LinF_Value  
    Конец А
```

При обработке оставшихся пикселей строки из суммы будут вычитаться все элементы, которые оказались за пределами ядра (слева), и прибавляться все элементы которые попали в ядро (справа).

```
Если  $X > 0$  то  
    Начало А  
    KX1 = X - RW - 1  
    Если KX1 < 0 то KX1 = 0  
    KX2 = X + RW  
    Если KX2 > Width - 1 то KX2 = Width - 1  
    Цикл А1 (DY от -RH до RH включительно)  
        Начало А1  
        KY = Y + DY  
        Если KY < 0 то KY=0  
        Если KY > Height - 1 то KY = Height -1  
        Summ_Value -= IMAGE[KY][KX1]  
        Summ_Value += IMAGE[KY][KX2]  
        Конец А2  
    Конец А1  
    LinF_Value = Summ_Value / Size  
    Запись в выходное изображение IMAGE2(Y,X) значения  
отклика LinF_Value  
    Конец А
```