

нии. Характеризует ресурсоемкость процессов кодирования и декодирования. Для нас наиболее важной является симметричность по времени: отношение времени кодирования ко времени декодирования. Иногда нам потребуется симметричность по памяти.

4. **Есть ли потери качества?** И если есть, то за счет чего изменяется степень сжатия? Дело в том, что у большинства алгоритмов сжатия с потерей информации существует возможность изменения степени сжатия.
5. **Характерные особенности алгоритма** и изображений, к которым его применяют. Здесь могут указываться наиболее важные для алгоритма свойства, которые могут стать определяющими при его выборе.

Используя данные критерии, приступим к рассмотрению алгоритмов архивации изображений.

Прежде, чем непосредственно начать разговор об алгоритмах, хотелось бы сделать оговорку. Один и тот же алгоритм часто можно реализовать разными способами. Многие известные алгоритмы, такие как RLE, LZW или JPEG, имеют десятки различающихся реализаций. Кроме того, у алгоритмов бывает несколько явных параметров, варьируя которые, можно изменять характеристики процессов архивации и разархивации. (См. примеры в разделе о форматах). При конкретной реализации эти параметры фиксируются, исходя из наиболее вероятных характеристик входных изображений, требований на экономию памяти, требований на время архивации и т.д. Поэтому у алгоритмов одного семейства лучшая и худшая степени сжатия могут отличаться, но качественно картина не изменится.

Методы обхода плоскости

Задача обхода плоскости возникает при обработке двумерных данных. Цель: создание одномерного массива D из двумерного S . Причем если предполагается последующее сжатие D , то желательно создавать его так, чтобы «разрывов» было как

можно меньше: каждый следующий элемент D_i , заносимый в D на i -ом шаге, является соседним (в плоскости) для предыдущего, занесенного в D на $(i-1)$ -ом шаге, D_{i-1} .

ЗМЕЙКА (ЗИГЗАГ-СКАНИРОВАНИЕ)

Обход массива S начинается с одного угла плоскости, заканчивается в противоположном по диагонали. Например, из левого верхнего в правый нижний:

1	2	6	7	15	16
3	5	8	14	17	24
4	9	13	18	23	25
10	12	19	22	26	29
11	20	21	27	28	30

На иллюстрации показан порядок выборки элементов из плоскости (с последующим занесением в одномерный массив). Значение из ячейки массива S , помеченной на рисунке номером i , заносится в $D[i]$.

Змейка выгодна в случаях, когда в одном из углов «особенность» — например, сосредоточены самые крупные коэффициенты. Применяется в алгоритме JPEG для обхода квадрантов (размером 8×8 точек).

ОБХОД СТРОКАМИ

Самый тривиальный метод. Именно он используется в самых распространенных графических форматах (BMP, TGA, RAS...) для хранения элементов изображений.

1	2	3	4	5	6
7	8	9	10	11	12
13	14	15	16	17	18
19	20	21	22	23	24
25	26	27	28	29	30

В варианте строк с разворотами для каждой второй строки делаем выборку в обратном направлении:

1	2	3	4	5	6
12	11	10	9	8	7
13	14	15	16	17	18
24	23	22	21	20	19
25	26	27	28	29	30

Точек «разрыва» нет, в отличие от варианта без разворотов. Совершенно аналогично можно делать обход столбцами.

Упражнение: Нарисуйте пример обхода плоскости столбцами с разворотами.

Обход полосами

Чаще всего сжатие лучше, если каждая область двумерного массива S не рассредоточена (равномерно «размазана») по всему одномерному D , а сконцентрирована в D компактно. В случае обхода строками понятие «области» отсутствует: каждый элемент считается «областью». Пытаясь обходить плоскость квадратами размером $N \times N$, приходим к идее обхода горизонтальными «полосами» шириной N :

1	4	7	10	13
2	5	8	11	14
3	6	9	12	15
16	19	22	25	28
17	20	23	26	29
18	21	24	27	30

В данном примере ширина полосы $N=3$. Если $N=1$, получаем обход строками.

ПОЛОСАМИ С РАЗВОРОТАМИ

То же самое, но с разворотами и столбцов внутри полос, и направлений самих полос:

1	6	7	12	13
2	5	8	11	14
3	4	9	10	15
28	27	22	21	16
29	26	23	20	17
30	25	24	19	18

Разрывов опять нет, но теперь еще и каждая точка принадлежит **области**, записанной в D **компактно**, без разрывов: ее элементы расположены внутри одного интервала ($D[i]$, $D[i+1]$, ..., $D[i+j]$), и элементов из других областей внутри этого интервала нет. Примеры таких областей — каждый из четырех углов размером 3×3 элемента.

Упражнение: Нарисуйте схему обхода квадрата 7×7 полосами шириной 3 с разворотами. Какой вариант лучше: $3+3+1$ или $3+2+2$? Какие области заходятся в D компактно?

ОБХОД РЕШЕТКАМИ

Для первой порции берем элементы из каждого N -го столбца каждой M -ой строки. Для второй — то же, но со сдвигом на один столбец. Так же и для следующих, а затем — со сдвигом на одну,

две, $(M-1)$ строки. Например, если $M=N=2$, то имеем четыре порции:

1	13	2	14	3	15	4	16
25	37	26	38	27	39	28	40
5	17	6	18	7	19	8	20
29	41	30	42	31	43	32	44
9	21	10	22	11	23	12	24
33	45	34	46	35	47	36	48

То есть плоскость разбивается на прямоугольники размером $M \times N$, задается обход плоскости прямоугольниками, а также обход внутри самих прямоугольников, и далее делается «одновременный» обход по каждому из них: сначала выбираются их первые элементы, затем вторые, третьи, и так далее, до последнего.

ОБХОД РЕШЕТКАМИ С УЧЕТОМ ЗНАЧЕНИЙ ЭЛЕМЕНТОВ

После того как обработаны первые элементы прямоугольников (в нашем примере – квадратов 2×2), если предположить, что они являются атрибутами своих областей, выгодно (для улучшения сжатия) группировать области с одинаковыми атрибутами, то есть с одинаковыми значениями этих первых элементов.

Допустим, атрибуты распределены так:

R		G		G		G	
L		R		G		G	
L		L		R		R	

Тогда имеет смысл дальше действовать так:

R	13	G	25	G	28	G	31
15	14	27	26	30	29	33	32
L	40	R	16	G	34	G	37
42	41	18	17	36	35	39	38
L	43	L	46	R	19	R	22
45	44	48	47	21	20	24	23

То есть сначала обходим квадраты с атрибутом «R», затем с атрибутом «G», и наконец с «L».

КОНТУРНЫЙ ОБХОД

Часть элементов принадлежит одной группе, часть – другой, причем контур задан:

1	1	1	1	1	1	1	1
1	1	2	2	1	1	1	1
1	2	2	2	2	1	1	1
1	2	2	2	2	2	1	1
1	1	1	2	1	1	1	1
1	1	1	1	1	1	1	1

36 элементов – из группы «1», а 12 – из группы «2».

Очевидно, что имеет смысл отдельно оформить элементы группы «1»:

1	29	28	27	26	22	21	31
2	30			25	23	20	32
3					24	19	33
4						18	34
5	8	9		13	14	17	35
6	7	10	11	12	15	16	36

и затем точно так же остальные элементы, принадлежащие группе «2».

КОНТУРНЫЙ ОБХОД С НЕИЗВЕСТНЫМИ КОНТУРАМИ

Рассмотрим предыдущий пример, то есть такое же распределение элементов групп по плоскости, но изначально, при начале обхода плоскости, это распределение неизвестно. Будем действовать таким методом:

1	44	41	40	35	34	29	28
2	43	42	39	36	33	30	27
3	45	46	38	37	32	31	26
4	9	10	47	48	19	20	25
5	8	11	14	15	18	21	24
6	7	12	13	16	17	22	23

Обходим плоскость «столбцами с разворотами», и, обнаруживая элемент другой группы (в элементах 9, 10, 14...), также делаем разворот на 180 градусов. Затем (шаги 45...48) обходим оставшуюся часть плоскости, содержащую (предположительно) элементы другой группы.

В итоге имеем:

- среди первых 36-и элементов 4 из группы «2», а 32 из группы «1».
- из последних 12-и элементов 8 из группы «2», а 4 из группы «1».

В первой части $4/36=1/9$ «исключений», во второй $4/12=1/3$.

А если бы делали просто обход «столбцами с разворотами»:

1	12	13	24	25	36	37	48
2	11	14	23	26	35	38	47
3	10	15	22	27	34	39	46
4	9	16	21	28	33	40	45
5	8	17	20	29	32	41	44
6	7	18	19	30	31	42	43

В итоге:

- среди первых 33-х элементов 12 из группы «2», а 21 из группы «1».
- из последних 15-и элементов все из группы «1».

То есть в большей части получившегося блока – более чем 1/3 «исключений».

«КВАДРАТНАЯ ЗМЕЙКА»

Рекурсивный метод для квадратных областей.

Если принять левый верхний элемент за первый, то для квадрата 2х2 возможны два варианта обхода без разрывов:

1	4
2	3

и

1	2
4	3

То есть либо первый переход внутри квадрата был сверху **вниз**, тогда пятым шагом будет переход к левому верхнему элементу квадрата **справа** (или к правому нижнему элементу квадрата **сверху**). Либо наоборот: первый переход внутри квадрата был **вправо**, тогда пятым шагом будет переход к квадрату **снизу** (или **слева**). Переформулируем так:

- если нужно выйти **к правому** (или верхнему) квадрату, то первый шаг – **вниз**, если **к нижнему** (или левому), то первый шаг – **вправо**;
- пройденным путем однозначно задается, к какому квадрату нужно выйти в каждый конкретный момент;
- только в самом начале есть выбор одного из двух вариантов обхода.

Например:

1	2	15	16		20		
4	3	14	13			24	
5	8	9	12				
6	7	10	11	32		28	
					36		
60		56				40	
			62				
64				48		44	

Первый шаг внутри квадрата 2x2 был вправо – значит в результате обхода этого квадрата 2x2 мы должны выйти к нижнему квадрату 2x2;

первый шаг перехода от 2x2 к 2x2 внутри 4x4 был вниз – значит мы должны выйти к правому 4x4;

первый переход внутри квадрата 16x16 был вправо – значит в результате обхода 16x16 – должны прийти к нижнему, и так далее.

Разрывов нет, и каждый элемент принадлежит области, записанной компактно.

Упражнение: Определите, какие числа должны быть в незаполненных клетках примера. Нарисуйте другой пример: при выборе второго элемента квадрата делаем переход не слева направо, а сверху вниз.

Для квадрата 3x3 можно взять такие два «шаблона»:

1	4	5
2	3	6
9	8	7

и

1	2	9
4	3	8
5	6	7

Первое правило будет «противоположным» первому правилу случая 2x2, но остальные два – такие же:

- если нужно выйти **к правому** (или верхнему) квадрату, то первый шаг – **вправо**, если **к нижнему** (или левому), то первый шаг – **вниз**;
- пройденным путем однозначно задается, к какому квадрату нужно выйти в каждый конкретный момент;
- только в самом начале есть выбор одного из двух вариантов.

Но для 3x3 есть еще и такие варианты:

1	6	7
2	5	8
3	4	9

и

1	2	3
6	5	4
7	8	9

Видно, что они совершенно эквивалентны, то есть взаимозаменяемы, поскольку в обоих случаях мы выходим в правый нижний угол. Точно так же эквивалентны и два варианта обхода ими квадрата 9x9:

1					18	19		
		9	10					27
		46	45					28
54					37	36		
55					72	73		
		63	64					81

и

1					54	55		
		9	46					63
		10	45					64
18					37	72		
19					36	73		
		27	28					81

Совершенно аналогично для 5x5, и затем 25x25 и так далее.

Если требуется обойти квадрат размера NxN, сначала определяем, произведением каких простых чисел является N, затем – задаем порядок этих чисел в произведении (а для каждого нечётного множителя – еще и направление обхода). Таким образом будет задан процесс обхода.

Если K, равное числу двоек в произведении, больше одного: $K > 1$, то сторона самого мелкого квадрата должна быть 2^{K-1} или 2^K элементов. Например, если $K=3$, то обход без разрывов 2x3x2x2 (от самого мелкого 2x2 к 6x6, затем к 12x12 и, наконец, к 24x24) невозможен:

		63	64					9	10		
	59			68			5			14	
55					72	1					18
54					37	36					19
	50			41			32			23	
		46	45					28	27		
...	...	...	...	...	...	73	...	...	...	...	...

Каждая ячейка этой таблицы – квадрат 2×2 ; нижние 5 строк пропущены: перейти к двум нижним квадратам 12×12 без разрыва не сможем.

Других ограничений при обходе «квадратной змейкой» нет.

Каждый квадрат со стороной $L > 2$ можно обходить обычной змейкой, а не «квадратной». Это выгодно в том случае, если наиболее различающиеся элементы сгруппированы в противоположных углах квадрата.

Упражнение: Нарисуйте порядок обхода квадрата 25×25 , а затем – квадрата 12×12 . Убедитесь, что разрывов нет, и каждый элемент принадлежит компактно записанной области.

Обход по спирали

Обход по спирали довольно тривиален. Строится квадрат 3×3 , затем 5×5 , затем 7×7 , 9×9 и так далее:

43	44	45	46	47	48	49
42	21	22	23	24	25	26
41	20	7	8	9	10	27
40	19	6	1	2	11	28

39	18	5	4	3	12	29
38	17	16	15	14	13	30
37	36	35	34	33	32	31

Если же строить круги радиуса 2, 3, 4 и т.д., неизбежно будут присутствовать точки разрыва. Спираль может быть и сходящейся. Суть ее можно показать с помощью этой же иллюстрации, только направление движения обратное: от 49 к 1. Кроме того, она может быть с разворотами:

1	2	3	4	5	6	7
24	25	40	39	38	37	8
23	26	41	42	43	36	9
22	27	48	49	44	35	10
21	28	47	46	45	34	11
20	29	30	31	32	33	12
19	18	17	16	15	14	13

Направление изменяется в точках, расположенных на диагонали: в данном примере это «25» и «41».

ОБЩИЕ МОМЕНТЫ ДЛЯ ПРЯМОУГОЛЬНЫХ МЕТОДОВ

В любом случае можно начать с одного из четырех углов, и дальше двигаться в одном из двух направлений: по вертикали и по горизонтали. Первое, то есть положение первого угла, влияния на степень сжатия почти не оказывает, особенно при сжатии изображений. А вот второе, выбор направления, может существенно улучшить сжатие, поскольку области в этих двух случаях (основное направление по вертикали или по горизонтали) будут сгруппированы по-разному.

Например, отсканированный текст лучше сжимать, обходя по вертикали, поскольку в нем больше длинных сплошных вертикальных линий, чем горизонтальных.

ОБЩИЕ МОМЕНТЫ ДЛЯ МЕТОДОВ СЛОЖНОЙ ФОРМЫ

Может возникнуть необходимость пометать уже обработанные точки плоскости, чтобы избежать лишних вычислений, предотвращающих повторное их занесение в D . Тогда есть два основных варианта: либо добавить по одному «флаговому» биту для каждой точки плоскости, либо выбрать (или добавить) значение для «флага», показывающего, что точка уже внесена в D , и записывать это значение на место уже внесенных точек.

Вопросы для самоконтроля

- 1) Какие параметры надо определить, прежде чем сравнивать два алгоритма компрессии?
- 2) Почему некорректно сравнивать временные параметры реализаций алгоритмов компрессии, оптимально реализованных на разных компьютерах? Приведите примеры ситуаций, когда архитектура компьютера дает преимущества тому или иному алгоритму.
- 3) Предложите пример своего класса изображений.
- 4) Какими свойствами изображений мы можем пользоваться, создавая алгоритм компрессии? Приведите примеры.
- 5) Что такое редактируемость?
- 6) Назовите основные требования приложений к алгоритмам компрессии.
- 7) Что такое симметричность?
- 8) Предложите пример своего класса приложений.
- 9) Приведите примеры аппаратных реализаций алгоритма сжатия изображений, с которыми вам приходилось сталкиваться (повседневные и достаточно новые).
- 10) Почему высокая скорость компрессии, высокое качество изображений и высокая степень компрессии взаимно противоречивы? Покажите противоречивость каждой пары условий.