

ЛАБОРАТОРНАЯ РАБОТА 5. РАЗРАБОТКА ПРОГРАММЫ ДЛЯ СЖАТИЯ ЦИФРОВЫХ ИЗОБРАЖЕНИЙ

1.1 Методы компрессии видеоданных

Цифровое представление ТВ-сигналов открывает широкие возможности по синтезу и обработке изображений, однако предъявляет серьезные требования к каналам связи: скорость передачи данных в цифровых потоках телевидения высокой четкости достигает 1.0 Гбайт/с.

Теоретической основой сжатия видеоданных является их *избыточность*: статистическая, по восприятию ТВ-изображений человеком, структурная и по спектру. Под *статистической избыточностью* понимают сильные корреляционные связи между отдельными элементами, строками и кадрами ТВ-изображения: коэффициенты корреляции составляют, соответственно, 0.93, 0.95 и 0.92. Диапазоны статистического влияния, определяемые промежутками времени, когда коэффициенты корреляции спадают до значения $1/e$, составляют примерно 30 элементов и 12 кадров ТВ-изображения. Усредненный по многим реализациям спектр ТВ-сигнала быстро падает по мере роста частоты. В типичных для телевизионного вещания изображениях часто встречаются протяженные яркостные и цветные поля, а резкие переходы и мелкие детали занимают малую долю площади ТВ-кадра. Это особенно заметно, если рассмотреть осциллограммы сигнала и разности сигналов соседних элементов: большую часть интервала разностный сигнал близок к нулю и лишь изредка значительно отклоняется от нуля. Зная значение предыдущего элемента изображения, можно с высокой вероятностью предсказать значение последующего элемента. Такого рода избыточность называют *пространственной*.

Соседние кадры ТВ-изображения очень мало отличаются друг от друга, даже если передаются движущиеся объекты. Межкадровая разность

Commented [AA1]: Мб спектральная? Или «по спектру» будет правильней?

Commented [AA2]: Спектр падает?

близка к нулю на большей части кадра. Это так называемая *временная избыточность* (рисунок 1.1).



Рисунок 1.1 – Соседние кадры ТВ-изображения и межкадровая разность

Commented [AA3]: Инвертированная

Избыточность *по восприятию* связана с особенностями зрения человека: например, цветовая разрешающая способность зрения примерно в четыре раза ниже, чем яркостная. Число воспринимаемых градаций яркости зависит от размеров фрагмента изображения, поэтому *мелкие детали можно квантовать на меньшее число уровней*. К качеству движущихся изображений зрение человека гораздо менее требовательно, чем к статичным кадрам, так что потери информации вплоть до некоторого предела вполне допустимы.

Структурная избыточность обусловлена способом передачи ТВ-сигнала: в нем, например, передаются постоянные по форме сигналы гашения и синхронизации, которые в цифровом сигнале нет необходимости воспроизводить. Устранение этих сигналов снижает цифровой поток на 23%.

Спектральная избыточность проявляется как результат излишне высокой частоты дискретизации. Принятая в телевидении ортогональная структура дискретизации не является оптимальной. Используя интерполяцию и передискретизацию, можно понизить частоту отсчетов ТВ-сигнала и выиграть в объеме передаваемой информации.

Все методы сжатия видеоданных делят на две группы: сжатие *без потери* и *с допустимой (незаметной наблюдателю) потерей* информации.

1.2 Сжатие видеоданных без потери информации

Когда требуется сжать программные файлы или массивы данных, сжатие следует выполнять без каких-либо потерь: утрата хотя бы одного байта может обернуться катастрофой. Известные программы обеспечивают сжатие за счет *оптимального кодирования символов сообщения* (например, для текста этим символом является буква) или за счет *устранения избыточной информации*. Идея метода кодирования информации по Хаффману заключается в том, чтобы *наиболее часто встречающиеся элементы алфавита кодировать короткими кодовыми группами, а наименее часто - длинными*, т.е. использовать кодовые группы переменной длины. Разумеется, выигрыш может быть получен лишь в случае, когда частоты использования отдельных элементов алфавита в типичных сообщениях заметно различаются. Например, для букв русского алфавита рассчитаны вероятности появления каждой буквы. Наиболее часто встречаются буквы *о* и *е*; значительно реже — *э* и *ф*. Это учтено в коде Морзе: например, букве *о* присвоен наиболее короткий код (точка).

Предположим, используемый алфавит состоит из $M = 32$ букв. Для равномерного кодирования символов этого алфавита нужны комбинации из 5 бит, так как $\log_2 M = 5$. Среднюю **информацию** на один независимый символ сообщения в теории информации называют *энтропией*

$$H = \sum_{i=1}^M -p_i \log_2 p_i, \quad (1.1)$$

где p_i — вероятность появления в сообщении символа с номером i .

Каждой букве соответствует энтропия **$H_0 = 5$ бит** только в том случае, когда все буквы равновероятны и $p_i = 1/32$. Энтропия H , рассчитанная с учетом разной вероятности появления букв, всегда меньше. Это означает, что описание всех букв кодовыми комбинациями из 5 бит является избыточным.

Commented [AA4]: Количество информации

Commented [AA5]: Нужно подробнее расписать, что такое H_0 , или перефразировать. Здесь под H_0 понимается частная энтропия, в то время как далее под H_0 понимается средняя энтропия сообщения с равновероятными символами

Избыточность сообщений характеризуют коэффициентом $R = 1 - (H/H_0)$, который может принимать значения от 0 до 1. При $H = H_0$ коэффициент $R = 0$ и избыточность отсутствует; при $H \ll H_0$ $R \rightarrow 1$, что означает малую информативность и высокую степень предсказуемости текста данного сообщения. Применение кода Хаффмана позволяет уравнивать энтропию и среднюю длину кодовой комбинации, нужную для описания одного символа сообщения. При кодировании элементов сообщения кодовыми группами переменной длины возникает проблема их разделения. Чтобы не использовать разделители (пробелы), код Хаффмана наделен свойством *префиксности*: ни одна его кодовая комбинация не является началом другой.

Границы применимости этого метода кодирования очевидны: если элементы данных распределены статистически равномерно, то сжатие невозможно. Лучший коэффициент компрессии равен 8; средний – 1,5; худший – 1,0. Это единственный алгоритм, который никогда не увеличивает длину кодового сообщения. Реализация алгоритма Хаффмана требует двух проходов по сообщению: в первом проходе набирают статистику, во втором – производят собственно кодирование. Если статистика источника сообщения известна априорно, то первый проход исключают и используют заранее рассчитанную кодовую таблицу.

Кодирование длин серий (*RLE - Run Length Encoding*) основано на сравнении значений соседних элементов по строке развертки. Серией называют последовательность элементов, значения которых не отличаются друг от друга. При таком методе кодирования передают данные о значении первого элемента серии или разностного сигнала на границе серий, а также длины серии или позиции последнего элемента серии относительно начала строки или последнего элемента предыдущей серии. Сообщение кодируют двумя байтами: первый байт – счетчик повторений элемента в серии; второй

Commented [t6]: А разве, например, код Шеннона-Фано в каком-то случае может увеличить длину кодового сообщения? Может я не правильно понял смысл фразы, но на мой взгляд, нужно заменить это предложение на что-нибудь менее вызывающее, например: код Хаффмана является оптимальным по критерию минимума средней длины кодового слова.

– значение элемента. Достигнутые коэффициенты компрессии: лучший – $(32 \div 64)$; средний – $(2 \div 3)$; худший – 0,5. Алгоритм ориентирован на изображения с небольшим количеством цветов: деловую и научную графику. Примером использования данного метода кодирования являются файлы изображений с расширением *PCX*. Кроме того, кодирование длин серий может быть использовано в *BMP*- и *Targa*-файлах.

Накопленный опыт показывает, что эффективность кодирования изображений без потерь информации невелика и зависит от исходных изображений. Лишь двухградационные (черно-белые) изображения можно действительно сжимать при использовании кодирования длин серий: примером служит факсимильная передача. В качественном полутоновом изображении не только содержатся практически все уровни между 0 и 255, но к тому же многие из них еще и встречаются с близкими частотами повторения. Поэтому сжатие данных методом кодирования по Хаффману в большинстве случаев неэффективно. Реальное сжатие палитровых изображений (*Super-VGA*, *Hi-Color*) составляет от 5% до 50%. Для изображений в режиме *True Color* достойное упоминания сжатие достижимо только в виде исключения. Иногда происходит даже увеличение объема файла, так как в нем должна быть записана дополнительная информация для интерпретации сжатых данных. Признано, что кодирование без потери информации не решает задачу сжатия видеоданных и имеет лишь вспомогательное значение.

Достижение коэффициентов сжатия, достаточных для реализации систем цифровой передачи видеоданных в полосе частот аналогового сигнала, оказалось возможным при допущении некоторой потери информации. Различают системы сжатия статичных и движущихся ТВ-изображений. В первом случае используют внутрикадровые статистические связи между соседними элементами и строками изображения; во втором – дополнительно учитывают статистические связи между соседними кадрами. В системах

сжатия с потерей информации применены известные и хорошо себя зарекомендовавшие методы: дискретно-косинусное преобразование ДКП и кодирование с предсказанием ДИКМ. Кроме того, в существенной мере реализуют возможности устранения избыточности по восприятию человеческим зрением.

1.3 Внутрикадровая компрессия видеоданных (стандарт *JPEG*)

В большинстве пакетов программ для обработки изображений используют стандарт внутрикадрового кодирования *JPEG* (*Joint Photographic Expert Group*). Основой стандарта *JPEG* является ДКП (английское название *DCT - discrete cosine transform*) и тот факт, что коэффициенты ДКП быстро уменьшаются по мере роста частоты. Подвергать ДКП весь кадр не имеет смысла по двум причинам. Первая причина принципиальная: несмотря на крайне низкую интенсивность высокочастотных составляющих ТВ-сигнала их нельзя исключать, так как они обеспечивают четкость изображения. Вторая причина обусловлена большой размерностью массива изображения и, как следствие, значительными вычислительными сложностями. Поэтому изображение делят на блоки и подвергают ДКП каждый блок в отдельности. В большинстве блоков мелкие детали практически отсутствуют и можно не передавать коэффициенты ДКП, значения которых ниже некоторого порога. В тех блоках, где присутствуют мелкие детали и резкие яркостные переходы, коэффициенты ДКП даже могут расти по мере роста частоты и их транслируют, чем сохраняют исходную четкость изображения.

Блок представляет собой матрицу $8 \times 8 = 64$ элементов и является *базовой структурной единицей* (сигнальной матрицей), над которой осуществляют все операции кодирования, в частности, ДКП и квантование полученных коэффициентов. В ТВ-кадре создают $704/8 = 88$ зон по горизонтали и

576/8=72 зоны по вертикали, что в общей сложности дает 72*88=6336 блоков, подлежащих ДКП в реальном масштабе времени. Матрица коэффициентов ДКП (трансформант)

$$\mathbf{A} = \mathbf{\Phi} \mathbf{U} \mathbf{\Phi}^T, \quad (1.2)$$

где матрица $\mathbf{\Phi}$ составлена из отсчетов базисных функций.

Отсчеты изображения $u(x, y)$ зависят от пространственных координат (номеров отсчетов) k и l , изменяющихся от 0 до 7. Алгоритмом *JPEG* предусмотрено, что перед выполнением ДКП из значений яркости в видеоданных вычитают число 128, благодаря чему значения отсчетов располагаются в диапазоне между -128 и +127. После умножений матриц, согласно выражению (1.2), получим матрицу коэффициентов ДКП \mathbf{A} . Коэффициенты ДКП $A(m, n)$ задают изображения в частотной плоскости, где m и n также изменяются от 0 до 7. Изображение представлено в виде суммы пространственных косинусоид, где $A(m, n)$ – их амплитуды, а координаты m, n – их частоты. Значение $A(0, 0)$ – пропорционально среднему уровню яркости в блоке. «Высокие» частоты передают мелкие детали изображения и в среднем имеют значительно меньшие амплитуды относительно $A(0, 0)$.

Двумерный спектр ДКП имеет важную особенность: основная энергия частотных составляющих сосредоточена в области нулевых и низких частот. Амплитуды высокочастотных составляющих либо малы, либо просто равны нулю.

Следующим шагом является квантование коэффициентов $\alpha[m, n]$ и сведение их к диапазону (-127÷127). Дело в том, что при выполнении преобразования динамический диапазон коэффициентов ДКП возрастает в 8 раз и составляет либо (0÷2047), либо (-1023÷1023) дискретных уровней. Чтобы избежать перехода к 11-битовому кодированию, нужно разделить полученные коэффициенты на 8, а результат округлить до ближайшего целого значения.

Commented [AA7]: Не указано, для чего это делают

Commented [AA8]: M6 -128?

Commented [AA9]: -1024?

Таблица 1.1. Пример матрицы **A** коэффициентов ДКП

$n \backslash m$	0	1	2	3	4	5	6	7
0	395	-132	78	19	5	0	3	0
1	130	-30	-55	31	9	0	12	0
2	150	70	0	3	0	0	0	0
3	-40	35	23	1	11	9	-12	6
4	23	-11	7	3	0	0	0	0
5	0	14	0	0	15	0	13	0
6	0	8	-11	0	8	2	-15	0
7	0	5	-4	5	0	1	-3	7

Commented [AA10]: Добавить ссылку в тексте

Пусть, например, передается поле с постоянной яркостью, близкой к максимальной (244 дискретный уровень). Матрица **U** составлена из чисел $244 - 128 = 116$. После ДКП в матрице **A** отличен от нуля только один элемент $\alpha(0, 0) = 928$. Значение средней яркости будет восстановлено после деления на 8: $928/8 = 116$.

Глаз человека наиболее чувствителен к градиентам яркости крупных деталей, поэтому для коэффициентов $\alpha(0, 0)$ шаг квантования должен быть минимален. Для более высоких пространственных частот можно применять квантование на меньшее число уровней. Конкретный шаг квантования назначают согласно кодовой таблице *JPEG*-стандарта (таблица 1.2). Для формирования таблицы коэффициентов в ряде случаев используют следующий алгоритм:


```
cv::Mat gamma = cv::Mat::zeros(8, 8, CV_32S);
int quality = 5; // Ввод коэффициента масштаба квантования
for (int i = 0; i < 8; i++) {
    for (int j = 0; j < 8; j++) {
        gamma.at<int>(i, j) = 8 + (i + j) * quality;
    }
}
```

Здесь **quality** (качество) играет роль *масштаба квантования*, который может быть принят от 1 до 31.

Commented [AA11]: Сделать акцент на том, что слагаемое 8 в алгоритме расчета коэффициентов как раз и учитывает «деление на 8», о котором говорилось ранее.

Таблица 1.2. Кодовая таблица

i/j	0	1	2	3	4	5	6	7
0	8	13	18	23	28	33	38	43
1	13	18	23	28	33	38	43	48
2	18	23	28	33	38	43	48	53
3	23	28	33	38	43	48	53	58
4	28	33	38	43	48	53	58	63
5	33	38	43	48	53	58	63	68
6	38	43	48	53	58	63	68	73
7	43	48	53	58	63	68	73	78

Назначение масштаба квантования позволяет управлять степенью сжатия и качеством восстановленного изображения.

Значения в ячейках с номерами i и j таблицы 1.2 являются коэффициентами, на которые делят соответствующие им по местоположению элементы матрицы **A**. После округления до целых чисел получаем новую матрицу, в которой значительно больше нулей (таблица 1.3). Если все же коэффициенты разложения отличны от нулей, для их кодирования применяют сокращенные по длительности кодовые группы. Например, для спектральных составляющих $\alpha(7,7)$ число уровней квантования составит

$2047/78 \approx 26$, что позволяет применить для их передачи $(4 \div 5)$ -битные кодовые группы.

Таблица 1.3. Коэффициенты преобразования после деления

$n \backslash m$	0	1	2	3	4	5	6	7
0	49	-10	4	0	0	0	0	0
1	10	-2	-2	1	0	0	0	0
2	8	3	0	0	0	0	0	0
3	-2	1	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0

Исключением является кодирование постоянной составляющей – элемента $\alpha(0,0)$, для которого коэффициент деления фиксирован и равен 8: значения этого элемента всегда передают не менее чем 256 уровнями, иначе будет заметно изменение яркости от блока к блоку (эффект шабровки, или шахматной структуры).

При восстановлении изображения элементы матрицы **A** умножают на соответствующие элементы кодовой матрицы (с учетом масштаба квантования), а полученные числа округляют. Эта операция восстанавливает исходные значения коэффициентов матрицы **A** с точностью до ошибок округления, возрастающих по мере роста номеров (m, n) . Так устраняют *избыточность по восприятию*, поскольку высокочастотные составляющие передают с ошибками квантования, которые зритель практически не замечает.

Следующий шаг сводится к преобразованию матрицы в одномерную последовательность кодовых групп путем *зигзаг-сканирования*. Схема

Commented [t12]: Элементы матрицы **A** уже округлены до целых. Кодовая матрица также содержит только целые элементы. Полученные после перемножения числа округлять не нужно.

коэффициентов ДКП неравномерно: для всех коэффициентов наиболее вероятны малые значения. Чтобы усилить это свойство для коэффициентов $\alpha(0,0)$, квантованию подвергают разности средних яркостей соседних блоков, которые оказываются в большинстве случаев малыми в силу существенной корреляционной связи между этими коэффициентами. Для коэффициентов с отличными от нуля значениями номеров (m, n) эти связи отсутствуют. Таблица кодирования по Хаффману определена стандартом.

В зависимости от информационного содержания фрагмента изображения длина кодовой последовательности, соответствующей блоку, может быть разной. В результате информационный поток на выходе кодера будет неравномерным. Для организации передачи по каналу связи необходим буфер памяти, в который данные записывают по мере их поступления, а считывание производят с постоянной скоростью. Чтобы не происходило переполнения или опустошения этого буфера, систему кодирования делают адаптивной. Когда резервы буфера иссякают, блок управления автоматически увеличивает масштаб квантования *quality* и скорость поступления данных в буфер снижается ценой роста ошибок квантования. Если же буфер опустошается, блок управления уменьшает масштаб квантования, что приводит к улучшению качества восстановленного изображения.

При передаче цветных изображений описанный алгоритм можно применить для *RGB*-данных, однако обычно так не делают. Сначала отделяют яркостную информацию от цветовой. Яркостную информацию кодируют непосредственно, а цветовую подвергают субдискретизации (*subsampling*). Сначала осуществляют пересчет из системы *RGB* в *YUV*, где *Y* - яркостной сигнал, а *U* и *V* соответствуют цветоразностным сигналам *R-Y* и *B-Y*. Значения *Y*-сигнала соответствуют яркости каждого элемента изображения, а цветовых – в зависимости от стандарта субдискретизации (рисунок 1.3). Яркостные элементы изображения показаны треугольниками без штриховки,

а цветовые заштрихованы. Например, в стандарте 4:1:1 цветовую информацию передают от каждого четвертого элемента, но в каждой строке; в стандарте 4:2:0 – через строку. Таким простым путем производят сжатие изображения более чем на 50%, причем после восстановления потери качества изображения практически незаметны.

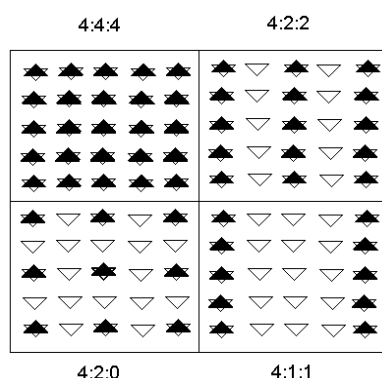


Рисунок 1.3 – Форматы представления видеоданных

Commented [AA14]: Может быть эту картинку перенести в 1 лабораторную, где рассказывается про дискретизацию?

1.4 Межкадровая компрессия видеоданных (стандарт *MPEG*)

Для сжатия ТВ-изображений, представляющих собой последовательность ТВ-кадров, применяют кодирование с предсказанием (ДИКМ - дифференциальную импульсно-кодовую модуляцию): по каналу связи передают лишь разность между истинным и предсказанным значениями отсчетов ТВ-сигнала. При межкадровом предсказании вперед текущий отсчет принимают равным отсчету с теми же координатами, но предыдущего кадра. В качестве предсказывающего устройства используют память на ТВ-кадр. Разностное изображение (см. рисунок 1.1) содержит ошибку предсказания, причем характерны обширные области с нулевыми значениями. Это особенно эффективно при передаче неподвижных изображений. Если изображение смещается, применяют компенсацию движения, для чего определяют векторы перемещения движущихся фрагментов изображения при

переходе от кадра к кадру. Эти векторы позволяют определить новое положение предсказанного отсчета (скомпенсировать его перемещение) и сохранить высокую точность предсказания.

Общее название стандартов сжатия движущихся изображений MPEG (Moving Picture Expert Group). Разработаны 4 варианта стандарта: MPEG-1, MPEG-2, MPEG-4 и MPEG-7. Стандарт MPEG-1 ориентирован на системы записи CD-ROM и низкоскоростные каналы передачи изображения со скоростями цифрового потока 1.5 Мбит/с и меньше. Стандарт MPEG-2 создан для кодирования сигналов вещательного телевидения, в том числе высокой четкости. Следующая модификация MPEG-4 первоначально была предназначена для организации телеконференций при передаче видеоданных по цифровым телефонным каналам. В настоящее время стандарт принят для наземного телевизионного вещания в ряде стран, в том числе в России. Стандарт MPEG-7 (Multimedia Content Description Interface) призван определить универсальную многоуровневую структуру описания аудио и видео информации. На высшем уровне прописываются свойства файла, такие как название, имя создателя, дата создания и т.д. На следующем уровне указывают особенности сжимаемой аудио или видео информации – цвет, текстура, тон или скорость. Формат сжатия MPEG-7 содержит богатый набор стандартных инструментов для описания мультимедийных данных, которые делают возможным автоматическое выделение информации, как по общим, так и по семантическим признакам, связанным с восприятием информации человеком.

Базовым объектом кодирования в стандарте *MPEG-2* является ТВ-кадр. Это обеспечивает преемственность со стандартом внутрикадрового кодирования *JPEG*. В *MPEG-2* используют три вида предсказания: *внутрикадровое, межкадровое предсказание вперед с компенсацией движения, межкадровое двунаправленное с компенсацией движения.*

Кодирование и передачу информации выполняют по макроблокам размером 16×16 элементов (рисунок 1.4). В состав каждого макроблока входят четыре блока 8×8 элементов, которые несут информацию о яркости изображения. За счет субдискретизации в макроблок входят лишь два блока, несущие информацию о цветности (заштрихованные элементы на рисунке 1.4). В результате каждый из макроблоков состоит из шести блоков, четыре из которых несут информацию о яркости, а по одному определяют цветовые U - и V -компоненты. Всего таких макроблоков 396. Следующие друг за другом макроблоки объединяют в независимые друг от друга разделы (*slices*), максимальным числом до 396. В предельном случае на кадр изображения приходится только один раздел из 396 макроблоков.

Commented [AA15]: Не очень понятно написано

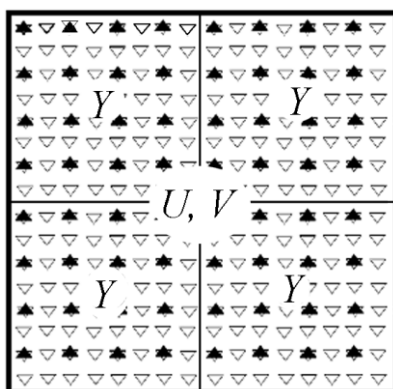


Рисунок 1.4 – Макроблок

ТВ-кадры объединяют в группы (*GOP - Group of Pictures*), представляющие из себя минимальный повторяемый набор из 12 последовательных кадров, которые могут быть декодированы независимо от других ТВ-кадров (рисунок 1.5). На первом этапе кодирования обрабатывают с применением алгоритмов *JPEG* так называемые I -кадры (*intra*), которые служат опорными при межкадровом предсказании P - и B -кадров (*predicted* и *bidirectionally predicted*). Типичной является группа вида: $I_0, B_1, B_2, P_3, B_4, B_5, P_6, B_7, B_8, P_9, B_{10}, B_{11}, I_{12}, B_{13}, B_{14}, P_{15}$ и т.д.

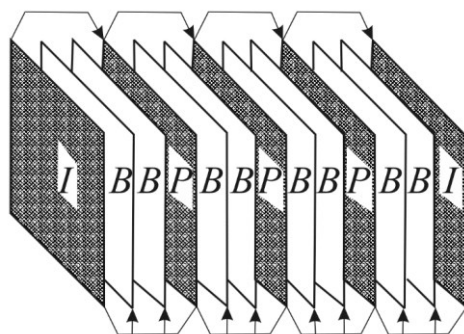


Рисунок 1.5 – Межкадровое кодирование

Все макроблоки *I*-кадров являются опорными (тип *intra*) и подвергаются независимому преобразованию по стандарту *JPEG*. Кодирование *P*-кадров выполняют с использованием алгоритмов компенсации движения и предсказанием вперед по предшествующим *I*- и *P*-кадрам (на рисунке 1.5 показано стрелками). По каналу связи передают лишь ошибки предсказания. В результате в *P*-кадрах в три раза выше степень сжатия, чем в *I*-кадрах.

Commented [AA16]: Чуть выше это уже было написано

В кодирующем устройстве для предсказания использована память на ТВ-кадр. *B*-кадры кодируют с использованием предсказания по соседним *I*- и *P*-кадрам (см. рисунок 1.5). Предсказание оказывается более точным, а объем передаваемых по каналу связи данных об ошибках предсказания сокращается. Поэтому степень сжатия в *B*-кадрах в полтора раза выше, чем в *P*-кадрах. Для восстановления *B*-кадров одновременно нужны предшествующие и следующие за ними *I*- и *P*-кадры. Нужно увеличить объемы памяти на передающем и приемном концах, а также изменить порядок передачи кадров. Сначала нужно передать *I*-кадр, по нему восстановить *P*-кадр, затем восстановить находящиеся между ними *B*-кадры и т.д. Новый порядок кодирования и передачи кадров в группе: *I*0, *P*3, *B*1, *B*2, *P*6, *B*4, *B*5, *P*9, *B*7, *B*8... Группы могут быть открытыми и закрытыми. Последним элементом закрытой группы является *P*-кадр, предсказанный на основе кадра этой же

Commented [AA17]: Выше это уже было написано

Commented [AA18]: Предложение соотрится обособленно

группы; открытой – *B*-кадр, для восстановления которого используют *I*-кадр последующей группы. Преимущество закрытых групп в том, что они позволяют выполнять прямые монтажные переходы перед очередным *I*-кадром без декодирования цифрового потока; недостаток – более сложный кодер, так как нужно, чтобы группа занимала строго фиксированный интервал времени.

В стандарте *MPEG-2* используют метод межкадрового предсказания *P*- и *B*-кадров с компенсацией перемещения по макроблокам. Напомним, что макроблок – это квадратная 16×16 элементов матрица. Именно для них определяют *векторы смещения* относительно *I*- и *P*-кадров. Вводят понятие зоны поиска перемещенного макроблока, которая обычно в 4 раза больше его размеров и равна 64×64 , что позволяет надеяться, что при быстром движении блок к следующему кадру не выйдет за пределы зоны поиска. Всего получается 9 зон поиска по вертикали и 11 зон по горизонтали. Вычисляют абсолютные значения модуля разности отсчетов сигналов (целевой функции) для всех возможных смещений (слепой поиск) макроблоков предыдущего (опорного) и последующего кадров. Вектор смещения $\mathbf{s} = (s_x, s_y)^T$ составлен из разностей координат старого положения макроблока и нового, где значение целевой функции минимально.

Более быстрым алгоритмом является логарифмический метод (деления отрезка пополам, или *дихотомии*), который выполняют за несколько шагов, в каждом из которых рассчитывают и сравнивают по 9 значений целевой функции в узловых точках. На последующих шагах область поиска сужают, повышая точность предсказания.

Пример реализации такого поиска за 3 шага дан на рисунке 1.6, где цифрами показаны пробные точки (положения центров макроблока) на каждой итерации поиска. Вычисляя значения целевой функции для пробных точек, сужают зону поиска на каждом шаге примерно в три раза по

Commented [AA19]: Двойная интерпретация. А по площади – в 16 раз больше его размеров

Commented [AA20]: На рисунке же вроде метод 3SS, а не логарифмический.

площади. Линейный размер шага меняется как 4:2:1, а диапазон возможных значений вектора смещения в данном примере равен (7,7).

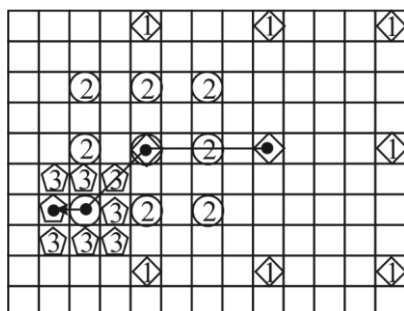


Рисунок 1.6 – Совмещение макроблоков методом дихотомии

Известен также метод телескопического поиска, рекомендуемый для последовательных *B*-кадров, основанный на использовании в качестве стартовой оценки вектора смещения текущего макроблока значения вектора из предыдущего *B*-кадра для этого же макроблока.

После того как все перемещения макроблоков выполнены (движения объектов скомпенсированы) вычисляют разность между предсказанными и фактическими значениями отсчетов в сигнальных матрицах. Выигрыш достигают за счет того, что ошибки предсказания оказываются существенно меньшими или даже равными нулю. Однако на практике полностью скомпенсировать изменение содержания изображений в соседних кадрах обычно не удается, так как одновременно с движением происходят изменения размеров и ракурса объектов. Полученные матрицы ошибок предсказания кодируют с применением ДКП так же, как и *I*-кадры, с той лишь разницей, что для разностного сигнала не приходится надеяться на существенные корреляционные связи между элементами. Поэтому высокие частоты столь же важны, как и низкие. По умолчанию все значения матрицы квантования (см. таблицу 1.2) принимают одинаковыми и равными 16. Сведения о

значениях векторов смещений для всех макроблоков включают в общий поток цифровых данных и используют при декодировании видеоданных.

Обычно смещения изображения внутри кадра незначительны и зачастую вектор смещения равен нулю. Тогда после кодирования все ДКП-коэффициенты блока также могут оказаться равными нулю. В этих случаях достаточно поставить определенную стандартом метку (флаг) и сразу перейти к следующему блоку. Для таких пропускаемых (*skipped*) блоков достигается самый короткий код. Аналогичная ситуация возникает с макроблоком, когда все его блоки имеют статус *skipped*, и соответствующие векторы смещения также равны нулю. В экстремальных случаях (например, при монтажных переходах, переключениях камер) ошибка может быть так велика, что выигрыша нет по сравнению с кодированием данного кадра как опорного (типа *I*).

Определение векторов смещений, использование различных стратегий предсказания требуют использования сложных вычислительных процедур, поэтому кодирующее устройство получается гораздо сложнее и дороже, чем декодирующее.

В результате видеокомпрессии происходят необратимые искажения изображений, которые при значительных коэффициентах сжатия могут быть замечены наблюдателями. Общий недостаток систем сжатия, основанных на разбиении всего изображения на блоки, заключается в том, что при больших коэффициентах сжатия восстановленные блоки плохо стыкуются друг с другом и становится заметной блочная структура полученного изображения. Еще раньше появляются *артефакты* – различного рода посторонние узоры (муары), которые прежде всего зависят от содержания изображения и алгоритма компрессии. Количество и интенсивность артефактов сильно зависят от степени сжатия, причем эта зависимость нелинейная. Артефакты проявляются на тех участках изображения, где содержится много

мелких деталей, резких яркостных и цветовых переходов. Их появлению способствуют быстрое движение объектов, смена сюжета, изменения фокусного расстояния объектива, панорамирование при съемке. В этих обстоятельствах происходит декорреляция ТВ-изображения, и система компрессии перегружается (переполняется буферная память). Чтобы справиться с возникшей ситуацией, вводят более грубое квантование, что и приводит к ошибкам при воспроизведении сжатых изображений. Ошибки проявляются в виде нестабильности контуров, появления окантовок и шумов в областях резких яркостных и цветовых переходов. При плохих соотношениях сигнал-шум в исходном ТВ-сигнале сжатие применять вообще нецелесообразно. Следует предостеречь от многократного применения сжатия к одному и тому же изображению, так как ошибки быстро накапливаются.

При формировании ТВ-программ используют сигналы от разных источников: цифровых видеокамер и видеомагнитофонов, систем спутникового телевидения и пр., где могут быть применены различные системы сжатия или даже одни и те же, но с разными параметрами. В этих условиях могут возникнуть конфликтные ситуации, в результате которых искажения резко возрастают. Свою лепту вносят системы нелинейного видеомонтажа, где часто применяют компрессию 10:1. Сказанное не опровергает систем сжатия, но требует внимательности при их использовании.

Commented [AA21]: Библийский термин в учебном пособии

2 Задание на моделирование

1. Оцените возможности по энтропийному кодированию исходного изображения, квантованного на 256 уровней. Используйте данные гистограммы распределения уровней яркости и просчитайте вероятности появления этих уровней; энтропию источника сообщения по формуле (1.1), а также избыточность сообщений $R = 1 - (H/H_0)$, где H_0 составляет 8 бит.

2. Произведите ДКП преобразование для блоков 8x8 (можно воспользоваться готовой функцией из библиотеки `opencv`). Постройте гистограмму появления коэффициентов ДКП по всем блокам, просчитайте вероятности появления определенных значений коэффициентов; энтропию источника сообщения по формуле (1.1), а также избыточность сообщений $R = 1 - (H/H_0)$, где H_0 составляет 8 бит.

3. Используйте либо таблицу кодовую таблицу *JPEG*-стандарта (табл. 1), либо программу формирования коэффициентов кодовой таблицы. Выполните деление коэффициентов ДКП в блоках на соответствующие элементы кодовой таблицы и округление результатов путем отбрасывания дробной части.

4. Реализуйте зигзаг-сканирование коэффициентов в блоках и кодирование длин серий (*RLE*-кодирование). Напоминаю, что *RLE*-кодирование заключается в подсчете числа повторяющихся значений данных и их передаче в виде двух чисел: первое число означает число повторений, а второе – собственно передаваемое значение конкретного отсчета. Например, последовательность чисел 27; 3; 5;5; 0;0;0;0;0;0; должна быть закодирована как 1;27;1;3;2;5;6;0. Не забудьте поставить код конца каждого блока.

5. Подсчитайте число бит закодированного Вами сообщения «Портрет» и оцените полученный выигрыш. Реальный выигрыш будет еще большим, так как не использовано энтропийное кодирование по Хаффману!

6. Восстановите изображение и сравните с оригиналом.

Контрольные вопросы

1. Перечислите основные виды избыточности, используемые при разработке методов сжатия изображения.

Commented [AA22]: Почему «рендеринг» видео занимает много времени, а воспроизведение происходит в режиме реального времени?
Какие виды артефактов появляются при сжатии видео? В каких случаях они наиболее выражены?

2. Что значит методы сжатия с потерями информации и без потерь? Приведите примеры методов сжатия информации без потерь.
3. Перечислите и поясните основные шаги метода сжатия по стандарту JPEG.
4. Какие из перечисленных шагов JPEG реализуют сжатие информации с потерями, а какие без потерь?
5. Сравните стандарты сжатия видеоданных MJPEG и MPEG.
6. Какие типы кадров используют в стандарте сжатия MPEG, и чем они друг от друга отличаются?
7. Какой профиль MPEG обеспечивает более высокий уровень компрессии с использованием или без использования В – кадров?
8. Каким образом в стандарте MPEG реализуют компенсацию движения?