Кодирование графики

Итак...

Все компьютерные графические изображения разделяют на два типа: **растровые** и **векторные**.

При использовании **растровой** графики исходное графическое изображение в процессе кодирования разбивается на отдельные маленькие фрагменты - **точки** (**пиксели**), причём каждой точке присваивается код её цвета. Информация о каждой точке (код её цвета) хранится в видеопамяти компьютера.

При использовании **векторной** графики изображение формируется из **объектов** (точка, линия, окружность, прямоугольник и т. д.), которые хранятся в памяти компьютера в виде графических примитивов и описывающих их математических формул. объём.

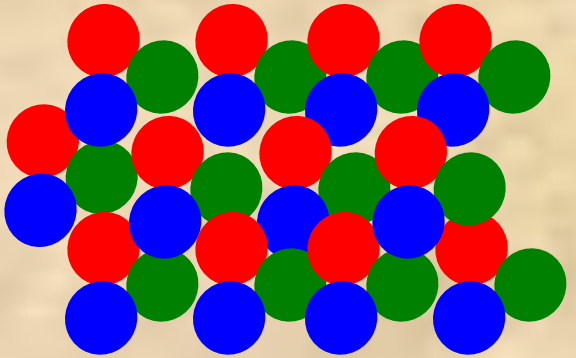
Растр. Формирование изображения

Изображение на экране монитора, телевизора или напечатанное принтером на листе бумаги кажется нам непрерывным, сплошным.

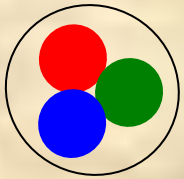
Однако это не так!

Изображение состоит из отдельных светящихся или отражающих точек, т.е оно **дискретизировано**.

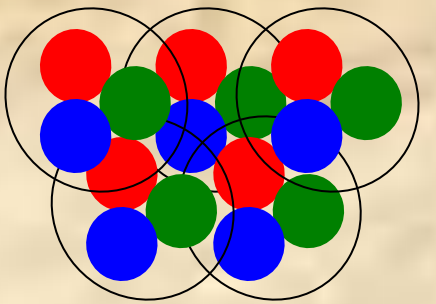
В этом можно убедиться, если близко посмотреть на экран телевизора : изображение состоит из отдельных светящихся точек красного, зеленого и синего цветов.



Соседние точки люминофора красного, зеленого и синего цветов образуют **триаду точек**, которая воспринимается нами как одна точка, цвет которой зависит от интенсивности составляющих цветов (RGB)



триада точек на экране

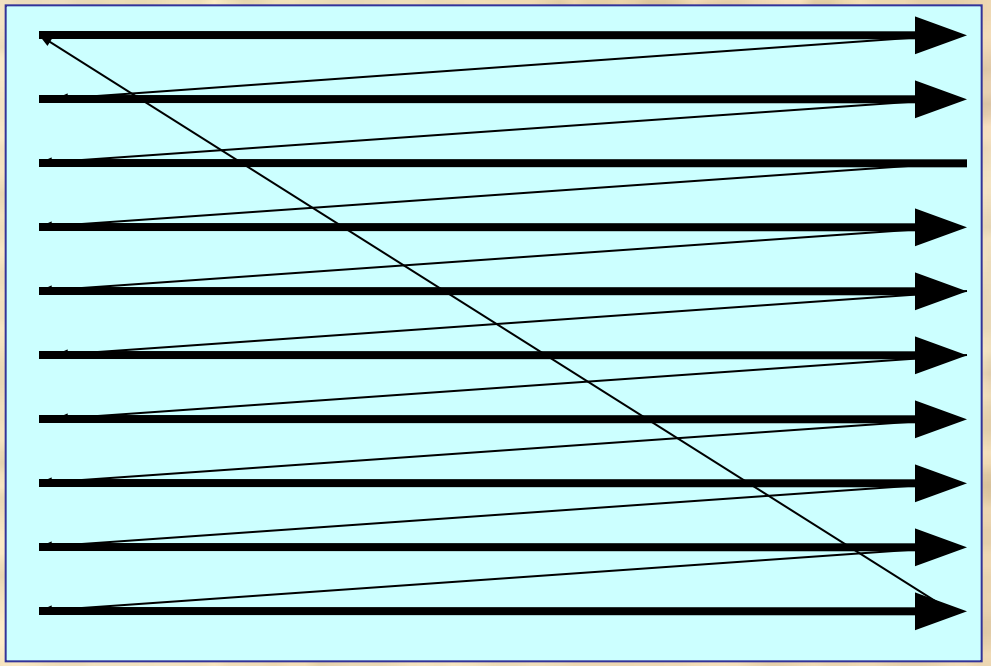


фрагмент экрана монитора из нескольких триад

При формировании изображения на экране используются **свойства нашего зрения**:

1. Объекты, имеющие малое угловое разрешение, глаз не различает – они сливаются в один объект
2. Инертность зрения – глаз не успевает различить отдельные объекты, если они ”мелькают” перед глазами с частотой свыше 20 Гц (больше 20 объектов в секунду)

Формирование изображения на экране



Модулированный электронный луч очень быстро пробегает экран монитора, формируя один кадр

линии, создающие изображение

линии обратного хода (погашены)

Формирование одного кадра на экране кинескопа (замедлено)

В секунду меняется 50-100 кадров, создавая эффект непрерывного движения

Расчет графического файла

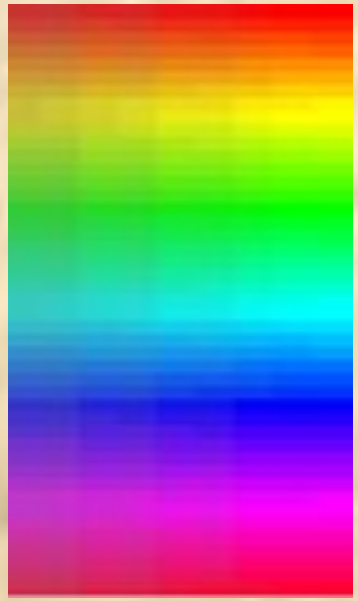
Качество кодирования изображения зависит от 2-х параметров:

Во-первых, качество кодирования изображения тем выше, чем меньше размер точки и соответственно большее количество точек составляет изображение

Во- вторых, чем больше количество цветов, то есть больше возможных состояний точки изображения, используется, тем более качественно кодируется изображение (каждая точка несет большее количество информации).

Используемый набор цветов образует

цветовую палитру.



Объем графического файла (картинки) зависит:

1. От числа пикселей в картинке, которое равно произведению ширины изображения (в пикселях) на его высоту.
2. От того, сколько бит информации необходимо для кодирования одного пикселя. Эта величина называется **глубиной цвета I .**

ОБЪЕМ ФАЙЛА = A ×B × I

где:
A – ширина изображения в пикселях;
B – высота изображения в пикселях;
I – глубина цвета в битах

Вспомним…

Количество разрядов двоичного кода (бит), используемого для кодирования цвета точки, называется **глубиной цвета**, а количество точек в изображении по горизонтали и вертикали называется **разрешающей способностью** экрана .

В настоящее время используются значения глубины цвета 4, 8, 16 или 24 бит на точку и разрешающая способность экрана 640 x 480, 800 x 600, 1024 x 768 и 1280 x 1024 точек по горизонтали и вертикали, соответственно.

Глубина цвета

- это количество бит, отводимых для кодирования одного пикселя.

Если для кодирования одного пикселя взять:

– 1 бит, то с его помощью мы можем получить только 2 цвета: черный (0) и белый (1) , т.е. черно-белое изображение;
– 2 бита – 4 цвета (00,01,10,11);
– 8 бит – 2^8 цветов = 256 цветов …и т.д.

Таким образом, число цветов можно определить по формуле:

$$N=2^I$$

где:

– N – количество цветов;
– I – битовая глубина цвета.

Вывод: чем больше бит применяется для кодирования 1 пикселя, тем больше цветов и реалистичнее изображение, но и размер файла тоже увеличивается.

Таким образом:

Объем файла точечной графики - это произведение ширины и высоты изображения в пикселях, умноженное на глубину цвета.

При этом совершенно безразлично, что изображено на фотографии. Если все три параметра одинаковы, то размер файла без сжатия будет одинаков для любого изображения.

Примеры расчета графических файлов

Пример 1: Определить размер 24 - битного графического файла с разрешением 800×600.

Решение:

Из условия файл имеет следующие параметры:

ширина A = 800 пикселей;
высота B = 600 пикселей;
глубина цвета I = 24 бит (3 байта).

Тогда по формуле:

ОБЪЕМ ФАЙЛА(V) = A ×B × I

V = 800×600×24 = 11520000 бит = 1440000 байт = 1406,25 Кбайт = =1,37 Мбайт.

Ответ: V = 1,37 Мб

Примеры расчета графических файлов

Пример 2: В процессе оптимизации изображения количество цветов было уменьшено с 65536 до 256. Во сколько раз при этом уменьшился объем файла.

Решение:

Из формулы $N=2^I$ следует, что глубина цвета $I=\log_2 N$.

Тогда глубина до оптимизации $I_1 = \log_2 65536 = 16$ бит, а после оптимизации $I_2 = \log_2 256 = 8$ бит.

При этом, размеры картинки в пикселях не изменились. Используя формулу для вычисления объема файла, имеем:

$$V_1 = a \times b \times 16 = 16ab \text{ и } V_2 = a \times b \times 8 = 8ab.$$

Составляем пропорцию: $V_1 : V_2 = 16ab : 8ab$.

То есть $V_1 : V_2 = 2$

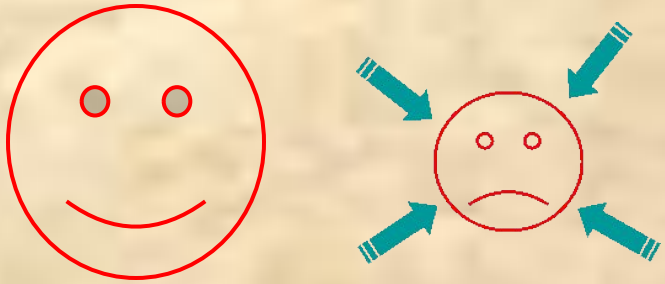
Ответ: объем изображения уменьшился в два раза.

Заключение

Итак, размер графического файла зависит от размеров изображения и количества цветов.

При этом качественное изображение с 24 или 32-битным кодированием получается довольно большим (мегабайты).

Это очень неудобно для хранения и передачи изображений (особенно в сети Интернет).



Поэтому графические файлы подвергают оптимизации