Московский Гуманитарный Университет

Реферат на тему:

Цвет и графика на ЭВМ

Выполнили студенты

Группы СКС101\_133:

Власенко А.В.

Москва 2010

Содержание

Понятие цвета с точки зрения ЭВМ и цели его применения в ИС

Основные принципы хранения в памяти ЭВМ графической информации

Цвет аддитивный и субтрактивный

Индексированный цвет, работа с палитрой

Таблица «безопасных» цветов

Модели цвета на ЭВМ

RGB-модель

Цветовая модель CMYK

Особые взаимоотношения двух цветовых моделей

Цветовая модель HSB

Цветовая модель HSV

Цветовая модель HLS

Цветовые модели Y\*\*

Цветовые модели YUV, YPbPr и YCbCr

Цветовая модель YIQ

Цветовые модели CIE L\*u\*v\* и CIE L\*a\*b\*

Точность воспроизведения цвета

Виды графики на ЭВМ и особенности её применения в ИС

Основные понятия. Представление цвета в машинной графике. Растровая и векторная графика. Понятие растра

Масштабируемая векторная графика (Scalable Vector Graphics)

Сравнение растровой и векторной графики

Классификация изображений и преобразования

Ввод изображения

Преобразования изображений

Основные области применения компьютерной графики

Научная графика

Деловая графика

Конструкторская графика

Художественная и рекламная графика

Компьютерная анимация

Мультимедиа

Особенности применения графики в Интернет

Создание графики

Форматы графических файлов

Создание фона HTML-документа

Прозрачность для GIF и PNG-изображений

Кодирование звуковой информации

Понятие цвета с точки зрения ЭВМ и цели его применения в ИС

Основные принципы хранения в памяти ЭВМ графической информации

Минимальной единицей изображения является символ, при отображении графики картинка строится из отдельных элементов - ПИКСЕЛОВ (от английских слов PICture ELement, означающих "элемент картинки"). Очень часто пиксель совпадает с точкой дисплея, но это совсем необязательно: например, в некоторых видеорежимах 1 пиксель может состоять из 2 или 4 точек экрана.

Каждый пиксель характеризуется цветом. Как и вся остальная информация в ЭВМ, цвет кодируется числом. В зависимости от количества допустимых цветов, число двоичных разрядов на один пиксель будет различным. Так, для черно-белой картинки закодировать цвет точки можно одним битом: 0 - черный, 1 - белый. Такое изображение называют монохромным (monochrome).

Для случая 16 цветов требуется уже по 4 разряда на каждую точку, а для 256 цветов - 8, т.е. 1 байт.

Если для кодировки отвести четыре бита, то можно закодировать 24=16 различных цветов, отвечающих комбинациям бит от 0000 до 1111. Если отвести 8 бит - то такой рисунок может содержать 28=256 различных цветов (от 00000000 до 11111111), 16 бит - 216=65 536 различных цветов (так называемый High Color). И, наконец, если отвести 24 бита, то потенциально рисунок может содержать 224=16 777 216 различных цветов и оттенков - вполне достаточно даже для самого взыскательного художника! В последнем случае кодировка называется 24-bit True Color. Следует обратить внимание на слово "потенциально": даже если в файле и отводится 24 бита на каждый пиксел, это еще не означает, что вы действительно сможете насладиться такой богатой палитрой - ведь технические возможности мониторов ограничены.

Для того, чтобы наглядно представить себе, как хранится в памяти ЭВМ простейшее изображение, рассмотрим для примера белый квадратик на черном фоне размером 4х4. В черно-белом режиме это будет выглядеть наиболее компактно (сначала для наглядности приведен двоичный, а затем шестнадцатеричный вид):

1111 F

1001 9

1001 9

1111 F

В режиме 16-цветной графики это же самое изображение потребует памяти в 4 раза больше:

1111 1111 1111 1111 F F F F

1111 0000 0000 1111 F 0 0 F

1111 0000 0000 1111 F 0 0 F

1111 1111 1111 1111 F F F F

Наконец, при 256 цветах на каждую точку требуется уже по байту и наш квадратик разрастется еще вдвое:

11111111 11111111 11111111 11111111 FF FF FF FF

11111111 00000000 00000000 11111111 FF 00 00 FF

11111111 00000000 00000000 11111111 FF 00 00 FF

11111111 11111111 11111111 11111111 FF FF FF FF

Обратите внимание на то, что белый цвет, как самый яркий, обычно имеет максимально возможный номер. Поэтому для черно-белого режима он равен 1, для 16-цветного - 15, а для 256 цветов - 255.

Осталось обсудить вопрос, как кодируются промежуточные цвета. Например, вполне естественно со стороны читателя спросить: какой номер имеет, например, красный цвет? К сожалению, методы кодирования цвета даже для одной и той же ЭВМ могут довольно существенно различаться. Причем не только в зависимости от конструкции дисплея, но даже от графического режима, в котором тот в данный момент работает! Более того, соответствие между номерами цветов и их представлением на экране можно переопределять по усмотрению пользователя (это называется изменением палитры). Поэтому давайте ограничимся в качестве примера стандартным 16-цветным набором для наиболее распространенного компьютера IBM PC:

0 - черный 8 - темно-серый

1 - синий 9 - ярко-синий

2 - зеленый A - ярко-зеленый

3 - голубой B - ярко-голубой

4 - красный C - ярко-красный

5 - розовый D - ярко-розовый

6 - коричневый E - ярко-желтый

7 - серый F - ярко-белый

Таким образом, графическая информация, также как числовая и текстовая, в конечном счете заносится в память в виде двоичных чисел.

Цвет аддитивный и субтрактивный

Аддитивный цвет получается при соединении света разных цветов. В этой схеме отсутствие всех цветов представляет собой чёрный цвет, а присутствие всех цветов - белый. Схема аддитивных цветов работает с излучаемым светом, например, монитор компьютера. В схеме субтрактивных цветов происходит обратный процесс. Здесь получается какой-либо цвет при вычитании других цветов из общего луча света. В этой схеме белый цвет появляется в результате отсутствия всех цветов, тогда как их присутствие даёт чёрный цвет. Схема субтрактивных цветов работает с отражённым светом.

Индексированный цвет, работа с палитрой

Индексированные палитры цветов - это наборы цветов, из которых можно выбрать необходимый цвет. Преимуществом ограниченных палитр является то, они что занимают гораздо меньше памяти, чем полные системы RGB и CMYK. Компьютер создаёт палитру цветов и присваивает каждому цвету номер от 1 до 256. Затем при сохранении цвета отдельного пикселя или объекта компьютер просто запоминает номер, который имел этот цвет в палитре. Для запоминания числа от 1 до 256 компьютеру необходимо всего 8 бит. Для сравнения полный цвет в системе RGB занимает 24 бита, а в системе CMYK - 32.

Таблица «безопасных» цветов

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 255.255.204 | 255.255.153 | 255.255.102 | 255.255.51 | 255.255.0 | 204.204.0 | | FFFFCC | FFFF99 | FFFF66 | FFFF33 | FFFF00 | CCCC00 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 255.204.102 | 255.204.0 | 255.204.51 | 204.153.0 | 204.153.51 | 153.102.0 | | FFCC66 | FFCC00 | FFCC33 | CC9900 | CC9933 | 96600 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 255.153.0 | 255.153.51 | 204.153.102 | 204.102.0 | 153.102.51 | 102.51.0 | | FF9900 | FF9933 | CC9966 | CC6600 | 996633 | 663300 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 255.204.153 | 255.153.102 | 255.102.0 | 204.102.51 | 153.51.0 | 102.0.0 | | FFCC99 | FF9966 | FF6600 | CC6633 | 993300 | 660000 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 255.102.51 | 204.51.0 | 255.51.0 | 255.0.0 | 204.0.0 | 153.0.0 | | FF6633 | CC3300 | FF3300 | FF0000 | CC0000 | 990000 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 255.204.204 | 255.153.153 | 255.102.102 | 255.51.51 | 255.0.51 | 204.0.51 | | FFCCCC | FF9999 | FF6666 | FF3333 | FF0033 | CC0033 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 204.153.153 | 204.102.102 | 204.51.51 | 153.51.51 | 153.0.51 | 51.0.0 | | CC9999 | CC6666 | CC3333 | 993333 | 990033 | 330000 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 255.102.153 | 255.51.102 | 255.0.102 | 204.51.102 | 153.102.102 | 102.51.51 | | FF6699 | FF3366 | FF0066 | CC3366 | 996666 | 663333 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 255.153.204 | 255.51.153 | 255.0.153 | 204.0.102 | 153.51.102 | 102.0.51 | | FF99CC | FF3399 | FF0099 | CC0066 | 993366 | 660033 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 255.102.204 | 255.0.204 | 255.51.204 | 204.102.153 | 204.0.153 | 153.0.102 | | FF66CC | FF00CC | FF33CC | CC6699 | CC0099 | 990066 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 255.204.255 | 255.153.255 | 255.102.255 | 255.51.255 | 255.0.255 | 204.51.153 | | FFCCFF | FF99FF | FF66FF | FF33FF | FF00FF | CC3399 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 204.153.204 | 204.102.204 | 204.0.204 | 204.51.204 | 153.0.153 | 153.51.153 | | CC99CC | CC66CC | CC00CC | CC33CC | 990099 | 993399 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 204.102.255 | 204.51.255 | 204.0.255 | 153.0.204 | 153.102.153 | 102.0.102 | | CC66FF | CC33FF | CC00FF | 9900CC | 996699 | 660066 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 204.153.255 | 153.51.204 | 153.51.255 | 153.0.255 | 102.0.153 | 102.51.102 | | CC99FF | 9933CC | 9933FF | 9900FF | 660099 | 663366 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 153.102.204 | 153.102.255 | 102.0.204 | 102.51.204 | 102.51.153 | 51.0.51 | | 9966CC | 9966FF | 6600CC | 6633CC | 663399 | 330033 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 204.204.255 | 153.153.255 | 102.51.255 | 102.0.255 | 51.0.153 | 51.0.102 | | CCCCFF | 9999FF | 6633FF | 6600FF | 330099 | 330066 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 153.153.204 | 102.102.255 | 102.102.204 | 102.102.153 | 51.51.153 | 51.51.102 | | 9999CC | 6666FF | 6666CC | 666699 | 333399 | 333366 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 51.51.255 | 51.0.255 | 51.0.204 | 51.51.204 | 0.0.153 | 0.0.102 | | 3333FF | 3300FF | 3300CC | 3333CC | 000099 | 000066 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 102.153.255 | 51.102.255 | 0.0.255 | 0.0.204 | 0.51.204 | 0.0.51 | | 6699FF | 3366FF | 0000FF | 0000CC | 0033CC | 000033 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 0.102.255 | 0.102.204 | 51.102.204 | 0.51.255 | 0.51.153 | 0.51.102 | | 0066FF | 0066CC | 3366CC | 0033FF | 003399 | 003366 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 153.204.255 | 51.153.255 | 0.153.255 | 102.153.204 | 51.102.153 | 0.102.153 | | 99CCFF | 3399FF | 0099FF | 6699CC | 336699 | 006699 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 102.204.255 | 51.204.255 | 0.204.255 | 51.153.204 | 0.153.204 | 0.51.51 | | 66CCFF | 33CCFF | 00CCFF | 3399CC | 0099CC | 003333 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 153.204.204 | 102.204.204 | 51.153.153 | 102.153.153 | 0.102.102 | 51.102.102 | | 99CCCC | 66CCCC | 339999 | 669999 | 006666 | 336666 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 204.255.255 | 153.255.255 | 102.255.255 | 51.255.255 | 0.255.255 | 0.204.204 | | CCFFFF | 99FFFF | 66FFFF | 33FFFF | 00FFFF | 00CCCC | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 153.255.204 | 102.255.204 | 51.255.204 | 0.255.204 | 51.204.204 | 0.153.153 | | 99FFCC | 66FFCC | 33FFCC | 00FFCC | 33CCCC | 009999 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 102.204.153 | 51.204.153 | 0.204.153 | 51.153.102 | 0.153.102 | 0.102.51 | | 66CC99 | 33CC99 | 00CC99 | 339966 | 009966 | 006633 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 102.255.153 | 51.255.153 | 0.255.153 | 51.204.102 | 0.204.102 | 0.153.51 | | 66FF99 | 33FF99 | 00FF99 | 33CC66 | 00CC66 | 009933 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 153.255.153 | 102.255.102 | 51.255.102 | 0.255.102 | 51.153.51 | 0.102.0 | | 99FF99 | 66FF66 | 33FF66 | 00FF66 | 339933 | 006600 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 204.255.204 | 153.204.153 | 102.204.102 | 102.153.102 | 51.102.51 | 0.51.0 | | CCFFCC | 99CC99 | 66CC66 | 669966 | 336633 | 003300 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 51.255.51 | 0.255.51 | 0.255.0 | 0.204.0 | 51.204.51 | 0.204.51 | | 33FF33 | 00FF33 | 00FF00 | 00CC00 | 33CC33 | 00CC33 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 102.255.0 | 102.255.51 | 51.255.0 | 51.204.0 | 51.153.0 | 0.153.0 | | 66FF00 | 66FF33 | 33FF00 | 33CC00 | 339900 | 009900 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 204.255.153 | 153.255.102 | 102.204.0 | 102.204.51 | 102.153.51 | 51.102.0 | | CCFF99 | 99FF66 | 66CC00 | 66CC33 | 669933 | 336600 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 153.255.0 | 153.255.51 | 153.204.102 | 153.204.0 | 153.204.51 | 102.153.0 | | 99FF00 | 99FF33 | 99CC66 | 99CC00 | 99CC33 | 669900 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 204.255.102 | 204.255.0 | 204.255.51 | 204.204.153 | 102.102.51 | 51.51.0 | | CCFF66 | CCFF00 | CCFF33 | CCCC99 | 666633 | 333300 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 204.204.102 | 204.204.51 | 153.153.51 | 153.153.102 | 153.153.0 | 102.102.0 | | CCCC66 | CCCC33 | 999966 | 999933 | 999900 | 666600 | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 255.255.255 | 204.204.204 | 153.153.153 | 102.102.102 | 51.51.51 | 0.0.0 | | FFFFFF | CCCCCC | 999999 | 666666 | 333333 | 000000 | |  |  |  |  |  |  | |

Источники

1. http://dvoika.net/education/Book1/contents.html
2. Из книги: Еремин Е.А. Как работает современный компьютер. - Пермь: изд-во ПРИПИТ

3. http://www.256.ru/lecture/lect-kgg0301.php (Тема 3. Система цветов.)

Модели цвета на ЭВМ

Как порой сложно описать цвет того неба, которое мы видим: у кого-то оно сине-голубое, у кого-то голубое с бирюзовым отливом, у кого-то серебристо-голубое и т.д. И было бы невозможно без способа точного описания цвета в стандартизированных цифровых выражениях.

Цветовые пространства, или цветовые модели являются средствами количественного описания цвета и различия между оттенками цвета.



Существует много цветовых моделей, но все они принадлежат к одному из трех типов:

- психологические (по восприятию);

- аддитивные (основанные на сложении);

- субтрактивные (основанные на вычитании).

При обработке изображений при подготовке к печати имеют дело с тремя цветовыми моделями: CIE Lab - психологическое цветовое пространство, RGB - аддитивное цветовое пространство и CMYK - субтрактивное цветовое пространство. А в компьютерной графике применяются три цветовые модели: RGB, CMYK и HSB.

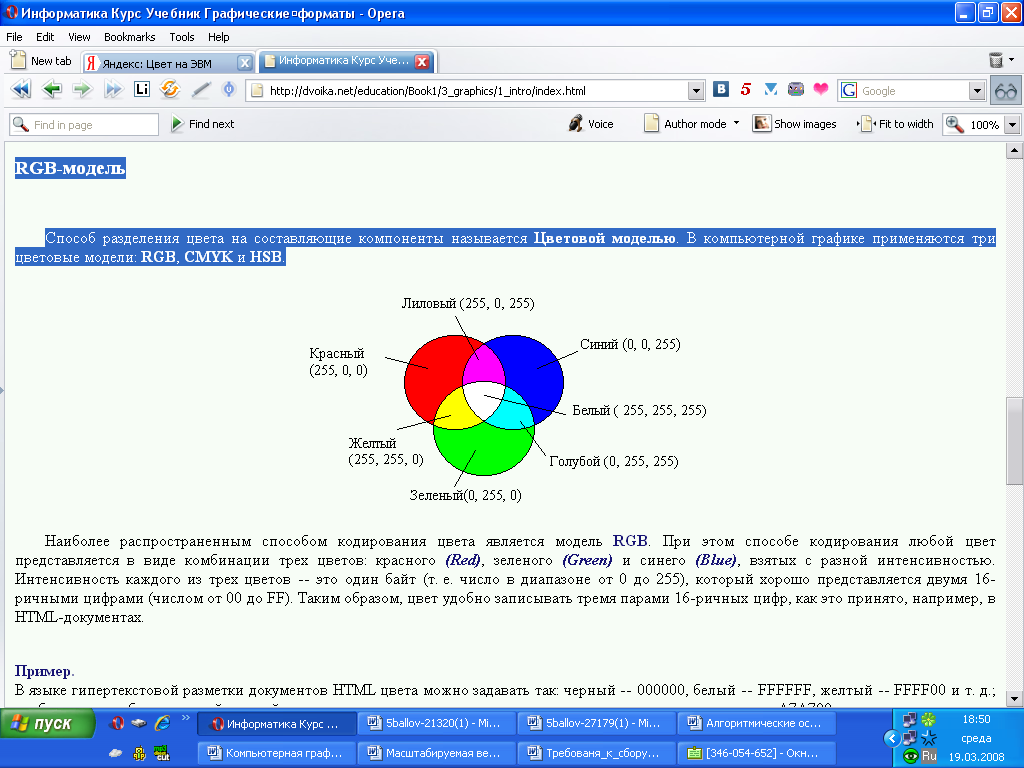
Любое преобразование цвета из одного пространства в другое влечет за собой потерю данных о цвете в изображении.

RGB-модель

Данная модель является "естественным языком" цвета для электронных устройств ввода изображения (мониторы, сканеры, цифровые камеры), в которых воспроизведение цвета основано на излучении или пропускании света, а не на его отражении от подложки при создании изображения.

Аддитивной она называется потому, что цвета в ней генерируются суммированием световых потоков. Сумма красного, зеленого и синего цветов максимальной одинаковой интенсивности дает белый цвет.

R - red (красный), G - green (зеленый), B - blue (голубой).



Наиболее распространенным способом кодирования цвета является модель RGB. При этом способе кодирования любой цвет представляется в виде комбинации трех цветов: красного (Red), зеленого (Green) и синего (Blue), взятых с разной интенсивностью. Интенсивность каждого из трех цветов -- это один байт (т. е. число в диапазоне от 0 до 255), который хорошо представляется двумя 16-ричными цифрами (числом от 00 до FF). Таким образом, цвет удобно записывать тремя парами 16-ричных цифр, как это принято, например, в HTML-документах.

Пример.

В языке гипертекстовой разметки документов HTML цвета можно задавать так: черный -- 000000, белый -- FFFFFF, желтый -- FFFF00 и т. д.; чтобы получить более темный желтый цвет, надо одинаково уменьшить интенсивности красного и зеленого -- A7A700.

Чем больше значение байта цветовой составляющей, тем ярче этот цвет. При наложении одной составляющей на другую яркость суммарного цвета также увеличивается.

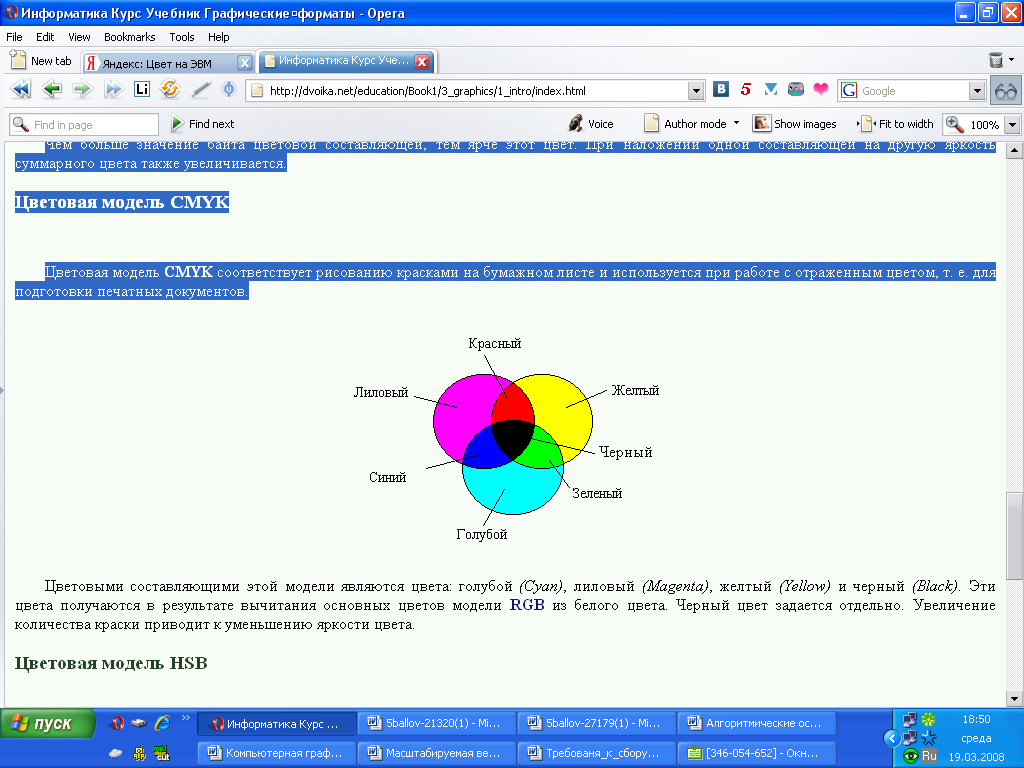
Цветовая модель CMYK

Цветовая модель CMYK соответствует рисованию красками на бумажном листе и используется при работе с отраженным цветом, т. е. для подготовки печатных документов.

В данной модели цвета при смешивании двух или более основных красок дополнительные цвета получаются посредством поглощения одних световых волн спектра белого цвета и отражения других. Так, голубая краска поглощает красный цвет и отражает зеленый и синий, а желтая поглощает синий цвет и отражает красный и зеленый.

В аддитивной модели RGB световые потоки суммируются, производя более яркие цвета, а в субтрактивной модели CMYK световые потоки вычитаются, генерируя более темные цвета. Если учесть светонепроницаемость бумаги, которая скорее отражает свет, чем пропускает его, то становится понятно, почему такие яркие цвета в изображении на мониторе становятся темными и унылыми в отпечатанном виде.

CMYK - cyan (голубой), magenta (пурпур), yellow (желтый), black (черный).



Цветовыми составляющими этой модели являются цвета: голубой (Cyan), лиловый (Magenta), желтый (Yellow) и черный (Black). Эти цвета получаются в результате вычитания основных цветов модели RGB из белого цвета. Черный цвет задается отдельно. Увеличение количества краски приводит к уменьшению яркости цвета.

Особые взаимоотношения двух цветовых моделей

Цветовые модели RGB и CMYK являются дополнительнительными друг к другу, по крайней мере, в первом приближении. Теоритечески так сказать. Но простого одноозначного соответствия между этими цветовыми простанствами не существует. А иначе бы зачем их надо было разделять на две модели?

Многие приятные для глаза цвета, которые видны на мониторе, не могут быть воспроизведены красками на оттиске.

Следует отметить, что при преобразовании цифрового изображения из модели RGB в CMYK отмечается сдвиг цвета к голубому. Точное значение сдвига зависит от используемых при печатании триад красок и типа бумаги, а также от технологии печати (листовая, рулонная, по сухому или по сырому, если речь идет об офсетной печати).

Таким образом, несмотря на то, что модели RGB и CMYK связаны друг с другом, однако их взаимные переходы (конвертирование) не происходят без потерь, так как цветовой охват у них разный. Снижение этих потерь требует выполнения сложных калибровок всех аппаратных средств и самих изображений.



Цветовая модель HSB

Системы цветов RGB и CMYK связаны с ограничениями, накладываемыми аппаратным обеспечением (монитор компьютера в случае RGB и типографские краски в случае CMYK).

Цветовая модель HSB наиболее удобна для человека, т. к. она хорошо согласуется с моделью восприятия цвета человеком. Компонентами модели HSB являются:

тон (Hue);

насыщенность (Saturation);

яркость цвета (Brightness).

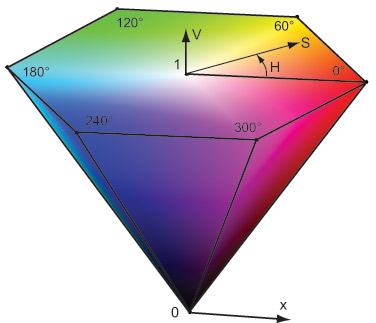
Тон - это конкретный оттенок цвета. Насыщенность характеризует его интенсивность или чистоту. Яркость же зависит от примеси черной краски, добавленной к данному цвету.

Значение цвета выбирается как вектор, выходящий из центра окружности. Точка в центре соответствует белому цвету, а точки по границе окружности - чистым цветам. Направление вектора определяет цветовой оттенок и задается в угловых градусах. Длина вектора определяет насыщенность цвета. Яркость цвета задают на отдельной оси.

#### Цветовая модель HSV

Цветовая модель HSV (от англ. Нue, Saturation, Value - тон, насыщенность, величина) является, в отличие от рассмотренных выше моделей, ориентированной на человека и его интуитивные представления о выборе цвета.

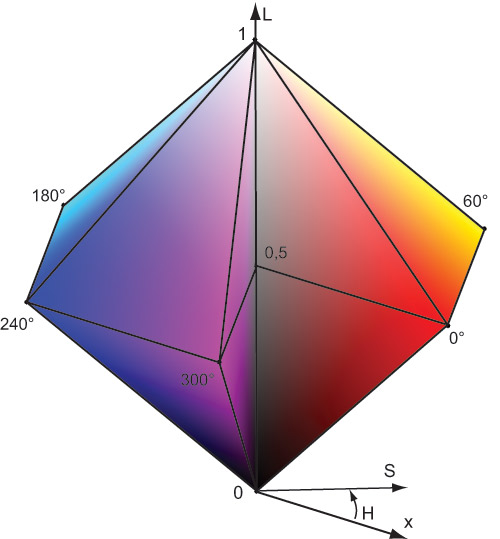
Рассмотрим цилиндрические координаты в трехмерном евклидовом пространстве, H - угол в горизонтальной плоскости от оси Ox, S - радиус в горизонтальной плоскости (расстояние до оси Oz), V - высота (по оси Oz). Все цветовое пространство представляет из себя перевернутую шестигранную пирамиду.



Концептуально, можно представить художника, который смешивает цвета. Вершины основания пирамиды соответствуют чистым основным цветам (красному, желтому, зеленому, цвету морской волны, синему и фиолетовому). При их смешивании друг с другом в разных пропорциях (в пространстве это будут линейные комбинации соответствующих векторов) точка, соответствующая цвету, перемещается по основанию пирамиды. Смешивая противоположные цвета (например, желтый и синий), можно получить белый. Добавляя к какому-либо чистому цвету черный, мы будем спускаться по пирамиде, получая различные оттенки, при этом диапазон S будет уменьшаться вплоть до нуля. На оси S = 0 (оттенки серого) значение H не определено.

#### Цветовая модель HLS

Цветовая модель HLS (от англ. Нue, Lightness, Saturation - тон, светлота, насыщенность) схожа с моделью HSV. Снова рассмотрим цилиндрические координаты в трехмерном евклидовом пространстве, H - угол в горизонтальной плоскость от оси Ox, S - радиус в горизонтальной плоскости (расстояние до оси Oz), L - высота (по оси Oz). Все цветовое пространство представляет из себя две соединенные основаниями шестигранные пирамиды.



Как видно эта модель получена из HSV вытягиванием вдоль вертикальной оси. Понятия H и S остались теми же, только по вертикальной оси теперь L вместо V. Концептуальное различие состоит в том, что в этой модели считается, что движение от чистых цветов (у которых L = 0,5, S = 1) как в направлении белого, так и черного (а не только черного, как в HSV) одинаково приводит к уменьшению информации в H (вплоть до того, что в вершинах H не определено (как впрочем, и на всей вертикальной оси S = 0)) и сужению диапазона S.

#### Цветовые модели Y\*\*

Существует несколько тесно связанных цветовых моделей, которые объединяет то, что в них используется явное разделение информации о яркости и цвете. Компонента Y соответствует одноименной компоненте в модели CIE XYZ и отвечает за яркость. Такие модели находят широкое применение в телевизионных стандартах, так как исторически необходима была совместимость с черно-белыми телевизорами, которые принимали только сигнал, соответствующий Y. Также они применяются в некоторых алгоритмах обработки и сжатия изображений и видео.

#### Цветовые модели YUV, YPbPr и YCbCr

Расcмотрим цветовую модель YUV. U и V отвечают за цветовую информацию и определяются через преобразование из RGB:

Y = 0,299R + 0,587G + 0,114B;

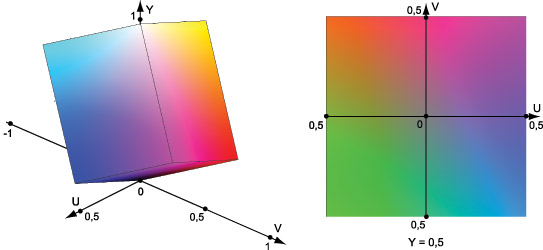
U = 0,492(B - Y)

= -0,147R - 0,289G + 0,436B;

V = 0,877(R - Y)

= 0,615R - 0,515G + 0,100B;

Модель YUV применяется в телевизионной системе PAL.



Цветовые модели YCbCr и YPbPr являются вариациями YUV с другими весами для U и V (им соответствуют Cb/Pb и Cr/Pr). YPbPr применяется для описания аналоговых сигналов (преимущественно в телевидении), а YCbCr - для цифровых. Для их определения используются два коэффициента: Kb и Kr. Тогда преобразование из RGB в YPbPr описывается так:

Переход от RGB к YPbPr



Выбор Kb и Kr зависит от того, какая RGB-модель используется (это в свою очередь зависит от воспроизводящего оборудования). Обычно берется, как и выше, Kb = 0, 114; Kr = 0, 299. В последнее время также используют Kb = 0, 0722; Kr = 0, 2126, что лучше отражает характеристики современных устройств отображения.

Из приведенных выше формул следует что при имеем следующие диапазоны ; . Для цифрового представления эти формулы видоизменяют для получения только положительных дискретных коэффициентов в диапазонах



Переход от RGB к YCbCr



В телевидении обычно берут minY = 16, maxY = 235, minC = 16, maxC = 240. В стандарте сжатия изображений JPEG используется полный 8-битный диапазон: minY = 0, maxY = 255, minC = 0, maxC = 255.

#### Цветовая модель YIQ

Цветовая модель YIQ применялась в телевизионной системе NTSC (I - от англ. in-phase, Q - от англ. quadrature; происходят от особенностей систем декодирования). Она тесно связана с моделью YUV, так как переход от YUV к YIQ является поворотом в плоскости UV = IQ на .



Y = 0,299R + 0,587G + 0,114B

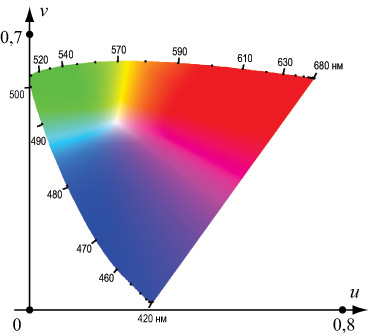
I = 0,735(R - Y) - 0,268(B - Y) = 0,596R - 0,274G + 0,321BQ = 0,478(R - Y) + 0,413(B - Y) = 0,211R - 0, 523G + 0,311B

Обратные преобразования для всех моделей получаются в результате применения обратной матрицы преобразования.

#### Цветовые модели CIE L\*u\*v\* и CIE L\*a\*b\*

У модели CIE XYZ все же есть существенный недостаток - неоднородность восприятия изменения цвета относительно расстояния на диаграмме цветности. В идеале хотелось бы, чтобы одинаковые расстояния между точками, соответствующими цветам на диаграмме цветности, соответствовали приблизительно одинаковому восприятию человеком отличий между этими парами цветов. Именно для этих целей CIE в 1976 году предложила модель L\*u\*v\*. L\* (от англ. Lightness) в этой модели соответствует яркости, скорректированной с учетом особенностей человеческого восприятия. Получающаяся диаграмма цветности представлена на рисунке. Эта модель рекомендуется для представления света от источников.

Также CIE в 1976 году с той же целью предложила и другую похожую модель L\*a\*b\*, которая получила несколько более широкое распространение. Эта модель рекомендуется для представления отраженного света.



Точность воспроизведения цвета

Цель полиграфического репродуцирования состоит в наиболее точном воспроизведении изображения оригинала на оттиске. К сожалению, необходимо отметить, что идентичное воспроизведение полутоновых, в особенности цветных, изображений практически недостижимо. Ниже приведены одни из главных причин этого:

- оттиск изготавливается на иной подложке, нежели оригинал (когда появляются различия в белизне, глянце, гладкости поверхности подложки между оригиналом и оттиском);

- оттиск изготавливается с использованием различных видов печатных красок, которые определяются и способом печати;

- оттиск полутонового изображения всегда имеет растровую структуру, в то время как оригинал может иметь непрерывную структуру полутона и контура.

Таким образом, если у вас в руках есть что-то, что вы хотите сделать точь-в-точь, то первоначально надо определить то, на каком материале оно сделано, каким способом (это минимум, что надо знать). Если вы затрудняетесь ответить на эти вопросы, обращайтесь к профессионалам-полиграфистам, иначе не избежать досадных ошибок, перепечаток и т.д.

Источники

1. http://www.intuit.ru/department/graphics/rastrgraph/1/ (Алгоритмические основы растровой графики информация [+] Авторы: Д.В. Иванов, А.А. Хропов, Е.П. Кузьмин, А.С. Карпов, В.С. Лемпицкий. 1. Лекция: Основные понятия. Представление цвета в машинной графике).

2. http://www.bestreferat.ru/referat-74412.html (Реферат: Цвет, цветовые модели, цветовые пространства).

3. http://dvoika.net/education/Book1/contents.html

Виды графики на ЭВМ и особенности её применения в ИС

Основные понятия. Представление цвета в машинной графике. Растровая и векторная графика. Понятие растра

Для представления графической информации на двумерной плоскости (например, экране монитора, странице книги и т.п.) в вычислительной технике применяются два основных подхода: растровый и векторный.

При векторном подходе графическая информация описывается как совокупность неких абстрактных геометрических объектов, таких как прямые, отрезки, кривые, прямоугольники и т.п.

Растровая графика же оперирует изображениями в виде растров. Неформально можно сказать, что растр - это описание изображения на плоскости путем разбиения всей плоскости или ее части на одинаковые квадраты и присвоение каждому квадрату своего цветового (или иного, например, прозрачности, для последующего наложения изображений друг на друга) атрибута. Если таких квадратов имеется конечное число, то получается, что непрерывная цветовая функция изображения приближенно представлена конечной совокупностью значений атрибутов. Иногда понятие растра определяют более широко: как разбиение плоскости (или ее участка) на равные элементы (т.е."замощение"), например шестиугольниками (гексагональный растр). Далее в этой книге расширенное толкование использоваться не будет.

С другой стороны, растр можно рассматривать как кусочно-постоянную аппроксимацию изображения, заданного как цветовая функция на плоскости. Такая точка зрения позволяет применять математический аппарат теории аппроксимации для работы с растровыми изображениями, о чем подробнее будет рассказано далее.

Формально, введем следующие определения:

Растр (англ. raster) - отображение вида



где , ,



обозначает множество всех подмножеств ,



C - множество значений атрибутов (как правило, цвет).

f(i, j) - элемент растра, называемый пикселем (англ. pixel (от picture element)), в русскоязычной литературе иногда также переводится как пиксел);

f(i, j) = (A(i, j),C(i, j)),

где

- область пикселя,



- атрибут пикселя (как правило, цвет).



Чаще всего мы будем пользоваться следующими двумя видами атрибутов:

C(i, j) = I(i, j) - интенсивность (или яркость) пикселя;

C(i, j) = {R(i, j),G(i, j),B(i, j)} - цветовые атрибуты в цветовой модели RGB (см. раздел 1.2).

Также иногда будут употребляться матричные обозначения:

Mij = (Aij ,Cij)

Aij может определяться двояко, в зависимости от того, с какой моделью мы хотим работать:

Aij := (i, j) - одна точка.

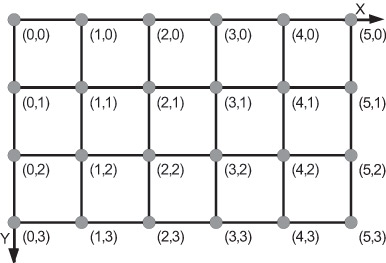
- квадрат.



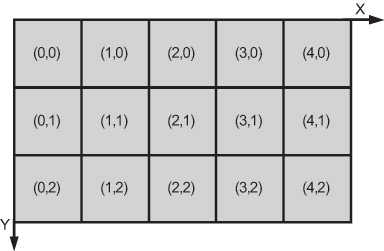
Пример

На реальных графических устройствах физически пиксели могут быть прямоугольниками, что иногда порождает дополнительные трудности.

В реальности, как правило, X и Y - ограниченные наборы неотрицательных целых чисел; такой растр называется прямоугольным. Для него применимо понятие Аспектовое отношение (англ. aspect ratio) - отношение ширины к высоте растра (|X|/|Y|). Чаще всего такое понятие употребляется в связи с физическими растрами (дисплеями, ПЗС-матрицами фотоаппаратов и т.д.) и записывается в виде простой дроби с ":", например "4:3".



Модель растра первого типа.



Модель растра второго типа.

Бесконечные растры (когда X и Y неограниченны) бывают удобны для описания алгоритмов, позволяя избежать особых ситуаций. Впрочем, самой сутью некоторых алгоритмов является как раз работа с граничными случаями.

Растровое представление является естественным в тех случаях, когда нам не известна дополнительная информация об изображаемых объектах (например, цифровым фотоаппаратом можно снимать изображения произвольного содержания). В случае же векторного описания примитивами являются более сложные объекты (линии и области, ограниченные линиями), что предполагает априорные знания о структуре изображения. В последнее время проявляется ярко выраженная тенденция к преобладанию устройств ввода-вывода двумерной графической информации, основанных на растровом принципе как более универсальном. Возникающая при выводе задача отображения геометрических объектов, заданных их математическим описанием (например, координатами концевых точек и цветом для отрезка), на растре, называемая растеризацией, рассмотрена в последующих разделах.

При построении алгоритмов, работающих с изображениями, можно также пользоваться информацией как непосредственно атрибутов пикселей, так и работать с примитивами более высокого порядка. В данной книге в основном рассматриваются алгоритмы первого типа, про которые говорят, что они работают в пространстве изображения (англ. image space), тогда как вторые работают в объектном пространстве (англ. object space) (эти термины чаще употребляются в трехмерной графике).

Устройства отображения растровой графики рассматриваются в следующей лекции.

Масштабируемая векторная графика (Scalable Vector Graphics)

SVG (Scalable Vector Graphics, в переводе масштабируемая векторная графика) - это язык для описания двухмерной графики в формате XML. Можно сказать, что SVG – это HTML для графики. Он включает в себя несколько типов различных объектов:

* векторную графику (линии, окружности, кривые)
* текст (с форматированием, заданием стилей и эффектов)
* растровые изображения (внедряемые в SVG–документ).

Графические возможности языка чрезвычайно велики и сравнимы с возможностями пакетов Corel Draw и Adobe Illustrator:

* произвольные заливки, штриховки, градиенты, заливки заданным растром, а также произвольным фрагментом векторного изображения,
* разнообразные стили линий - линии произвольной толщины, штриховые, с различными стилями угловых и конечных точек,
* возможность использование кривых Безье (кубических и квадратичных),
* поддержка прозрачности всего изображения или его части, с заданной степенью,
* возможность применения фильтров, типичных для растровой графики – размытие, попиксельное смешивание, наложение,
* поддержка Unicode – возможность отображения в одном SVG документе текстов на различных языках и различными шрифтами,
* расположение текстовых строк вдоль произвольных кривых,
* встроенные развитые средства анимации, позволяющие без программирования в режиме реального времени менять практически все атрибуты изображения или его фрагмента,
* встроенные средства для создания интерактивности, а также поддержка JavaScript и DOM,
* поддержка каскадных таблиц стилей (CSS),
* поддержка антиалиасинга.

Название "масштабируемая векторная графика" отражает следующие превосходства языка:

* SVG–документы могут быть отображены на самых различных устройствах – от экранов мобильных телефонов и карманных компьютеров до мониторов PC и ноутбуков.
* SVG–документы представляются в векторном формате, следовательно, их можно масштабировать без потери качества по сравнению с растровыми изображениями
* SVG–графика может быть создана "на лету" при помощи любого языка программирования (Javascript, Java или C#) и возможно управление видом этой графики в режиме работы приложения.

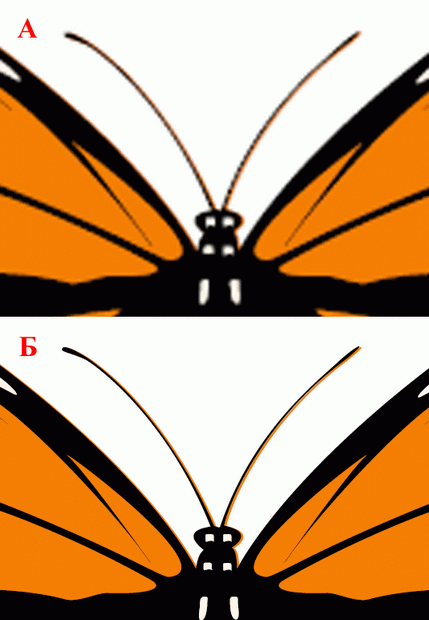


Рис. Увеличенный фрагмент растрового (А) и векторного SVG (Б) изображений.

У растрового изображения заметна "зазубренная" структура.

Объектная модель SVG документа основана на принятом консорциумом W3C в 2000 году стандарте DOM (Document Object Model) Level-2. Это дает возможность использования стандартных языков для начального построения и произвольного манипулирования любыми элементами изображения. Составной частью стандарта SVG является модель событий, позволяющая привязать произвольный сценарий к любому фрагменту SVG документа и достаточно легко реализовывать интерактивные графические изображения.

SVG разрабатывается консорциумом W3C при поддержке таких крупных компаний, как Microsoft, Adobe, Corel, IBM, Hewlett-Packard, Sun Microsystems, Canon, Kodak и многих других (всего более двадцати). От их совместных усилий зависят стандартизация многих деталей языка, а также скорость его создания.

Стандарт SVG 1.0 был принят в качестве спецификации консорциумом в сентябре 2001 г. Стандарт SVG 1.1 и его версии SVG mobile profiles (SVG Basic and SVG Tiny) были приняты консорциумом в качестве рекомендации в январе 2003 г. Ведутся работы по созданию стандарта SVG 1., который сейчас имеет статус черновика. Эта версия языка будет содержать ряд новых свойств: встроенные атрибуты фигурного форматирования текста, поддержка аудио и видео, а также более полная поддержка DOM.

На момент написания этого курса рабочей версией является SVG 1.1. Поэтому именно его мы и будем рассматривать.

|  |  |  |
| --- | --- | --- |
| Сравнение растровой и векторной графики | | |
| Критерий сравнения | Растровая графика | Векторная графика |
| Способ представления изображения | Растровое изображение строится из множества пикселей. | Векторное изображение описывается в виде последовательности команд. |
| Представление объектов реального мира | Растровые рисунки эффективно используются для представления реальных образов. | Векторная графика не позволяет получать изображения фотографического качества. |
| Качество редактирования изображения | При масштабировании и вращении растровых картинок возникают искажения. | Векторные изображения могут быть легко преобразованы без потери качества. |
| Особенности печати изображения | Растровые рисунки могут быть легко напечатаны на принтерах. | Векторные рисунки иногда не печатаются или выглядят на бумаге не так, как хотелось бы. |

Классификация изображений и преобразования

Классификация

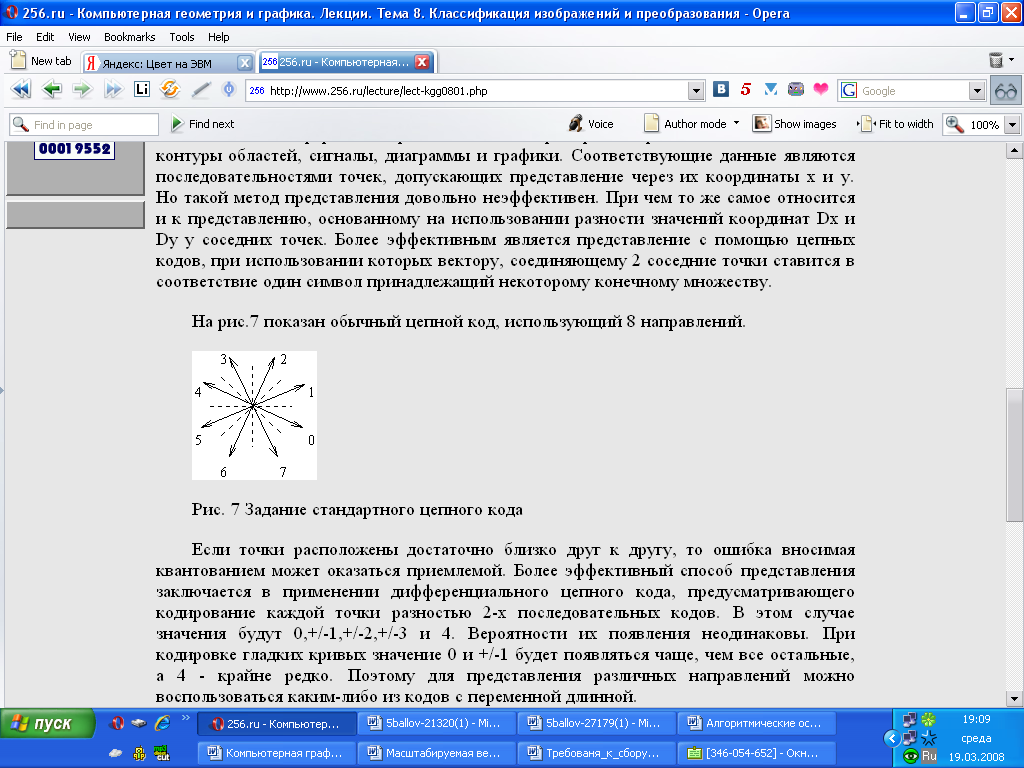
Говоря об обработке изображений с помощью ЭВМ целесообразно выделить 4 класса изображений. Эта классификация связана не столько с природой зрительного восприятия изображений, сколько с подходом к их представлению и обработке.

Класс 1 - тоновые и цветные изображения. В этот класс входят изображения, имеющие вид обычных телевизионных изображений. Они обеспечивают довольно точное воспроизведение реальности и представляются матрицами с целочисленными элементами. Для обозначения этих элементов используют термины "элементы изображения", "пиксел", "пэл". В большинстве прикладных задач эти матрицы имеют очень большие размеры объектов - 512х512 элементов, являющиеся наиболее общепринятыми. В связи с этим представление изображений не всегда хранятся в памяти в виде обычных матриц и часто используются более изощренные разновидности структур данных. Цветные изображения могут представляется либо при помощи 3-х матриц (для красного, синего и зеленого цветов в отдельности), либо с помощью другой матрицы таким образом, что отдельные биты любого элемента представляют различные цвета, так как человеческий глаз порой не в состоянии различать уровни освещенности, различающиеся друг от друга менее чем на 1%, то для представления цветного изображения достаточно затрачивать по 1-му байту на цвет одного пикселя. Приемлемых результатов удается добиться, используя по три бита для передачи любого из двух цветов и два бита для передачи третьего.

Класс 2 - 2-х уровневые или представляемые в нескольких цветах изображения. Изображение книжной страницы - это типичный пример 2-го класса, т.е. чёрно-белое изображение. Подобное изображение можно представлять матрицами, затрачивая по 1 биту на элемент, а также в виде "карт", так как на этих изображениях имеются хорошо различимые области одного цвета. Поэтому такие изображения объединены в один класс. Одна из проблем, возникающих в связи с использованием 1-го бита для представления любого пиксела, заключается в отсутствие стандартного для различных типов ЭВМ и устройств визуального отображения способов объединения битов в байт, и байт в слово. То есть, крайний слева пиксел может представляться, как наименее, так и наиболее значимыми битами байта. Поэтому пользователь должен думать о выборе наиболее подходящего способа представления в применяемом им устройстве.

Класс 3 - непрерывные кривые и линии. Примеры изображений 3-го класса - это контуры областей, сигналы, диаграммы и графики. Соответствующие данные являются последовательностями точек, допускающих представление через их координаты х и у. Но такой метод представления довольно неэффективен. При чем то же самое относится и к представлению, основанному на использовании разности значений координат Dх и Dу у соседних точек. Более эффективным является представление с помощью цепных кодов, при использовании которых вектору, соединяющему 2 соседние точки ставится в соответствие один символ принадлежащий некоторому конечному множеству.

На рисунке показан обычный цепной код, использующий 8 направлений.



Задание стандартного цепного кода

Если точки расположены достаточно близко друг к другу, то ошибка вносимая квантованием может оказаться приемлемой. Более эффективный способ представления заключается в применении дифференциального цепного кода, предусматривающего кодирование каждой точки разностью 2-х последовательных кодов. В этом случае значения будут 0,+/-1,+/-2,+/-3 и 4. Вероятности их появления неодинаковы. При кодировке гладких кривых значение 0 и +/-1 будет появляться чаще, чем все остальные, а 4 - крайне редко. Поэтому для представления различных направлений можно воспользоваться каким-либо из кодов с переменной длинной.

При таком способе кодирования обычно затраты в среднем не превышают 2-х бит на точку.

Класс 4 - точки или многоугольники. Изображения класса 4 состоят из множеств отдельных точек, отстоящих друг от друга так далеко, что для их представления цепным кодом пользоваться нельзя. Вместо него следует использовать матрицу, содержащую их координаты х и у. Соответствующая аппаратура отображения позволяет соединить точки прямыми или простыми кривыми. В прикладных задачах машинной графики чаще всего используют изображения именно этого типа. Несмотря на то, что визуальное отображение может относится к классу 2 или к классу 1, его внутреннее изображение принадлежит классу 4. Во многих прикладных задачах используется одна из следующих форм представления:

1) Аппроксимация поверхностей многогранниками. В этом случае грани обычно треугольные. После проектирования изображение состоит из многоугольников.

2) Криволинейная аппроксимация поверхностей. На поверхности вычерчивается ряд кривых, описание которых потом используется для получения проекций, воспроизводимых в виде изображений 3 класса.

3) Аппроксимация участками поверхности высшего порядка. Этот способ аналогичен первому способу, за исключением того, что в качестве элементов, образующих поверхность объекта используется не плоские многоугольники, а участки поверхности высшего порядка.

Во всех случаях положение объекта определяется некоторым небольшим числом точек и поэтому изображение класса 4 предоставляют наибольший интерес для машинной графики.

Ввод изображения

Изображение, представленное в аналоговой форме, необходимо преобразовать в некоторую числовую матрицу, а затем можно приступать к его обработке на ЭВМ. Процесс такого преобразования называется дискретизацией и состоит из 2-х процессов: выборки и квантования. Первый процесс заключается в выборе на поле наблюдения начального множества точек, в любой выбранной точке измеряются характеристики изображения, которые потом используются на всех последующих этапах обработки изображения. Так как ЭВМ располагает ограниченной памятью, то результаты полученных измерений описывается конечным числом символов. Такая процедура называется процессом квантования. Характеризуя плотность размещения выборочных точек, часто говорят о пространственном разрешении, а, характеризуя точность представления результатов указанных измерений, говорят о тоновом или цветовом разрешении.

Во многих устройствах дискретизации изображений используется телевизионные камеры, так как они обеспечивают преобразование светового сигнала в электрический. К электрическому сигналу можно уже применять процессы выборки и квантования, используя для этого аналого-цифровой преобразователь (АЦП). Но возникает проблема - это очень большая скорость воспроизведения данных на выходе стандартных телевизионных камер. Системы телевизионного вещания передают 30 кадров в секунду и любой кадр содержит около 500 строк растра. Если вести обработку в реальном масштабе времени необязательно, то можно перестроить телекамеру так, чтобы она работала с меньшей скоростью или воспользоваться каким-либо устройством, осуществляющем сжатие полосы частот. В специализированных устройствах дискретизации изображений, просмотр изображения в соответствующем порядке осуществляется при помощи особой технологии управления световыми пучками. В барабанных сканирующих устройствах изображение закрепляется на вращающемся барабане, а световой пучок перемещается параллельно оси барабана. Устройства дискретизации этого типа обычно работают медленнее по сравнению с телевизионными камерами, но результаты получаются лучше.

Преобразования изображений

Преобразование изображения класса 1 в изображение класса 2 - это процесс сегментации, обеспечивает выделение областей приблизительно одинакового цвета и (или) яркости. Часто термин "сегментация" используется для обозначения процесса поиска однородностей в смысле некоторого более сложного свойства (типа текстур).

Преобразование изображений класса 2 в класс 3 заключается в построении или отслеживании контура. Это преобразование обеспечивает отображение заданной области в некоторую замкнутую кривую. Второе допустимое преобразование - прореживание, заключается в отображении области в некоторый граф, называемый остовом области.

Преобразование из класса 3 в класс 4, называемое сегментацией кривых, предназначено для отыскания критических точек конура. Если это многоугольники, то такими точками являются углы. Применяется при распознавании образов, для их реализации может потребоваться применение сложных математических методов.

Преобразование из класса 4 в класс 3 включает процессы интерполирования, обеспечивающего проведение гладкой кривой через некоторое множество точек и аппроксимация, обеспечивающая проведение гладкой кривой рядом с некоторым множеством точек.

Преобразование из класса 3 в класс 2. Если в качестве входного изображения задается контур, то часто возникает задача заполнения контура (задача штриховки). Причем если рассматривать штриховку, то яркость или цвет некоторой области не остаются одинаковыми, а изменяется в соответствии с определенными правилами. Если входным изображением служит остов, то для восстановления области необходимо использовать процедуру расширения.

Преобразование из класса 2 в класс 1. Если изображение, воспроизводимое на экране в нескольких цветах, то оно часто оказывается ущербным в эстетическом отношении, т.к. легко обнаруживаются контуры (границы) между объектами. Некоторого сглаживания изображения можно добиться с помощью фильтра нижних частот или подмешиванием низкочастотного шума.

Преобразования из классов с меньшим номером в класс с большим номером относится к сфере интересов распознавания образов. Обратные преобразования - это сфера интересов машинной графики. При обработке изображений используются и те и другие преобразования, а также преобразования, не выводящие изображения за пределы соответствующего класса. Таким образом, улучшение качества изображения является внутриклассовым изображением, а сжатие изображения часто оказывается преобразованием, переводящим его из класса 1 в класс 2.

Еще один важный класс задач - это преобразования, связывающие двухмерное изображение и трехмерные объекты. Термин "проектирование" используется для обозначения операций, при помощи которых трехмерный объект преобразовывается в двухмерное изображение. Часто двухмерное изображение преобразуется в одномерный массив. Для операции восстановления трехмерного объекта по его изображению, используется термин "обратное проектирование". Эти задачи используются в 2-х прикладных областях, например, в аксиальной поперечной томографии, или когда поперечное сечение трехмерного объекта восстанавливается по набору рентгеновских проекций. Для обозначения процедур, обеспечивающих решение этой задачи используется термин "алгоритмы воспроизведения". В машинной графике часто требуется воспроизвести некоторую проекцию трехмерного представления пространственного объекта.

Основные области применения компьютерной графики

Научная графика

Первые компьютеры использовались лишь для решения научных и производственных задач. Чтобы лучше понять полученные результаты, производили их графическую обработку, строили графики, диаграммы, чертежи рассчитанных конструкций. Первые графики на машине получали в режиме символьной печати. Затем появились специальные устройства - графопостроители (плоттеры) для вычерчивания чертежей и графиков чернильным пером на бумаге. Современная научная компьютерная графика дает возможность проводить вычислительные эксперименты с наглядным представлением их результатов.

Деловая графика

Область компьютерной графики, предназначенная для наглядного представления различных показателей работы учреждений. Плановые показатели, отчетная документация, статистические сводки - вот объекты, для которых с помощью деловой графики создаются иллюстративные материалы. Программные средства деловой графики включаются в состав электронных таблиц.

Конструкторская графика

Используется в работе инженеров-конструкторов, архитекторов, изобретателей новой техники. Этот вид компьютерной графики является обязательным элементом САПР (систем автоматизации проектирования). Средствами конструкторской графики можно получать как плоские изображения (проекции, сечения), так и пространственные трехмерные изображения.

Художественная и рекламная графика

Стала популярной во многом благодаря телевидению. С помощью компьютера создаются рекламные ролики, мультфильмы, компьютерные игры, видеоуроки, видеопрезентации. Графические пакеты для этих целей требуют больших ресурсов компьютера по быстродействию и памяти. Отличительной особенностью этих графических пакетов является возможность создания реалистических изображений и "движущихся картинок". Получение рисунков трехмерных объектов, их повороты, приближения, удаления, деформации связано с большим объемом вычислений. Передача освещенности объекта в зависимости от положения источника света, от расположения теней, от фактуры поверхности, требует расчетов, учитывающих законы оптики.

Компьютерная анимация

Получение движущихся изображений на экране дисплее. Художник создает на экране рисунке начального и конечного положения движущихся объектов, все промежуточные состояния рассчитывает и изображает компьютер, выполняя расчеты, опирающиеся на математическое описание данного вида движения. Полученные рисунки, выводимые последовательно на экран с определенной частотой, создают иллюзию движения.

Мультимедиа

Объединение высококачественного изображения на экране компьютера со звуковым сопровождением. Наибольшее распространение системы мультимедиа получили в области обучения, рекламы, развлечений.

Источники

1. http://www.256.ru/lecture/lect-kgg0301.php (Тема 3. Система цветов.)

2. http://www.intuit.ru/department/graphics/rastrgraph/ (Алгоритмические основы растровой графики Авторы: Д.В. Иванов, А.А. Хропов, Е.П. Кузьмин, А.С. Карпов, В.С. Лемпицкий)

3. http://www.intuit.ru/department/graphics/svg/ (Алгоритмические основы растровой графики Авторы: Д.В. Иванов, А.А. Хропов, Е.П. Кузьмин, А.С. Карпов, В.С. Лемпицкий)

Особенности применения графики в Интернет

Создание графики

Рисунки стали неотъемлемой частью HTML-документов, поэтому просто невозможно не рассмотреть вопросы применения графики в гипертексте. Тема, с точки зрения дизайна, поистине необъятная, но, к счастью, с практической стороны дело обстоит намного проще. Так же, как и в случае с гипертекстовыми редакторами, в этой области мы сталкиваемся с большим количеством программных средств, призванных облегчить жизнь (а точнее процесс рисования) авторам Web-страниц. Графика для Интернета несколько отличается от обычной, и эти отличия будут рассмотрены в данной главе.

Форматы графических файлов

Создавать рисунки можно самыми разными способами, используя любые (общепринятые и экзотические) графические пакеты и форматы файлов. Но в конечном итоге потребуется преобразовать результаты своей работы в один из стандартных для Интернета форматов. Сделать это очень легко: надо открыть ваш рисунок в графическом редакторе и выполнить команду. Сохранить как (Save As), выбрав для нового файла формат GIF, JPG или PNG. Из числа наиболее популярных программ для обработки графики хотелось бы упомянуть PhotoShop 5.x фирмы Adobe или Microsoft Photo Editor. Кроме того, для иллюстрации практической работы я выбрал интересный, на мой взгляд, пакет Gif Construction Set for Windows 95, созданный фирмой Alchemy Mindworks. Наверное, не у каждого читателя этой книги будет возможность установить на своем компьютере мощный пакет для рисования и анимации, а также потратить время на его изучение, поэтому в приведенных примерах я старался использовать простые графические редакторы, например, MS Paint.

Самым распространенным форматом графических файлов для HTML-документов является GIF (Graphic Interchange Format). Для кодирования цвета в нем используется 8 бит, то есть допускается 256 различных цветов или столько же градаций серого. Наборы цветов (палитры) могут быть различными. Один GIF-файл может содержать несколько изображений, позволяющих создавать движущиеся или изменяющиеся образы. В этом случае требуется согласование палитр различных изображений, составляющих один файл. Похожая проблема возникает, если монитор работает в режиме 256 цветов, а на экране одновременно воспроизводятся несколько изображений с разными палитрами. Очевидно, что для части картинок качество цветопередачи будет ухудшено.

Само по себе 8-битовое кодирование цвета предполагает, что размер графического файла должен быть относительно небольшим. Но, кроме того, для уменьшения размера файла используется еще и сжатие изображения. Существует две разновидности GIF-файлов: сжатые и обычные, в которых сжатие отсутствует. Компактность файла и дополнительные преимущества стали причиной того, что данный формат прочно утвердился в качестве стандарта де-факто для Интернета.

Дополнительное преимущество заключаются в том, что хранение информации в файле может быть организовано таким образом, чтобы при выводе рисунка происходило чередование строк. То есть вначале будут выводиться строки с номерами 1, 5, 9 и т. д., затем с номерами 2, 6,10 и т. д., и так до тех пор, пока весь рисунок не будет отображен. Для наблюдателя такой рисунок вначале кажется нечетким, а затем четкость изображения увеличивается. При передаче данных по сети это особенно выгодно, так как не требует полной передачи файла и позволяет увидеть изображение сразу же, хотя и в несколько размытом виде.

Другое преимущество GIF — возможность сделать часть изображения прозрачной. Прозрачным может стать только один цвет. Так, создание прозрачного фона позволяет более естественно вписать рисунок в документ и избежать появления прямоугольника, обозначающего границы изображения.

Для формата GIF разработано несколько спецификаций. В соответствии со спецификацией GIF89a графический файл может состоять из нескольких блоков.

• Блок заголовка HEADER содержит информацию о размере экрана и палитре.

• Блок текста PLAIN TEXT позволяет добавлять символьные данные к рисунку.

• Блок изображения IMAGE содержит одну картинку, импорт которой может осуществляться не только в формате GIF. При создании движущихся изображений в один файл включается несколько таких блоков.

• Блок управления CONTROL используется для размещения флагов прозрачности

и ожидания, а также для определения временной задержки при выводе нового блока изображения.

• Блок приложения APPLICATION предназначен для хранения служебной информации.

• Блок комментария COMMENT используется для размещения произвольной информации. Эти данные не выводятся на экран во время воспроизведения GIF-изображения.

• Блок цикла LOOP необходим для многократного воспроизведения движущейся картинки. В этом блоке задается число повторений при показе «ролика».

Вторым подходящим для Интернета графическим форматом является JPEG (JPG), названный так в честь своего разработчика — Join Photographic Experts Group. Этот формат обеспечивает 24-битовое кодирование цвета и лучше подходит для хранения таких изображений, как фотографии. Недостатком формата является возможность искажения цвета в результате сжатия данных.

Для Интернета был разработан еще один формат графики — PNG (Portable Network Graphics). Он создан с целью замены формата GIF. В отличие от своего прототипа новый формат позволяет использовать как 8-битовое, так и 24-битовое кодирование цвета. Кроме того, алгоритм создания прозрачности усовершенствован. С помощью альфа-канала прозрачность может быть задана для участков изображения, содержащих разные цвета. Для 24-битовых форматов прозрачность может быть неполной, то есть разработчик устанавливает ее величину в процентах (от 0 до 100). К сожалению, броузеры пока не научились воспроизводить полупрозрачные изображения.

Создание фона HTML-документа

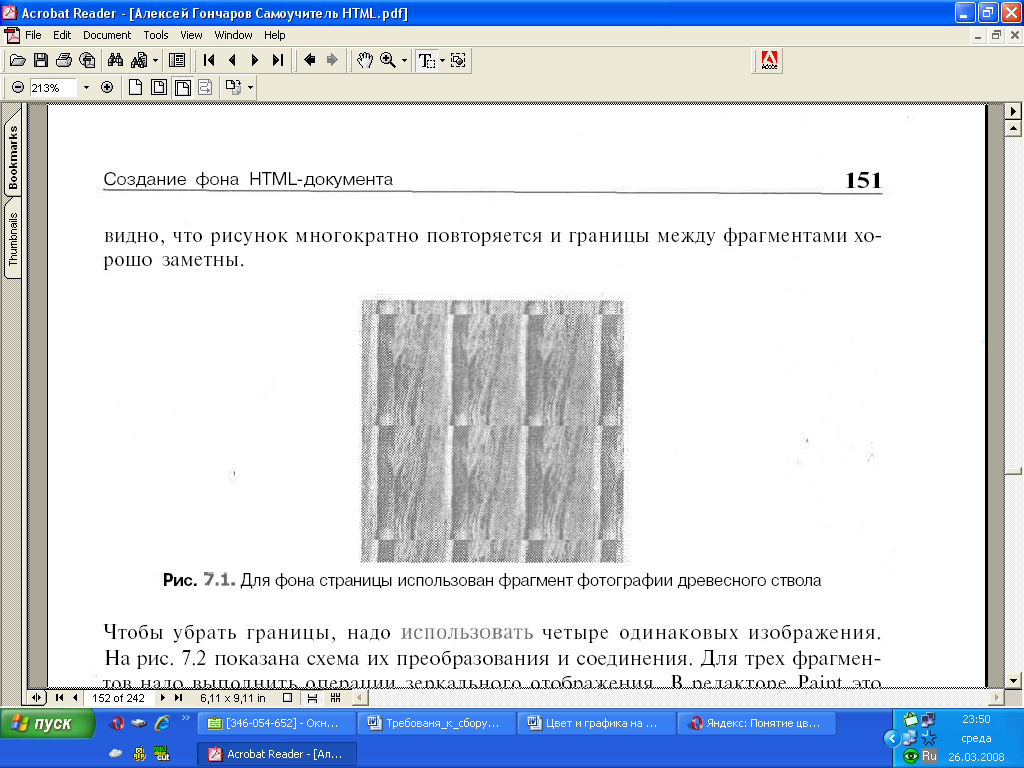
К фоновому рисунку HTML-документа, безусловно, предъявляется ряд требований. В зависимости от того, какой вид хочет придать своей странице автор, выбирается направление конструирования фона. До недавнего времени классическим решением было создание бледно-серого фона с таким же бледным, но рельефным рисунком. Здесь очень многое зависит от художника, но современные графические редакторы позволяют создавать похожие эффекты и автоматически. Такой фон не должен ощутимо снижать контрастность страницы и мешать чтению текста.

В последнее время в моде на фоновые рисунки произошли изменения. Все чаще можно встретить белый фон. Белый цвет вне конкуренции, и тут не надо что-либо объяснять. Другое направление — использование бледного фона, но другого цвета, например, бежевого, голубого или зеленого. Сам рисунок выглядит как произвольный узор из точек, напоминая поверхность кожи или камня. Здесь опять-таки, полет фантазии художника ограничивается необходимостью обеспечения контрастности.

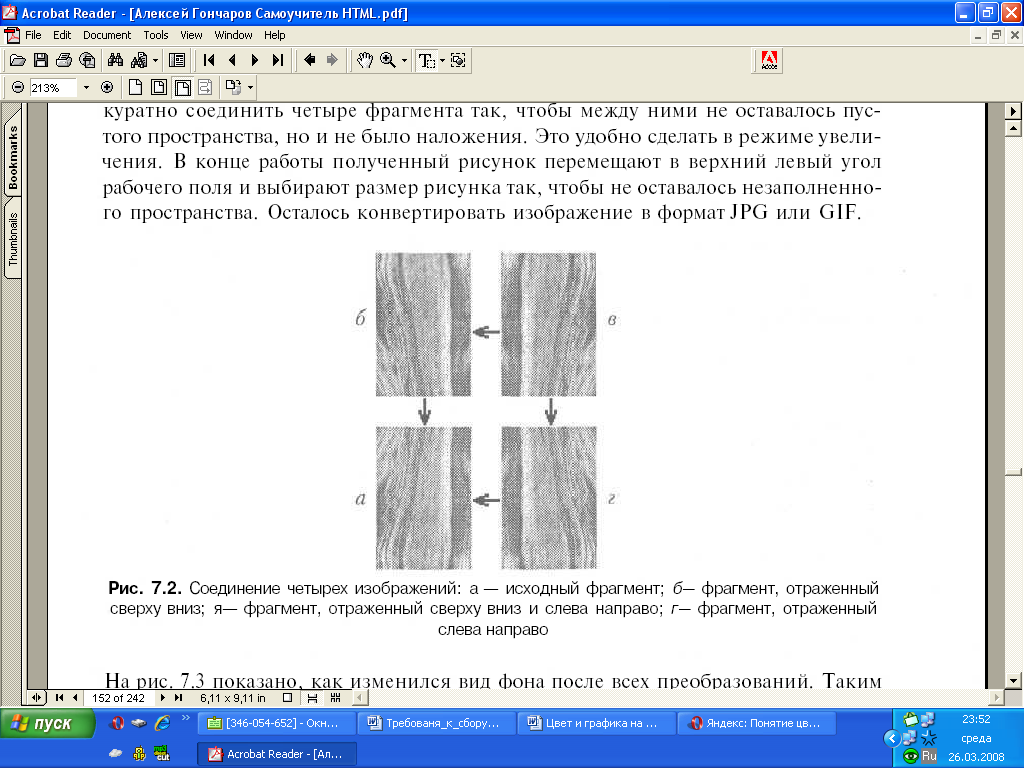
Наконец, для экзотических страниц самым популярным фоном остается черный. Его разнообразят изображениями звезд (для страниц с космической тематикой), создавая ночное небо, или дополняют кроваво-красными надписями или рисунками (если надо сделать «круто»).

В любом случае источником фона служит рисунок небольшого размера. Заранее невозможно предугадать, какой размер на экране займет документ, так как браузер выполняет автоматическое форматирование. В результате рисунок фона (фрагмент) будет тиражироваться так, чтобы заполнить все пространство окна. Разработчик страницы должен решить, как будут совмещаться левая и правая, а также верхняя и нижняя стороны исходного рисунка.

Рассмотрим процедуру подготовки изображения для фона. В качестве примера я хочу использовать текстуру. Как правило, она имеет нерегулярный рисунок и не содержит ярко выраженных деталей. Фрагментами могут быть изображения облаков, деревьев, камней, поверхности воды. Основой фона документа может стать и фрагмент фона картины, выполненной акварелью или маслом. Все это легко найти в Сети. Так, например, для получения древесного рисунка вовсе не обязательно искать изображение среза дерева. Можно использовать подходящую фотографию дерева и вырезать кусочек изображения ствола. Один раз мне попалась фотография старого дерева с содранной временем и ветром корой. Кусочек фотографии пригодился для создания «орехового» фона Web-страницы. Видно, что рисунок многократно повторяется и границы между фрагментами хорошо заметны.

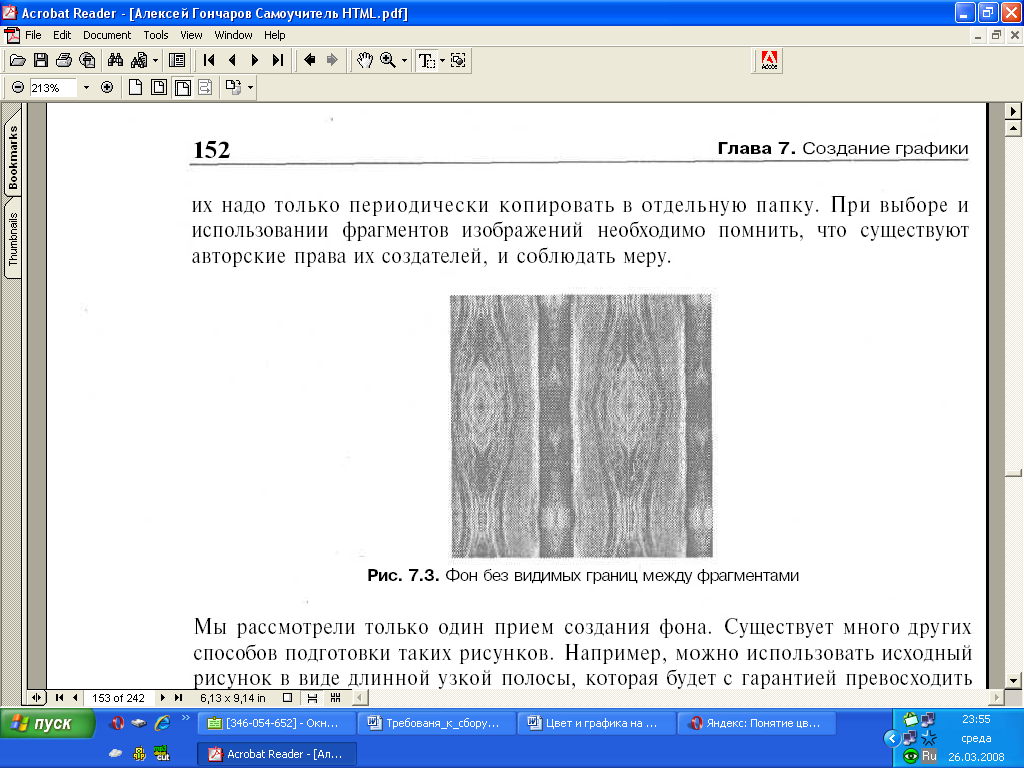


Чтобы убрать границы, надо использовать четыре одинаковых изображения. Для трех фрагментов надо выполнить операции зеркального отображения. В редакторе Paint это можно сделать, выбрав команду Рисунок > Отразить/повернуть. Затем требуется аккуратно соединить четыре фрагмента так, чтобы между ними не оставалось пустого пространства, но и не было наложения. Это удобно сделать в режиме увеличения. В конце работы полученный рисунок перемещают в верхний левый угол рабочего поля и выбирают размер рисунка так, чтобы не оставалось незаполненного пространства. Осталось конвертировать изображение в формат JPG или GIF.



Соединение четырех изображений: а — исходный фрагмент; б— фрагмент, отраженный сверху вниз; в— фрагмент, отраженный сверху вниз и слева направо; г— фрагмент, отраженный слева направо.

Далее показано, как изменился вид фона после всех преобразований. Таким образом, если вы путешествуете по Web, имеет смысл коллекционировать интересные фотографии и рисунки. Они все равно записываются в кэш на вашем диске, и их надо только периодически копировать в отдельную папку. При выборе и использовании фрагментов изображений необходимо помнить, что существуют авторские права их создателей, и соблюдать меру.



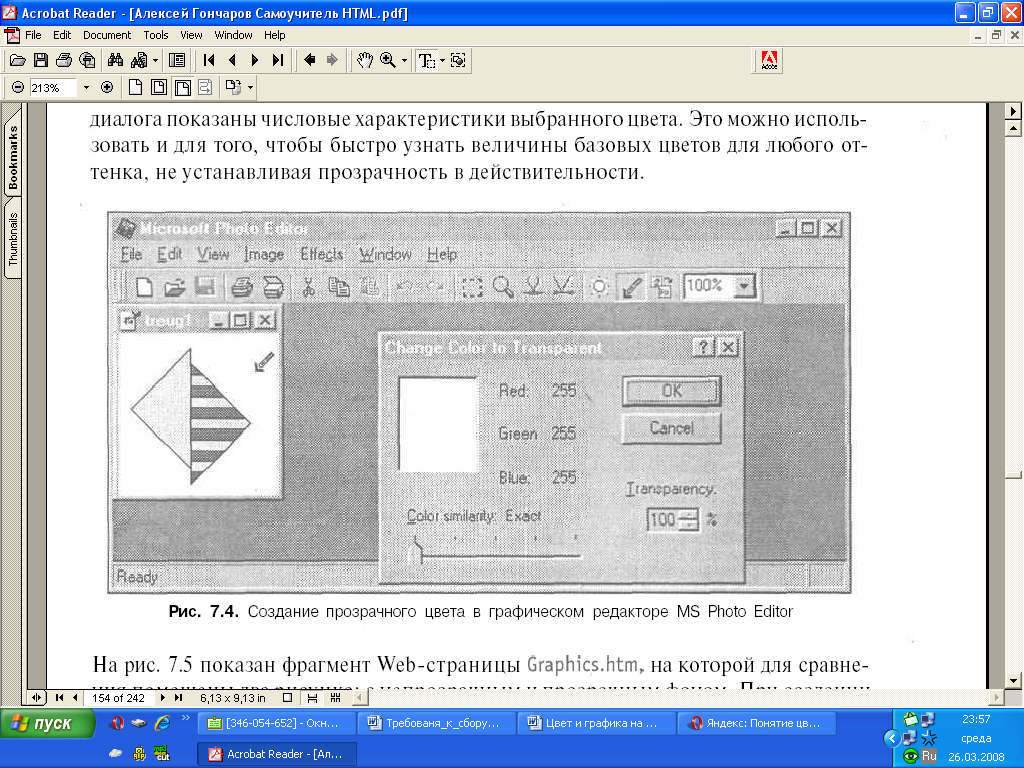
Мы рассмотрели только один прием создания фона. Существует много других способов подготовки таких рисунков. Например, можно использовать исходный рисунок в виде длинной узкой полосы, которая будет с гарантией превосходить ширину окна броузера. Высота полосы может быть уменьшена вплоть до одного пиксела. Учитывая, что в каждом ряде фона исходный рисунок будет воспроизводиться с самого начала, можно получить такие эффекты, как вертикальные полосы или плавные переходы цвета.

К исходному файлу фона предъявляются два основных требования: файл не должен быть большим, а фон не должен препятствовать свободному чтению документа.

Прозрачность для GIF и PNG-изображений

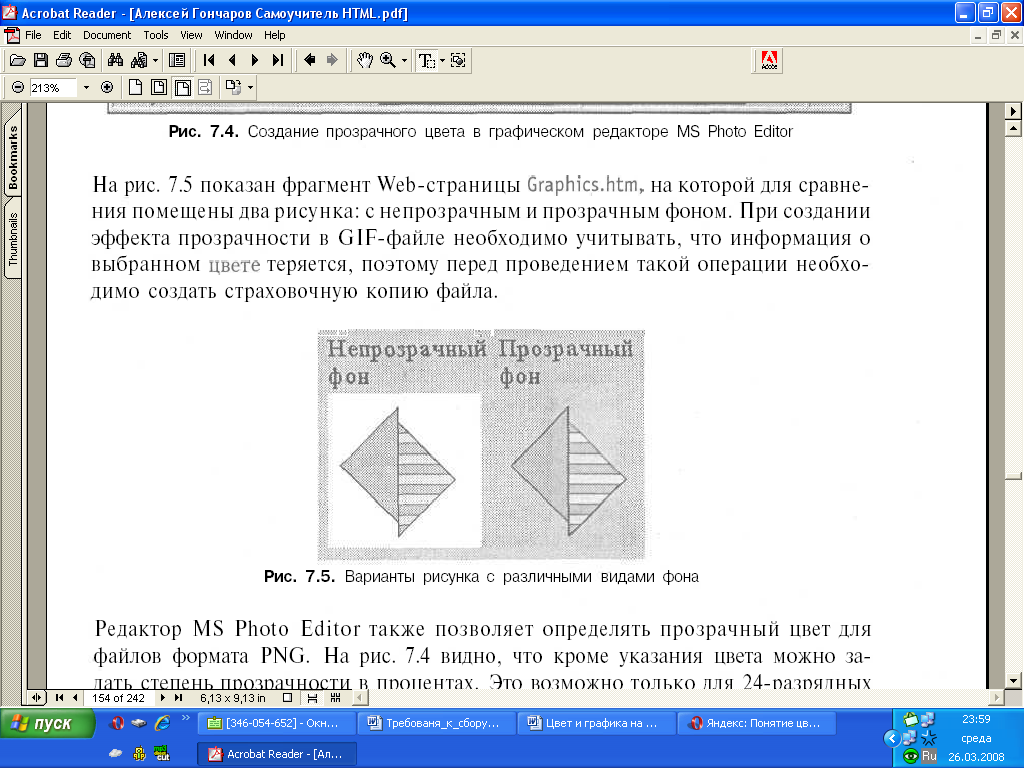
Выше, в разделе «Форматы графических файлов», уже упоминалось, что один из цветов изображения формата GIF можно сделать прозрачным. Чаще всего прозрачным делают цвет фона. Рисунок с прозрачным фоном, размещенный в документе, смотрится совсем иначе. Он как бы сливается с документом, становясь его неотъемлемой деталью.

Многие графические редакторы позволяют устанавливать прозрачность. Так, показан редактор изображений MS Photo Editor.



Для открытия окна, которое позволит выбрать нужный цвет, следует щелкнуть на кнопке Set Transparent Color (установить прозрачный цвет). Затем, когда указатель мыши превратится в наклонную стрелку, щелкните на нужном цвете. Обратите внимание, что в окне диалога показаны числовые характеристики выбранного цвета. Это можно использовать и для того, чтобы быстро узнать величины базовых цветов для любого оттенка, не устанавливая прозрачность в действительности.

Далее показан фрагмент Web-страницы Graphics.htm, на которой для сравнения помещены два рисунка: с непрозрачным и прозрачным фоном. При создании эффекта прозрачности в GIF-файле необходимо учитывать, что информация о выбранном цвете теряется, поэтому перед проведением такой операции необходимо создать страховочную копию файла.



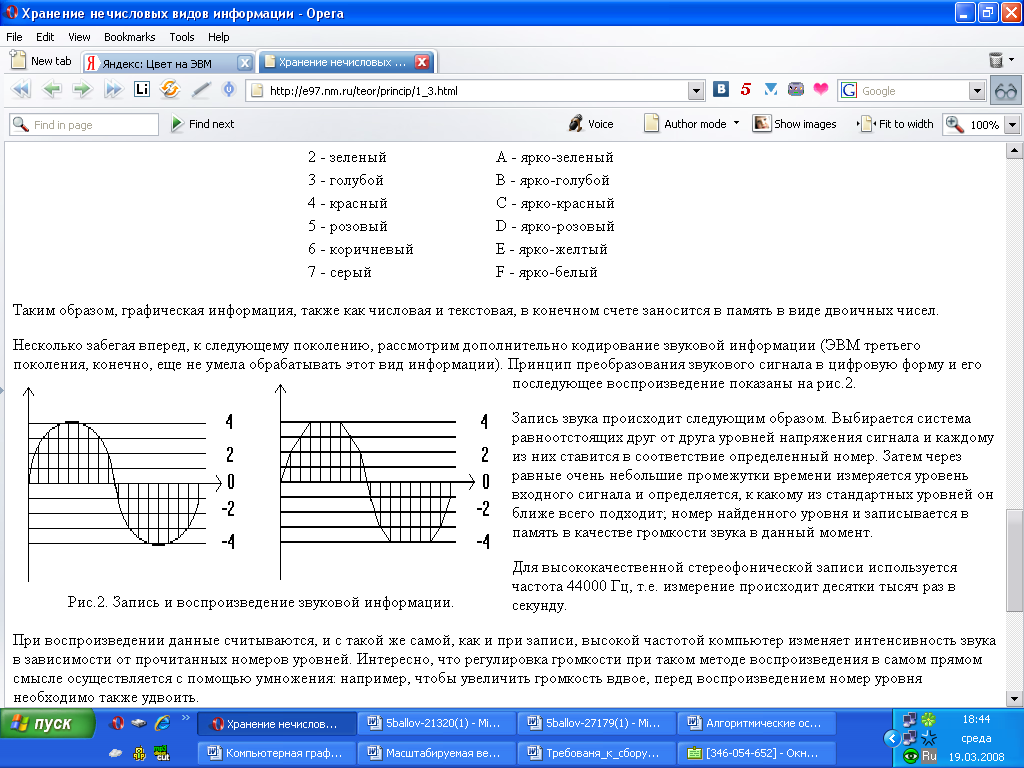
Редактор MS Photo Editor также позволяет определять прозрачный цвет для файлов формата PNG. На рисунке работы редактора видно, что кроме указания цвета можно задать степень прозрачности в процентах. Это возможно только для 24-разрядных файлов. Разумеется, любое изображение может быть конвертировано в такой формат.

Источники

1. Гончаров А. Самоучитель HTML. — СПб.: Питер

Кодирование звуковой информации

Несколько забегая вперед, к следующему поколению, рассмотрим дополнительно кодирование звуковой информации (ЭВМ третьего поколения, конечно, еще не умела обрабатывать этот вид информации). Принцип преобразования звукового сигнала в цифровую форму и его последующее воспроизведение показаны на рисунке.



Запись звука происходит следующим образом. Выбирается система равноотстоящих друг от друга уровней напряжения сигнала и каждому из них ставится в соответствие определенный номер. Затем через равные очень небольшие промежутки времени измеряется уровень входного сигнала и определяется, к какому из стандартных уровней он ближе всего подходит; номер найденного уровня и записывается в память в качестве громкости звука в данный момент.

Для высококачественной стереофонической записи используется частота 44000 Гц, т.е. измерение происходит десятки тысяч раз в секунду.

При воспроизведении данные считываются, и с такой же самой, как и при записи, высокой частотой компьютер изменяет интенсивность звука в зависимости от прочитанных номеров уровней. Интересно, что регулировка громкости при таком методе воспроизведения в самом прямом смысле осуществляется с помощью умножения: например, чтобы увеличить громкость вдвое, перед воспроизведением номер уровня необходимо также удвоить.

Таким образом, рассмотрев принципы хранения в ЭВМ различных видов информации, можно сделать важный вывод о том, что все они так или иначе преобразуются в числовую форму и кодируются набором нулей и единиц. Благодаря такой универсальности представления данных, если из памяти наудачу извлечь содержимое какой-нибудь ячейки, то принципиально невозможно определить, какая именно информация там закодирована: текст, число или картинка.

Источник

1. http://e97.nm.ru/teor/princip/1\_3.html