

## Теоретические сведения по обработке изображений для выполнения лабораторной работы №2

### 1. Получение негатива (для произвольных изображений)

$$R' = 255 - R$$

$$G' = 255 - G$$

$$B' = 255 - B$$

Значения R, G, B – цветовые компоненты исходного изображения.

### 2. Обработка медианным фильтром (делать только для полутоновых изображений)

Устранение шума с помощью медианного фильтра - выбор медианы среди значений яркости пикселей в некоторой окрестности.

Определение медианы:

$A_i, i = 1, n;$  - отсортированный набор чисел,

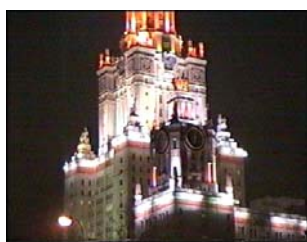
$A_{[n/2]}$  – медиана набора.

Медианный фильтр радиусом  $r$  – выбор медианы среди пикселей в окрестности  $[-r, r]$ , т.е. новый цвет пикселя с координатами  $x, y$  — это медиана среди всех пикселей в квадрате, ограниченном  $x-r, x+r, y-r, y+r$ .

### 3. Построение гистограммы (для полутоновых строится одна гистограмма, для цветных 3 - одна на каждый канал)

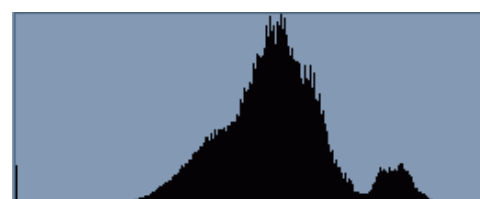
Гистограммы - представления распределения частот выбранных переменных. Каждому интервалу значений переменной соответствует некий счетчик (столбец гистограммы). Значение счетчика (высота столбца) соответствует частоте попадания значения переменной в данный интервал. Гистограмма – это график распределения тонов на изображении. На горизонтальной оси отображается шкала яркостей тонов от белого до черного, а на вертикальной оси указано число пикселей заданной яркости на изображении.

Пример.



0

255



0

255

### 4, 6 Линейная коррекция (для полутоновых применяется к одному каналу, для цветных

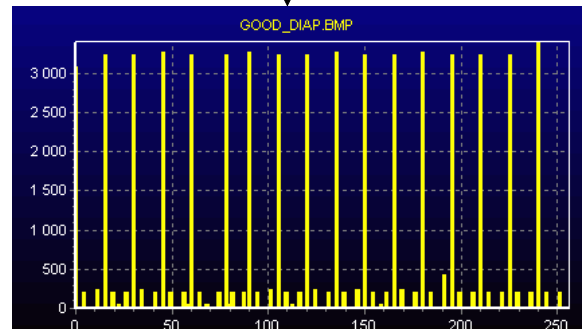
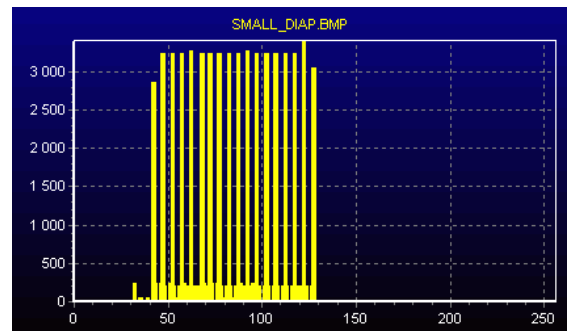
— к каждому из 3-х по отдельности)

Коррекция - к изображению применяется преобразование яркостей:  $y=f(x)$

$x$  – цвет пикселя на исходном изображении,  
 $y$  – цвет пикселя после коррекции.

$$y = (x - x_{\min}) * \frac{255}{x_{\max} - x_{\min}}$$

Компенсация узкого диапазона яркостей – линейное растяжение:

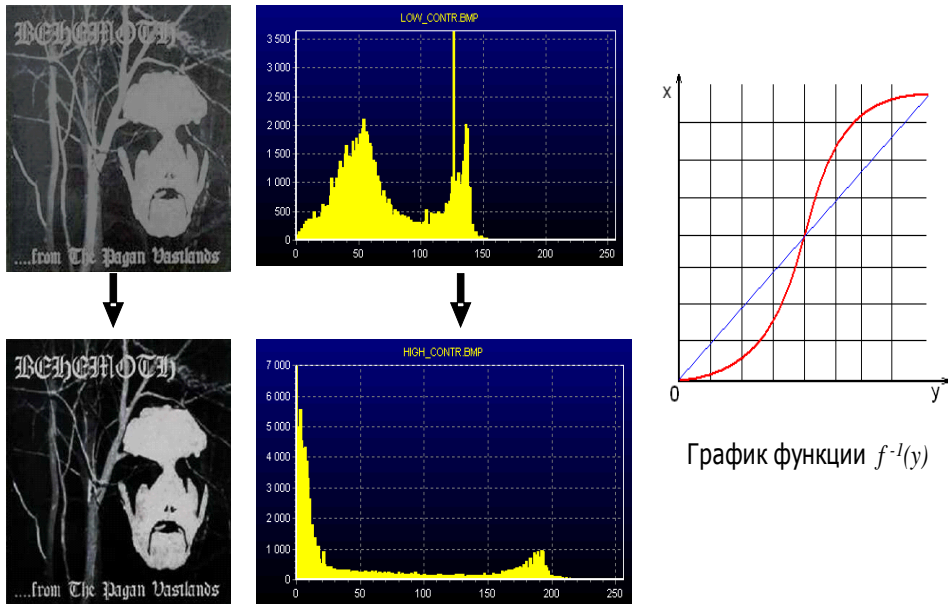


## 5 Нелинейная коррекция (для полутоновых применяется к одному каналу, для цветных — к каждому из 3-х по отдельности)

Нелинейная компенсация недостаточной контрастности (например, логарифмическая)  
Цель – сжатие динамического диапазона при визуализации данных.

$$y = c \cdot \log(1 + x)$$

Где  $c$  — некоторая константа (задается пользователем)



## 7 Поворот изображения (для произвольных изображений)

Новые координаты каждого пикселя рассчитываются по формулам:

$$\begin{aligned} x(k, l) &= (k, x_0)\cos(\mu) + (l, y_0)\sin(\mu) + x_0; \\ y(k, l) &= (k, x_0)\sin(\mu) - (l, y_0)\cos(\mu) + y_0; \end{aligned}$$

$x_0, y_0$  - центр поворота (задается пользователем),  
 $\mu$  - угол поворота (задается пользователем).  
 $(x, y)$  – расстояние между точками  $x, y$

Если пиксель попадает за пределы изображения, он отбрасывается. Пиксели, которые становятся свободными — им присваивается значение фона (задается пользователем)

## 8 Фильтрация произвольным фильтром (для произвольных изображений)

Фильтр всегда задается некоторой матрицей (чаще всего, квадратной  $n \times n$ , где  $n$  - нечетное).

$$F = \begin{bmatrix} F_{1,1} & \dots & F_{1,n} \\ \dots & \dots & \dots \\ F_{n,1} & \dots & F_{n,n} \end{bmatrix}$$

В этом случае значение цвета пикселя вычисляется по формуле :

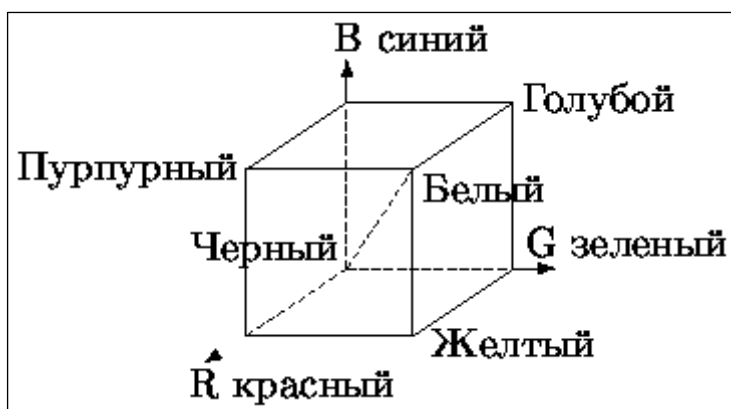
$$I(x, y) = \frac{1}{n^2} \sum_{i=-n/2}^{n/2} \sum_{j=-n/2}^{n/2} I(x-i, y-j) * F(i+n/2, j+n/2)$$

Размерность фильтра и значения коэффициентов задаются пользователем.

## 9, 10 Преобразование цветowych моделей (для произвольных изображений)

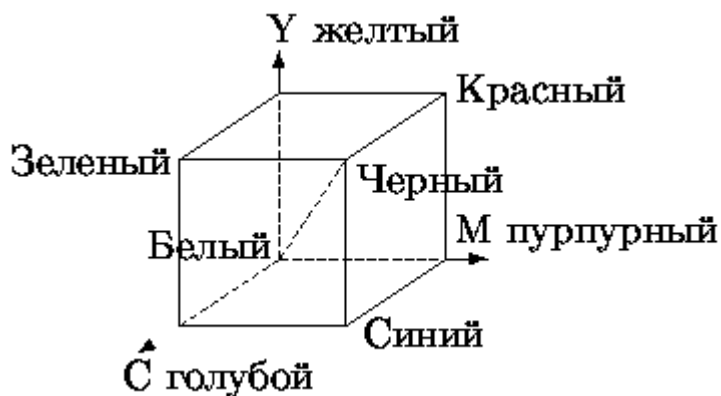
Назначение цветовой модели - дать средства описания цвета в пределах некоторого цветового охвата, в том числе и для выполнения интерполяции цветов. Наиболее часто в компьютерной графике используются модели RGB, CMY, YIQ, HSV и HLS.

RGB (Red, Green, Blue - красный, зеленый, синий) - аппаратно-ориентированная модель, используемая в дисплеях для аддитивного формирования оттенков самосветящихся объектов (пикселей экрана). Система координат RGB - куб с началом отсчета (0,0,0), соответствующим черному цвету.. Максимальное значение RGB - (255,255,255) соответствует белому цвету.



Модель RGB

CMY (Cyan, Magenta, Yellow - голубой, пурпурный, желтый) - аппаратно-ориентированная модель, используемая в полиграфии для субтрактивного формирования оттенков, основанного на вычитании слоев краски части падающего светового потока. Цвета модели CMY являются дополнительными к цветам модели RGB, т.е. дополняющими их до белого. Таким образом система координат CMY - тот же куб, что и для RGB, но с началом отсчета в точке с RGB координатами (255,255,255), соответствующей белому цвету.



Цветовой куб модели CMY

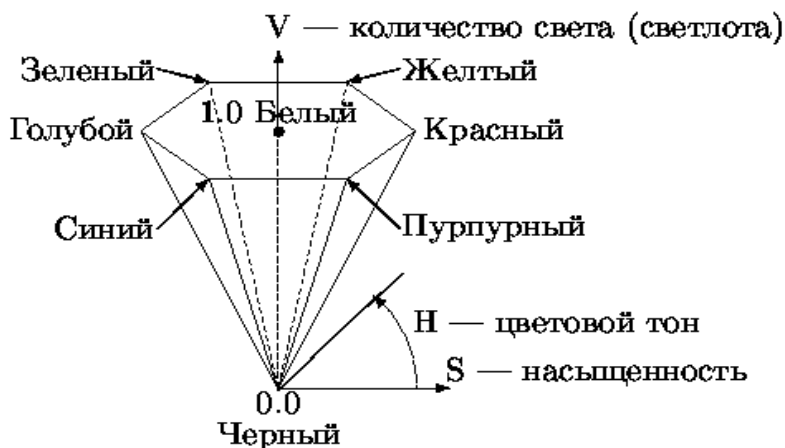
Преобразования между пространствами RGB и CMY определяются следующим образом:  $[R\ G\ B] = [255\ 255\ 255] - [C\ M\ Y]$ .

Причем единичный вектор-строка в модели RGB - представление белого цвета, а в модели CMY - черного.

## HSV

HSV (Hue, Saturation, Value - цветовой тон, насыщенность, количество света или светлота) - модель, ориентированная на человека и обеспечивающая возможность явного задания требуемого оттенка цвета.

Подпространство, определяемое данной моделью - перевернутый шестигранный конус.



По вертикальной оси конуса задается V - светлота, меняющаяся от 0 до 255. Значению  $V = 0$  соответствует вершина конуса, значению  $V = 255$  - основание конуса; цвета при этом наиболее интенсивны.

Цветовой тон H задается углом, отсчитываемым вокруг вертикальной оси. В частности,  $0^\circ$  - красный,  $60^\circ$  - желтый,  $120^\circ$  - зеленый,  $180^\circ$  - голубой,  $240^\circ$  - синий,  $300^\circ$  - пурпурный, т.е. дополнительные цвета расположены друг против друга (отличаются на  $180^\circ$ ).

Насыщенность S определяет насколько близок цвет к "чистому" пигменту и меняется от 0 на вертикальной оси V до 255 на боковых гранях шестигранного конуса.

Точка  $V = 0$ , в которой находится вершина конуса, соответствует черному цвету. Значение S при этом может быть любым в диапазоне 0-255. Точка с координатами  $V = 255$ ,  $S = 0$  - центр основания конуса соответствует белому цвету. Промежуточные значения координаты V при  $S=0$ , т.е. на оси конуса, соответствуют серым цветам. Если  $S = 0$ , то значение оттенка H считается неопределенным.

```
//На входе R,G,B - цвет пикселя, на выходе H,S,V - цвет
//пикселя в другой системе
begin
  max_value:=max_of(R,G,B);
  min_value:=min_of(R,G,B);
  diff:=max_value-min_value;
  V:=max_value;
  if max_value<>0 then s:=(diff / max_value) else S:=0;
  if S=0 then h:=undefined
  else
    begin
      r:=(max_value-R) / diff;
      g:=(max_value-G) / diff;
      b:=(max_value-B) / diff;
      if R=max_value
        then H:=b_dist-g_dist
```

```

else if G=Max_value
then H:=2+r_dist-b_dist
else if B=max_value
then H:=4+g_dist-r_dist;
H:=H*60;
if H<0 then H:=H+360;
end;
end;

```

### 11 Градиентный метод выделения контуров (реализовывать только для полутоновых изображений)

Контур (граница) на изображении — это множество пикселей, в которых наблюдается резкое изменение (перепад) яркости или цвета. Самый простой способ — оценивать градиент (изменение) по величине и направлению.

$$|\nabla g(x, y)| = \sqrt{\left(\frac{\partial g}{\partial x}\right)^2 + \left(\frac{\partial g}{\partial y}\right)^2}$$

Для того, чтобы не вычислять производную, переходят к конечным разностям и формула для градиента приобретает вид:

$$|\nabla g(x, y)| = \sqrt{[g(x+1, y) - g(x, y)]^2 + [g(x, y+1) - g(x, y)]^2}$$

Алгоритм поиска границы заключается в том, что для каждого пикселя оценивается значение градиента и если модуль градиента больше некоторого порога  $T$ , то пиксель считается лежащим на границе. Такие пиксели необходимо пометить каким-либо цветом (например, белым), а все остальные либо оставить без изменения, либо заменить на цвет фона. Выбор варианта отображения результата и величины порога  $T$  (1-255) осуществляется пользователем.

### 13 Зашумление (для произвольных изображений)

К каждому цветовому компоненту пикселя добавляется некоторая поправка (шум), которая генерируется случайным образом.

Например:

$$C' = C + (\text{Noise} * 255 - 127)$$

где

Noise - уровень шума (0...1)

C - старый цвет

C' - новый цвет

При расчете формулы следует учитывать, что значение может стать меньше 0 или больше 255. В этом случае необходимо принудительно ограничивать значение цвета.

Уровень шума вводится пользователем.

### 14 Эффект «стекла» (для произвольных изображений)

Расчет новых координат пикселей по формулам:

$$x(k, l) = k + (\text{rand}(1, 1) - 0.5) * 10;$$

$$y(k, l) = l + (\text{rand}(1, 1) - 0.5) * 10;$$



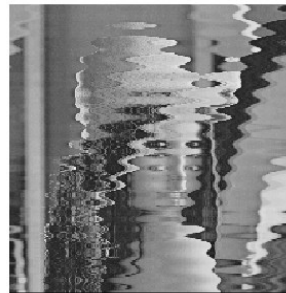
### 15 Эффект «волны» (для произвольных изображений)

Расчет новых координат пикселей по формуле:

$$x(k, l) = k + 20\sin(2\pi l / 128); y(k, l) = l;$$

или

$$x(k, l) = k + 20\sin(2\pi k / 30); y(k, l) = l;$$



### 16 Коррекция гаммы (для произвольного изображения)

Коррекция цвета пикселя в соответствии с формулой

$$y = c * x^\gamma$$

Параметры  $c$  и  $\gamma$  вводятся пользователем

### 17 Фильтр Sobel (для произвольного изображения)

Фильтрация фильтром в соответствии с п. 7, используя матрицу

$$h_1 = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

**18    Фильтр Prewitt (для произвольных изображений)**

Фильтрация фильтром в соответствии с п. 7, используя матрицу

$$h_1 = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$$