

*Государственное образовательное учреждение высшего профессионального  
образования*

*«Московский государственный технический университет*

*имени Н. Э. Баумана»*

*(МГТУ им. Н.Э. Баумана)*

---

ФАКУЛЬТЕТ

«Информатика и системы управления»

КАФЕДРА

«Программное обеспечение ЭВМ и информационные технологии»

**РАСЧЁТНО - ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**  
к курсовой работе на тему:

Разработка программы по выделению цветовой информации

Студент

\_\_\_\_\_ Киселев А.М.  
(Подпись, дата)

Руководитель курсового проекта

\_\_\_\_\_ Оленев А.А.  
(Подпись, дата)

Москва 2018

## Содержание

Введение . . . . .	3
1 Аналитический раздел . . . . .	4
1.1 Проблема человеческого восприятия . . . . .	4
1.2 Преобладающий цвет . . . . .	5
1.3 Цвет . . . . .	5
1.3.1 Цветовые модели и пространства . . . . .	5
1.3.1.1 RGB . . . . .	6
1.3.1.2 CIE-XYZ . . . . .	7
1.3.1.3 CIE-LAB . . . . .	7
1.3.2 Проблема нахождения различий между цветами . . . . .	8
1.3.3 Квантование цвета . . . . .	9
1.3.3.1 Алгоритм квантования цветов медианным сечением: . . . . .	9
1.3.3.2 Алгоритм кластеризации k-средних: . . . . .	10
1.3.3.3 Алгоритм LBA . . . . .	10
1.3.4 Цветовые гистограммы . . . . .	11
1.4 MPEG-7 . . . . .	11
1.5 Выделение преобладающих цветов . . . . .	13
2 Конструкторский раздел . . . . .	14
2.1 Идея . . . . .	14
2.2 Алгоритм . . . . .	14

## Введение

Цвет - один из важнейших атрибутов визуальной информации. Во многих вещах мы полагаемся на цвета: будь то светофор, кино, фотографии, картины. Наличие каких-то конкретных оттенков задает нужное настроение и атмосферу.

С каждым годом растет объем информации, которую нужно уметь качественно и быстро обрабатывать. Поэтому поисковые системы стараются расширить функционал своих продуктов, чтобы удовлетворить совершенно различные требования и желания пользователя. Одним из таких расширений является использование преобладающих цветов изображения. Данную технологию поисковые системы могут использовать в совершенно разных ключах. Помимо поиска по тегам(ключевым словам) поиск проводится по цветовому критерию. Например, мы хотим найти изображения по тегу 'футбол' и указываем доминирующий цвет - 'зеленый'. Таким образом, мы получаем множество картинок, которые с большой вероятностью содержат футбольное поле. Другой пример поиск по тегу 'море' с преобладающим цветом - 'красный'. Здесь мы уже получим изображения, которые содержат море, и преобладающий красный оттенок(вероятнее всего это будет море и закат).

Помимо поисковых систем, преобладающий цвет изображения может использоваться и в других продуктах. Системы, которые выводят визуальную информацию на экраны компьютера или телевизора могут автоматически генерировать подсветку вокруг экрана, которая будет соответствовать доминирующим цветам текущего отображаемого изображения. Такая технология позволяет увеличить эффект присутствия, снизить утомляемость глаз во время темного времени суток и дополнить ореолом интенсивность изображения на самом экране, визуально увеличивая размер картинки.

Оба рассмотренных примера работают с огромными объемами информации, что требует умения быстро обработать поступающие данные. Есть несколько алгоритмов, которые позволяют выделить преобладающие цвета изображения, но у каждого есть какие-либо недостатки, которые могут отразиться на последующей работе.

# 1 Аналитический раздел

## 1.1 Проблема человеческого восприятия

Для решения поставленной задачи в первую очередь надо выяснить, что такое преобладающий цвет. Это понятие неоднозначно и включает в себе следующую проблему. Цвет может быть преобладающим чисто математически/физически, а может быть преобладающим с точки зрения человека. Изображение, которое на 70% состоит из черного цвета и на 30% из оранжевого вносит неопределенность в выборе более преобладающего цвета. В первом случае доминантный цвет рассматривается как нечто физическое. В таком случае наиболее доминантный цвет - черный, потому что он занимает большую часть картинка. Во втором больше преобладает оранжевый, т.к с точки зрения человека, глаз в первую очередь обратит внимание на более яркую, выразительную точку, чем темную и тусклую.



Рисунок 1.1 — Наглядное изображение оранжевого круга на черном фоне

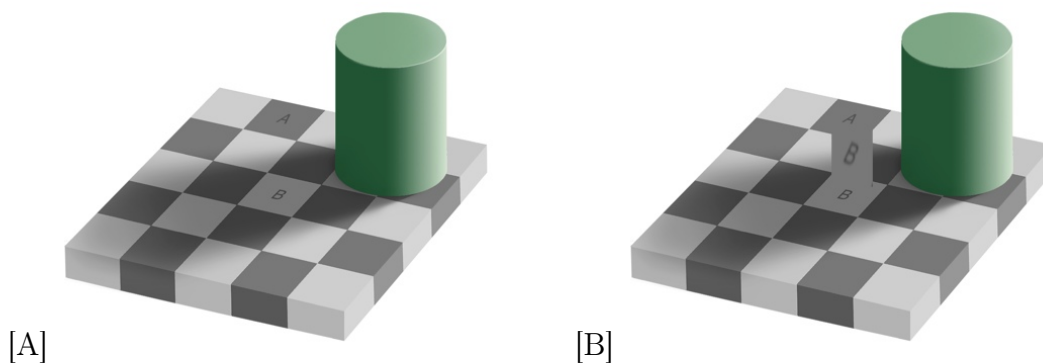


Рисунок 1.2 — Демонстрация психологических факторов человеческого восприятия(рисунок А-квадрат В кажется светлее квадрата А, но на самом деле они одного цвета(рисунок В))

В работе будет рассматриваться первый случай, где не учитываются психологические аспекты человеческого восприятия. Следует выделить четкое понятие доминантного цвета.

## 1.2 Преобладающий цвет

*Доминантный(преобладающий) цвет* – цвет, который представляет группу цветов пикселей, объединенных в единый кластер с помощью квантования.

Данное понятие было введено в стандарте MPEG-7 в 2002 году, одним из визуальных дескрипторов которого является дескриптор доминантного цвета(DCD – dominant color descriptor)

## 1.3 Цвет

С физической точки зрения цвет представляет собой свет, который, отражаясь от объекта, попадает в глаз человека. Восприятие цвета человеком может зависеть от психологического состояния индивида, от местоположения объекта, от строения глаза человека, от окружающего света и т.д. То есть восприятие цвета человеком достаточно субъективно. Свет в свою очередь можно описать как волну, длина которой возбуждает разные рецепторы человеческого глаза. То есть, индивид будет понимать какого цвета объект перед ним в зависимости от того, в какой диапазон попадет длина волны света, отраженного от этого объекта.

### 1.3.1 Цветовые модели и пространства

Модель цвета – абстрактная математическая модель представления цветов в виде кортежей чисел.

В какой-то момент необходимо было придумать модель цвета. Как и любое другое явление природы описать его так, чтобы цветовую информацию можно было удобно и эффективно использовать. Проблема описания цвета в форме математики была решена еще до появления компьютеров. Одним из первых таких описаний была RGB(Red, Green, Blue) модель, идея которого заключалась в представлении всех цветов с помощью трех базовых понятий - красного, зеленого и синего.

Цветовое пространство – модель представления цвета, основанная на использовании цветовых координат. Цветовое пространство ориентировано на то, чтобы каждый оттенок модели был представлен в виде отдельной точки, имеющей свои собственные координаты. Например, каждая точка цветового пространства RGB имеет три координаты, которые характеризуют ее положение в пространстве, где каждая соответствует своей компоненте: красному, зеленому или синему.

RGB не является одним единственным пространством. Список основных цветовых пространств:

- а) RGB, sRGB, Adobe RGB
- б) CIEXYZ, CIELAB
- в) CMY(K)

г) HSL, HSV

### 1.3.1.1 RGB

*RGB* – пространство характеризуется свойством аддитивности: стоит на составление цвета из трех базовых – красного(Red), синего(Blue) и зеленого(Green). Данное пространство часто называют цветовым кубом, потому что каждый базовый параметр цвета, представленного в этой модели, может восприниматься как координата трехмерного прямоугольного пространства.

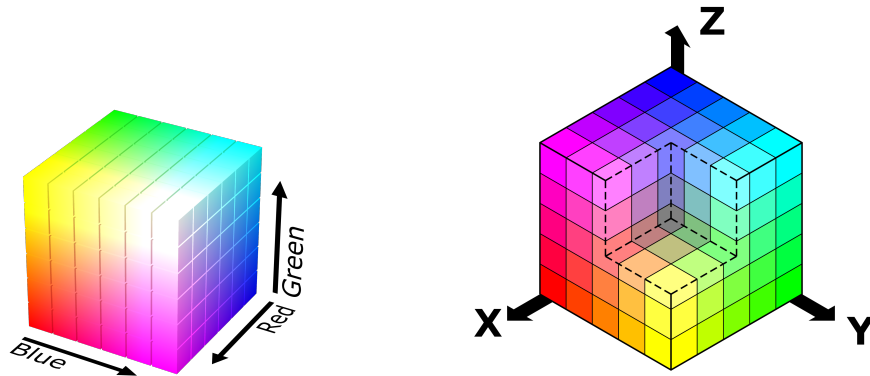


Рисунок 1.3 — Представление цветового куба

RGB пространство способно представить  $2^{24} = 16\,777\,216 / 2^8 * 2^8 * 2^8$  цветов (это может зависеть от качества графического оборудования). Числовые диапазоны значений (red, green, blue) могут быть представлены в разных видах: в процентах (от 0% до 100%), в виде действительных значений от 0 до 1 (обычно используется при работе с цветовым кубом, при аналитических анализах), в виде целых чисел от 0 до 255 (в компьютере информация представляется в виде битов и байтов)

Цветовая модель RGB была построена на основе закона Грассмана, который утверждает, что человек воспринимает свет линейно:

$\lambda$  – длина волны.

$I_1(\lambda)$  и  $I_2(\lambda)$  – спектральные функции, задающие источники света.

$I_h$  – спектральная функция, задающая световой поток, который воспримет человек.

$$I_h(\lambda) = I_1(\lambda) + I_2(\lambda) \quad (1.1)$$

Спектральная функция  $I(\lambda)$  – функция, которая в общем виде описывает световой поток.

Перевод RGB в CIE-LAB:

RGB  $\rightarrow$  CIE-XYZ  $\rightarrow$  CIE-LAB

### 1.3.1.2 CIE–XYZ

*CIEXYZ* (*CIE - International Commission on Illumination*) – модель, которая является экстраполяцией RGB модели. Данная модель охватывает все цвета, видимые человеком. Когда модель RGB расширили до видимых цветов появились отрицательные числа и чтобы избавиться от них были введены мнимые основные цвета  $X$  (мнимый красный),  $Y$  (мнимый зеленый),  $Z$  (мнимый синий).

### 1.3.1.3 CIE–LAB

*CIELAB* ( $L^*a^*b^*$ ) – цветовая модель, которая может отображать цвета за пределами, распознаваемыми человеком. Основывается на трех параметрах:  $L$  – яркости (Lightness) и двух цветовых каналов  $a$  и  $b$ . Она была построена на основе модели CIE–XYZ с ориентированием на человеческое восприятие. Цветовое пространство данной модели максимально приближено к тому, как люди воспринимают цветовые переходы.

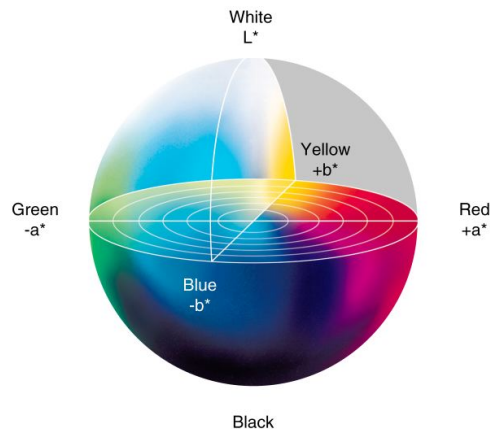


Рисунок 1.4 — Представление цветовой модели CIALAB.

RGB  $\rightarrow$  CIE–XYZ:

Значения цветности можно рассчитать так ( $X$ ,  $Y$ ,  $Z$  – значения, представляющие CIE–XYZ):

$$\begin{aligned} x &= \frac{X}{X + Y + Z} \\ y &= \frac{Y}{X + Y + Z} \\ z &= \frac{Z}{X + Y + Z} \end{aligned} \tag{1.2}$$

Если цвет задан таким образом ( $x$ ,  $y$ ,  $Y$ ), где ( $x$ ,  $y$ ) – точка на диаграмме цветности и  $Y$  – яркостная компонента, то из формул ( 1.2) следует, что:

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} \frac{Y}{y}x \\ Y \\ \frac{Y}{y}(1-x-y) \end{pmatrix} \quad (1.3)$$

Тогда, если базисные RGB цвета заданы как  $(x_R, y_R, Y_R)$   $(x_G, y_G, Y_G)$   $(x_B, y_B, Y_B)$ , то получаем следующую формулу преобразования:

$$\begin{aligned} z_R &= 1 - x_R - y_R \\ z_G &= 1 - x_G - y_G \\ z_B &= 1 - x_B - y_B \end{aligned} \quad (1.4)$$

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{bmatrix} \frac{Y_R}{y_R}x_R & \frac{Y_G}{y_G}x_G & \frac{Y_B}{y_B}x_B \\ Y_R & Y_G & Y_B \\ \frac{Y_R}{y_R}z_R & \frac{Y_G}{y_G}z_G & \frac{Y_B}{y_B}z_B \end{bmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix} \quad (1.5)$$

CIE-XYZ  $\rightarrow$  CIE-LAB:

Пусть  $(X_W, Y_W, Z_W)$  – точка белого в CIEXYZ пространстве и  $F(s)$  представлена так:

$$F(s) = \begin{cases} 7.787s + \frac{16}{116}; & 0 \leq s \leq 0.008856 \\ s^{1/3}; & s \geq 0.008856 \end{cases} \quad (1.6)$$

$$\begin{aligned} L^* &= 116F\left(\frac{Y}{Y_W}\right) - 16 \\ a^* &= 500[F\left(\frac{X}{X_W}\right) - F\left(\frac{Y}{Y_W}\right)] \\ b^* &= 200[F\left(\frac{Y}{Y_W}\right) - F\left(\frac{Z}{Z_W}\right)] \end{aligned} \quad (1.7)$$

### 1.3.2 Проблема нахождения различий между цветами

Большинство людей могут отличать цвета и их оттенки чисто визуально, опираясь на собственные цветовые ощущения. Листья деревьев, кирпичи дома – все это содержит в себе оттенки одного цвета, который человек в состоянии различить интуитивно.

Но компьютер не может опираться ни на какие ощущения, поэтому встает проблема нахождения различий между цветами. Как компьютер может определить, что цвета похожи или совершенно разные?

Работая в цветовом пространстве, можно использовать Евклидово расстояние, но оно будет работать не так эффективно, как может показаться. В большинстве случаев оно будет находить расстояние корректно (здесь уже все зависит от цветового пространства), но в некоторых цветовых пространствах могут возникнуть случаи,



когда Евклидово расстояние в цветовом пространстве между цветами будет одинаковым, но цвета, которые сравниваются, будут совершенно разными. Например, в RGB, в CMYK могут возникнуть такие ситуации.

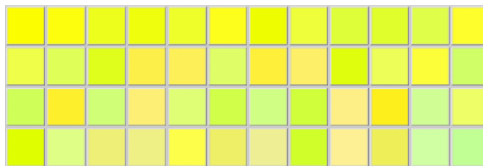


Рисунок 1.5 — Евклидово расстояние, примененное к RGB

### 1.3.3 Квантование цвета

Квантование – разбиение диапазона значений некоторой величины на конечное число уровней и округление этих значений до ближайших к ним уровней.

Квантование цвета в изображении важнейшая задача в поиске преобладающих цветов. Она позволяет уменьшить количество цветов отображаемой картинки. Это активно используется при сжатии, позволяя уменьшить цветовую палитру, при добавлении эффектов на изображение. Следующие алгоритмы позволяют решить данную задачу:

- а) Линейно-блочный алгоритм, основанный на слиянии блоков(LBA – Linear Block Algorithm).
- б) Алгоритм квантования цветов медианным сечением.
- в) Алгоритм кластеризации k-средних.

#### 1.3.3.1 Алгоритм квантования цветов медианным сечением:

Данный метод заключается в разбиении цветового пространства на параллелепипеды со сторонами, параллельными осям цветового пространства RGB.

Первый шаг заключается в нахождении минимального параллелепипеда, который содержит все цвета, представленные в изображении.

На втором шаге происходит определение самой длинной стороны параллелепипеда и сортировка всех значений вдоль выбранного направления. Далее параллелепипед разделяется по медиане множества значений выбранного направления на две части. Отсюда, получится два параллелепипеда, которые содержат примерно одинаковое количество значений.

Предыдущая процедура повторяется до тех пор, пока не будет получено N параллелепипедов, где N - количество цветов новой палитры. После этого требуется заполнить палитру цветами, которая будет описывать изображение. Для каждого параллелепипеда нужно рассчитать цвет, который будет представлять его(либо

центральная точка параллелепипеда, либо среднее арифметическое значение точек, попавших в него)

На самом изображении остается проанализировать пиксель (найти параллелепипед, в который попадает данная точка) и заменить цвет пикселя на цвет, который представляет весь параллелепипед.

### 1.3.3.2 Алгоритм кластеризации k-средних:

Метод кластеризации основан на центроидах – точках, которые представляют собой центр кластера.

Первый шаг алгоритма заключается в инициализации центроидов, количество которых равно  $N$  (размер требуемой палитры). Инициализация центроидов очень важный момент, который сильно влияет на работу всего алгоритма. Можно взять случайные центроиды, но это возможно приведет к погрешности.

Далее выделяется два основных шага данного алгоритма: нахождение кластеров, которые определяются путем кратчайшего расстояния от точки до центроида, и рассчитывание центра масс получившихся кластеров, смещение центроидов на полученные центры масс. Эти два шага повторяются до тех пор, пока центроиды не стабилизируются и больше не будут смещаться.

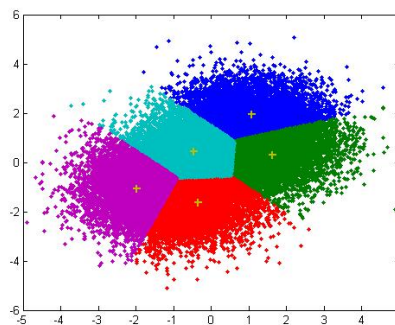


Рисунок 1.6 — Результат работы k-means с представленными центроидами

### 1.3.3.3 Алгоритм LBA

Первый шаг алгоритма заключается в разделение RGB пространства на 8 равных частей. Далее следует определить центроиды каждой части. Эти центроиды будут представлять собой цвет, полученный в процессе квантования цвета.

Пусть  $x = (x^R, x^G, x^B)$  представляют цвет пикселя, где  $x^R$  – компонента красного цвета,  $x^G$  – компонента зеленого цвета,  $x^B$  – компонента синего цвета и  $c_i$  – цвет, полученный в процессе квантования для  $i$ -ого раздела rgb. Усредненное значение цветов для каждого раздела может быть рассчитано так:

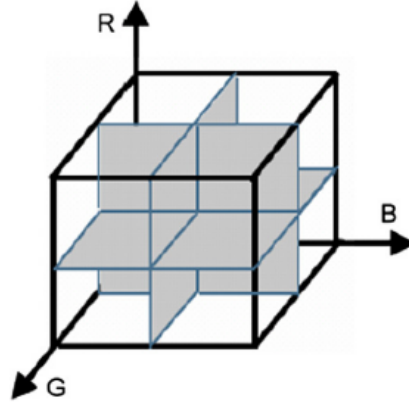


Рисунок 1.7 — Разделение цветового пространства RGB

$$\vec{x}_i = \frac{\sum_{x \in c_i} x}{\sum_{x \in c_i} 1} \quad (1.8)$$

После получения усредненных значений цвета, для каждого раздела можно получить цвет квантования  $c_i = (\vec{x}^R, \vec{x}^G, \vec{x}^B)$ ,  $1 \leq i \leq 8$ . Далее для всех пар соседних разделов рассчитываем расстояние между  $c_i$ , объединения затем похожие кластеры, используя следующие выражения:

$$\begin{aligned} x^R &= x_1^R \times \left( \frac{P_{R,1}}{P_{R,1} + P_{R,2}} \right) + x_2^R \times \left( \frac{P_{R,2}}{P_{R,1} + P_{R,2}} \right) \\ x^G &= x_1^G \times \left( \frac{P_{G,1}}{P_{G,1} + P_{G,2}} \right) + x_2^G \times \left( \frac{P_{G,2}}{P_{G,1} + P_{G,2}} \right) \\ x^B &= x_1^B \times \left( \frac{P_{B,1}}{P_{B,1} + P_{B,2}} \right) + x_2^B \times \left( \frac{P_{B,2}}{P_{B,1} + P_{B,2}} \right) \end{aligned} \quad (1.9)$$

В выражении ( 1.9)  $P_R, P_G, P_B$  – процентное представление компонент RGB пространства. Процесс слияния происходит до тех пор, пока Евклидово пространство между соседними разделами начнет быть больше, чем заданное приближение.

### 1.3.4 Цветовые гистограммы

Гистограмма – график статического распределения цифрового изображения с различной яркостью, в котором по горизонтальной оси представлена яркость, а по вертикали – относительное число пикселей с конкретным значением яркости. С помощью гистограммы определяется насыщенность изображения (либо сумарная, либо разделенная по цветовым каналам).

## 1.4 MPEG-7

MPEG-7 – стандарт ISO/IEC, разработанный группой MPEG (Moving Picture Experts Group). Аббревиатура расшифровывается как Multimedia Content Description

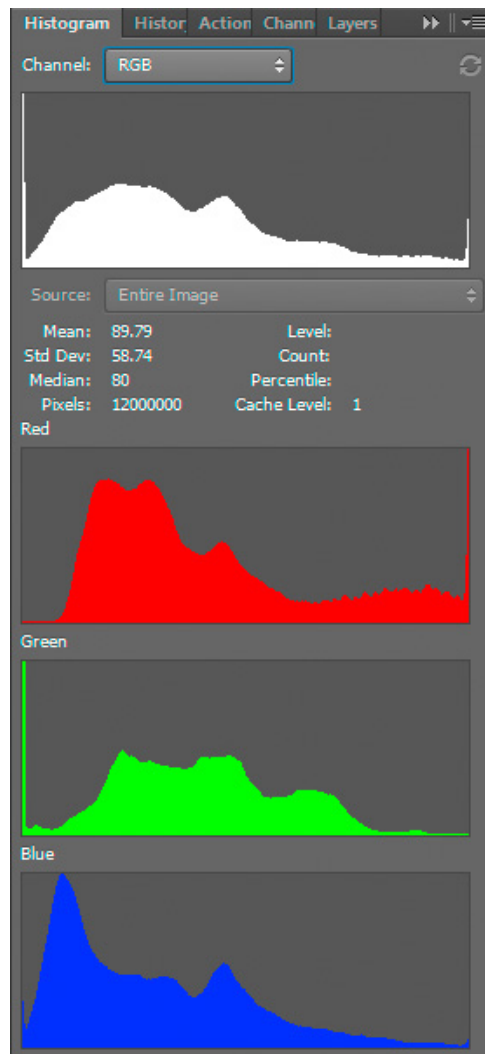


Рисунок 1.8 — Гистограммы, характеризующие изображение в программе Photoshop CS6

Interface(Мультимедиа-интерфейс для описания содержимого). Имеет цель стандартизировать и описать мультимедийное пространство.

Стандарт состоит из семи частей:

- а) Системы MPEG-7.
- б) Язык описания определений MPEG-7.
- в) Audio – дескрипторы и схемы описания аудио материала.
- г) Visual – дескрипторы и схемы описания визуального материала.
- д) Multimedia Descriptor Schemes – дескрипторы и схемы описания общих характеристик описания мультимедиа.
- е) Reference Software – программные реализации соответствующих частей стандарта.
- ж) Conformance – базовые принципы и процедуры тестирования рабочих характеристик практических реализаций стандарта.

## 1.5 Выделение преобладающих цветов

Выделение преобладающих цветов целиком опирается на квантование цветов изображения. Другими словами, задача сводится к определению кластеров и их центроидов, которые и будут являться преобладающими цветами.

## **2 Конструкторский раздел**

Рассмотрим задачу выделения преобладающих цветов в режиме реального времени в большом потоке данных.

### **2.1 Идея**

При поступлении большого потока данных, требуется сразу выделять доминантные цвета. Каждый раздел, который характеризует преобладающий цвет, должен иметь свой вес – частоту возникновения в изображении. Далее должно происходить квантования цветов. Здесь очень многое зависит от самого алгоритма, ведь он должен работать не только быстро, но и более менее правильно. После завершения остается выдать самый весомый преобладающий цвет и продолжить работу со следующим изображением в очереди.

### **2.2 Алгоритм**

Изображение с большим разрешением имеет много пикселей. С точки зрения цветов большое количество пикселей может быть избыточно, поэтому для их уменьшение изображения следует уменьшить. Это позволит алгоритму квантования обрабатывать более меньшие объемы. На этапе анализа были приведены несколько алгоритмов и выбор был сделан в сторону самого быстрого алгоритма – LBA. Данный алгоритм очень быстр и эффективен с математической точки зрения. Позволяет выделить только самые основные цвета, которые не пересекаются между собой, что связано с методом слияния кластеров.