



**Спецкурс ОСФИ 2015**

**Лекция 2**

**16 марта 2015**

# Цвет. Цветовые пространства. Гамма-коррекция

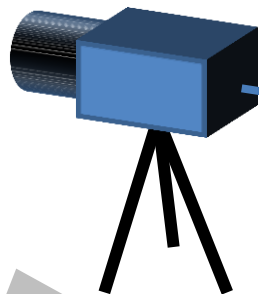
---

Алексей Игнатенко, к.ф.-м.н.

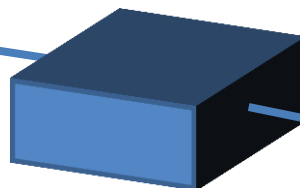
Лаборатория компьютерной графики и  
мультимедиа ВМК МГУ

# Свет и цвет в графической системе

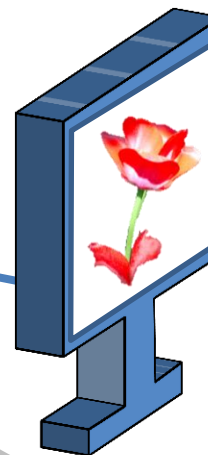
---



Получение  
изображения



Хранение и  
обработка



Вывод  
изображения

Наблюдатель



# Цветность и яркость

---

Восприятие человека позволяет различать

- Яркость
- Оттенок
- Насыщенность

Оттенок+Насыщенность = Цветность (chrominance), Хрома  
Яркость (luminance)

# Передача (отображение) цвета

---

Как соответствуют друг другу

- Видимый глазом свет
- Цвет на мониторе / проекторе
- Цвет на фотографии
- Цвет в графическом редакторе
- Цвет объектов в OpenGL?

# Машинное представление цвета

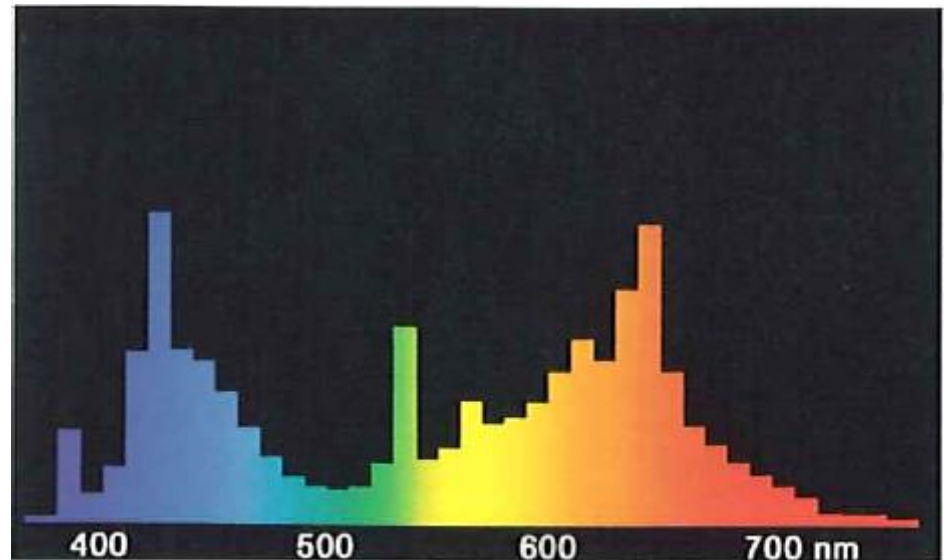
---

- Проблема 1: Как однозначно описать цвет?
  - Цвет – это не энергетический спектр!
  - Очень сложный механизм восприятия!
- Проблема 2: Цифровое представление цвета в компьютере

# Машинное представление цвета: квантованный спектр

---

- Можно взять видимый спектр (380-780нм) и квантовать его с небольшим шагом (обычно 5-10нм)
- 40 float на точку = 160В на пиксель
- Изображение 1Мп = 160 мегабайт!

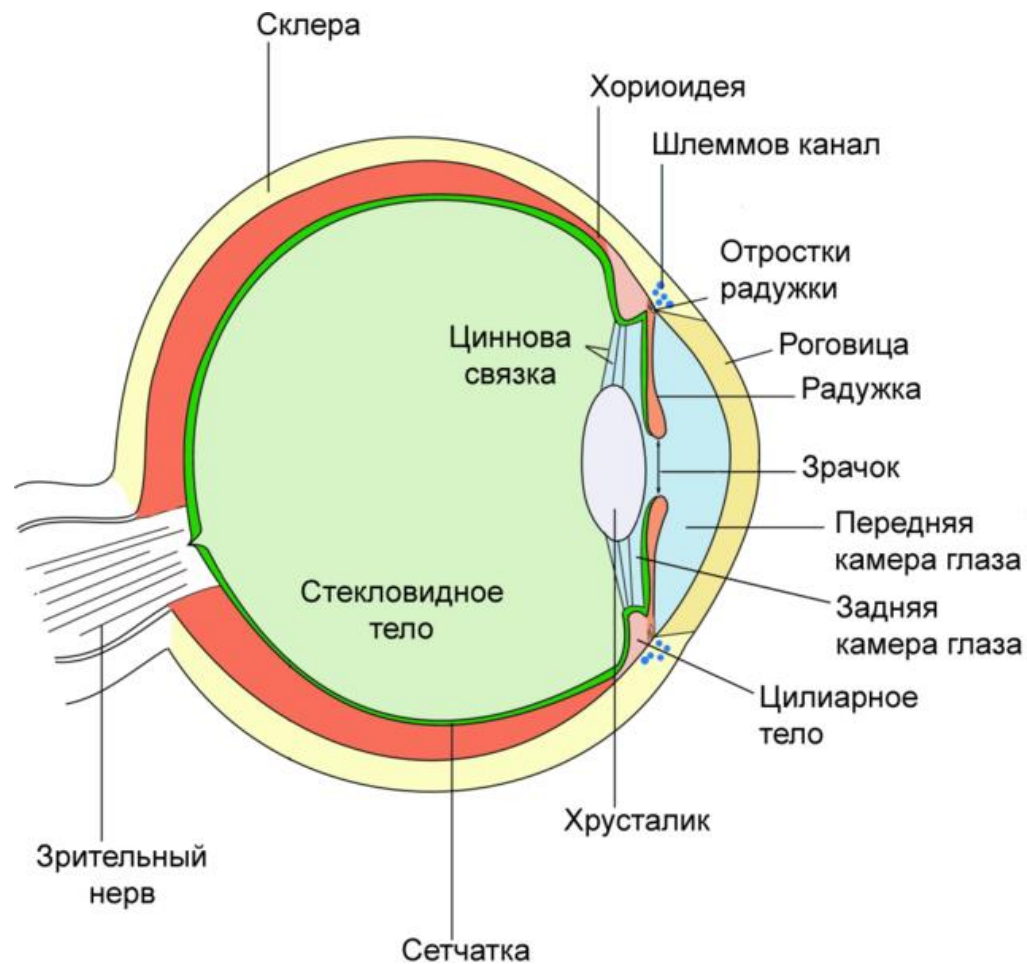


# Зрения человека

---

Нам нужно только то,  
что видит человек

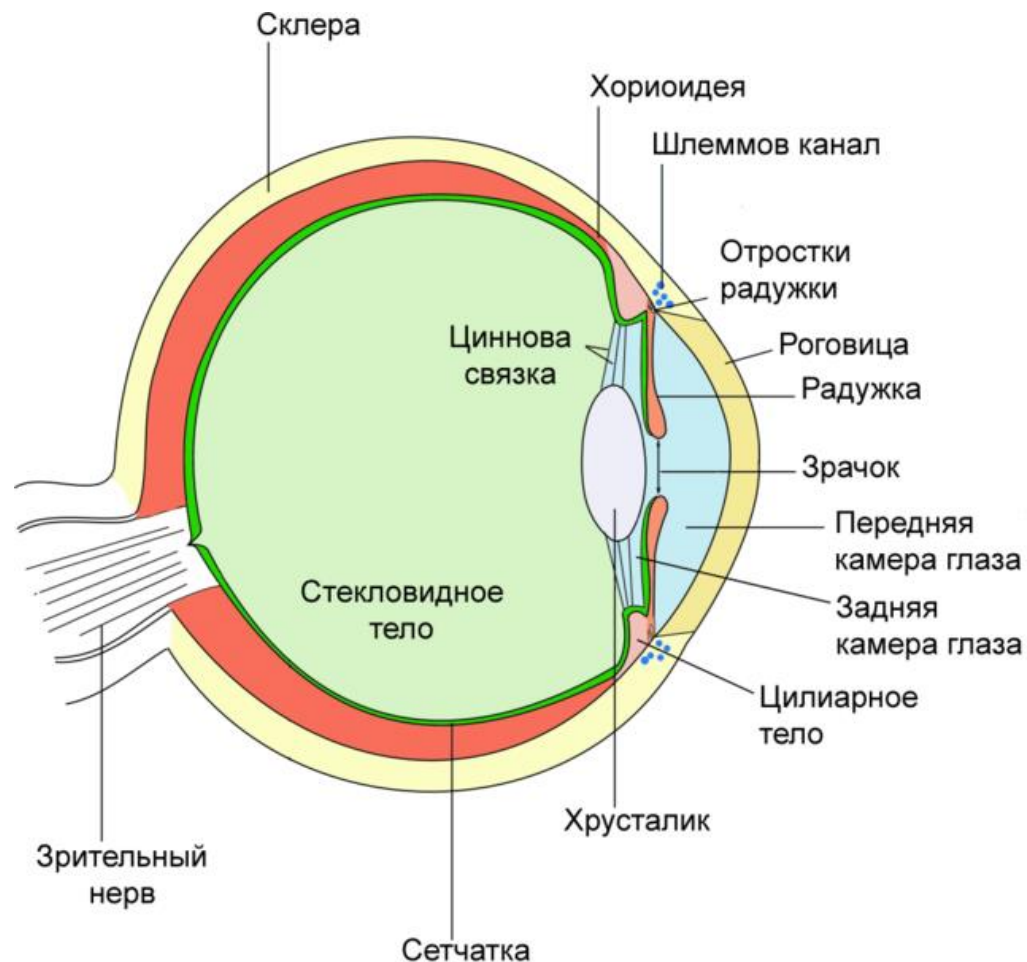
Надо разобраться в  
устройстве светового  
восприятия



# Как мы видим свет

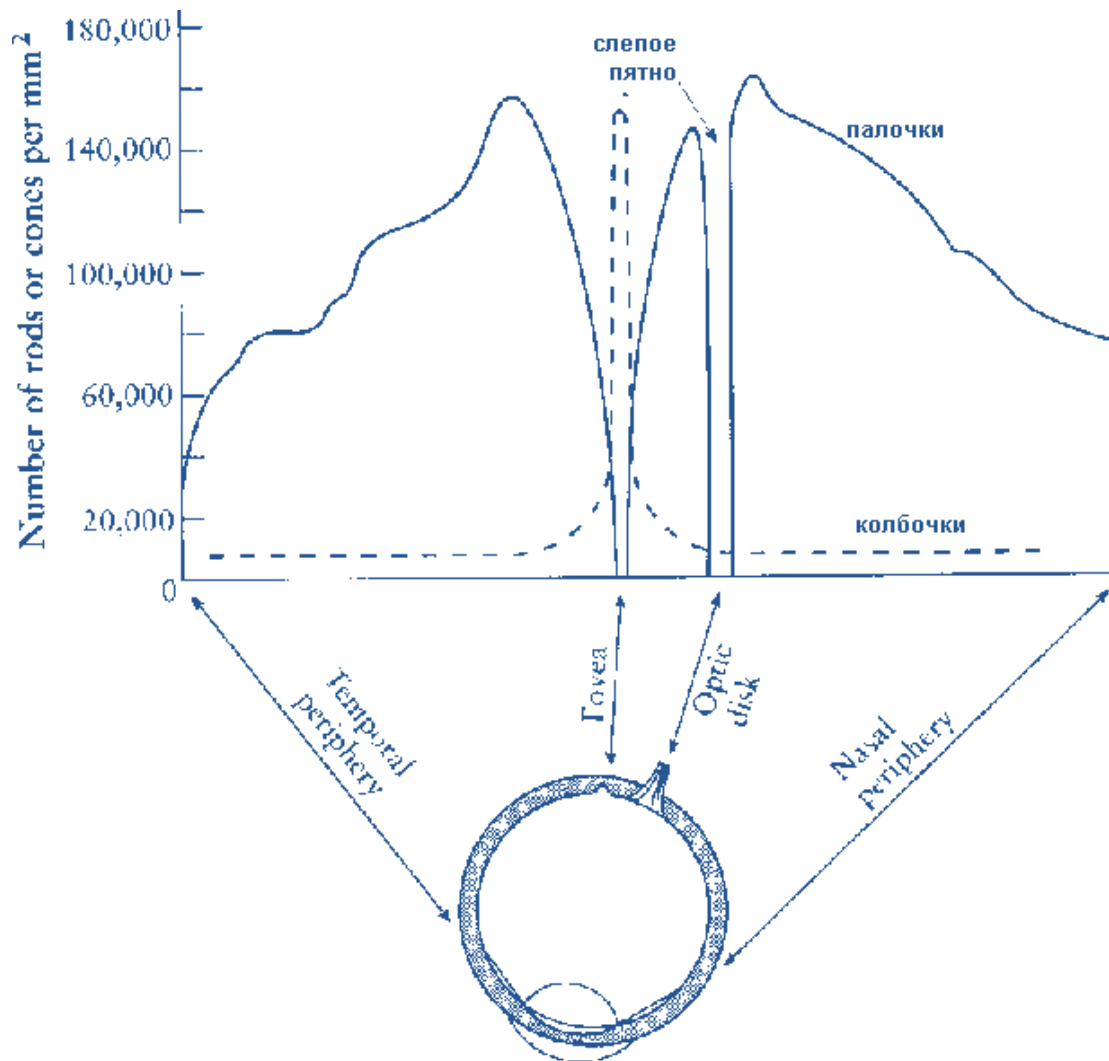
## Как мы видим свет:

- Световые лучи входят в глаз через роговицу
  - фокусировка
- Проходят через зрачок, окруженный радужкой
  - Изменение количества света
- Проходят через хрусталик
  - дальнейшая фокусировка
- Проходят через стекловидное тело
- Попадают на сетчатку





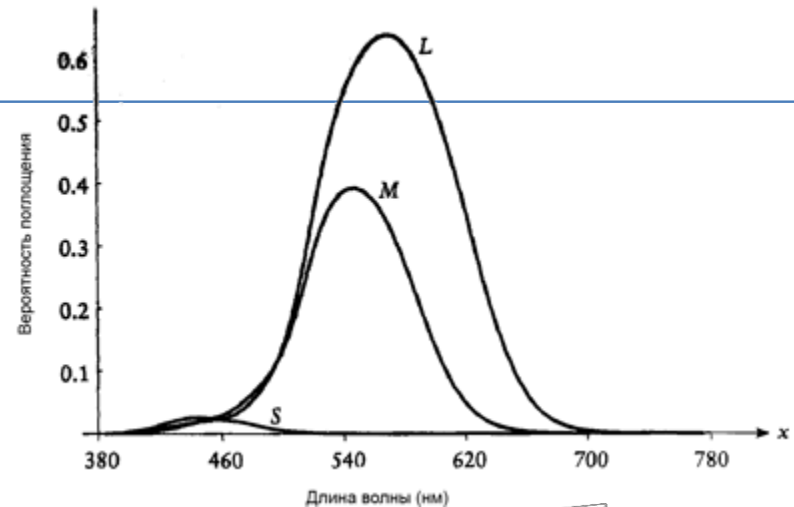
# Колбочки (цвет) и палочки



В каждом глазу 6 млн колбочек и 120 млн палочек

# Спектральное восприятие цвета: Колбочки

- Три вида колбочек
- Колбочки каждого вида содержат свой особый пигмент
- Три типа колбочек называют S, M и L
- Пики чувствительности приходятся примерно на 440 нм, 545 нм и 580 нм

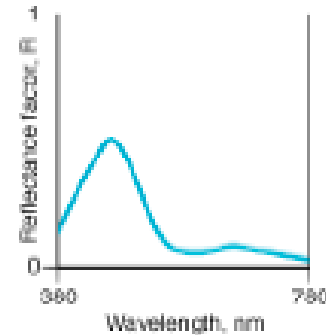
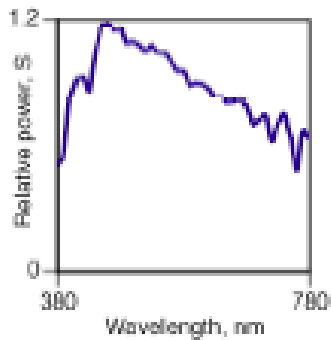


М. В. Ломоносов 1756

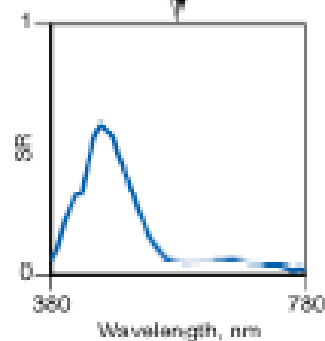
Томас Юнг 1807

Гельмгольц 1852

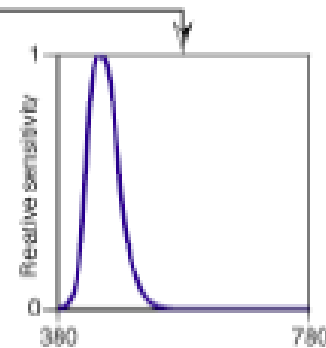
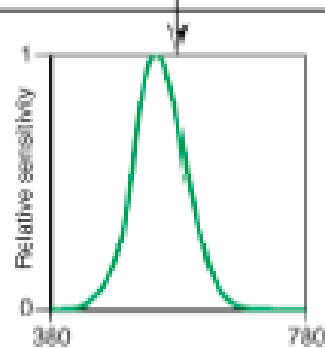
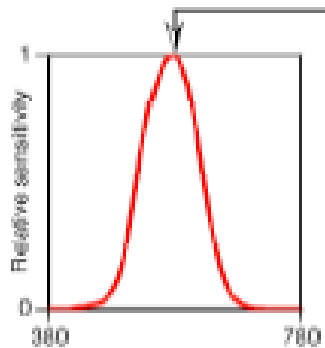
# Трихроматическая теория



object



reflected light



S, M, L filters

# Что такое цвет?

---

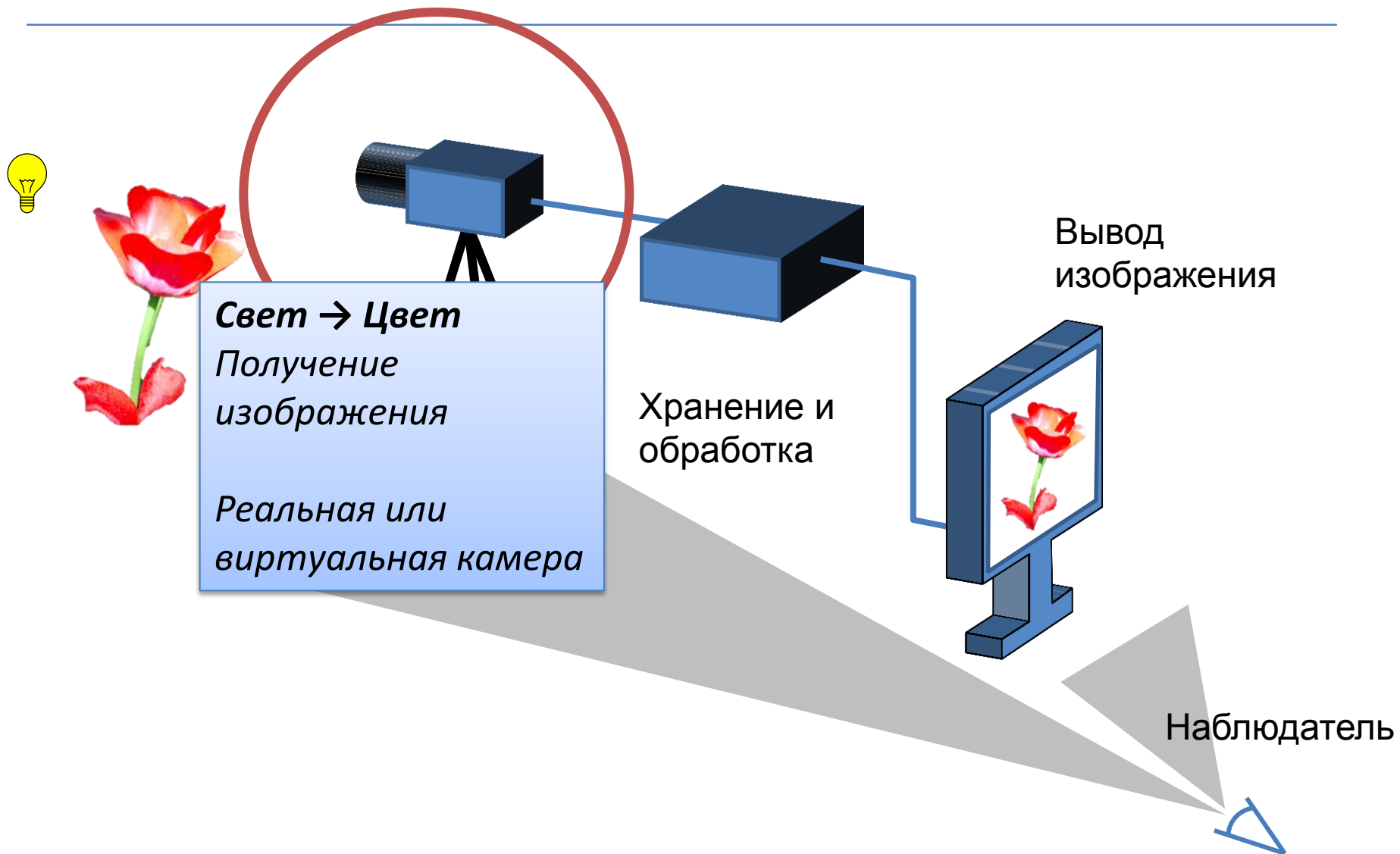
- **Цвет:** воспринимаемый результат воздействия света видимой части спектра
- Нет наблюдателя – нет восприятие, нет и цвета!
- Каждому спектру соответствует цвет
- Для цвета можно найти несколько спектров (метамеризм)

# Подбор цветов

---

- Не нужно моделировать произвольный спектр
- **Трех** чисел достаточно для описания цвета
- Нужно разработать принцип численного (количественного) представления цвета

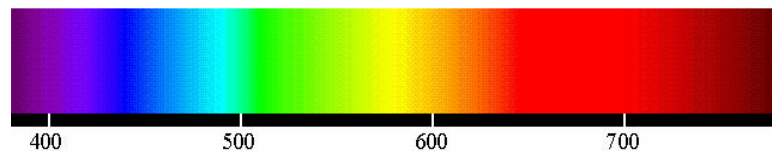
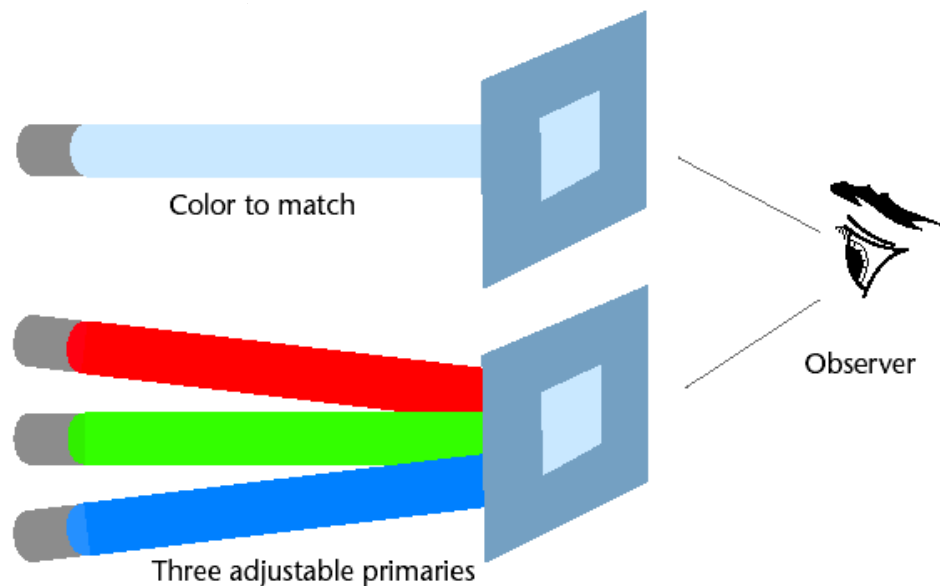
# Свет и цвет в графической системе



# Эксперименты по подбору цветов

- 1920е-1930е
- Экран размером 2 градуса
- Монохроматический исходный цвет
- Три источника света – основные цвета R, G, B (монохроматические) создают целевой цвет
- Наблюдатель может менять интенсивность каждого источника

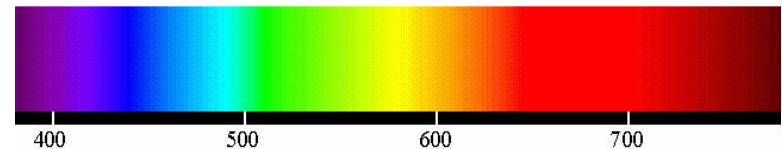
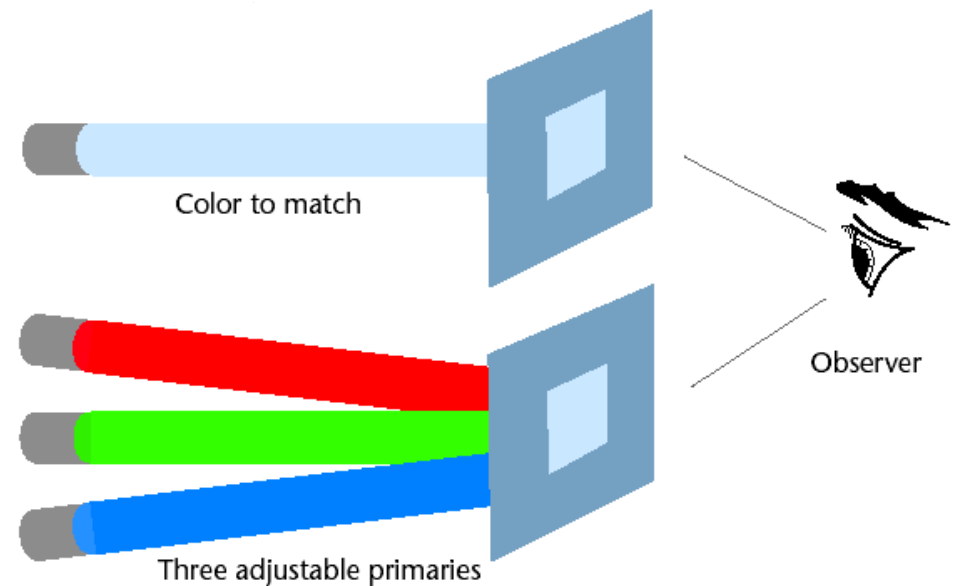
Монохроматический цвет  
380-780нм с шагом 5нм



# Эксперименты по подбору цветов (2)

- Большую часть цветов можно задать как сумму:  
 $C = rR + gG + bB$  (аддитивное соответствие)
- Некоторые цвета нельзя задать таким способом, вместо этого:  
 $C + rR = gG + bB$  (разностное сопоставление)
  - Создает проблемы для устройств вывода – нельзя создать лампу, которая забирает энергию
  - Позволяет использовать любые разные базовые света

Монохроматический цвет  
380-780нм с шагом 5нм





# Эксперименты по подбору цветов: проблемы

---

Результаты верны только для

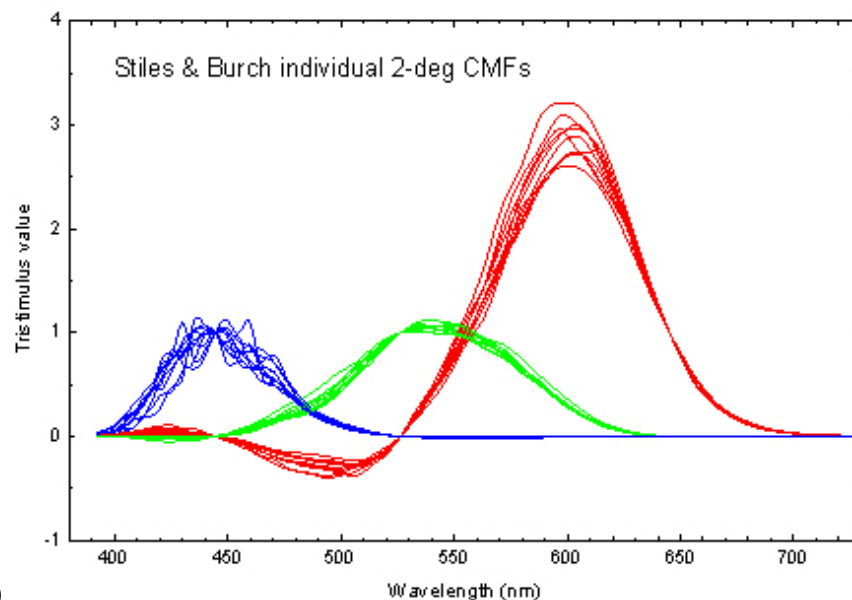
- конкретного наблюдателя
- для данных основных цветов (ламп)
- для монохроматических целевых цветов

Для практического использования необходимо  
расширить их

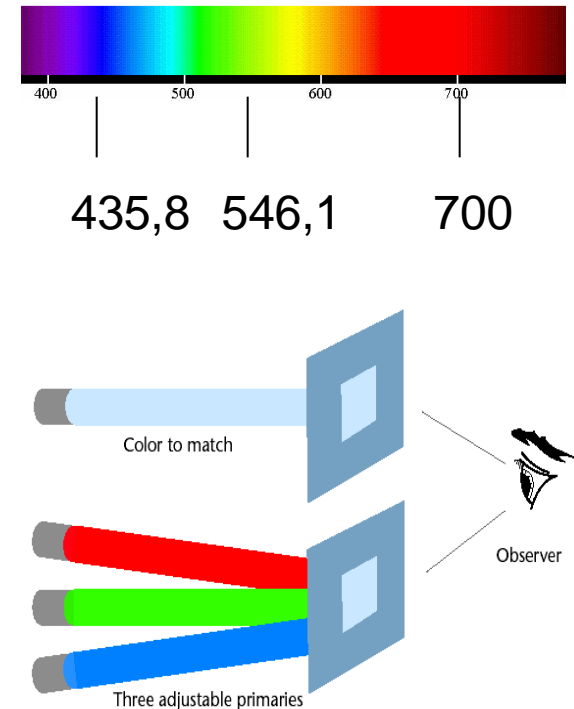
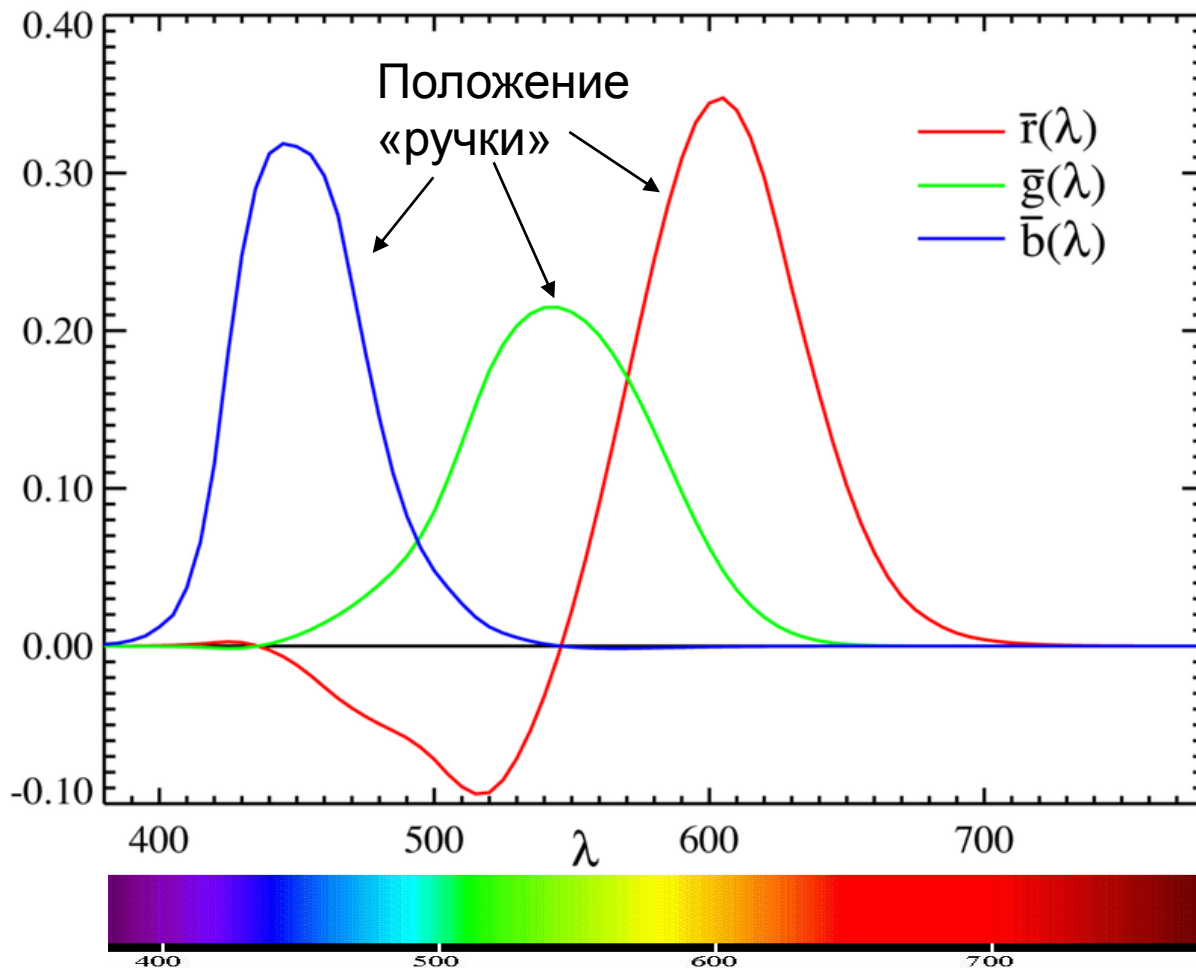
- На более широкий класс наблюдателей
- На более широкий класс базовых цветов
- На более широкий класс целевых цветов

# Эксперименты CIE (МКО) 1931г

- Эксперименты по воспринимаемому соответствию цветов были проведены на большом количестве людей
- Для людей с нормальным цветовосприятием результаты оказались достаточно близки
  - их можно усреднить
- В 1931 году на их основе МКО стандартизовал понятие стандартного наблюдателя
- Вывод: результаты экспериментов по соответствию цветов для стандартного наблюдателя могут быть применены к любому человеку с нормальным зрением



# Эксперименты по подбору цветов: результаты CIE



# Можно ли найти тройку чисел для любого спектра?

---

Мы уже знаем:

- Любое излучение – сумма монохроматических излучений разной интенсивности (амплитуды волны)
- Любой цвет может быть описан тройкой чисел
- Как представить монохроматические цвета с помощью тройки чисел (из экспериментов CIE) для данных базовых цветов

Возможно ли на основе этой информации найти тройки чисел для любого цвета?

Да! Закон аддитивности Грассмана

# Закон аддитивности Грассмана

---

- Эмпирический закон о линейности человеческого зрения (Hermann Grassman)
- Аддитивность:
  - Если наблюдатель задаст цвет лучей 1 и 2 как  $R_1B_1G_1$  и  $R_2B_2G_2$  относительно заданных основных цветов
  - То цвет их комбинации цвет будет равен
$$R = R_1 + R_2$$
$$G = G_1 + G_2$$
$$B = B_1 + B_2$$
- Верность для любого уровня интенсивности
  - $kC_1 = kC_2$ , если  $C_1 = C_2$

# Закон аддитивности Грассмана (2)

---

- Позволяет использовать конечный набор соответствий цветов для моделирования бесконечного набора
- Любое спектральное распределение может быть задано как взвешенная сумма монохроматических источников =>
- Если задать RGB-соответствия для этих цветов, то RGB для любого спектрального цвета будет взвешенной суммой RGB монохроматических цветов

$$R = \int_{380}^{780} C(\lambda) r(\lambda) d\lambda$$

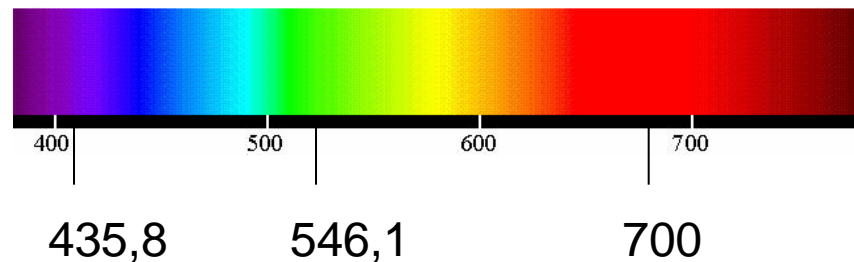
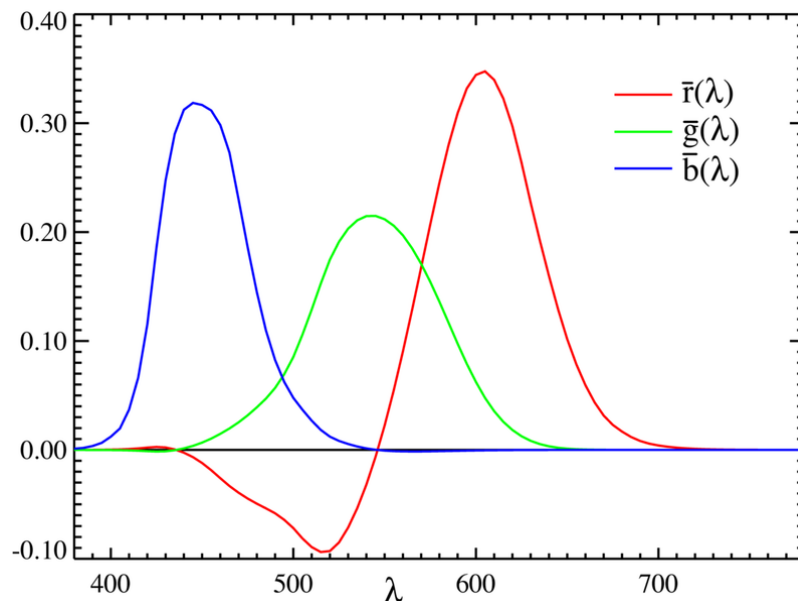
$$G = \int_{380}^{780} C(\lambda) g(\lambda) d\lambda$$

$$B = \int_{380}^{780} C(\lambda) b(\lambda) d\lambda$$

# Пространство CIE RGB 1931

Кривые  $r(\lambda)$ ,  $g(\lambda)$ ,  $b(\lambda)$  и спецификация базовых источников света задают трехмерное аддитивное пространство **CIE RGB 1931**

- Мы умеем для любого спектра находить точку в этом пространстве
- Не все точки пространства соответствуют видимым спектрам
  - Невидимые
  - Отрицательные спектры



# Переход между цветовыми пространствами

---

- Хотим создать другое цветовое пространство с источниками  $X(\lambda)$ ,  $Y(\lambda)$ ,  $Z(\lambda)$
- Пусть знаем координаты этих источников  $(r_1, g_1, b_1)$ ,  $(r_2, g_2, b_2)$ ,  $(r_3, g_3, b_3)$  в RGB
- Следовательно:

Комбинация новых источников с неизвестными коэффициентами

$$\begin{aligned} C &= xX + yY + zZ = \\ &= x(r_1R + g_1G + b_1B) + y(r_2R + g_2G + b_2B) + z(r_3R + g_3G + b_3B) = \\ &= \underbrace{(xr_1 + yr_2 + zr_3)}_r R + \underbrace{(xg_1 + yg_2 + zg_3)}_g G + \underbrace{(xb_1 + yb_2 + zb_3)}_b B \end{aligned}$$



## Переход между цветовыми пространствами (2)

---

$$\begin{pmatrix} r \\ g \\ b \end{pmatrix} = \begin{pmatrix} r_1 & r_2 & r_3 \\ g_1 & g_2 & g_3 \\ b_1 & b_2 & b_3 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

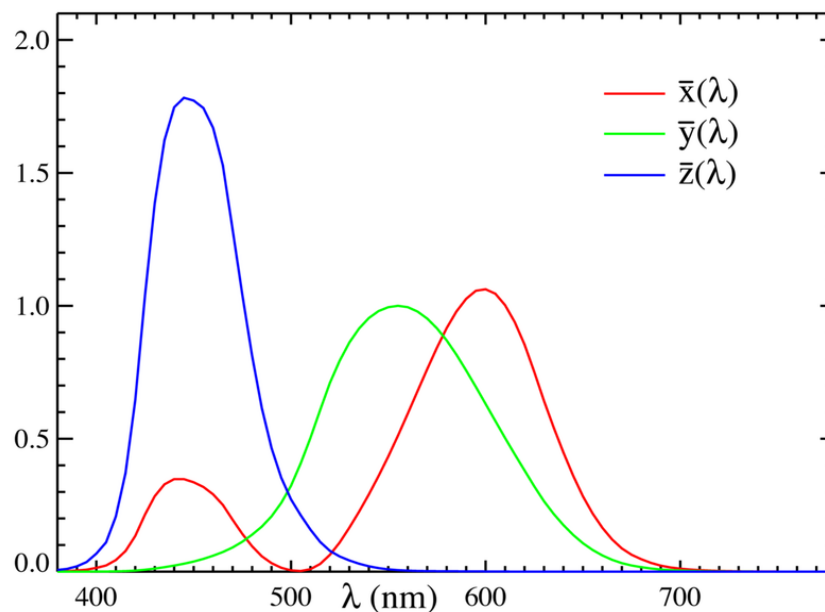
Переход между  
цветовыми  
пространствами –  
линейное  
преобразование

# Пространство CIE XYZ 1931

Задача: создать новое цветовое пространство XYZ, более удобное в работе, чем CIE RGB

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \frac{1}{0.17697} \begin{bmatrix} 0.49 & 0.31 & 0.20 \\ 0.17697 & 0.81240 & 0.01063 \\ 0.00 & 0.01 & 0.99 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

- Базовые цвета  $x(\lambda)$ ,  $y(\lambda)$ ,  $z(\lambda)$  всюду неотрицательны
- $y(\lambda)$  соответствует стандартной функции спектральной эффективности CIE
- Точка белого «равной энергии» должна соответствовать  $x=y=z=1/3$ 
  - «плоское» спектральное распределение



Названо CIE XYZ 1931

# Пространство CIE xy

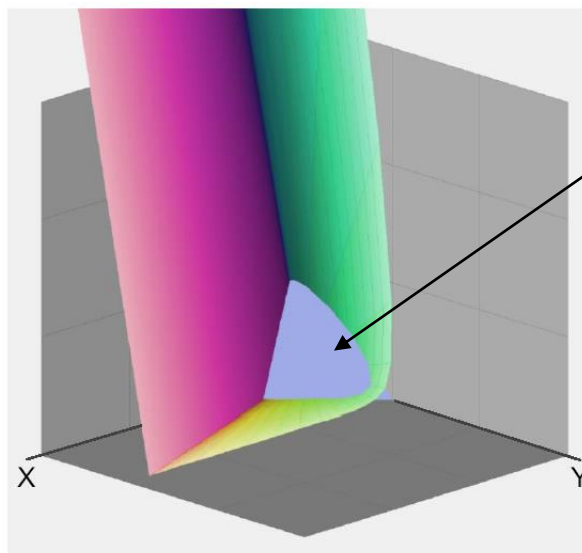
---

- Задача: разделить интенсивность и цвет
  - Вектора разной длины проецируются в одну точку
  - Прямые сохраняются

- Пространство CIE xy

$$x = \frac{X}{X + Y + Z}$$

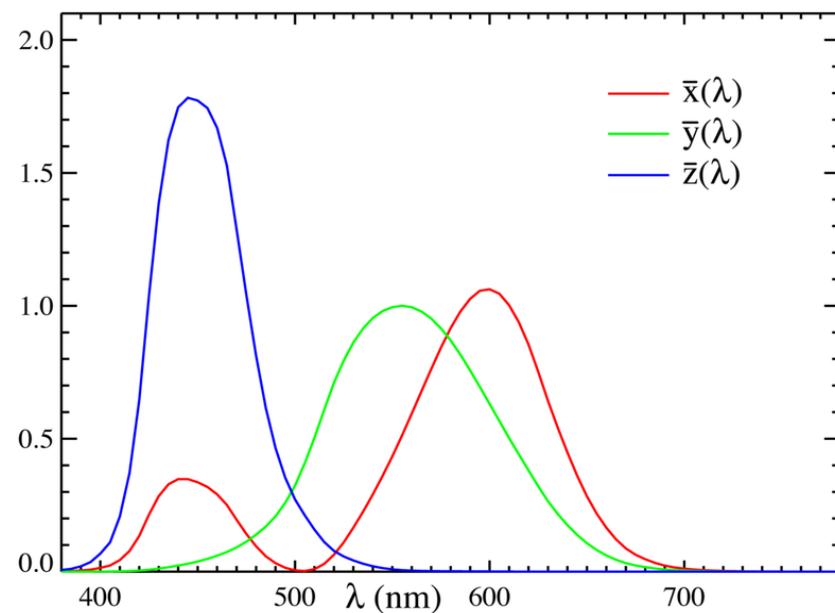
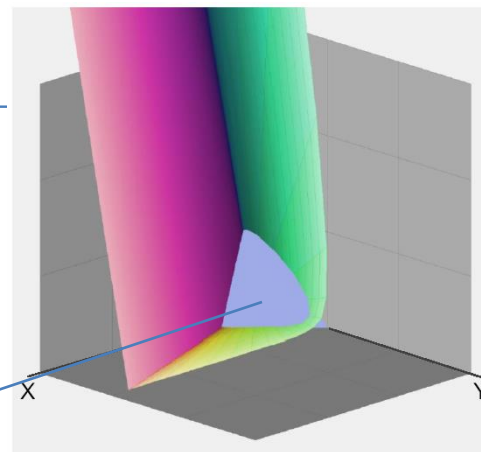
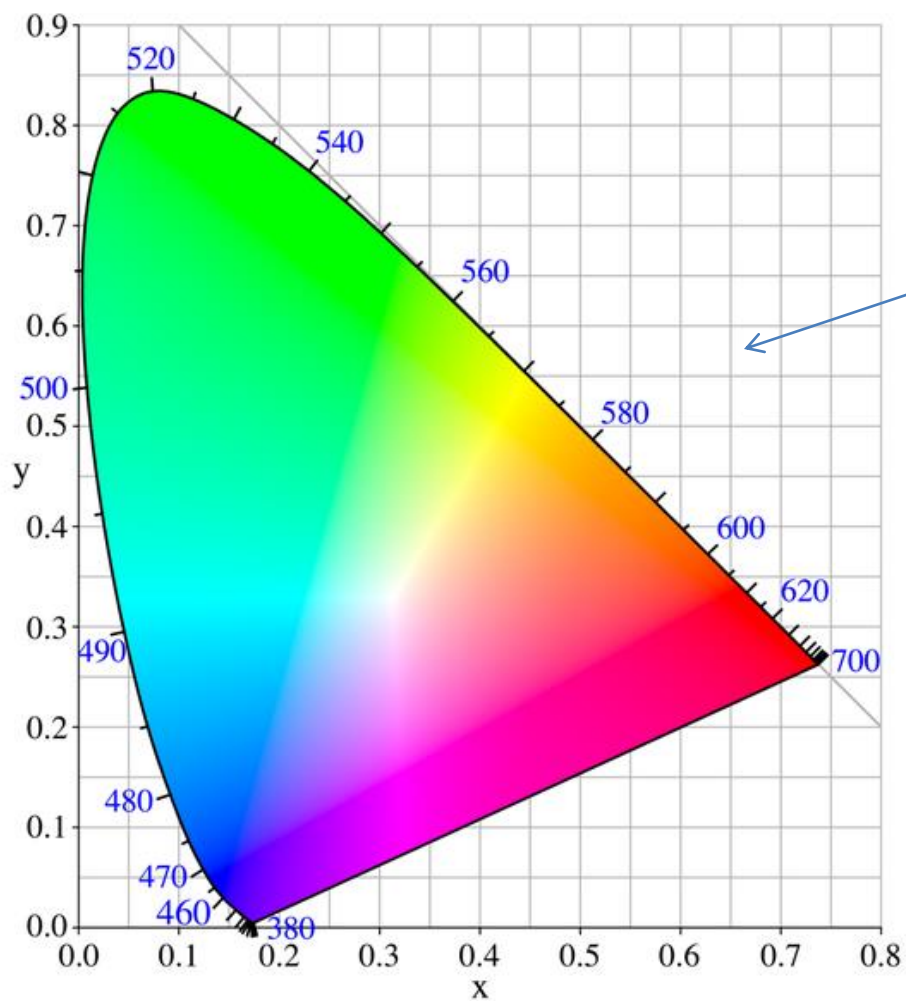
$$y = \frac{Y}{X + Y + Z}$$



Проекция на  
плоскость

$$X+Y+Z=1$$

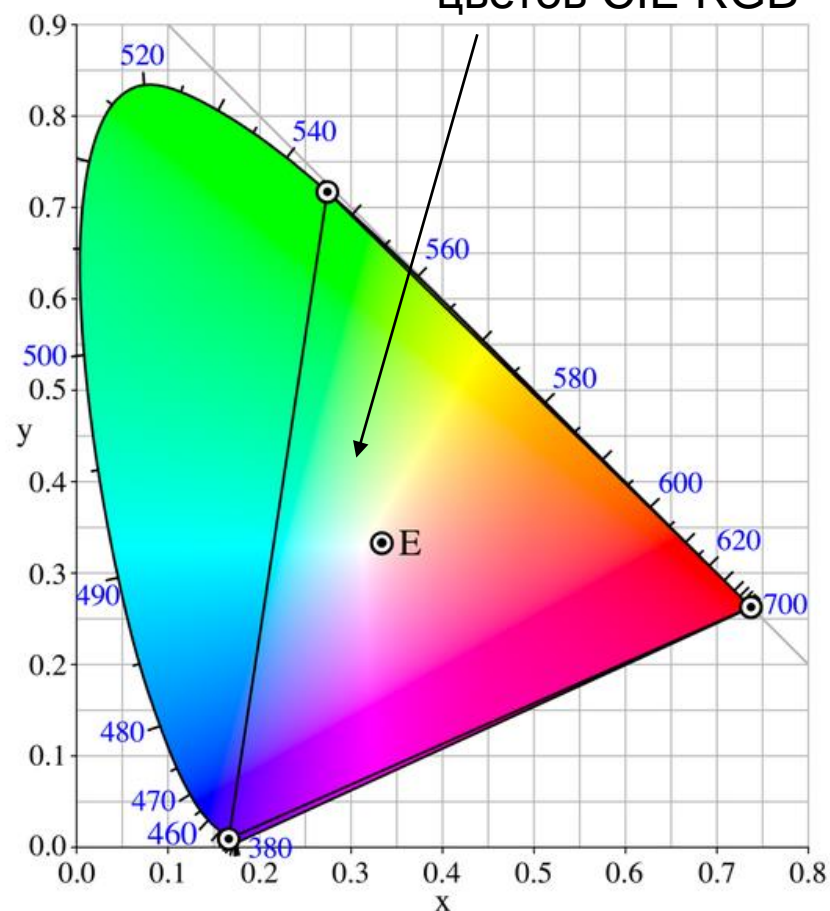
# Диаграмма цветности



# Свойства диаграммы цветности

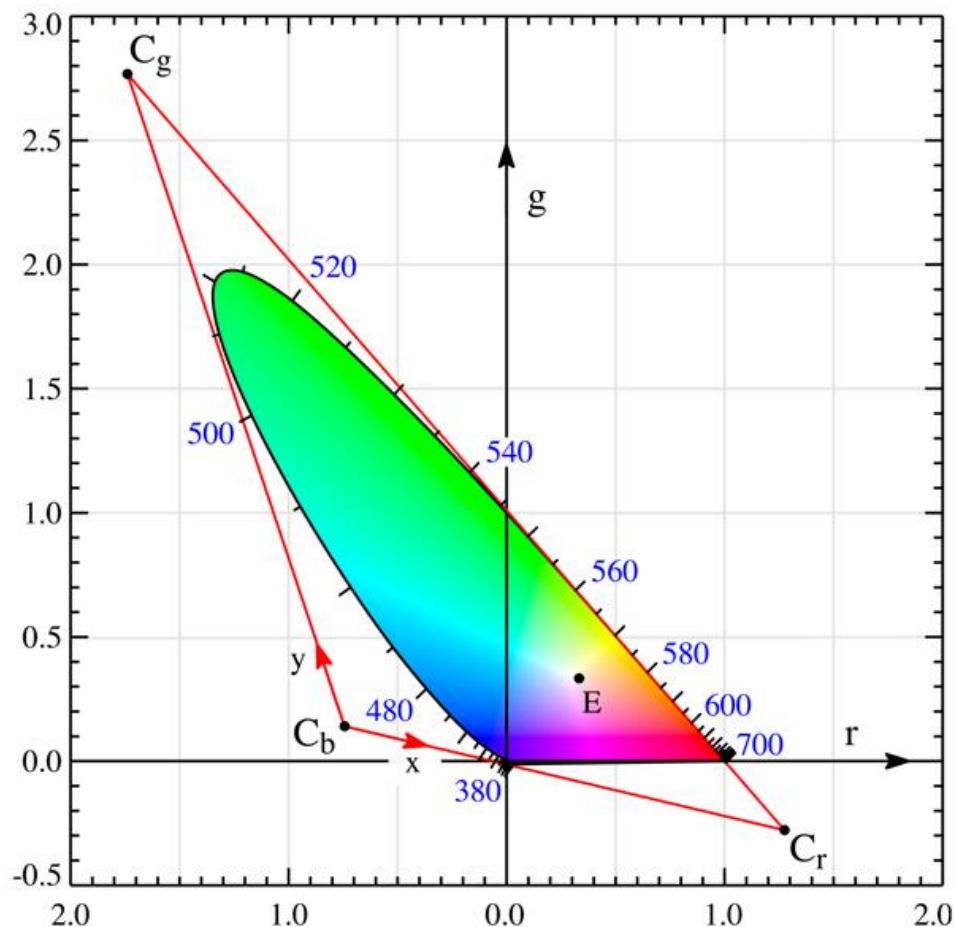
Область  
ОСНОВНЫХ  
ЦВЕТОВ CIE RGB

- На диаграмме представлены все цвета, видимые человеку
  - Внутри спектральной кривой
- Все цвета, которые могут быть получены смешением любых двух, лежат на прямой между ними
- Все цвета, которые могут быть получены смешением трех цветов, лежат внутри треугольника
  - Смешивая три данных реальных источника света, невозможно получить все цвета, видимые человеком



# CIE XYZ на диаграмме цветности rg

- Все видимые цвета попадают внутрь CIE XYZ
- Но базовые источники невозможно физически воспроизвести (не имеют цвета)
  - супернасыщенные



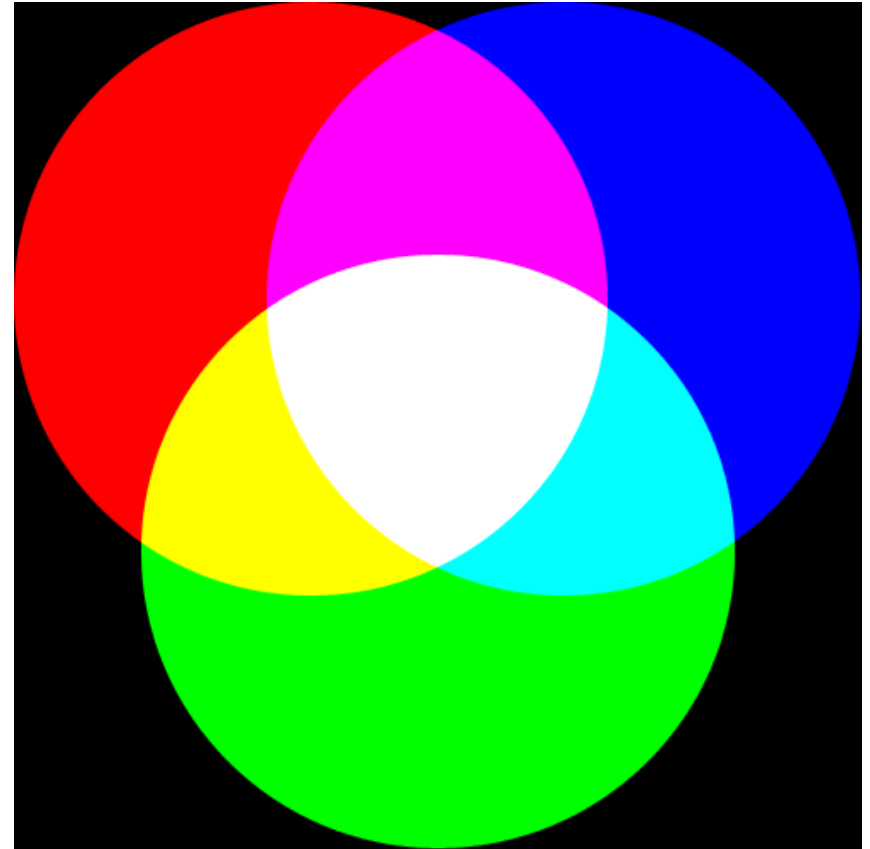
# Цветовые пространства и модели

---

- Цветовая модель – абстрактная математическая модель описания цвета набором чисел (обычно тремя)
  - Не имеет функции отображения в абсолютное цветовое пространство
  - Нельзя использовать в прикладных задачах без привязки к абсолютному пространству
- Цветовое пространство = модель + отображение в некоторое исходное (reference) пространство
  - Цвета не зависят от внешних факторов

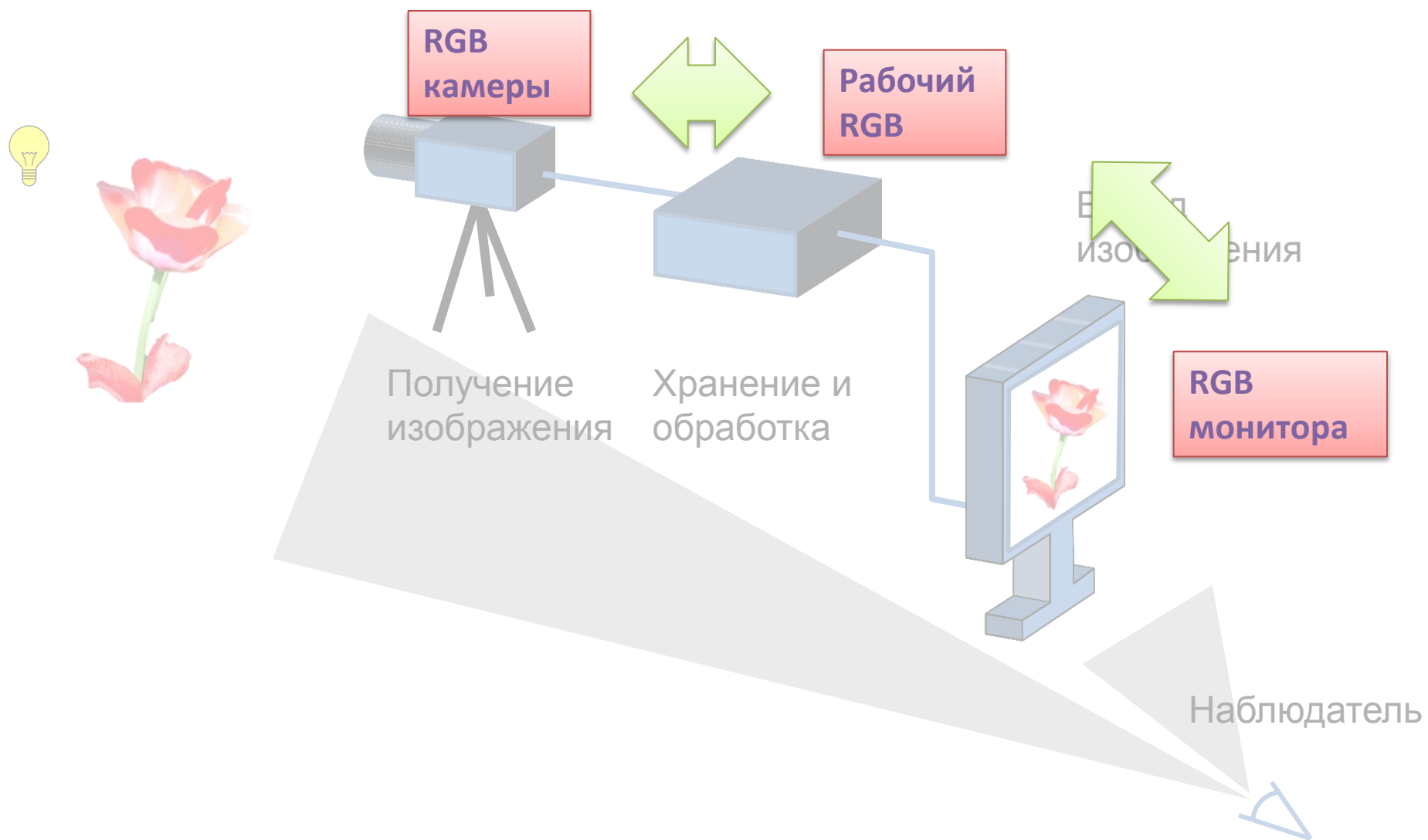
# Модель RGB

- Основана на аддитивной комбинации трех основных цветов – красного (**R**ed), зеленого (**G**reen), синего (**B**lue)
- Описывает системы, основанные на испускании света для получения нужного цвета (телевизоры, мониторы)
- Сами по себе значения  $r, g, b$  не несут физического смысла
  - Нужна привязка к исходному цветовому пространству
- Наиболее часто применяется в компьютерной графике, т.к. компьютерная графика работает с изображениями на мониторе





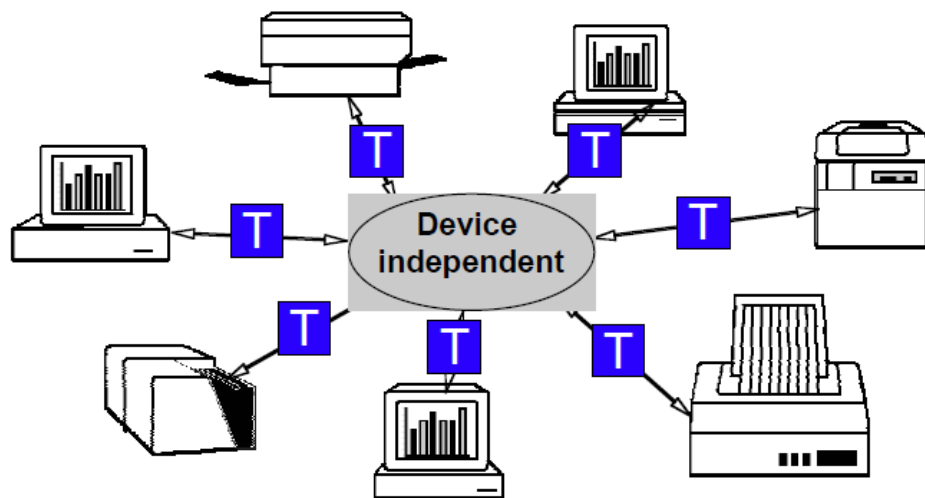
# Свет и цвет в графической системе



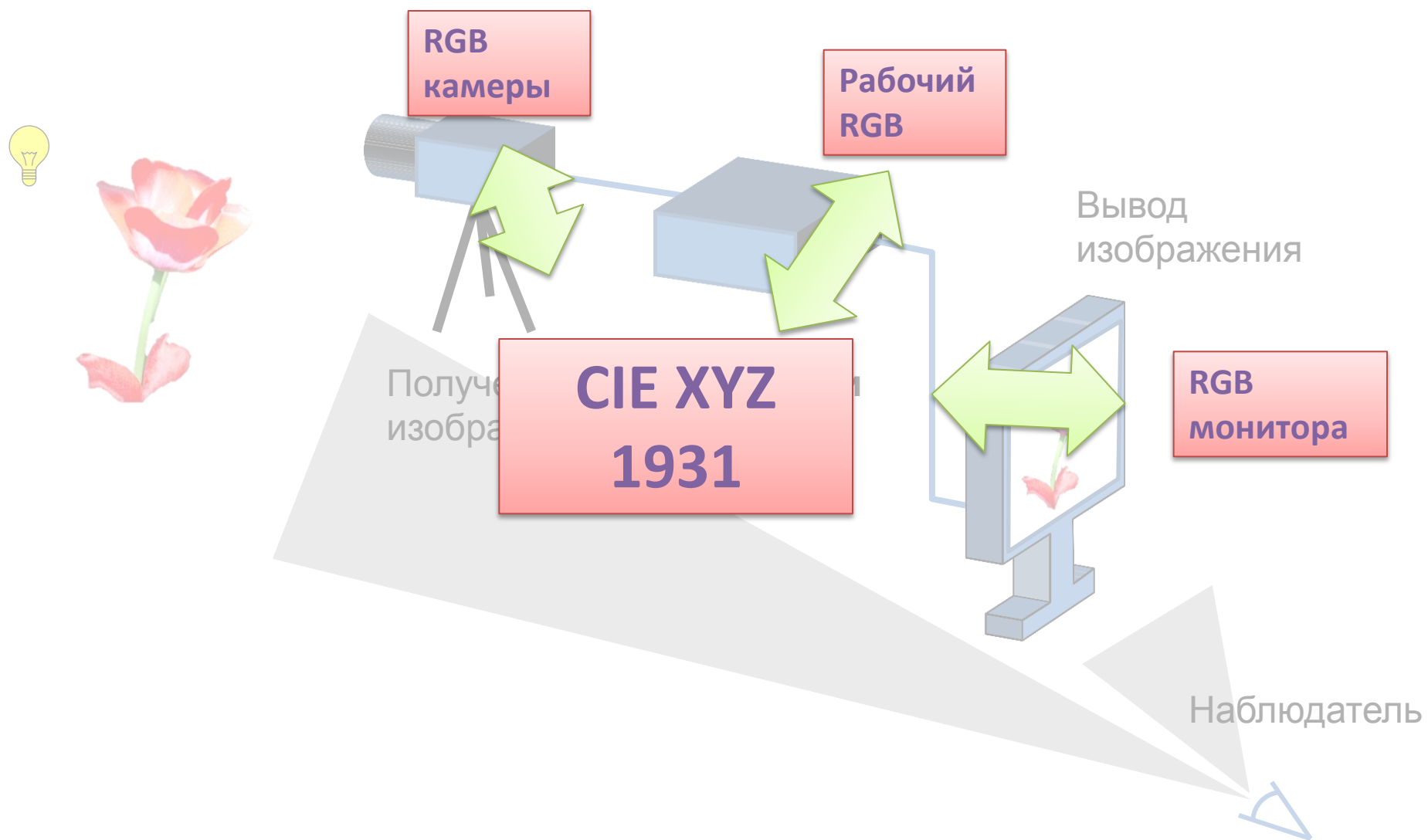
# Спецификация RGB пространств

---

- На практике существует множество RGB-пространств
  - Пространства конкретных устройств
  - Стандартизированные «рабочие» пространства для устройств
- Пространство можно задать матрицей **RGB->XYZ**
- Однако обычно используются ху-цвета базовых источников и соотношение их яркостей (точка белого)

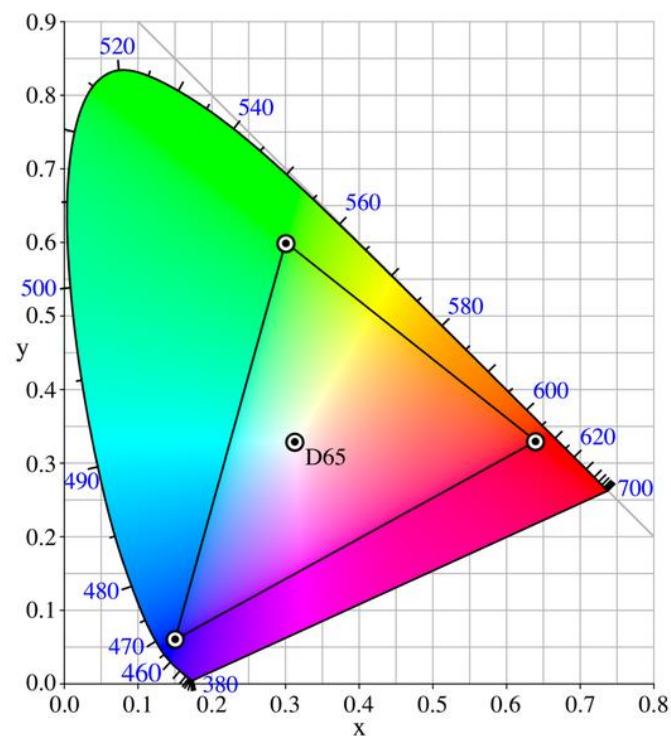


# Свет и цвет в графической системе



# Спецификация RGB базовых источников

- Три фосфора задают аддитивное цветовое пространство
- Для полной спецификации обычно задают
  - $x$ -координаты для  $r, g, b$ -фосфоров
  - точку белого (относительная яркость)
- Примеры пространств:
  - NTSC RGB (телевизоры)
  - HDTV RGB (телевизоры)
  - sRGB (мониторы)
- При передаче сигнала (например, телевизионного) цвет кодируется в предположении о соответствии фосфоров монитора (телевизора) стандарту
  - Если не соответствуют, но монитор должен включать в себя коррекцию (аппаратную или программную)



Пространство sRGB  
(основные цвета и точка  
белого)

# Спецификация RGB элементов: точка белого

---

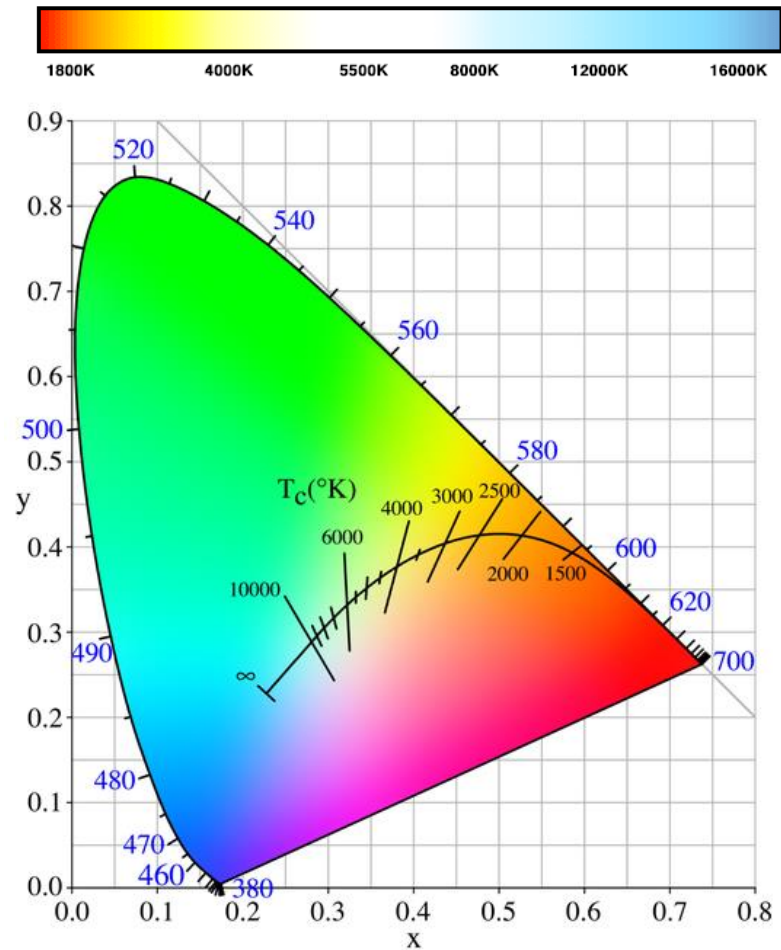
- Точка белого – цвет, который считается белым в данных условиях
- Для монитора – цвет, который испускают фосфоры с максимальной яркостью (1,1,1)
  - Фактически задает относительные яркости фосфоров
- Существуют стандартные точки белого
  - CIE common white points

# Точка белого: некоторые стандартные точки белого

Имя	CIE 1931		CCT	Прим
	x	y		
E	1/3	1/3	5400	Точка равной энергии
D55	0.33242	0.34743	5500	
D65	0.31271	0.32902	6500	TV, sRGB
D75	0.29902	0.31485	7500	
A	0.44757	0.40745	2856	Лампа накаливания

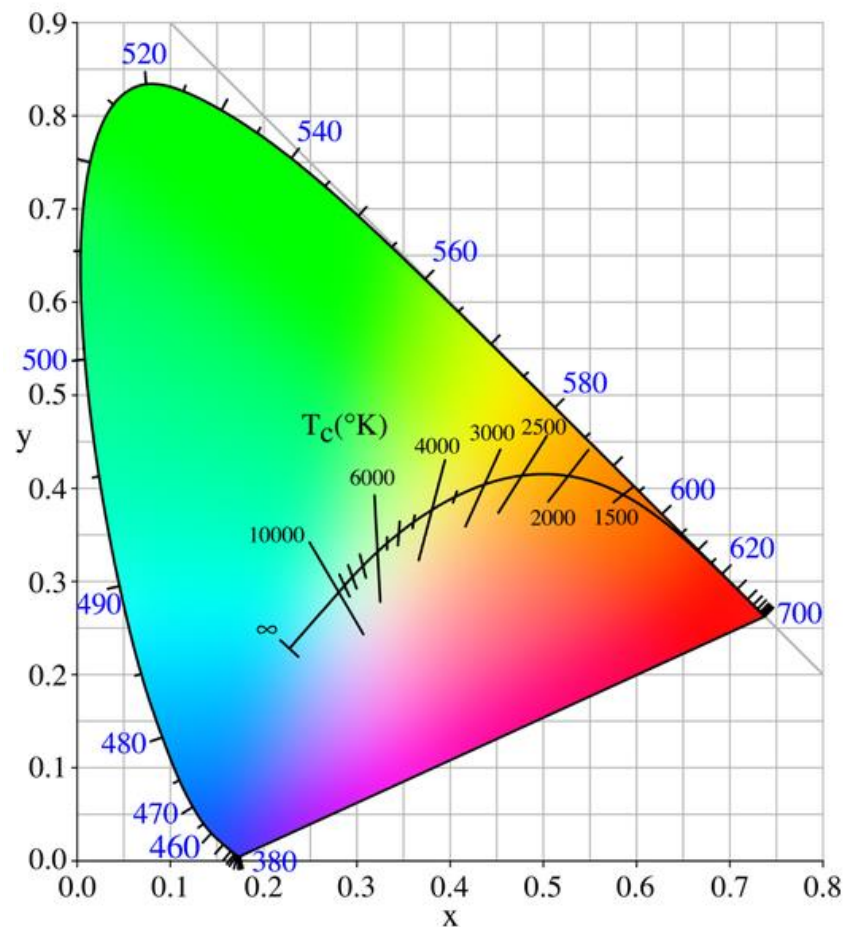
# Цветовая температура

- Цветовая температура – характеристика видимого света
- Сравнение цвета с цветом нагретого черного тела (black body radiator)
- Большинство источников света построены на излучении нагретого тела, поэтому их удобно описывать с помощью цветовой температуры
  - Можно сопоставить с реальным освещением



# Цветовая температура: примеры

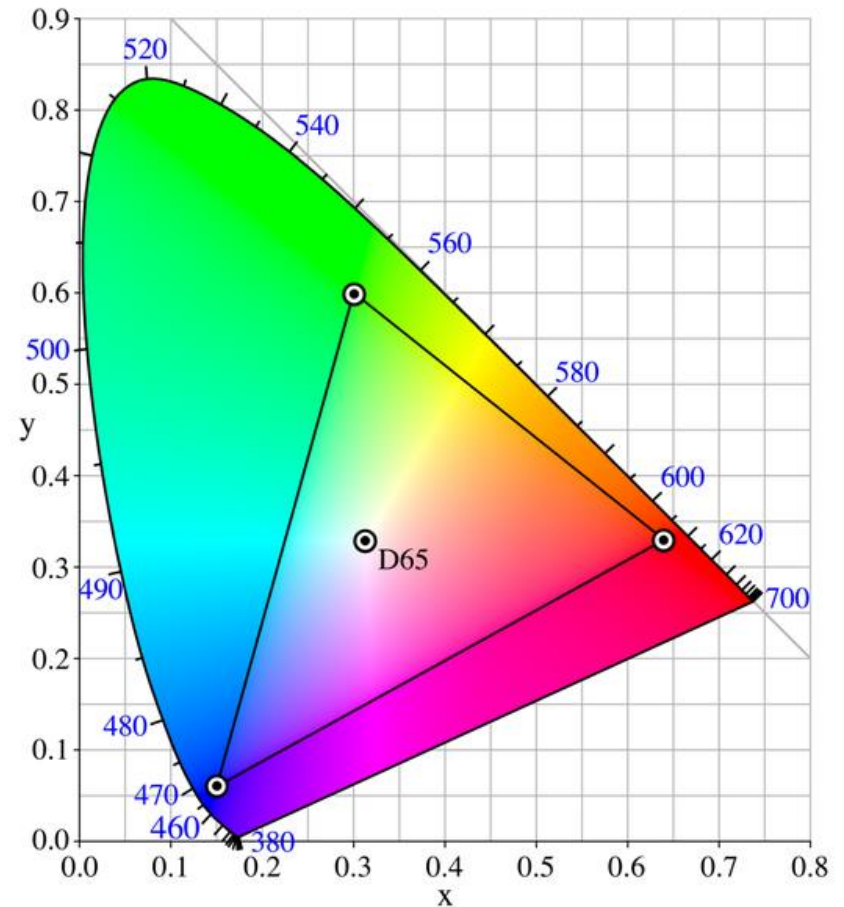
- 1600 К: восход и закат
- 1800 К: свеча
- 2800 К: лампа накаливания
- 3200 К: студийные лампы
- 5200 К: яркое полуденное солнце
- 5500 К: усредненный дневной свет
- 6000 К: облачное небо
- 20000 К: ярко-синее чистое небо
- 28000 - 30000 К: молния





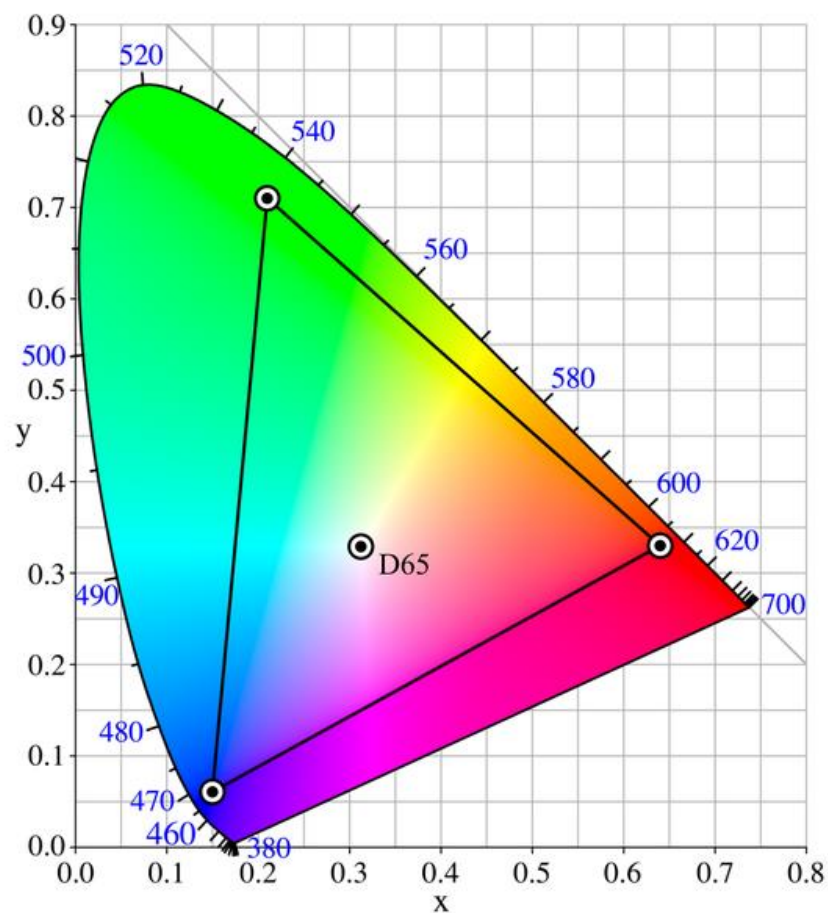
# Пространство sRGB

- Создано Microsoft, Hewlett-Packard
- Стандартизировано в 1996г.
- На данный момент широко используется:
  - Мониторы
  - Фотоаппараты
- Если для изображения не указано цветовое пространство, можно считать, что это sRGB
- Недостатки: исходные цвета сильно внутри видимой человеком области



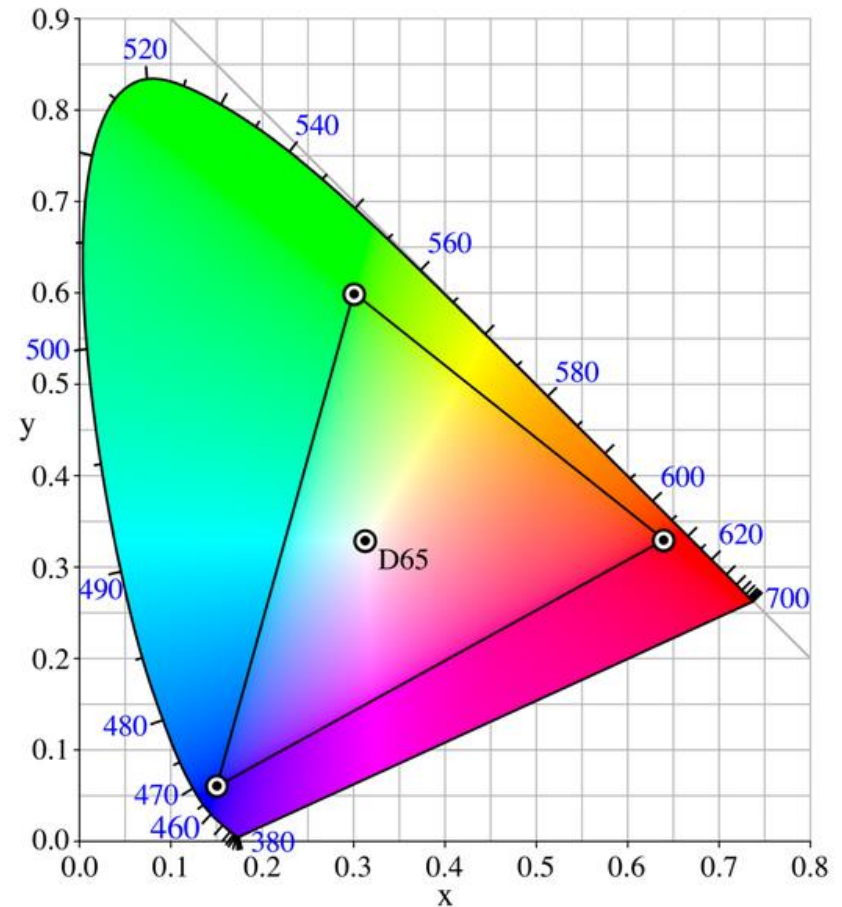
# Пространство Adobe RGB

- Разработано Adobe в 1998
- Цель – иметь возможность работать на мониторе с большинством цветов, доступных в модели CMYK на принтерах
- Более широкий диапазон передаваемых цветов (gamut)
- Проблема: 8 бит на цвет может не хватать



# Отображения передаваемых диапазонов цветовых пространств

- Цветовые пространства имеют разные диапазоны передаваемых цветов (gamut)
- Например, не все цвета изображения с профилем Adobe RGB могут быть показаны на мониторе с фосфорами sRGB
- Нужно преобразовать исходное изображение таким образом, чтобы все его цвета попадали в передаваемый диапазон устройства
- Процесс называется отображением передаваемого диапазона (gamut mapping)
- Два типа непередаваемых цветов
  - Невозможна коррекция цветности ( $l < 0$ )
  - Возможна коррекция цветности, но невозможна коррекция интенсивности ( $l > 1$ )



# Отображения передаваемых диапазонов цветовых пространств: подходы

---

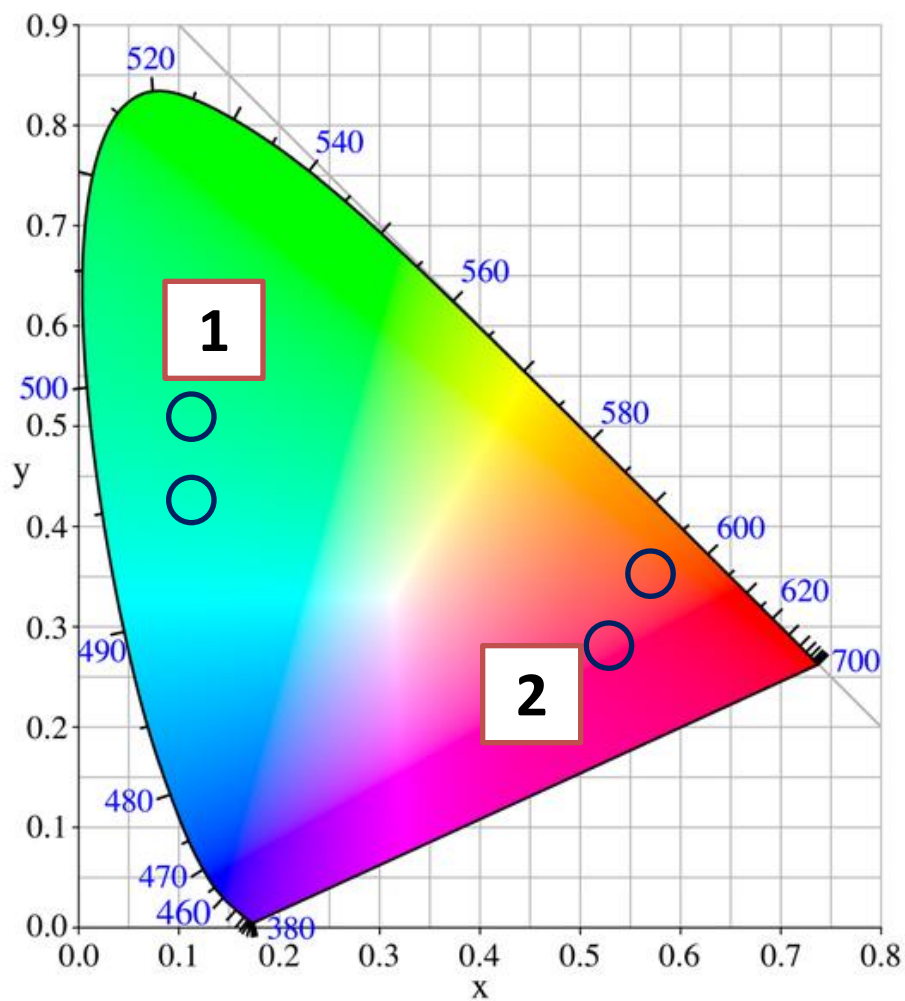
- Применяется после применения преобразования в целевое пространство
- Локальные и глобальные подходы
- Примеры локальных
  - Масштабирование цвета пикселя до попадания в диапазон
  - Отсечение по  $[0,1]$
  - ...
- Пример глобального подхода:
  - Поиск наименьших и наибольших компонент цвета и масштабирование цветов всего изображения для попадания в диапазон

# Однородные и интуитивные цветовые пространства

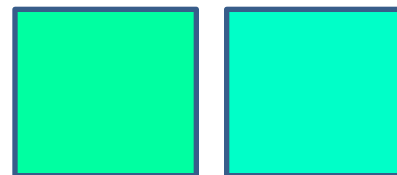
---

- Пространства XYZ и RGB недостаточно интуитивно
  - Нет осмысленных значений у компонент X,Z (Y означает светимость)
  - XYZ и RGB нелинейны для восприятия
    - Изменение значений xyz не означает пропорциональное изменение цвета
- Было разработано несколько цветовых пространств, обладающих заданными свойствами

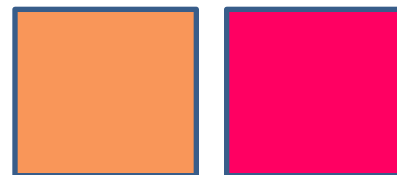
# Разница цветов и расстояние



1



2



# CIE 1976 L\*a\*b

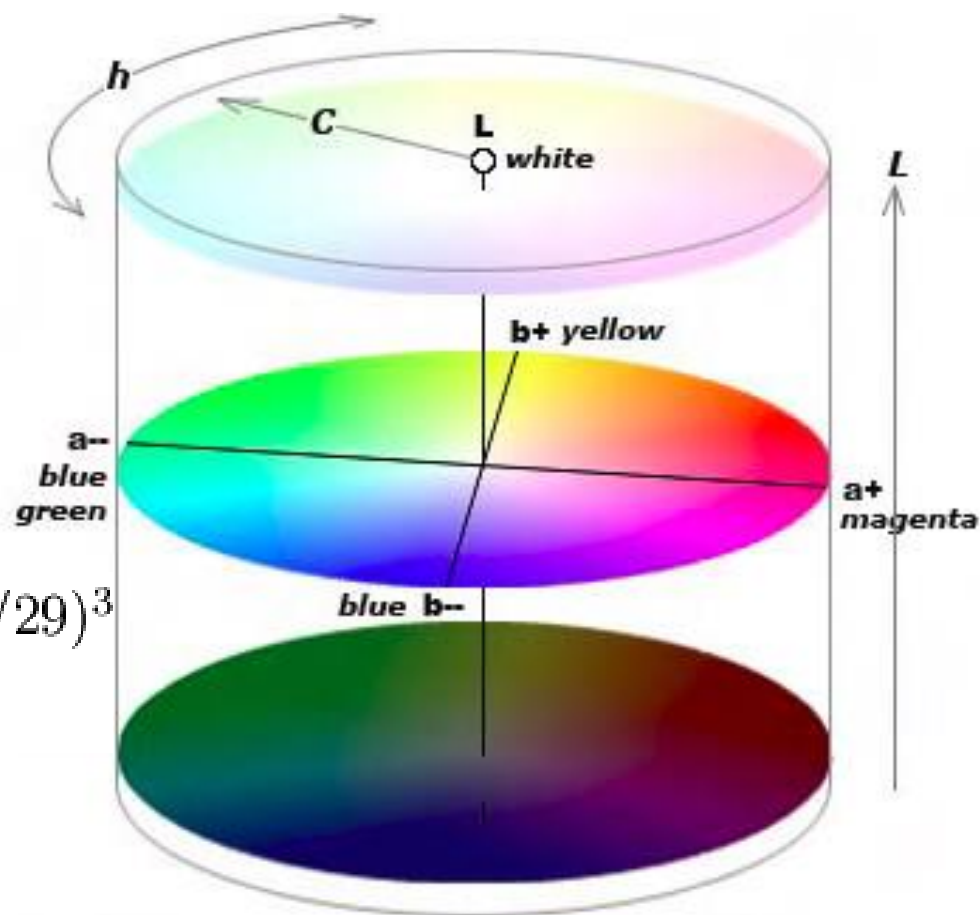
$$L^* = 116 f(Y/Y_n) - 16$$

$$a^* = 500 [f(X/X_n) - f(Y/Y_n)]$$

$$b^* = 200 [f(Y/Y_n) - f(Z/Z_n)]$$

$$f(t) = \begin{cases} t^{1/3}, & t > (6/29)^3 \\ \frac{1}{3} \left(\frac{29}{6}\right)^2 t + \frac{4}{29} & t \leq (6/29)^3 \end{cases}$$

$X_n, Y_n, Z_n$  – точка белого



# CIE 1976 L\*a\*b

---



$dE=5$



$dE=20$



$dE=80$



# Интуитивные цветовые модели: HSV

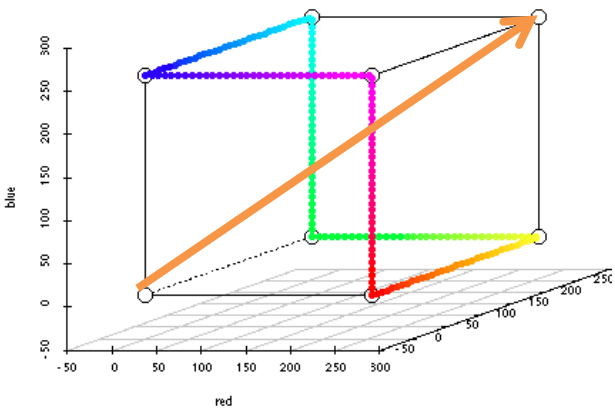
---

- RGB, CMY(К) ориентированы на работу с аппаратурой и неудобны для задания цвета человеком
- Субъективные атрибуты цвета
  - Цветовой тон
  - Насыщенность
  - Светлота
- Психофизические эквиваленты: доминирующая длина волны, чистота, яркость
- Модель HSV удобна для задания цвета человеком
  - Hue Saturation Value

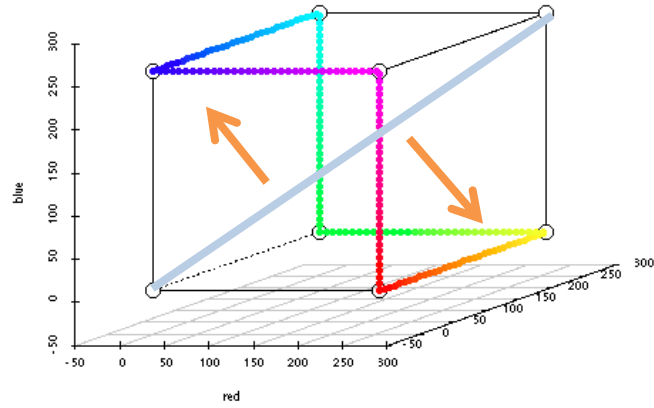
# HSV (2)

---

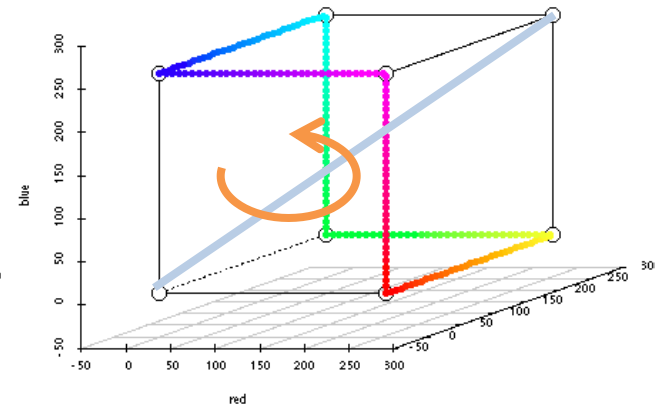
V



S

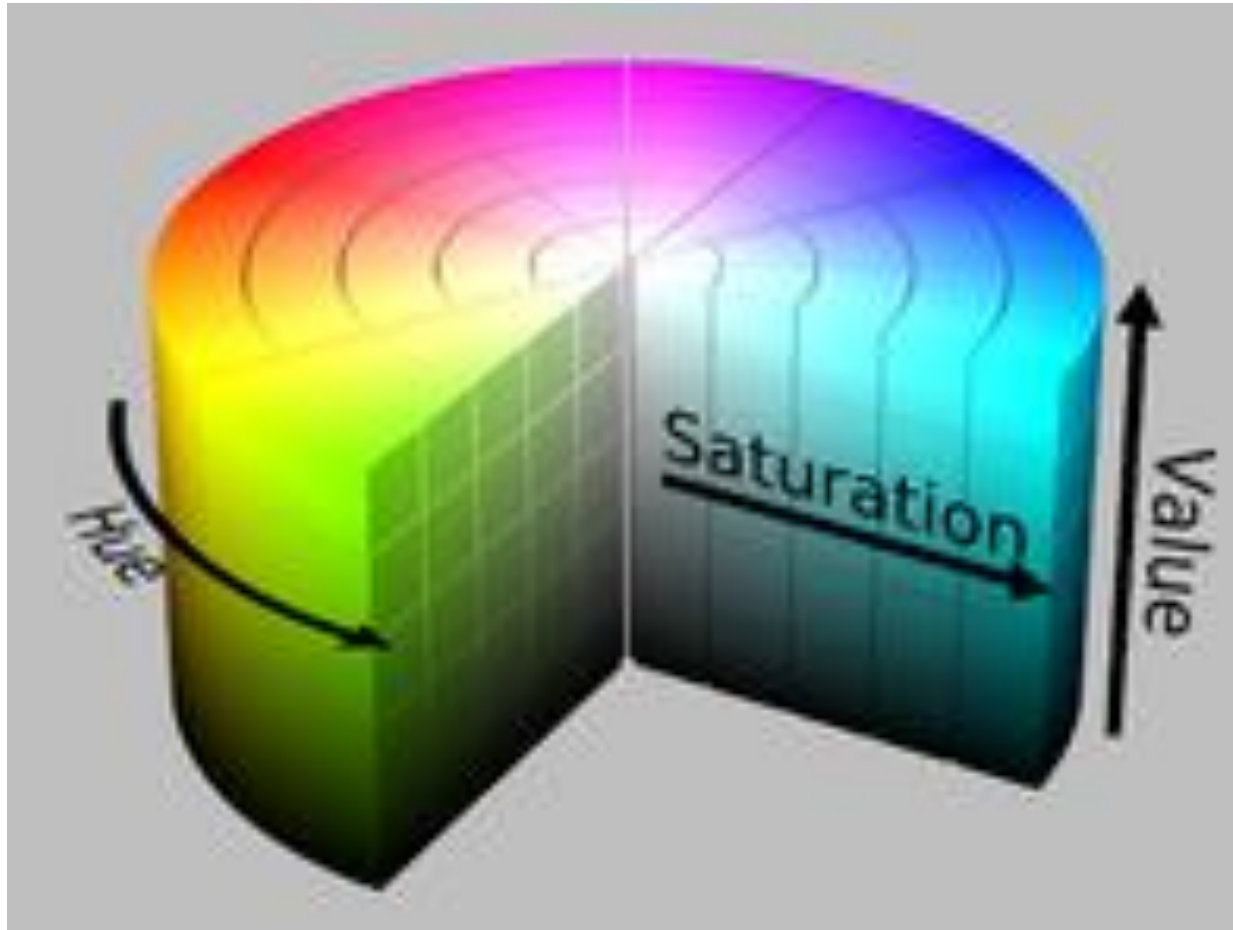


H



# HSV (3)

---



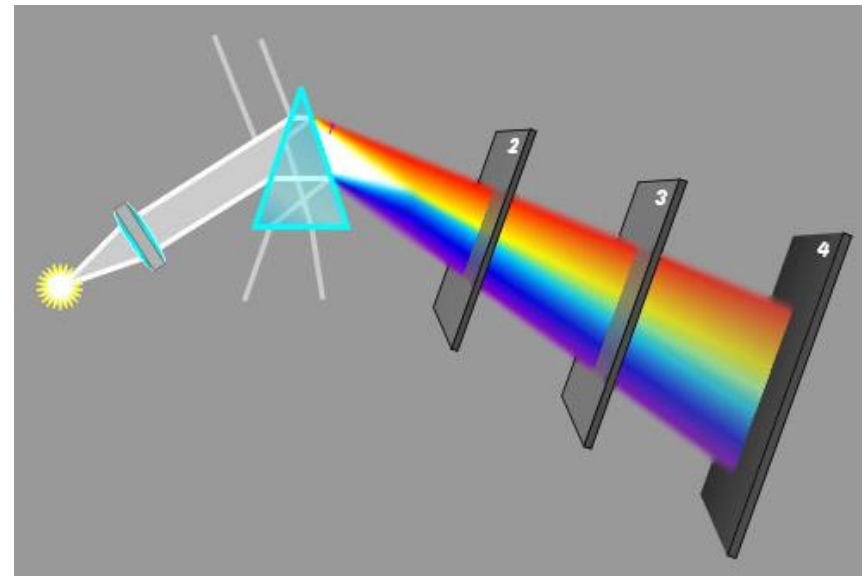
# Цветовые пространства

---

- Исходные (reference) цветовые пространства:
  - CIE XYZ
  - CIE L\*a\*b
  - CIE RGB (не используется)
- Цветовые модели:
  - RGB
  - CMYK
  - YIQ
  - HSV
  - HSL
- Производные цветовые пространства:
  - sRGB (RGB)
  - Adobe RGB (RGB)
  - Apple RGB (RGB)

# Ограничения трехцветных пространств

- Нельзя использовать при физических вычислениях, включающих явление дифракции, интерференции
  - Радуга
- Аддитивные пространства имеют достаточно узкий диапазон передачи цвета



# ИТОГИ

---

- Точка белого – применяется при описании реальных RGB-систем (и не только). Часто характеризуется цветовой температурой
- Цветовая температура – температура абсолютно черного тела, соответствующему данному цвету
- Хроматическая адаптация – процесс преобразования цветового пространства для учета адаптации человека к освещению
- Построение спектра по цвету – задача, которую приходится решать в системах фотореалистичного синтеза для получения спектров источников и спектров отражений

# Светлота (lightness)

---

- Человеческое зрение имеет нелинейный отклик на яркость.
- Источник света яркостью 18% по сравнению с базовым будет казаться вполовину менее ярким.
- Восприятие света человеком описывается светлотой источника
  - Яркость (luminance) описывает спектральную составляющую восприятия
  - Светлота описывает относительную мощностную характеристику восприятия

$$L^* = 116 \left( \frac{Y}{Y_n} \right)^{\frac{1}{3}} - 16; \quad 0.008856 < \frac{Y}{Y_n}$$

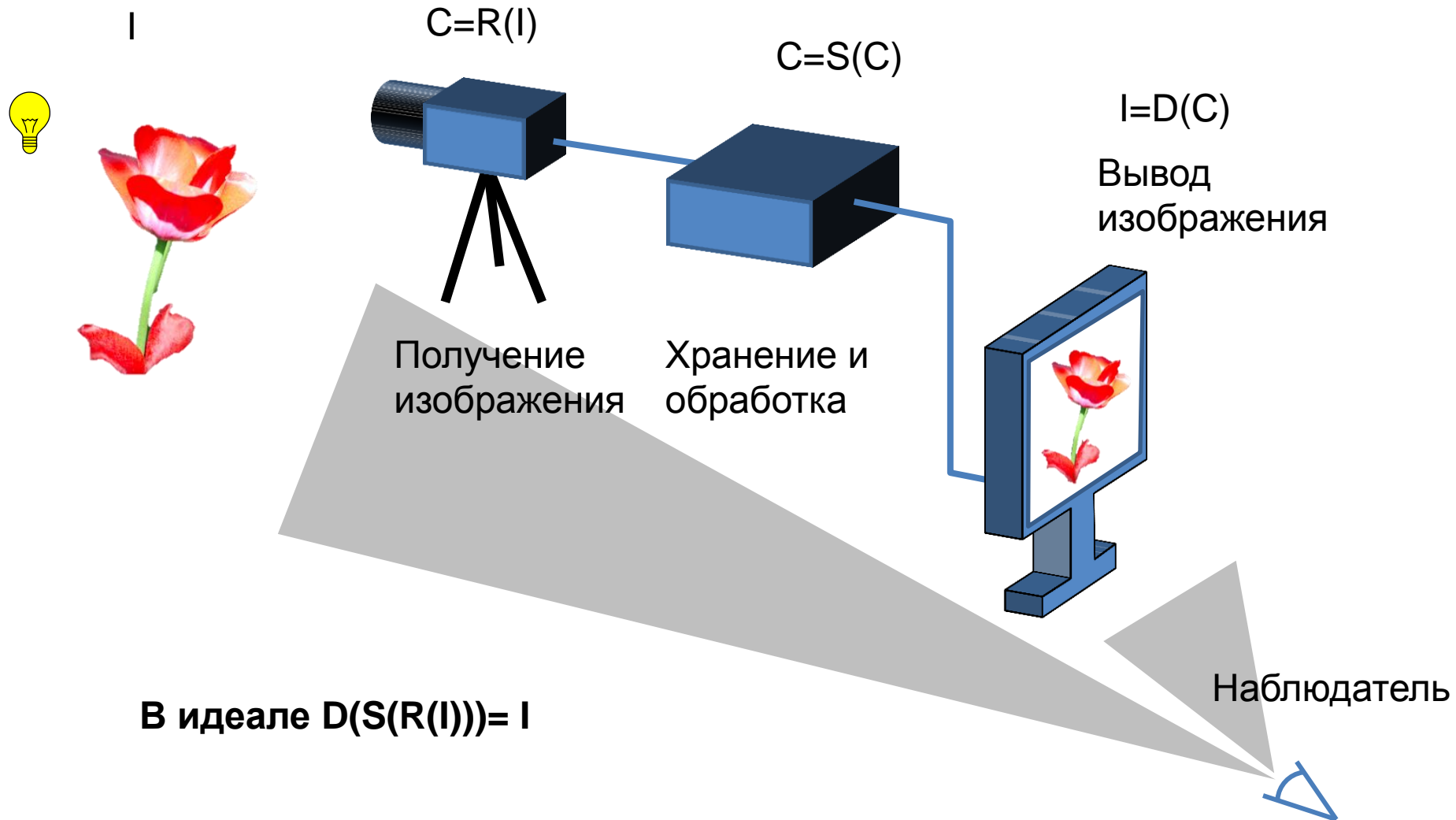
# Что хранит пиксель изображения?

---

- Если убрать цветовую составляющую
- Желательно кодировать энергетическую яркость (radiance)
- Ну или хотя бы световую яркость (luminance)
  - Легко получить с помощью преобразования в хуз и рассмотрения компоненты Y
- Вопрос в линейности!



# Тракт передачи изображений и передающие функции



# Проблемы с передающими функциями

---

- Особенности передающих функцией обусловлены физическим устройством приемника и дисплея
- При хранении информации в цифровом виде неизбежна дискретизация сигнала
  - Потеря информации!
- Передающие функции дисплея и камеры

# Почему необходимо корректировать яркость?

---

**Причина 1:** Нелинейность передающей функции CRT-дисплеев

**Причина 2:** Необходимость нелинейного кодирования яркости для более полного использования ограниченного диапазона представления яркости в ЭВМ

**Причина 3:** Особенности восприятия интенсивностей человеческим зрительным аппаратом

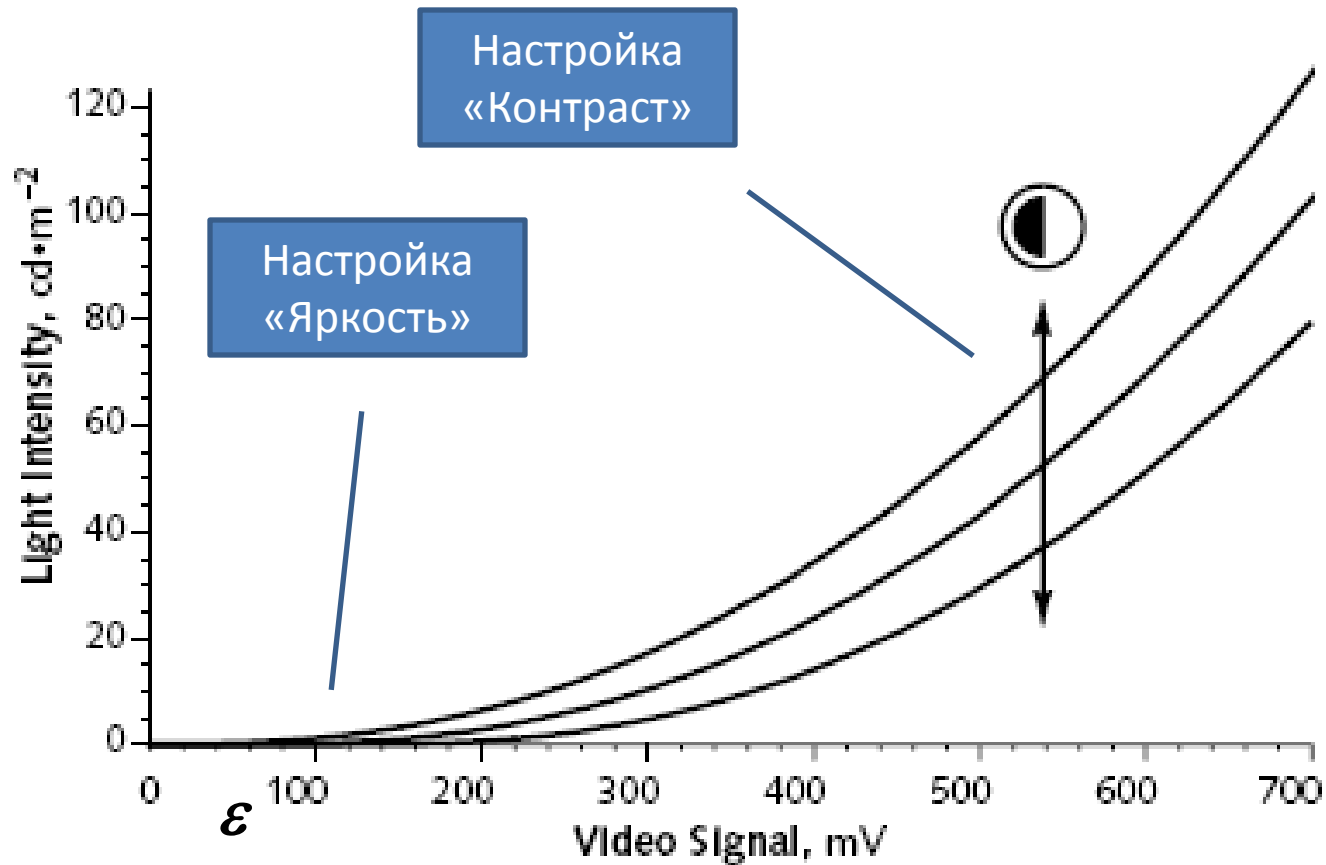
# Причина 1: Передающая функция монитора

---

- Интенсивность света, генерируемого физическим устройством не является линейной функцией входящего сигнала
- CRT-устройства (телевизоры, мониторы) имеют степенную зависимость интенсивности излучения от входящего напряжения:

$$D(C) = k(C + \varepsilon)^\gamma$$

# Причина 1: Передающая функция монитора



# Что такое гамма?

---

- Гамма – характеристика нелинейности интенсивности сигнала, выдаваемого физическим устройством вывода.
- Обычно 2.35 – 2.55
- Гамма-преобразование – нелинейное преобразование формы  $k(V)^\gamma$
- Гамма-коррекция – процесс компенсации нелинейного преобразования устройства вывода. Преобразование формы  $kV^{(1/\gamma)}$
- Гамма коррекция необходима для более точной передачи интенсивностей монитором
- Но не только!

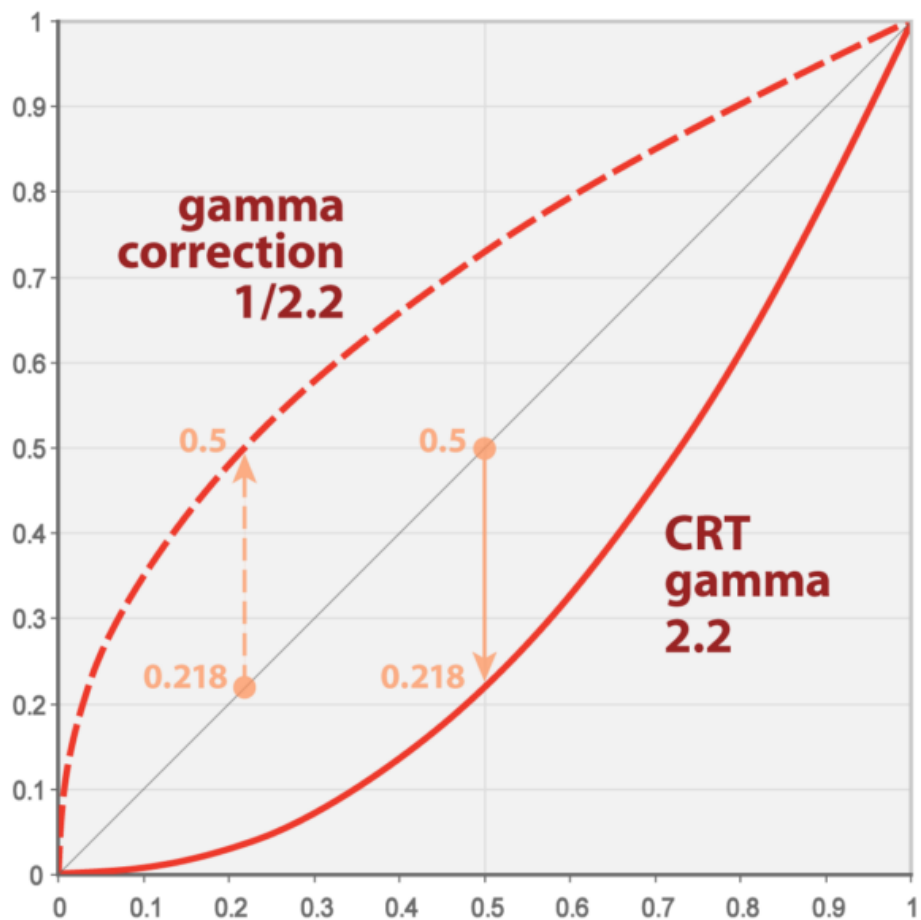
# Gamma compression & Gamma expansion

---

- Гамма-сжатие = гамма-коррекция
- Гамма-расширение = гамма-преобразование

# Функция гамма-коррекции

---



Применяется к относительным яркостям!



# CRT и LCD-мониторы

---

- LCD-мониторы аппаратно полностью линейны по передаче интенсивности
- Но ведут себя как CRT, делая гамма-преобразование перед выводом изображения
- Почему?
  - Для лучшего соответствия «логарифмическому» восприятию яркости человеком (см. далее)
  - Для совместимости

## Причина 2: Кодирование яркости с учетом восприятия яркости человеком

---

- Имеем на входе произвольный диапазон яркости
- Хотим закодировать его в компьютере с заданным количеством бит на пиксель (целочисленно)
- При различном кодировании шаг между соседними значениями цвета будет разным

**Какой должен быть шаг, чтобы при просмотре изображения человек видел плавные градации?**

## Причина 2: Кодирование яркости с учетом восприятия яркости человеком

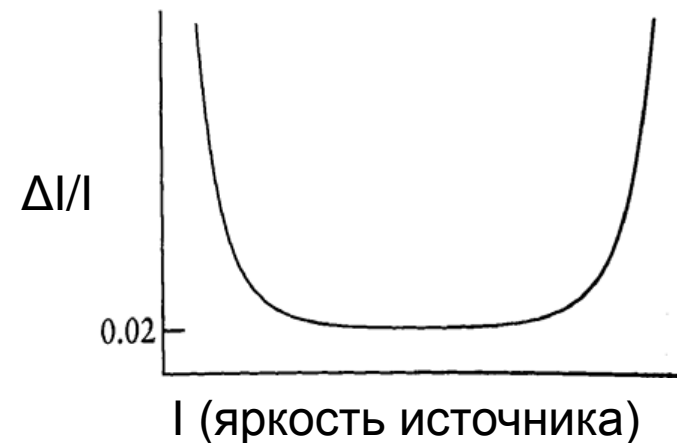
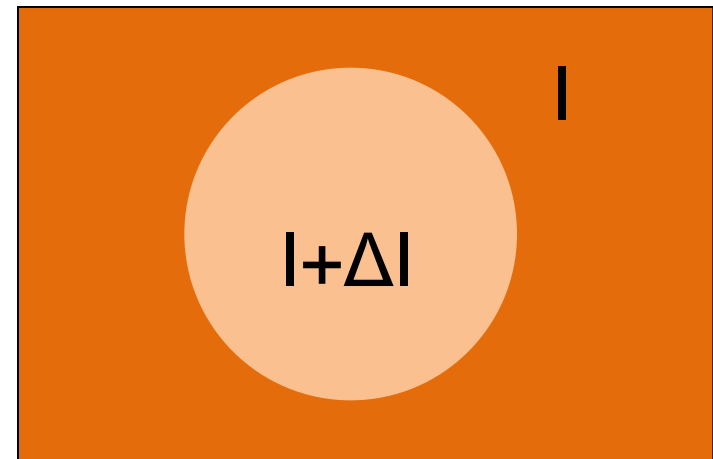
---

- Человеческое зрение имеет логарифмическую характеристику восприятия интенсивности света
- Приближаем функцией степени

$$L^* = 116 \left( \frac{Y}{Y_n} \right)^{\frac{1}{3}} - 16; \quad 0.008856 < \frac{Y}{Y_n}$$

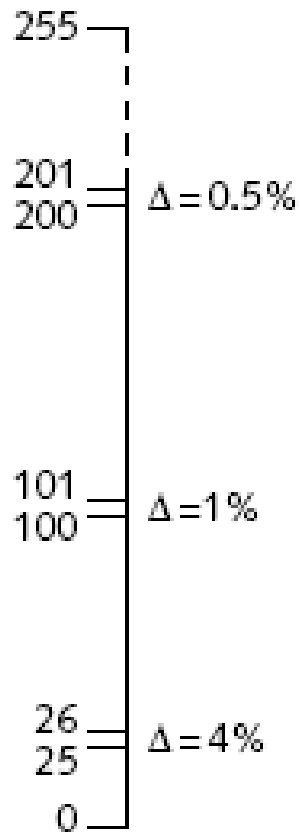
# Чувствительность к контрасту: минимальная различимая разница (закон Вебера)

- Задача – найти минимальную различимую разницу интенсивностей  $\Delta I$ 
  - JND – just noticeable difference
- Экспериментально получена кривая чувствительности к контрасту (закон Вебера)
- $\Delta I/I \sim \text{const} \sim 0.02$
- Вывод: глаз реагирует на относительные интенсивности!
- Вывод: меньше 2% шаг человек не различит



# Причина 2: Линейное кодирование яркости

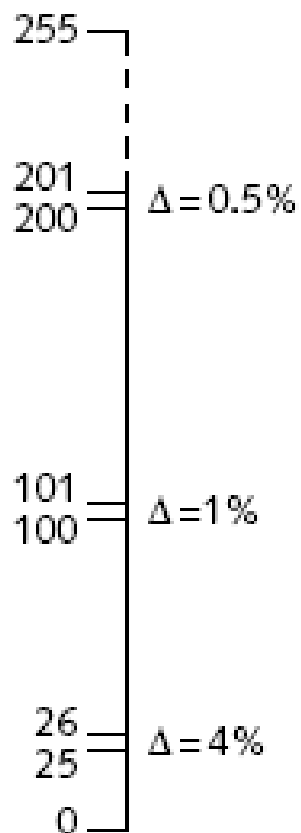
---



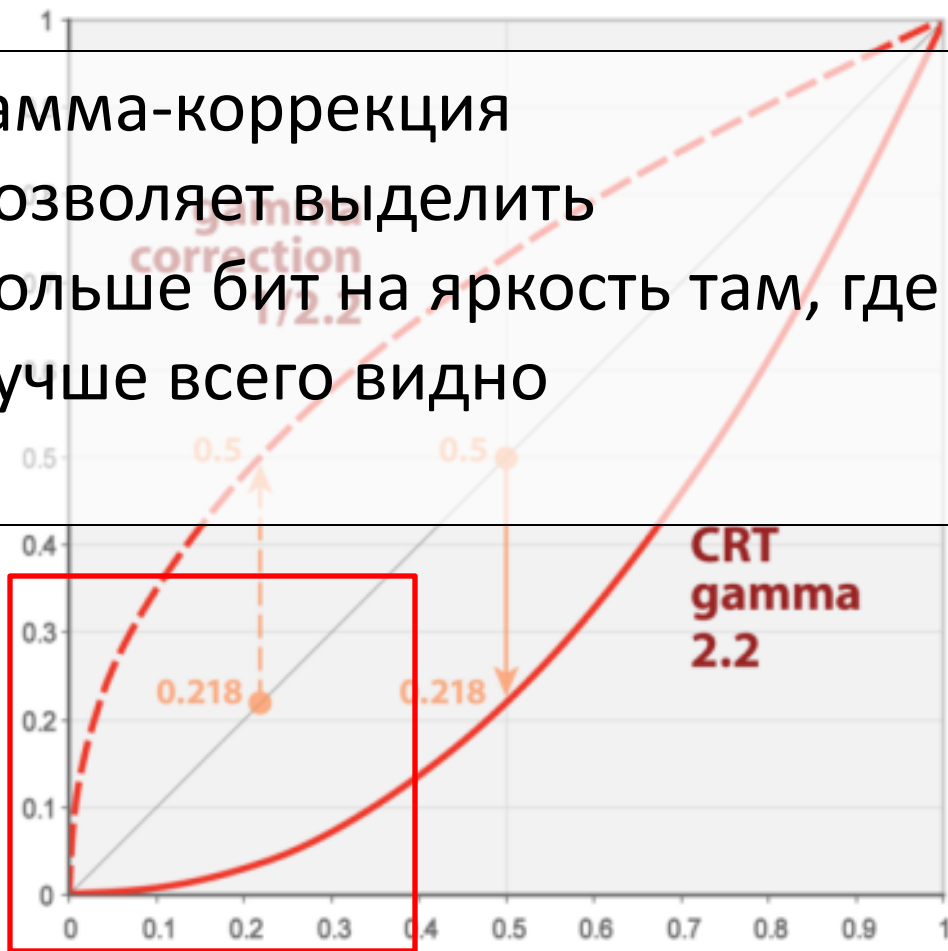
На яркостях меньше 100 будут видимые границы или ступени

Яркости больше 200 идут излишне «часто», т.к. неразличимы.

# Причина 2: Кодирование яркости

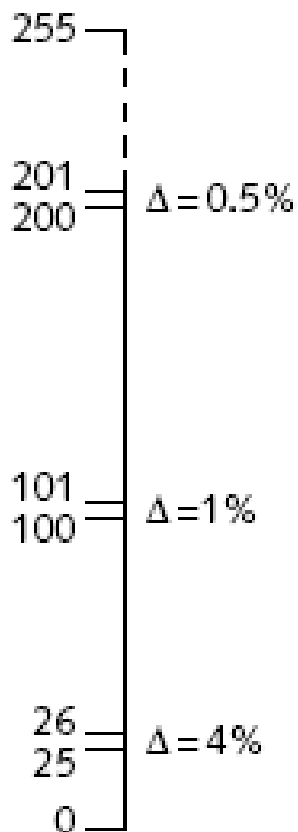


- Гамма-коррекция позволяет выделить больше бит на яркость там, где это лучше всего видно



# Причина 2: Кодирование яркости

---



Без гамма-коррекции:

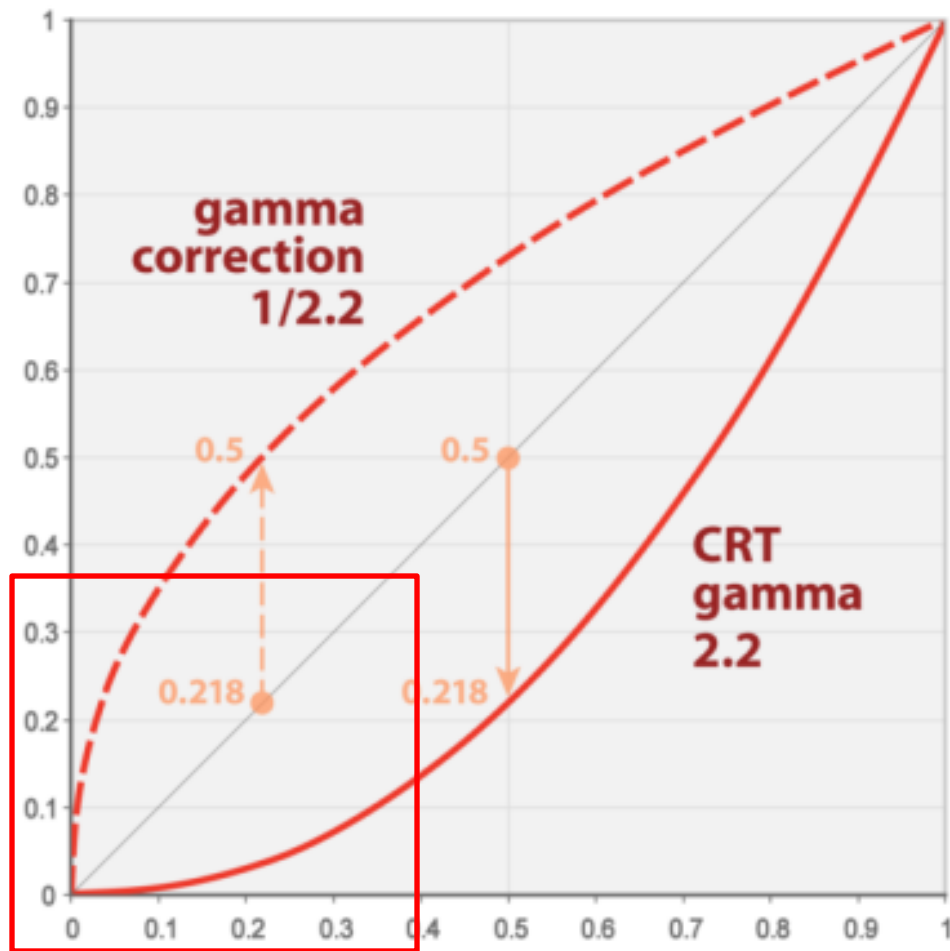
26/25 - разница 4%

После гамма-коррекции

88/90 - разница 1,8%

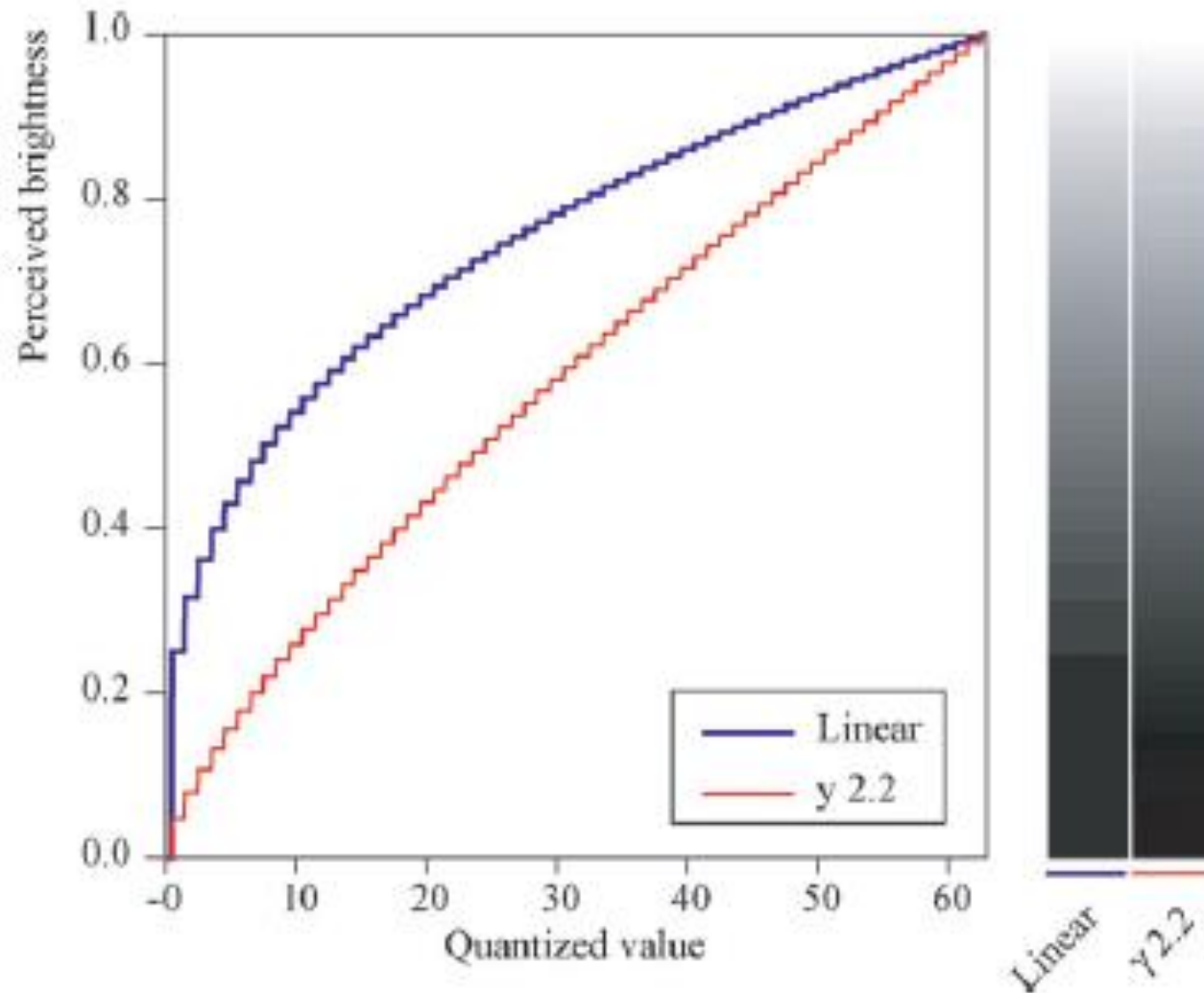
# Причина 2: Кодирование яркости: Функция монитора

- Монитор выполняет «аналоговое» расширение диапазона яркостей, сжимая темные области
- Тем самым увеличивается «плотность» градаций, оставляя разницу в пределах 1-2%





# Сравнение линейного и гамма-кодирования



# Кодирование яркости: примеры

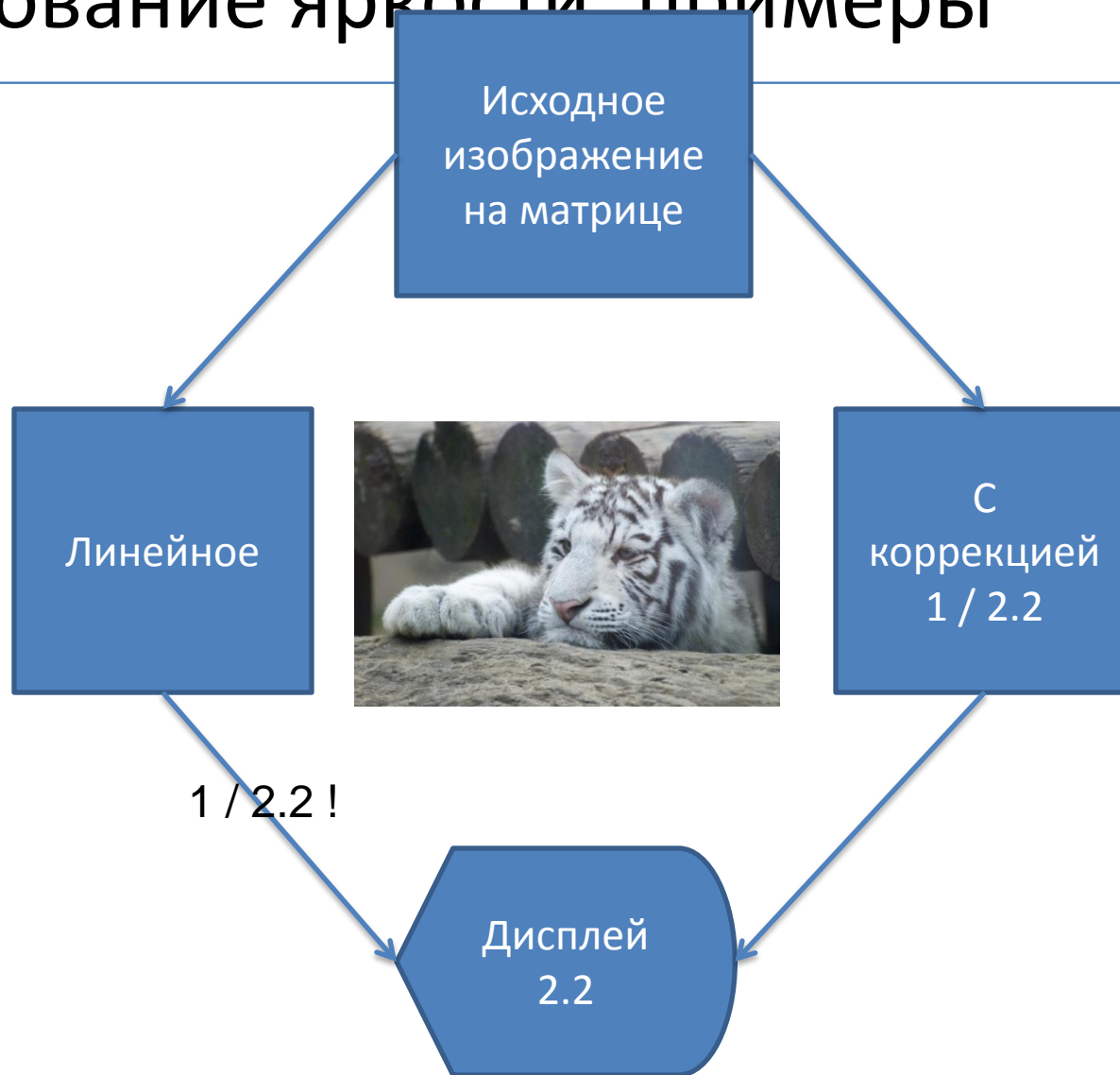
---

На входе – неквантованное изображение



Задача: сохранить его в формате: 24 бит на пиксель

# Кодирование яркости: примеры



# Кодирование яркости: примеры

---

Вариант 1: линейная дискретизация



(показано БЕЗ гамма-коррекции!)

# Кодирование яркости: примеры

---

Вариант 1: линейная дискретизация



(показано с гамма-коррекцией!)

# Кодирование яркости: примеры

---

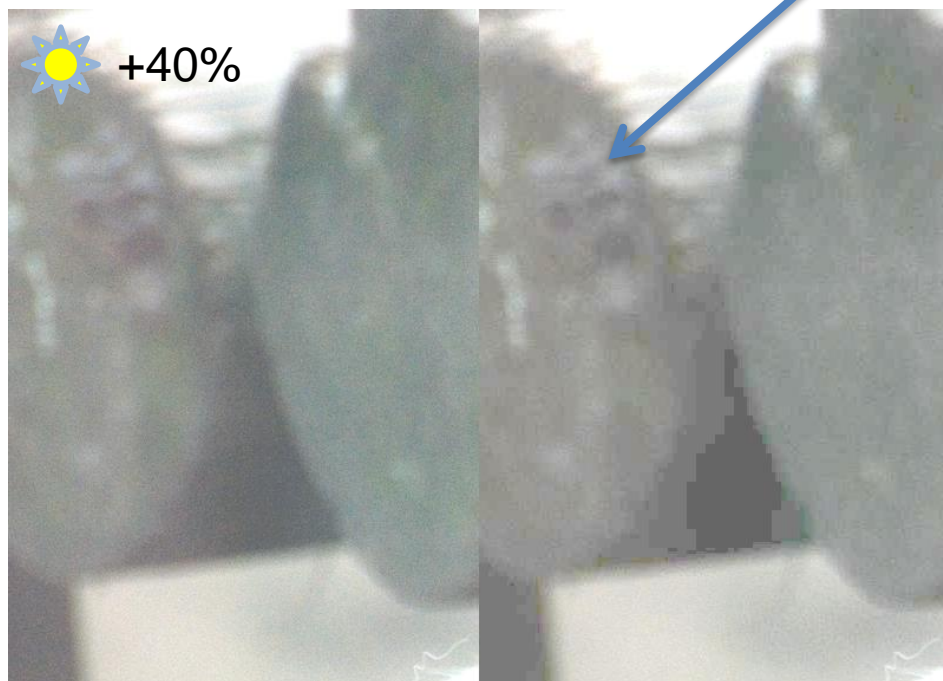
Вариант 2: нелинейная дискретизация



(Гамма-коррекция сразу в файле)

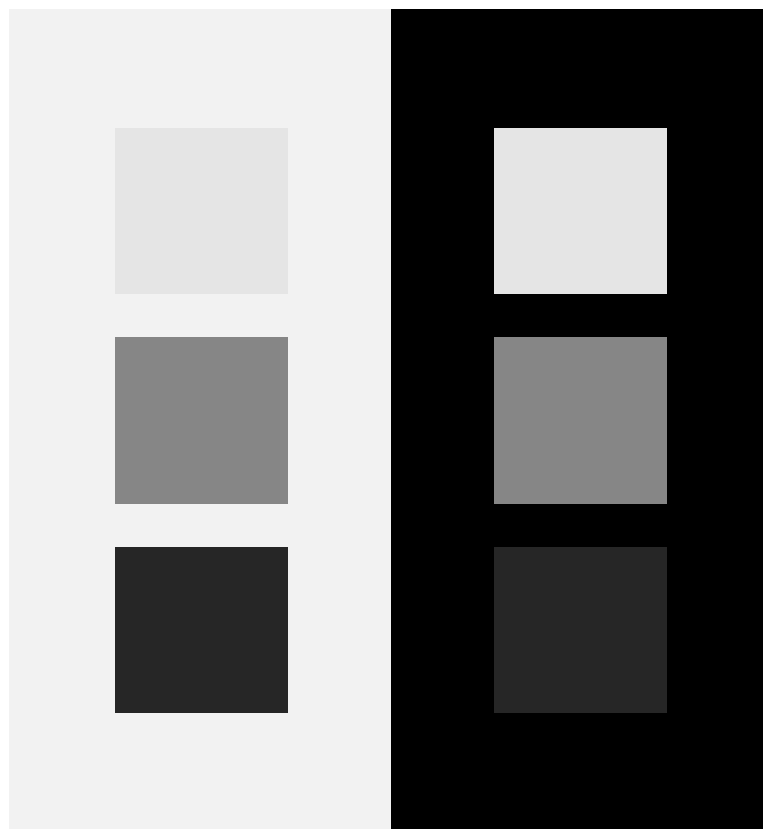
# Кодирование яркости: сравнение

---



# Причина 3: Одновременный контраст

- Зрительная система человека адаптируется к уровню окружающего освещения
- При ярком свете контраст повышается
  - черное кажется контрастно черным
- В темноте контраст понижается
  - черное становится серым





# Одновременный контраст

---

- Можно использовать системы с суммарной передающей функцией с  $\gamma \neq 1.0$ , чтобы компенсировать разные освещения при создании и при просмотре изображения!
- $\gamma > 1.0$  больше контраст
- $\gamma < 1.0$  меньше контраст

# Одновременный контраст: применение в видео

---

- Используется в видео
- «Недокоррекция»: предполагается  $\gamma=2.5$ , а корректируется 2.2
- Результирующая  $\gamma \approx 1.1$ 
  - изображение более контрастно, что правильно при просмотре в темном окружении

$\gamma=2.2$  (результатирующая 1.0)



Предполагаем  $\gamma=2.5$   
(результатирующая 1.1)



Предполагаем  $\gamma=2.8$   
(результатирующая 1.27)



# Гамма в Windows

---

- Гамма не корректируется в драйвере!
- Для того, чтобы корректно вывести изображение на монитор, оно должно быть в нелинейном пространстве, с полной гамма-коррекцией
- С камеры уже приходят такие изображения, так что с ними ничего делать не нужно
  - При условии совпадения гаммы!
  - Гамма может быть в профиле (например, sRGB)
- Если ничего не сказано, используйте гамму 2.2
  - Прописано в sRGB

# Примеры изображений!

Без гамма-коррекции



Гамма 1.8



# Примеры изображений!

Без гамма-коррекции



Гамма 2.2





# Примеры изображений!

Без гамма-коррекции



Гамма 2.5



# Примеры изображений



# Еще пример

Гамма-коррекция 1.0 (нет)



Гамма-коррекция 1.7



# Как узнать, какая гамма у монитора???

---

- 1) померять вручную (откалибровать)
- 2) считать ее равной 2.2
  
- Обычно гамма у CRT-монитора лежит в пределах 2.35-2.55
  
- У LCD должно быть так же, но в современных стараются делать 2.2

# Как работать с гаммой

---

- Варианты использования изображений для синтеза фотореалистичных изображений:
  - Текстуры
  - Источники света (панорамы и т.п.)
- Алгоритмы, работающие с изображениями, должны использовать линейное представление излучения

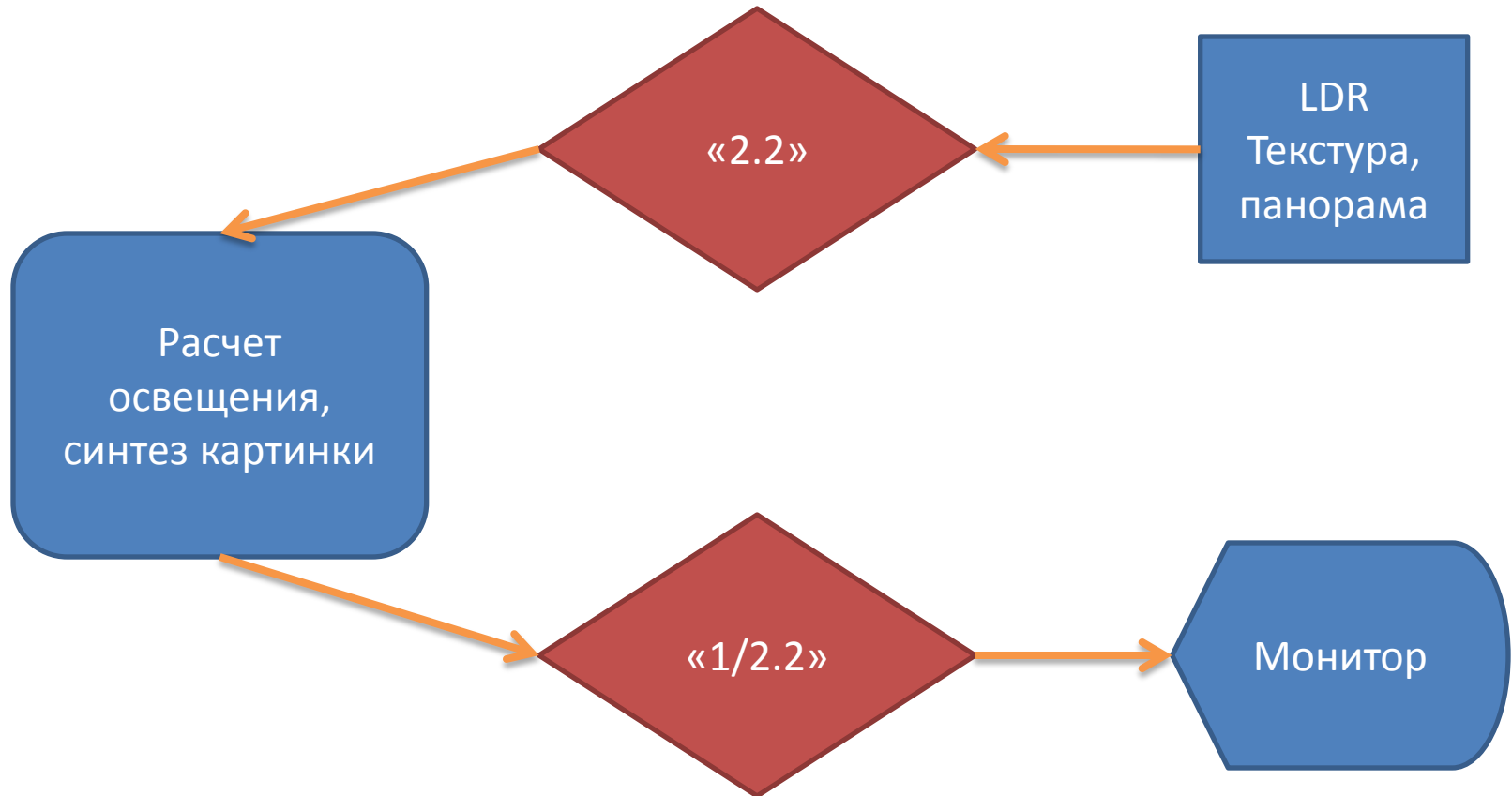
# Как понять, какая гамма в изображении?

---

- Если есть профиль, часто гамма указывается с профилем
- 99% процентов обычных изображений – JPEG, BMP и т.п. с гамма-коррекцией
- HDR-изображения без гамма-коррекции
- На платформе Windows это гамма 2.2. На других платформах может отличаться.
- Если в изображении нет профиля или просмотрщик не поддерживает профили, изображение может выглядеть неправильно!

# Процесс использования изображений в процессе синтеза

---



# Пример: работа с текстурами без и с гамма-коррекцией!

---

- Изображение 1 (JPEG): 200
- Изображение 2 (JPEG) : 50.
- Нужно наложить поверх с прозрачностью 0.4.
- Цвет =  $200 * 0.4 + 50 (1 - 0.4) = 80 + 30 = 110$



# Пример: работа с текстурами без и с гамма-коррекцией!

---

- Изображение 1 (JPEG, sRGB): 200
- Изображение 2 (JPEG, sRGB) : 50.
- Делаем гамма-преобразование, чтобы получить корректные значения излучения для данного источника.
- $(200 / 255) ^{2.2} = 149,4$  (округлено)
- $(50 / 255) ^{2.2} = 7,0$  (округлено)
- $\text{Цвет} = 149,4 * 0.4 + 7,0 (1 - 0.4) = 92,5$
- Перед выводом делаем опять гамма-коррекцию:
- $(92,5 / 255)^{(1/2.2)} = 160,8151$  !!!
- VS 110 !

# Пример: смешивание изображений

---

50%





**50% без гамма-преобразования**



**50%, с гамма-преобразованием**

# А почему там шум какой-то вылез?

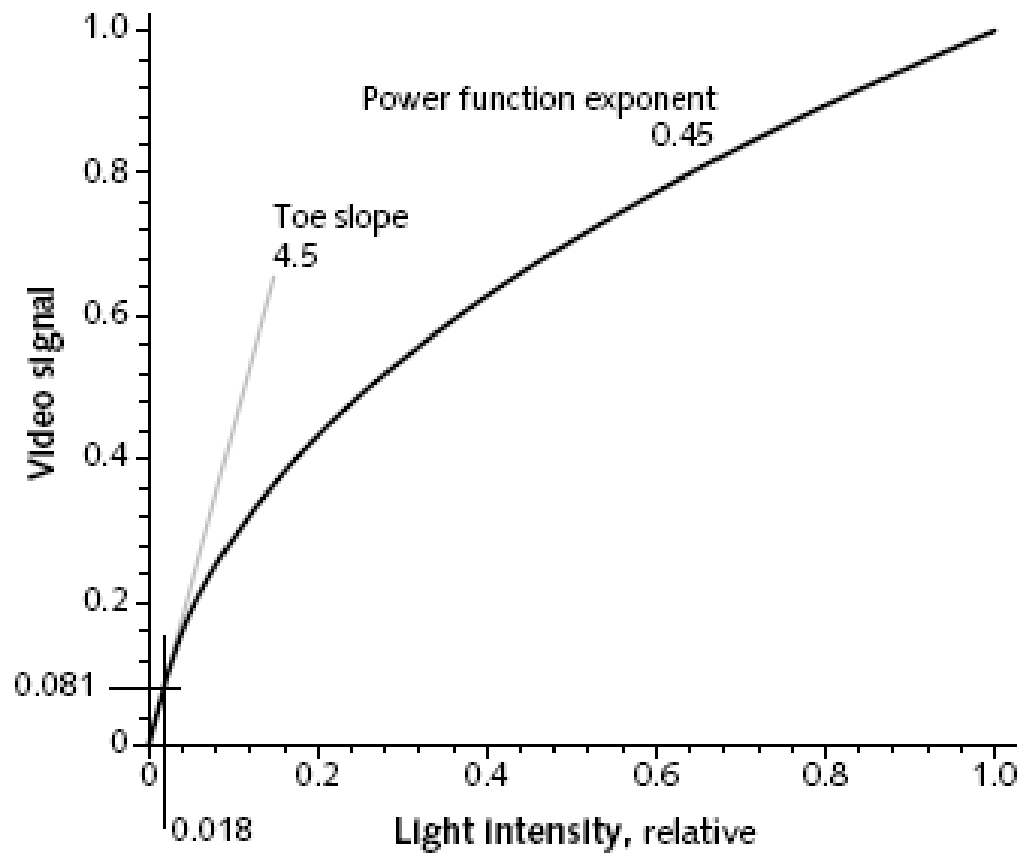
- Потому что произошла потеря точности из-за преобразования
- $C = (\text{БУТЕ})C^{(2.2)}$
- ...
- $C = (\text{БУТЕ})C^{(1/2.2)}$



Вывод: если учитывать гамму, надо работать в floating-point или сразу готовить текстуры / освещение в линейном диапазоне

# Прямая около нуля

---



# Определение гаммы

