Дополнительные главы компьютерной графики

Лекция 1

Copyright © С. Свердлов, 2009-2017

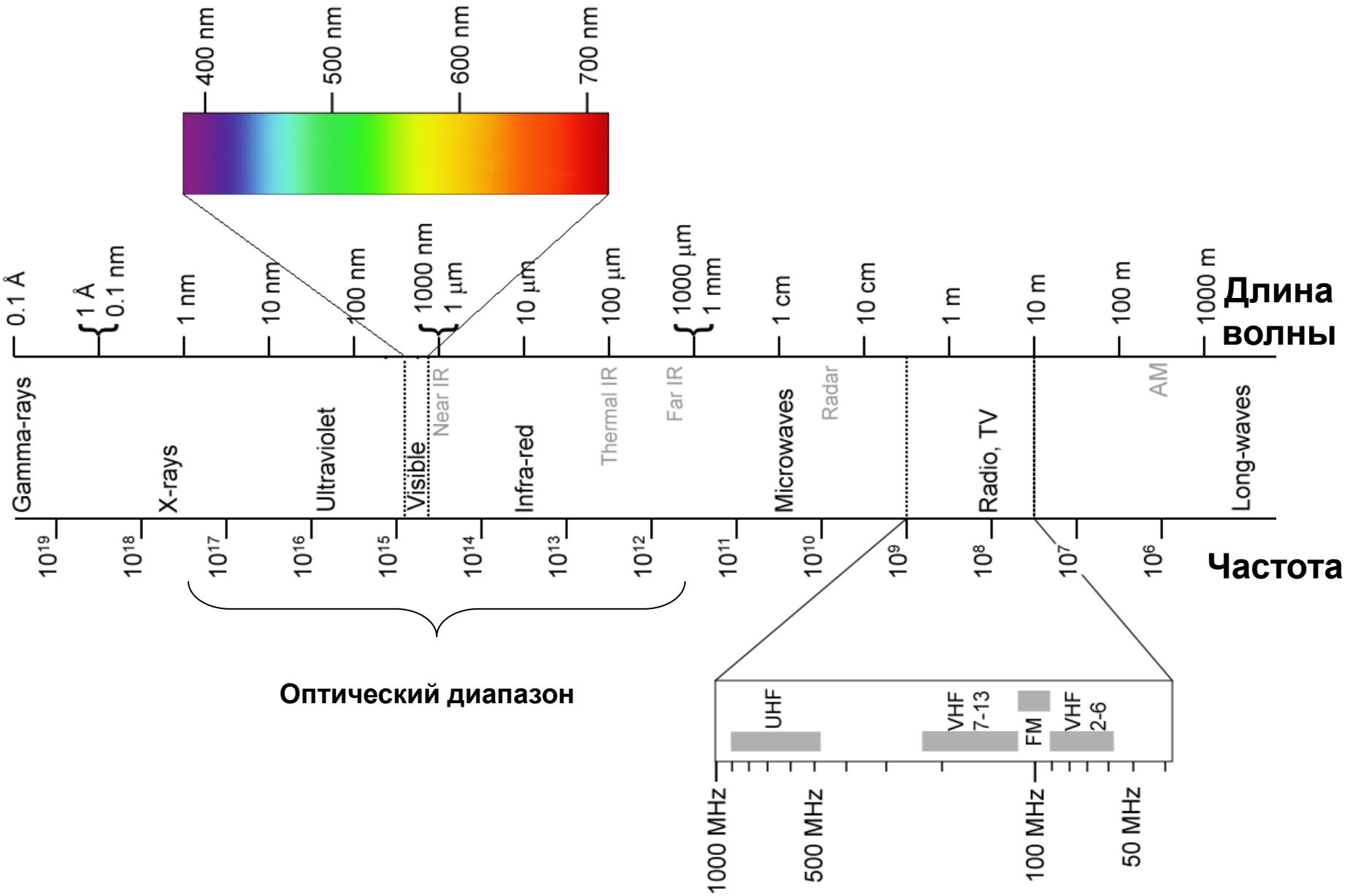
http://www.uni-vologda.ac.ru/~c3c

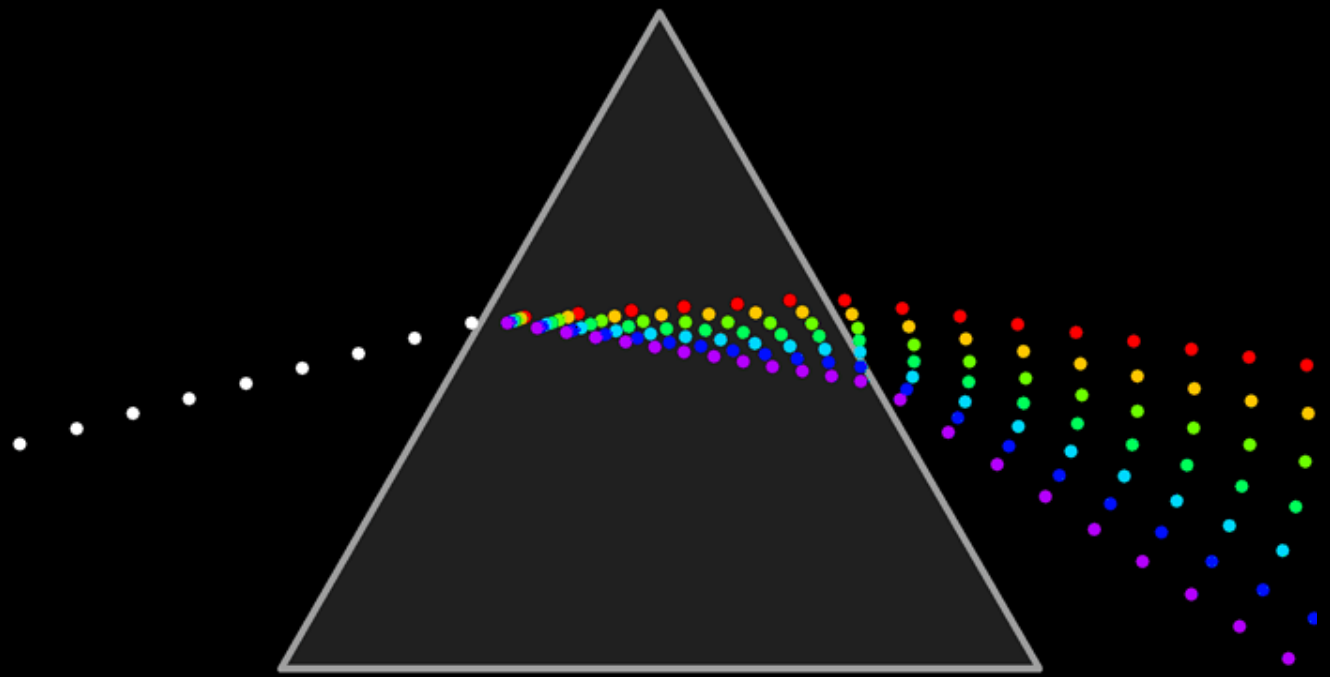
ЦВЕТ

Излучение

Радиометрия

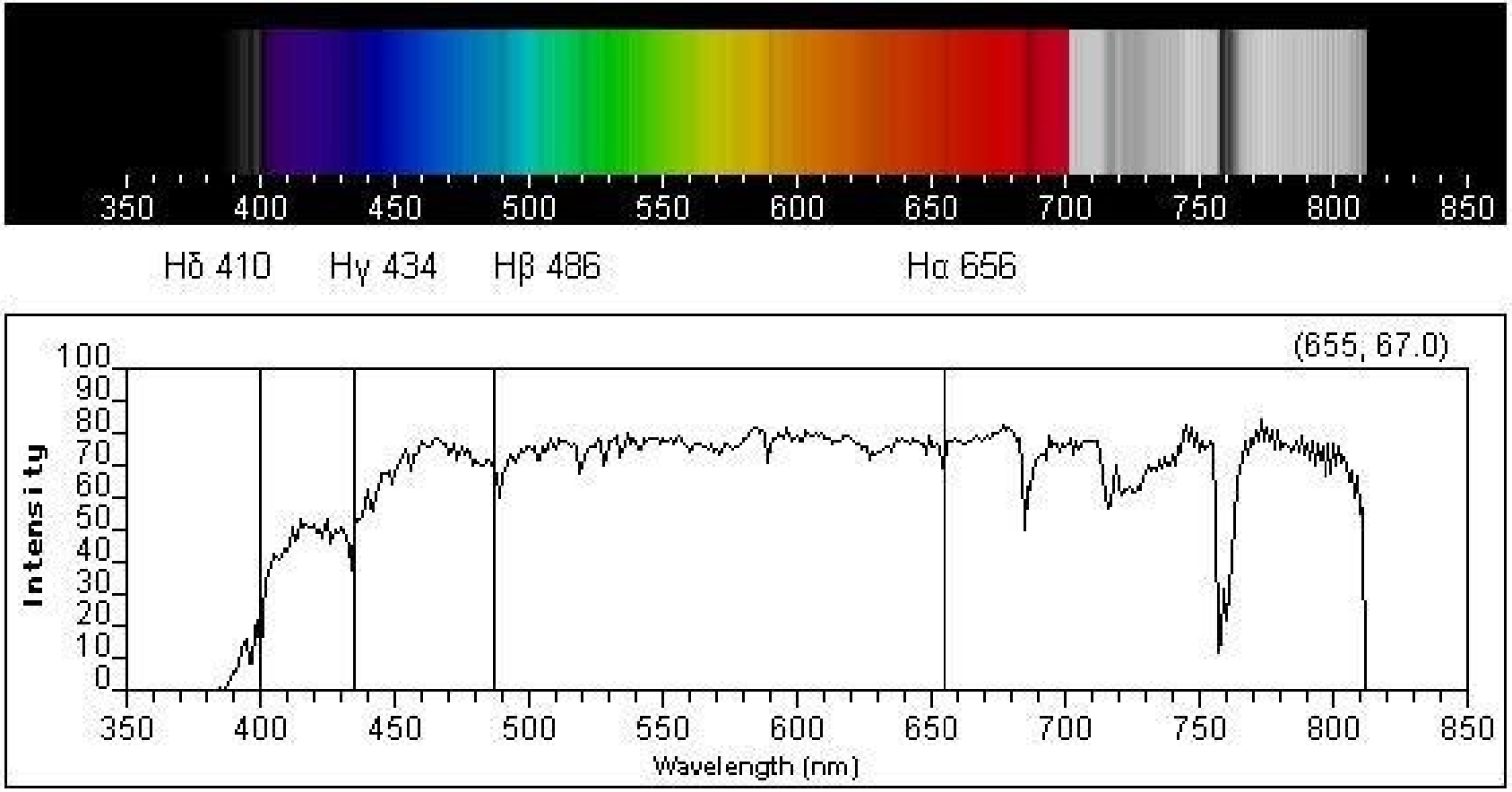
Электромагнитное излучение

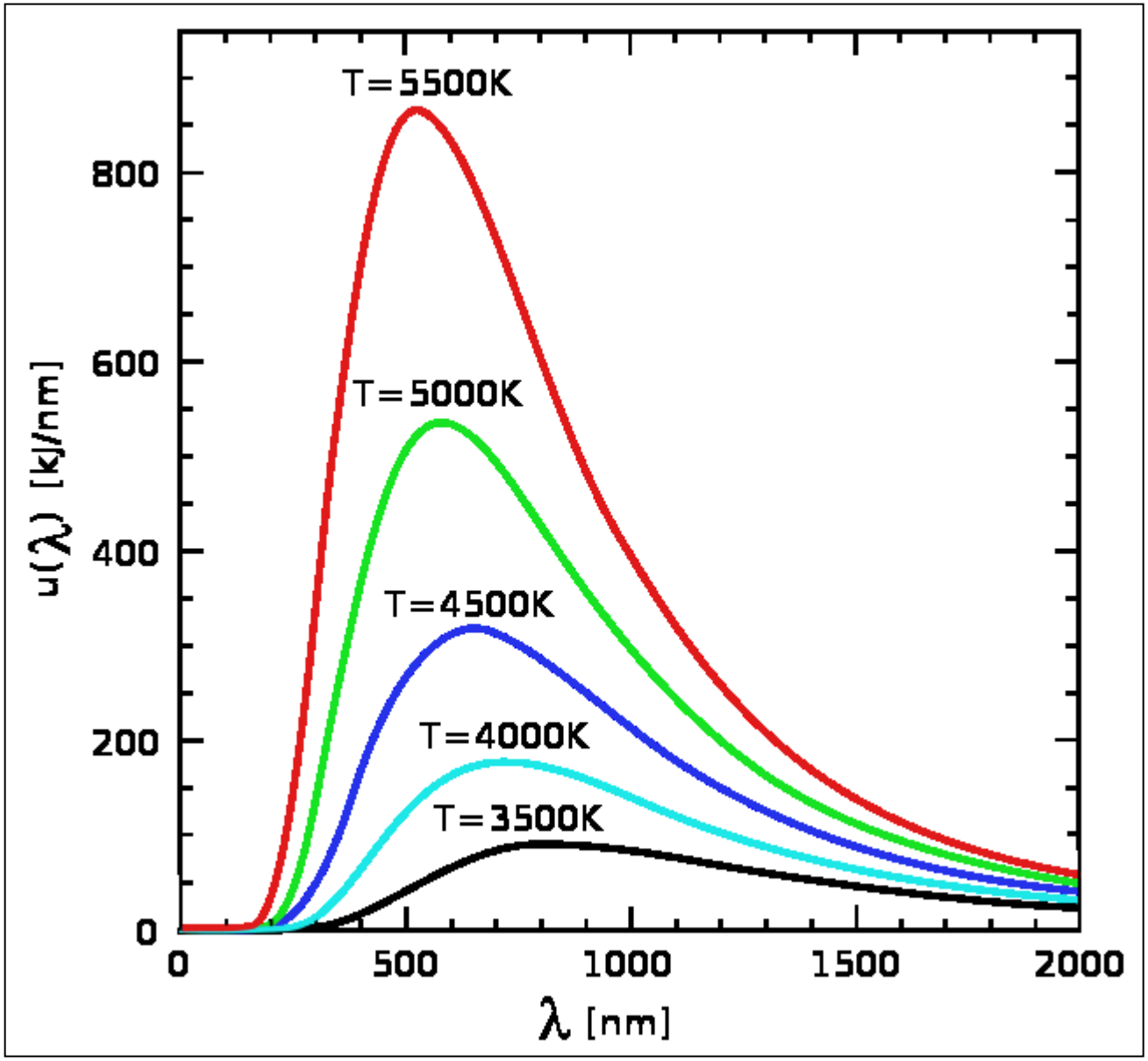




Важны спектральные свойства – зависимость от длины волны

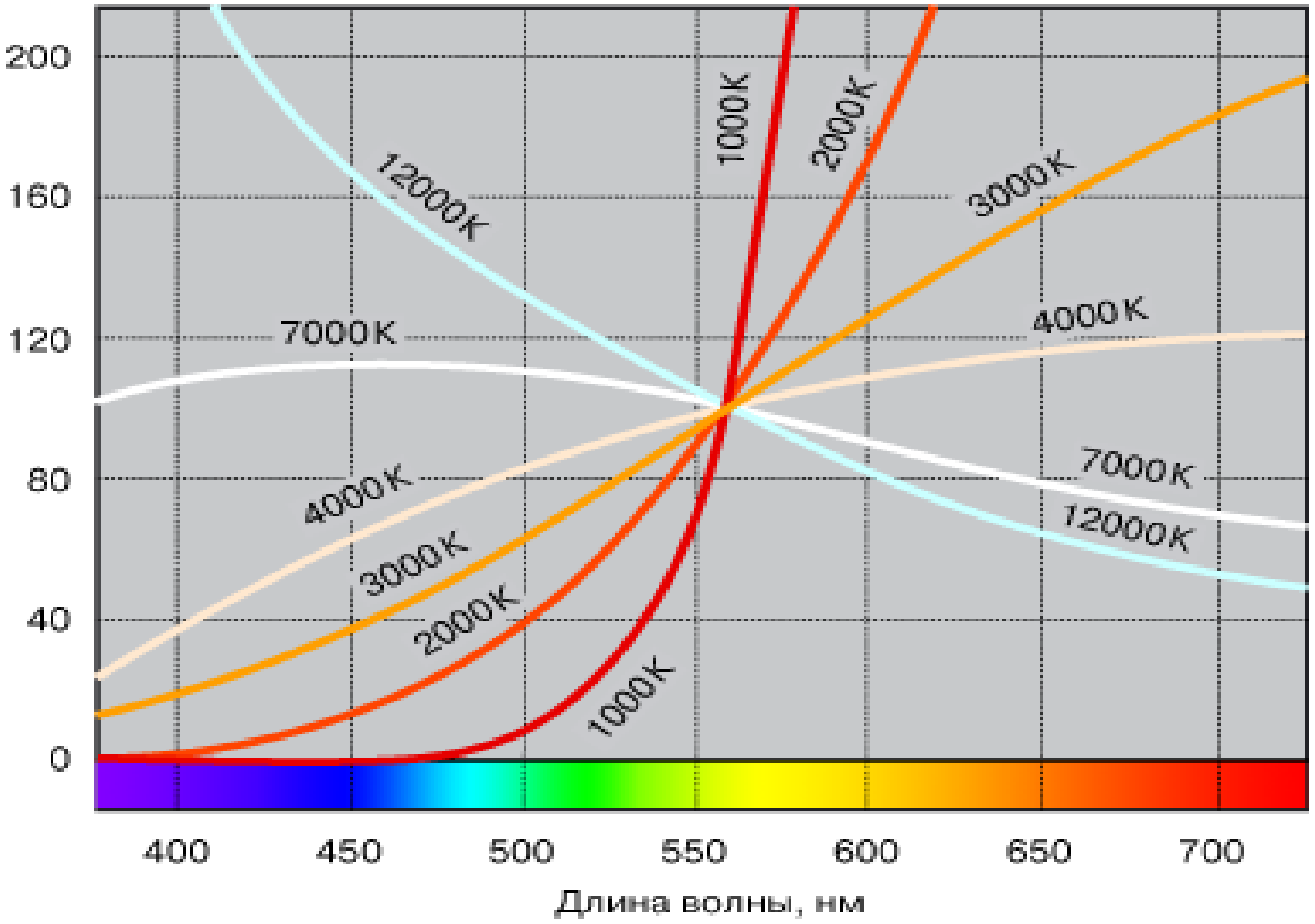
Спектр излучения и спектральное распределение энергии



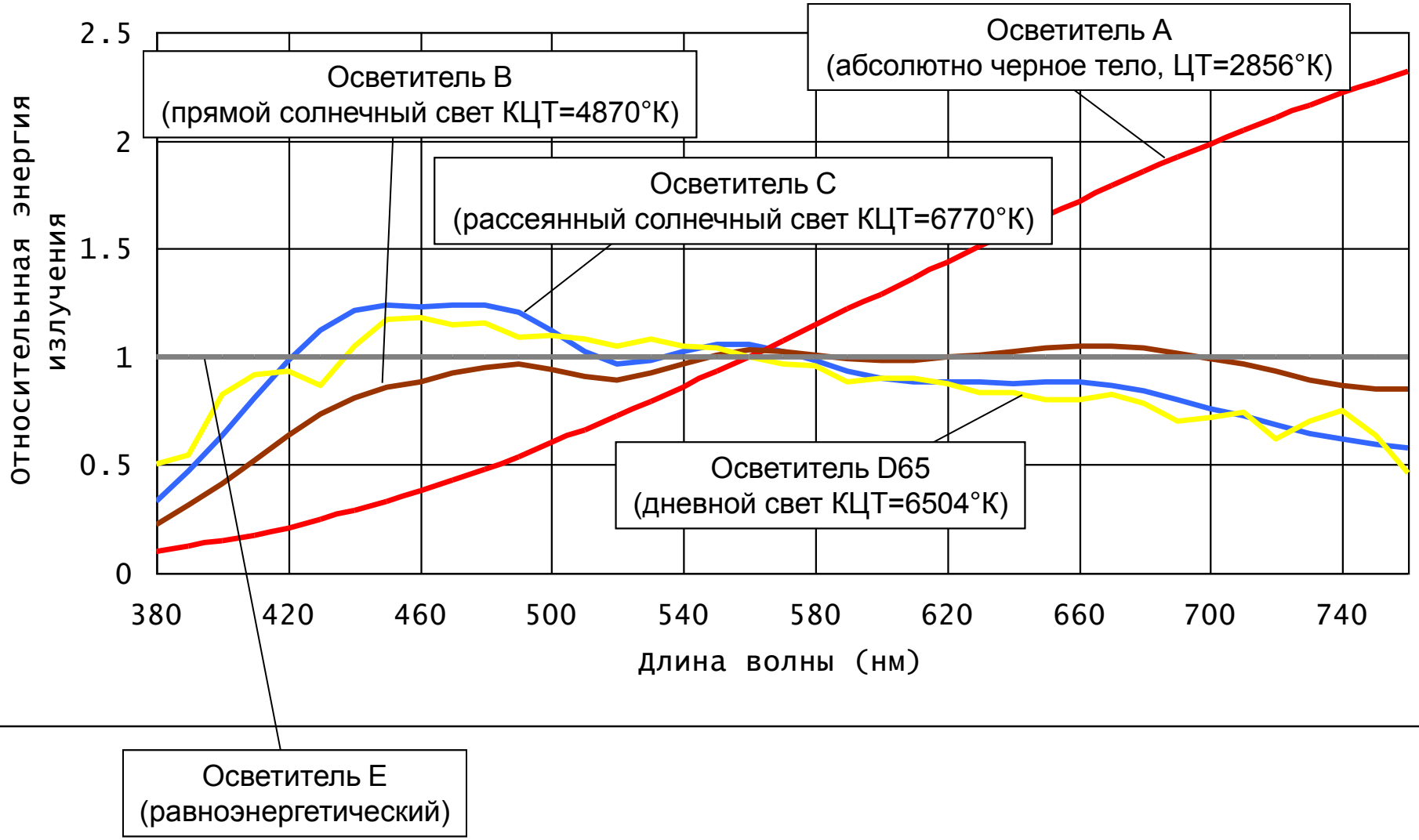


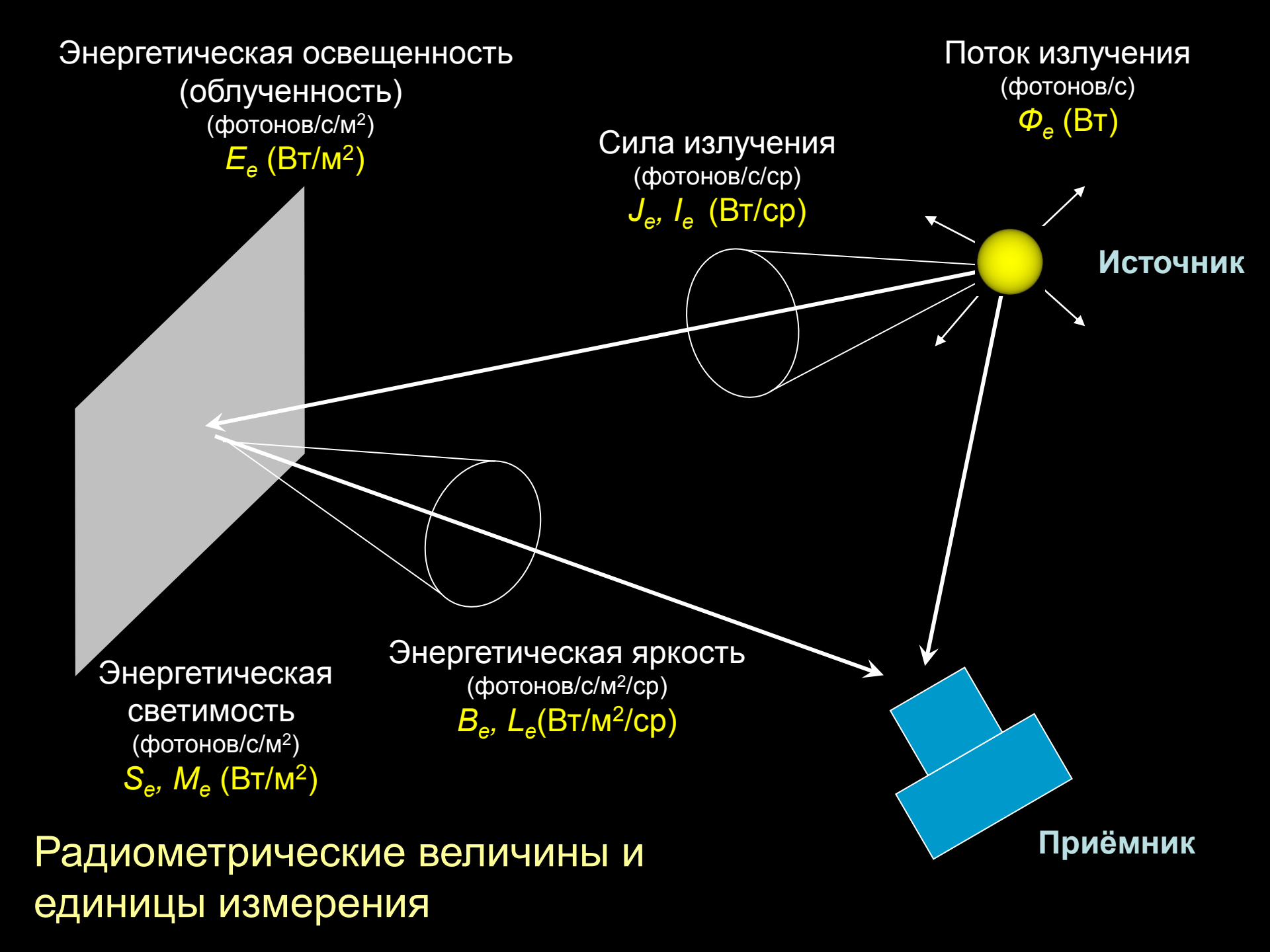
Спектральная характеристика излучения абсолютно черного тела

Спектральное распределение энергии излучения абсолютно черного тела в видимом диапазоне

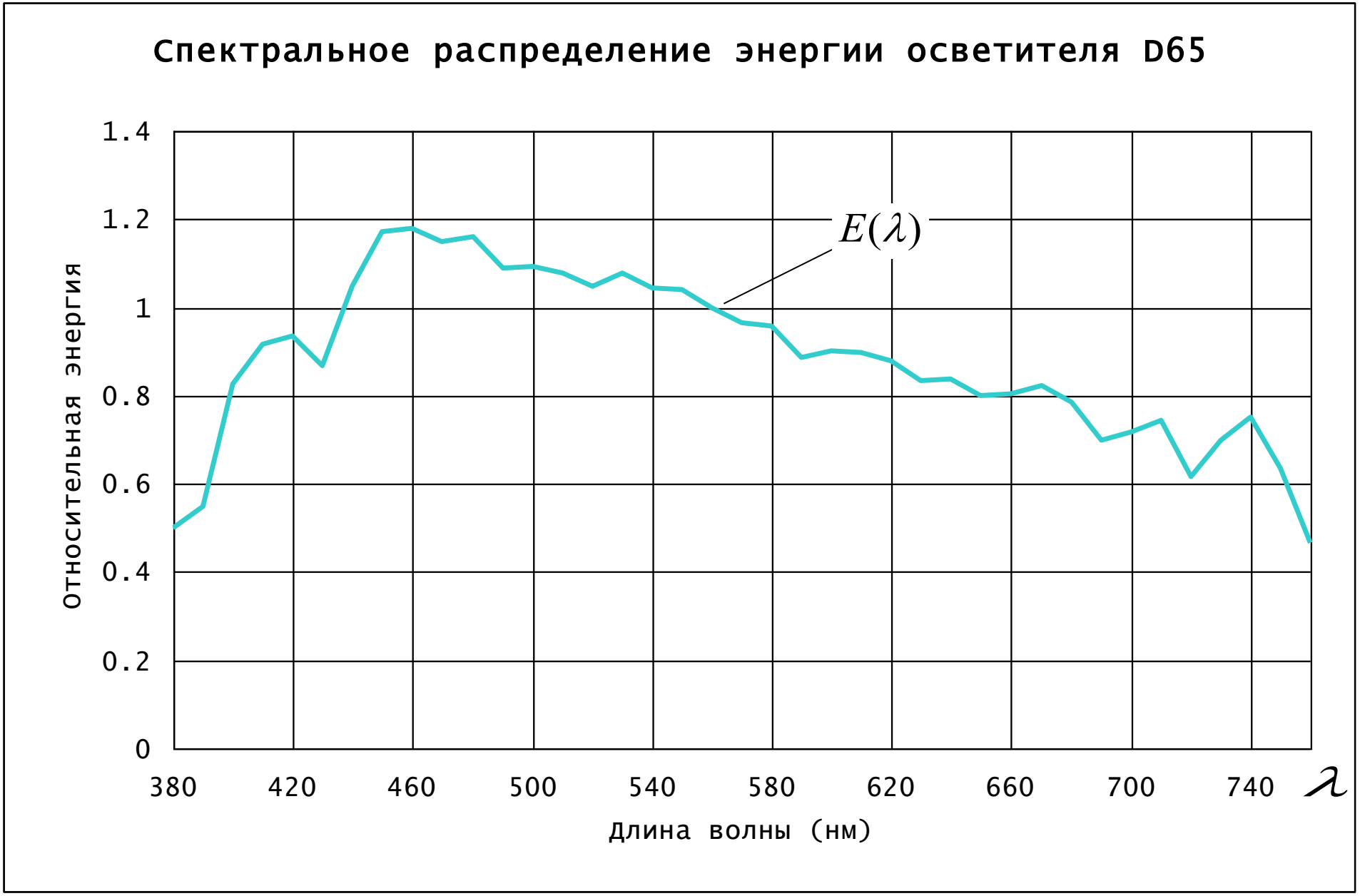


Спектральное распределение энергии излучения осветителей CIE (МКО)



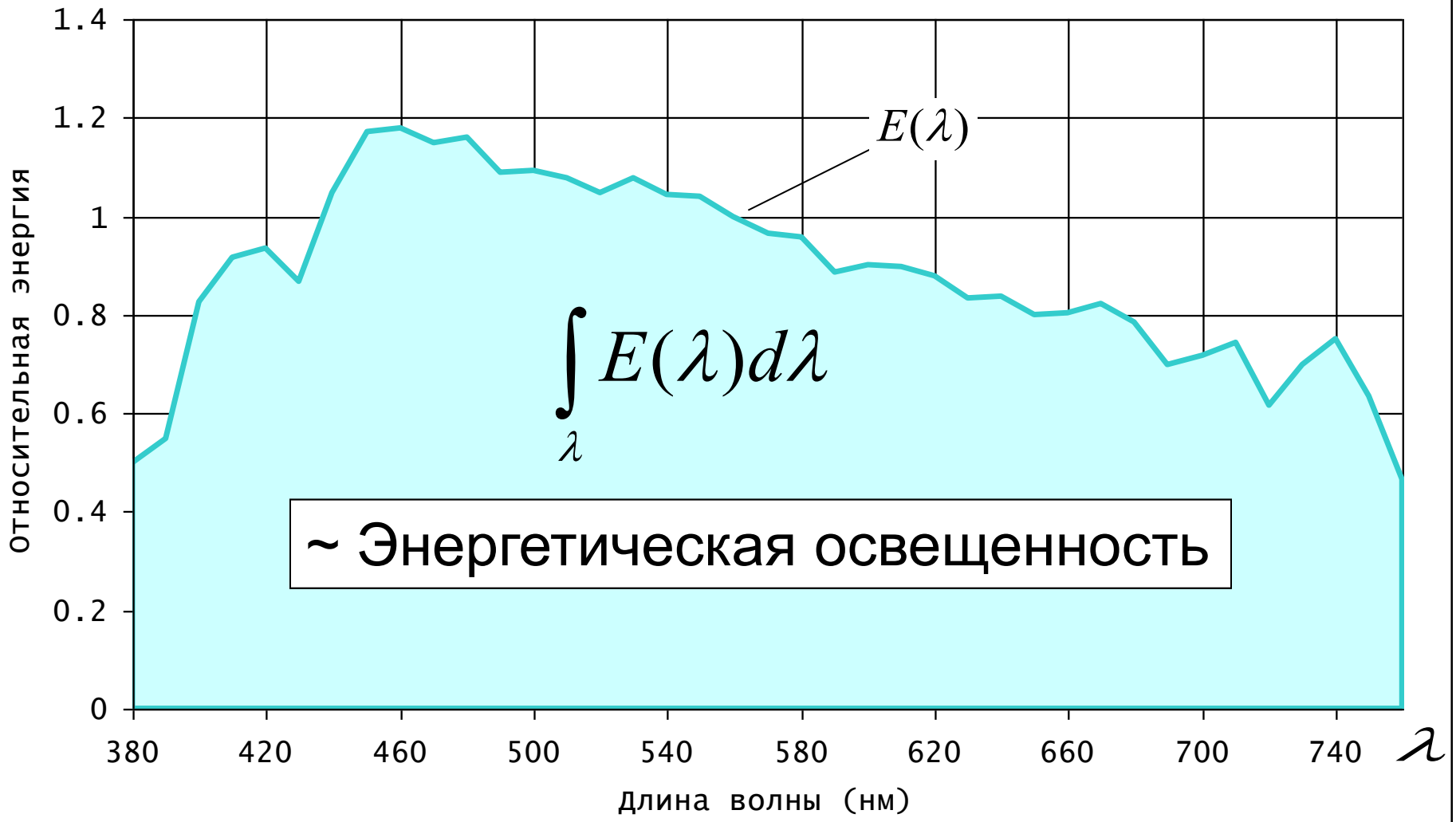


Радиометрические величины и единицы измерения



Спектральная энергетическая освещенность: $E(\lambda)$, Вт/м^2/нм

Спектральное распределение энергии осветителя D65

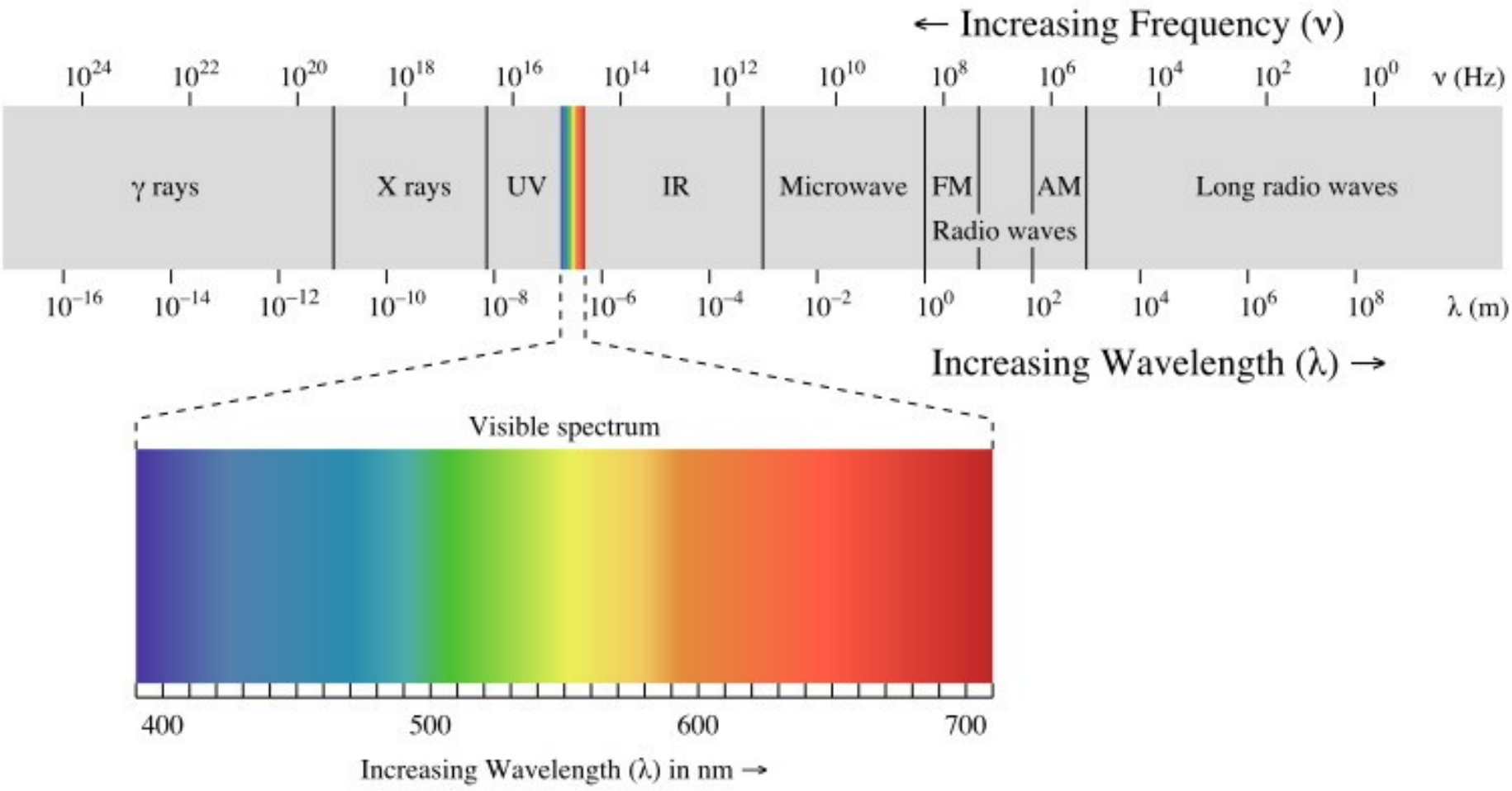


Свет

Фотометрия

Видимый свет

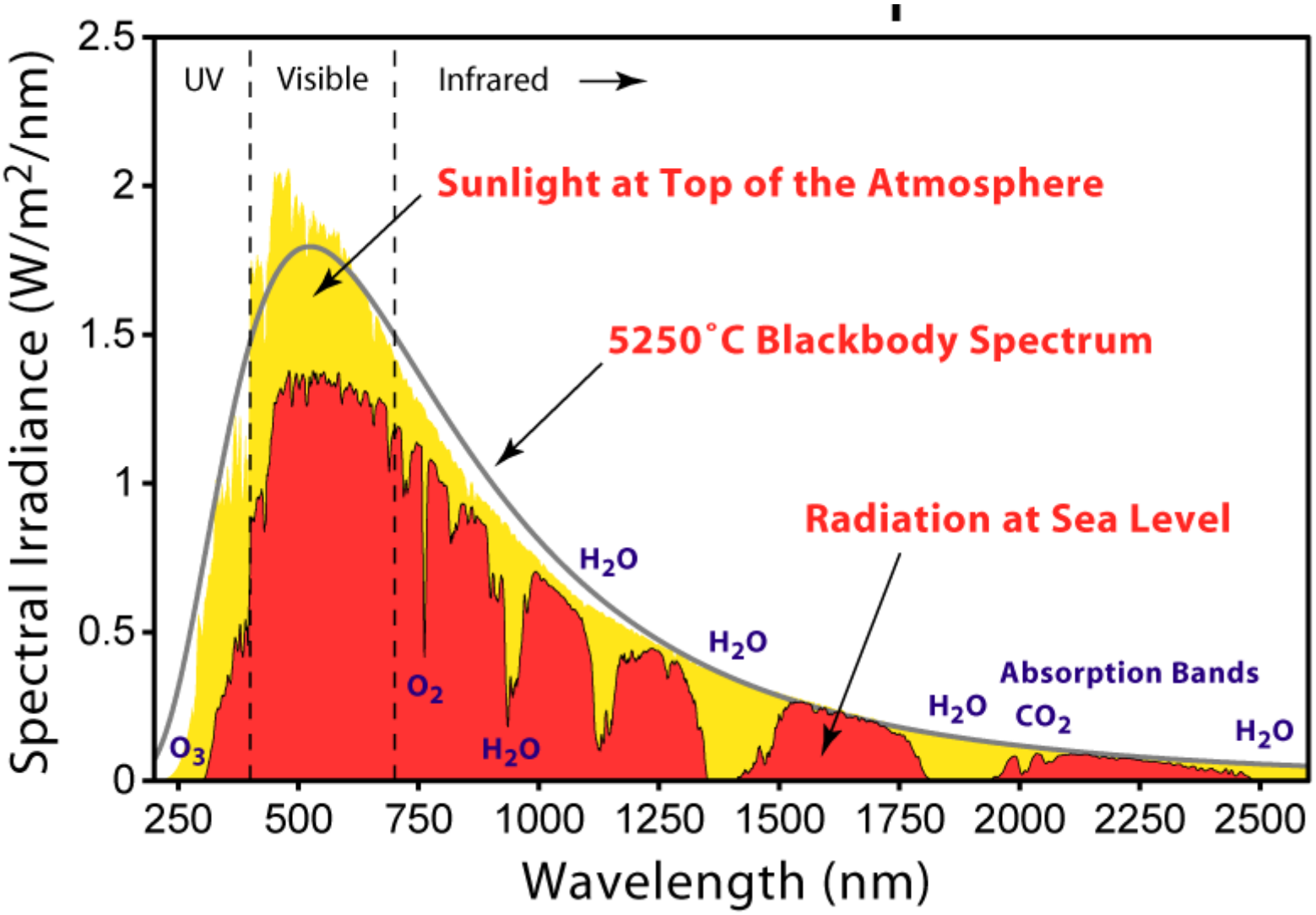
Обязательно присутствие наблюдателя (хотя бы виртуальное)



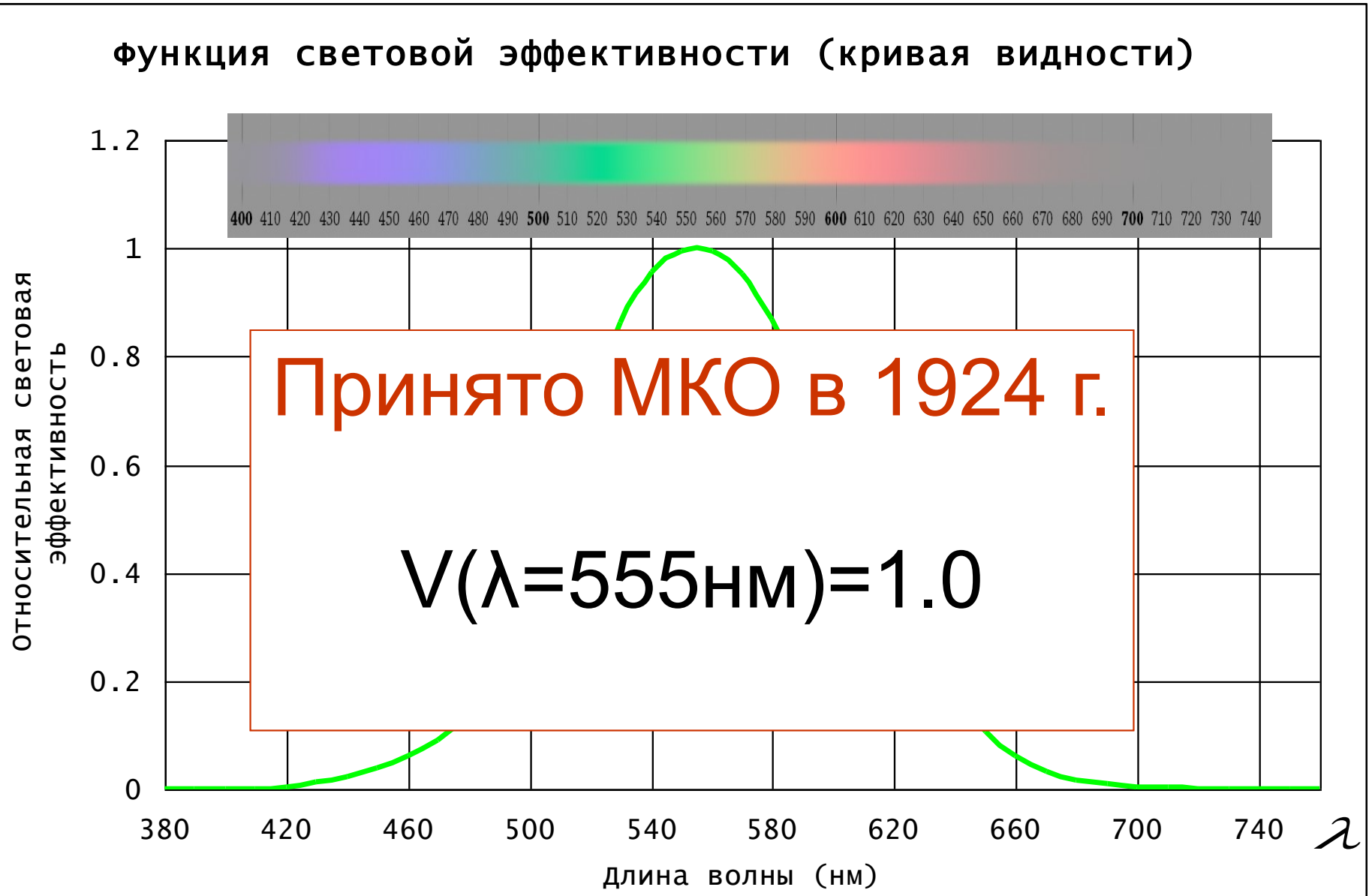
Вопрос: Почему человек видит именно в таком диапазоне?

Ответ: Потому что спектр излучения Солнца лежит преимущественно в этом диапазоне.

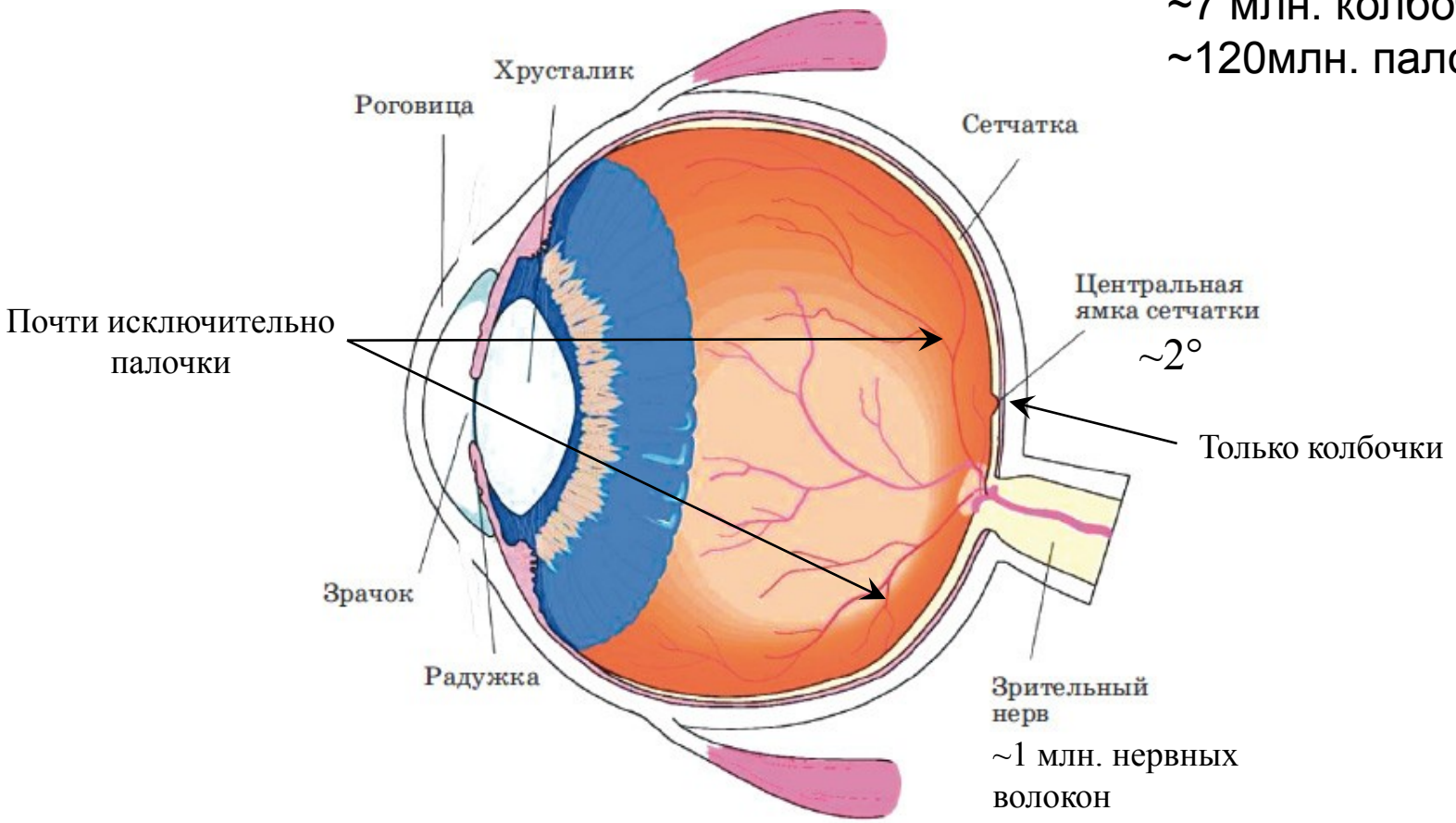
Спектр излучения Солнца



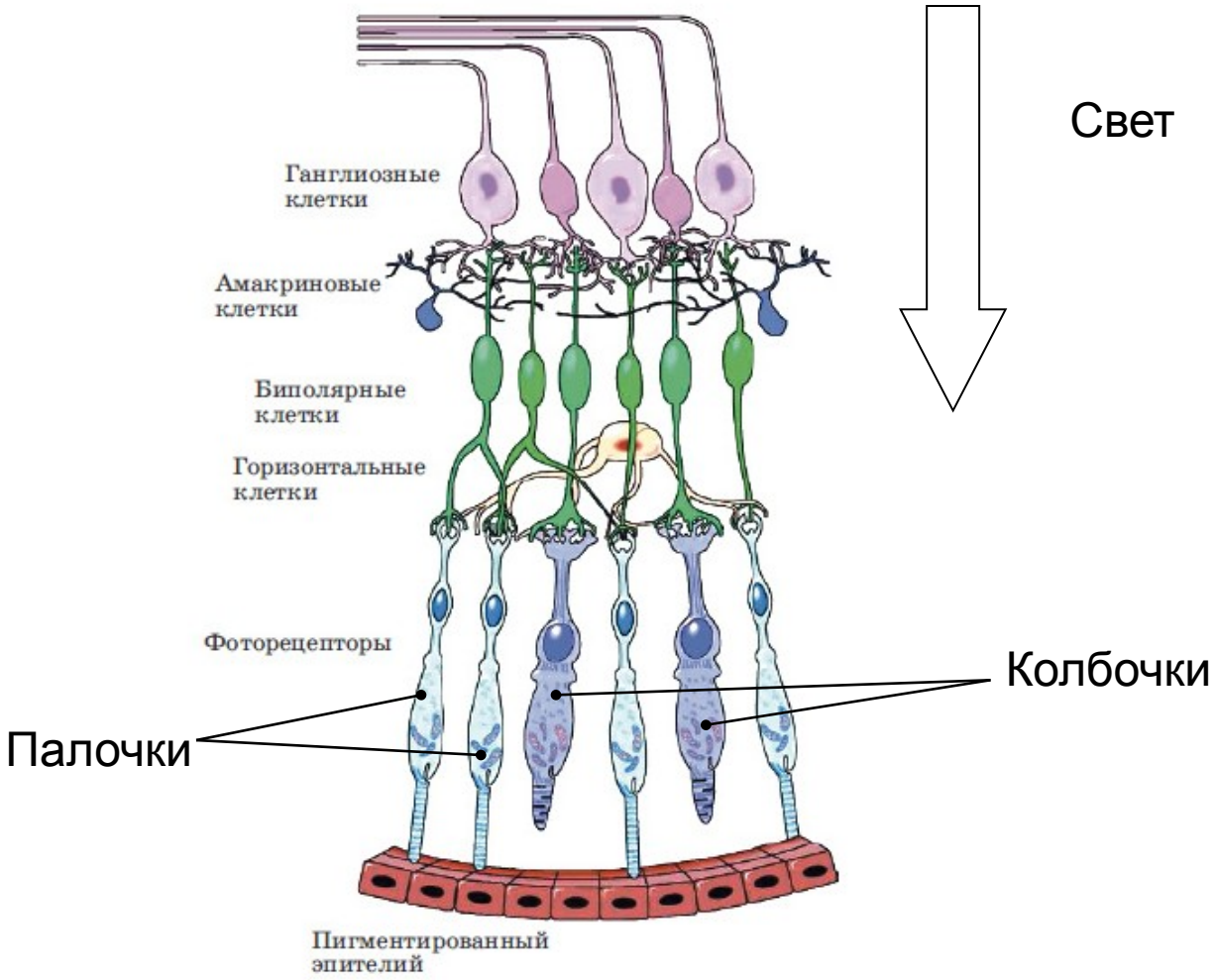
Спектральная чувствительность глаза



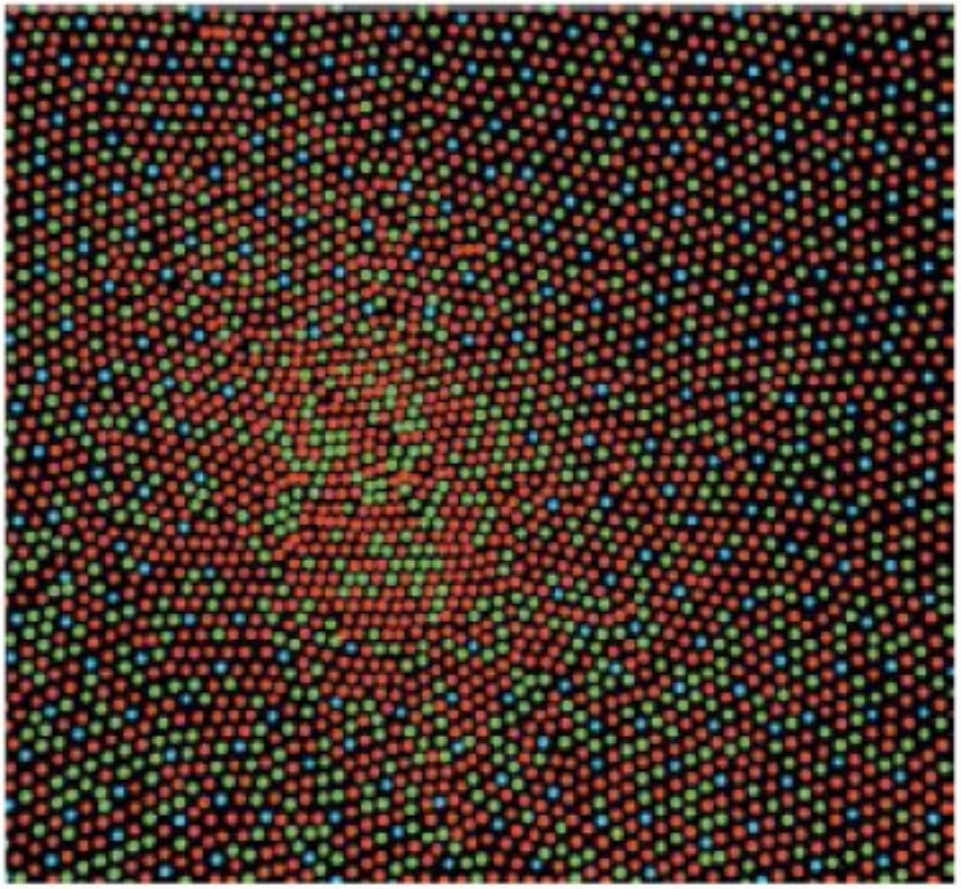
~7 млн. колбочек
~120млн. палочек



Строение глаза

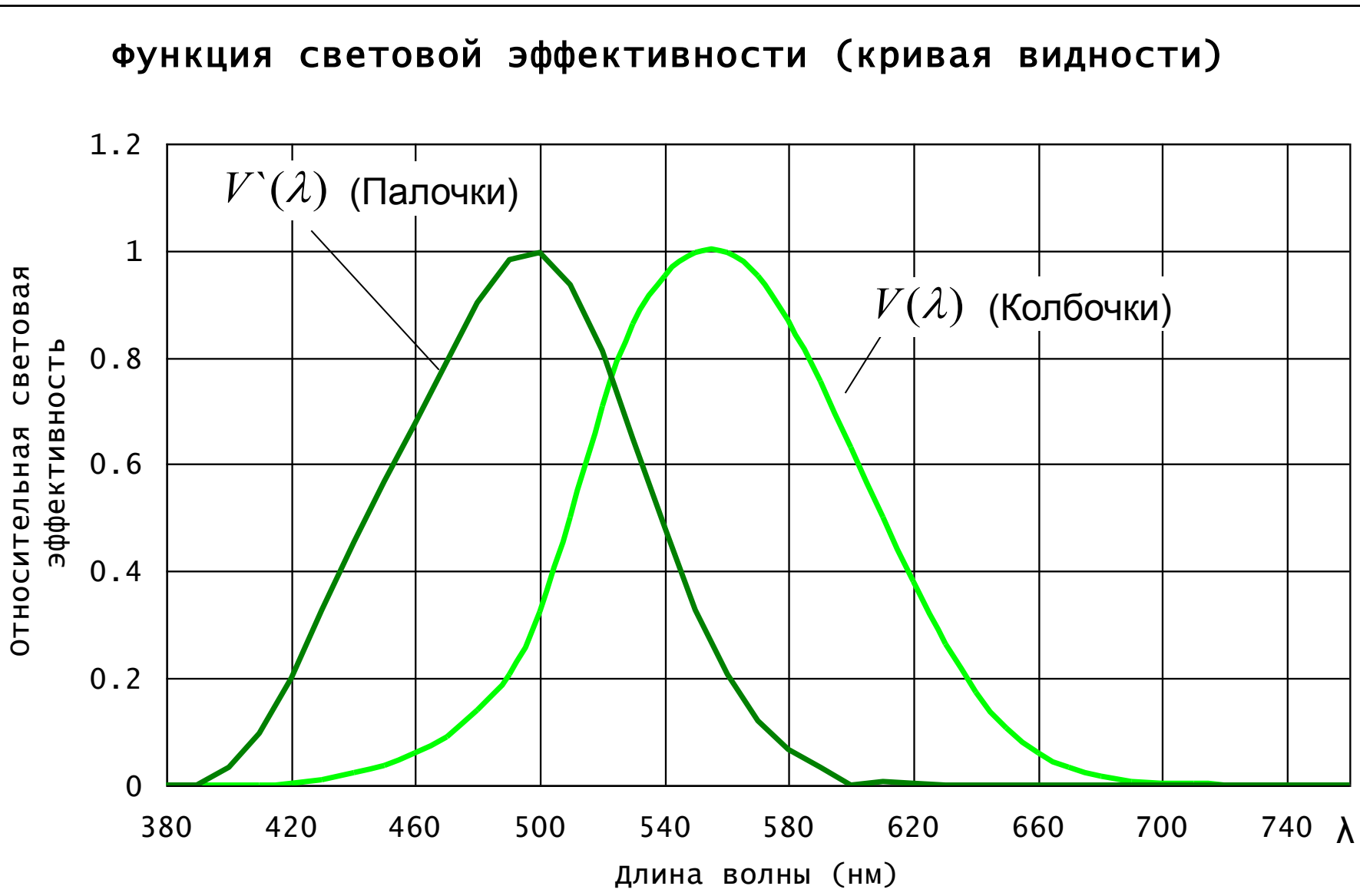


Палочки и колбочки в сетчатке глаза

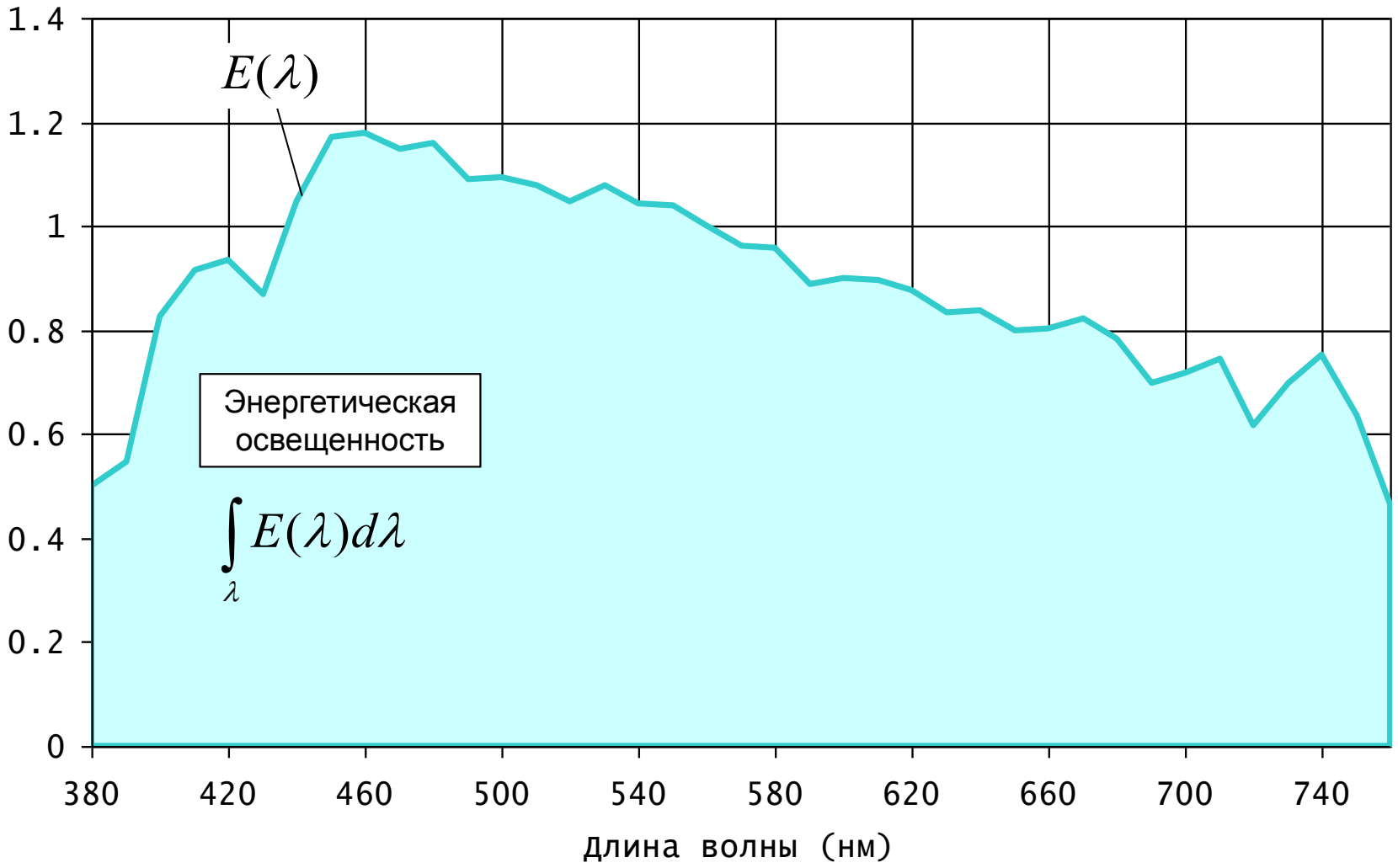


Заселенность сетчатки (центральная ямка) колбочками (модель)

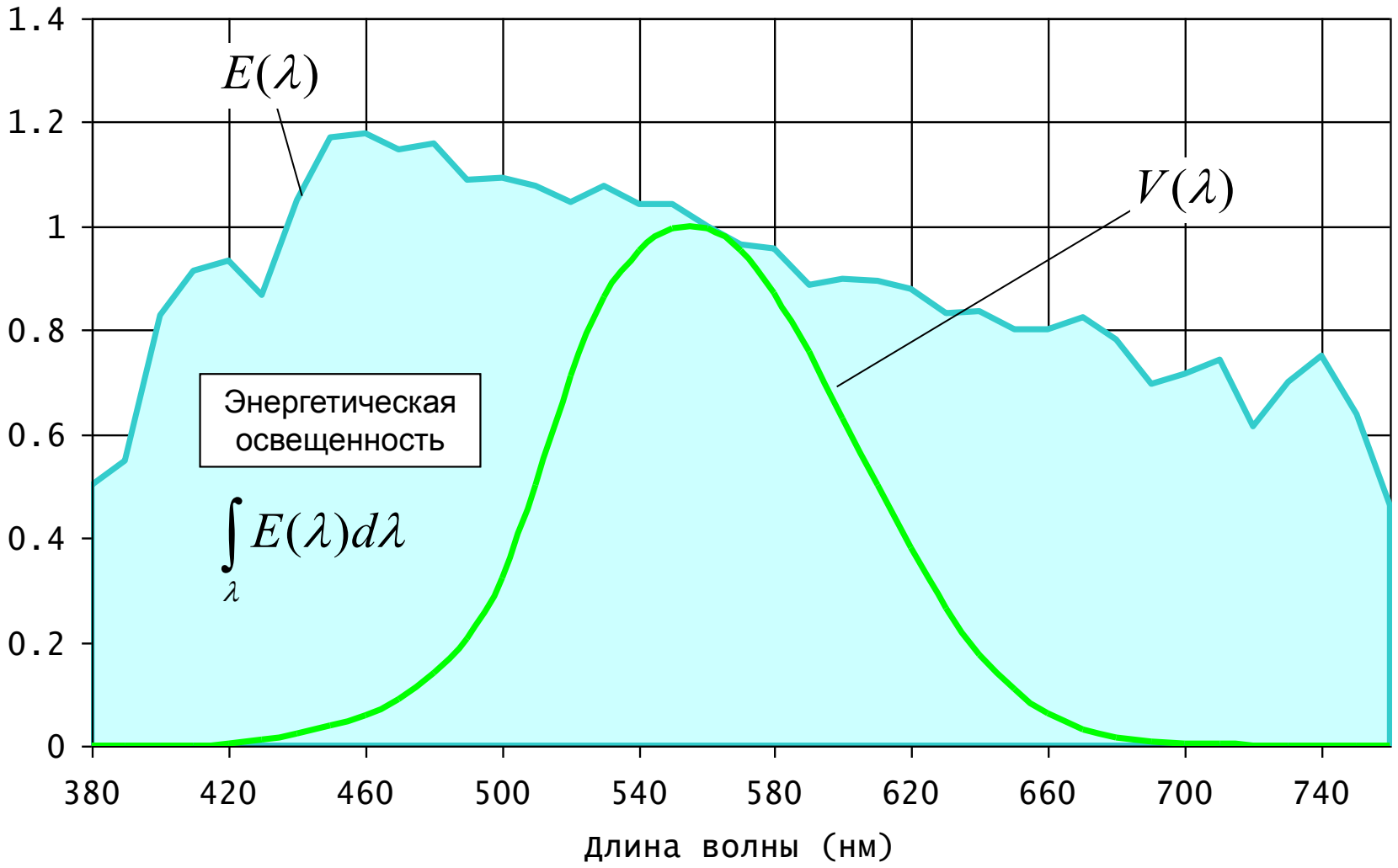
Фотопическое (дневное) и скотопическое (сумеречное) зрение



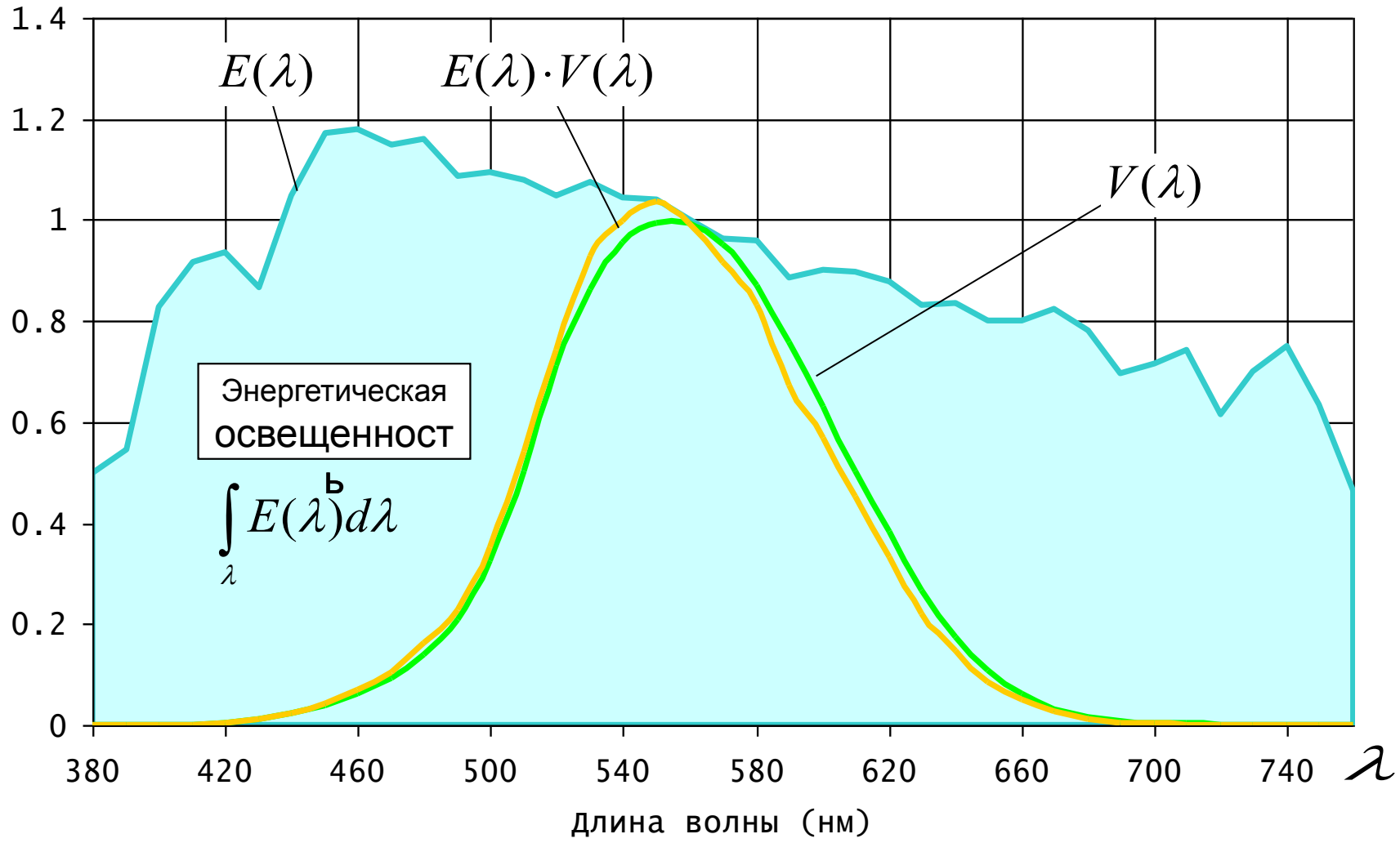
Фотометрические величины



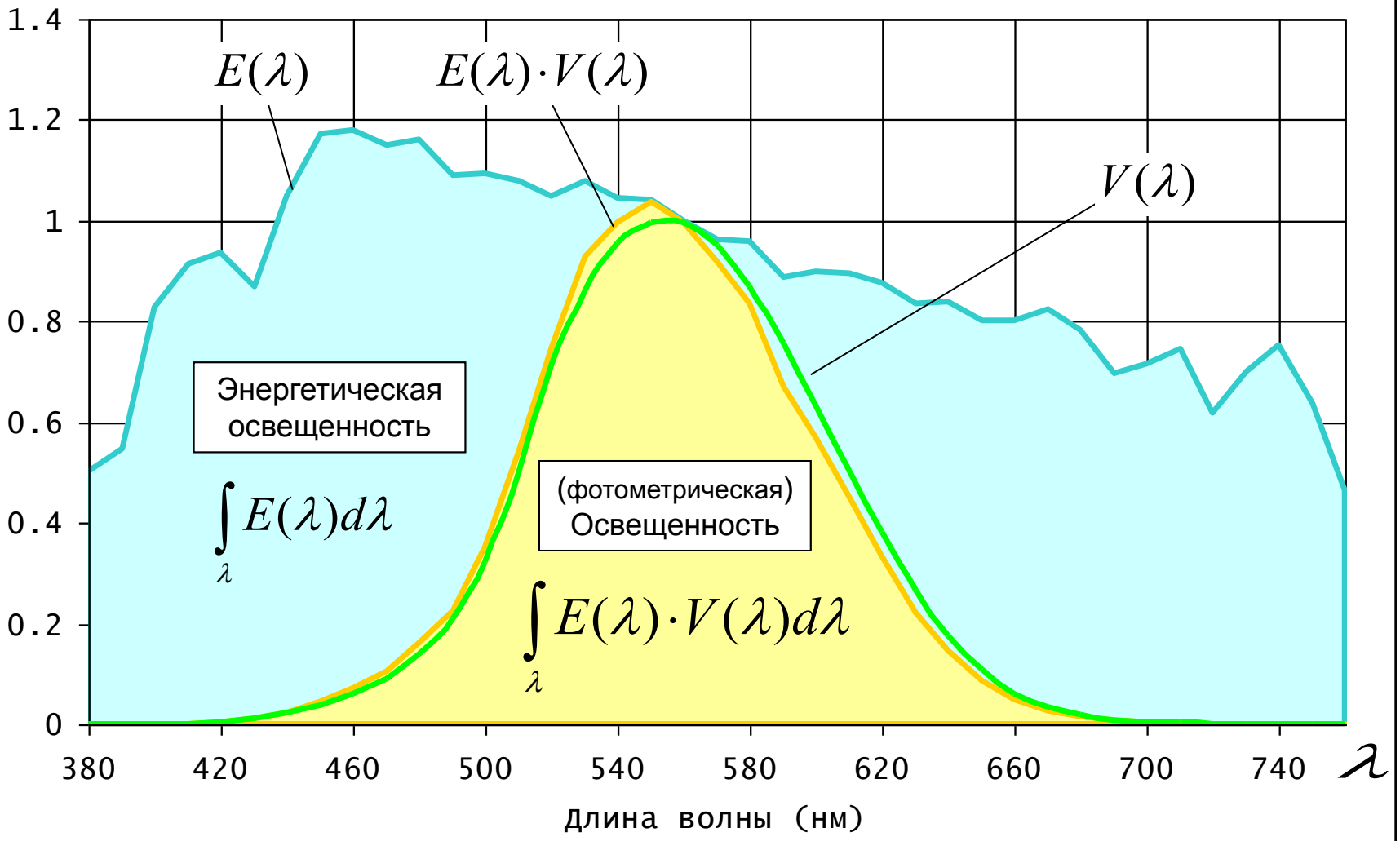
Фотометрические величины



Фотометрические величины



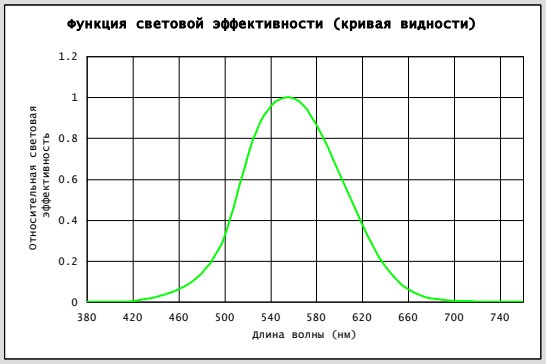
Фотометрические величины



Связь радиометрических и фотометрических величин

Радиометрические (энергетические) величины

Вт, Вт/м^2,
Вт/ср, Вт/м^2/ср



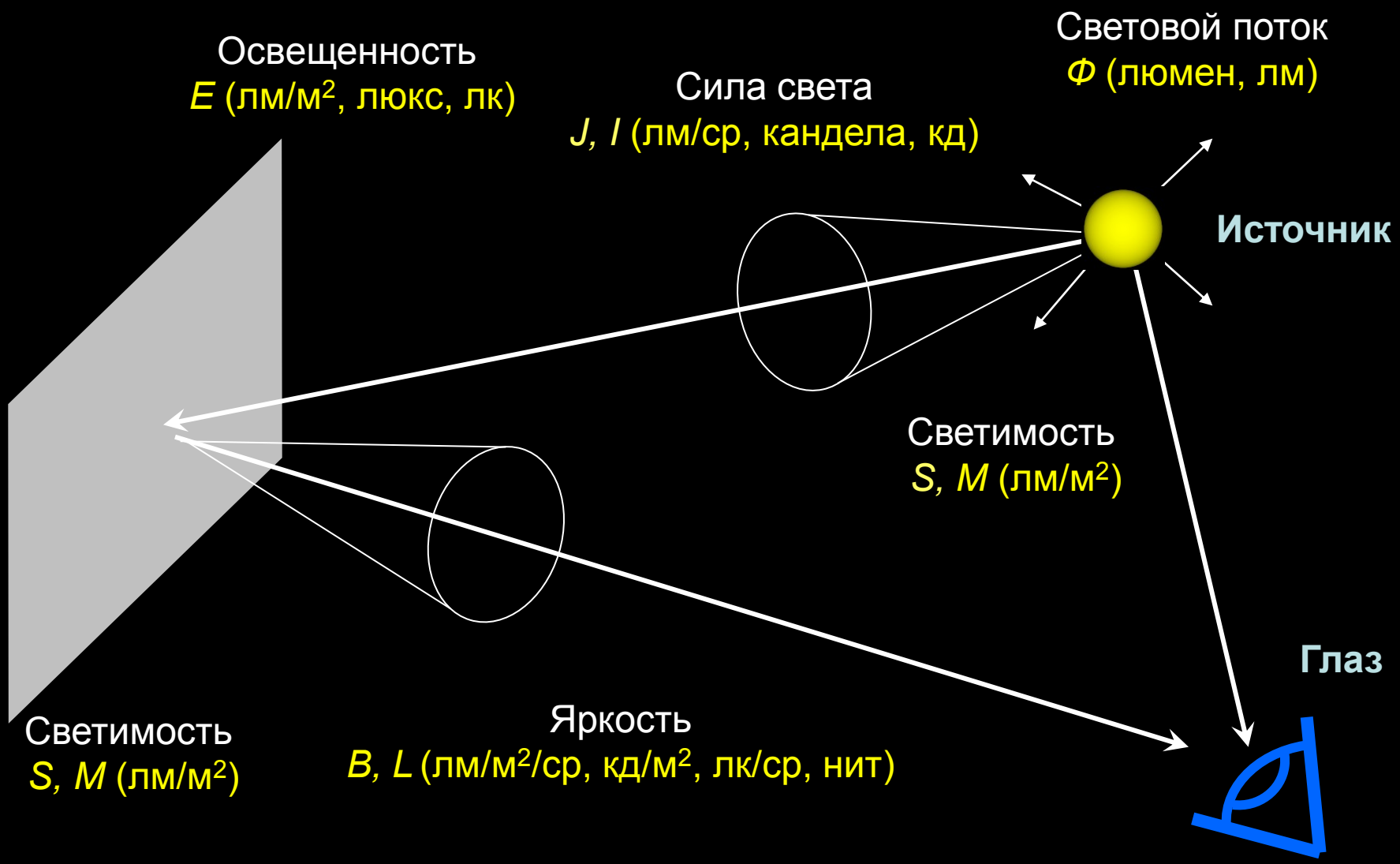
Функция световой эффективности

Фотометрические величины

лм, лм/м^2,
лм/ср, лм/м^2/ср

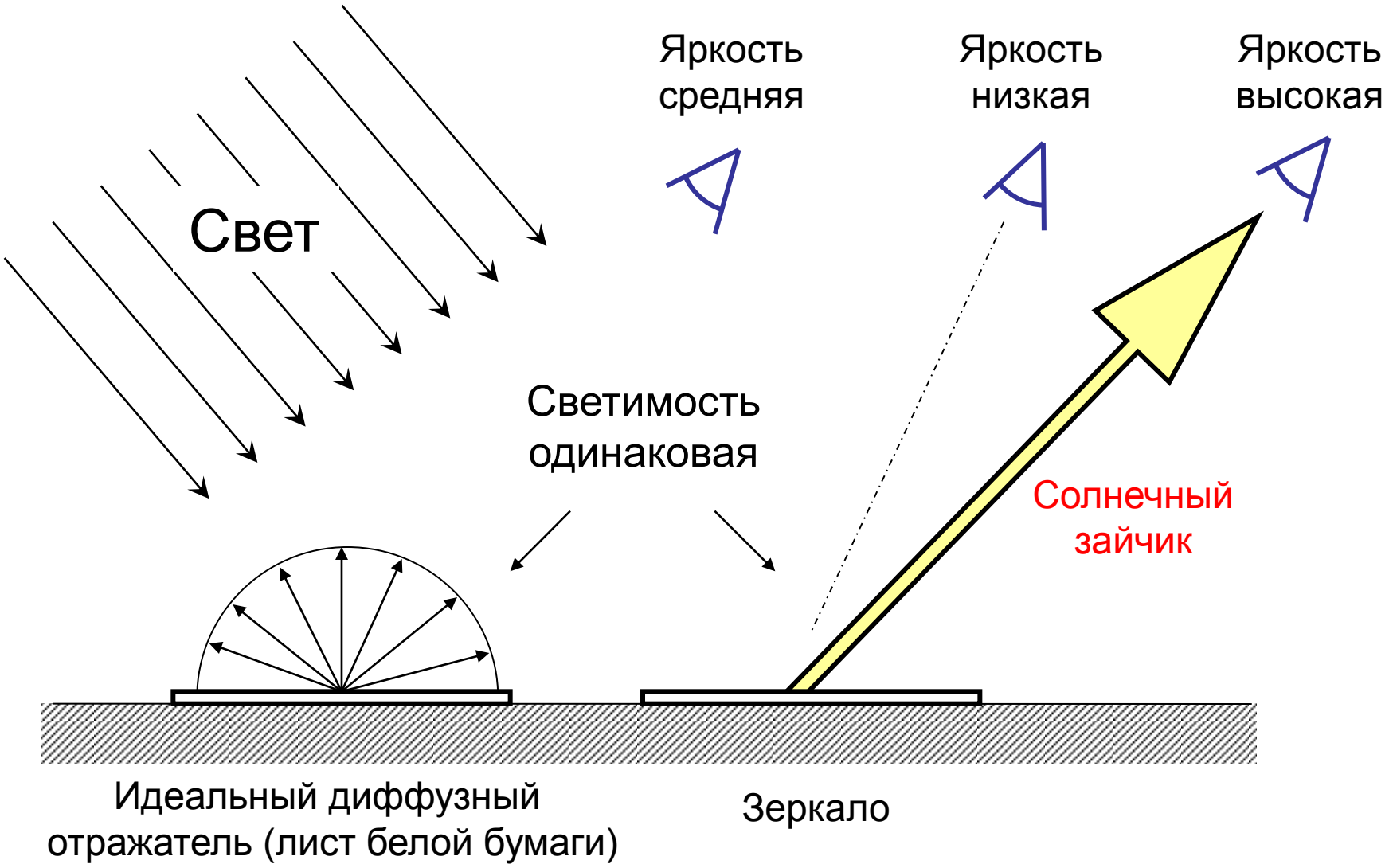
$$\Phi = K_m \int_{\lambda=380}^{\lambda=740} \Phi_e(\lambda) \cdot V(\lambda)\, d\lambda$$

$$K_m = 683 \frac{лм}{Вт}$$



Фотометрические величины и единицы измерения

Яркость.
Важность учета направления (телесного угла)



Пример. Сила света *J* лампы накаливания мощностью (электрической) 100Вт равна 100кд (свечей). Каков коэффициент полезного действия лампы?

$$Полный\ световой\ поток:$$

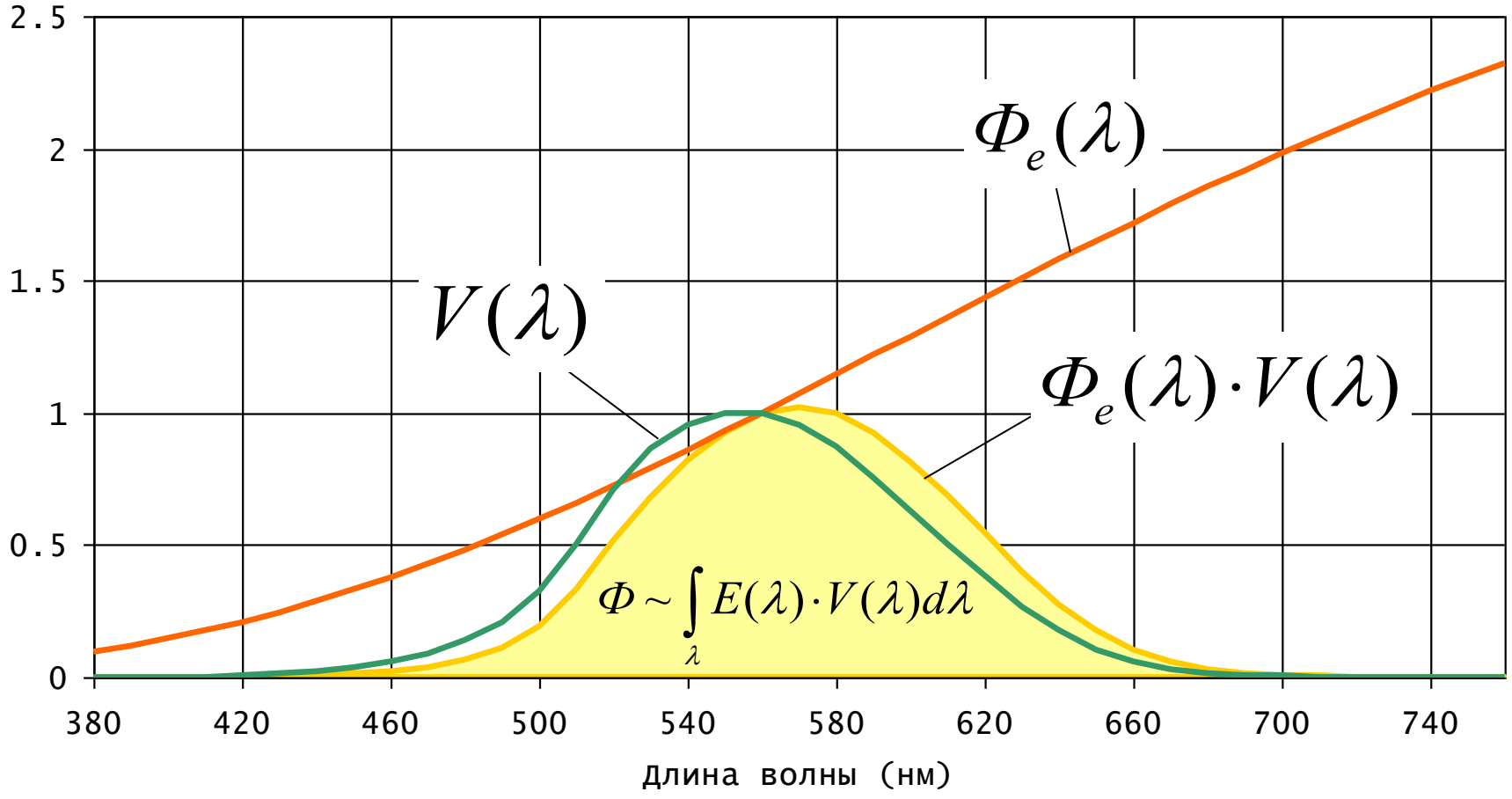
$$Ф = 4 \cdot \pi \cdot J = 4 \cdot 3.14 \cdot 100 = 1256 лм$$

$$1 Вт \sim 683 лм;$$

$$100 Вт \sim 100 \cdot 683 = 68300 лм;$$

$$к.п.д. = \frac{1256}{68300} \cdot 100\% = 1{,}84\%$$

Ответ зрительной системы. Осветитель А



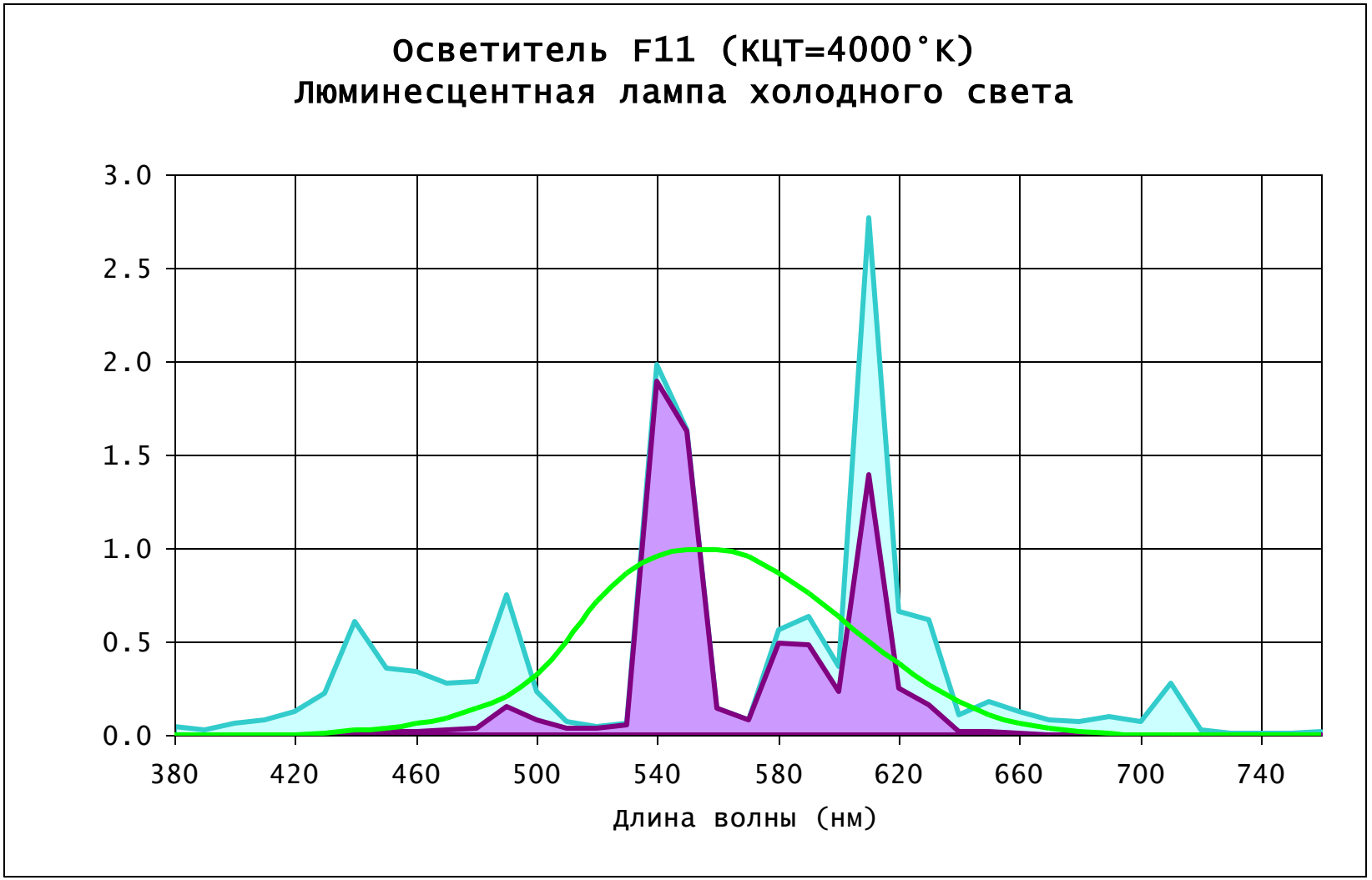
Пример 2. Говорят, что энергосберегающие лампы в 5 раз эффективней ламп накаливания. Не врут?

Лампа энергосберегающая GP 20W.

На упаковке:

- Ватт = 20
- Люмен = 1250
- 2700К Теплый свет

Вывод: не врут!


Осветитель F11 (КЦТ=4000°K)
Люминесцентная лампа холодного света
3.0
2.5
2.0
1.5
1.0
0.5
0.0
380
420
460
500
540
580
620
660
700
740
Длина волны (нм)

Светоотдача

Лампа накаливания 100Вт:

1250лм/100Вт = 12,5 лм/Вт

Энергосберегающая лампа GP 20W:

1250лм/20Вт = 62,5 лм/Вт

Светоотдача

Тип источника	Световая отдача (лм/вт)	Относительная световая отдача
Лампа накаливания 100 Вт	13.8	2.0%
Лампа накаливания 200 Вт	15.2	2.2%
Галогеновая лампа 500 Вт	19.8	2.9%
Светодиод	10–200	1.5–15%
Люминесцентная лампа	40–104	6–12%

Вопрос: Каков теоретический максимум светоотдачи?

Википедия

Светоотдача

Тип источника	Световая отдача (лм/вт)	Относительная световая отдача
Лампа накаливания 100 Вт	13.8	2.0%
Лампа накаливания 200 Вт	15.2	2.2%
Светодиод	10–200	1.5–15%
Люминесцентная лампа	40–104	6–12%
Теоретический максимум	**683**	**100%**

Вопрос: Каков должен быть спектр источника с максимальной светоотдачей?

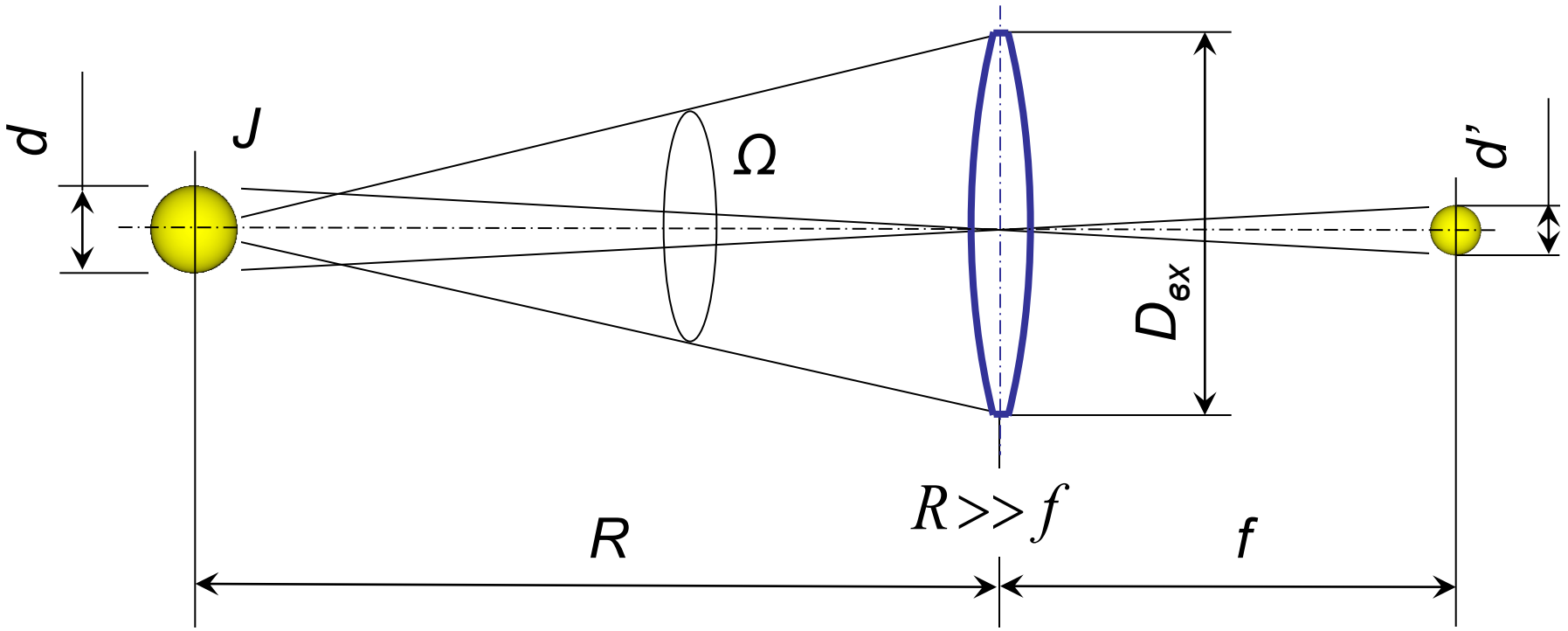
Википедия

Светоотдача

Тип источника	Световая отдача (лм/вт)	Относительная световая отдача
Лампа накаливания 100 Вт	13.8	2.0%
Лампа накаливания 200 Вт	15.2	2.2%
Галогеновая лампа 100 Вт	16.7	2.4%
Галогеновая лампа 200 Вт	17.6	2.6%
Галогеновая лампа 500 Вт	19.8	2.9%
Светодиод	10–200	1.5–15%
Люминесцентная лампа	40–104	6–12%
Теоретический максимум (монохроматический зелёный свет 555 нм)	**683**	**100%**

Википедия

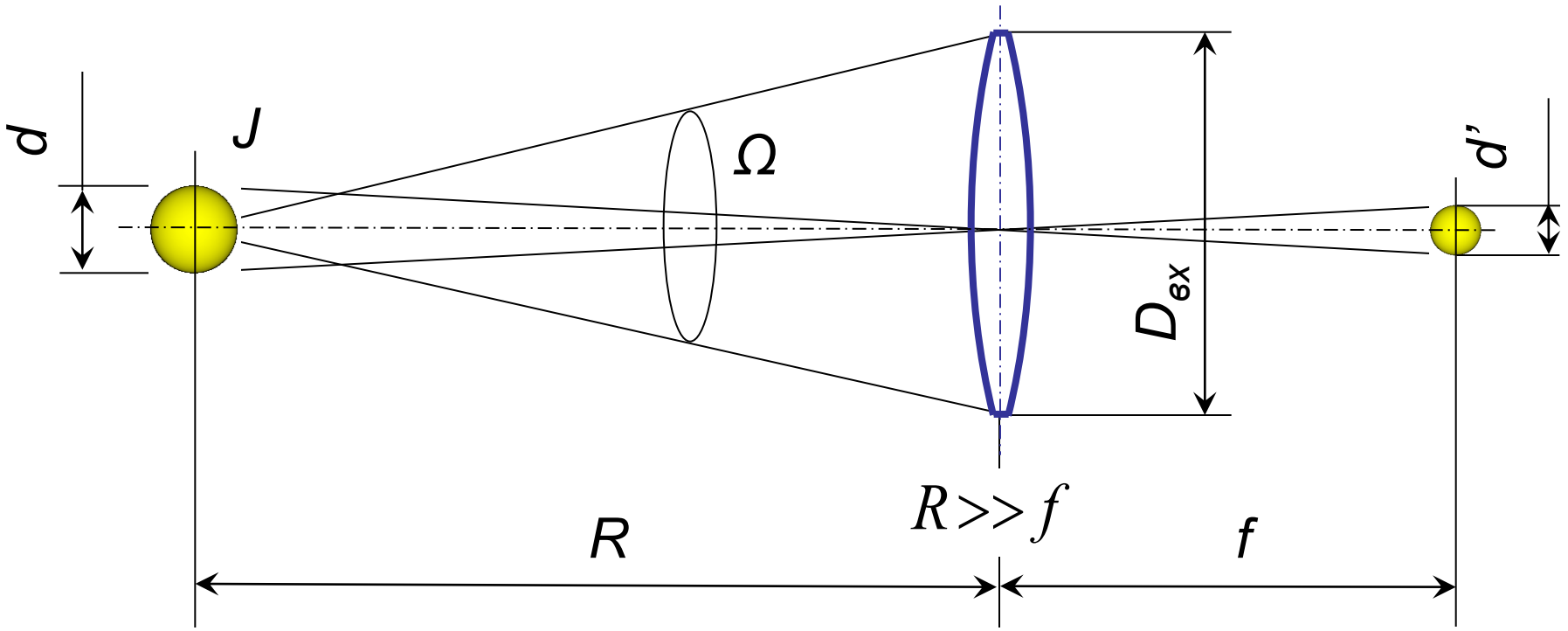
Инвариантность яркости



$$\Phi = J \cdot \Omega; \qquad \Omega = \frac{S_{вх}}{R^2} = \frac{\pi D_{вх}^2}{4R^2}; \qquad \Phi = \frac{J \pi D_{вх}^2}{4R^2};$$

$$E' = \frac{\Phi}{S'} = \frac{4\Phi}{\pi d'^2} = \frac{J D_{вх}^2}{d'^2 R^2}; \qquad d' = \frac{f \cdot d}{R}$$

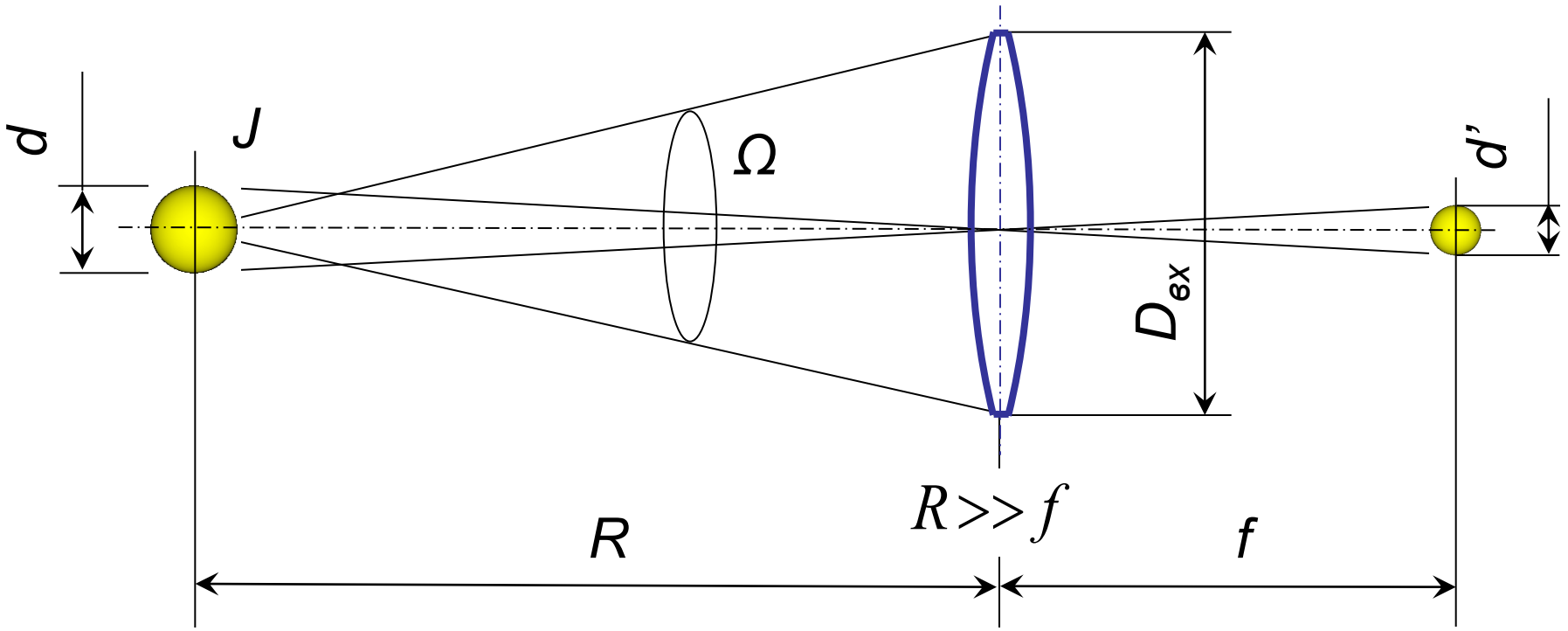
Освещенность сенсора



$$E' = \frac{JD_{вх}^2}{d'^2 R^2}; \qquad d' = \frac{f \cdot d}{R}; \qquad E' = \frac{JD_{вх}^2 R^2}{f^2 d^2 R^2} = \frac{JD_{вх}^2}{f^2 d^2};$$

$$\frac{D_{вх}^2}{f^2} = \frac{1}{k^2}; \qquad E' = \frac{JD_{вх}^2}{f^2 d^2} = \frac{J}{k^2 d^2}$$

Освещенность сенсора (продолжение)

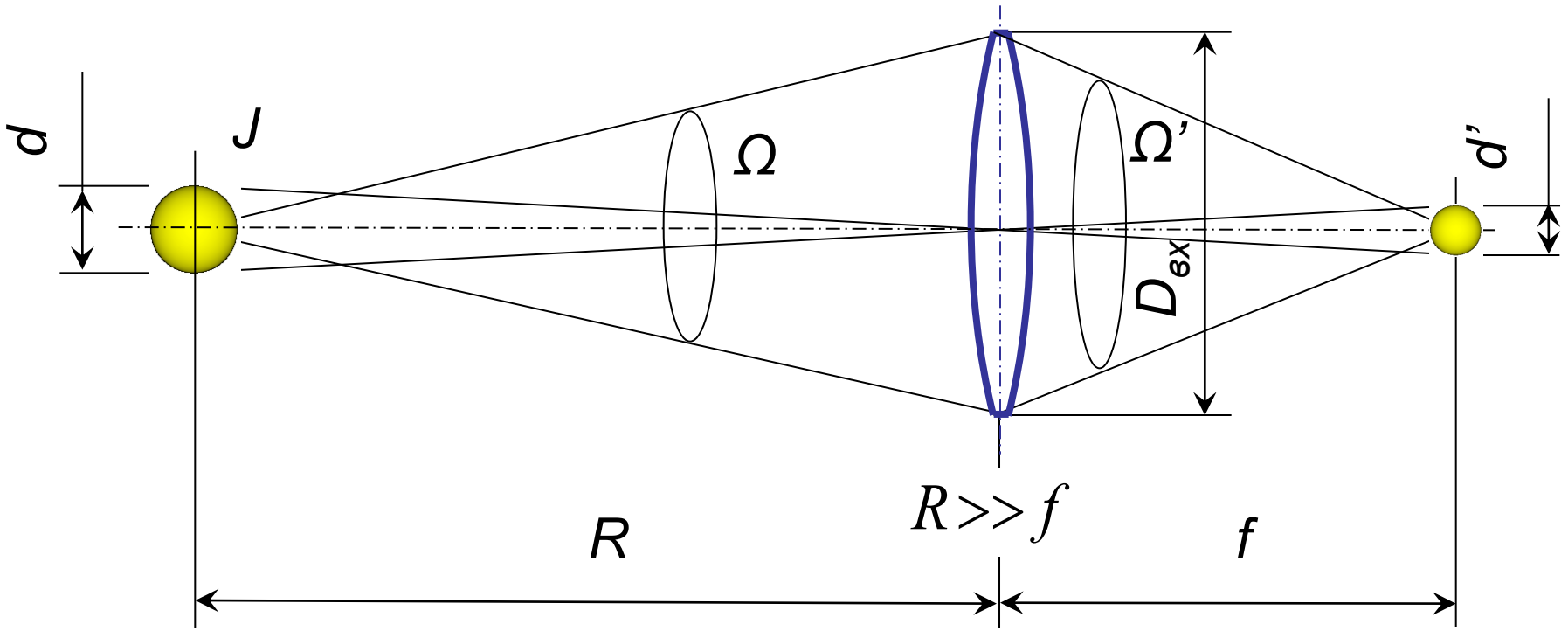


$$E' = \frac{J}{k^2 d^2} \qquad S = \frac{\pi d^2}{4};\ d^2 = \frac{4S}{\pi};$$

$$E' = \frac{J}{k^2 d^2} = \frac{\pi J}{4k^2 S}; \qquad B = \frac{J}{S};$$

$$E' = \frac{\pi B}{4k^2}$$

Яркость изображения



$$\Phi = J \cdot \Omega; \quad \Omega = \frac{S_{вх}}{R^2}; \quad \Phi' = \Phi = J\frac{S_{вх}}{R^2}; \quad B' = \frac{\Phi'}{\Omega' S'} = \frac{JS_{вх}}{R^2 \Omega' S'};$$

$$\Omega' = \frac{S_{вх}}{f^2}; \quad S' = S\frac{f^2}{R^2}; \quad B' = \frac{JS_{вх}}{R^2 \Omega' S'} = \frac{JS_{вх} f^2 R^2}{R^2 S_{вх} S f^2} = \frac{J}{S} = B;$$

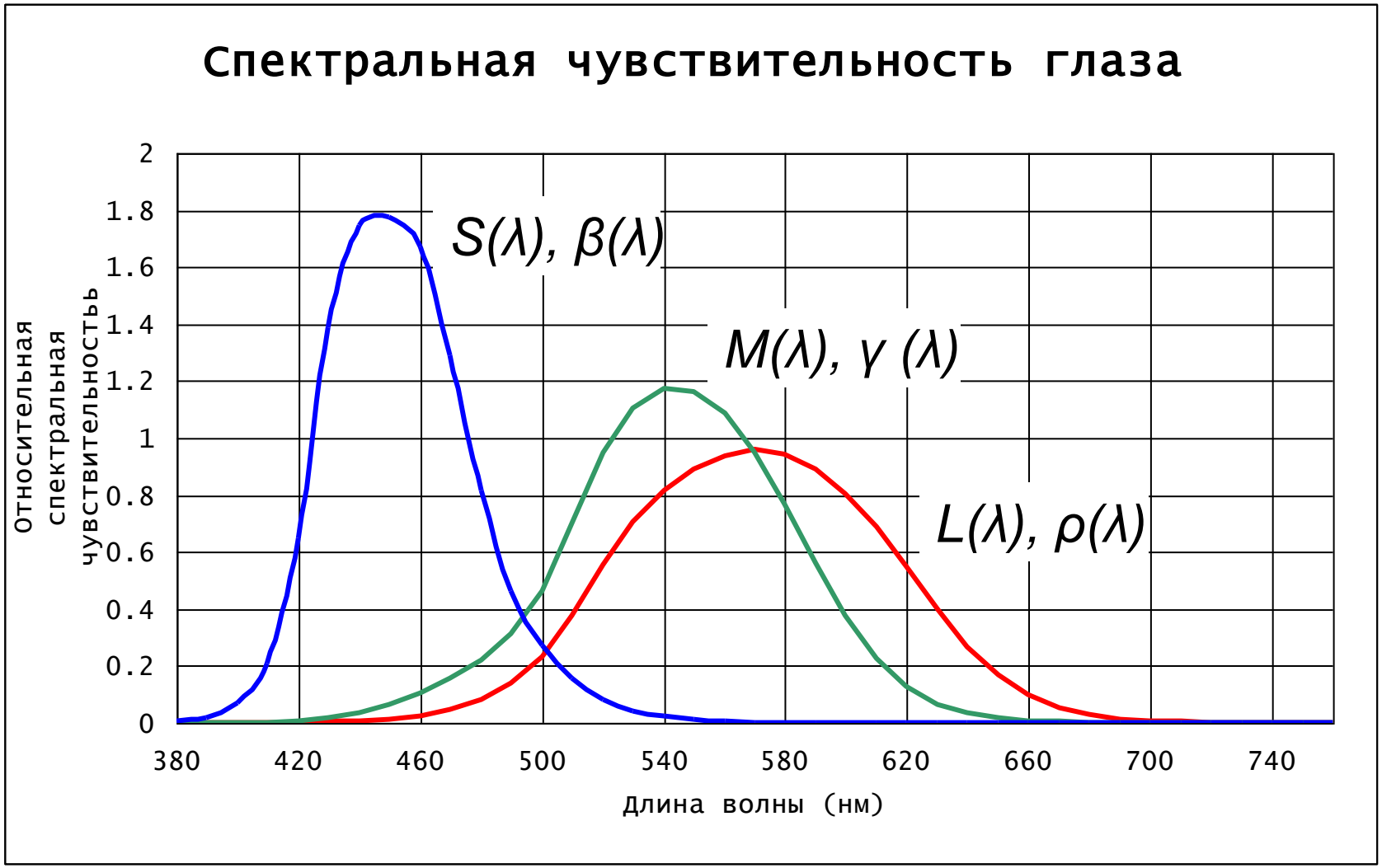
Яркость и освещенность

- Освещенность сенсора пропорциональна яркости объекта.
- Реакция сенсора пропорциональна его освещенности.
- Освещенность изображения объекта не зависит от расстояния до объекта.
- Яркость изображения равна яркости объекта (при отсутствии потерь). Инвариантность яркости

Цвет

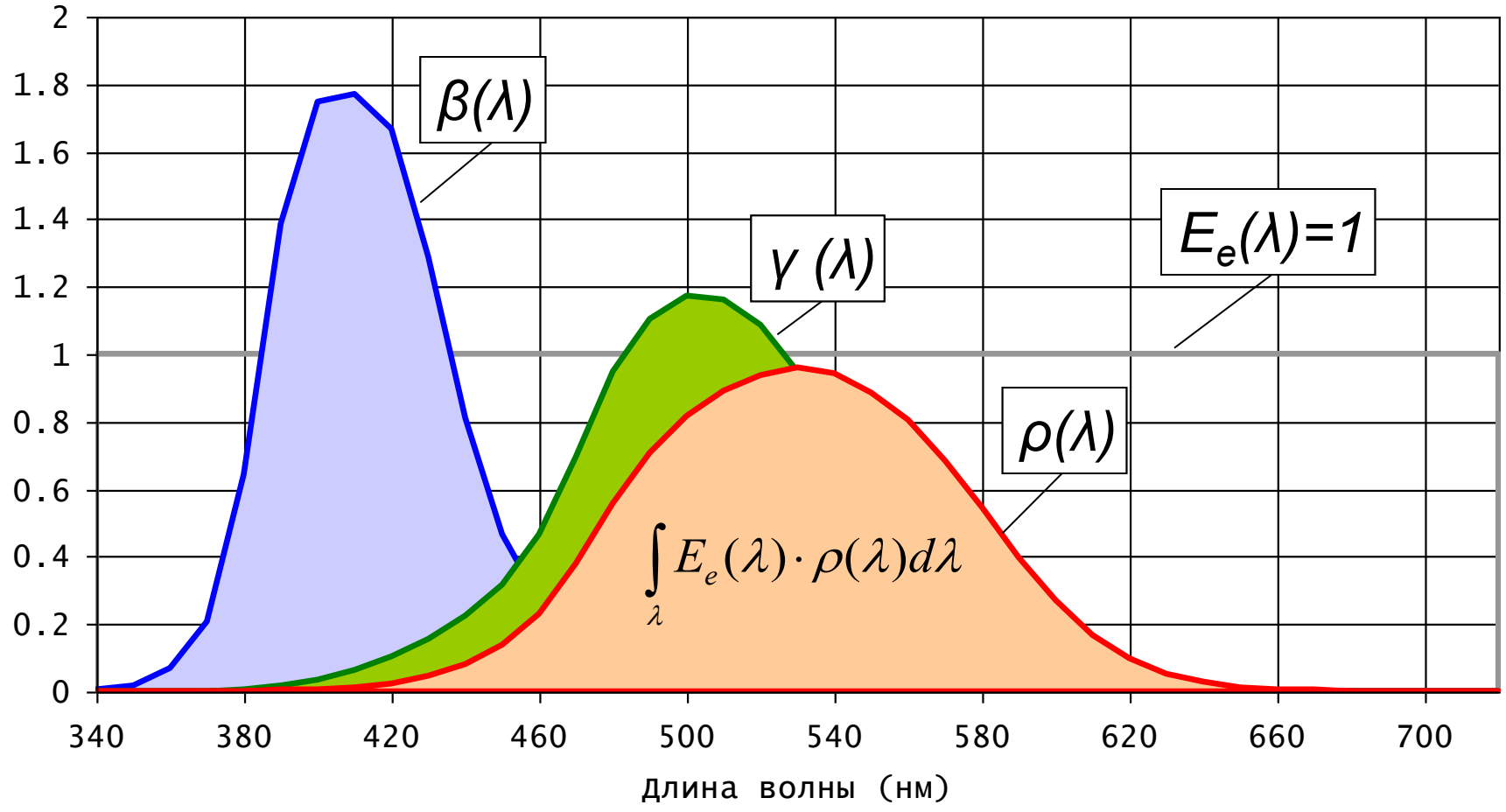
Колориметрия

Спектральная чувствительность колбочек (по Е.Н. Юстовой)



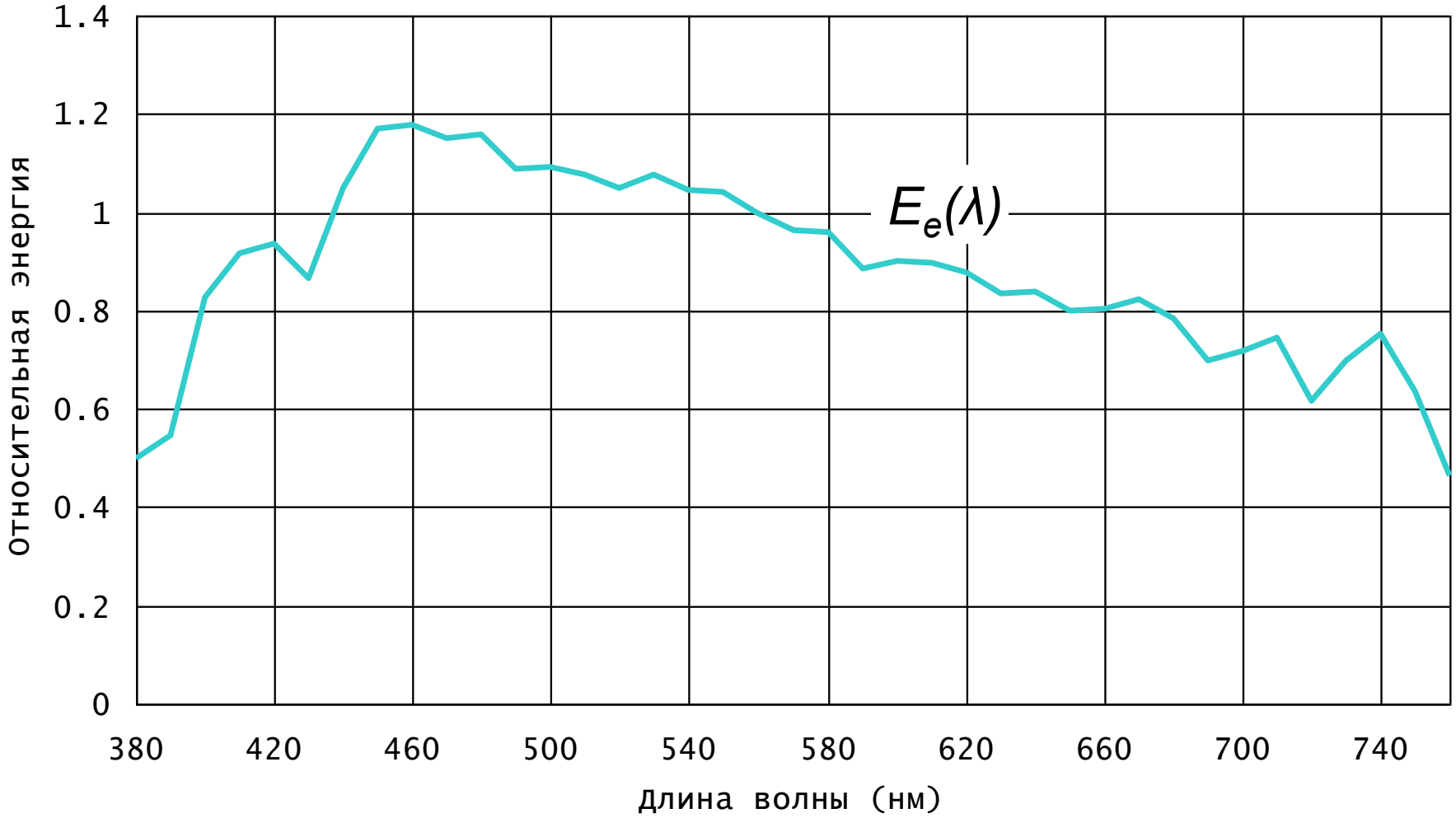
Нормировка: $\int_{\lambda} \rho(\lambda)d\lambda = \int_{\lambda} \gamma(\lambda)d\lambda = \int_{\lambda} \beta(\lambda)d\lambda$

Ответ зрительной системы. Осветитель E

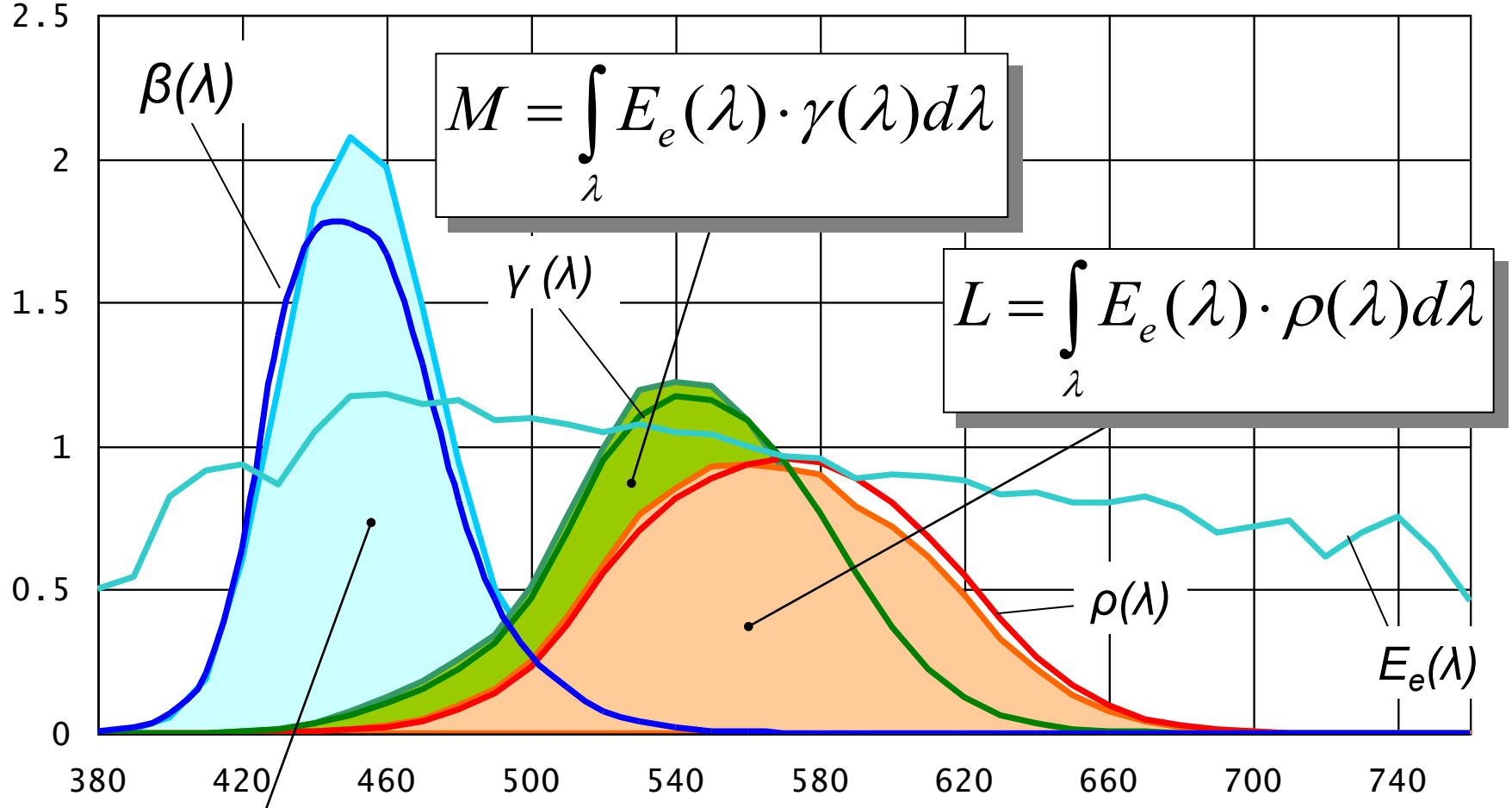


$$\int_\lambda E_e(\lambda)\cdot\rho(\lambda)d\lambda = \int_\lambda E_e(\lambda)\cdot\gamma(\lambda)d\lambda = \int_\lambda E_e(\lambda)\cdot\beta(\lambda)d\lambda$$

Спектральное распределение энергии осветителя D65



Ответ зрительной системы.Осветитель D65



Цветовые координаты

$$L = \int_{\lambda} E_e(\lambda) \cdot \rho(\lambda) d\lambda$$

$$M = \int_{\lambda} E_e(\lambda) \cdot \gamma(\lambda) d\lambda$$

$$S = \int_{\lambda} E_e(\lambda) \cdot \beta(\lambda) d\lambda$$

Цветовые координаты. Следствия

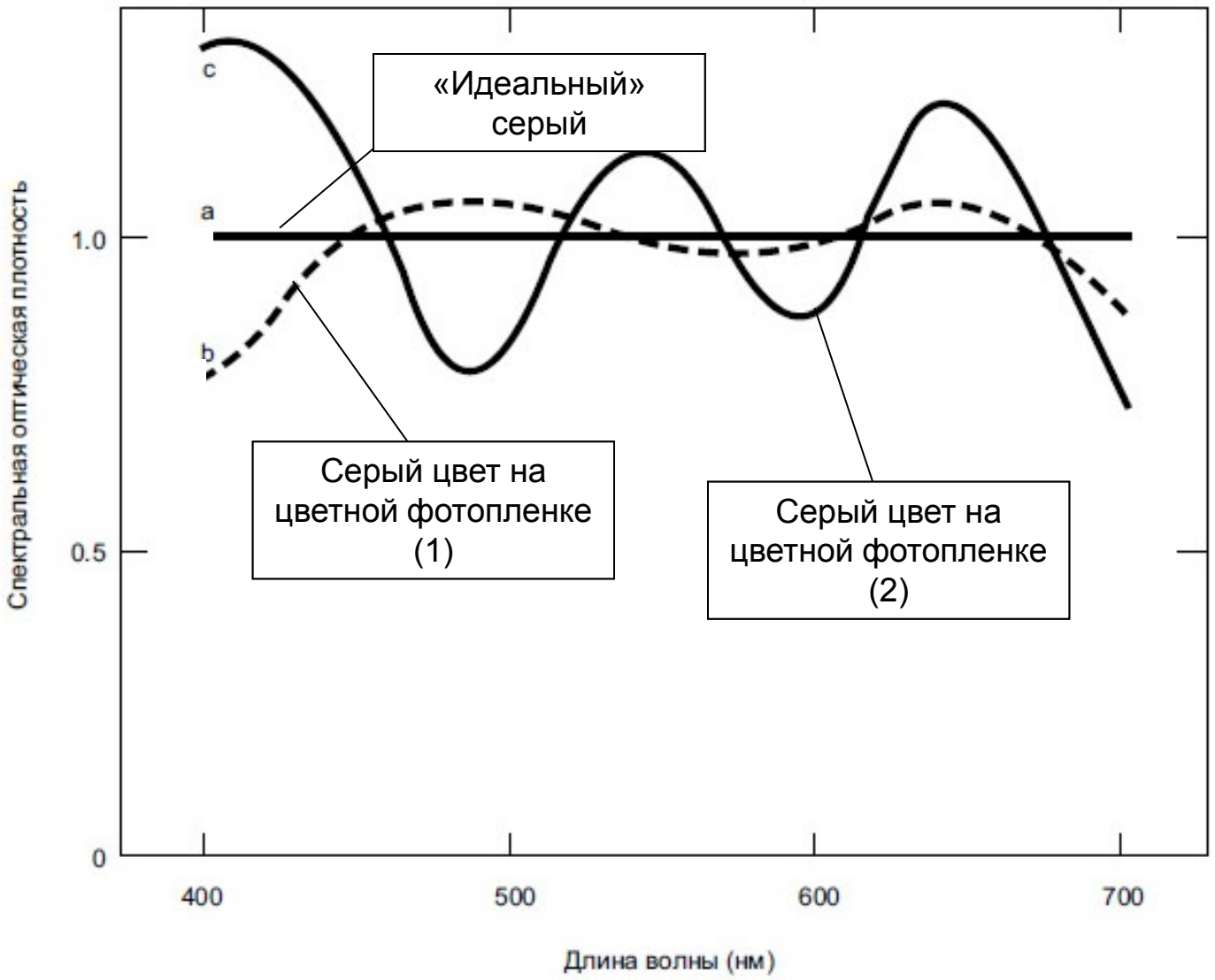
- Ответ зрительной системы характеризуются тремя числами (координатами цвета). *Первый закон Грассмана (трехмерность).*
- Если известны *L, M, S*, знать спектр необязательно.
- Разным спектрам могут соответствовать одинаковые координаты цвета – метамерия (метамеризм).
- *L, M, S* (фундаментальные трехстимульные значения, колбочковые ответы) однозначно характеризуют цвет в стандартных условиях наблюдения (стандартное окружение, стандартная яркость, стандартное поле зрения, ...).
- Разные цветовые ощущения при одинаковых *L, M, S* могут быть обусловлены механизмами адаптации зрения.

Дополнительные главы компьютерной графики

Лекция 2

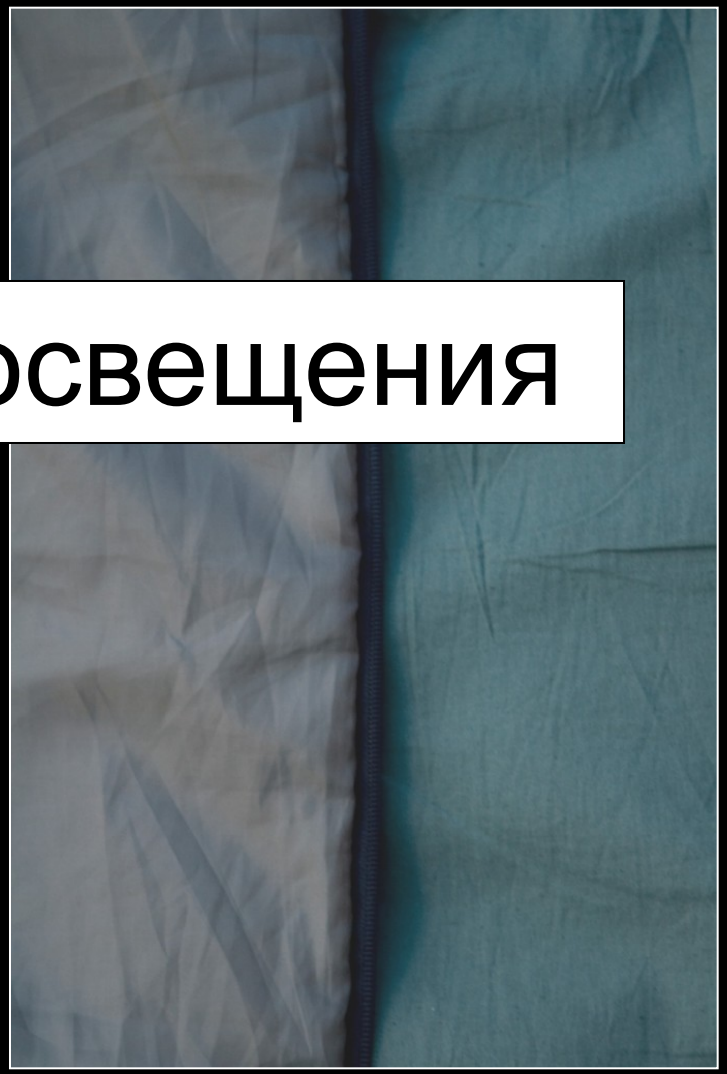
Copyright © С. Свердлов, 2009-2017
http://www.uni-vologda.ac.ru/~c3c

Пример метамерных спектров



Р. Хант

Практический пример метамерии



Метамерия освещения

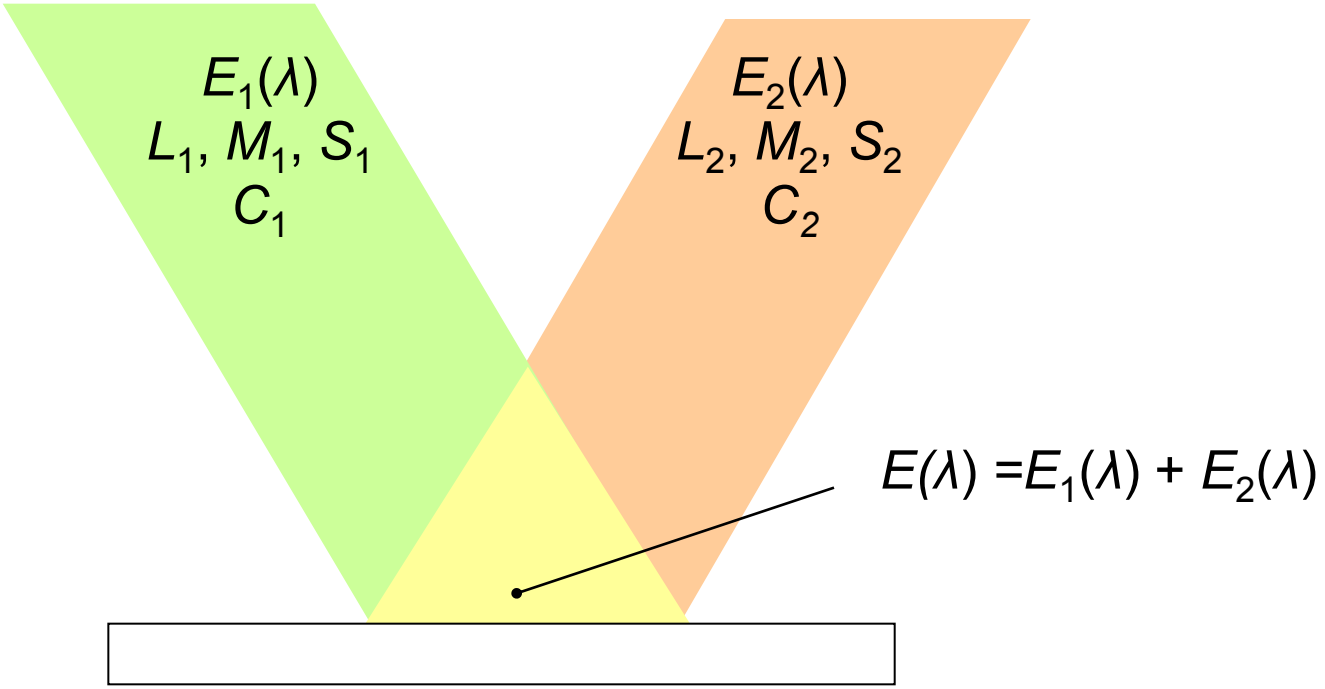
Солнечный свет
(цвета одинаковы)

Свет внутри синей палатки
(цвета разные)

- Метамерия – основа цветовоспроизведения

- Краски, цветная фотопленка, цветные отпечатки, цветные мониторы воспроизводят не спектр, а эквивалентные цветовые стимулы.

Цвет смеси



$$L_1 = L(C_1) = \int_\lambda E_1(\lambda)\rho(\lambda)d\lambda \qquad L_2 = L(C_2) = \int_\lambda E_2(\lambda)\rho(\lambda)d\lambda$$

$$M_1 = M(C_1) = \int_\lambda E_1(\lambda)\gamma(\lambda)d\lambda \qquad M_2 = M(C_2) = \int_\lambda E_2(\lambda)\gamma(\lambda)d\lambda$$

$$S_1 = S(C_1) = \int_\lambda E_1(\lambda)\beta(\lambda)d\lambda \qquad S_2 = S(C_2) = \int_\lambda E_2(\lambda)\beta(\lambda)d\lambda$$

Цвет смеси

$$L(C) = L(C_1 + C_2)$$

Цветовые координаты смеси равны сумме цветовых координат составляющих. *Второй закон Грассмана (аддитивность).*

Третий закон Грассмана (пропорциональность)

$$L(kC) = \int_{\lambda} k \cdot E(\lambda)\rho(\lambda)d\lambda = k\int_{\lambda} E(\lambda)\rho(\lambda)d\lambda = kL$$

$$M(kC) = \int_{\lambda} k \cdot E_1(\lambda)\gamma(\lambda)d\lambda = k\int_{\lambda} E(\lambda)\gamma(\lambda)d\lambda = kM$$

$$S(kC) = \int_{\lambda} k \cdot E(\lambda)\beta(\lambda)d\lambda = k\int_{\lambda} E(\lambda)\beta(\lambda)d\lambda = kS$$

Изменение энергии излучения (яркости) в *k* раз пропорционально меняет цветовые координаты

$$C_1(L_1, M_1, S_1) = C_2(L_2, M_2, S_2)$$
$$kC_1 = kC_2$$

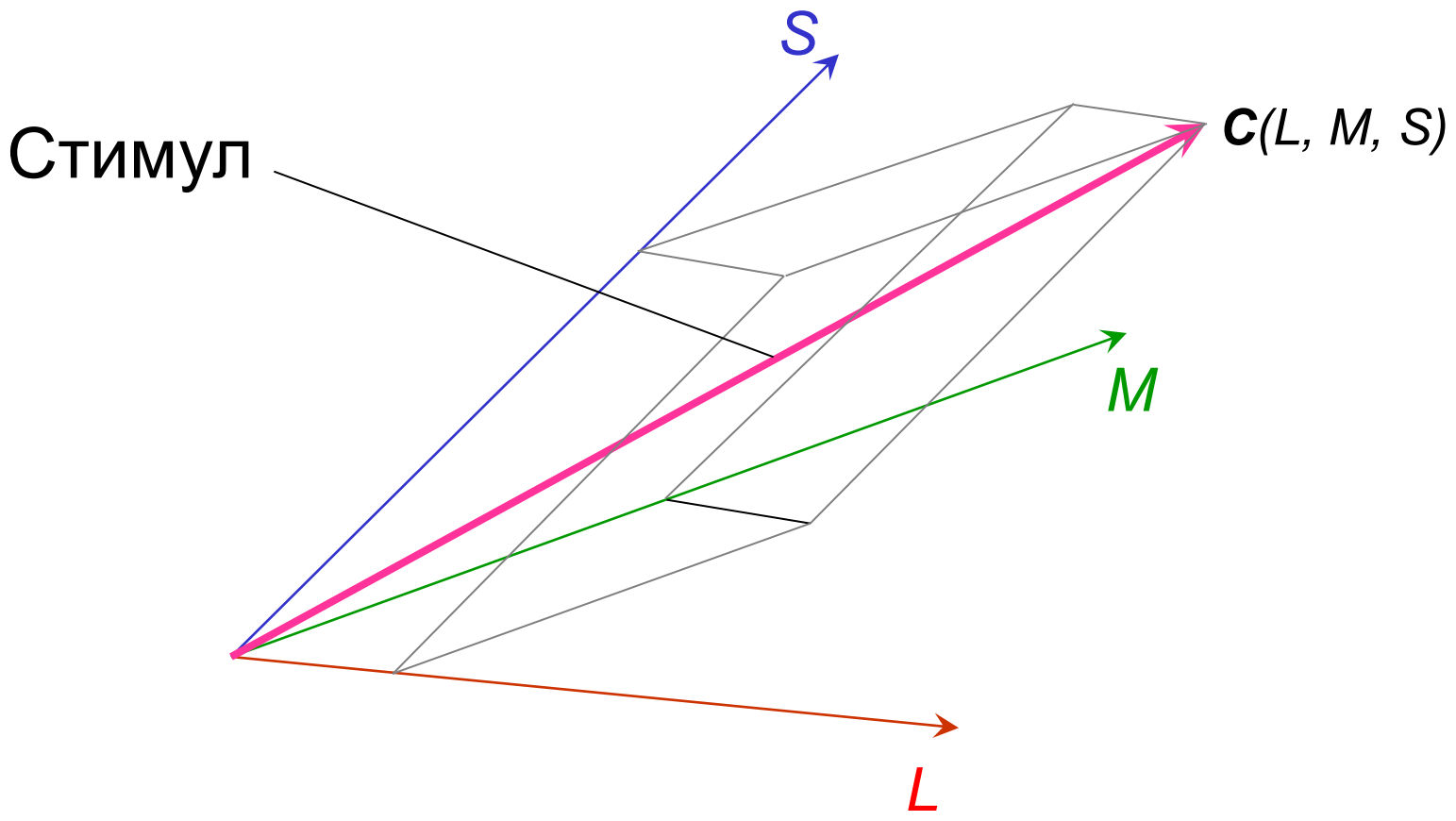
Цветовые равенства инвариантны к уровню яркости

Герман Гюнтер Грассман (нем. *Hermann Günther Grassmann*, 1809—1877) — физик, математик и филолог.

Законы Грассмана были сформулированы на основе эмпирических наблюдений.

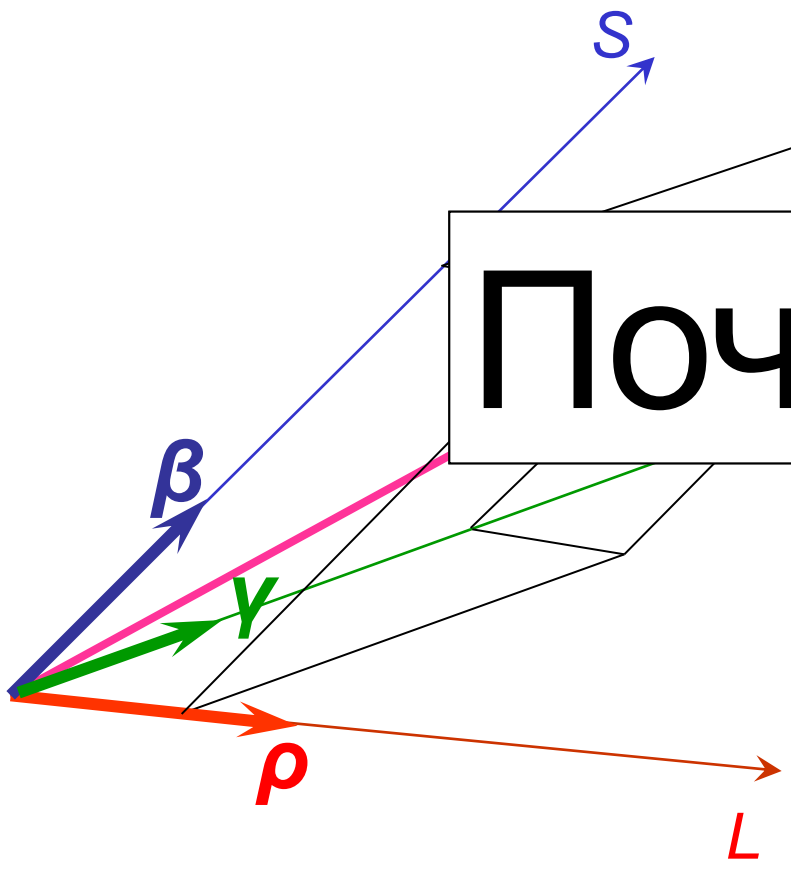
Цветовое пространство

Трехстимульные значения (стимулы) ***C***(*L*, *M*, *S*) образуют 3-мерное линейное векторное пространство



Базис пространства *LMS*

Базис пространства *LMS* образуют единичные векторы (орты) осей *L*, *M*, *S*: **ρ**, **γ**, **β**

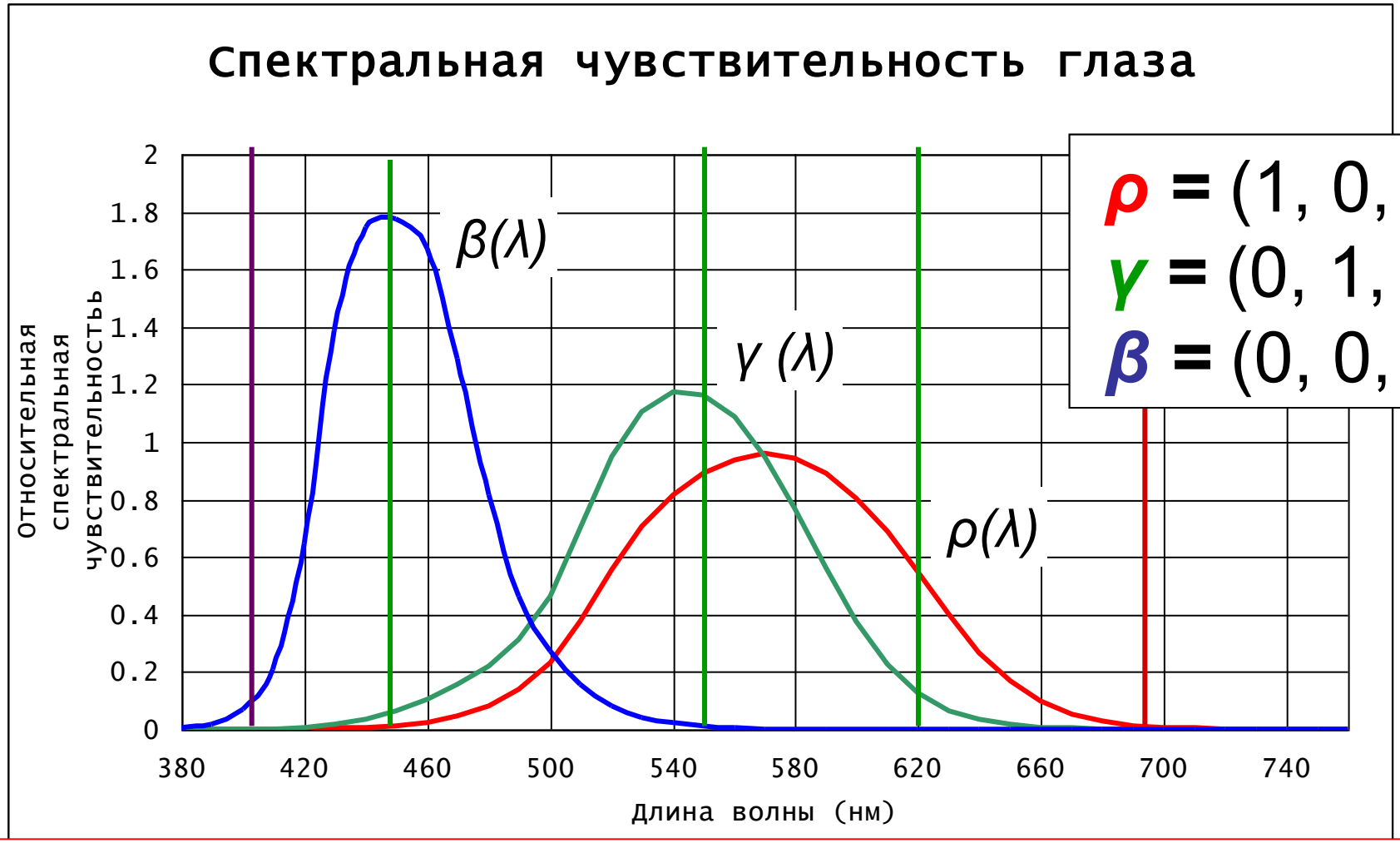


Почему?

ρ = (1, 0, 0)
γ = (0, 1, 0)
β = (0, 0, 1)

Базисные (кардинальные) стимулы пространства *LMS* физически нереализуемы!

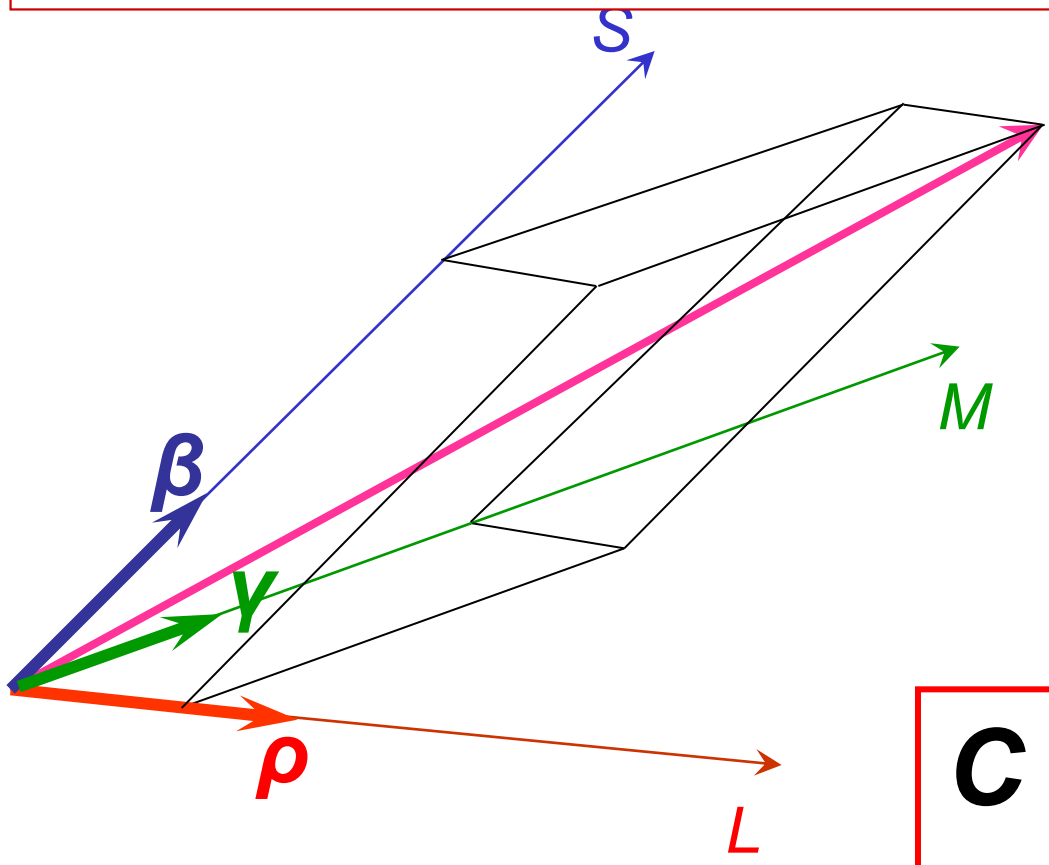
Базис пространства *LMS*



Не существует спектра, дающего такие **ρ**, **γ**, **β**

Базис пространства *LMS*

Первый закон Грассмана (полная формулировка)
Каждый цветовой стимул может быть получен как сумма трех линейно-независимых стимулов



$\rho = (1, 0, 0)$
$\gamma = (0, 1, 0)$
$\beta = (0, 0, 1)$

$$C = L\rho + M\gamma + S\beta$$

Что такое цвет?

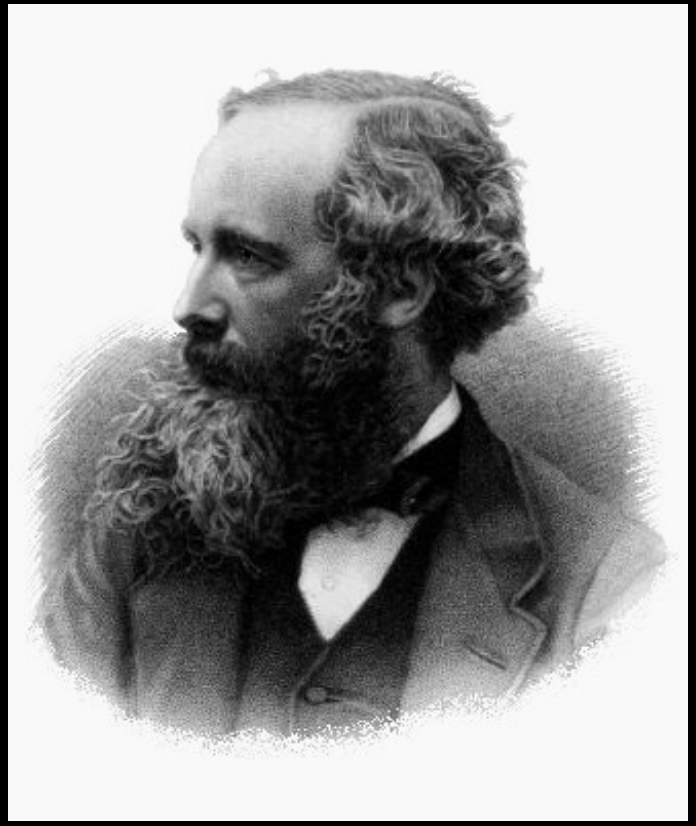
- **Цвет** – группа излучений, производящих одинаковое впечатление. (Э.Шрёдингер)
- **Цвет** – это атрибут зрительного ощущения, содержащий некую комбинацию хроматических и ахроматических компонентов. Этот атрибут может быть описан как по хроматическому компоненту (например: желтый, оранжевый, коричневый, красный, розовый, зеленый, синий, фиолетовый, и т.д.), так и по ахроматическому (белый, серый, черный и т.д.), а также квалифицирован как яркий, тусклый, светлый, темный и т.д. или описан комбинацией перечисленных наименований. («Международный осветительный словарь»)
- **Цвет** – качественная субъективная характеристика электромагнитного излучения оптического диапазона, определяемая на основании возникающего физиологического зрительного ощущения и зависящая от ряда физических, физиологических и психологических факторов. (Википедия)
- **Цвет** в колориметрии есть свойство, общее всем спектральным составам излучения, визуально неразличимым в колориметрических условиях наблюдения. (ГОСТ 13088-67, Н. Д. Нюберг, Е. Н. Юстова)

Цвет ~ Стимул

Михаи́л Васи́льевич Ломоно́сов (8 (19) ноября 1711, деревня Мишанинская, Россия — 4 (15) апреля 1765, Санкт-Петербург, Российская империя) — первый русский учёный-естествоиспытатель мирового значения, энциклопедист, химик и физик;

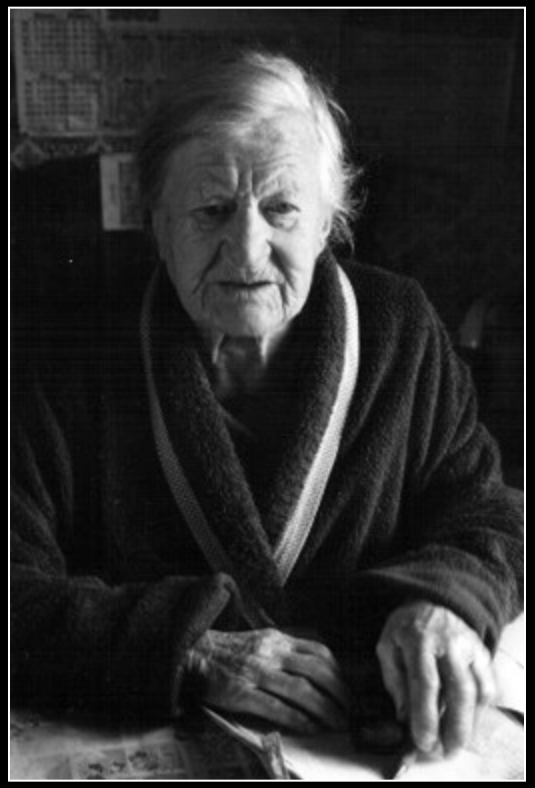
То́мас Юнг (иногда пишут в современном прочтении *Янг*, англ. *Thomas Young*; 13 июня1773, Милвертон, графство Сомерсет — 10 мая 1829, Лондон) — английский физик, врач, астроном и востоковед, один из создателей волновой теории света.

Герман Людвиг Фердинанд фон Гельмгольц (нем. *Hermann von Helmholtz*; 31 августа 1821, Потсдам — 8 сентября 1894, Шарлоттенбург) — немецкий физик, физиолог и психолог.



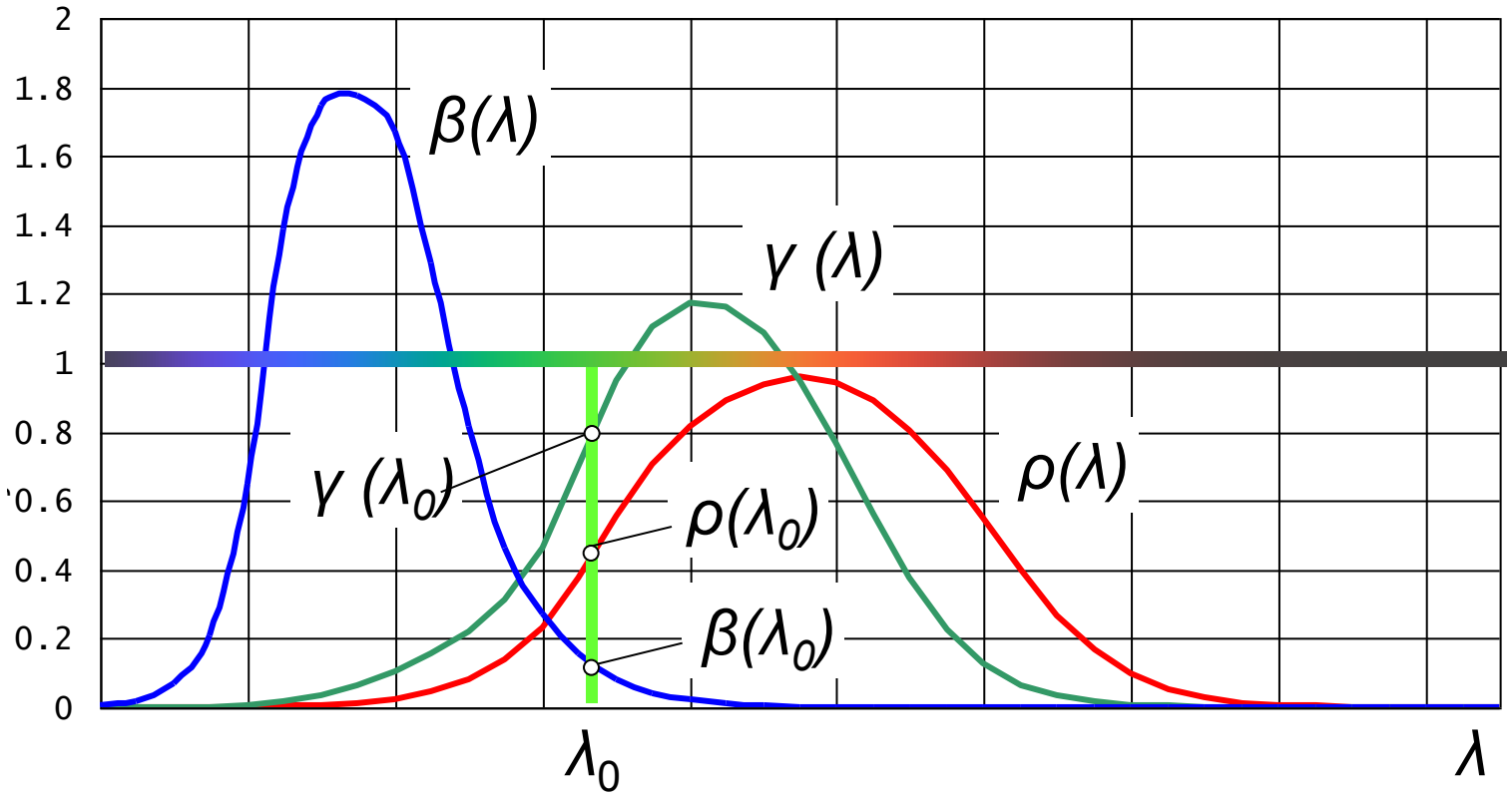
Джеймс Клерк (Кларк) Ма́ксвелл (англ. *James Clerk Maxwell*; 13 июня 1831, Эдинбург — 5 ноября 1879, Кембридж) — британский физик и математик.

Эрвин Рудольф Йозеф Александр Шрёдингер (нем. *Erwin Rudolf Josef Alexander Schrödinger*; 12 августа 1887, Вена — 4 января 1961, там же) — австрийский физик-теоретик, один из создателей квантовой механики. Лауреат Нобелевской премии по физике (1933).

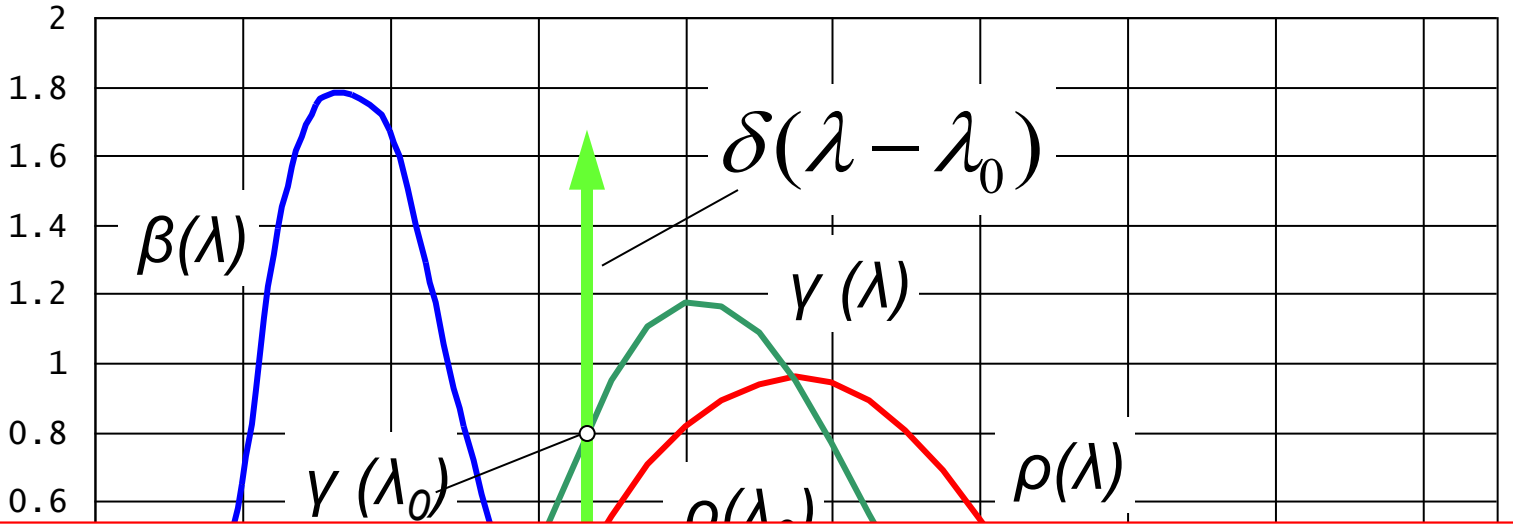


Елизавета Николаевна Юстова (29 октября 1910, Варшава — 15 февраля 2008, Шапки, Ленинградская обл.) — советский и российский учёный, крупный специалист в области колориметрии, доктор технических наук, почётный член Метрологической академии.

Колбочковые ответы на монохроматические стимулы



Колбочковые ответы на монохроматические стимулы



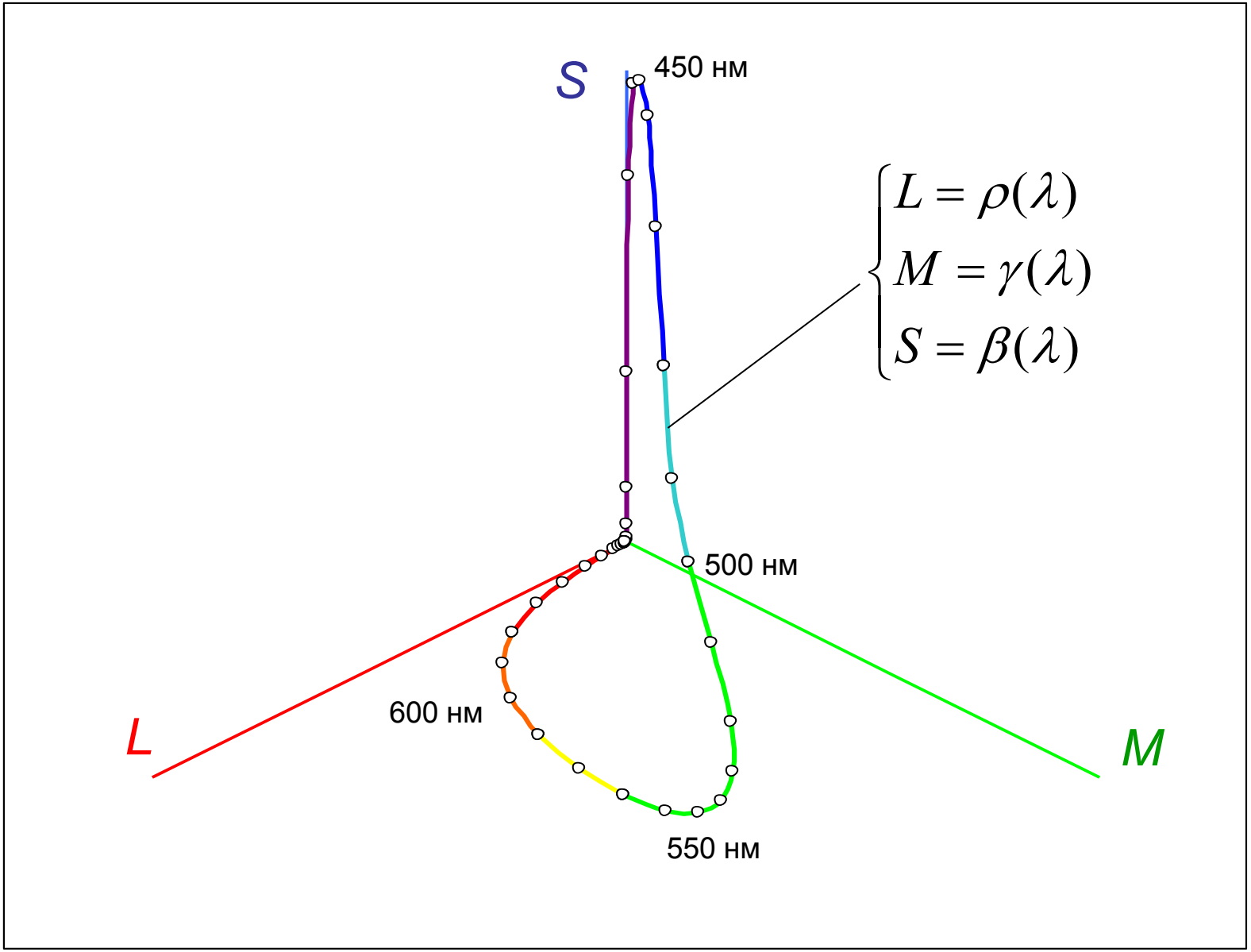
Функции спектральной чувствительности являются параметрическими уравнениями линии (локуса) спектральных цветов в пространстве *LMS*

$$\int_{\lambda} \delta(\lambda-\lambda_0)\cdot\rho(\lambda)d\lambda = \rho(\lambda_0)$$

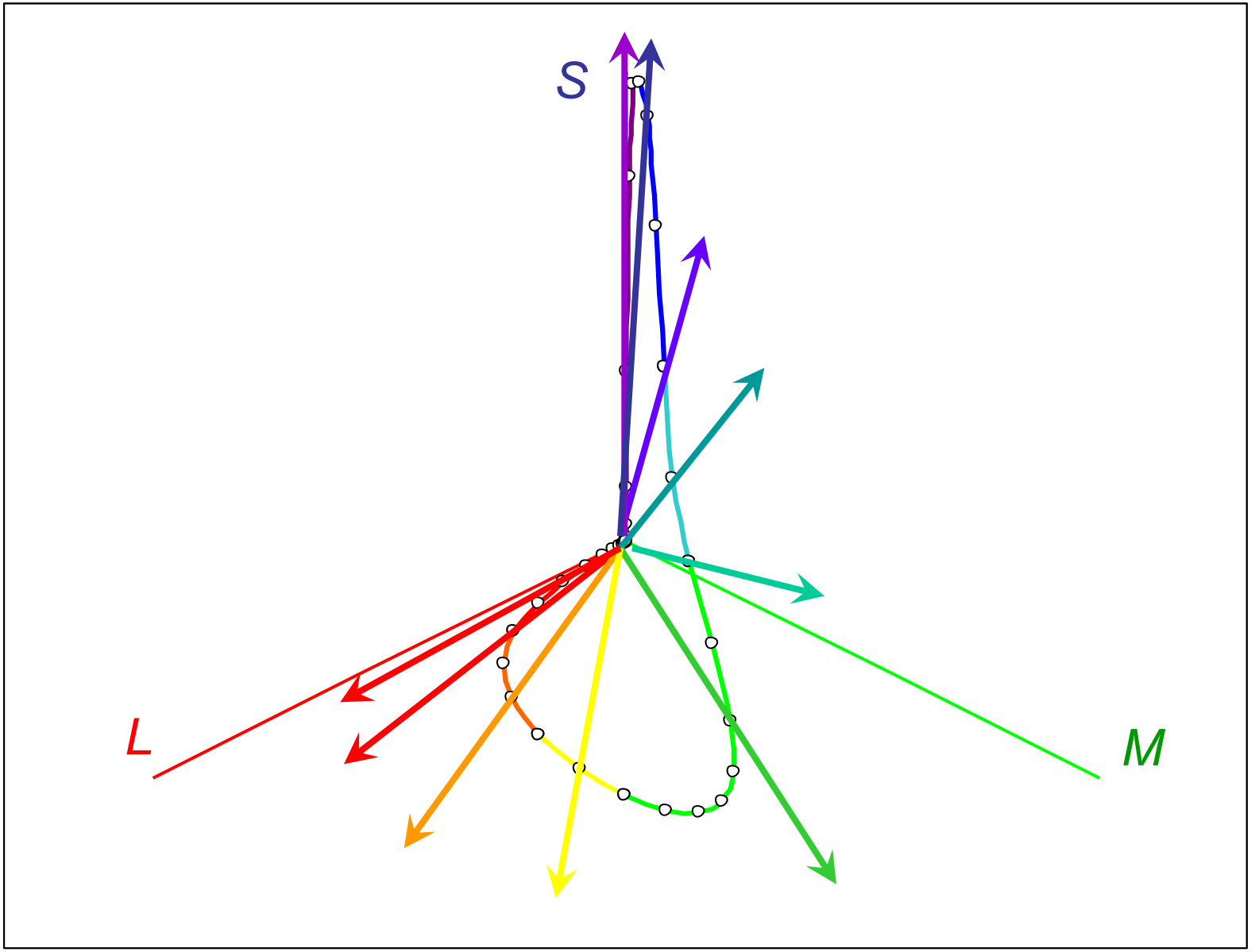
$$\int_{\lambda} \delta(\lambda-\lambda_0)\cdot\gamma(\lambda)d\lambda = \gamma(\lambda_0)$$

$$\int_{\lambda} \delta(\lambda-\lambda_0)\cdot\beta(\lambda)d\lambda = \beta(\lambda_0)$$

Локус спектральных цветов

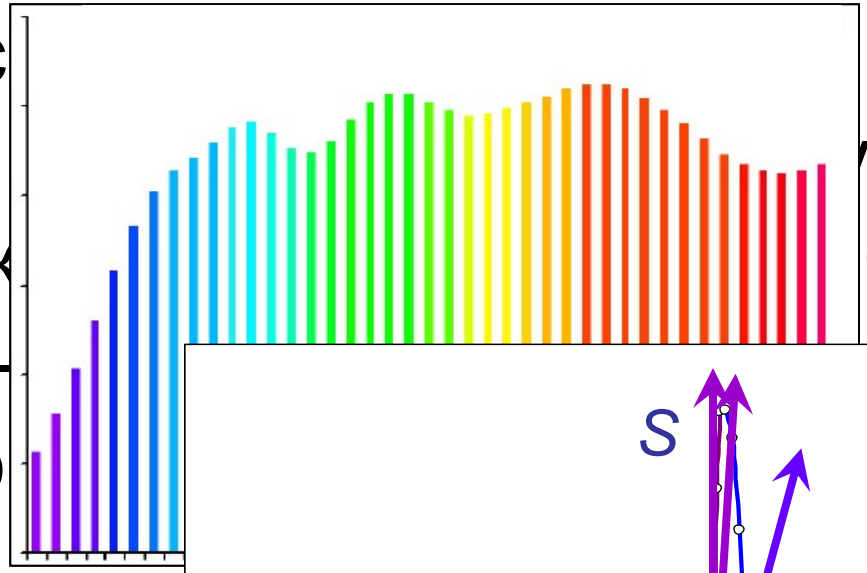


Конус спектральных цветов

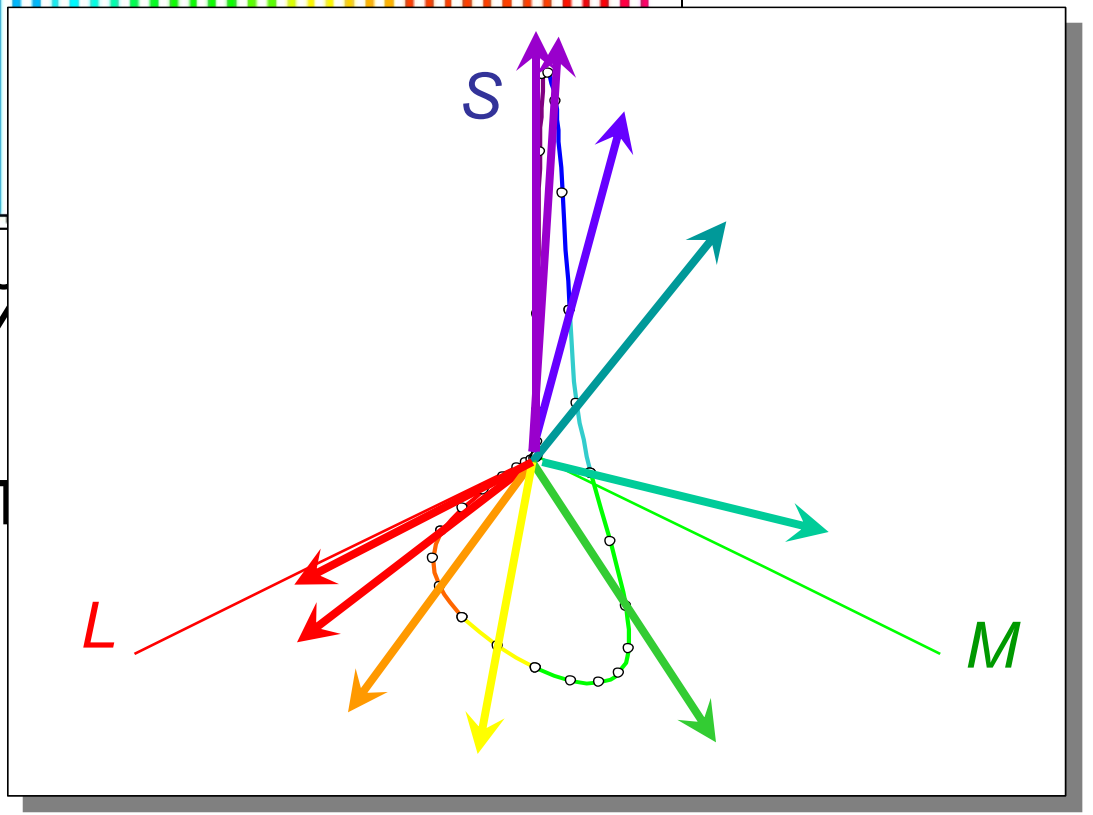


Цветовой конус

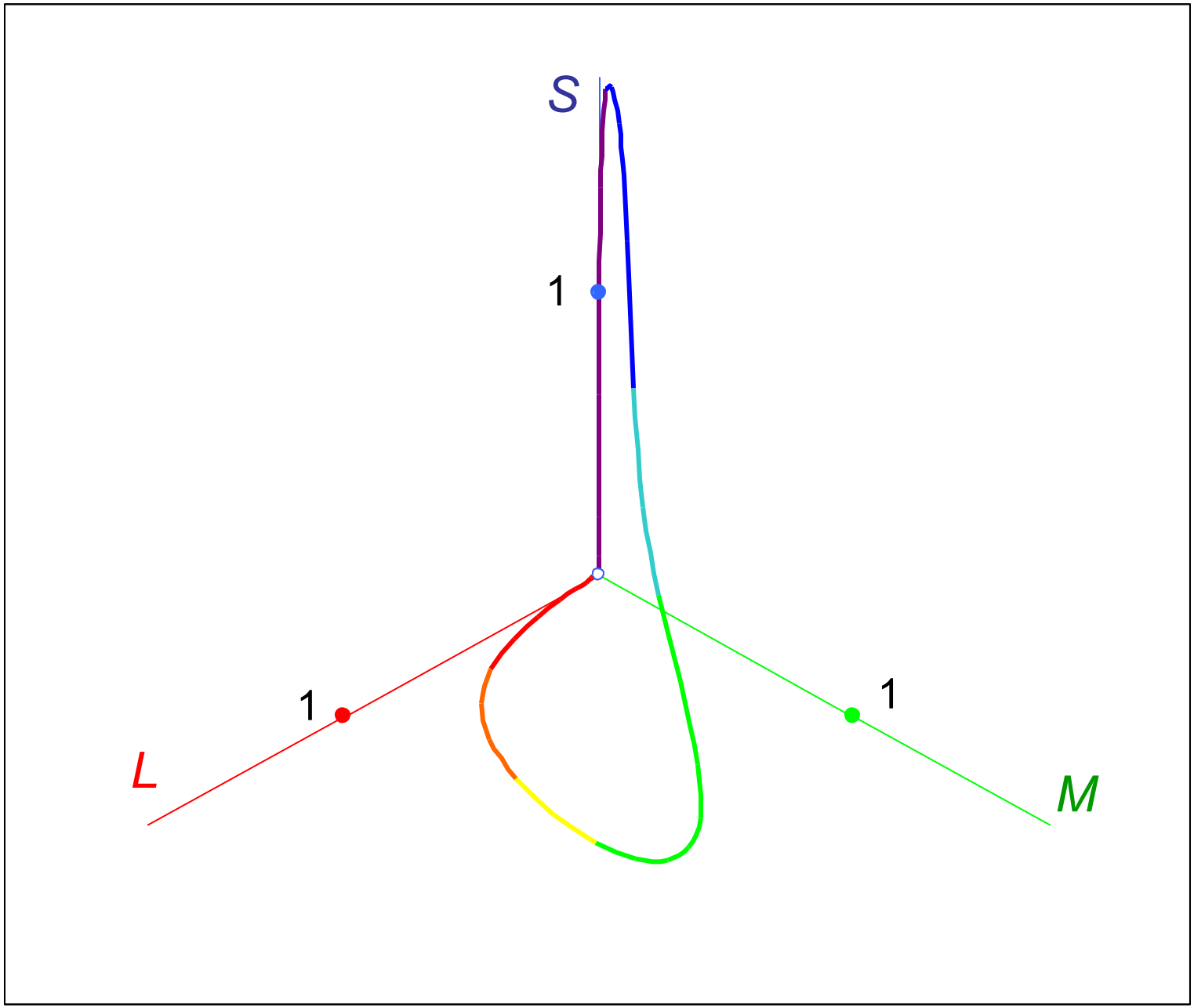
- Векторы вс
  человеком
  конуса спек
  излучение -
  весами) мо



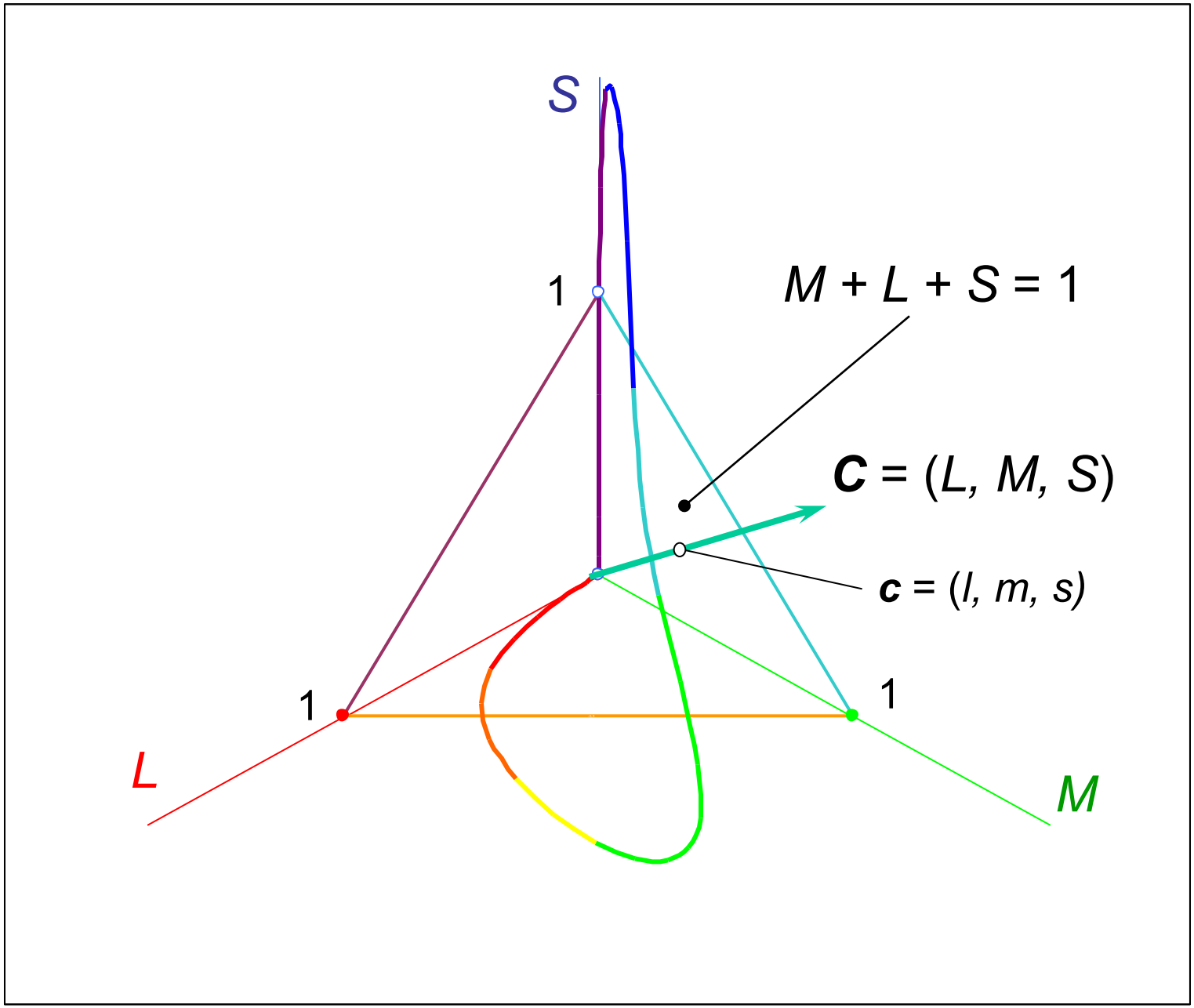
- Конус выпуклый
- Ось *M* (и *L*) не п



Треугольник цветности



Треугольник цветности



Треугольник цветности

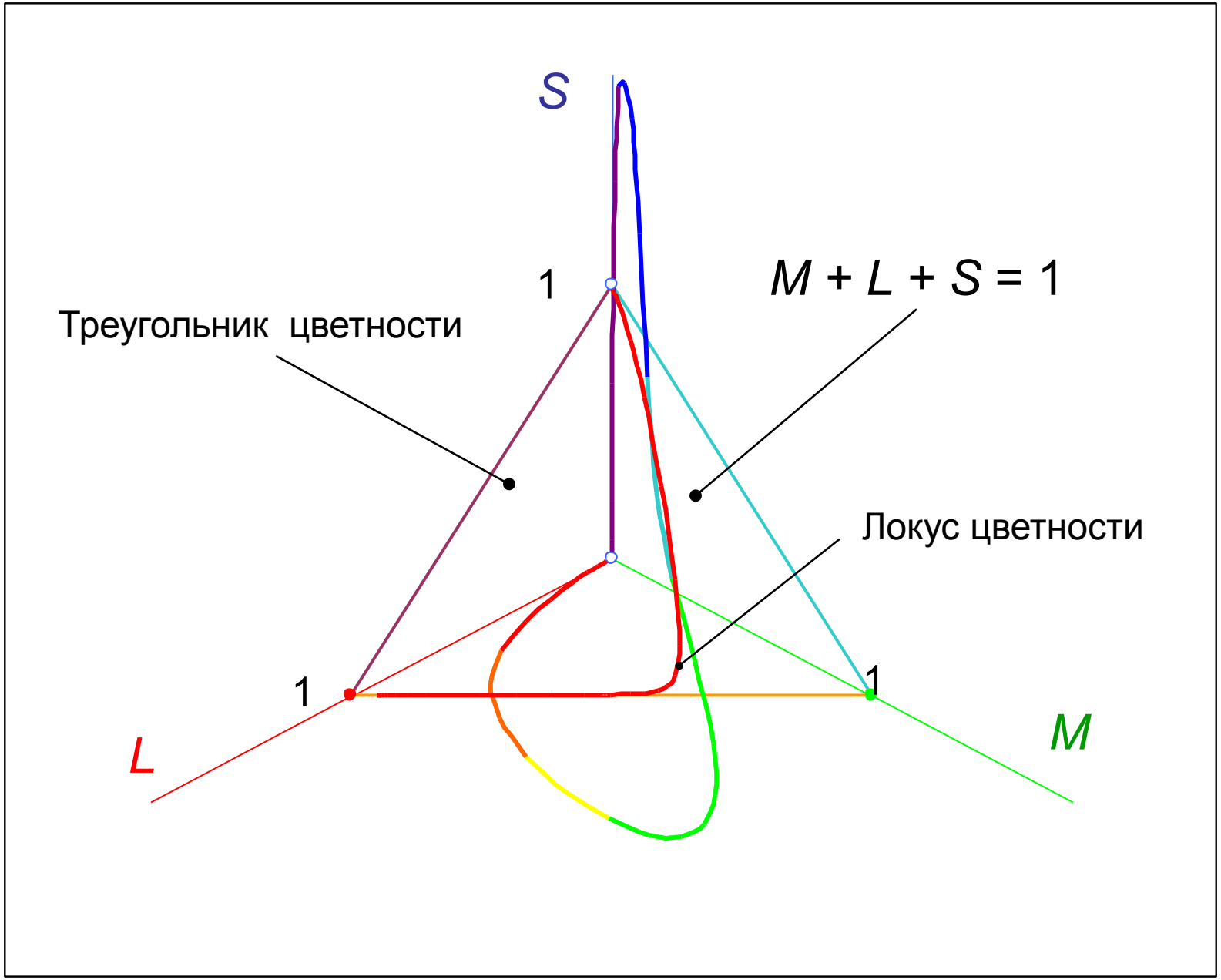
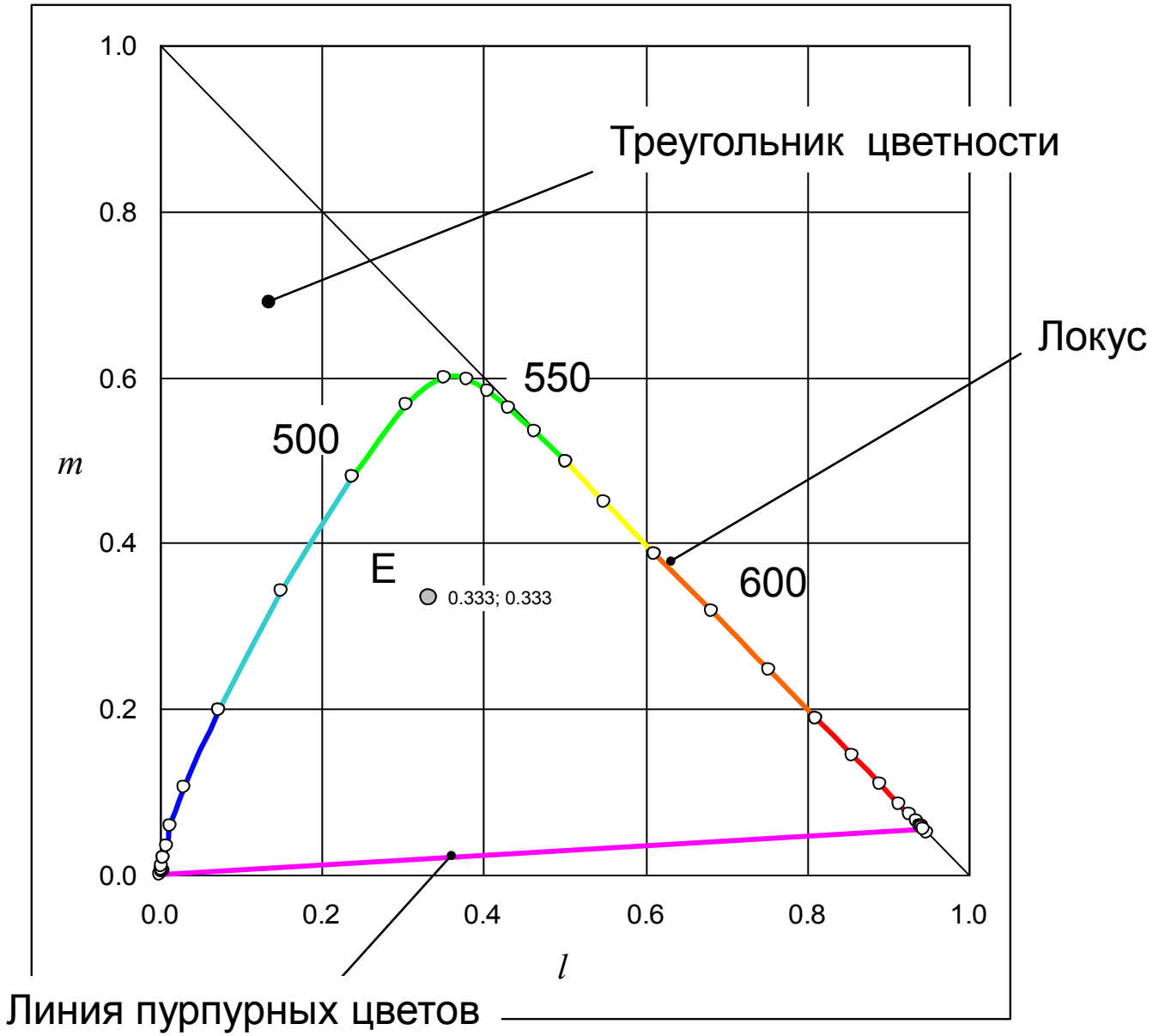
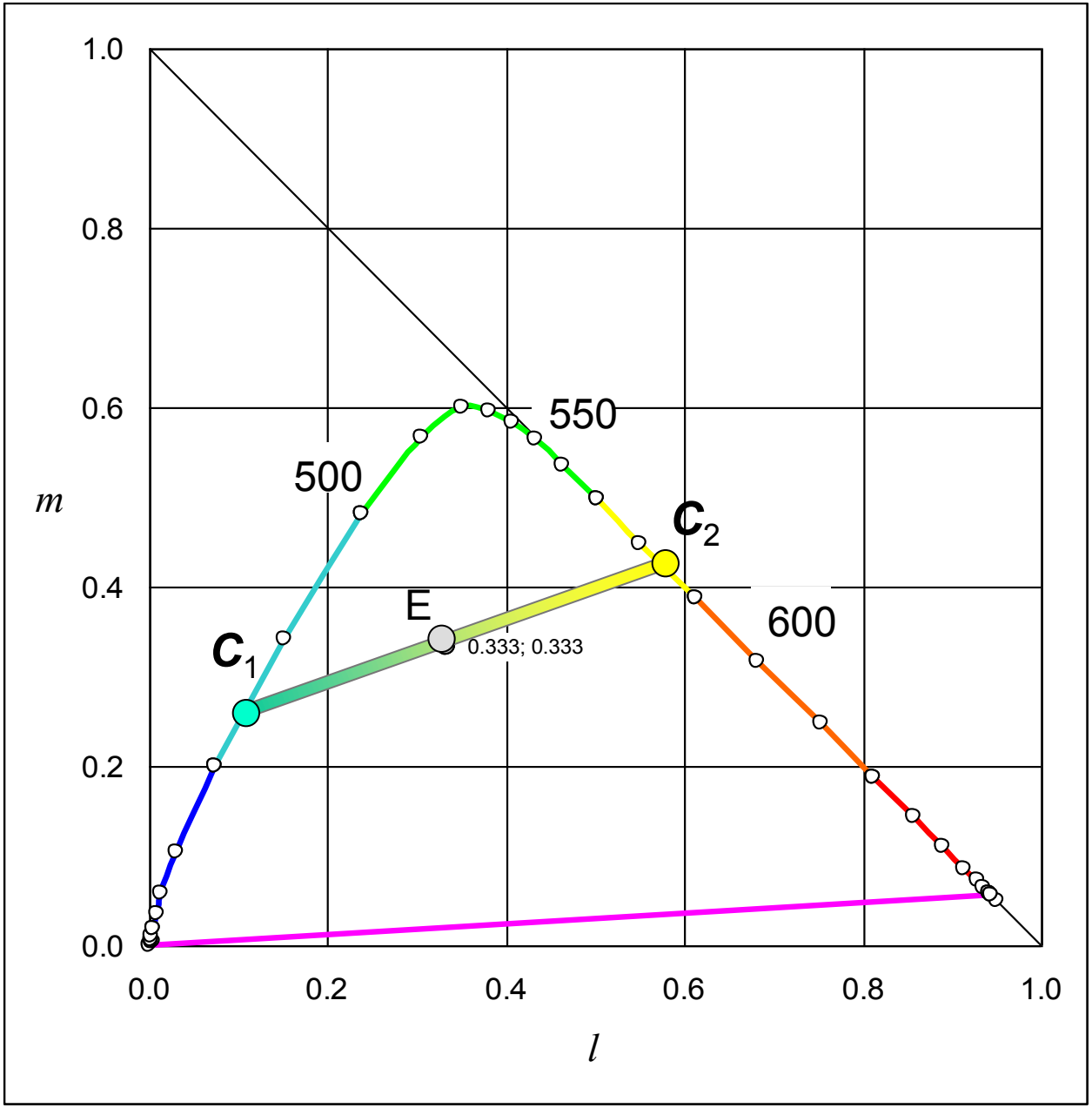


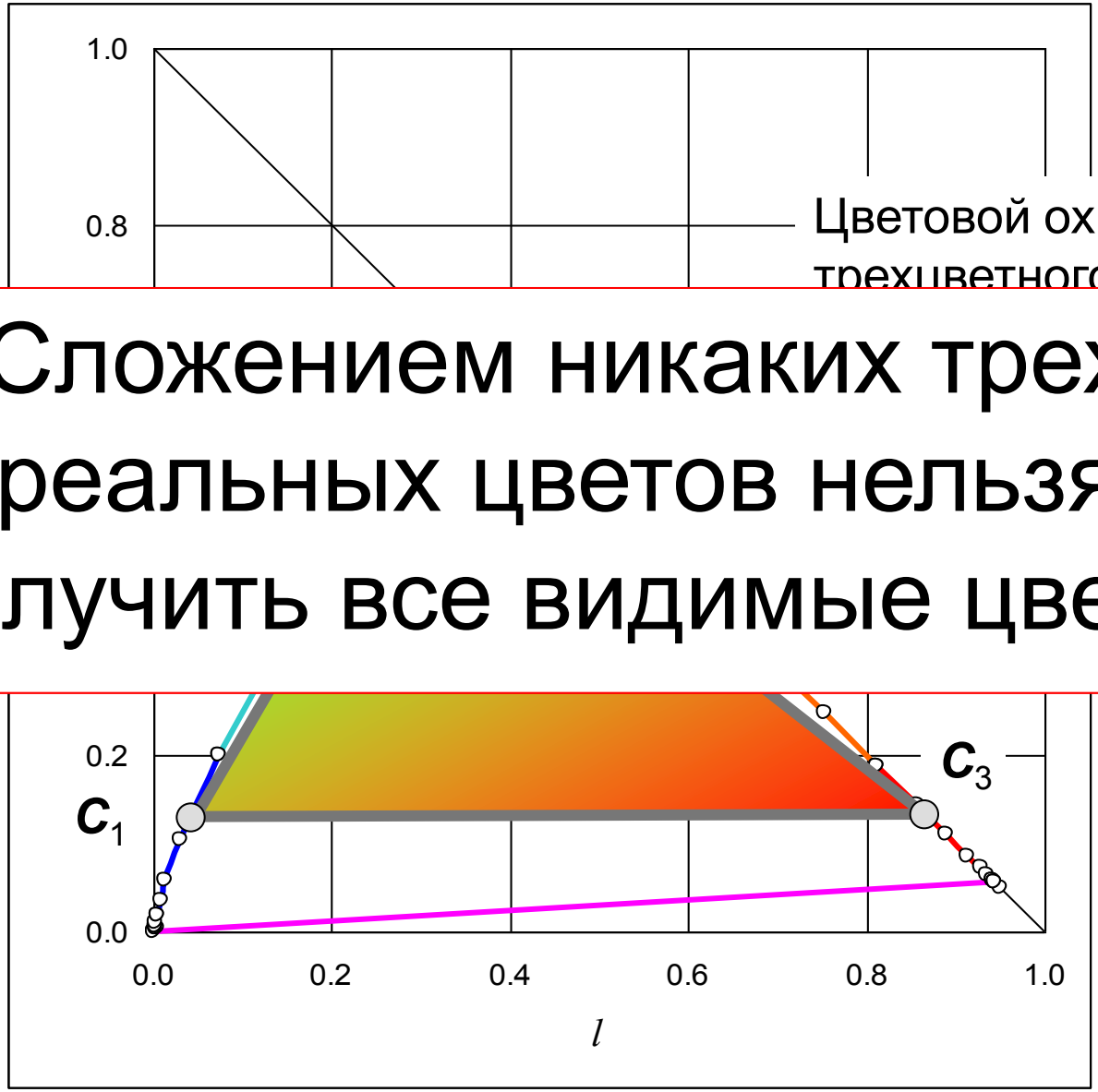
Диаграмма цветности *LMS*



Сложение цветов

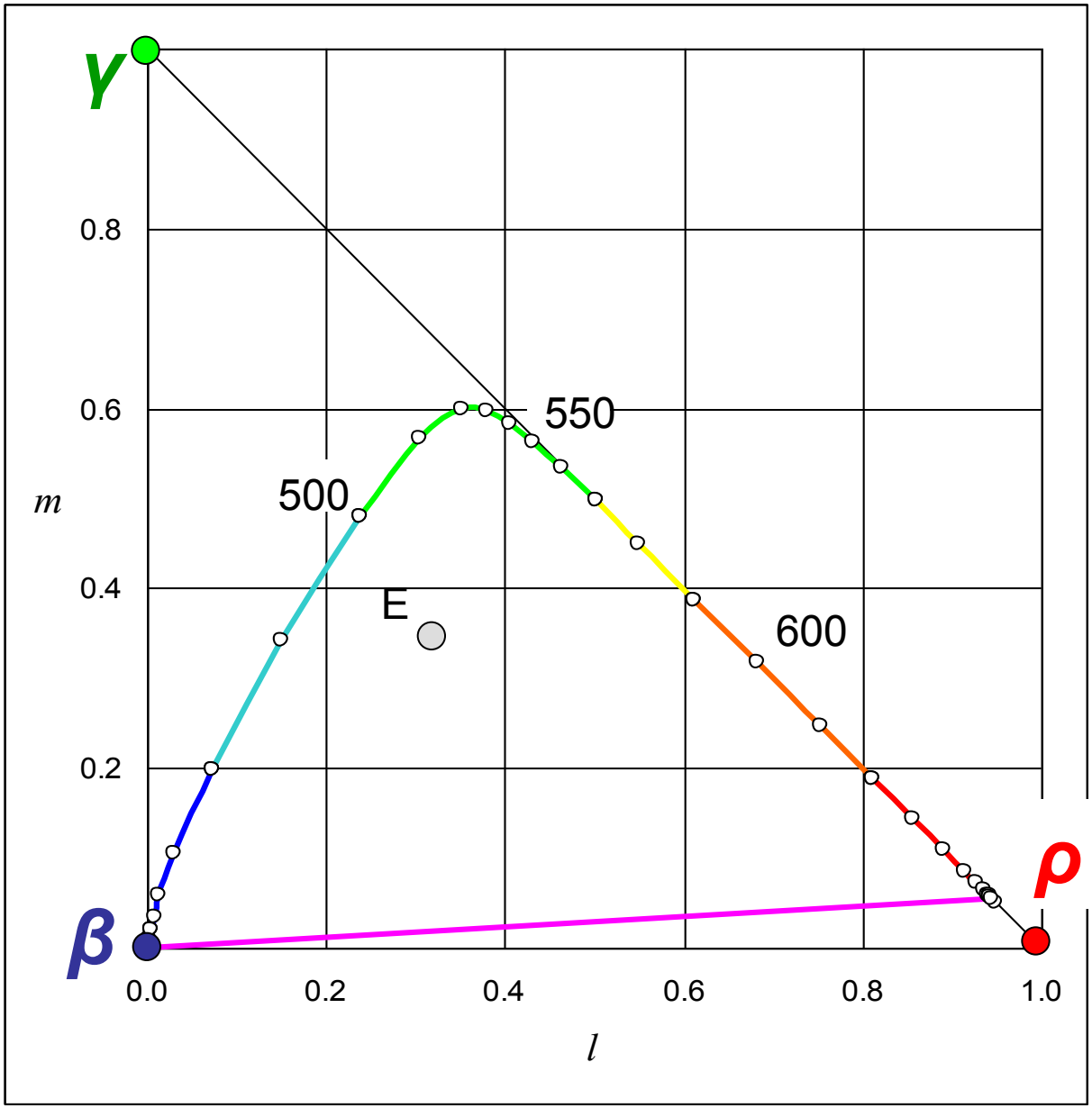


Цветовоспроизведение



Сложением никаких трех реальных цветов нельзя получить все видимые цвета

Кардинальные стимулы *LMS* на диаграмме цветности



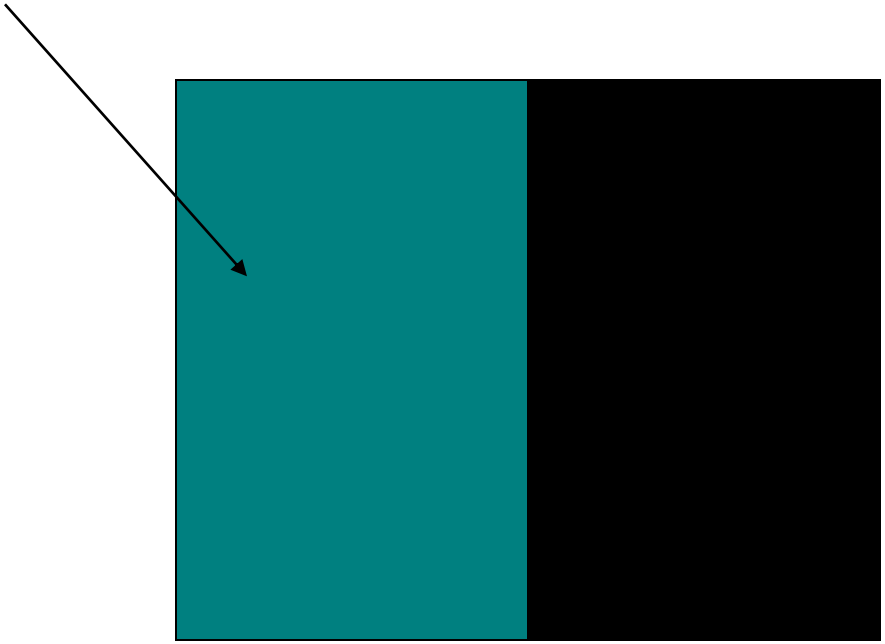
Цветовые измерения

Трехцветный колориметр

- Задача – определить координаты цвета опытным путем.
- Функции спектральной чувствительности глаза неизвестны.
- Метод – определение цветового равенства «на глаз».

Опыт по уравниванию цвета (1)

Измеряемый цвет



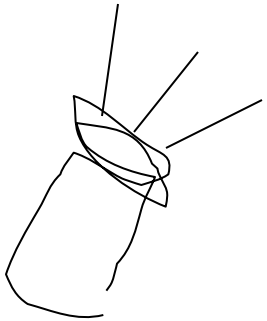
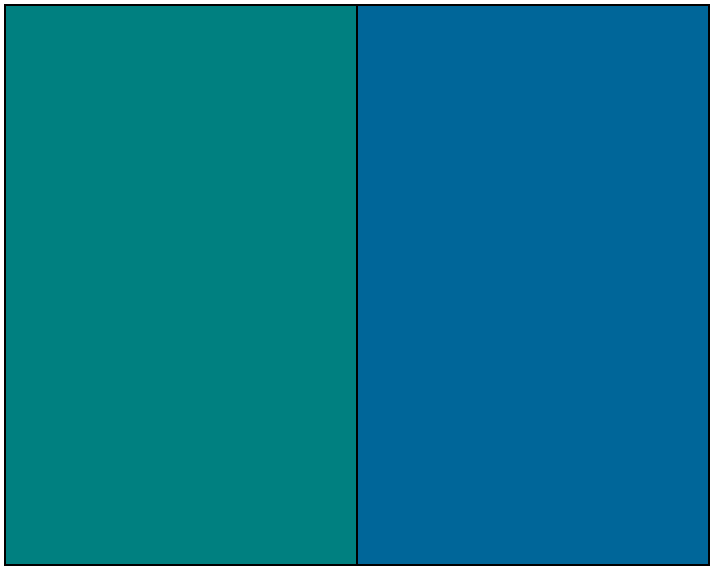
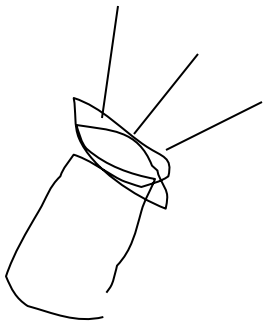
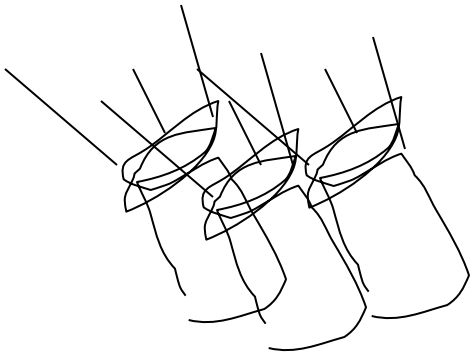


Рисунок William T. Freeman, MIT

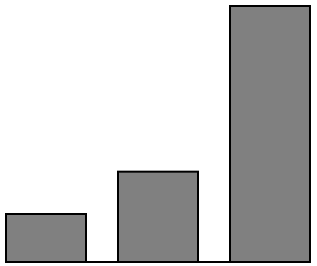
Опыт по уравниванию цвета (2)





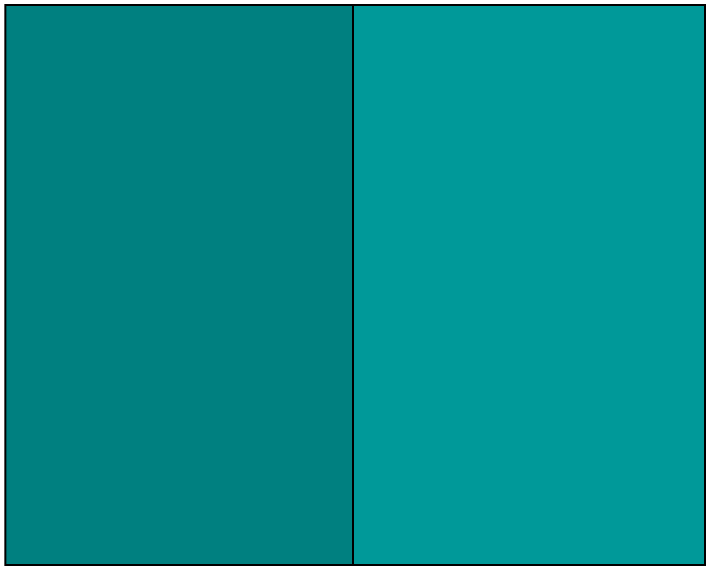


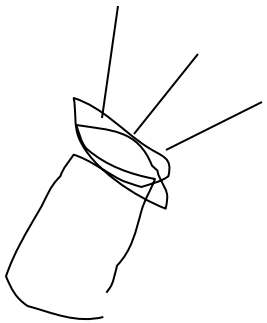
Доли цветов колориметра

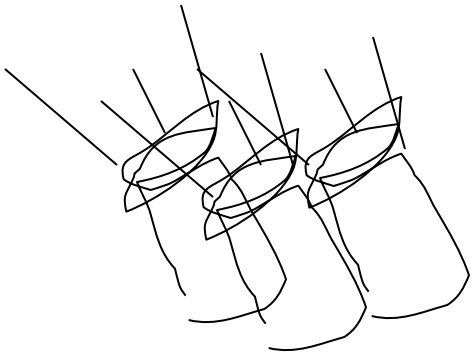


p_1 p_2 p_3

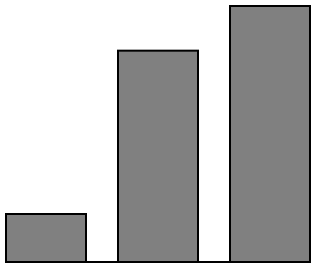
Опыт по уравниванию цвета (3)





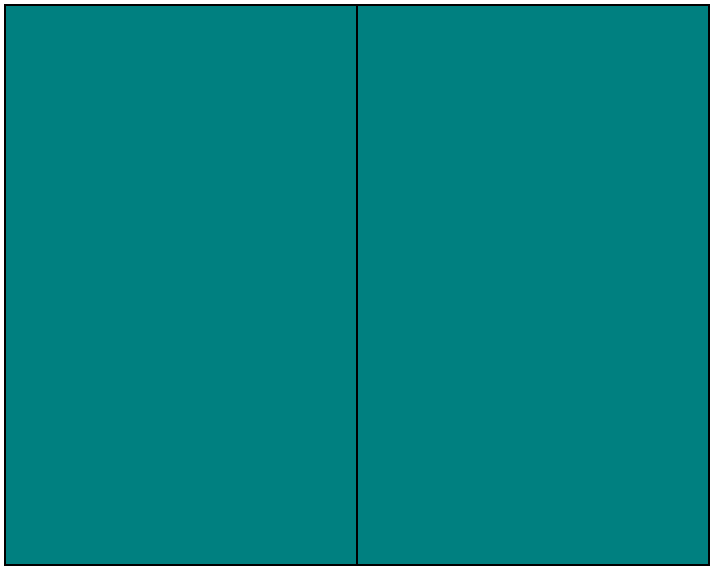


Доли цветов колориметра

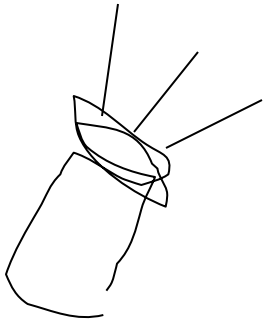


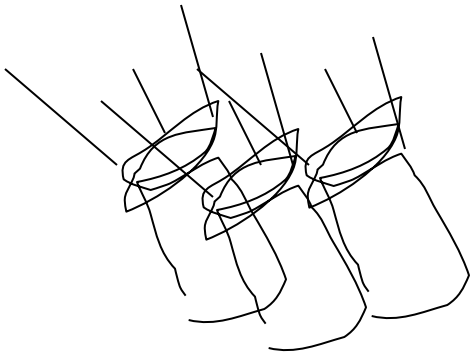
p_1 p_2 p_3

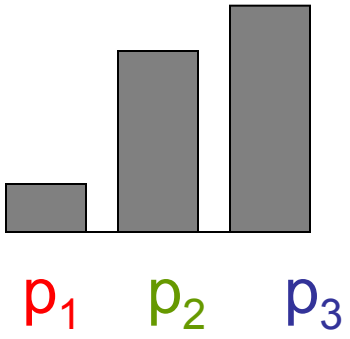
Опыт по уравниванию цвета (4)



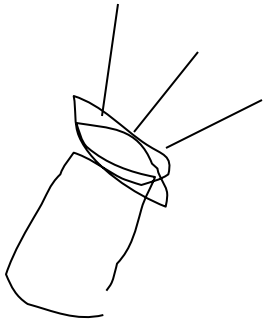
Координаты цвета



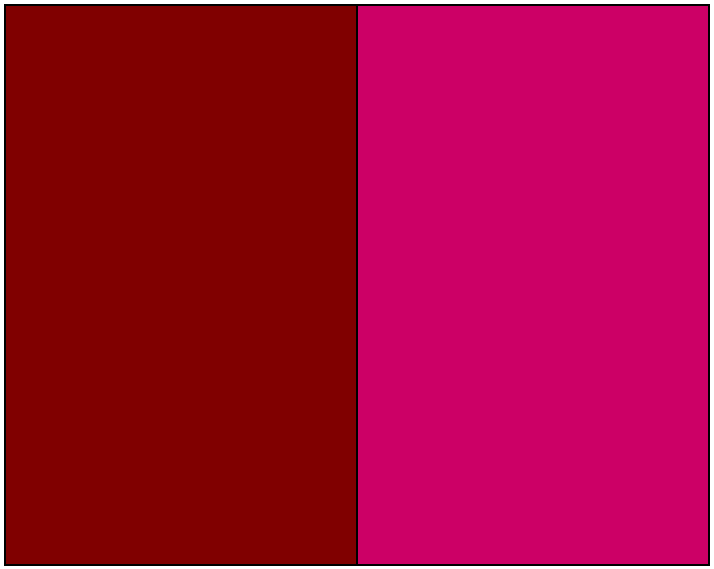


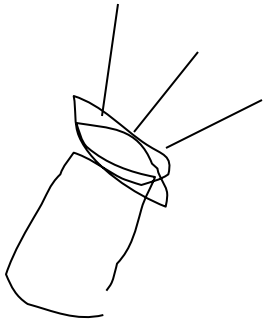


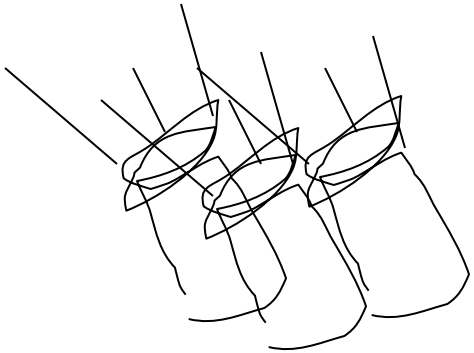
Опыт по уравниванию цвета (5)

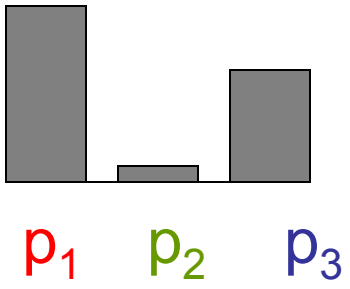


Опыт по уравниванию цвета (5)

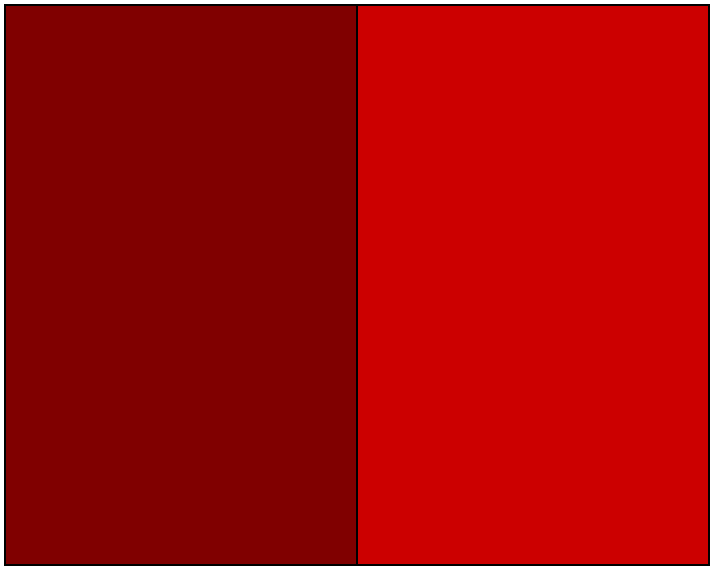


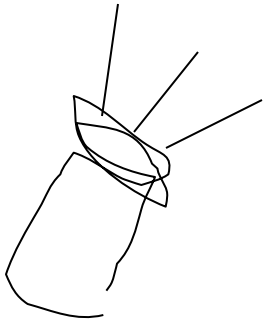


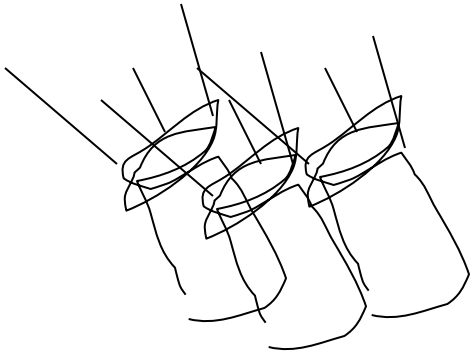


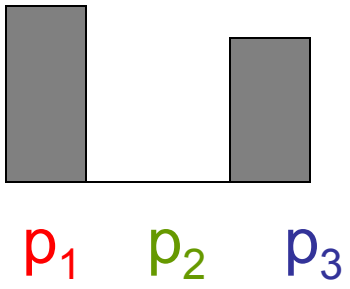


Опыт по уравниванию цвета (7)



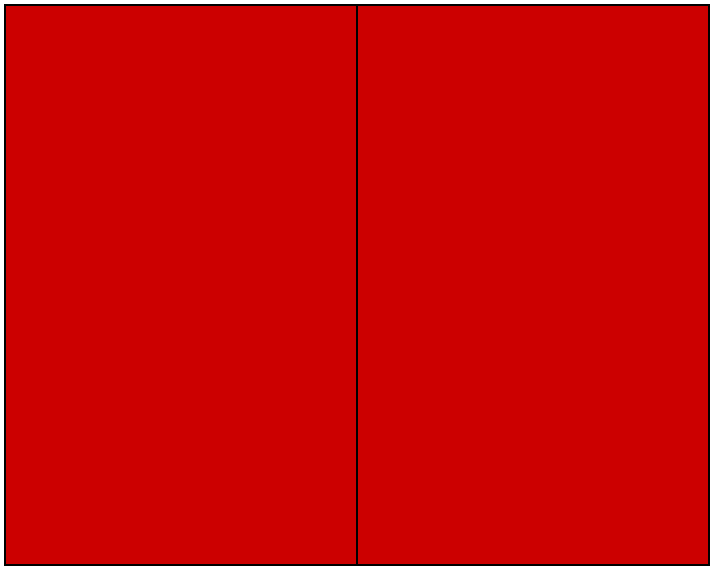




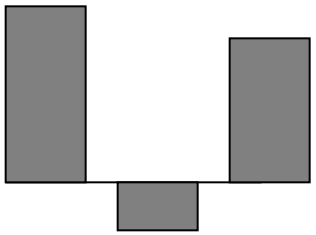


Опыт по уравниванию цвета (5)

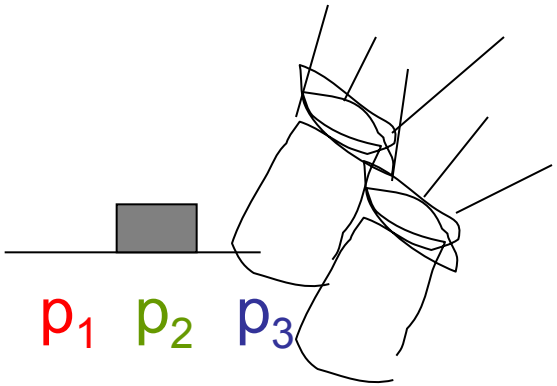
Для уравнивания потребовалось «отрицательное» количество p_2



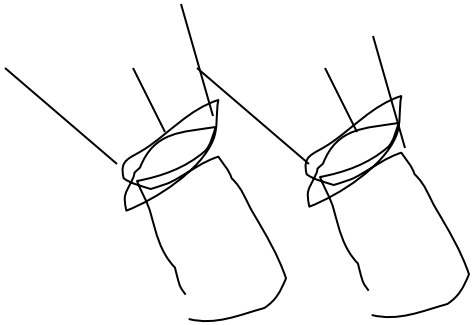
Координаты цвета

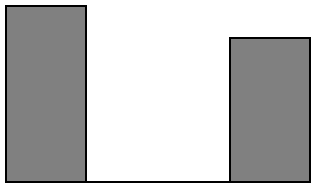


p_1 p_2 p_3



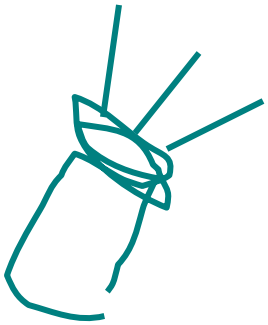
p_1 p_2 p_3

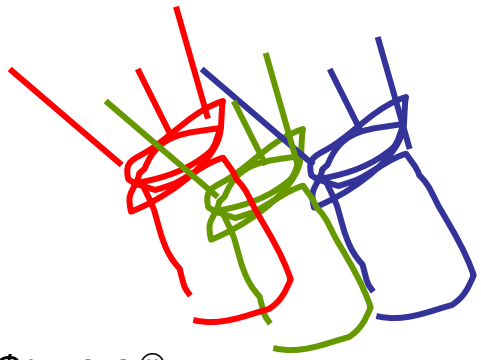




p_1 p_2 p_3

- Координаты цвета измерены в системе координат колориметра с базисом (кардинальными стимулами) $\mathbf{p}_1$, $\mathbf{p}_2$, $\mathbf{p}_3$
- Можно ли узнать координаты этого цвета в *LMS*?
  - Если спектральные чувствительности колбочек (цветовые координаты $\mathbf{p}_1$, $\mathbf{p}_2$, $\mathbf{p}_3$ в *LMS*) неизвестны – нельзя.
  - Если известны – преобразованием системы координат (вычислением).
- Можно ли найти функции спектральной чувствительности для данной системы координат?
  - Да, можно! Для этого надо определить локус в этой системе координат – найти координаты спектральных цветов.





Колориметр Б. Фримана ☺

Дополнительные главы компьютерной графики

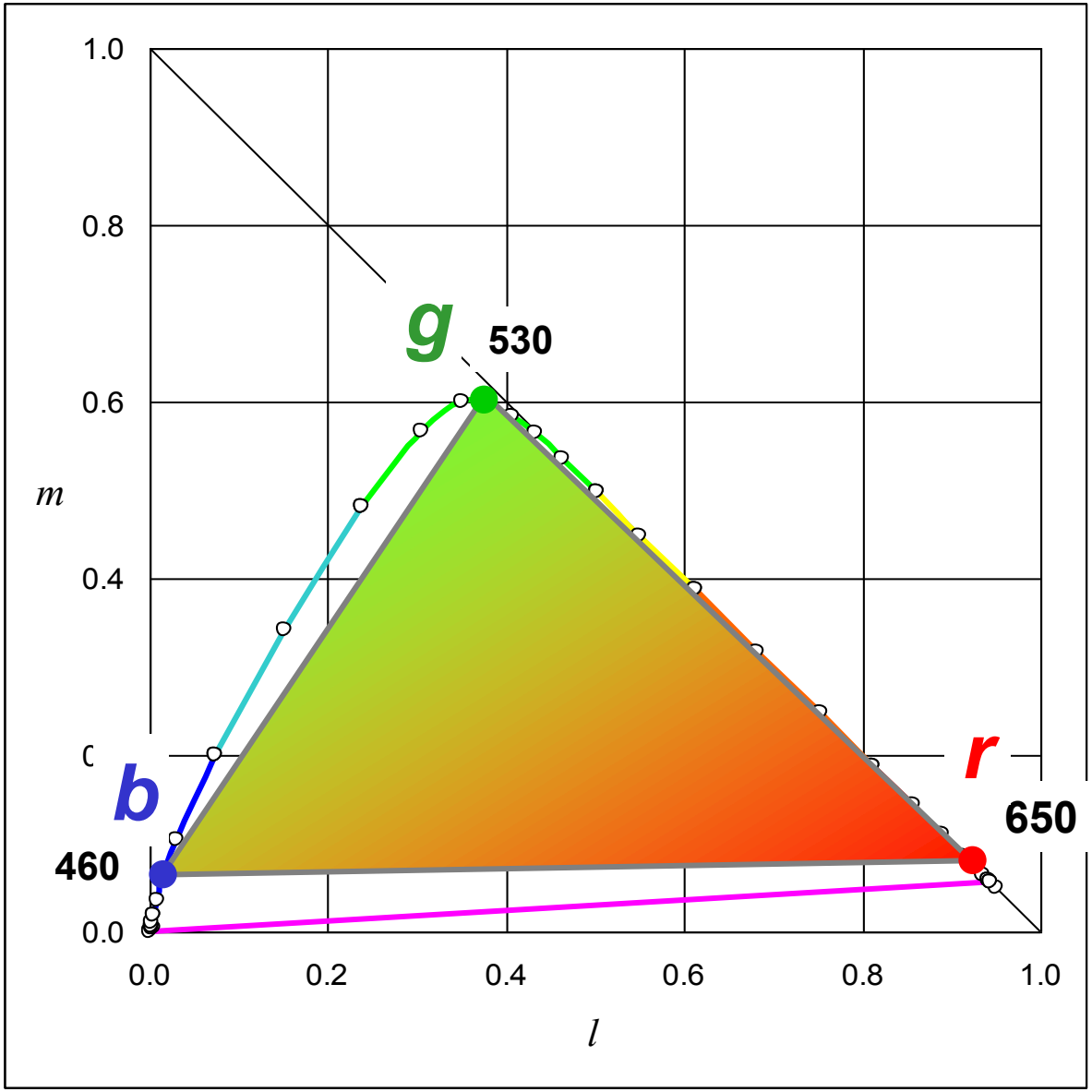
Лекция 3

Copyright © С. Свердлов, 2009-2017
http://www.uni-vologda.ac.ru/~c3c

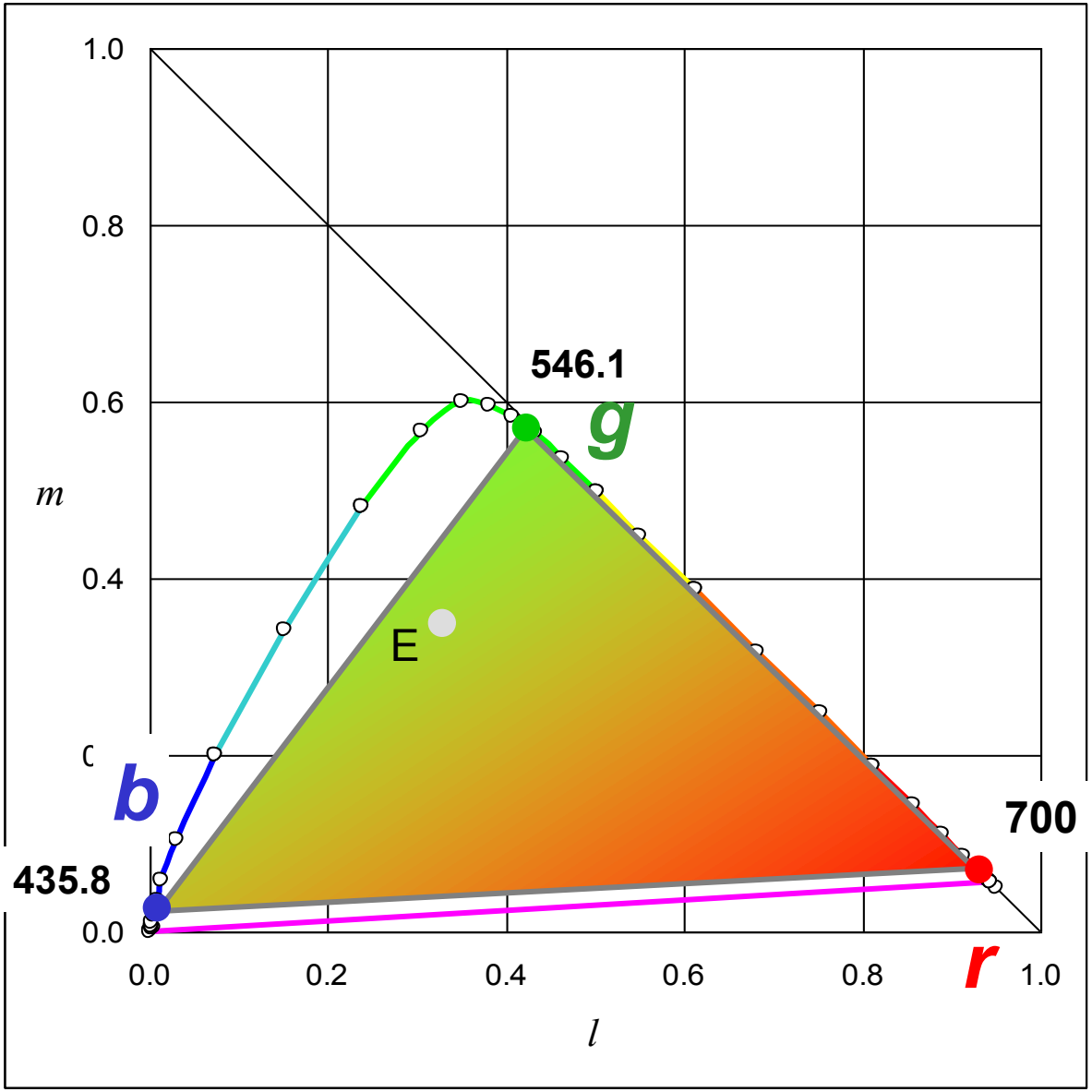
Опыты Дж. Гилда и В. Д. Райта по уравниванию цветов

- Поле зрения колориметра 2°
- Дж. Гилд (John Guild)
  - 1931 г.
  - Колориметр с широкополосными кардинальными стимулами
  - 7 человек
- В. Д. Райт (W. D. Wright)
  - 1928-1929 гг.
  - Колориметр с монохроматическими кардинальными стимулами: λ = 460нм, 530нм, 650нм.
  - 10 человек

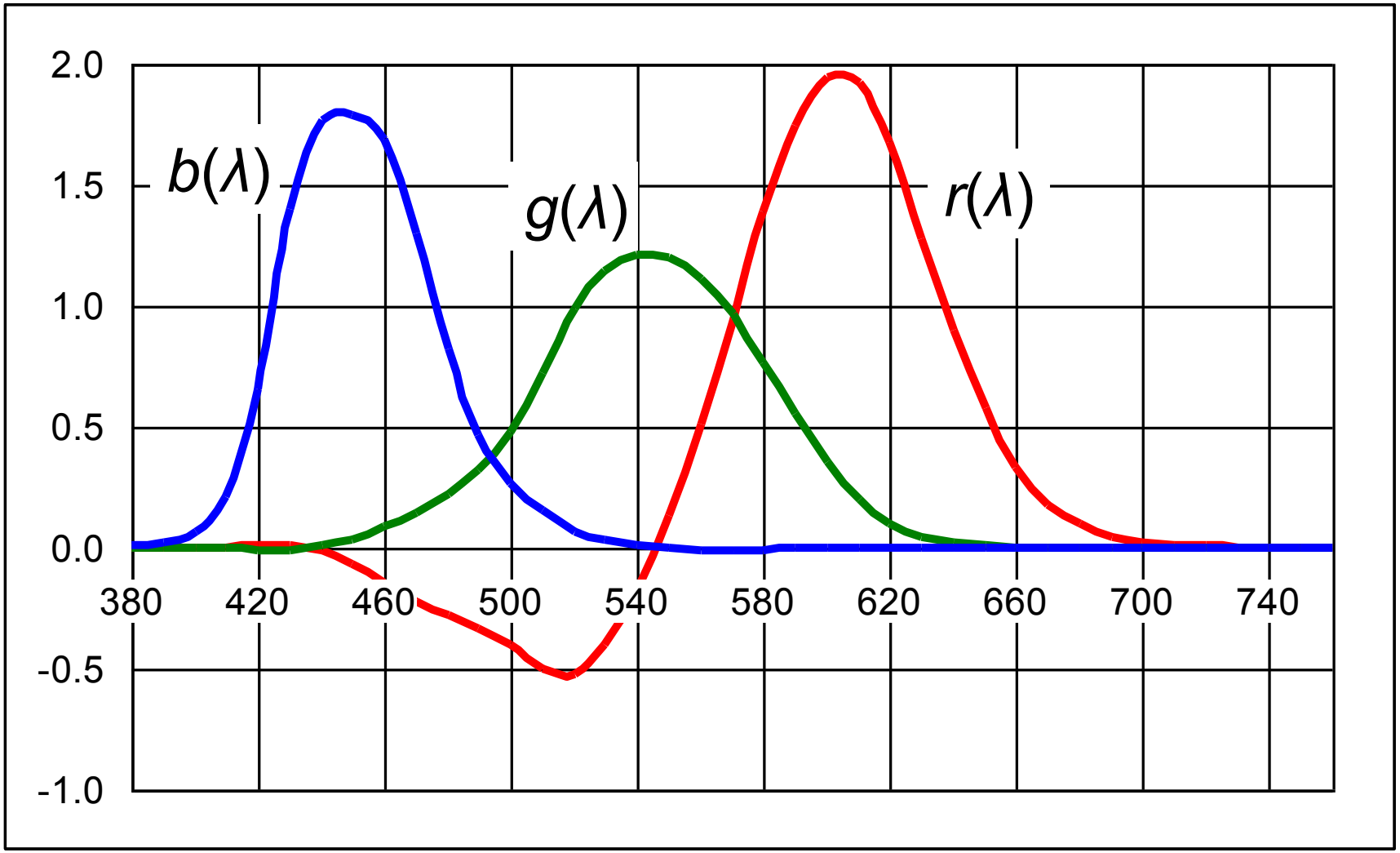
Колориметр Райта



(Виртуальный) колориметр МКО

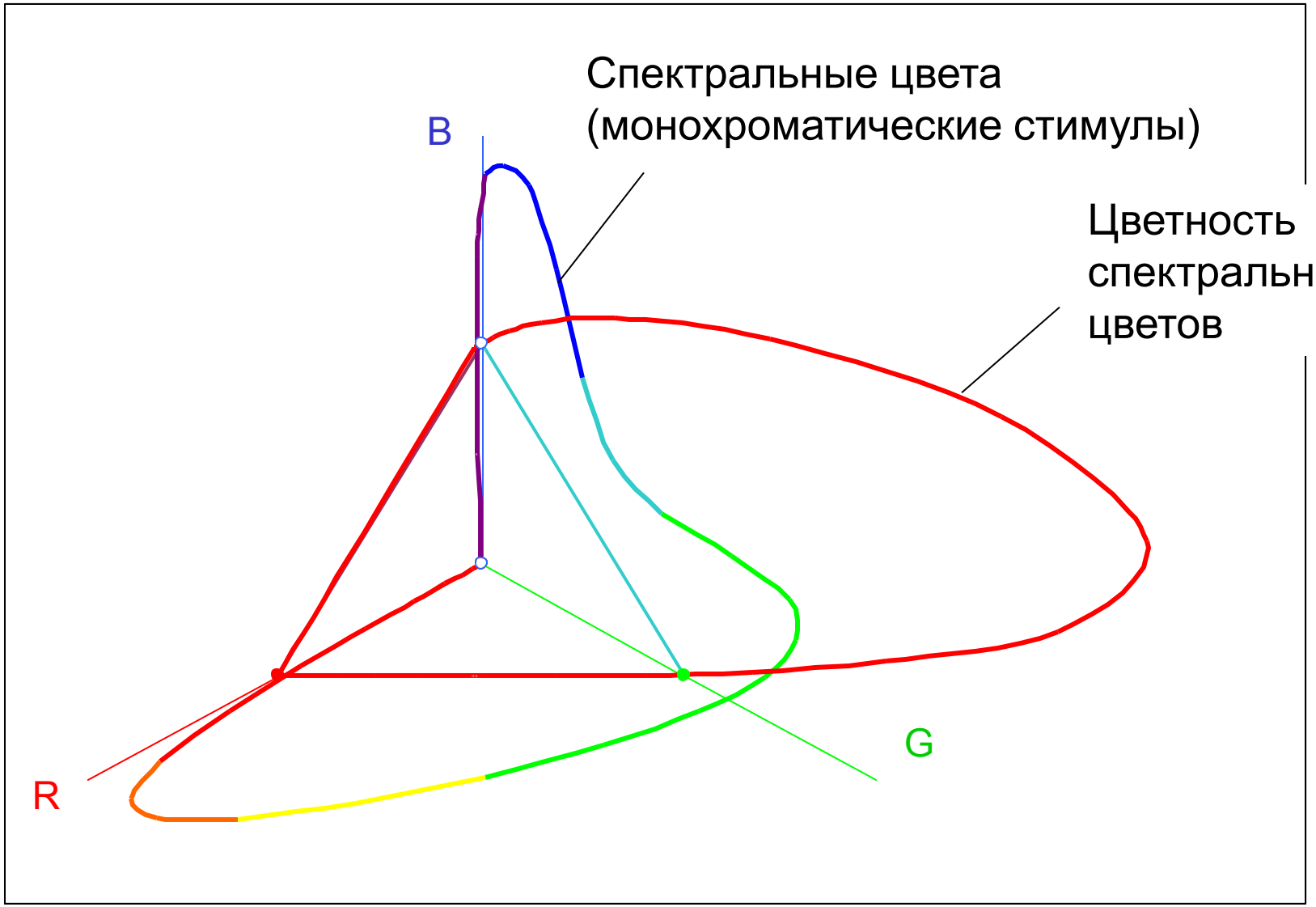


Стандартный наблюдатель МКО 1931 (RGB)



Функции спектральной чувствительности

Стандартный наблюдатель МКО 1931 (RGB)



Локус спектральных цветов и локус цветности

Стандартный наблюдатель МКО 1931 (RGB)

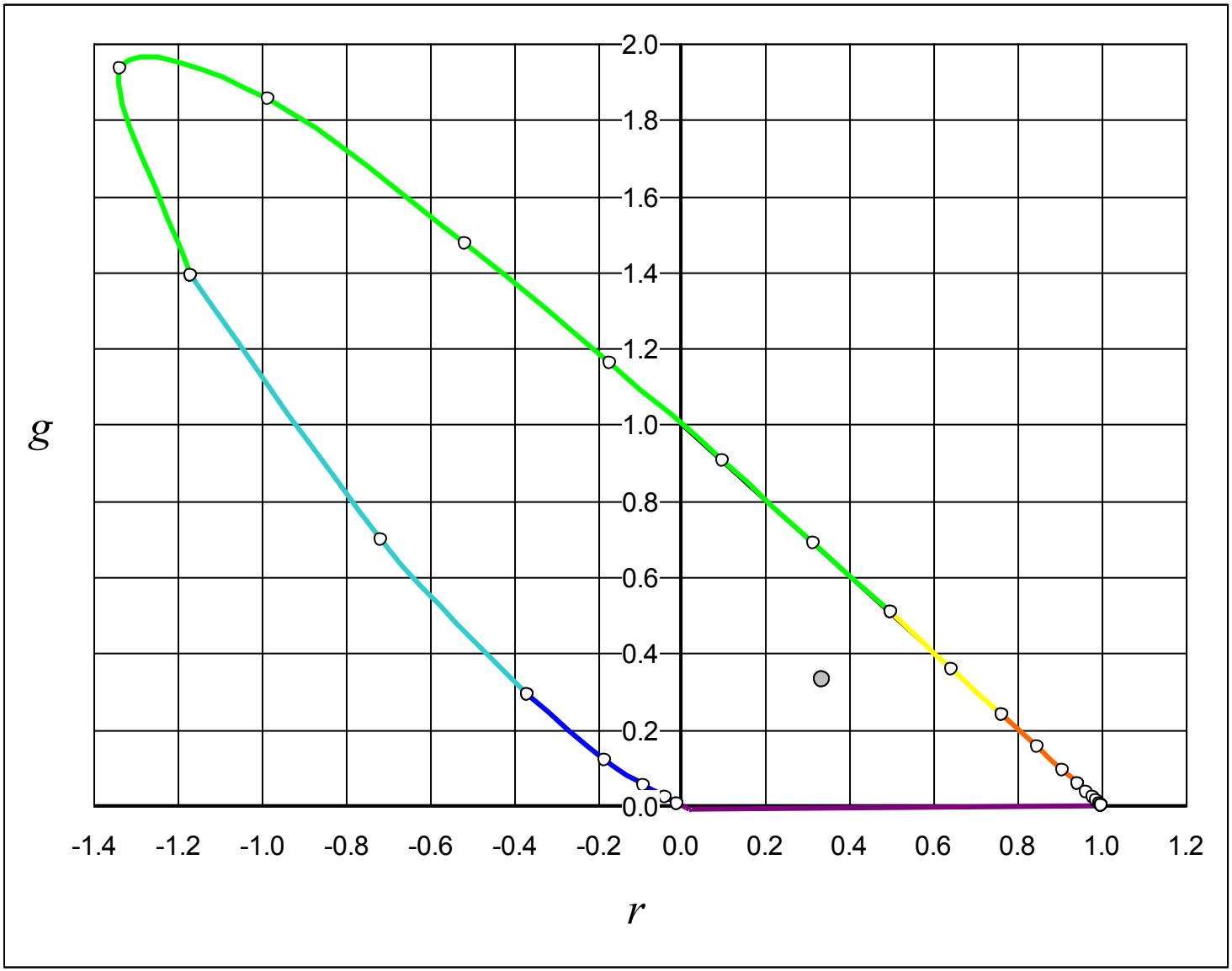


Диаграмма цветности

Преобразование цветовых координат

Преобразование цветовых координат

Дано: Цвет (стимул) **C** в цветовом пространстве *RGB*:

$$\mathbf{C} = R\mathbf{r} + G\mathbf{g} + B\mathbf{b} \qquad (1)$$

Преобразование цветовых координат

Дано: Цвет (стимул) **C** в цветовом пространстве *RGB*:

$$\mathbf{C} = R\mathbf{r} + G\mathbf{g} + B\mathbf{b} \qquad (1)$$

Этот же стимул в пространстве с кардинальными стимулами **x**, **y**, **z**:

$$\mathbf{C} = X\mathbf{x} + Y\mathbf{y} + Z\mathbf{z} \qquad (2)$$

Преобразование цветовых координат

Дано: Цвет (стимул) **C** в цветовом пространстве *RGB*:

$$\mathbf{C} = R\mathbf{r} + G\mathbf{g} + B\mathbf{b} \qquad (1)$$

Этот же стимул в пространстве с кардинальными стимулами **x**, **y**, **z**:

$$\mathbf{C} = X\mathbf{x} + Y\mathbf{y} + Z\mathbf{z} \qquad (2)$$

Координаты кардинальных стимулов **x, y, z** в *RGB*:

$$\begin{aligned} \mathbf{x} &= R_x\mathbf{r} + G_x\mathbf{g} + B_x\mathbf{b} \\ \mathbf{y} &= R_y\mathbf{r} + G_y\mathbf{g} + B_y\mathbf{b} \\ \mathbf{z} &= R_z\mathbf{r} + G_z\mathbf{g} + B_z\mathbf{b} \end{aligned} \qquad (3)$$

Преобразование цветовых координат

Дано: Цвет (стимул) **C** в цветовом пространстве *RGB*:

$$\mathbf{C} = R\mathbf{r} + G\mathbf{g} + B\mathbf{b} \qquad (1)$$

Этот же стимул в пространстве с кардинальными стимулами **x**, **y**, **z**:

$$\mathbf{C} = X\mathbf{x} + Y\mathbf{y} + Z\mathbf{z} \qquad (2)$$

Координаты кардинальных стимулов **x, y, z** в *RGB*:

$$\begin{aligned} \mathbf{x} &= R_x\mathbf{r} + G_x\mathbf{g} + B_x\mathbf{b} \\ \mathbf{y} &= R_y\mathbf{r} + G_y\mathbf{g} + B_y\mathbf{b} \\ \mathbf{z} &= R_z\mathbf{r} + G_z\mathbf{g} + B_z\mathbf{b} \end{aligned} \qquad (3)$$

Найти: *X, Y, Z*

Преобразование цветовых координат (2)

$$\mathbf{C} = X\mathbf{x} + Y\mathbf{y} + Z\mathbf{z} \tag{2}$$

$$\begin{aligned} \mathbf{x} &= R_x\mathbf{r} + G_x\mathbf{g} + B_x\mathbf{b} \\ \mathbf{y} &= R_y\mathbf{r} + G_y\mathbf{g} + B_y\mathbf{b} \\ \mathbf{z} &= R_z\mathbf{r} + G_z\mathbf{g} + B_z\mathbf{b} \end{aligned} \tag{3}$$

Преобразование цветовых координат (2)

$$\mathbf{C} = X\mathbf{x} + Y\mathbf{y} + Z\mathbf{z} \qquad (2)$$

$$\begin{aligned} \mathbf{x} &= R_x\mathbf{r} + G_x\mathbf{g} + B_x\mathbf{b} \\ \mathbf{y} &= R_y\mathbf{r} + G_y\mathbf{g} + B_y\mathbf{b} \\ \mathbf{z} &= R_z\mathbf{r} + G_z\mathbf{g} + B_z\mathbf{b} \end{aligned} \qquad (3)$$

Подставим **x, y, z** из (3) в (2):

$$\mathbf{C} = X(R_x\mathbf{r} + G_x\mathbf{g} + B_x\mathbf{b}) + Y(R_y\mathbf{r} + G_y\mathbf{g} + B_y\mathbf{b}) + Z(R_z\mathbf{r} + G_z\mathbf{g} + B_z\mathbf{b})$$

Преобразование цветовых координат (2)

$$\mathbf{C} = X\mathbf{x} + Y\mathbf{y} + Z\mathbf{z} \qquad (2)$$

$$\begin{aligned} \mathbf{x} &= R_x\mathbf{r} + G_x\mathbf{g} + B_x\mathbf{b} \\ \mathbf{y} &= R_y\mathbf{r} + G_y\mathbf{g} + B_y\mathbf{b} \\ \mathbf{z} &= R_z\mathbf{r} + G_z\mathbf{g} + B_z\mathbf{b} \end{aligned} \qquad (3)$$

Подставим **x, y, z** из (3) в (2):

$$\mathbf{C} = X(R_x\mathbf{r} + G_x\mathbf{g} + B_x\mathbf{b}) + Y(R_y\mathbf{r} + G_y\mathbf{g} + B_y\mathbf{b}) + Z(R_z\mathbf{r} + G_z\mathbf{g} + B_z\mathbf{b})$$

Раскроем скобки, сгруппируем коэффициенты при **r**, **g**, **b**:

$$\mathbf{C} = (XR_x + YR_y + ZR_z)\mathbf{r} + (XG_x + YG_y + ZG_z)\mathbf{g} + (XB_x + YB_y + ZB_z)\mathbf{b} \qquad (4)$$

Преобразование цветовых координат (2)

$$\mathbf{C} = X\mathbf{x} + Y\mathbf{y} + Z\mathbf{z} \qquad (2)$$

$$\begin{aligned} \mathbf{x} &= R_x\mathbf{r} + G_x\mathbf{g} + B_x\mathbf{b} \\ \mathbf{y} &= R_y\mathbf{r} + G_y\mathbf{g} + B_y\mathbf{b} \\ \mathbf{z} &= R_z\mathbf{r} + G_z\mathbf{g} + B_z\mathbf{b} \end{aligned} \qquad (3)$$

Подставим **x, y, z** из (3) в (2):

$$\mathbf{C} = X(R_x\mathbf{r} + G_x\mathbf{g} + B_x\mathbf{b}) + Y(R_y\mathbf{r} + G_y\mathbf{g} + B_y\mathbf{b}) + Z(R_z\mathbf{r} + G_z\mathbf{g} + B_z\mathbf{b})$$

Раскроем скобки, сгруппируем коэффициенты при **r**, **g**, **b**:

$$\mathbf{C} = (XR_x + YR_y + ZR_z)\mathbf{r} + (XG_x + YG_y + ZG_z)\mathbf{g} + (XB_x + YB_y + ZB_z)\mathbf{b} \qquad (4)$$

Поскольку по (1) также:

$$\mathbf{C} = R\mathbf{r} + G\mathbf{g} + B\mathbf{b} \qquad (1)$$

приравняем коэффициенты при при **r**, **g**, **b** в (1) и (4)

Преобразование цветовых координат (3)

$$\mathbf{C} = (XR_x + YR_y + ZR_z)\mathbf{r} + (XG_x + YG_y + ZG_z)\mathbf{g} + (XB_x + YB_y + ZB_z)\mathbf{b} \quad (4)$$

$$\mathbf{C} = R\mathbf{r} + G\mathbf{g} + B\mathbf{b} \quad (1)$$

приравняем коэффициенты при **r**, **g**, **b** в (1) и (4):

$$R = XR_x + YR_y + ZR_z$$

$$G = XG_x + YG_y + ZG_z$$

$$B = XB_x + YB_y + ZB_z$$

Преобразование цветовых координат (3)

$$\mathbf{C} = (XR_x + YR_y + ZR_z)\mathbf{r} + (XG_x + YG_y + ZG_z)\mathbf{g} + (XB_x + YB_y + ZB_z)\mathbf{b} \quad (4)$$

$$\mathbf{C} = R\mathbf{r} + G\mathbf{g} + B\mathbf{b} \quad (1)$$

приравняем коэффициенты при **r**, **g**, **b** в (1) и (4):

$$R = XR_x + YR_y + ZR_z$$

$$G = XG_x + YG_y + ZG_z$$

$$B = XB_x + YB_y + ZB_z$$

В матричной форме:

$$\begin{pmatrix} R \\ G \\ B \end{pmatrix} = \begin{pmatrix} R_x + R_y + R_z \\ G_x + G_y + G_z \\ B_x + B_y + B_z \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}$$

Преобразование цветовых координат (4)

$$\begin{pmatrix} R \\ G \\ B \end{pmatrix} = \begin{pmatrix} R_x & R_y & R_z \\ G_x & G_y & G_z \\ B_x & B_y & B_z \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}$$

или:

$$\begin{pmatrix} R \\ G \\ B \end{pmatrix} = \mathrm{T} \cdot \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}, \text{ где } \mathrm{T} = \begin{pmatrix} R_x & R_y & R_z \\ G_x & G_y & G_z \\ B_x & B_y & B_z \end{pmatrix}$$

Преобразование цветовых координат (4)

$$\begin{pmatrix} R \\ G \\ B \end{pmatrix} = \begin{pmatrix} R_x & R_y & R_z \\ G_x & G_y & G_z \\ B_x & B_y & B_z \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}$$

или:

$$\begin{pmatrix} R \\ G \\ B \end{pmatrix} = \mathrm{T} \cdot \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}, \text{ где } \mathrm{T} = \begin{pmatrix} R_x & R_y & R_z \\ G_x & G_y & G_z \\ B_x & B_y & B_z \end{pmatrix}$$

или:

$$\mathbf{C}_{RGB} = \mathrm{T} \cdot \mathbf{C}_{XYZ}, \quad \text{где } \mathbf{C}_{RGB} = \begin{pmatrix} R \\ G \\ B \end{pmatrix}; \ \mathbf{C}_{XYZ} = \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}$$

Преобразование цветовых координат (4)

$$\begin{pmatrix} R \\ G \\ B \end{pmatrix} = \begin{pmatrix} R_x & R_y & R_z \\ G_x & G_y & G_z \\ B_x & B_y & B_z \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}$$

или:

$$\begin{pmatrix} R \\ G \\ B \end{pmatrix} = \mathrm{T} \cdot \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}, \text{ где } \mathrm{T} = \begin{pmatrix} R_x & R_y & R_z \\ G_x & G_y & G_z \\ B_x & B_y & B_z \end{pmatrix}$$

или:

$$\mathbf{C}_{RGB} = \mathrm{T} \cdot \mathbf{C}_{XYZ}, \quad \text{где } \mathbf{C}_{RGB} = \begin{pmatrix} R \\ G \\ B \end{pmatrix}; \; \mathbf{C}_{XYZ} = \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}$$

тогда:

$$\mathbf{C}_{XYZ} = \mathrm{T}^{-1} \cdot \mathbf{C}_{RGB}$$

Преобразование цветовых координат (5)

$$\mathbf{x} = R_x\mathbf{r} + G_x\mathbf{g} + B_x\mathbf{b}$$
$$\mathbf{y} = R_y\mathbf{r} + G_y\mathbf{g} + B_y\mathbf{b}$$
$$\mathbf{z} = R_z\mathbf{r} + G_z\mathbf{g} + B_z\mathbf{b}$$

$$\begin{pmatrix}\mathbf{x}\\ \mathbf{y}\\ \mathbf{z}\end{pmatrix} = \begin{pmatrix}R_x & G_x & B_x\\ R_y & G_y & B_y\\ R_z & G_z & B_z\end{pmatrix}\begin{pmatrix}\mathbf{r}\\ \mathbf{g}\\ \mathbf{b}\end{pmatrix}$$

$$\mathrm{T} = \begin{pmatrix}R_x & R_y & R_z\\ G_x & G_y & G_z\\ B_x & B_y & B_z\end{pmatrix}$$

$$\begin{pmatrix}\mathbf{x}\\ \mathbf{y}\\ \mathbf{z}\end{pmatrix} = \mathrm{T}^*\begin{pmatrix}\mathbf{r}\\ \mathbf{g}\\ \mathbf{b}\end{pmatrix}$$

$$\mathbf{C}_{RGB} = \mathrm{T}\cdot\mathbf{C}_{XYZ}$$

$$\mathbf{C}_{XYZ} = \mathrm{T}^{-1}\cdot\mathbf{C}_{RGB}$$

Преобразование цветовых координат (5)

$$\mathbf{x} = R_x\mathbf{r} + G_x\mathbf{g} + B_x\mathbf{b}$$
$$\mathbf{y} = R_y\mathbf{r} + G_y\mathbf{g} + B_y\mathbf{b}$$
$$\mathbf{z} = R_z\mathbf{r} + G_z\mathbf{g} + B_z\mathbf{b}$$

$$\begin{pmatrix}\mathbf{x}\\ \mathbf{y}\\ \mathbf{z}\end{pmatrix} = \begin{pmatrix}R_x & G_x & B_x\\ R_y & G_y & B_y\\ R_z & G_z & B_z\end{pmatrix}\begin{pmatrix}\mathbf{r}\\ \mathbf{g}\\ \mathbf{b}\end{pmatrix}$$

$$\mathrm{T} = \begin{pmatrix}R_x & R_y & R_z\\ G_x & G_y & G_z\\ B_x & B_y & B_z\end{pmatrix}$$

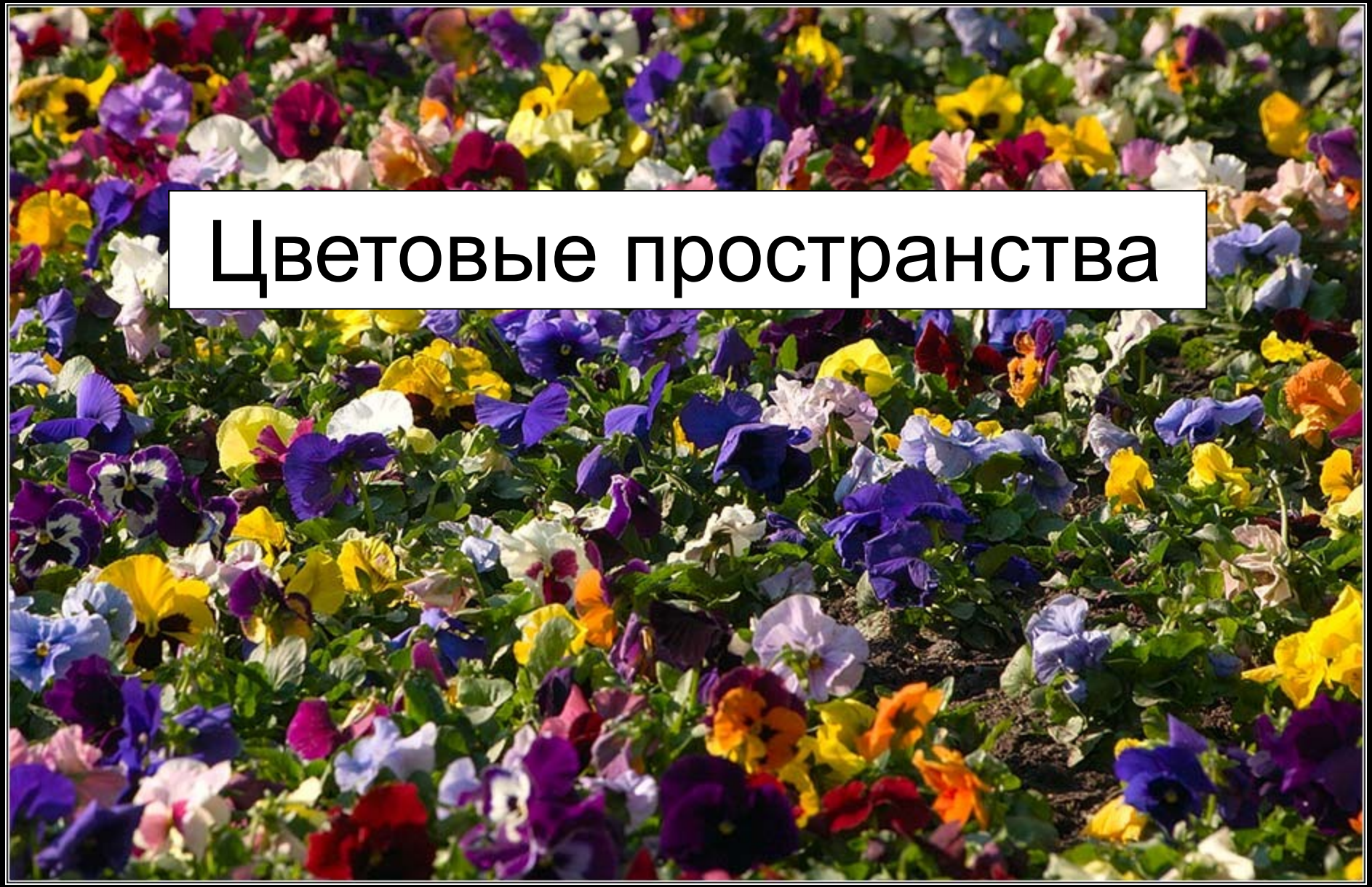
$$\begin{pmatrix}\mathbf{x}\\ \mathbf{y}\\ \mathbf{z}\end{pmatrix} = \mathrm{T}^*\begin{pmatrix}\mathbf{r}\\ \mathbf{g}\\ \mathbf{b}\end{pmatrix}$$

$$\mathbf{C}_{RGB} = \mathrm{T}\cdot\mathbf{C}_{XYZ}$$

$$\mathbf{C}_{XYZ} = \mathrm{T}^{-1}\cdot\mathbf{C}_{RGB}$$

В силу симметрии:

$$\mathrm{T}^{-1} = \begin{pmatrix}X_r & X_g & X_b\\ Y_r & Y_g & Y_b\\ Z_r & Z_g & Z_b\end{pmatrix}$$



Цветовое пространство

Цветовое пространство МКО XYZ (CIE XYZ)

- Пространство CIE XYZ представляет собой линейное преобразование пространства CIE RGB (и LMS).
- Функция спектральной чувствительности $y(\lambda)$ совпадает с кривой видности $V(\lambda)$

$$y(\lambda) = V(\lambda) = 1.000\ r(\lambda) + 4.591\ g(\lambda) + 0.060\ b(\lambda)$$

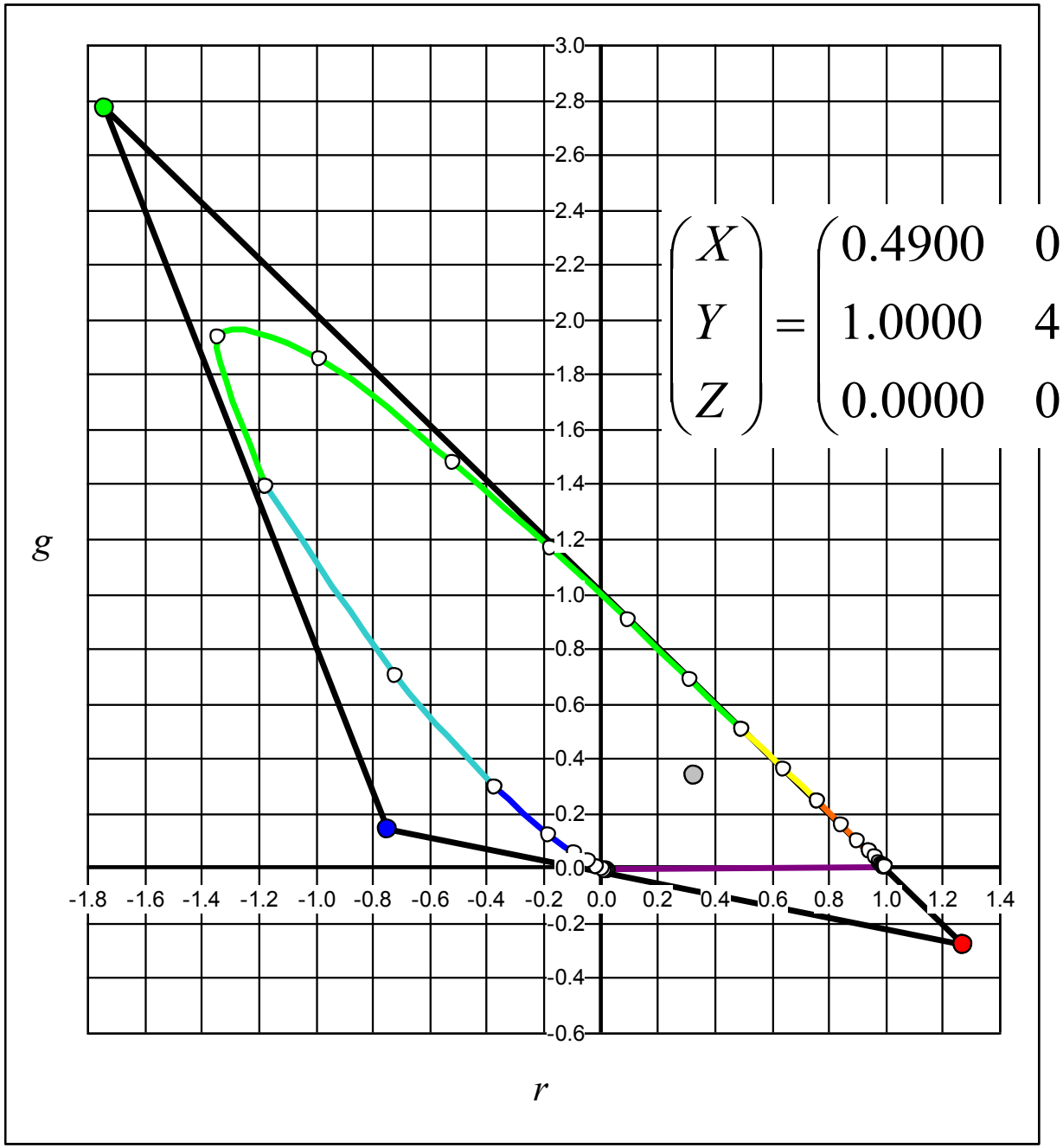
- $V = 1.000\ R + 4.591\ G + 0.060\ B$ – плоскость постоянной яркости

  $1.000\ R + 4.591\ G + 0.060\ B = 0$ – плоскость нулевой яркости

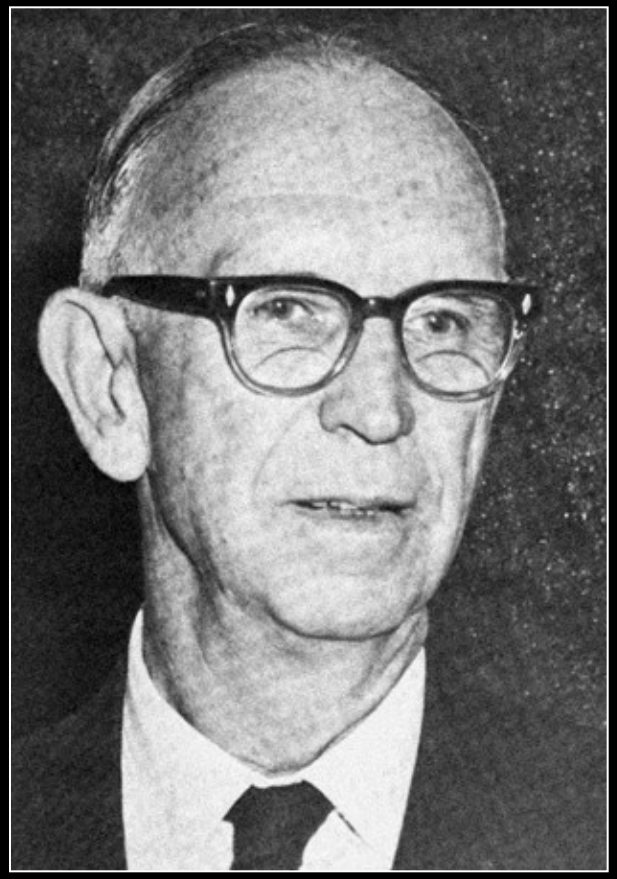
  Оси X и Z располагаются в плоскости нулевой яркости

- Значения $x(\lambda)$, $y(\lambda)$, $z(\lambda)$ неотрицательны.

Цветовое пространство МКО XYZ (CIE XYZ)

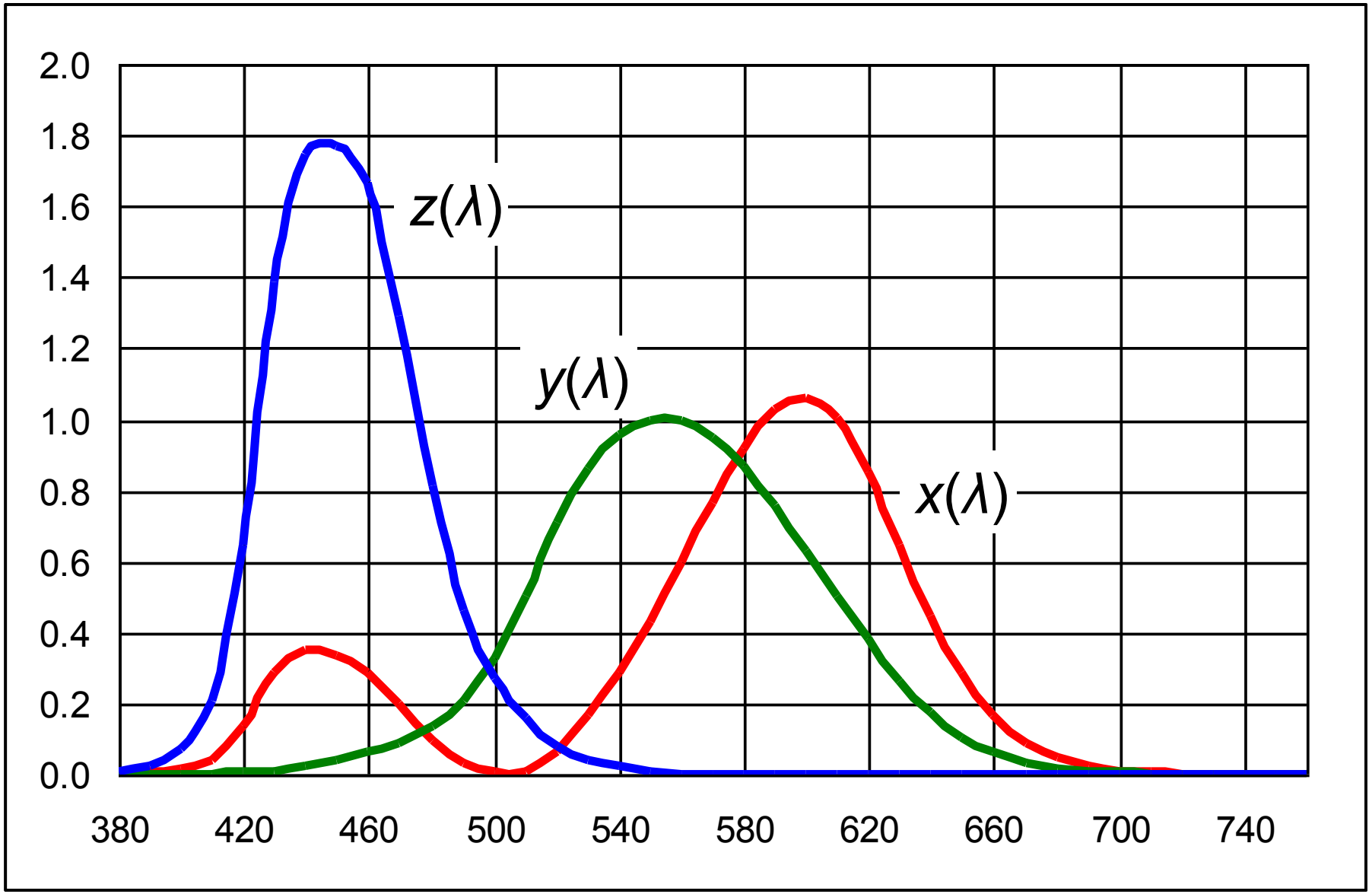


$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} 0.4900 & 0.3100 & 0.2000 \\ 1.0000 & 4.5907 & 0.0060 \\ 0.0000 & 0.0100 & 0.9900 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix}$$



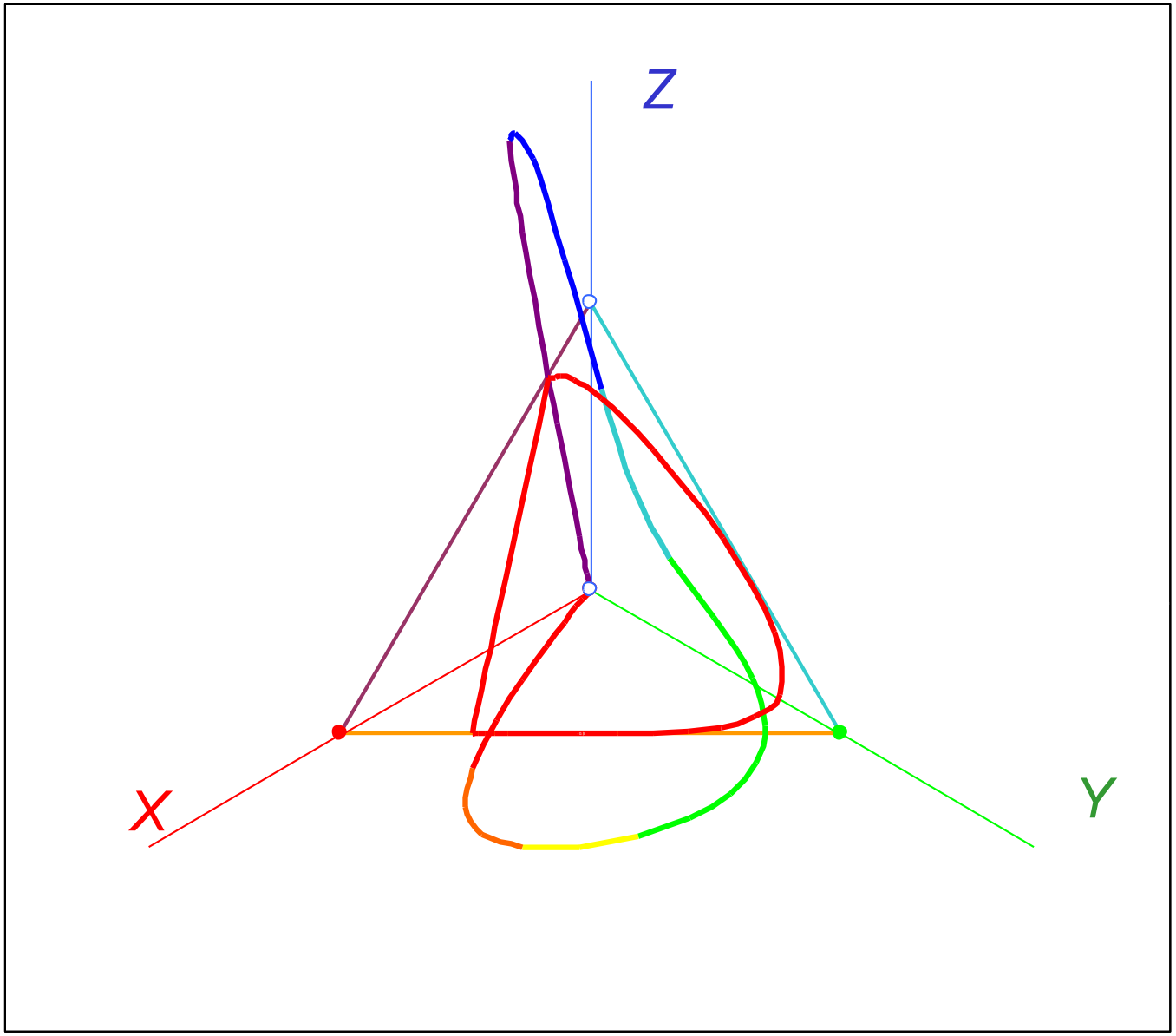
Deane Brewster Judd (Nov. 15, 1900 – Oct. 15, 1972) was an American physicist who made important contributions to the fields of colorimetry, color discrimination, color order, and color vision.

Цветовое пространство МКО XYZ (CIE XYZ)



Функции спектральной чувствительности (кривые сложения)

Цветовое пространство МКО XYZ (CIE XYZ)



Локус спектральных цветов и локус цветности

Цветовое пространство МКО XYZ (CIE XYZ) 1931

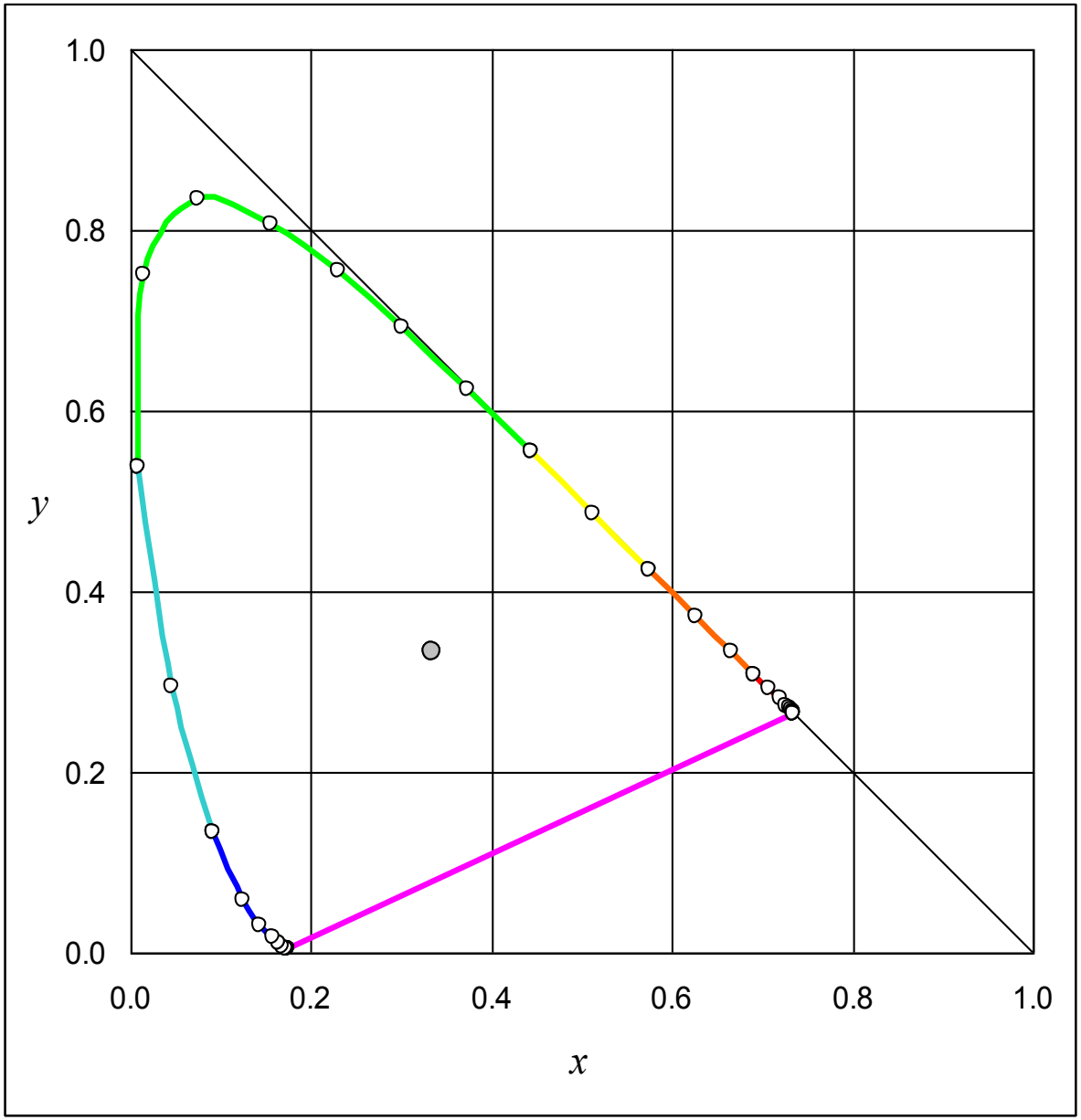
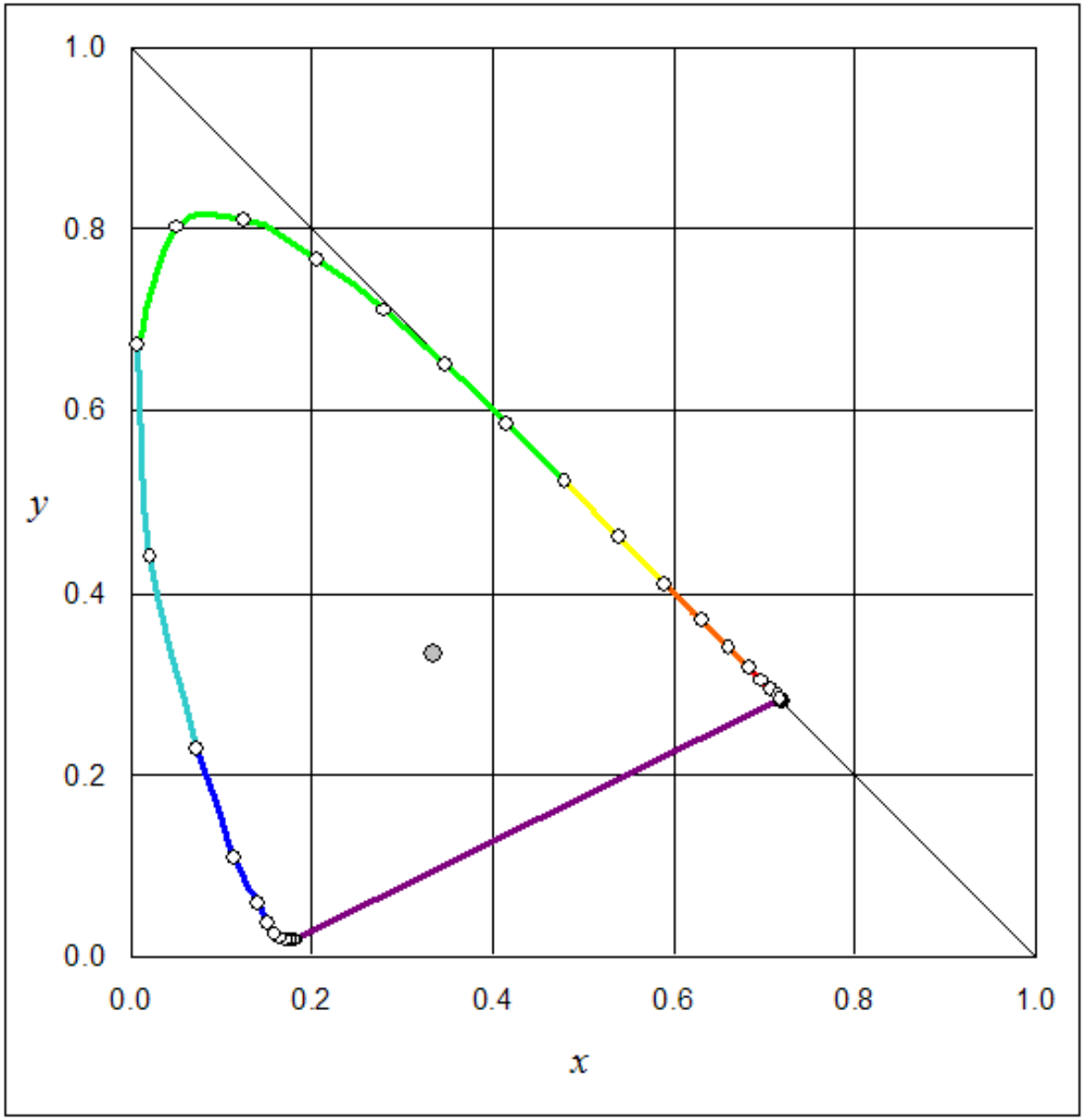


Диаграмма цветности

Диаграмма цветности CIE XYZ 1931 и CIE XYZ 1964 (10°)



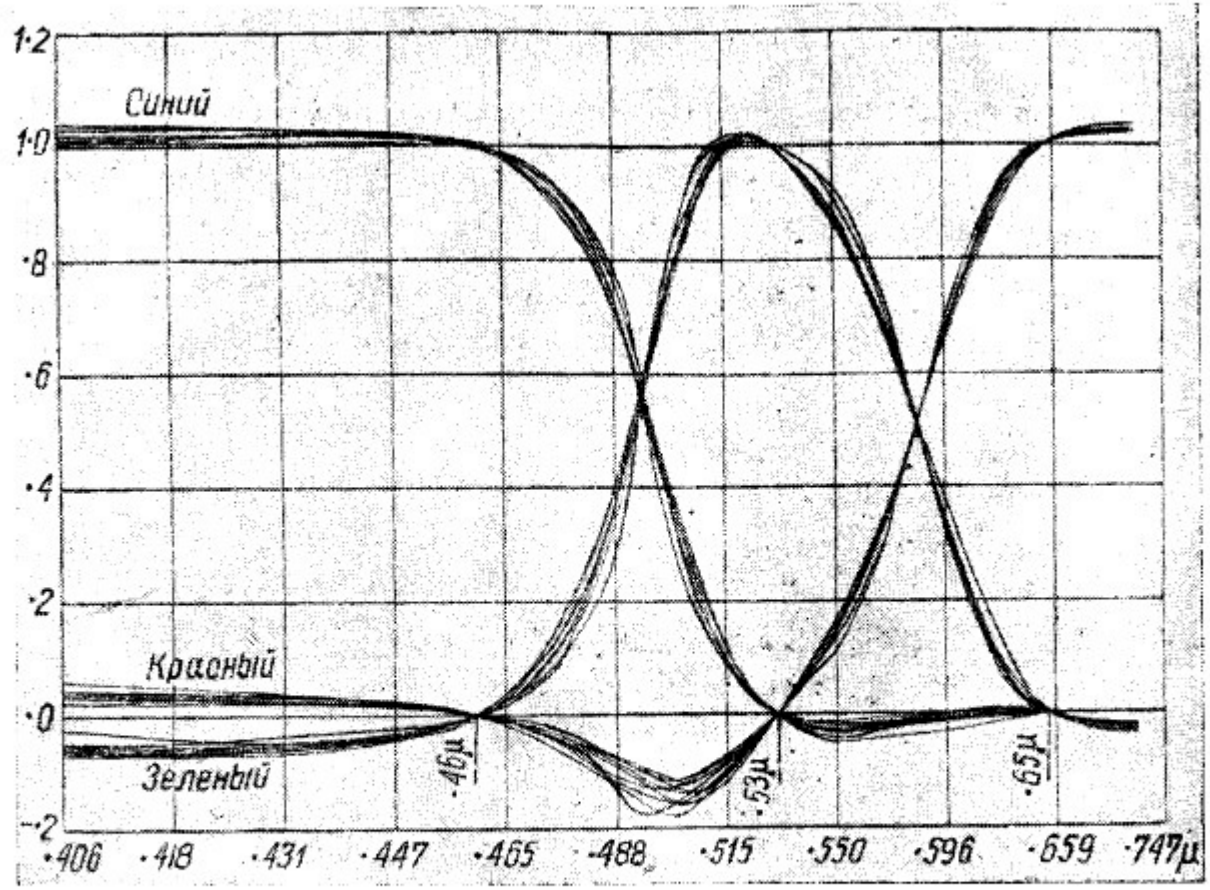
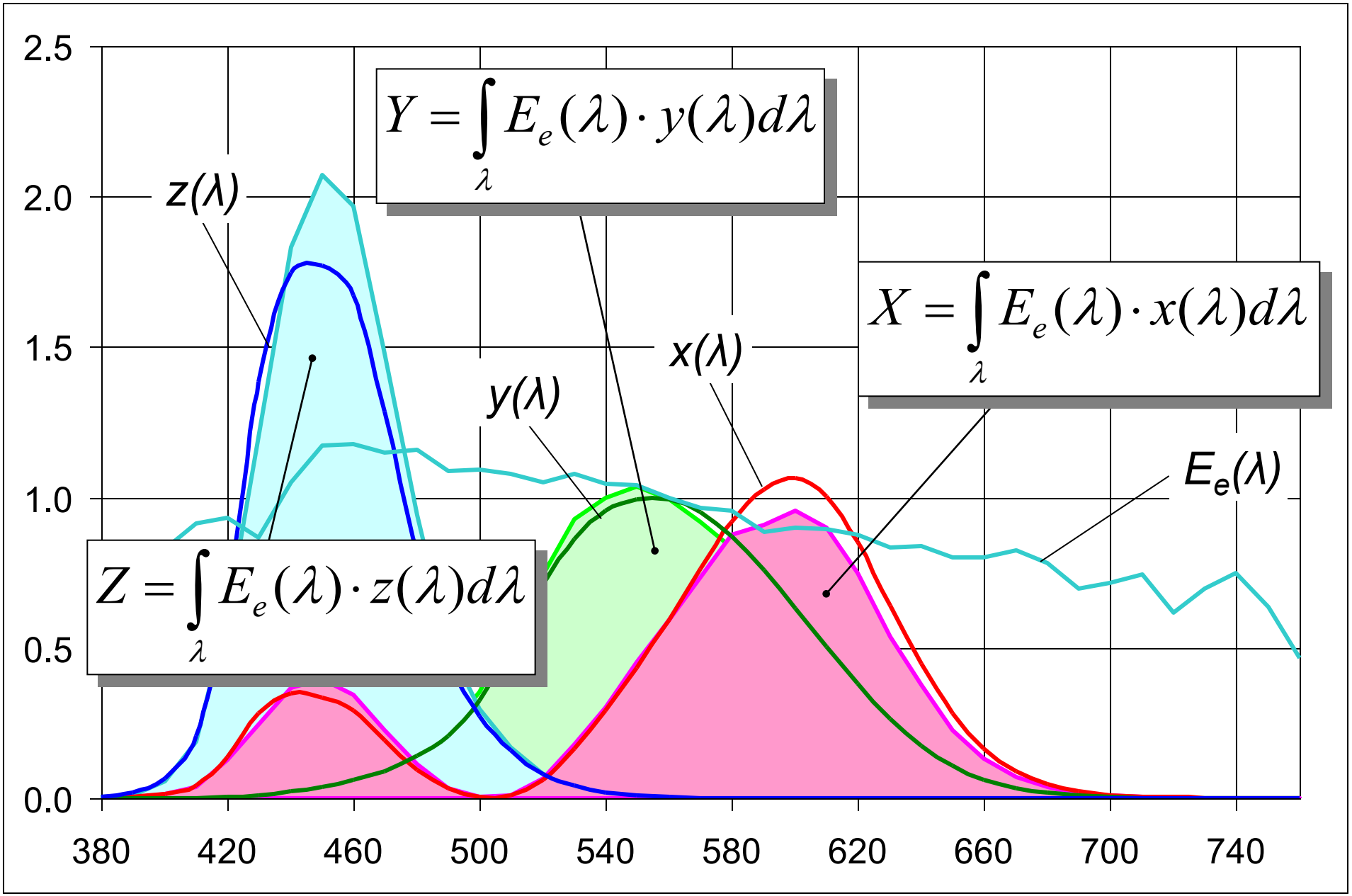


Рис. 14. Спектральные характеристики цветного зрения десяти нормальных наблюдателей.

Разброс измерения кривых сложения (М. М. Гуревич «Цвет и его измерение», М., 1950)

Расчет XYZ-координат



Координатная система xyY

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

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

$$Y = Y$$

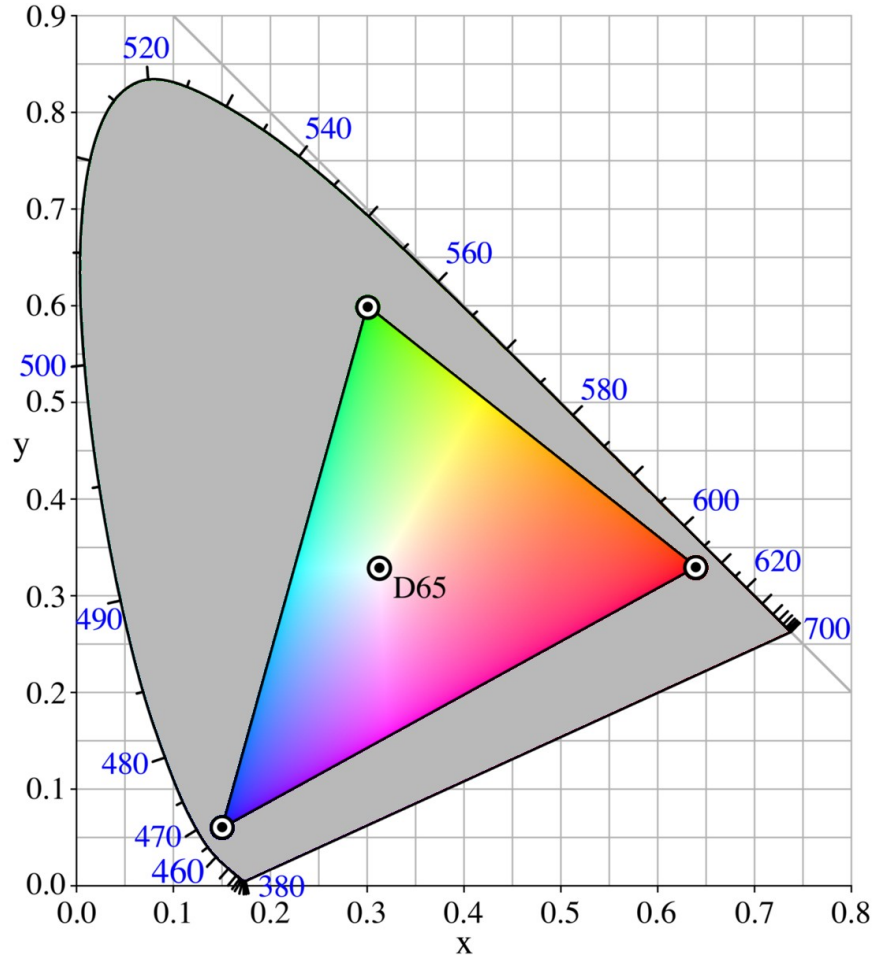
$$X = \frac{xY}{y}$$

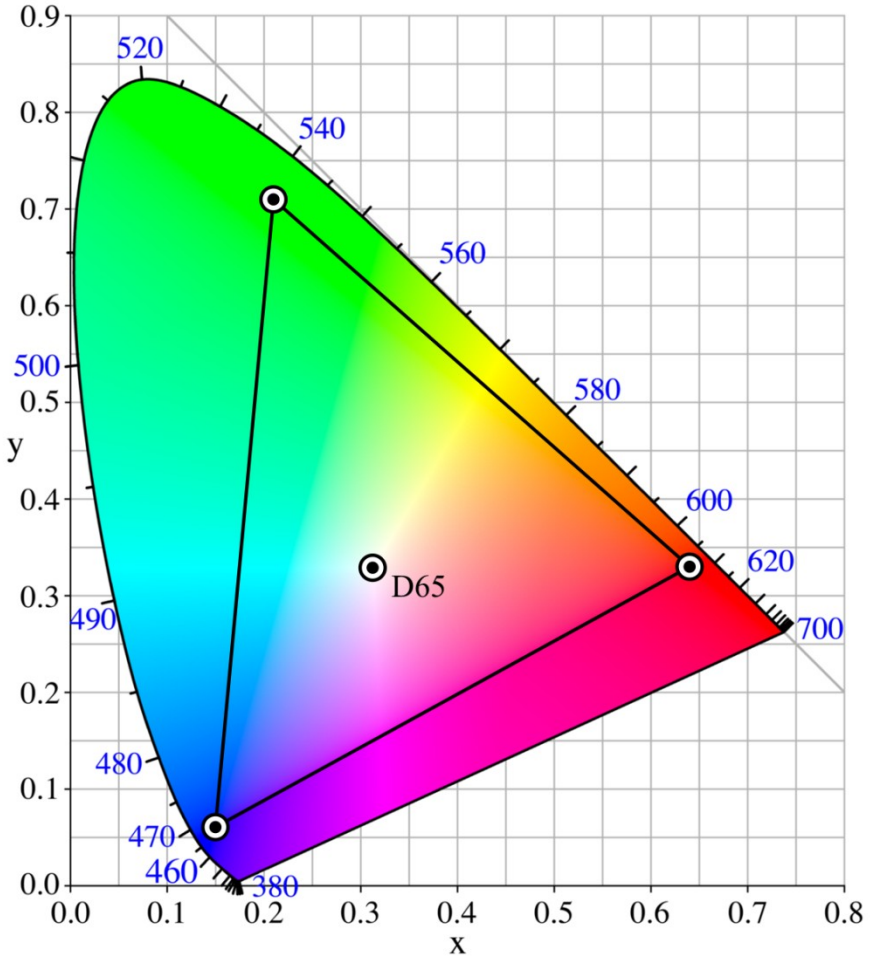
$$Y = Y$$

$$Z = \frac{(1-x-y)Y}{y}$$

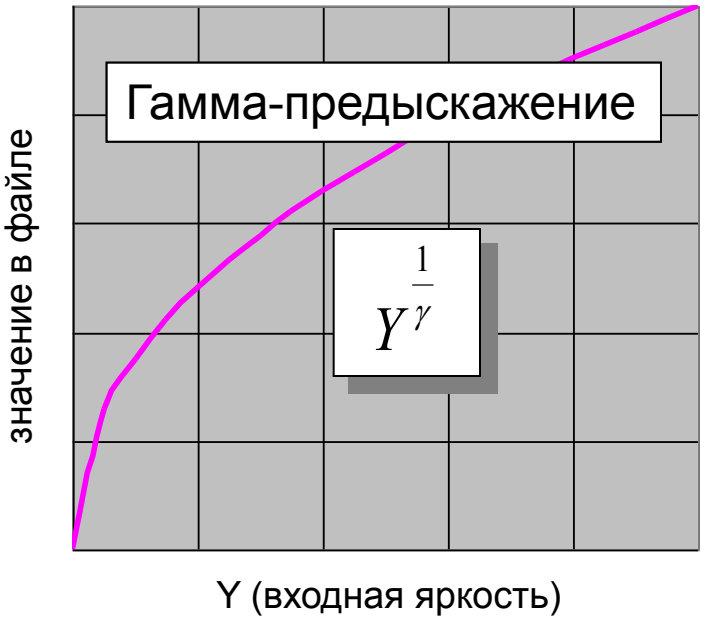
Цветовые пространства sRGB и Adobe RGB

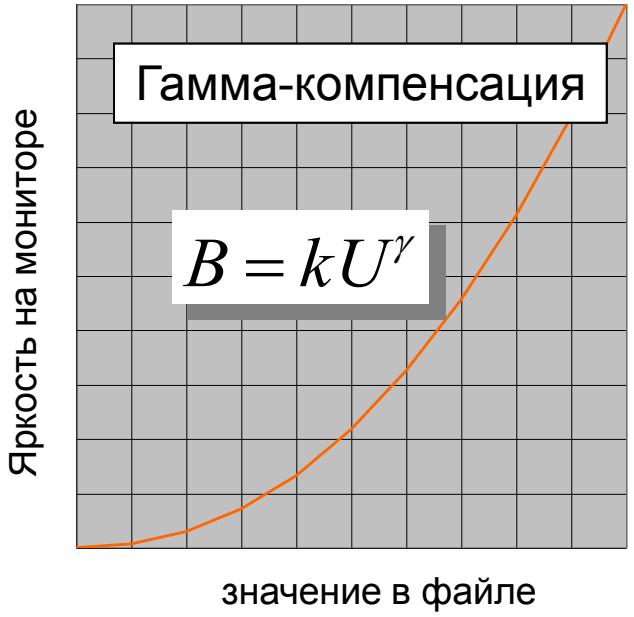
Название											
Adobe RGB (1998)	2.2	D65	0.6400	0.3300	0.2974	0.2100	0.7100	0.6274	0.1500	0.0600	0.0753
sRGB	≈2,2	D65	0.6400	0.3300	0.2127	0.3000	0.6000	0.7152	0.1500	0.0600	0.0722

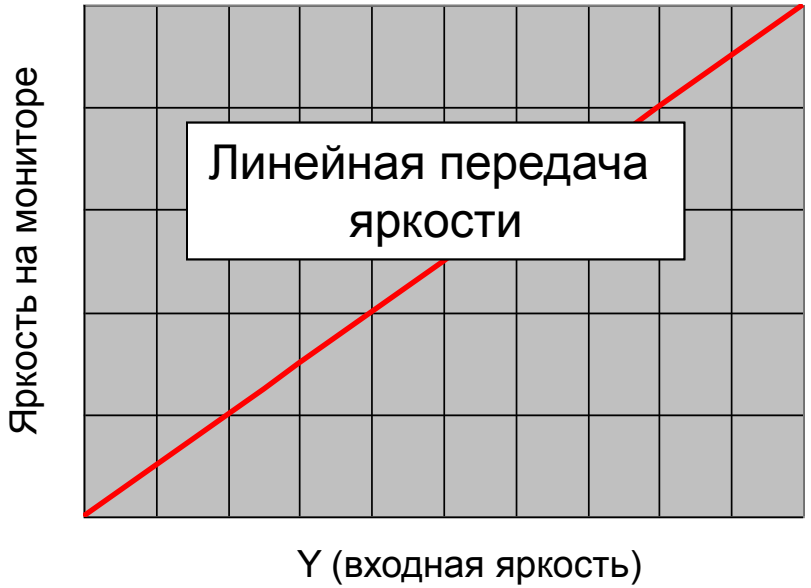




Гамма-коррекция







Вычисление sRGB-координат

1. Вычисление линейных RGB:

$$\begin{bmatrix} R_{\text{linear}} \\ G_{\text{linear}} \\ B_{\text{linear}} \end{bmatrix} = \begin{bmatrix} 3.2406 & -1.5372 & -0.4986 \\ -0.9689 & 1.8758 & 0.0415 \\ 0.0557 & -0.2040 & 1.0570 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

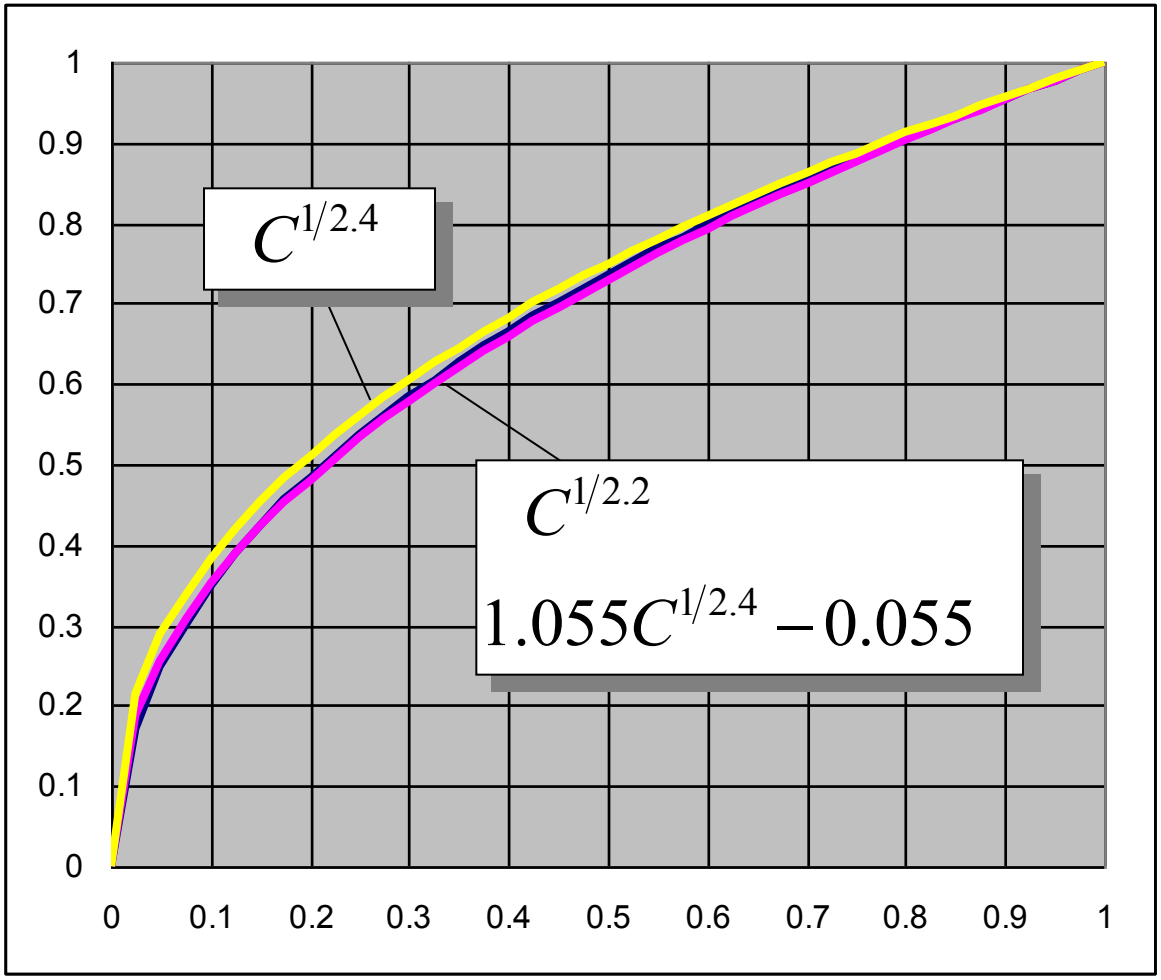
2. Вычисление нелинейных RGB:

$$C_{\text{srgb}} = \begin{cases} 12.92 C_{\text{linear}}, & C_{\text{linear}} \leq 0.0031308 \\ (1+a) C_{\text{linear}}^{1/2.4} - a, & C_{\text{linear}} > 0.0031308 \end{cases}$$

$$a = 0.055$$

$$R, G, B, X, Y, Z \in [0,1]$$

Упрощенное sRGB



$$C_{srgb} = C_{linear}^{1/2.2}$$

sRGB→XYZ

$$C_{\text{linear}} = \begin{cases} \frac{C_{\text{srgb}}}{12.92}, & C_{\text{srgb}} \le 0.04045 \\ \left(\frac{C_{\text{srgb}}+a}{1+a}\right)^{2.4}, & C_{\text{srgb}} > 0.04045 \end{cases}$$

$$a = 0.055$$

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.4124 & 0.3576 & 0.1805 \\ 0.2126 & 0.7152 & 0.0722 \\ 0.0193 & 0.1192 & 0.9505 \end{bmatrix} \begin{bmatrix} R_{\text{linear}} \\ G_{\text{linear}} \\ B_{\text{linear}} \end{bmatrix}$$

$$R, G, B, X, Y, Z \in [0,1]$$

Мораль

- Не существует единого RGB-пространства.
- Преобразования цветов на изображении, имеющие физический смысл, нужно выполнять, используя линейные цветовые RGB-пространства (XYZ, LMS, $sRGB_{linear}$, Lpy,).
- Закон сложения цветов действует только в линейных цветовых пространствах.
- sRGB – нелинейное цветовое пространство.

Числовой пример

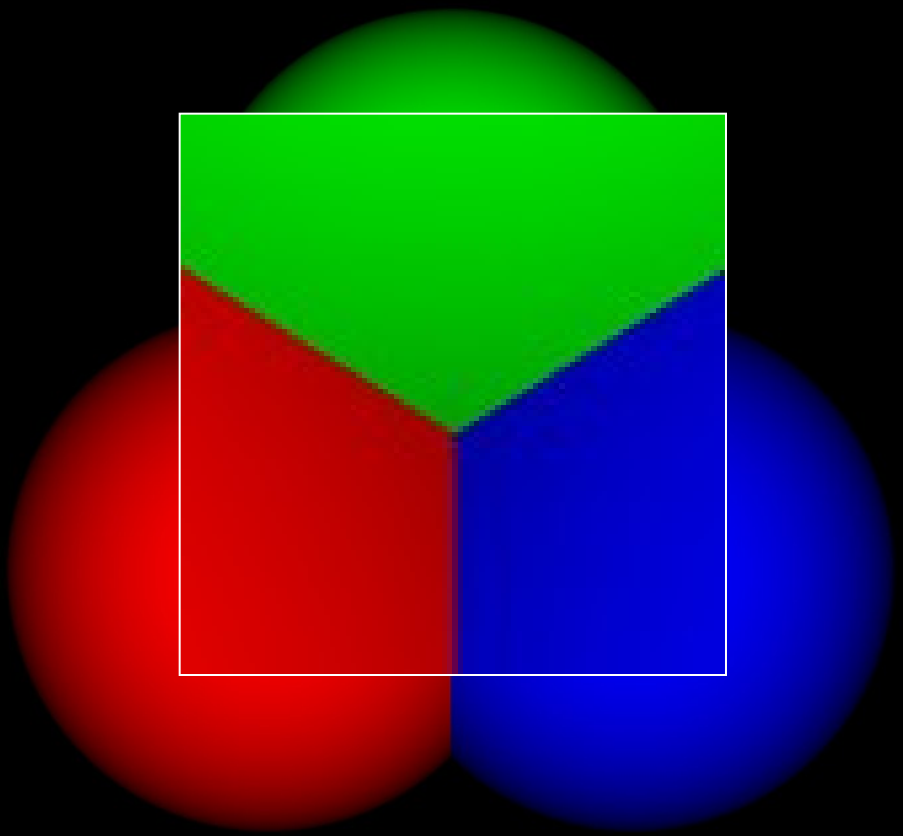
Среднее арифметическое 25 и 81 равно $\frac{25+81}{2}=53$

$\gamma = 2.0$ $\left(\frac{\sqrt{25}+\sqrt{81}}{2}\right)^2=49$

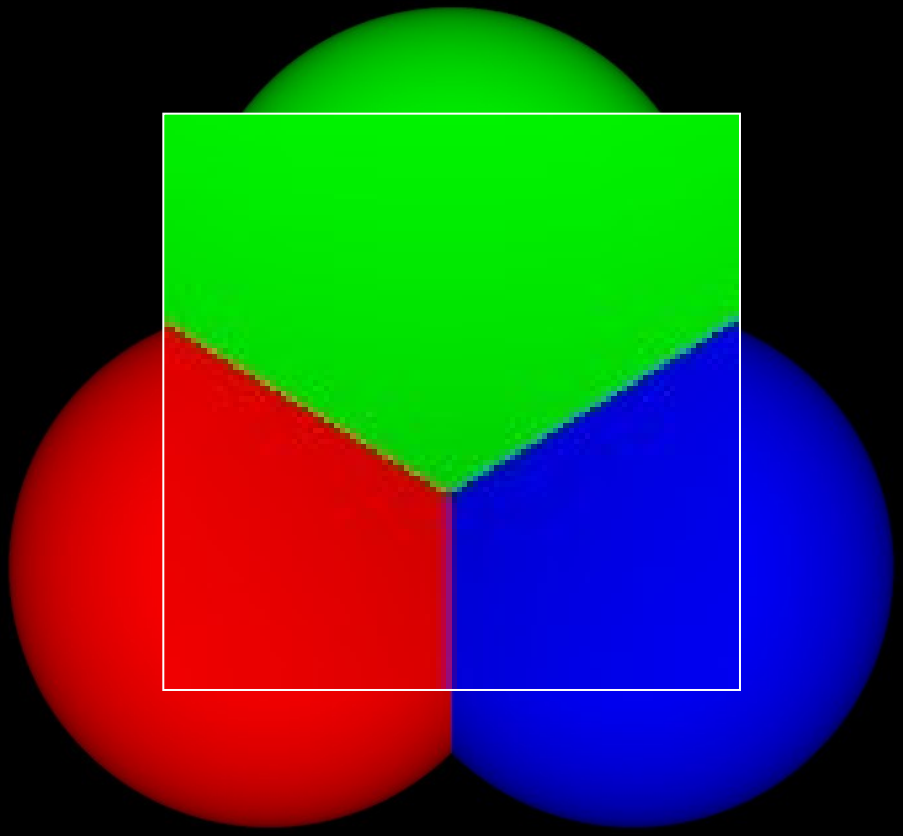
Среднее арифметическое 36 и 0 равно 18

$\gamma = 2.0$ $\left(\frac{\sqrt{36}}{2}\right)^2=9$

Пример
Трассировка лучей

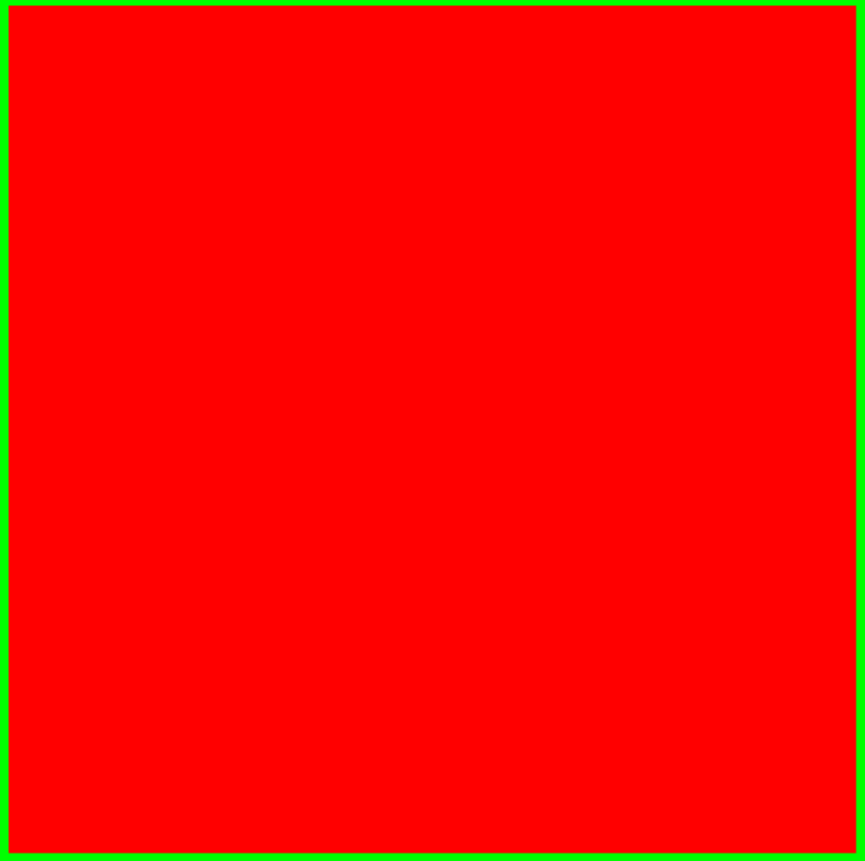


Расчет в нелинейном sRGB

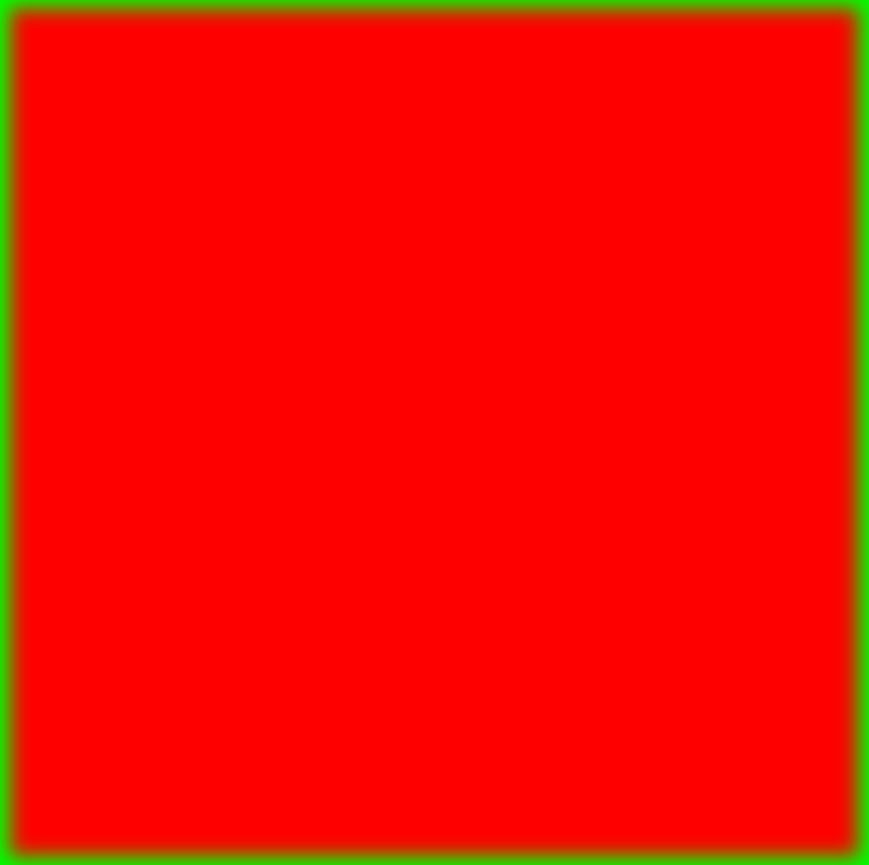


Расчет в линейном sRGB

Красное на зеленом



Красное на зеленом



Размытие по Гауссу. Adobe Photoshop CS6

Красное на зеленом

Светлота (преобразование в оттенки серого)

Размытие по Гауссу. Adobe Photoshop CS6

Красное на зеленом

Размытие по Гауссу. Плагин «C.F.Gauss»

Красное на зеленом

Светлота (преобразование в оттенки серого)

Размытие по Гауссу. Плагин «C.F.Gauss»

Цветовое пространство *Lab*

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

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

$$[f(Y/Y_n) - f(Z/Z_n)]$$

Светлота белого = 100 (отношение к яркости белого)

Для заданного белого a=0, b=0

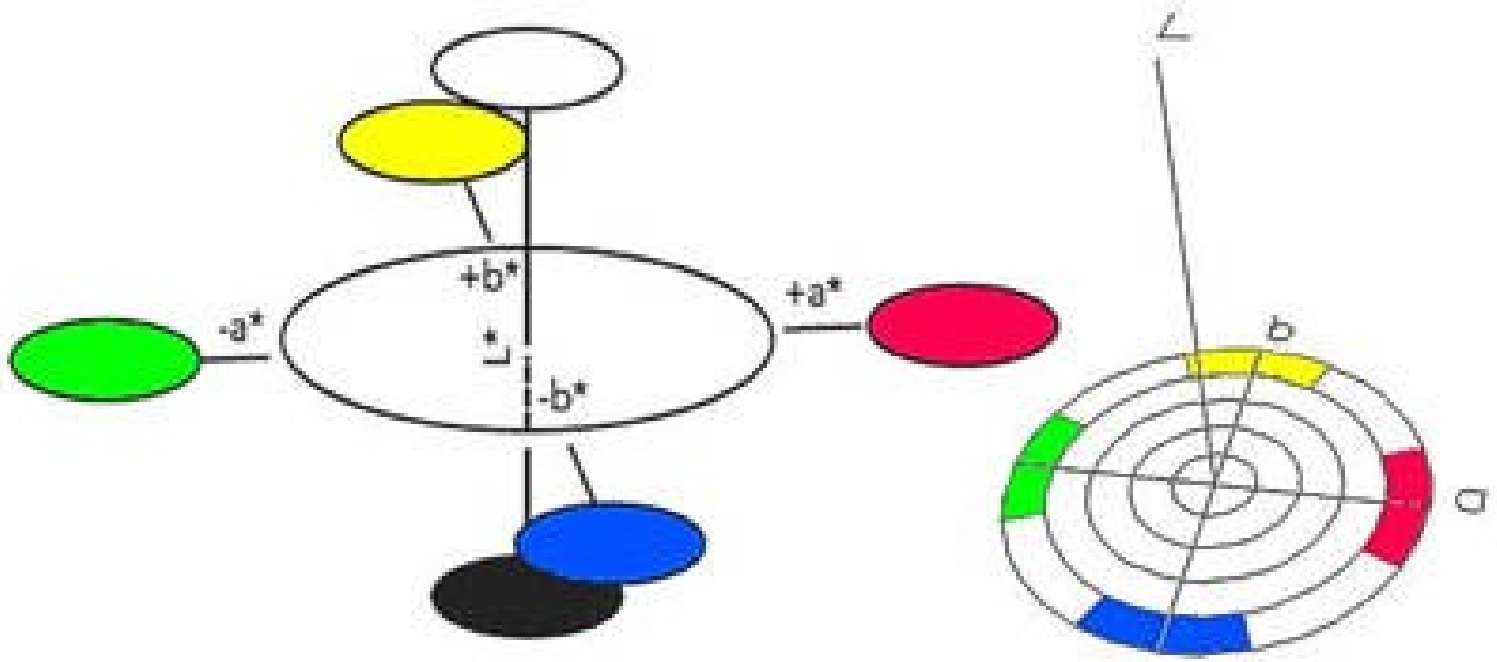
$$f(t) = \begin{cases} t^{1/3}, \\ \frac{1}{3}\left(\frac{29}{6}\right)^2 t + \frac{4}{29} \end{cases}$$

Значения X_n, Y_n и Z_n это координаты белой точки в значениях CIE XYZ (буква *n* означает «нормализованность»). По умолчанию опорный белый – D50.

Цветовое пространство *Lab*. *Lab* → *XYZ*

$$
\begin{aligned}
Y &= Y_n f^{-1}\left(\tfrac{1}{116}\left(L^* + 16\right)\right) \\
X &= X_n f^{-1}\left(\tfrac{1}{116}\left(L^* + 16\right) + \tfrac{1}{500}a^*\right) \\
Z &= Z_n f^{-1}\left(\tfrac{1}{116}\left(L^* + 16\right) - \tfrac{1}{200}b^*\right)
\end{aligned}
$$

$$
f^{-1}(t) = \begin{cases} t^3 & \text{if } t > \tfrac{6}{29} \\ 3\left(\tfrac{6}{29}\right)^2\left(t - \tfrac{4}{29}\right) & \text{otherwise} \end{cases}
$$



CIE Lab

Формула цветового отличия (1976):

$$\Delta E = \sqrt{(\Delta L)^2 + (\Delta a)^2 + (\Delta b)^2}$$

CIE LCh_{ab}

$$C^*_{ab} = \sqrt{a^{*2} + b^{*2}}$$

$$h^*_{ab} = arctg\left(\frac{b^*}{a^*}\right)$$

CIE Lab

- Приблизительно равномерное по ощущению цветовое пространство (перцепционное пространство)
- Нелинейное цветовое пространство
- Аппаратно-независимое стандартизованное пространство

Дополнительные главы компьютерной графики

Лекция 4

Copyright © С. Свердлов, 2009-2017
http://www.uni-vologda.ac.ru/~c3c

Ортогональная координатная система для RGB-пространства

Ортогональная координатная система для RGB-пространства

1. $R = G = B$ – «серая» ось светлоты

Ортогональная координатная система для RGB-пространства

1. $R = G = B$ – «серая» ось светлоты

2. $0 \le R, G, B \le 1$

Ортогональная координатная система для RGB-пространства

1. $R = G = B$ – «серая» ось светлоты

2. $0 \le R, G, B \le 1$

3. Светлота $L = Y$

Ортогональная координатная система для RGB-пространства

1. $R = G = B$ – «серая» ось светлоты

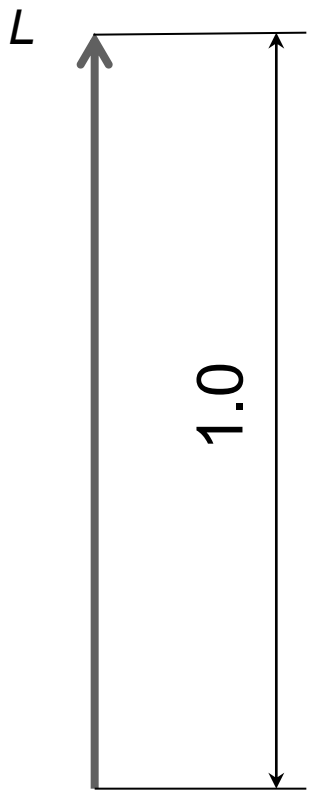
2. $0 \le R, G, B \le 1$

3. Светлота $L = Y$

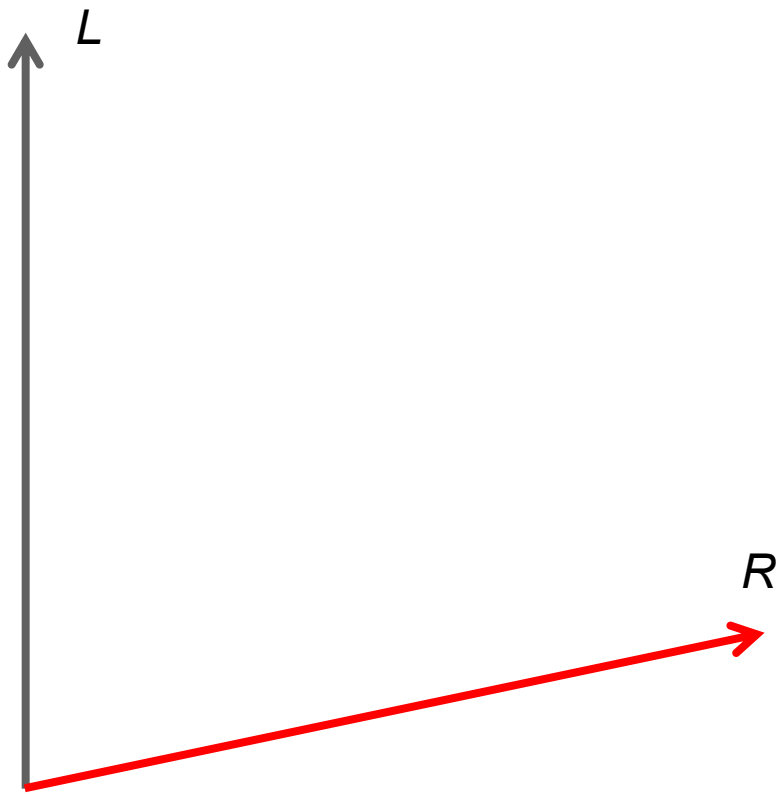
$$L = 0.2126R + 0.7152G + 0.0722B$$

Проекции ортов *RGB* на ось *L*

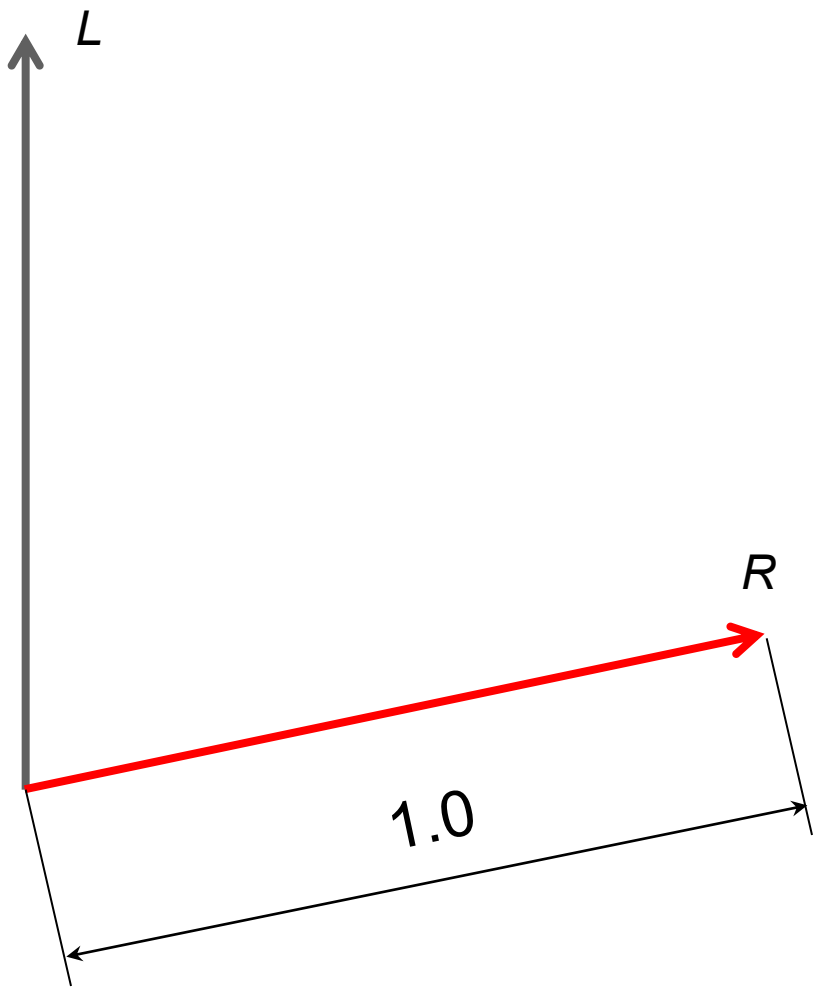
Проекции ортов *RGB* на ось *L*



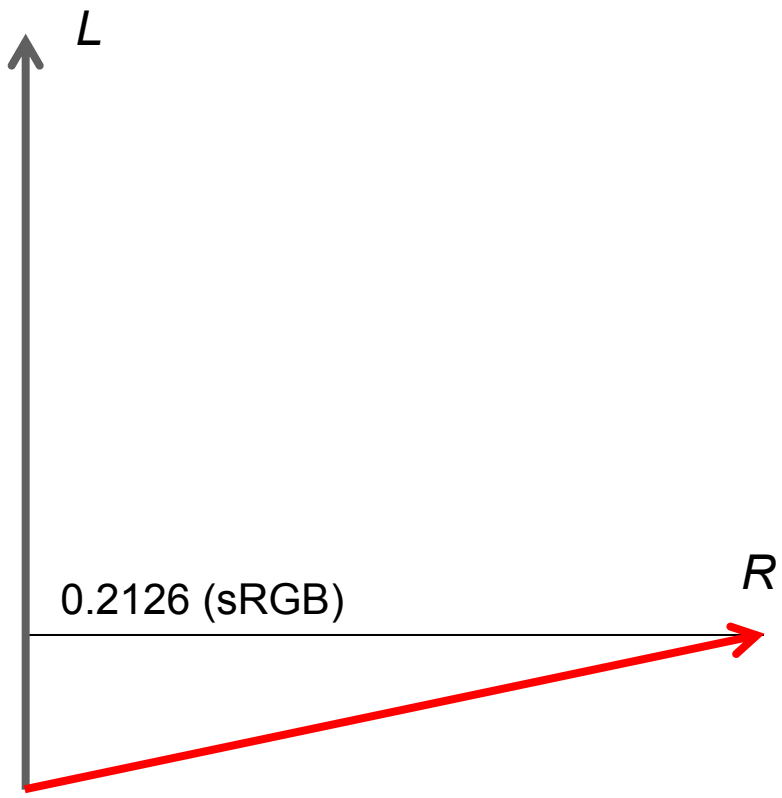
Проекции ортов *RGB* на ось *L*



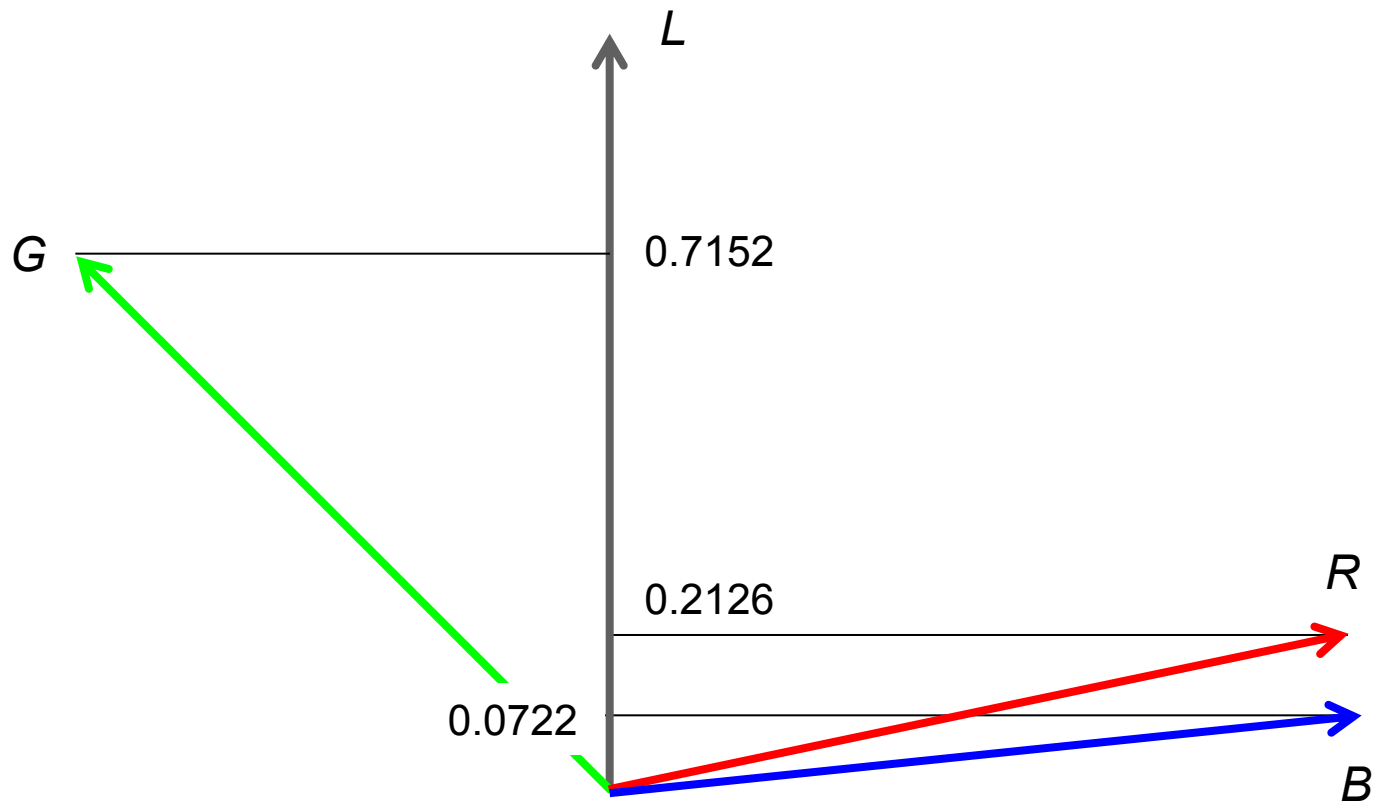
Проекции ортов *RGB* на ось *L*



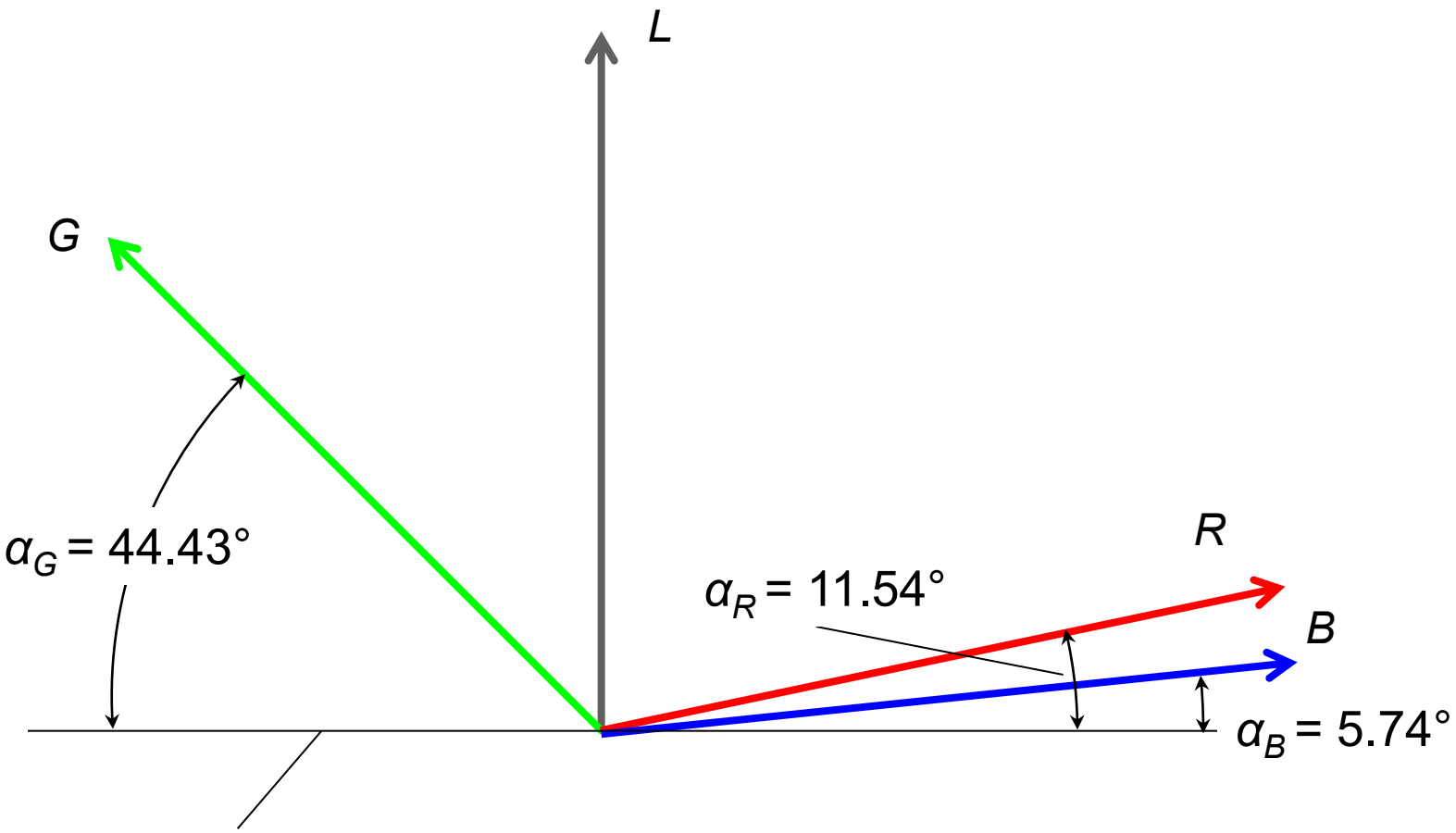
Проекции ортов *RGB* на ось *L*



Проекции ортов *RGB* на ось *L*

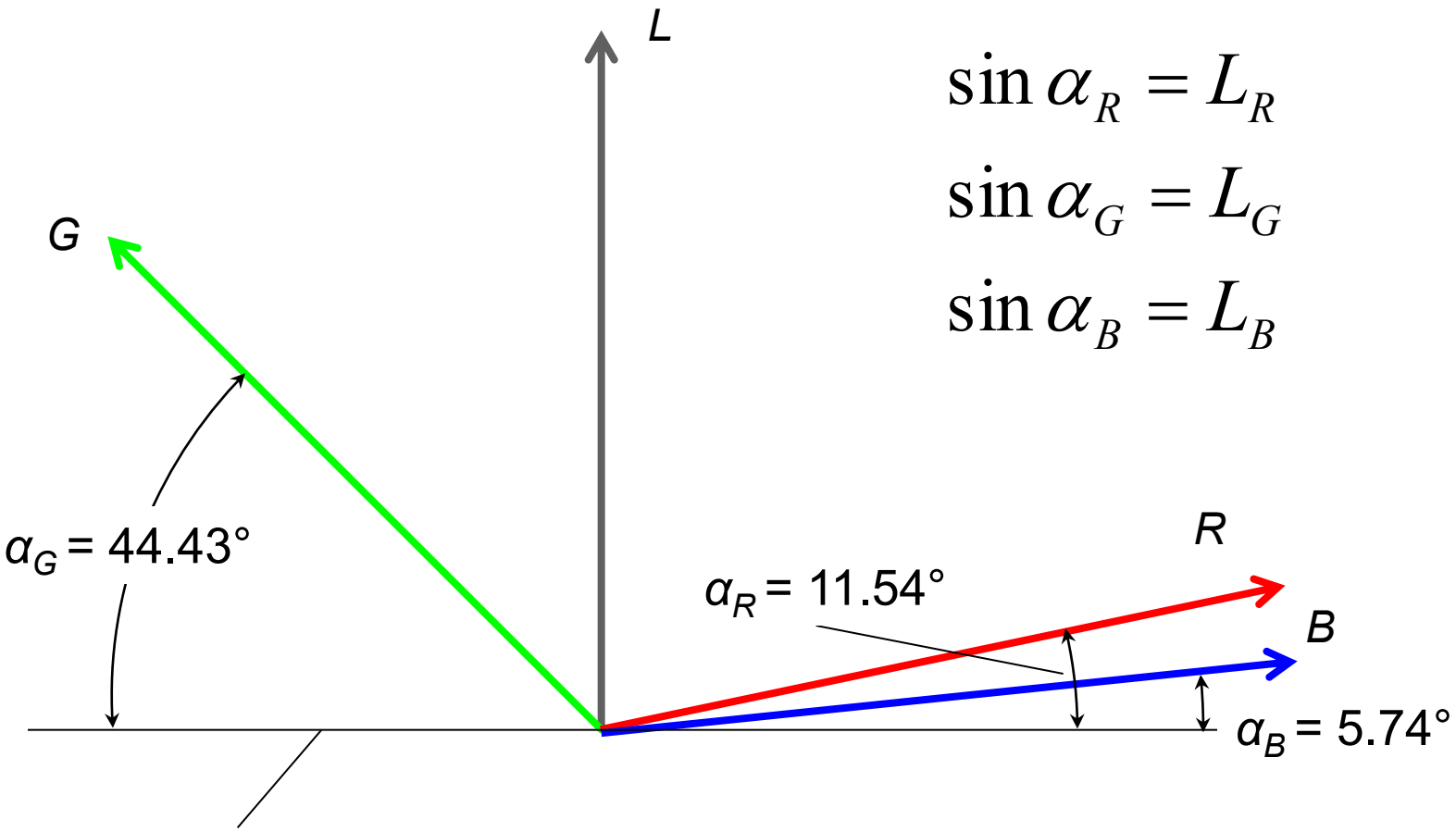


Проекции ортов *RGB* на ось *L*



Плоскость (*L* = 0), в которой расположены две оставшиеся оси ортогональной ЦКС

Проекции ортов *RGB* на ось *L*



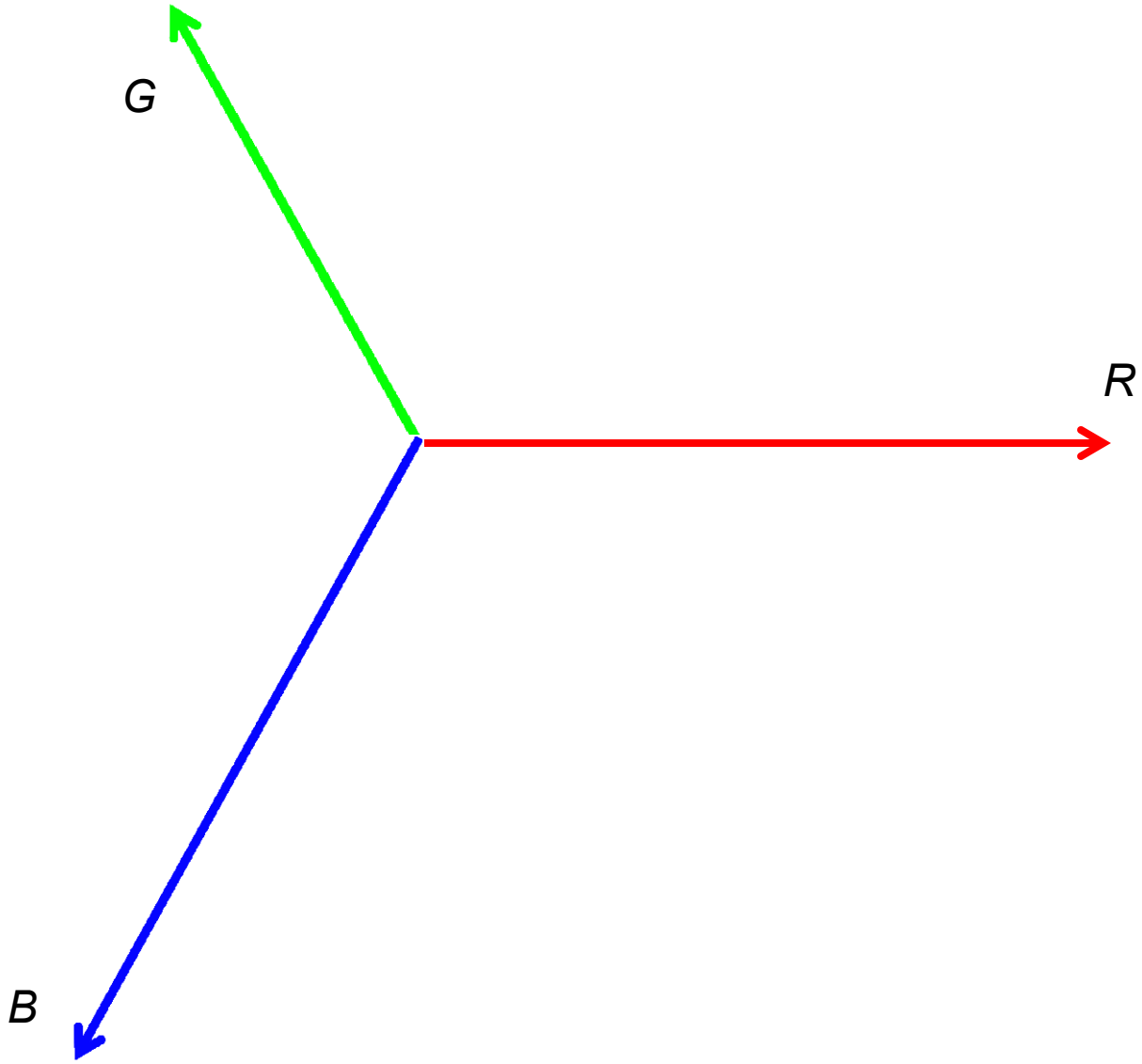
$$\sin\alpha_R = L_R$$

$$\sin\alpha_G = L_G$$

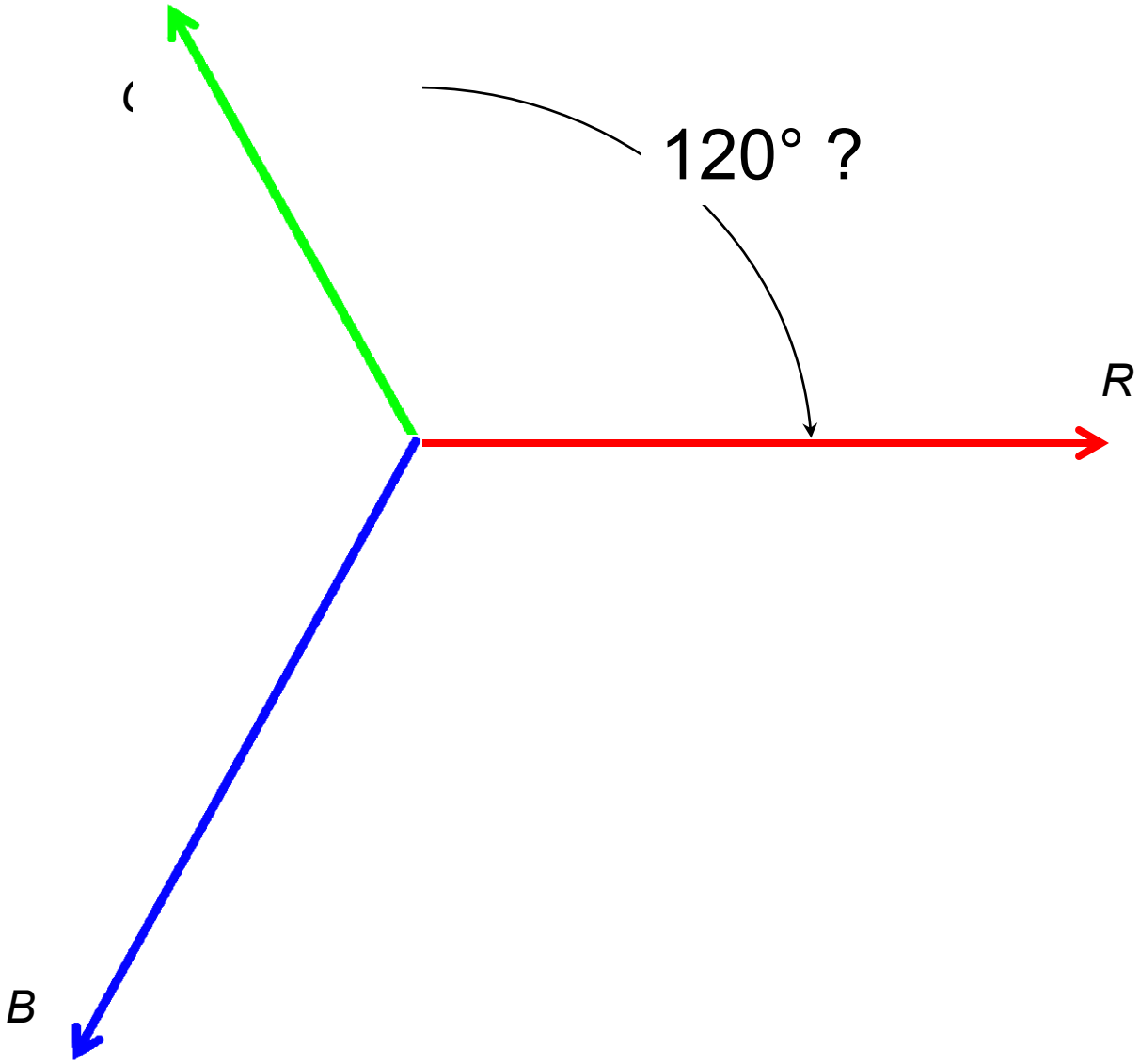
$$\sin\alpha_B = L_B$$

Плоскость (*L* = 0), в которой расположены две оставшиеся оси ортогональной ЦКС

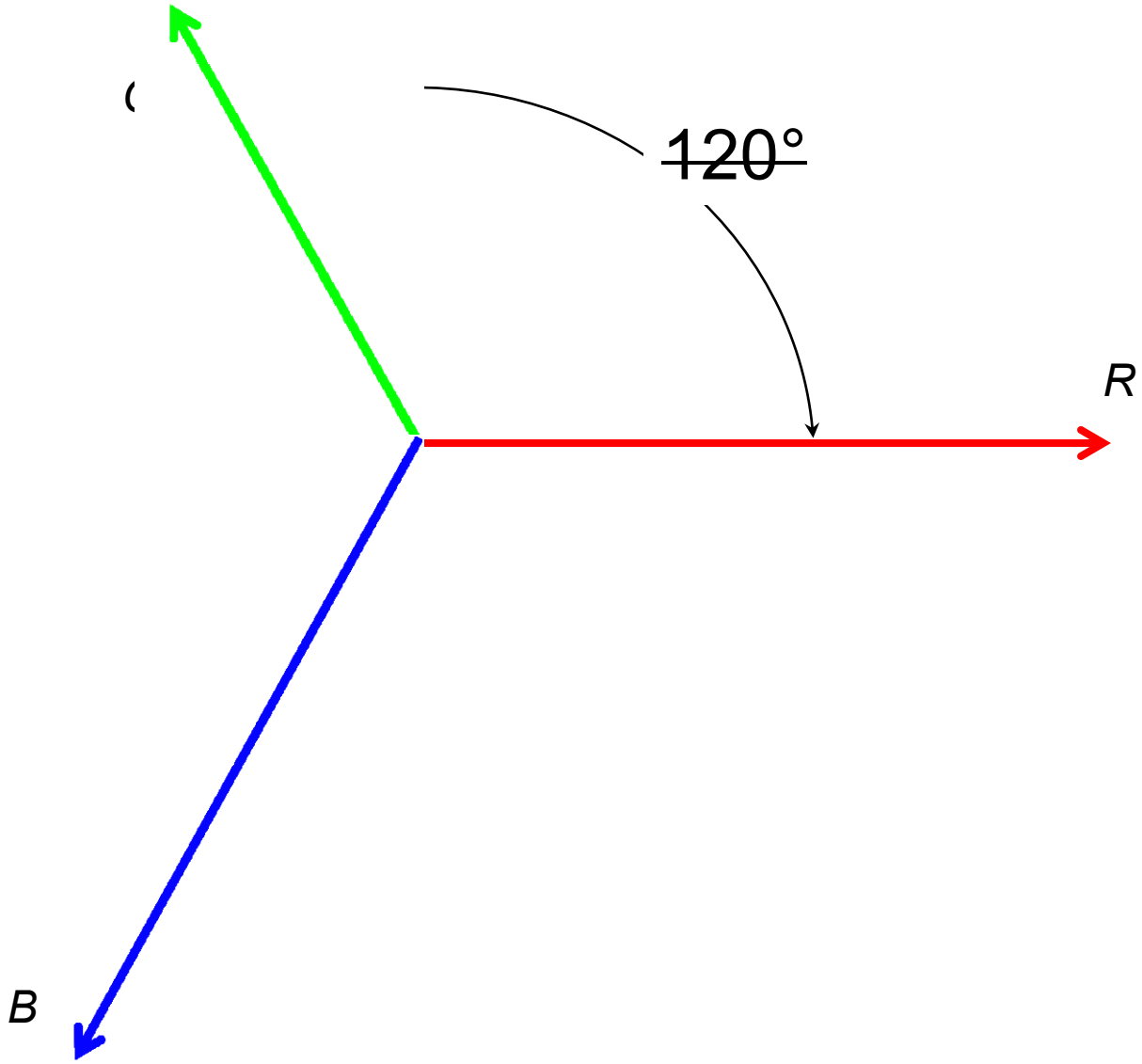
Положение осей *RGB* по цветовому тону. Вид с вершины оси *L*



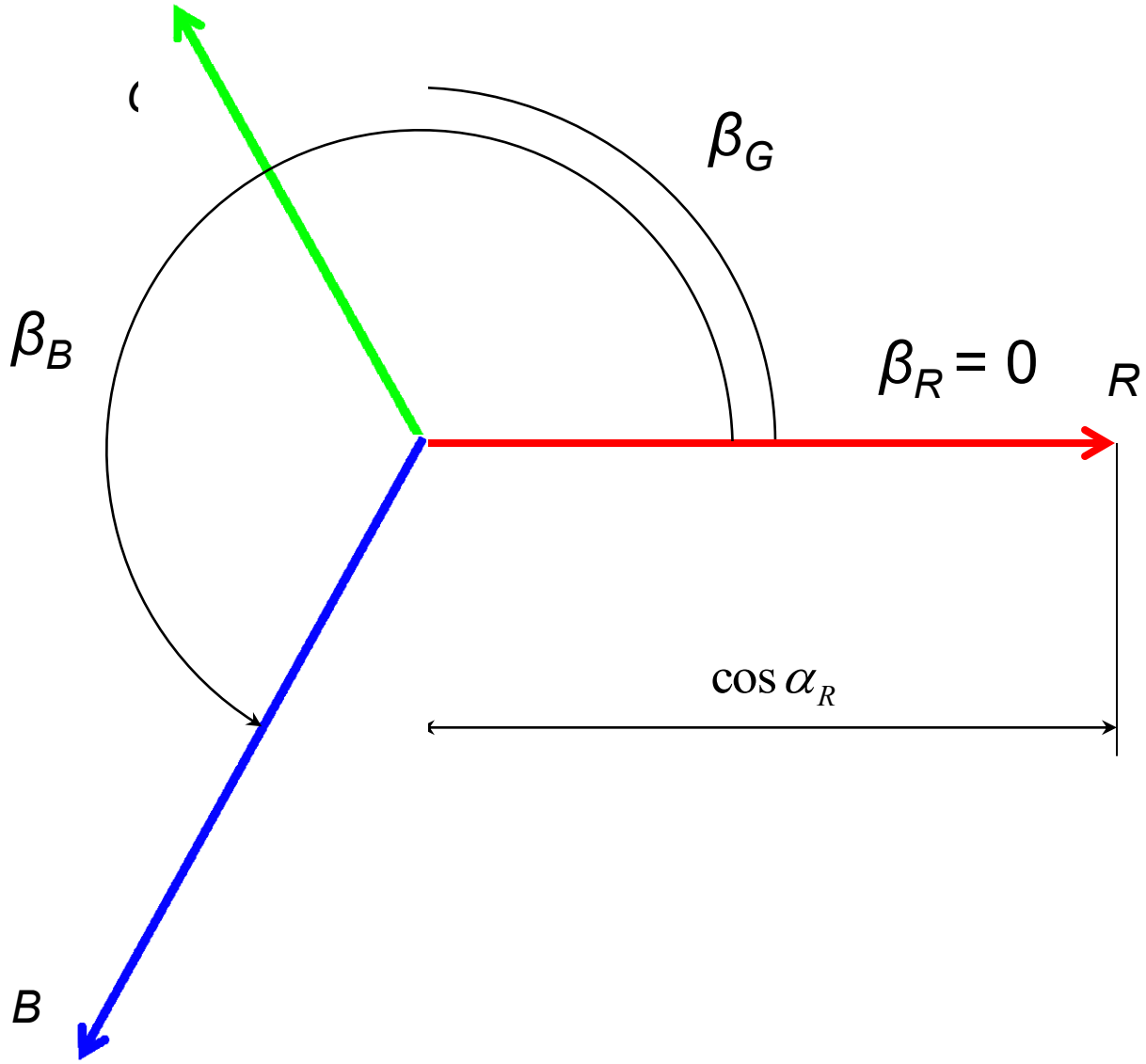
Положение осей *RGB* по цветовому тону. Вид с вершины оси *L*



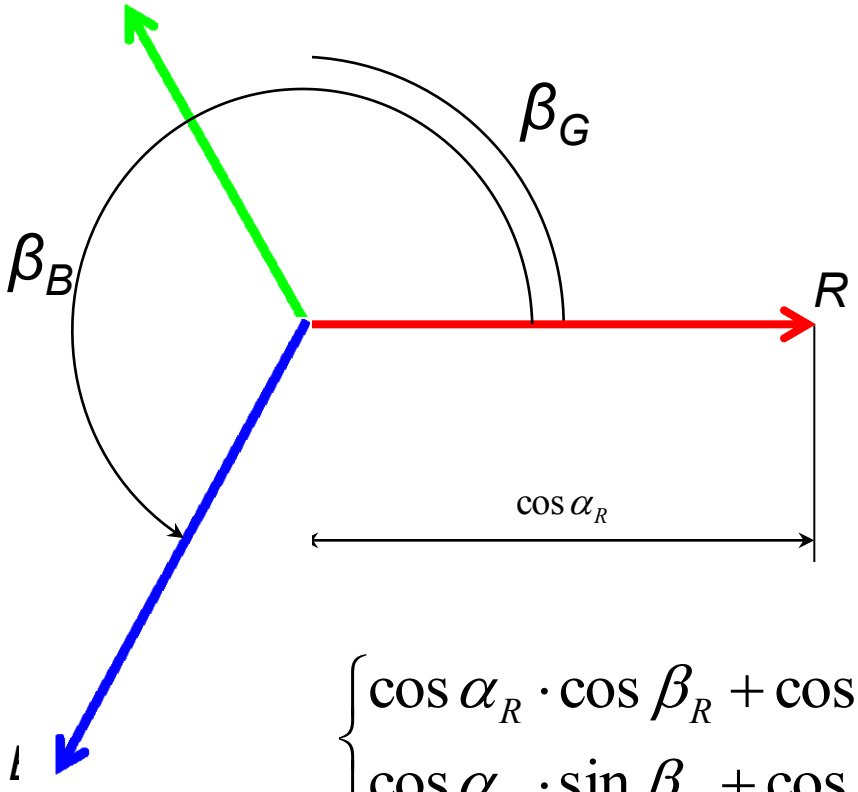
Положение осей *RGB* по цветовому тону. Вид с вершины оси *L*



Положение осей *RGB* по цветовому тону. Вид с вершины оси *L*

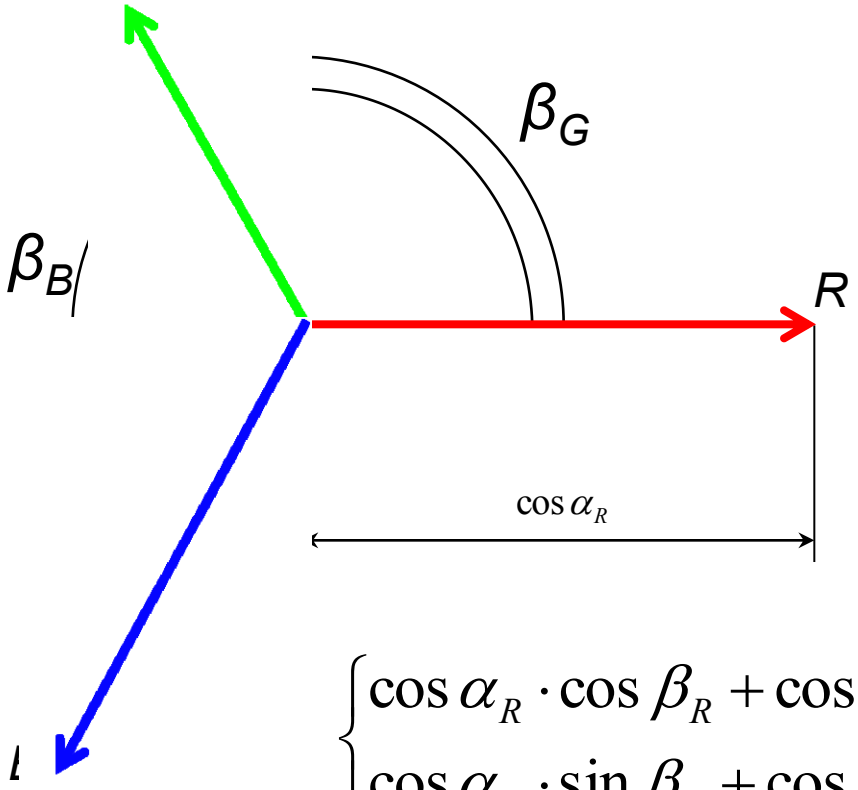


Положение осей *RGB* по цветовому тону. Вид с вершины оси *L*



$$\begin{cases} \cos\alpha_R \cdot \cos\beta_R + \cos\alpha_G \cdot \cos\beta_G + \cos\alpha_B \cdot \cos\beta_B = 0 \\ \cos\alpha_R \cdot \sin\beta_R + \cos\alpha_G \cdot \sin\beta_G + \cos\alpha_B \cdot \sin\beta_B = 0 \end{cases}$$

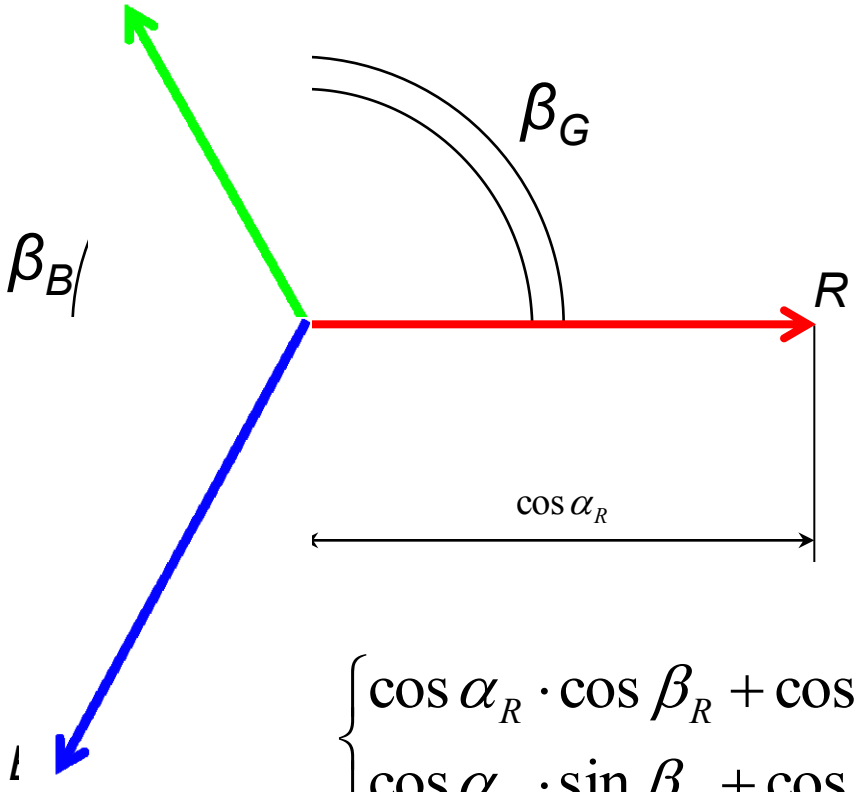
Положение осей *RGB* по цветовому тону. Вид с вершины оси *L*



$$\begin{cases} \cos\alpha_R \cdot \cos\beta_R + \cos\alpha_G \cdot \cos\beta_G + \cos\alpha_B \cdot \cos\beta_B = 0 \\ \cos\alpha_R \cdot \sin\beta_R + \cos\alpha_G \cdot \sin\beta_G + \cos\alpha_B \cdot \sin\beta_B = 0 \end{cases}$$

$$\begin{cases} \cos\alpha_R \cdot 1 + \cos\alpha_G \cdot \cos\beta_G + \cos\alpha_B \cdot \cos\beta_B = 0 \\ \cos\alpha_R \cdot 0 + \cos\alpha_G \cdot \sin\beta_G + \cos\alpha_B \cdot \sin\beta_B = 0 \end{cases}$$

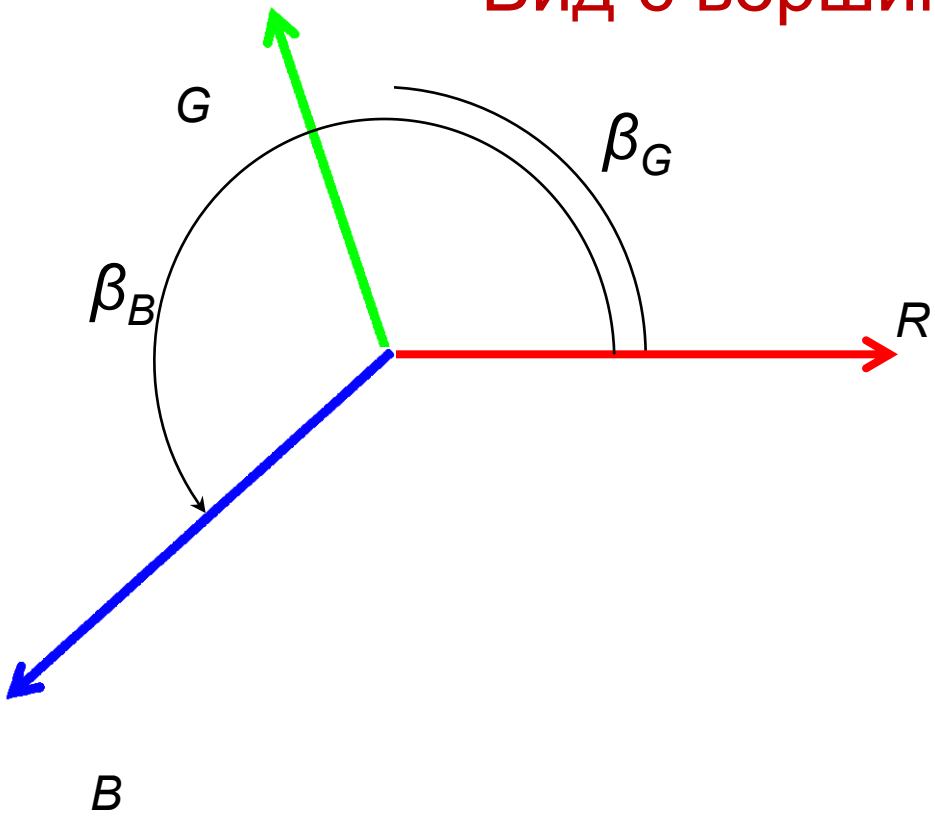
Положение осей *RGB* по цветовому тону. Вид с вершины оси *L*



$$\begin{cases} \cos\alpha_R \cdot \cos\beta_R + \cos\alpha_G \cdot \cos\beta_G + \cos\alpha_B \cdot \cos\beta_B = 0 \\ \cos\alpha_R \cdot \sin\beta_R + \cos\alpha_G \cdot \sin\beta_G + \cos\alpha_B \cdot \sin\beta_B = 0 \end{cases}$$

$$\begin{cases} \cos\alpha_G \cdot \cos\beta_G + \cos\alpha_B \cdot \cos\beta_B = -\cos\alpha_R \\ \cos\alpha_G \cdot \sin\beta_G + \cos\alpha_B \cdot \sin\beta_B = 0 \end{cases}$$

Положение осей *RGB* по цветовому тону. Вид с вершины оси *L*

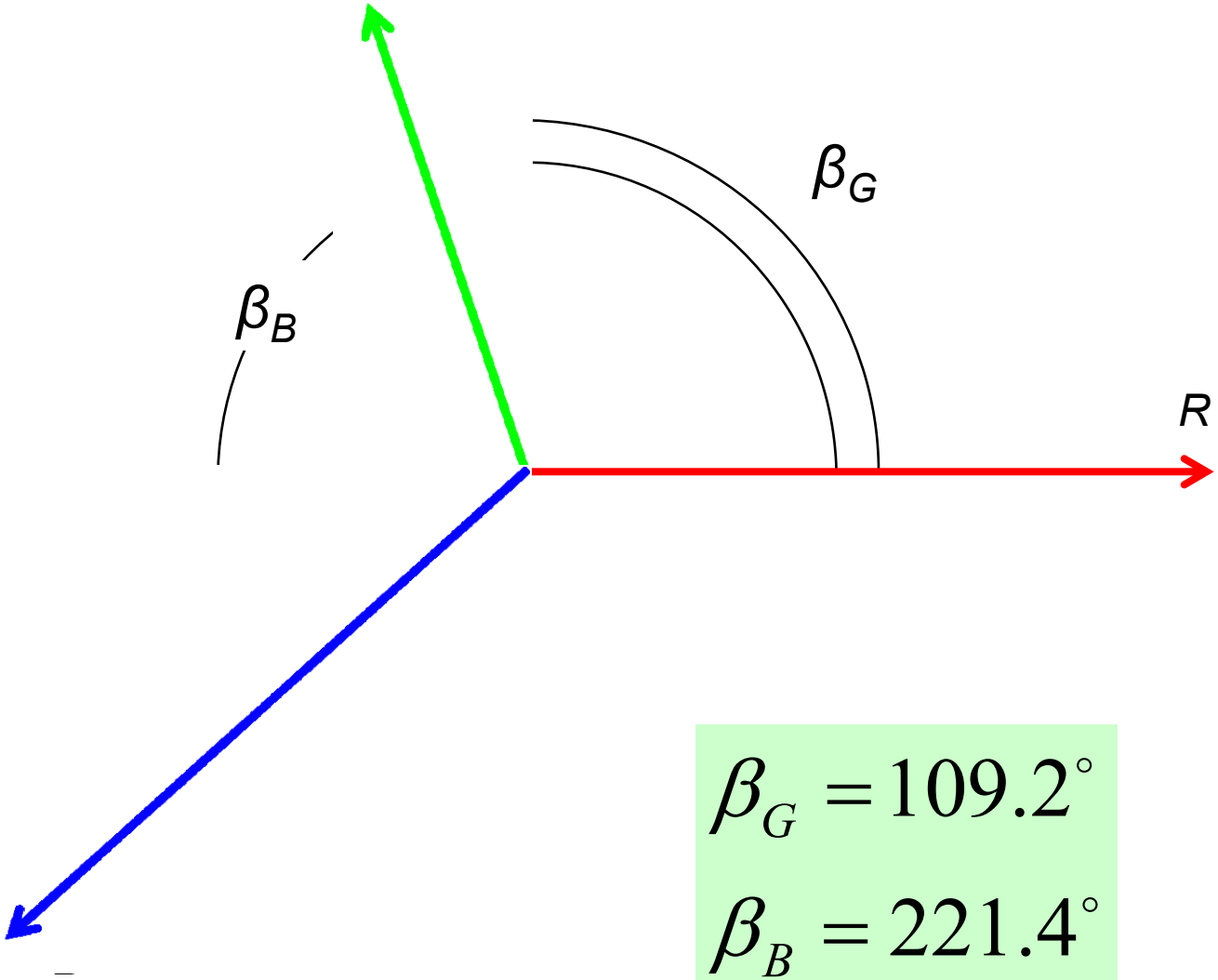


$$\beta_G = 109.2^\circ$$

$$\beta_B = 221.4^\circ$$

$$\begin{cases} \cos\alpha_G \cdot \cos\beta_G + \cos\alpha_B \cdot \cos\beta_B = -\cos\alpha_R \\ \cos\alpha_G \cdot \sin\beta_G + \cos\alpha_B \cdot \sin\beta_B = 0 \end{cases}$$

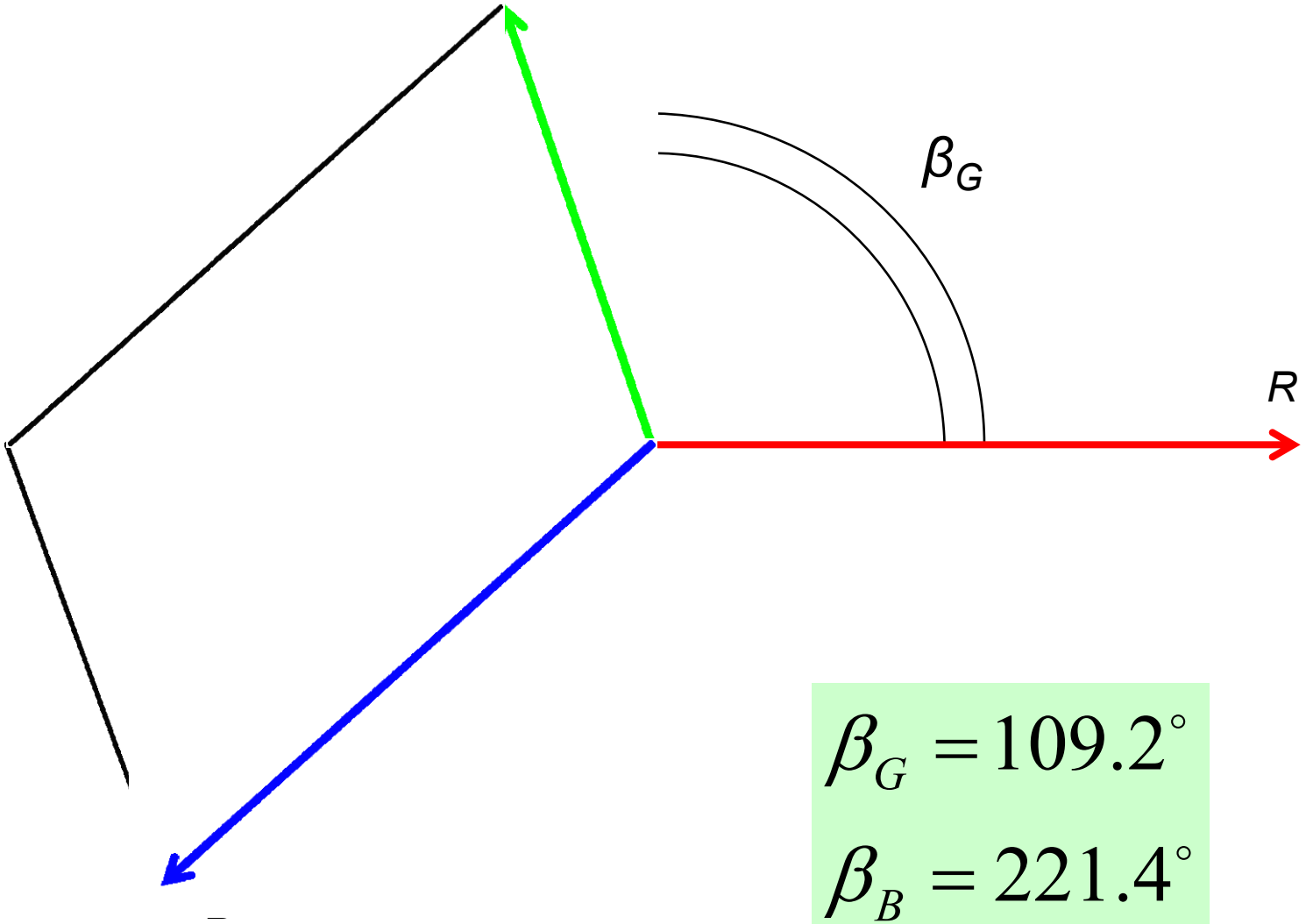
Положение осей *RGB* по цветовому тону. Вид с вершины оси *L*



$$\beta_G = 109.2^\circ$$

$$\beta_B = 221.4^\circ$$

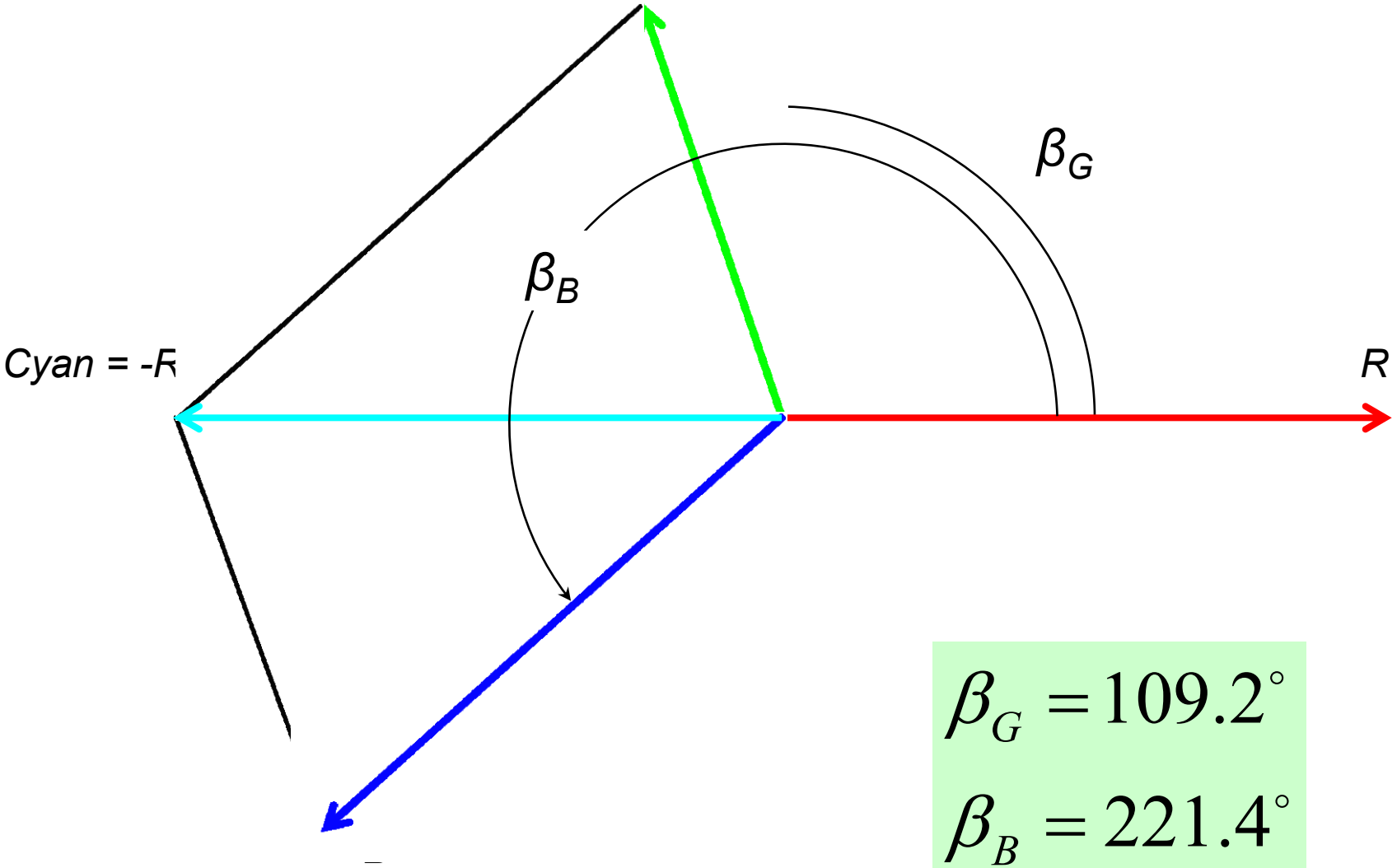
Положение осей *RGB* по цветовому тону. Вид с вершины оси *L*



$$\beta_G = 109.2^\circ$$

$$\beta_B = 221.4^\circ$$

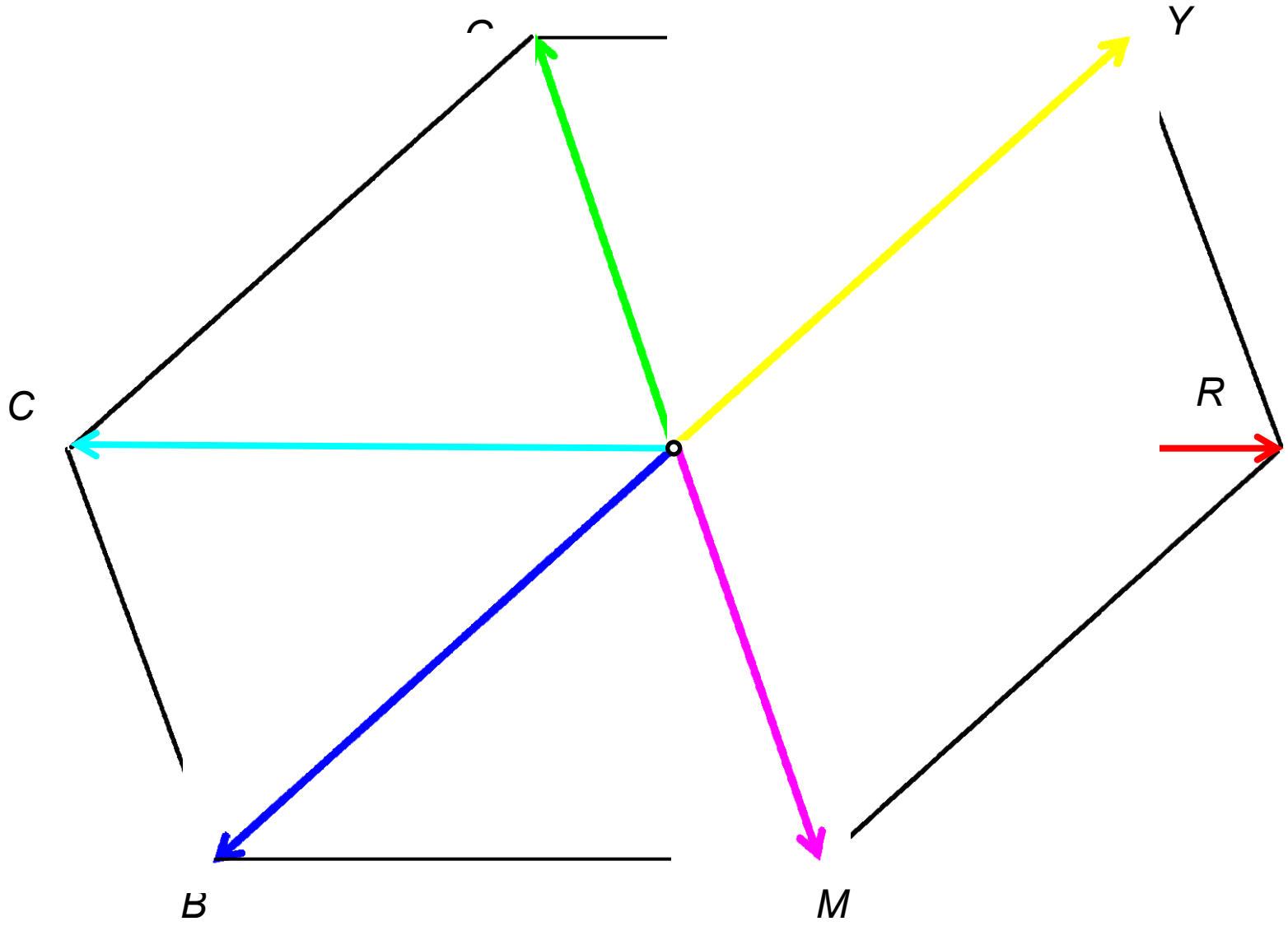
Положение осей *RGB* по цветовому тону. Вид с вершины оси *L*



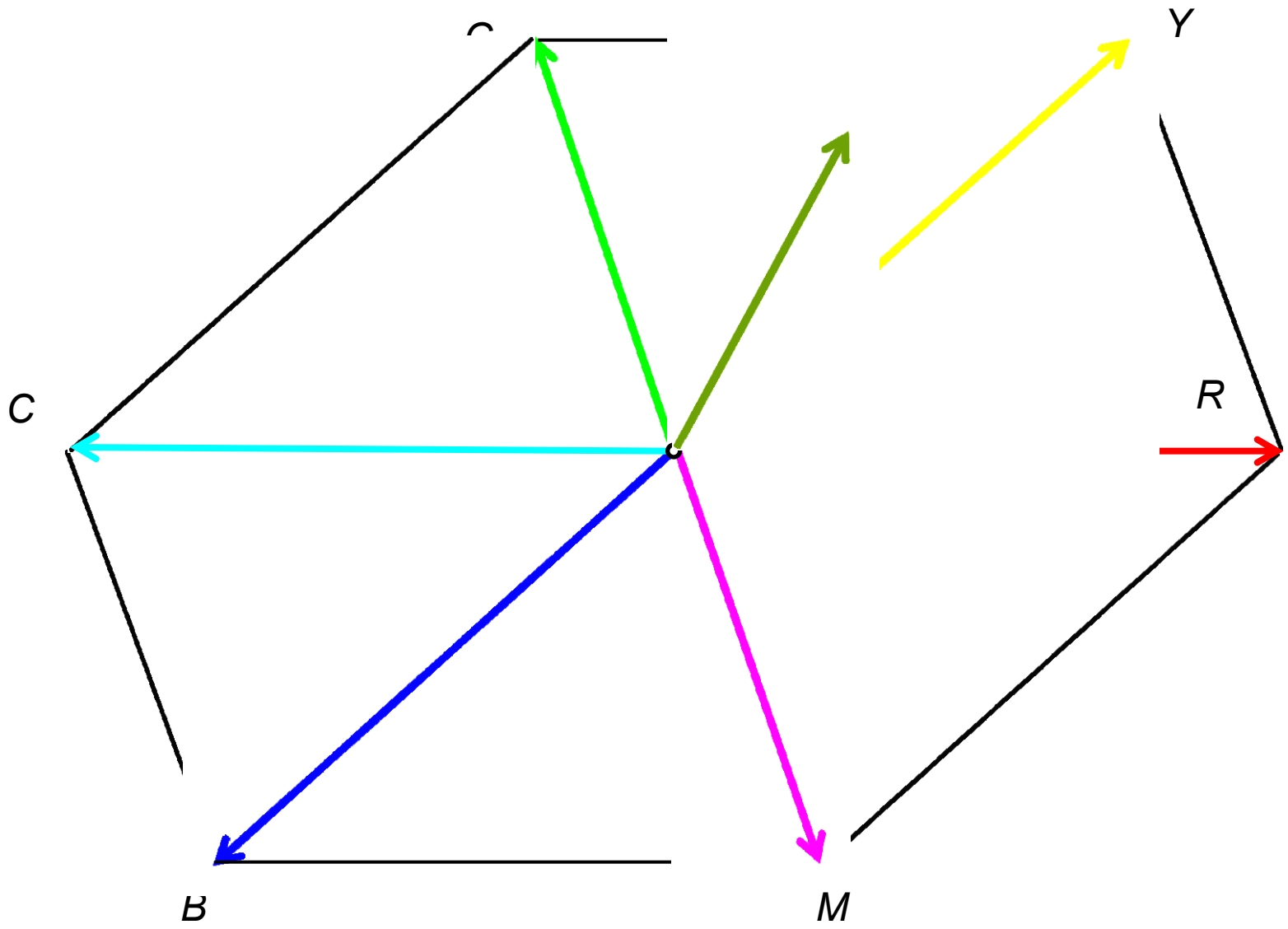
$$\beta_G = 109.2^\circ$$

$$\beta_B = 221.4^\circ$$

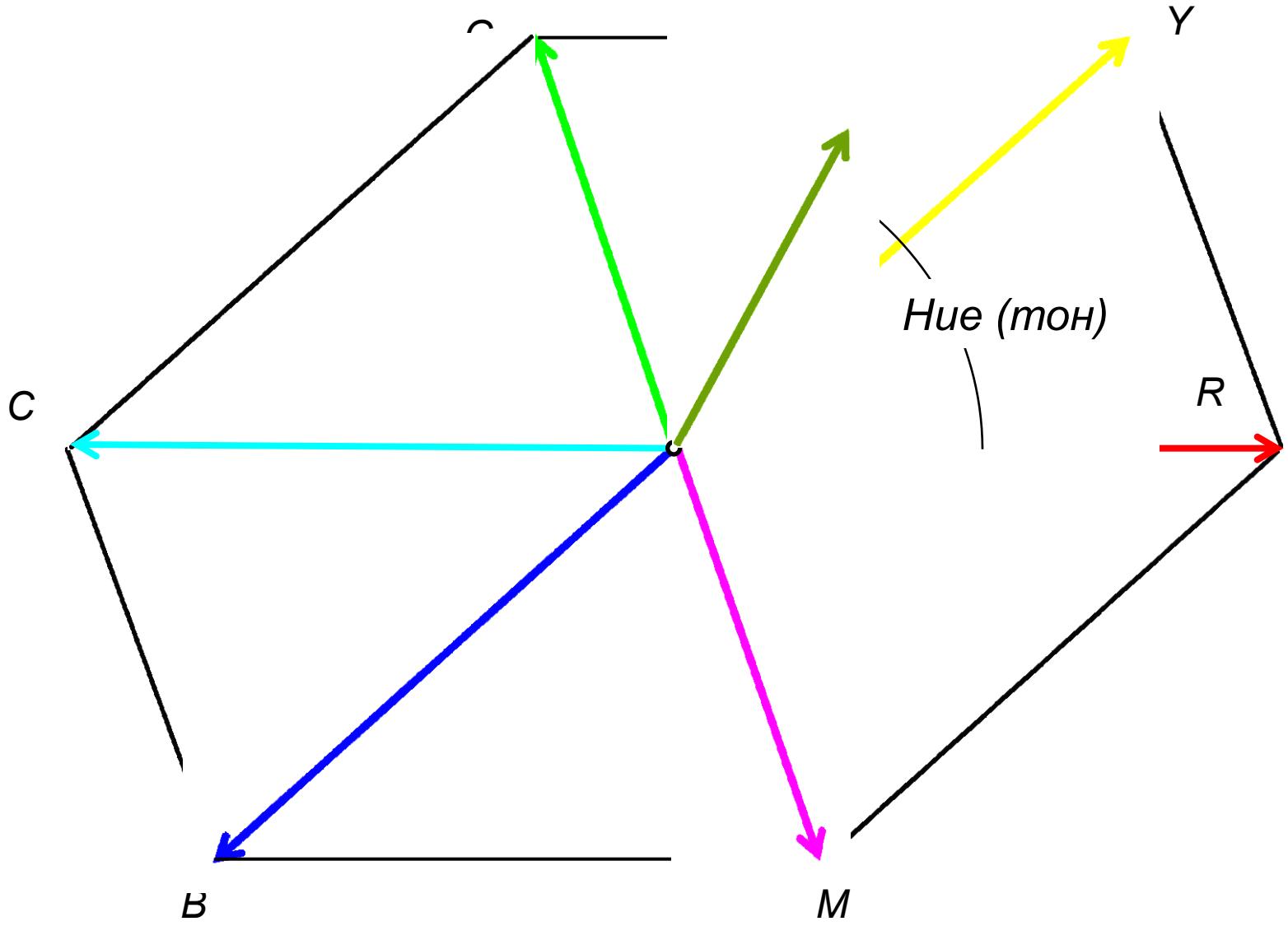
Охват RGB-пространства в проекции на плоскость постоянной светлоты



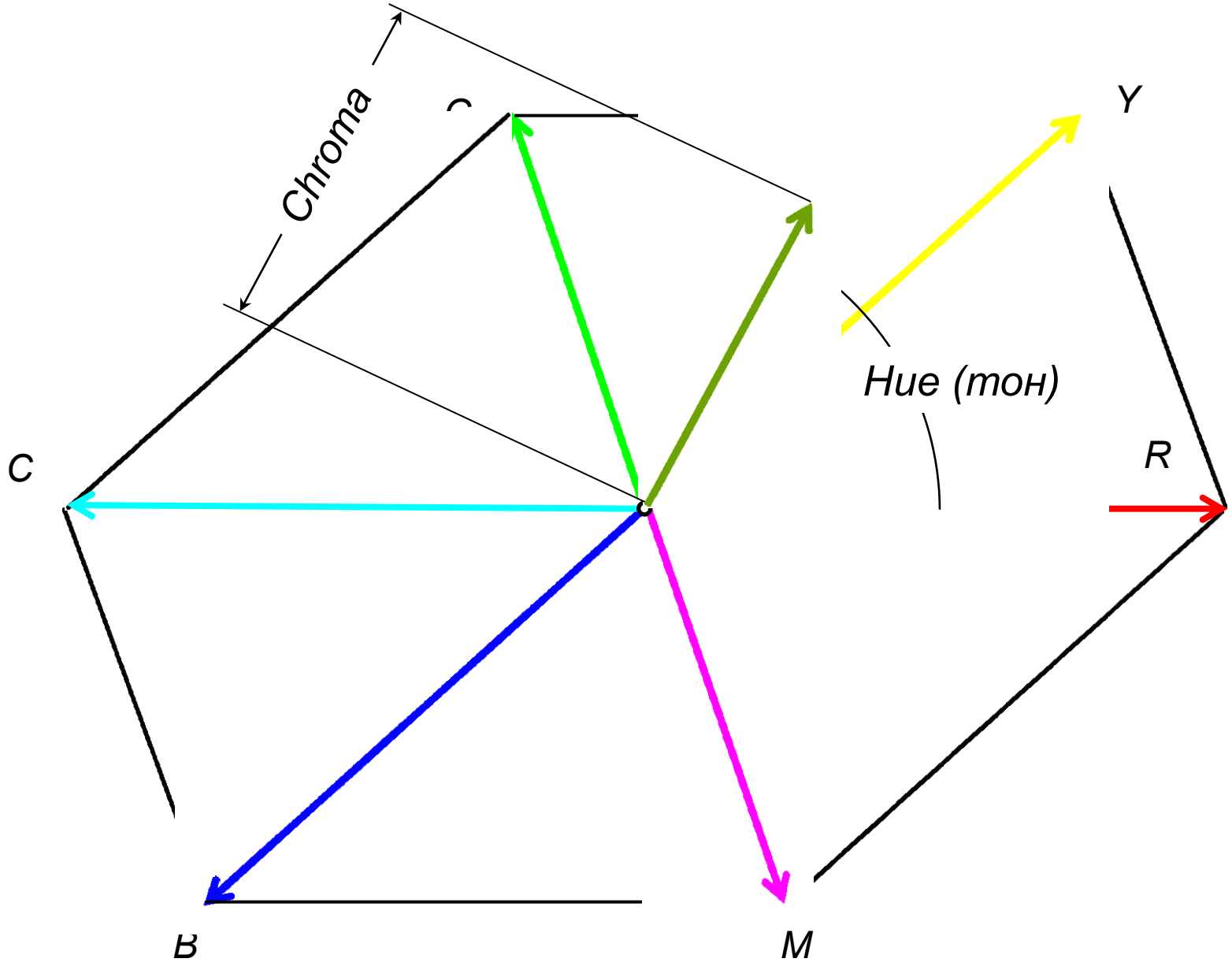
Охват RGB-пространства в проекции на плоскость постоянной светлоты



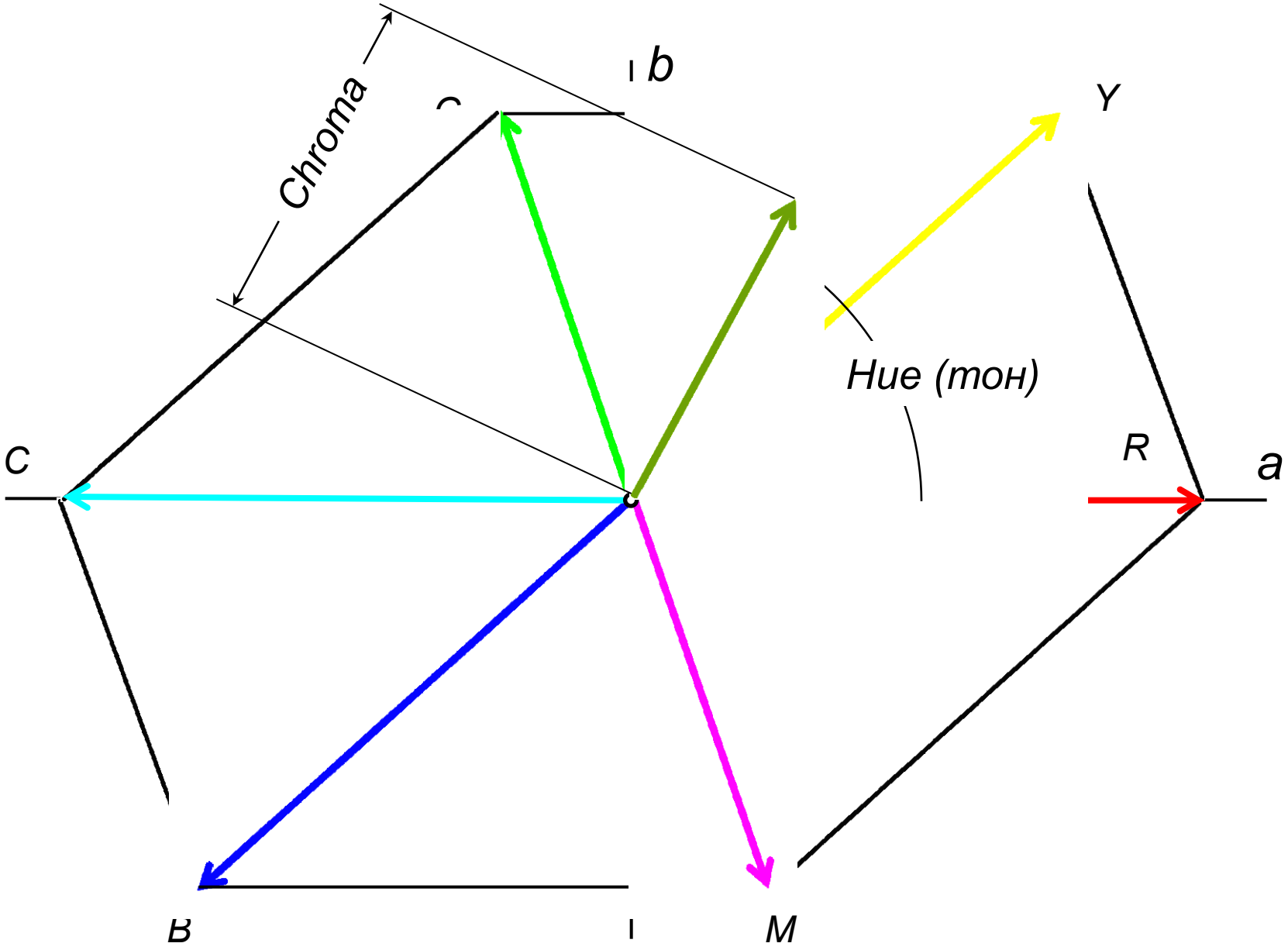
Охват RGB-пространства в проекции на плоскость постоянной светлоты



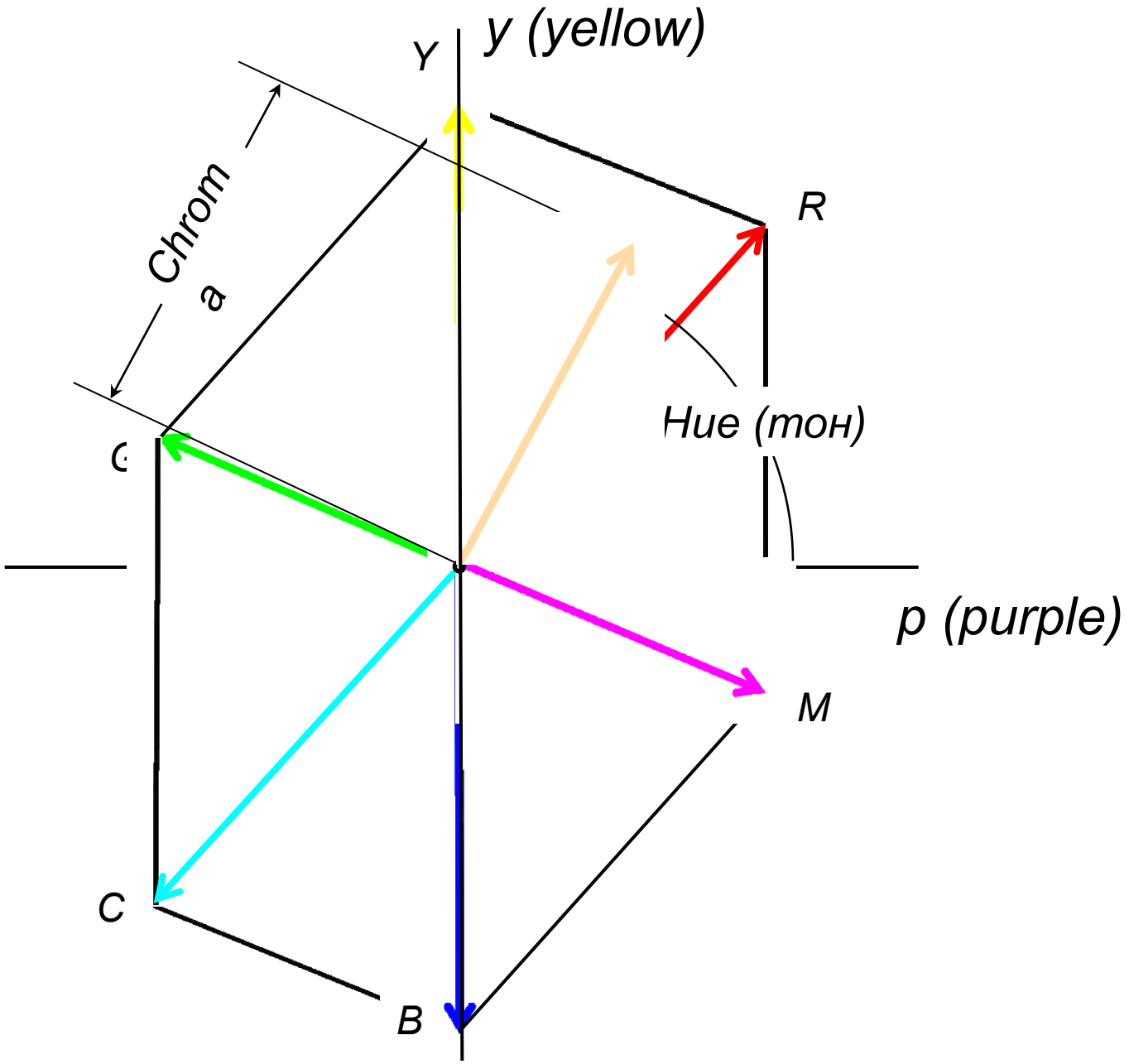
Охват RGB-пространства в проекции на плоскость постоянной светлоты



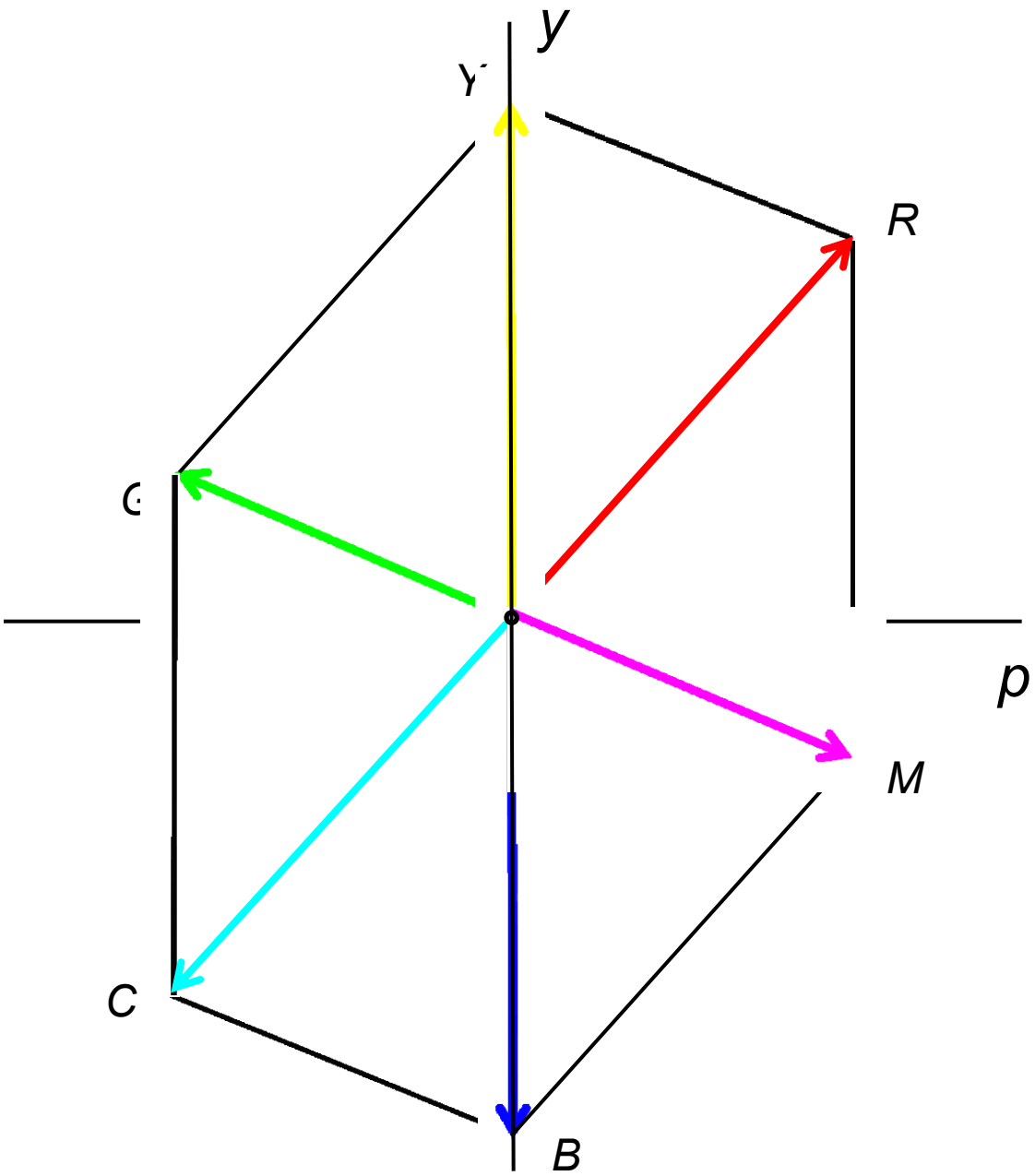
Охват RGB-пространства в проекции на плоскость постоянной светлоты



Цветовая координатная система Lpy



Цветовая координатная система Lpy



$h_R = 48.6^\circ$ $c_R = 0.98$

$h_Y = 90.0^\circ$ $c_Y = 0.99$

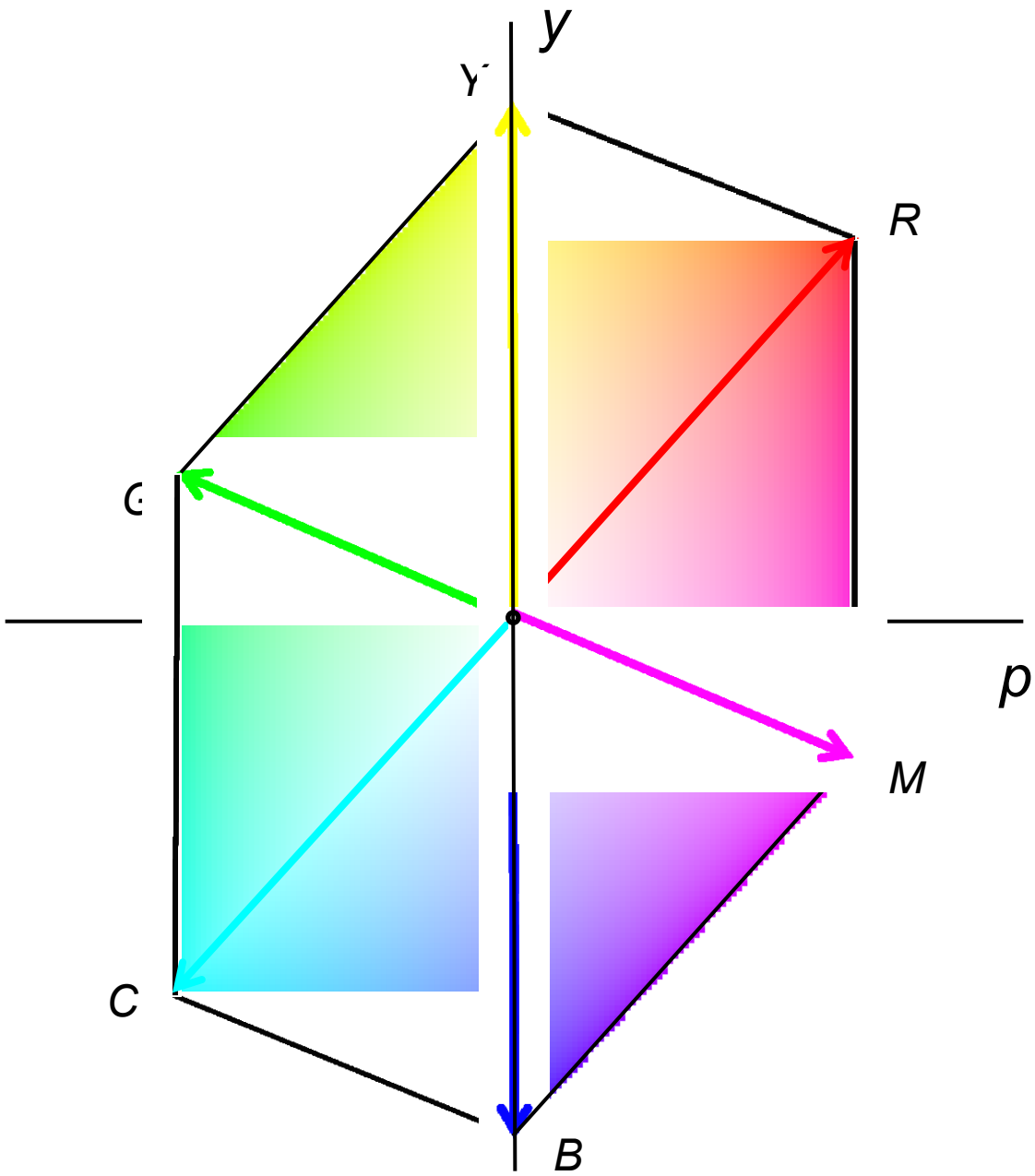
$h_G = 157.7^\circ$ $c_G = 0.70$

$h_C = 228.6^\circ$ $c_C = 0.98$

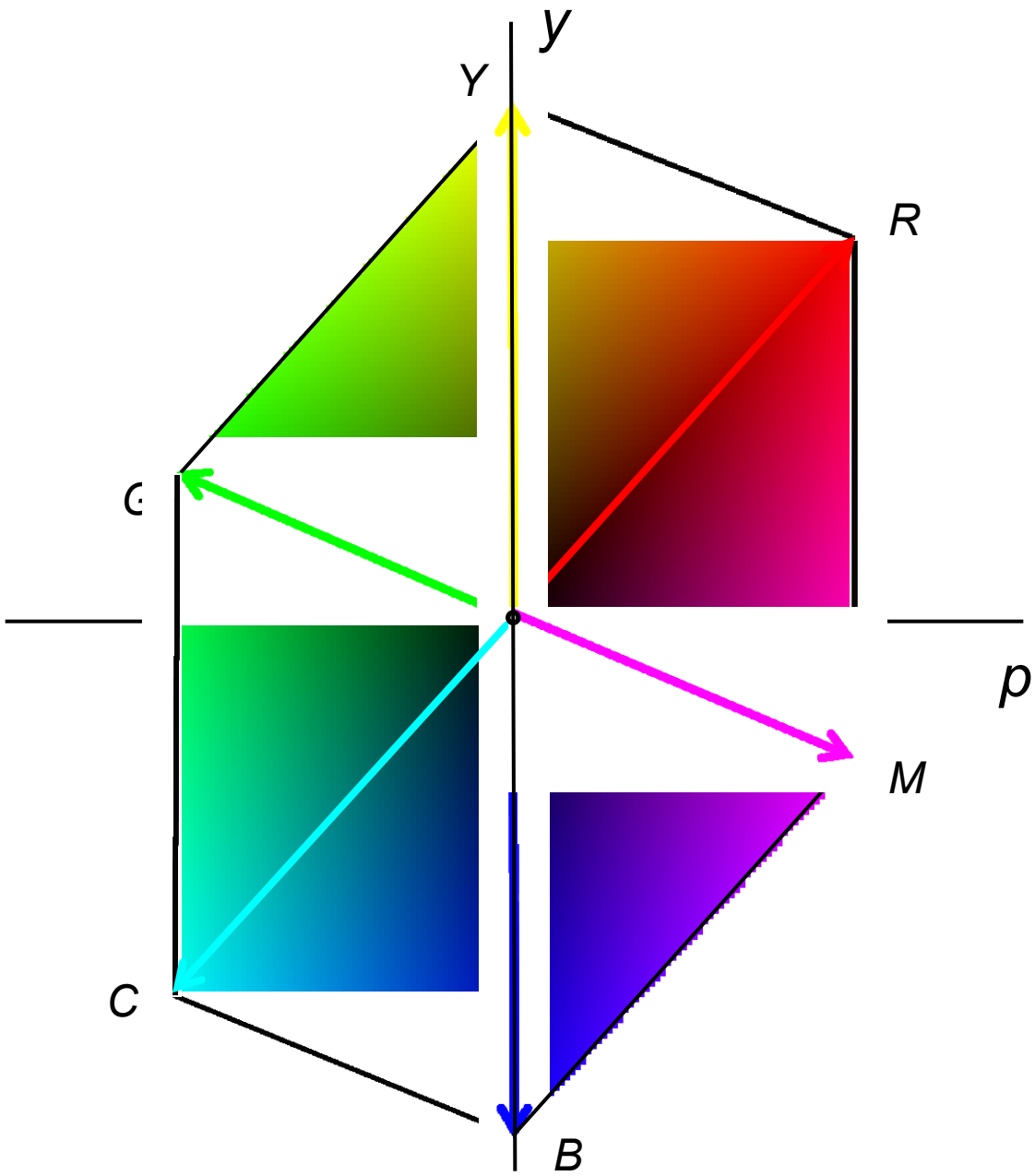
$h_B = 270.0^\circ$ $c_B = 0.99$

$h_M = 337.7^\circ$ $c_M = 0.70$

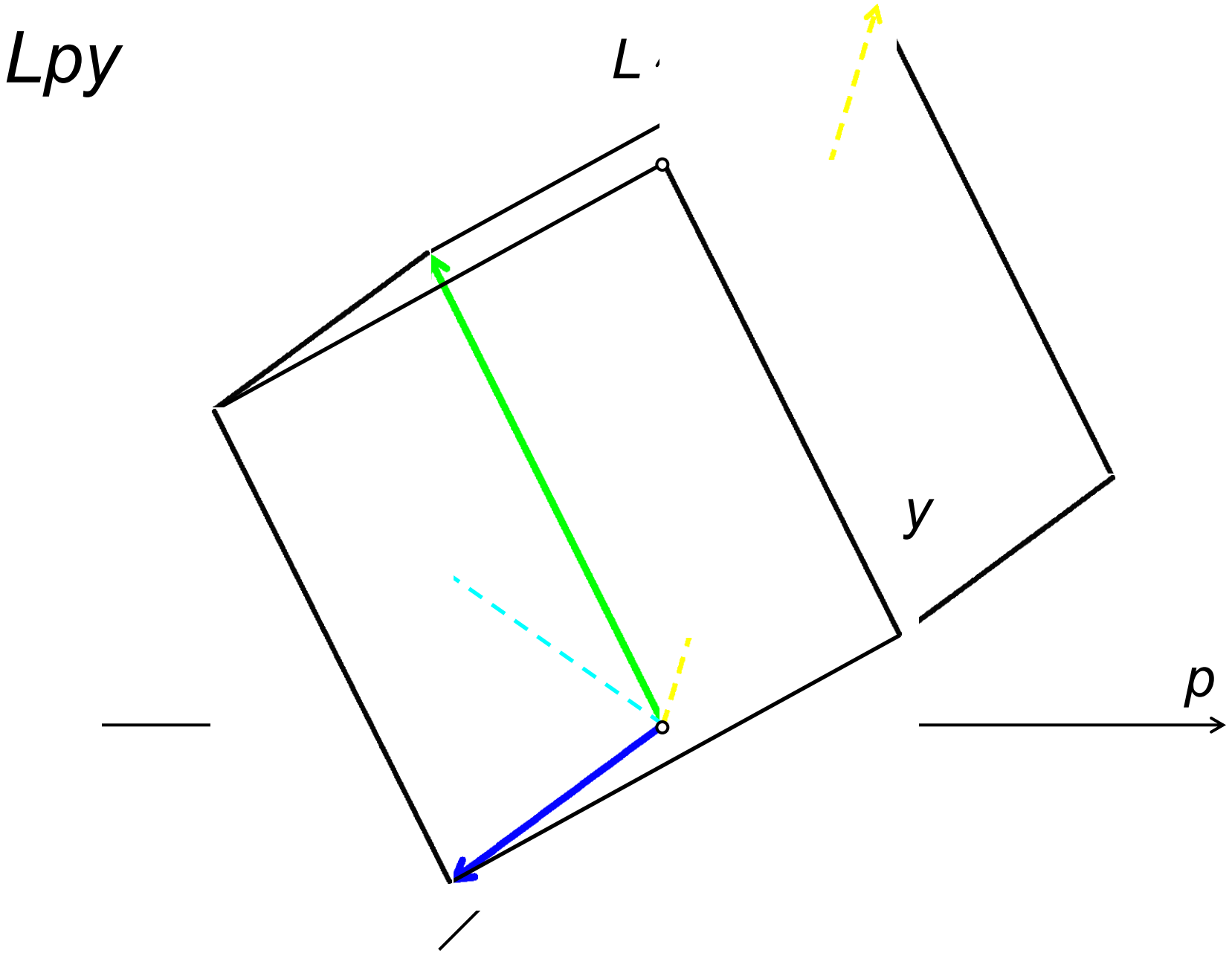
Цветовая координатная система Lpy



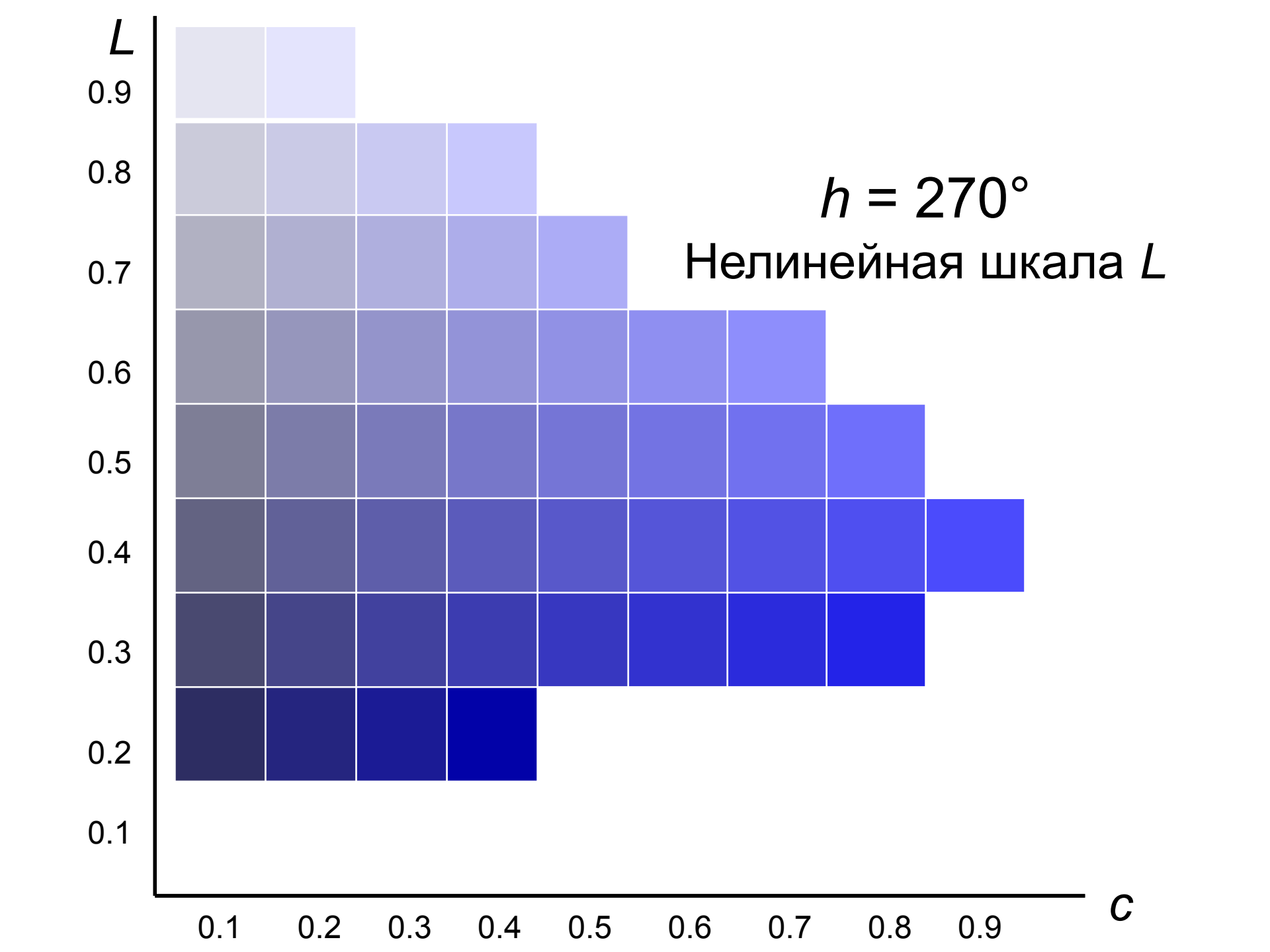
Цветовая координатная система Lpy

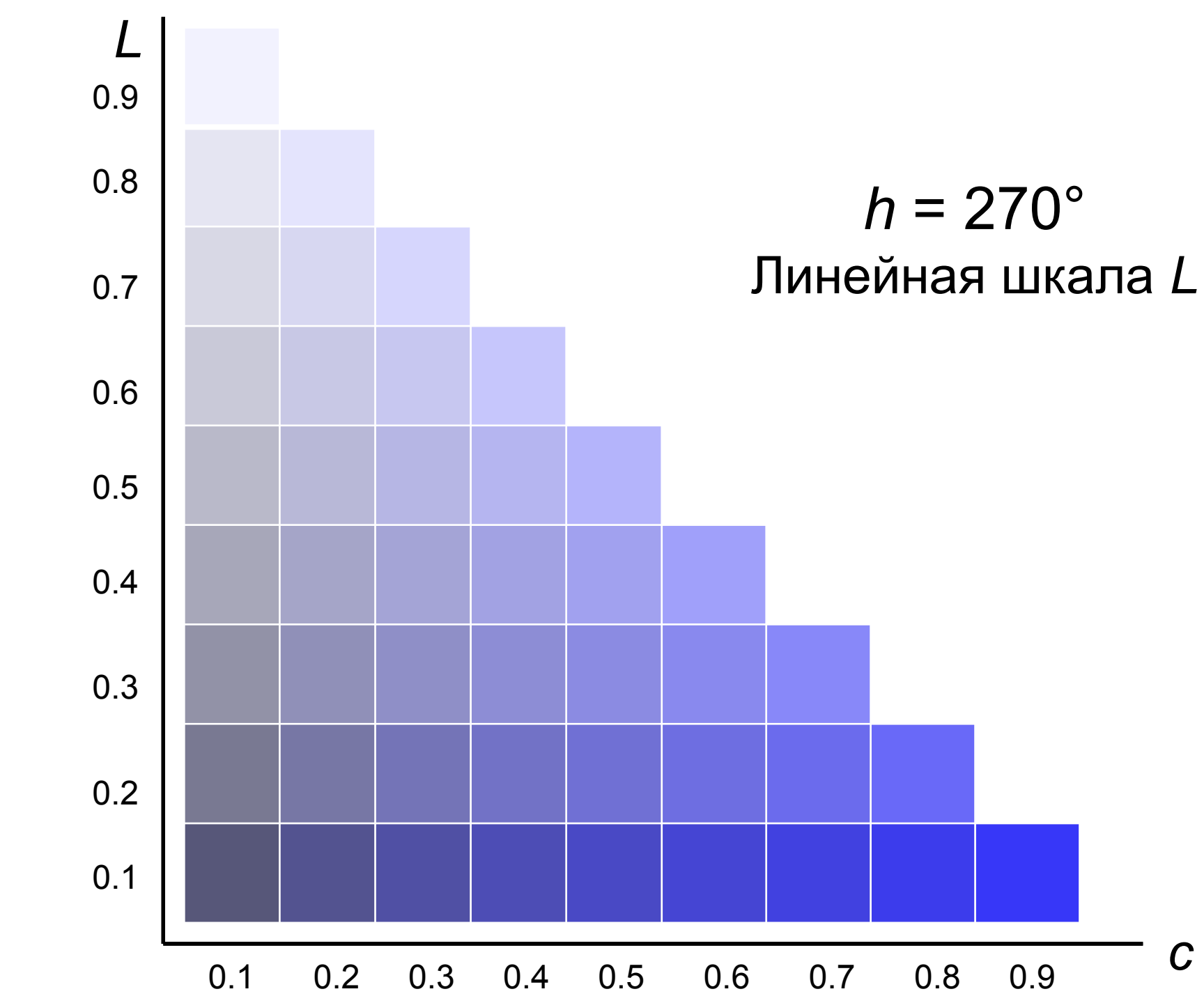


C3C Color Wizard 1.5
25%
Открыть по Ctrl+F
Предупреждения
Коррекция одним кликом
Тон
0,0°
Цветность
0,0
Баланс по среднему
Авто Экспозиция
Авто Насыщенность
Экспозиция
0,00
Вуаль
0,0
Светлота
Контраст
Темное
Яркое
Резкость
Микроконтраст
Тени
Света
Насыщенность
Красочность
Крути колесо на фото [Shift] (↑↓)
Крути с Alt (↑↓)
Сдвиг тона
Пипетки сдвига цветового тона:
Сохранить рецепт как...
Загрузить рецепт...
Donate
Готово
Отмена
Справка (сайт)
X: 2642 Y: 1255
R: 68 G: 23 B: 16
L: 14.9 (1.9) C: 5.1 H: 51.1° S: 72.4% (2.69)
sRGB



Проекция *Ca*


L
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
h = 270°
Нелинейная шкала L
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
c


L
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
h = 270°
Линейная шкала L
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
c

Ортогонализация ЦКС и связанные вопросы

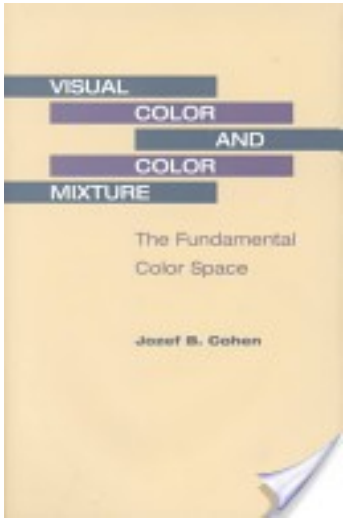
- Цветовые координаты – скалярные произведения:

$$L = \int_{\lambda} E_e(\lambda) \cdot \rho(\lambda) d\lambda \qquad M = \int_{\lambda} E_e(\lambda) \cdot \gamma(\lambda) d\lambda \qquad S = \int_{\lambda} E_e(\lambda) \cdot \beta(\lambda) d\lambda$$

- Процесс Грама — Шмидта

- *Беклемишев Д. В.* Курс аналитической геометрии и линейной алгебры. — М.: Наука.

- *Jozef B. Cohen* Visual Color and Color Mixture: The Fundamental Color Space



Преобразование $sRGB_{лин} \leftrightarrow Lpy$

Преобразование $sRGB_{лин} \leftrightarrow Lpy$

$$\begin{bmatrix} L \\ p \\ y \end{bmatrix} = \begin{bmatrix} 0.2126 & 0.7152 & 0.0722 \\ 0.6468 & -0.6468 & 0.0000 \\ 0.7325 & 0.2649 & -0.9974 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

Преобразование $sRGB_{лин} \leftrightarrow Lpy$

$$\begin{bmatrix} L \\ p \\ y \end{bmatrix} = \begin{bmatrix} 0.2126 & 0.7152 & 0.0722 \\ 0.6468 & -0.6468 & 0.0000 \\ 0.7325 & 0.2649 & -0.9974 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.0000 & 1.1355 & 0.0724 \\ 1.0000 & -0.4107 & 0.0724 \\ 1.0000 & 0.7248 & -0.9302 \end{bmatrix} \begin{bmatrix} L \\ p \\ y \end{bmatrix}$$

Lch(py)
Saturation

Lch(py)
Saturation

$$c = \sqrt{p^2 + y^2}$$

$$\cos h = \frac{p}{c}; \quad \sin h = \frac{y}{c}$$

$$s = \frac{c}{L}$$

Алгоритмы обработки изображений

Представление изображения в программе

- Предпочтительно простое непосредственное представление – массив (матрица) триад (RGB, Lab) или «сплошной» массив.

- Быть осторожным при использовании абстракций, характерных для API, ориентированных на построение оконного интерфейса (TBitmap, TImage, …)

- Входные и выходные данные разделены: `inData → outData`

```
type
   tPixel = record
      R, G, B: byte; {word, single}
    { L, a, b: byte; }
   end;

   tPicture  = array [0..n-1, 0..m-1] of tPixel;
   tLPicture = array [0..n*m-1] of tPixel;
   tSPicture = array [0..3*n*m-1] of byte; {word, single}
```

Гибкое представление для изображений большого размера – массив указателей на строку

```
type
   tPixel = record
      R, G, B: byte; {word, single}
    { L, a, b: byte; }
   end;

   tLine  = array [0..n-1] of tPixel;
   pLine  = ^tLine;
   tPicture  = array [0..m-1] of pLine;
```

Массив указателей на массивы

```
inline float** newArray(int n, int m) {
   float** a = new float*[n];

   for(int i=0; i<n; i++) {
      try{
         a[i] = new float[m];
      }
      catch( bad_alloc &ba) {
         Assert(false, BAD_ALLOC_Error);
      }
   }

   return a;
}

inline void freeArray(float** &a, int n)
{
   for(int i=0; i<n; i++)
      delete [] a[i];
   delete [] a;
   a = NULL;
}
```

```
float** arr = newArray(height, width);
…

for(int j=0; j<m; j++)
   arr[i][j] = y[j];

…
freeArray(arr, height);
```

Представление изображения

Монохромное изображение - функция двух переменных

$$f(x, y)$$

Цветное изображение - вектор - функция двух переменных

$$\mathbf{C} = \mathbf{f}(x, y)$$

$$\mathbf{C} = \begin{pmatrix} R \\ G \\ B \end{pmatrix}$$

Простые поэлементные алгоритмы

```
P  : tPicture;
Pl : tLPicture;
Ps : tSPicture;
```

$$P'[i,j].R = f(P[i,j].R)$$
$$P'[i,j].G = f(P[i,j].G)$$
$$P'[i,j].B = f(P[i,j].B)$$

$$Ps'[i] = f(Ps[i])$$

Примеры:

1. Гамма-коррекция
2. «Кривая»
3. ...

Гамма-коррекция

$$f(x) = M\left(\frac{x}{M}\right)^{\gamma}$$

$$M = 255, 32767, \ldots$$

```
1. P[i,j].R := round(M*Power(p[i,j].R/M, gamma));
   P[i,j].G := round(M*Power(p[i,j].G/M, gamma));
   P[i,j].B := round(M*Power(p[i,j].B/M, gamma));
               i=0..n-1; j=0..m-1

2. PL[i].R := round(M*Power(PL[i].R/M, gamma));
   PL[i].G := round(M*Power(PL[i].G/M, gamma));
   PL[i].B := round(M*Power(PL[i].B/M, gamma));
               i=0..n*m-1;

3. Ps[i] := round(M*Power(Ps[i]/M, gamma)),
   i=0..3*n*m-1,
```

Гамма-коррекция (продолжение)

$$f(x) = M\left(\frac{x}{M}\right)^{\gamma}$$

$$M = 255, 32767, \ldots$$

3. `Ps[i] := round(M*Power(Ps[i]/M, gamma)), i=0..3*n*m-1;`

4. **Правильный способ:**

   a) `T: array[0..M] of 0..M; {array[byte] of byte}`

   b) `T[j] := round(M*Power(j/M, gamma)), j=0..M;`

   c) `Ps[i] := T[Ps[i]], i=0..3*n*m-1;`

Минимальная программа-бенчмарка

```
WriteLn('Create data');
MakeCurve(curve);
MakePicture(LP, 3*1024*1024*mp);
WriteLn('Start');
StartTimer;
for i := 0 to 3*1024*1024*mp-1 do
   LP[i] := curve[LP[i]];
StopTimer;
WriteLn('Stop');
Writeln('Time = ', GetTime:0:1, ' ms');
```

Плагины для Adobe® Photoshop®

- Плагин (plug-in) – подключаемый модуль.
- Виды плагинов для PS: **фильтры,** форматы, экспорт, импорт...
- Плагин-фильтр – это **dll**-библиотека. Расширение файла **.8bf**
- Преимущества плагинов-фильтров:
  - Разработчик освобожден от задач загрузки и сохранения изображения, поддержки различных форматов файлов, в простых случаях – от необходимости организации диалога и т.п.
  - Плагин интегрирован в мощный графический редактор, что облегчает его использование в процессе комплексного редактирования фотографии.
  - Плагин интегрирован в популярный графический редактор, что увеличивает шансы плагина быть востребованным.
- Недостатки: пользователю нужен PS (возможна работа PS-плагинов в других редакторах: Adobe Photoshop Elements, ...).

Дополнительные главы компьютерной графики

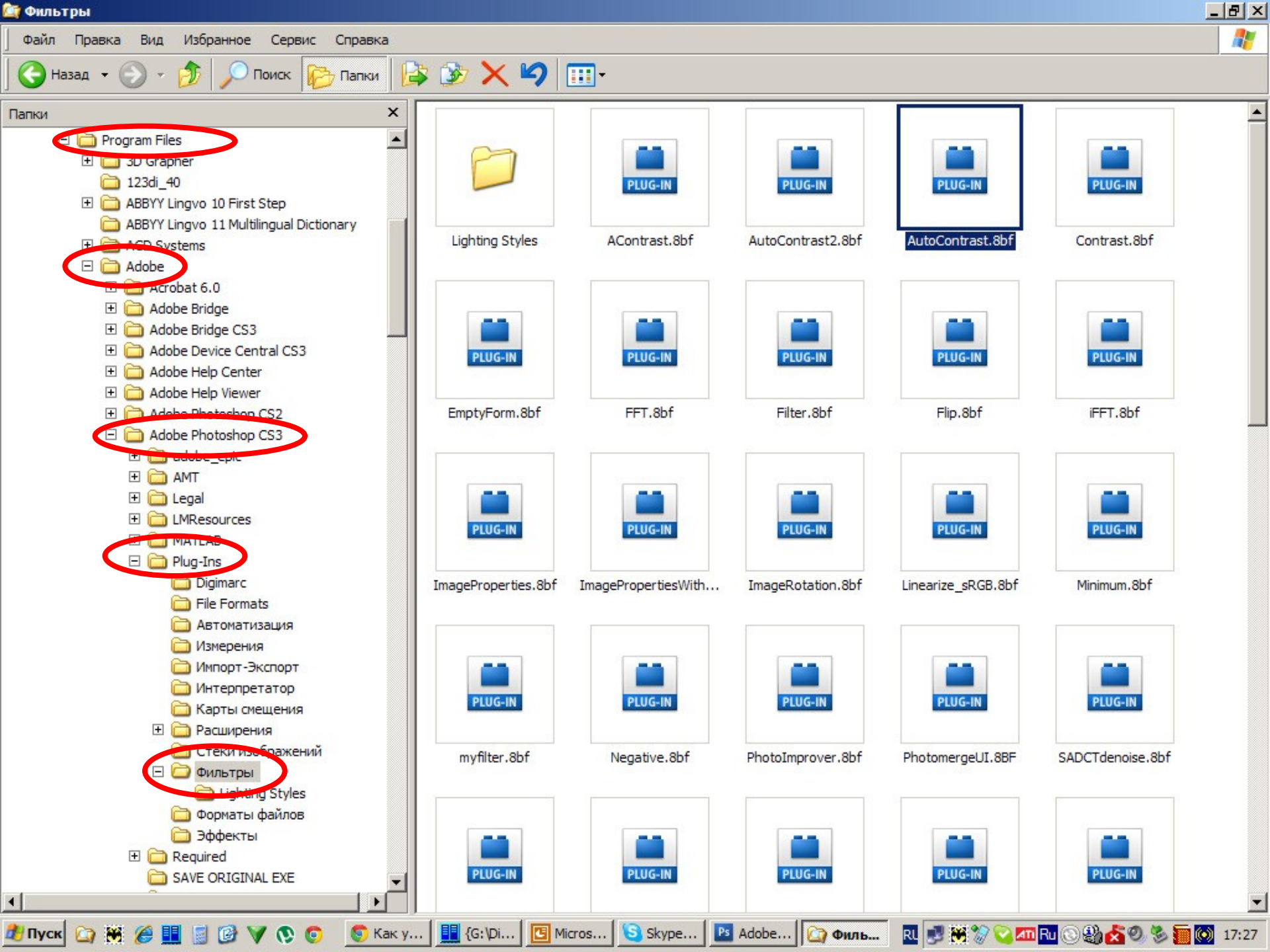
Лекция 5

Copyright © С. Свердлов, 2009-2017
http://www.uni-vologda.ac.ru/~c3c

Как подключить и запустить плагин-фильтр

- http://www.uni-vologda.ac.ru/~c3c/plug-ins/how.htm
- Скопировать файл плагина (.8bf) в папку Adobe Photoshop:

```
C:\Program Files\Adobe\Adobe Photoshop XXX\Plug-Ins\Filters
C:\Program Files\Adobe\Adobe Photoshop XXX\Plug-Ins\Фильтры
```

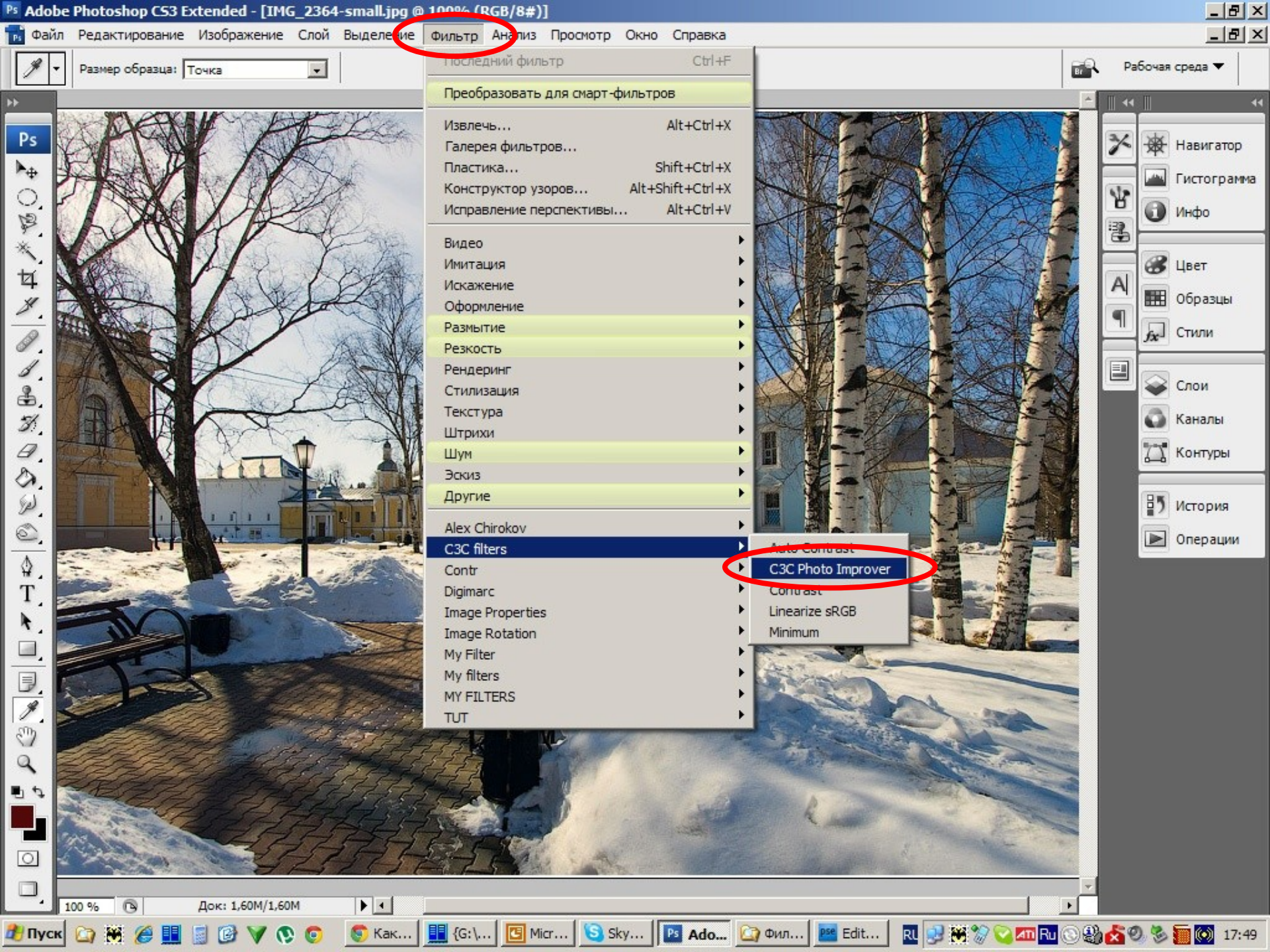

Фильтры
Файл Правка Вид Избранное Сервис Справка
Назад Поиск Папки
Папки
Program Files
3D Grapher
123di_40
ABBYY Lingvo 10 First Step
ABBYY Lingvo 11 Multilingual Dictionary
Adobe
Acrobat 6.0
Adobe Bridge
Adobe Bridge CS3
Adobe Device Central CS3
Adobe Help Center
Adobe Help Viewer
Adobe Photoshop CS3
AMT
Legal
LMResources
Plug-Ins
Digimarc
File Formats
Автоматизация
Измерения
Импорт-Экспорт
Интерпретатор
Карты смещения
Расширения
Фильтры
Форматы файлов
Эффекты
Required
SAVE ORIGINAL EXE
Lighting Styles
AContrast.8bf
AutoContrast2.8bf
AutoContrast.8bf
Contrast.8bf
EmptyForm.8bf
FFT.8bf
Filter.8bf
Flip.8bf
iFFT.8bf
ImageProperties.8bf
ImagePropertiesWith...
ImageRotation.8bf
Linearize_sRGB.8bf
Minimum.8bf
myfilter.8bf
Negative.8bf
PhotoImprover.8bf
PhotomergeUI.8BF
SADCTdenoise.8bf
Пуск
17:27

Как подключить и запустить плагин-фильтр

- http://www.uni-vologda.ac.ru/~c3c/plug-ins/how.htm
- Подключить
  - Скопировать файл плагина (.8bf) в папку Adobe Photoshop:

```
C:\Program Files\Adobe\Adobe Photoshop XXX\Plug-Ins\Filters
C:\Program Files\Adobe\Adobe Photoshop XXX\Plug-Ins\Фильтры
```

- Запустить
  - Загрузить фотографию
  - Меню **Filter (Фильтр)**


Adobe Photoshop CS3 Extended - [IMG_2364-small.jpg @ 100% (RGB/8#)]
Файл
Редактирование
Изображение
Слой
Выделение
Фильтр
Анализ
Просмотр
Окно
Справка
Размер образца:
Точка
Рабочая среда
Последний фильтр Ctrl+F
Преобразовать для смарт-фильтров
Извлечь... Alt+Ctrl+X
Галерея фильтров...
Пластика... Shift+Ctrl+X
Конструктор узоров... Alt+Shift+Ctrl+X
Исправление перспективы... Alt+Ctrl+V
Видео
Имитация
Искажение
Оформление
Размытие
Резкость
Рендеринг
Стилизация
Текстура
Штрихи
Шум
Эскиз
Другие
Alex Chirokov
C3C filters
Contr
Digimarc
Image Properties
Image Rotation
My Filter
My filters
MY FILTERS
TUT
C3C Photo Improver
Contrast
Linearize sRGB
Minimum
Навигатор
Гистограмма
Инфо
Цвет
Образцы
Стили
Слои
Каналы
Контуры
История
Операции
100 %
Док: 1,60M/1,60M
Пуск
17:49

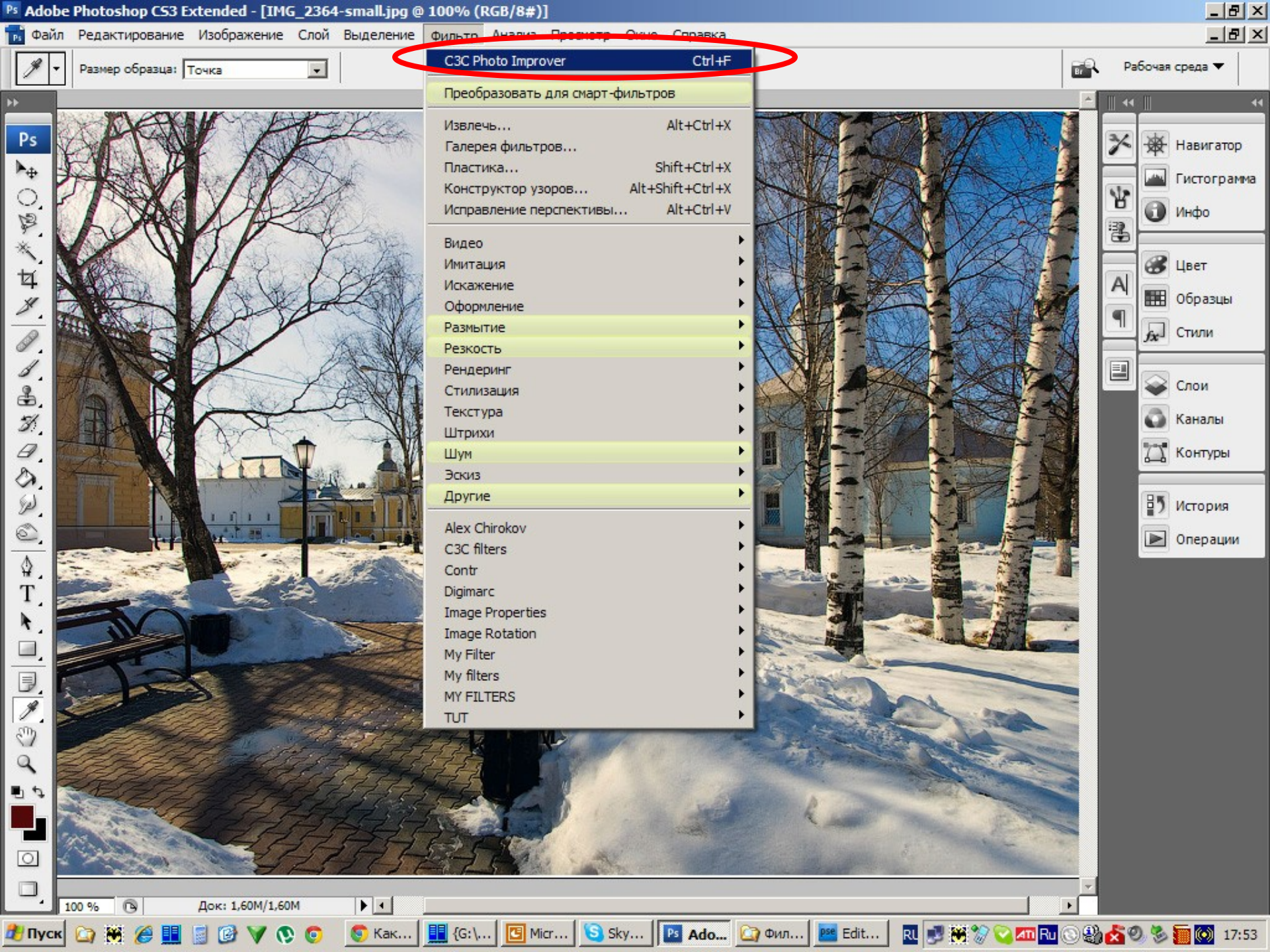
Как подключить и запустить плагин-фильтр

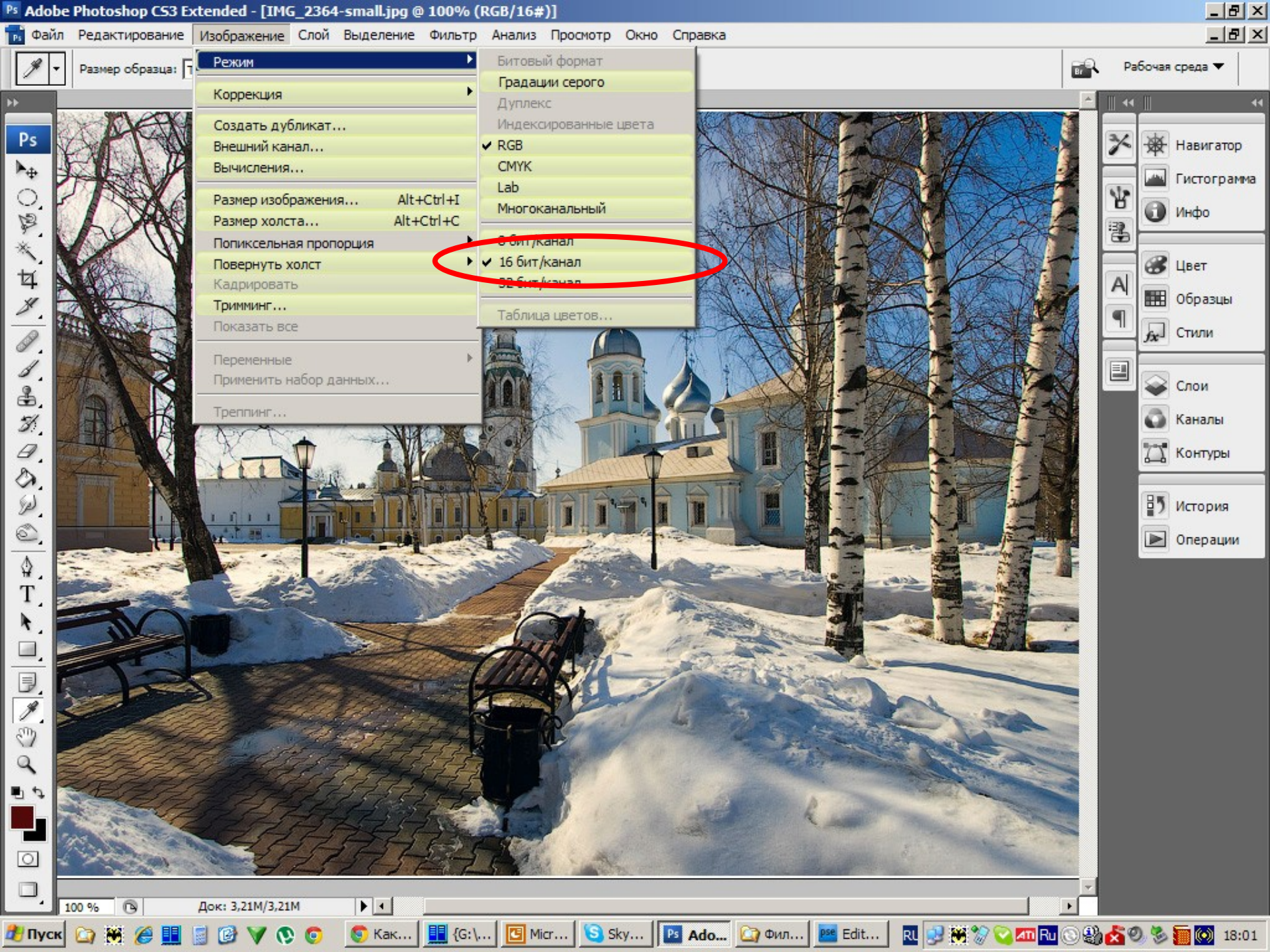
- http://www.uni-vologda.ac.ru/~c3c/plug-ins/how.htm
- Подключить
  - Скопировать файл плагина (.8bf) в папку Adobe Photoshop:

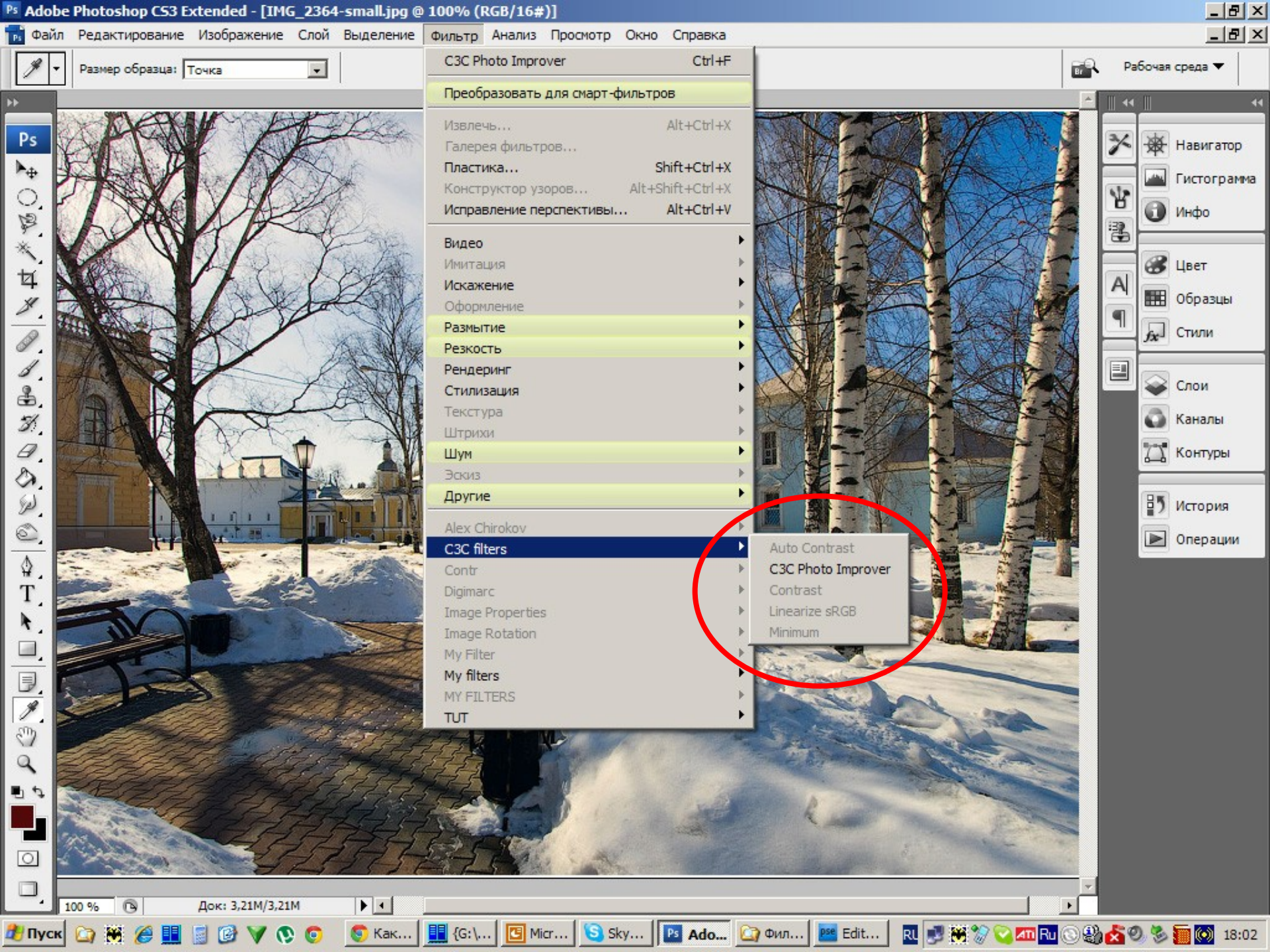
`C:\Program Files\Adobe\Adobe Photoshop XX\Plug-Ins\Filters`

`(C:\Program Files\Adobe\Adobe Photoshop XX\Plug-Ins\Фильтры)`

- Запустить
  - Загрузить фотографию
  - Меню **Filter (Фильтр)**
- Повторный запуск с теми же параметрами
  - Меню **Filter/Last Filter (Ctrl-F)**


Adobe Photoshop CS3 Extended - [IMG_2364-small.jpg @ 100% (RGB/8#)]
Файл
Редактирование
Изображение
Слой
Выделение
Фильтр
Анализ
Просмотр
Окно
Справка
Размер образца:
Точка
Рабочая среда
C3C Photo Improver Ctrl+F
Преобразовать для смарт-фильтров
Извлечь... Alt+Ctrl+X
Галерея фильтров...
Пластика... Shift+Ctrl+X
Конструктор узоров... Alt+Shift+Ctrl+X
Исправление перспективы... Alt+Ctrl+V
Видео
Имитация
Искажение
Оформление
Размытие
Резкость
Рендеринг
Стилизация
Текстура
Штрихи
Шум
Эскиз
Другие
Alex Chirokov
C3C filters
Contr
Digimarc
Image Properties
Image Rotation
My Filter
My filters
MY FILTERS
TUT
Навигатор
Гистограмма
Инфо
Цвет
Образцы
Стили
Слои
Каналы
Контуры
История
Операции
100 %
Док: 1,60M/1,60M
Пуск
17:53


Adobe Photoshop CS3 Extended - [IMG_2364-small.jpg @ 100% (RGB/16#)]
Файл
Редактирование
Изображение
Слой
Выделение
Фильтр
Анализ
Просмотр
Окно
Справка
Размер образца:
Рабочая среда
Режим
Коррекция
Создать дубликат...
Внешний канал...
Вычисления...
Размер изображения... Alt+Ctrl+I
Размер холста... Alt+Ctrl+C
Попиксельная пропорция
Повернуть холст
Кадрировать
Тримминг...
Показать все
Переменные
Применить набор данных...
Треппинг...
Битовый формат
Градации серого
Дуплекс
Индексированные цвета
RGB
CMYK
Lab
Многоканальный
8 бит/канал
16 бит/канал
32 бит/канал
Таблица цветов...
Навигатор
Гистограмма
Инфо
Цвет
Образцы
Стили
Слои
Каналы
Контуры
История
Операции
100 %
Док: 3,21M/3,21M
Пуск
18:01


Adobe Photoshop CS3 Extended - [IMG_2364-small.jpg @ 100% (RGB/16#)]
Файл Редактирование Изображение Слой Выделение Фильтр Анализ Просмотр Окно Справка
Размер образца: Точка
Рабочая среда
C3C Photo Improver Ctrl+F
Преобразовать для смарт-фильтров
Извлечь... Alt+Ctrl+X
Галерея фильтров...
Пластика... Shift+Ctrl+X
Конструктор узоров... Alt+Shift+Ctrl+X
Исправление перспективы... Alt+Ctrl+V
Видео
Имитация
Искажение
Оформление
Размытие
Резкость
Рендеринг
Стилизация
Текстура
Штрихи
Шум
Эскиз
Другие
Alex Chirokov
C3C filters
Contr
Digimarc
Image Properties
Image Rotation
My Filter
My filters
MY FILTERS
TUT
Auto Contrast
C3C Photo Improver
Contrast
Linearize sRGB
Minimum
Навигатор
Гистограмма
Инфо
Цвет
Образцы
Стили
Слои
Каналы
Контуры
История
Операции
100 %
Док: 3,21M/3,21M
Пуск
18:02

Программирование плагинов

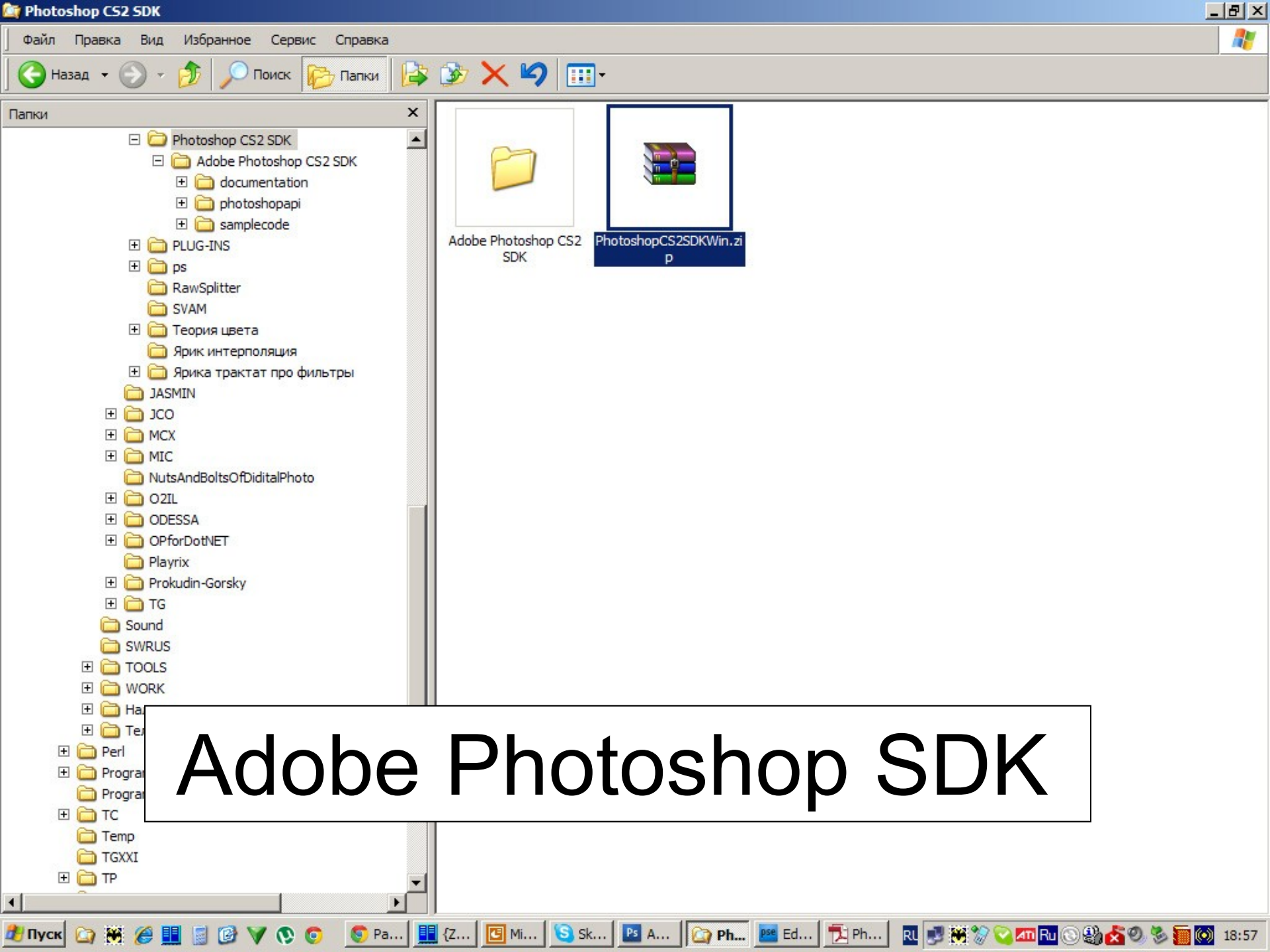
Разработка плагинов для Adobe Photoshop

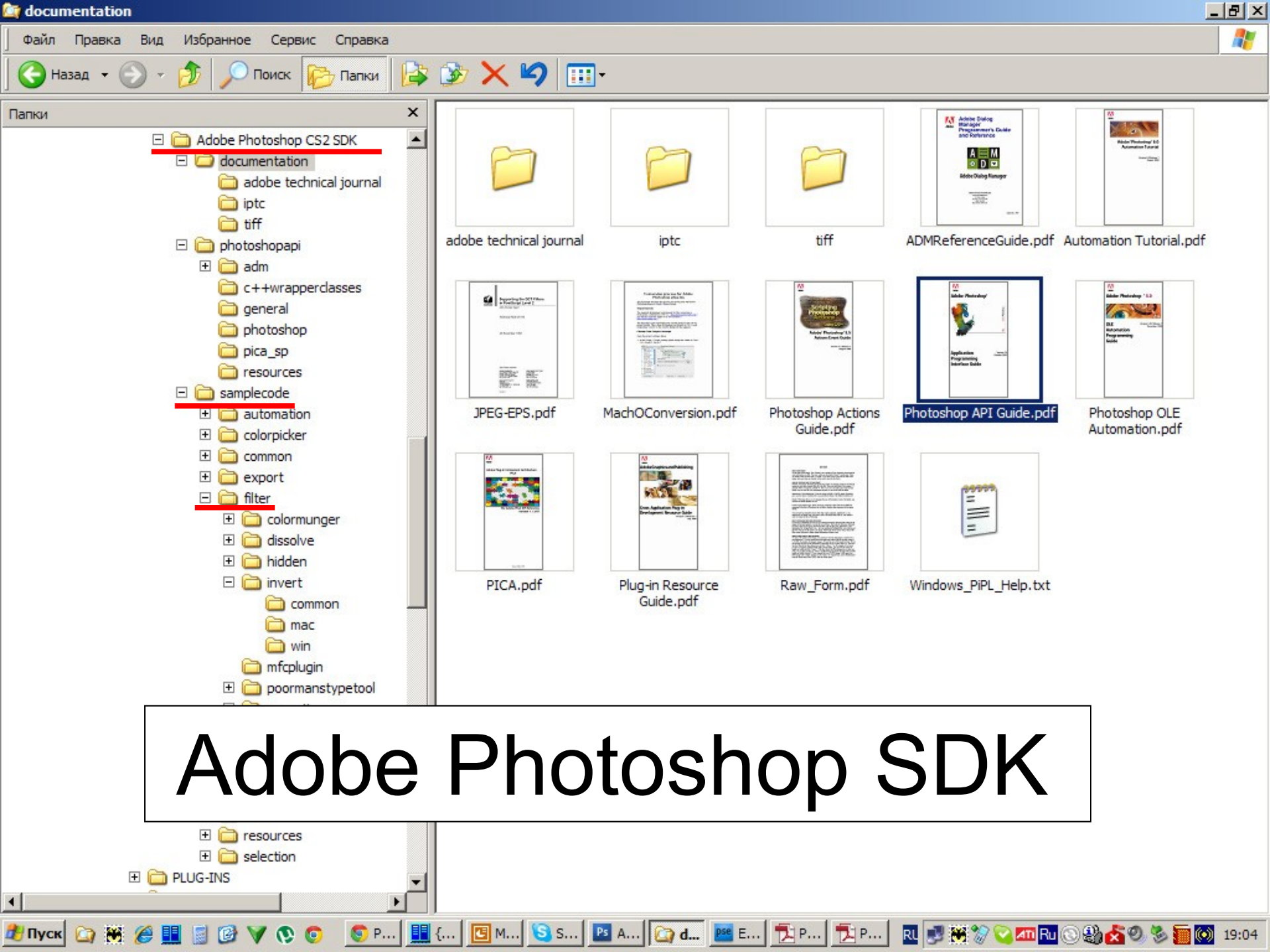
Автор: Ярослав Музыкантов muzykantov@mail.ru

Содержание

1. Введение
2. Общая структура плагина
3. Порядок взаимодействия редактора и фильтрационного плагина
4. Несложный пример плагина
    1. Создаем ресурсы
    2. Пишем плагин
    3. Компилируем и запускаем плагин
5. Работа с памятью
6. Функции обратного вызова (callback-функции)
    1. Непосредственные функции (direct callbacks)
    2. Пакеты функций (callback suites)
    3. Пакет функций буфера (Buffer suite)
    4. Пакет функций портов каналов (Cannel Ports suite)
7. Запись FilterRecord
8. Ресурсы PiPL
    1. Основные свойства
    2. Описание кода
    3. Свойства фильтров
9. Пример полноценного плагина
10. Средства разработки
11. Заключение
12. Ссылки

© Ярослав Музыкантов muzykantov@mail.ru
При использовании материалов обязательна ссылка на сайт
www.uni-vologda.ac.ru/students/pm03/mya


Photoshop CS2 SDK
Файл Правка Вид Избранное Сервис Справка
Назад Поиск Папки
Папки
Photoshop CS2 SDK
Adobe Photoshop CS2 SDK
documentation
photoshopapi
samplecode
PLUG-INS
ps
RawSplitter
SVAM
Теория цвета
Ярик интерполяция
Ярика трактат про фильтры
JASMIN
JCO
MCX
MIC
NutsAndBoltsOfDiditalPhoto
O2IL
ODESSA
OPforDotNET
Playrix
Prokudin-Gorsky
TG
Sound
SWRUS
TOOLS
WORK
Perl
TC
Temp
TGXXI
TP
Adobe Photoshop CS2 SDK
PhotoshopCS2SDKWin.zip
Adobe Photoshop SDK
Пуск
18:57


documentation
Файл Правка Вид Избранное Сервис Справка
Назад Поиск Папки
Папки
Adobe Photoshop CS2 SDK
documentation
adobe technical journal
iptc
tiff
photoshopapi
adm
c++wrapperclasses
general
photoshop
pica_sp
resources
samplecode
automation
colorpicker
common
export
filter
colormunger
dissolve
hidden
invert
common
mac
win
mfcplugin
poormanstypetool
resources
selection
PLUG-INS
adobe technical journal
iptc
tiff
ADMReferenceGuide.pdf
Automation Tutorial.pdf
JPEG-EPS.pdf
MachOConversion.pdf
Photoshop Actions Guide.pdf
Photoshop API Guide.pdf
Photoshop OLE Automation.pdf
PICA.pdf
Plug-in Resource Guide.pdf
Raw_Form.pdf
Windows_PiPL_Help.txt
Adobe Photoshop SDK
Пуск
19:04

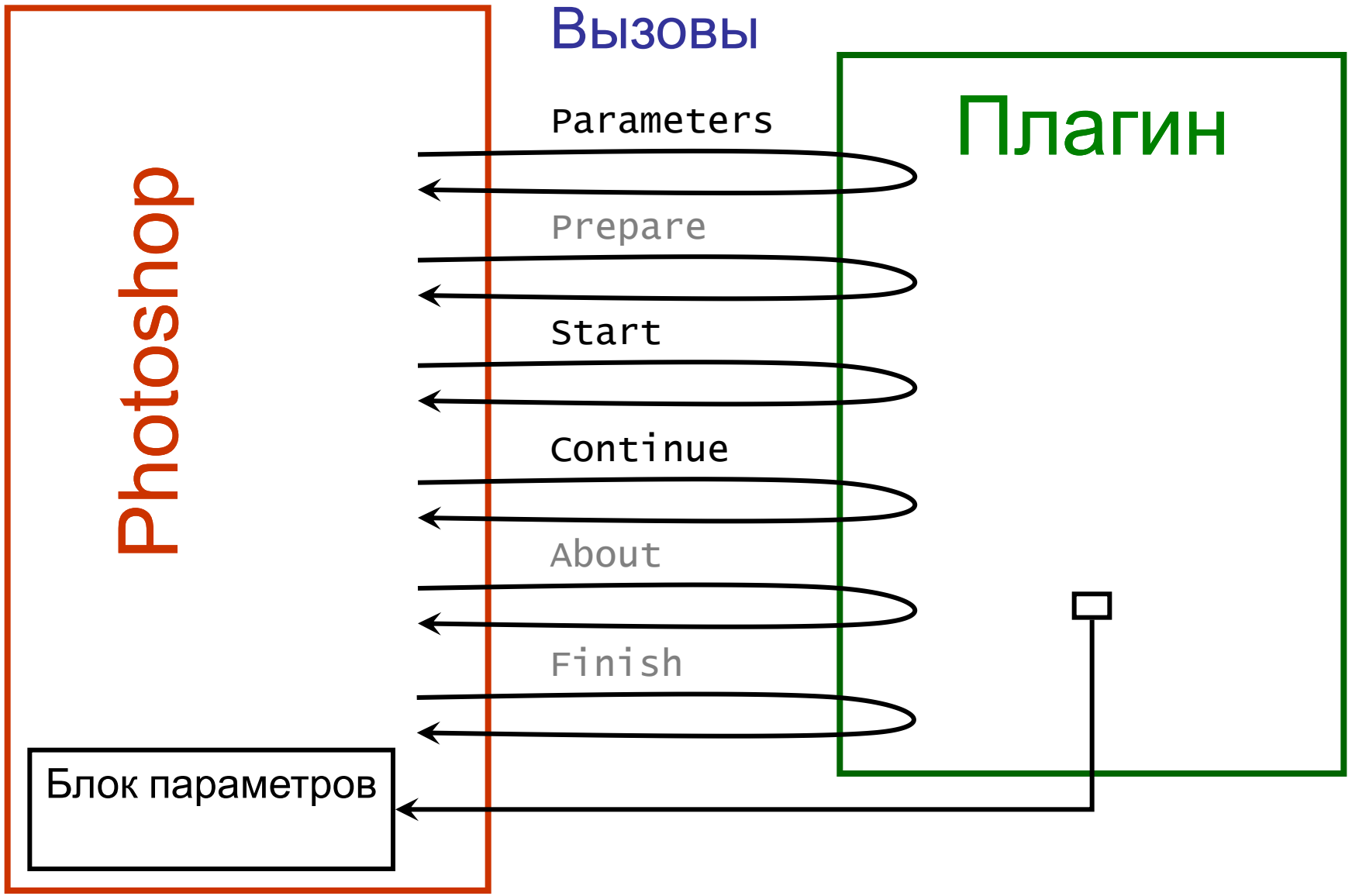
Adobe

Adobe Photoshop®

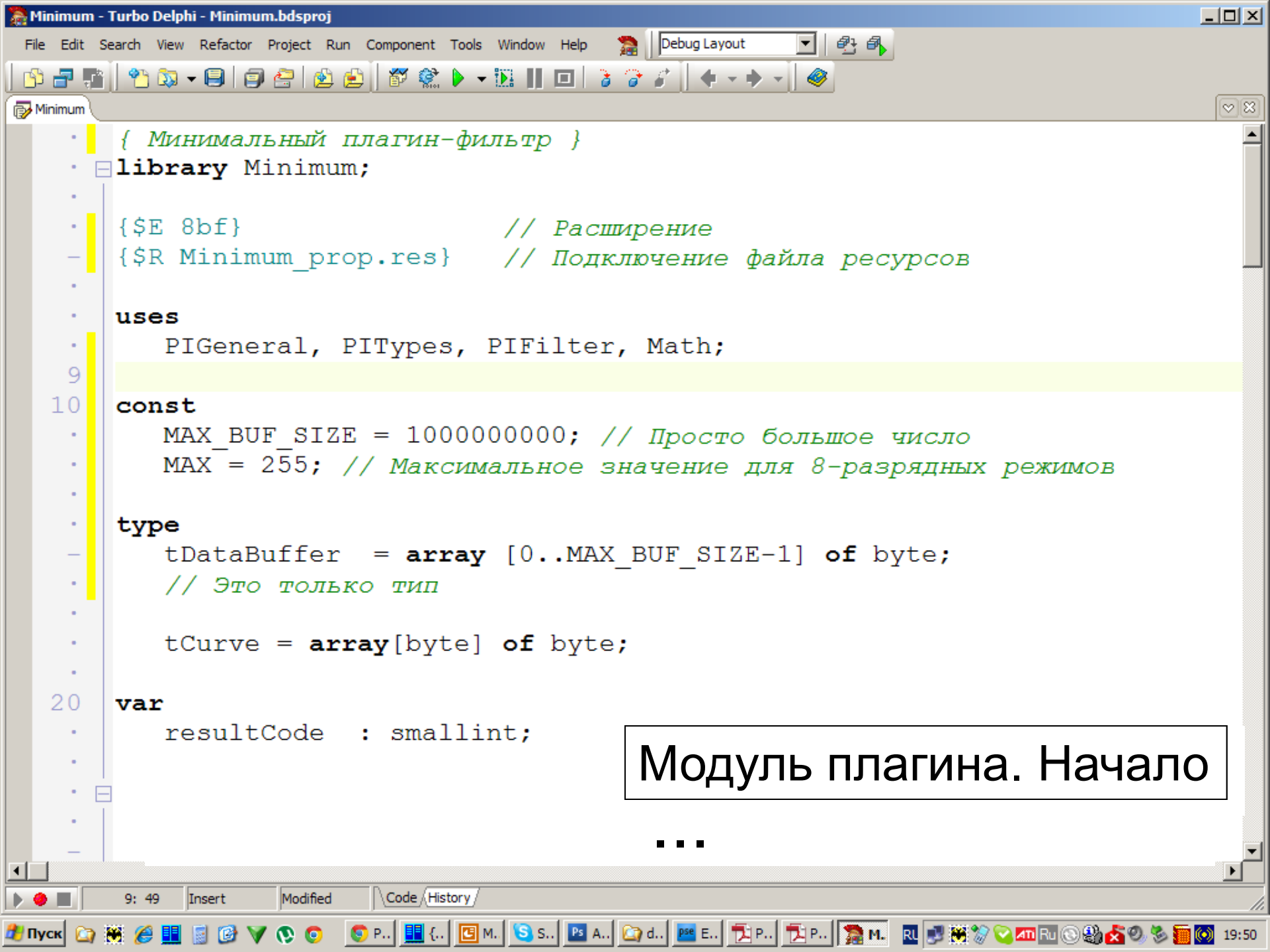
Application Programming Interface Guide

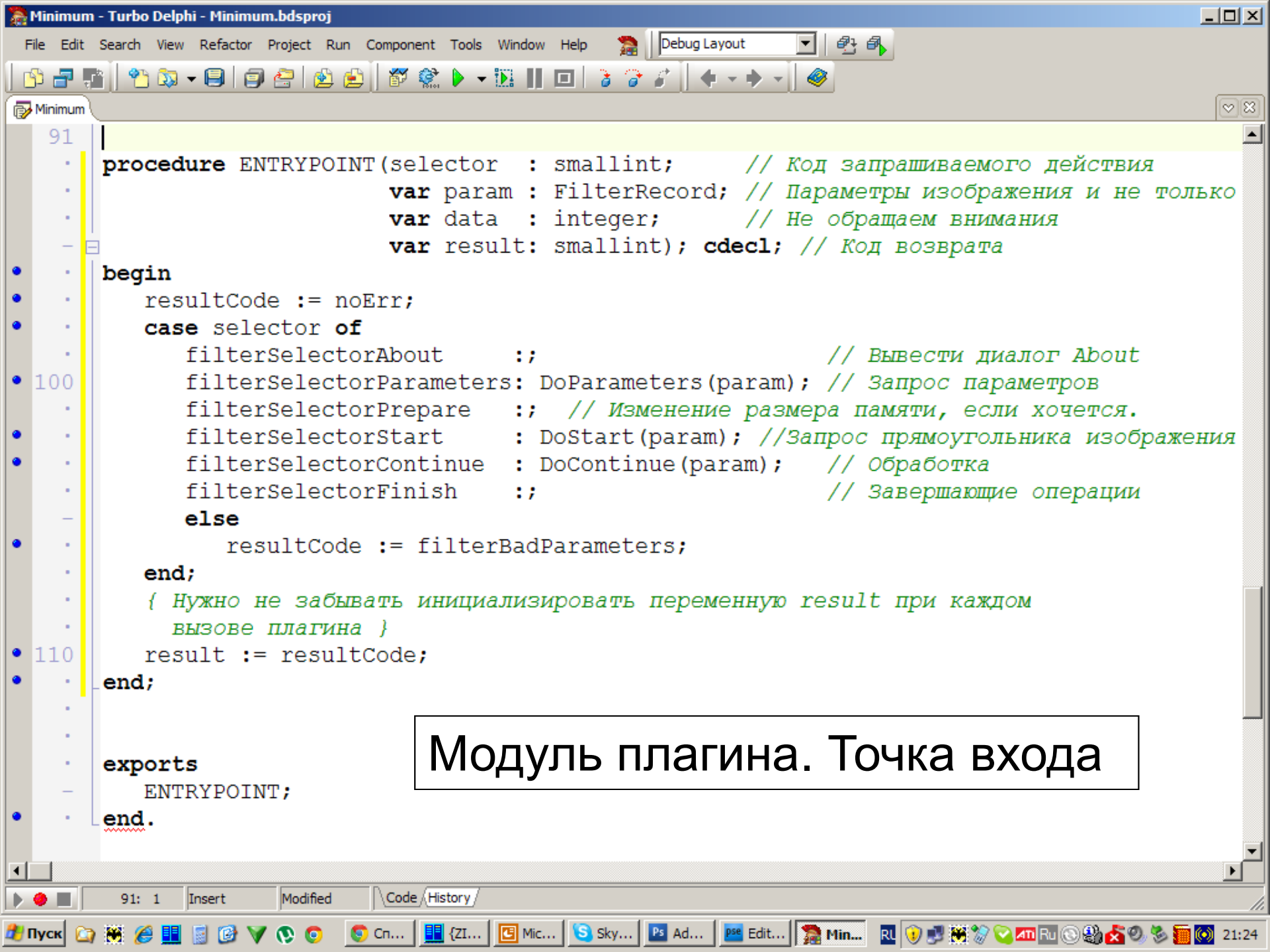
Version CS
October 2003

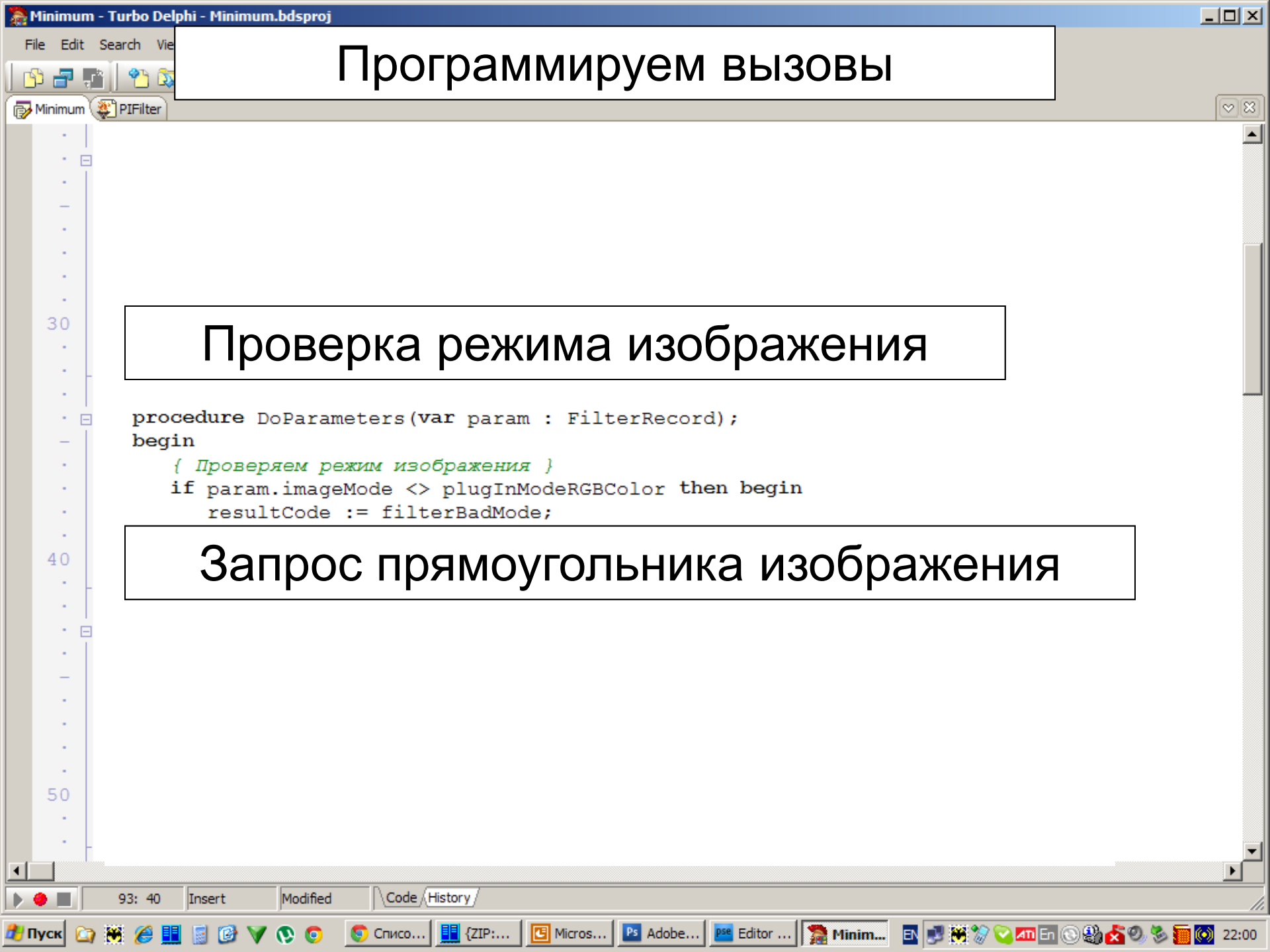
Взаимодействие плагина с Adobe® Photoshop®



Простейший плагин


Minimum - Turbo Delphi - Minimum.bdsproj
File Edit Search View Refactor Project Run Component Tools Window Help
Debug Layout
Minimum
{ Минимальный плагин-фильтр }
library Minimum;
{$E 8bf} // Расширение
{$R Minimum_prop.res} // Подключение файла ресурсов
uses
PIGeneral, PITypes, PIFilter, Math;
const
MAX_BUF_SIZE = 1000000000; // Просто большое число
MAX = 255; // Максимальное значение для 8-разрядных режимов
type
tDataBuffer = array [0..MAX_BUF_SIZE-1] of byte;
// Это только тип
tCurve = array[byte] of byte;
var
resultCode : smallint;
Модуль плагина. Начало
...
9: 49
Insert
Modified
Code
History
Пуск
19:50


Minimum - Turbo Delphi - Minimum.bdsproj
File Edit Search View Refactor Project Run Component Tools Window Help
Debug Layout
Minimum
91
procedure ENTRYPOINT(selector : smallint; // Код запрашиваемого действия
var param : FilterRecord; // Параметры изображения и не только
var data : integer; // Не обращаем внимания
var result: smallint); cdecl; // Код возврата
begin
resultCode := noErr;
case selector of
filterSelectorAbout :; // Вывести диалог About
100 filterSelectorParameters: DoParameters(param); // Запрос параметров
filterSelectorPrepare :; // Изменение размера памяти, если хочется.
filterSelectorStart : DoStart(param); //Запрос прямоугольника изображения
filterSelectorContinue : DoContinue(param); // Обработка
filterSelectorFinish :; // Завершающие операции
else
resultCode := filterBadParameters;
end;
{ Нужно не забывать инициализировать переменную result при каждом
вызове плагина }
110 result := resultCode;
end;
exports
ENTRYPOINT;
end.
Модуль плагина. Точка входа
91: 1
Insert
Modified
Code
History
Пуск
21:24



Программируем вызовы

Проверка режима изображения

```
procedure DoParameters(var param : FilterRecord);
begin
   { Проверяем режим изображения }
   if param.imageMode <> plugInModeRGBColor then begin
      resultCode := filterBadMode;
```

Запрос прямоугольника изображения

Программируем вызовы. Основная обработка

Minimum - Turbo Delphi - Minimum.bdsproj

File

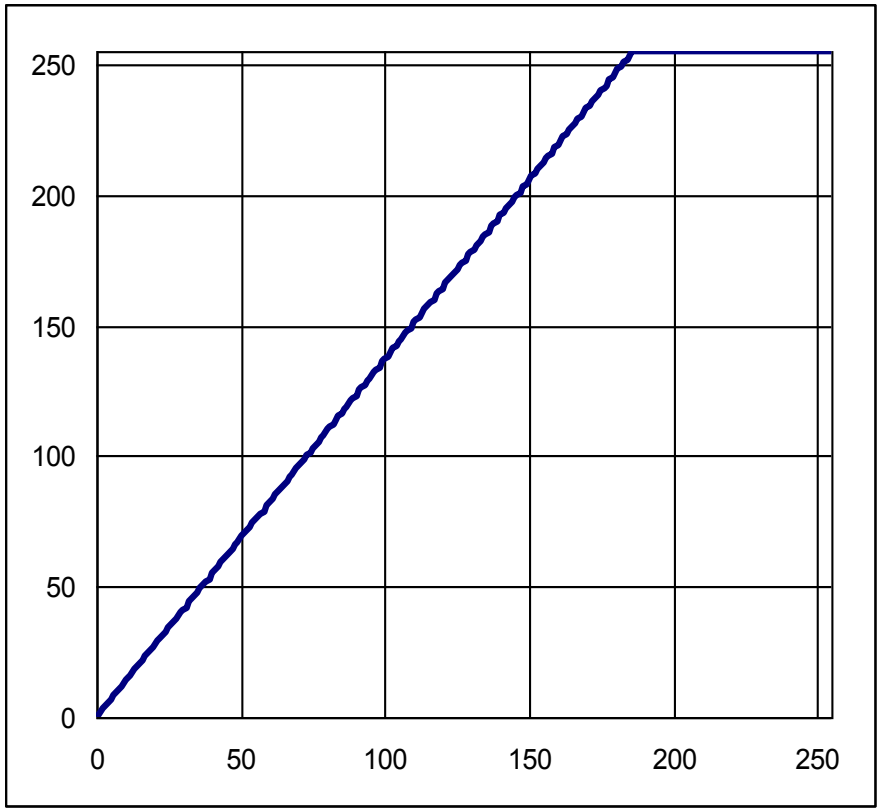
Minimum | PIFilter

```
  procedure DoContinue(var param : FilterRecord);
  var
     inData    : ^tDataBuffer;
     outData   : ^tDataBuffer;
     w, h      : integer;
     i         : integer;
     curve     : tCurve;

const
   MAX_BUF_SIZE = 1000000000; // Просто большое число
   MAX = 255; // Максимальное значение для 8-разрядных режимов

type
   tDataBuffer  = array [0..MAX_BUF_SIZE-1] of byte;
   // Это только тип

   tCurve = array[byte] of byte;

     SetRect(param.inRect, 0, 0, 0, 0);
     param.outRect := param.inRect;
     param.maskRect := param.inRect;
  end;
```

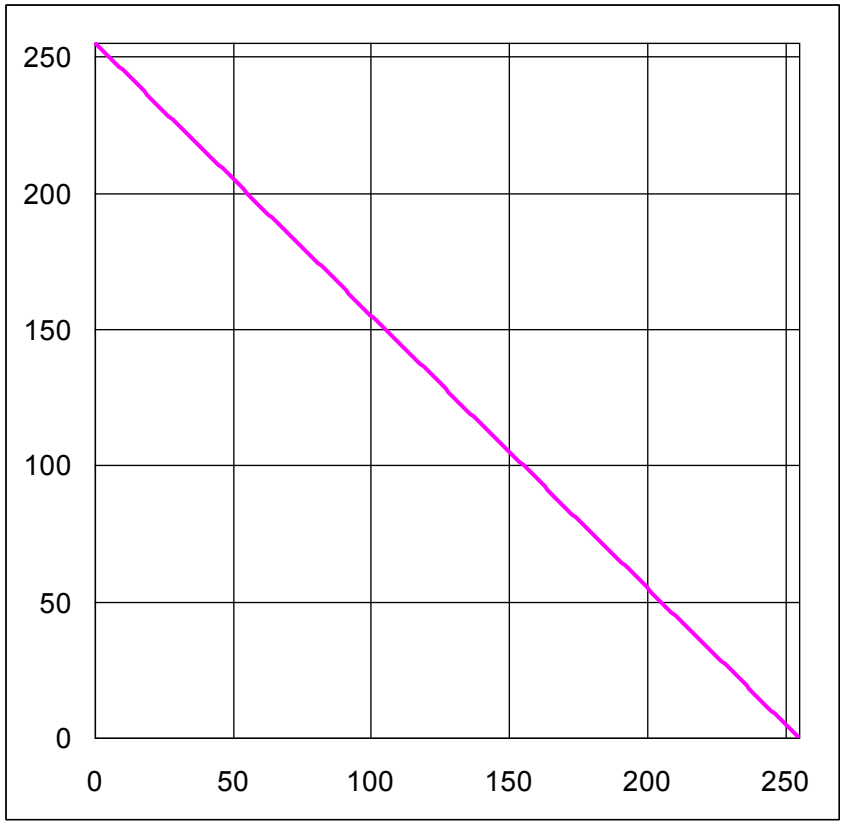
92: 4 | Insert | Modified | Code | History

Пуск | Списо... | {ZIP:... | Micros... | Adobe... | Editor ... | Minim... | 22:10

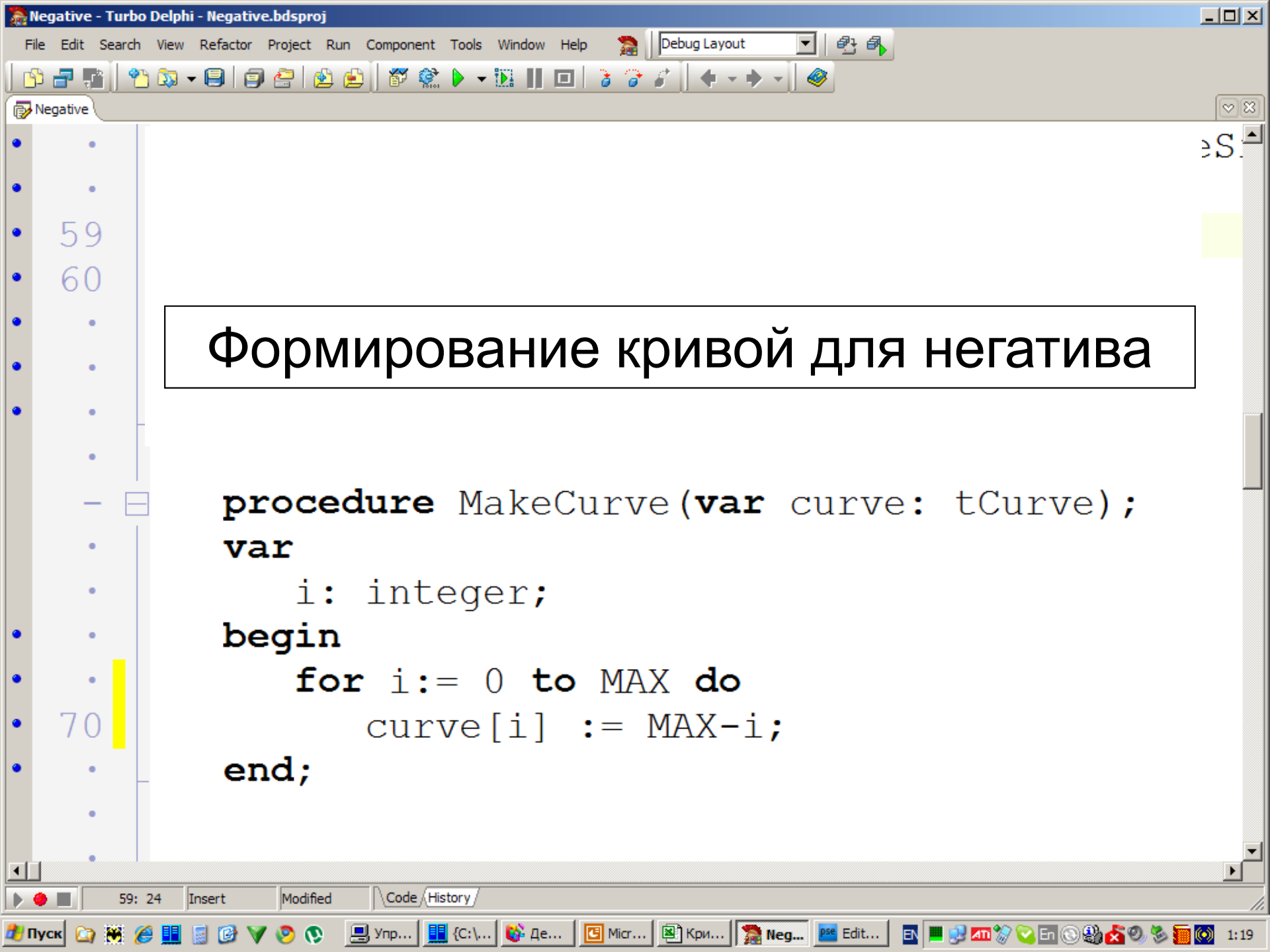
Варианты кривых

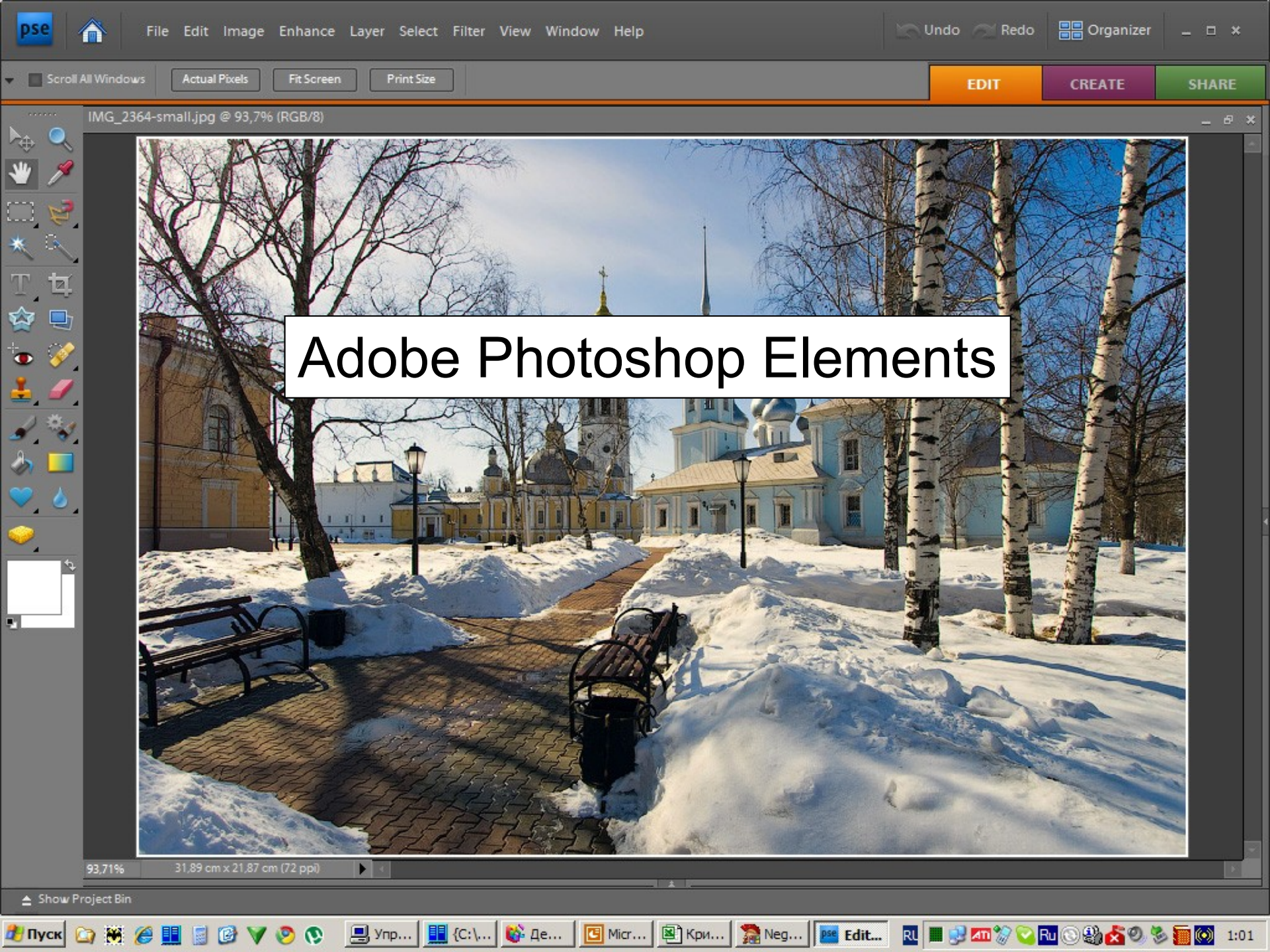


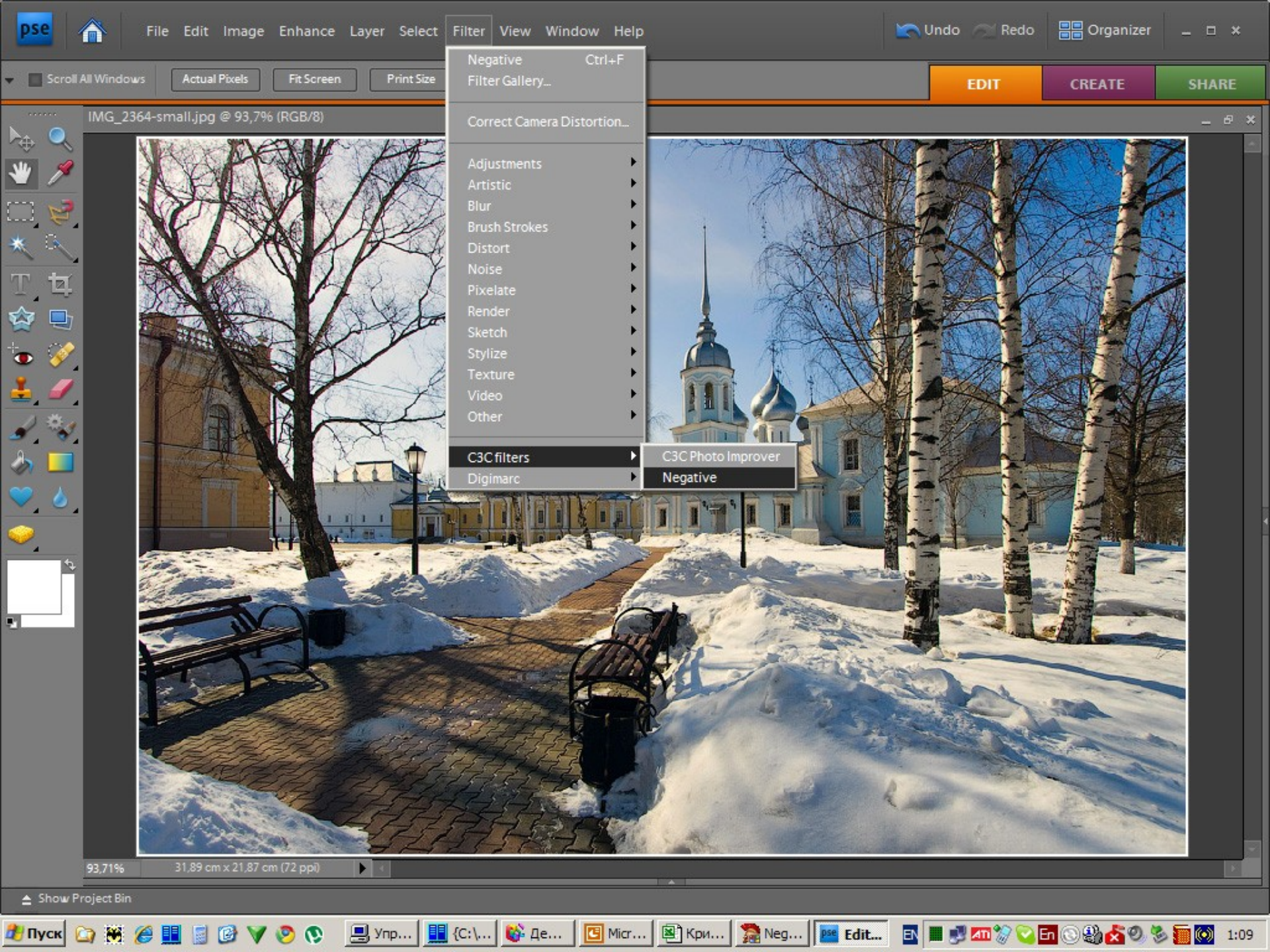
Экспозиция (увеличение)

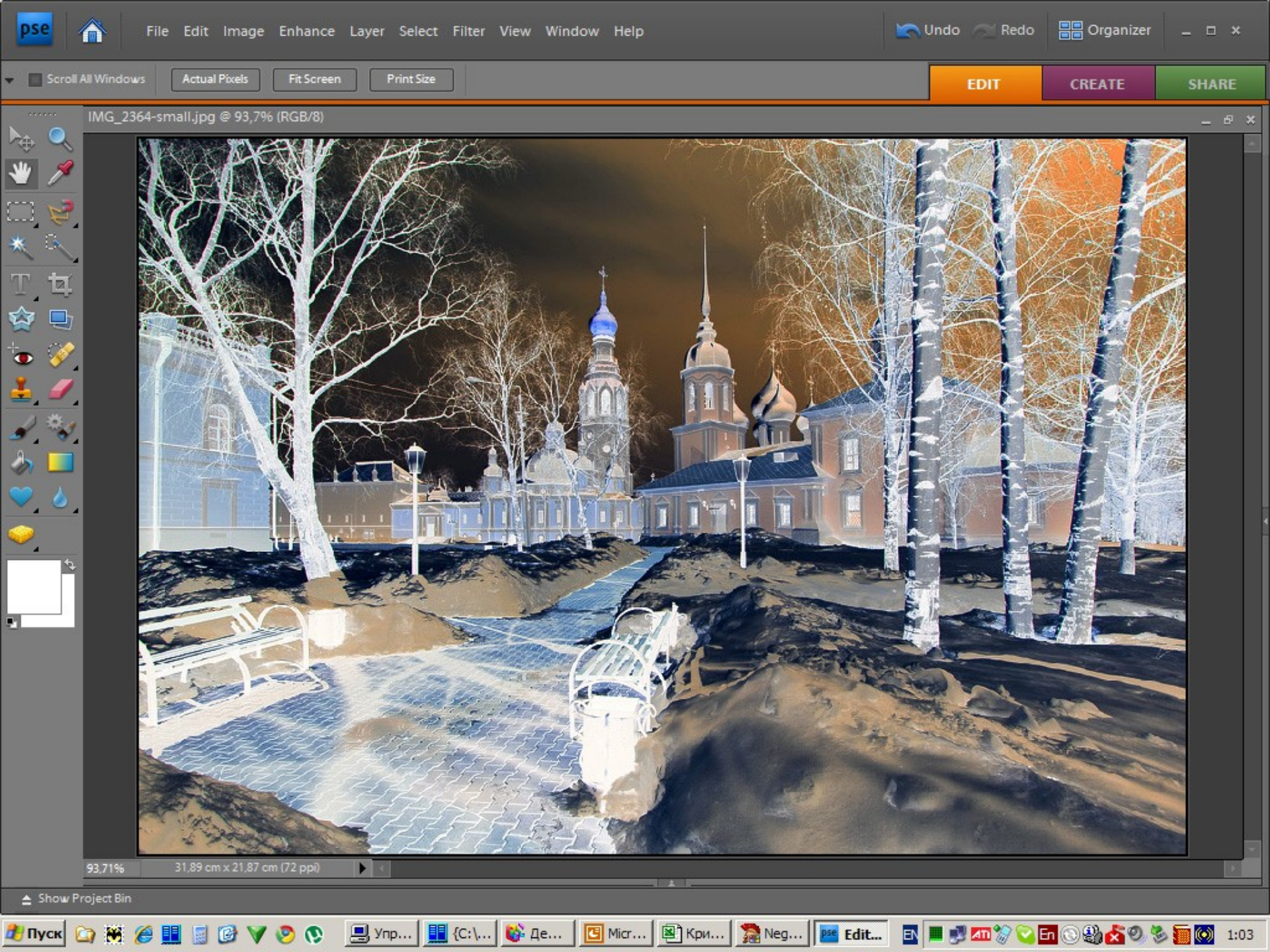


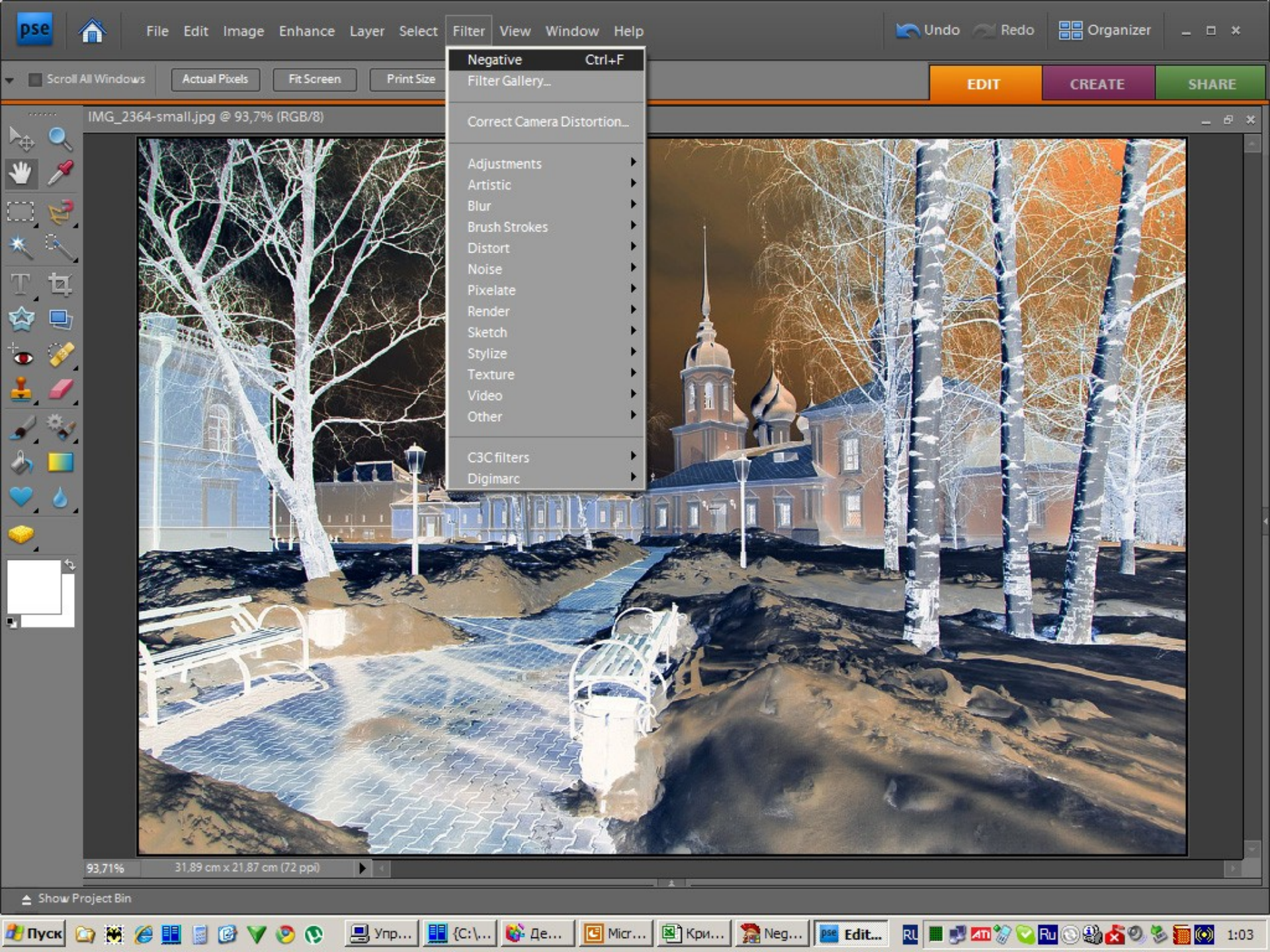
Негатив

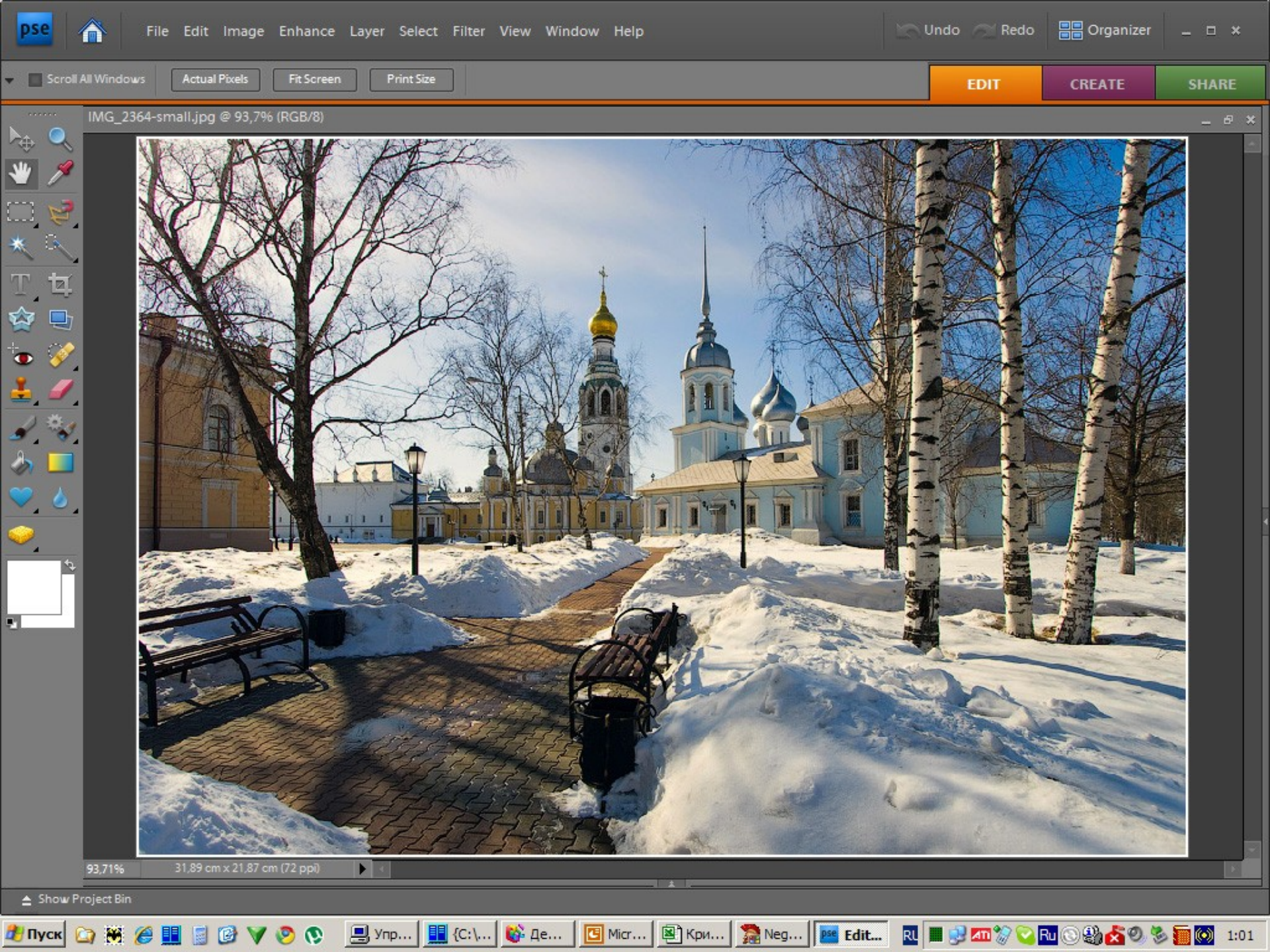

Negative - Turbo Delphi - Negative.bdsproj
File Edit Search View Refactor Project Run Component Tools Window Help
Debug Layout
Negative
59
60
70
Формирование кривой для негатива
procedure MakeCurve(var curve: tCurve);
var
i: integer;
begin
for i:= 0 to MAX do
curve[i] := MAX-i;
end;
59: 24
Insert
Modified
Code
History
Пуск
1:19


File Edit Image Enhance Layer Select Filter View Window Help
Undo
Redo
Organizer
Scroll All Windows
Actual Pixels
Fit Screen
Print Size
EDIT
CREATE
SHARE
IMG_2364-small.jpg @ 93,7% (RGB/8)
Adobe Photoshop Elements
93,71%
31,89 cm x 21,87 cm (72 ppi)
Show Project Bin
Пуск
1:01

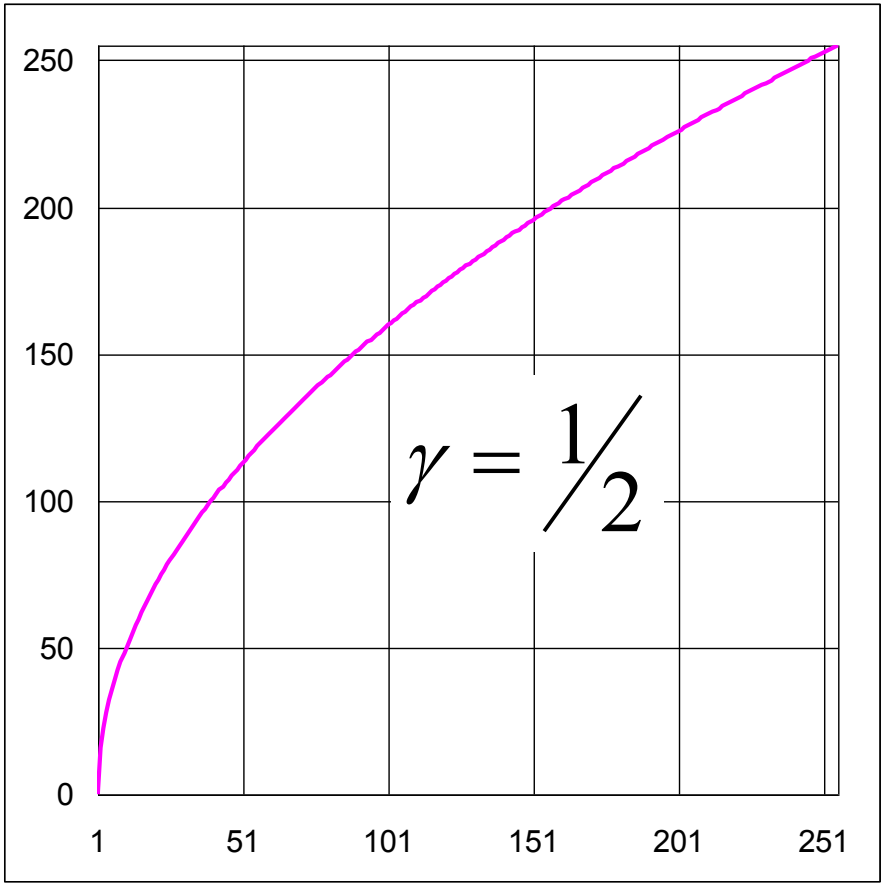

File Edit Image Enhance Layer Select Filter View Window Help
Undo Redo Organizer
Scroll All Windows
Actual Pixels
Fit Screen
Print Size
EDIT
CREATE
SHARE
IMG_2364-small.jpg @ 93,7% (RGB/8)
Negative Ctrl+F
Filter Gallery...
Correct Camera Distortion...
Adjustments
Artistic
Blur
Brush Strokes
Distort
Noise
Pixelate
Render
Sketch
Stylize
Texture
Video
Other
C3C filters
Digimarc
C3C Photo Improver
Negative
93,71%
31,89 cm x 21,87 cm (72 ppi)
Show Project Bin
Пуск
1:09


pse
File Edit Image Enhance Layer Select Filter View Window Help
Undo Redo Organizer
Scroll All Windows
Actual Pixels
Fit Screen
Print Size
EDIT
CREATE
SHARE
IMG_2364-small.jpg @ 93,7% (RGB/8)
93,71%
31,89 cm x 21,87 cm (72 ppi)
Show Project Bin
Пуск
Упр...
{C:\...
Де...
Micr...
Кри...
Neg...
Edit...
1:03

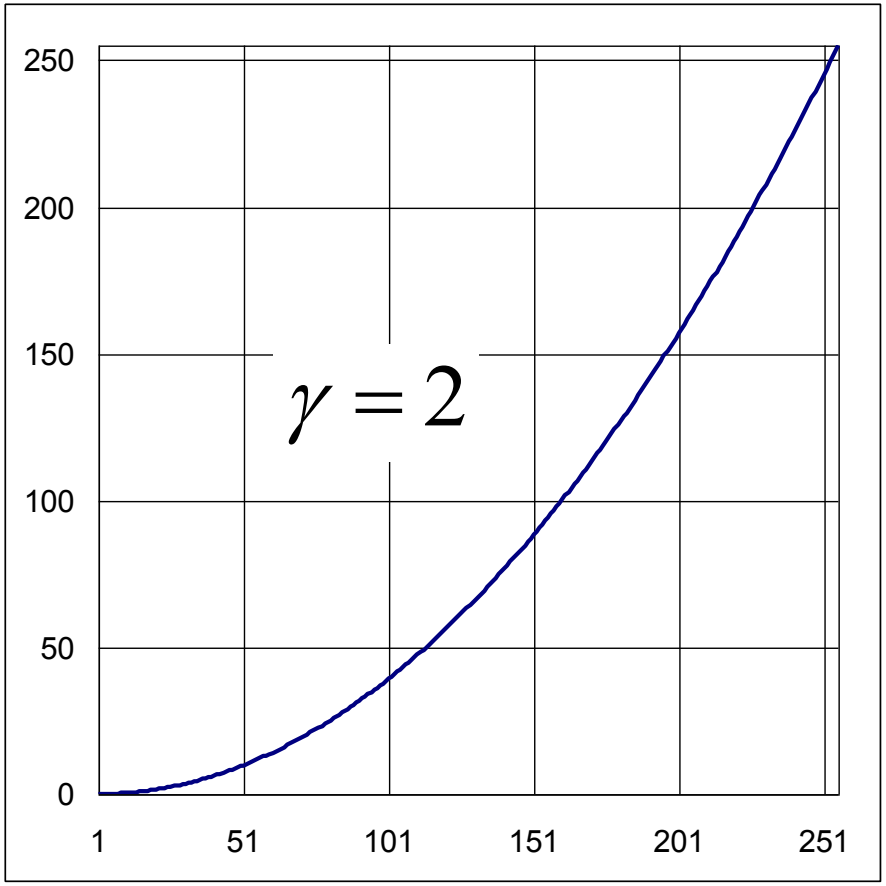

File Edit Image Enhance Layer Select Filter View Window Help
Undo Redo Organizer
Scroll All Windows
Actual Pixels
Fit Screen
Print Size
EDIT
CREATE
SHARE
Negative Ctrl+F
Filter Gallery...
Correct Camera Distortion...
Adjustments
Artistic
Blur
Brush Strokes
Distort
Noise
Pixelate
Render
Sketch
Stylize
Texture
Video
Other
C3C filters
Digimarc
IMG_2364-small.jpg @ 93,7% (RGB/8)
93,71%
31,89 cm x 21,87 cm (72 ppi)
Show Project Bin
Пуск
Упр...
{C:\...
Де...
Micr...
Кри...
Neg...
Edit...
1:03


File Edit Image Enhance Layer Select Filter View Window Help
Undo Redo Organizer
Scroll All Windows
Actual Pixels
Fit Screen
Print Size
EDIT
CREATE
SHARE
IMG_2364-small.jpg @ 93,7% (RGB/8)
93,71%
31,89 cm x 21,87 cm (72 ppi)
Show Project Bin
Пуск
Упр...
{C:\...
Де...
Micr...
Кри...
Neg...
Edit...
1:01

Варианты кривых

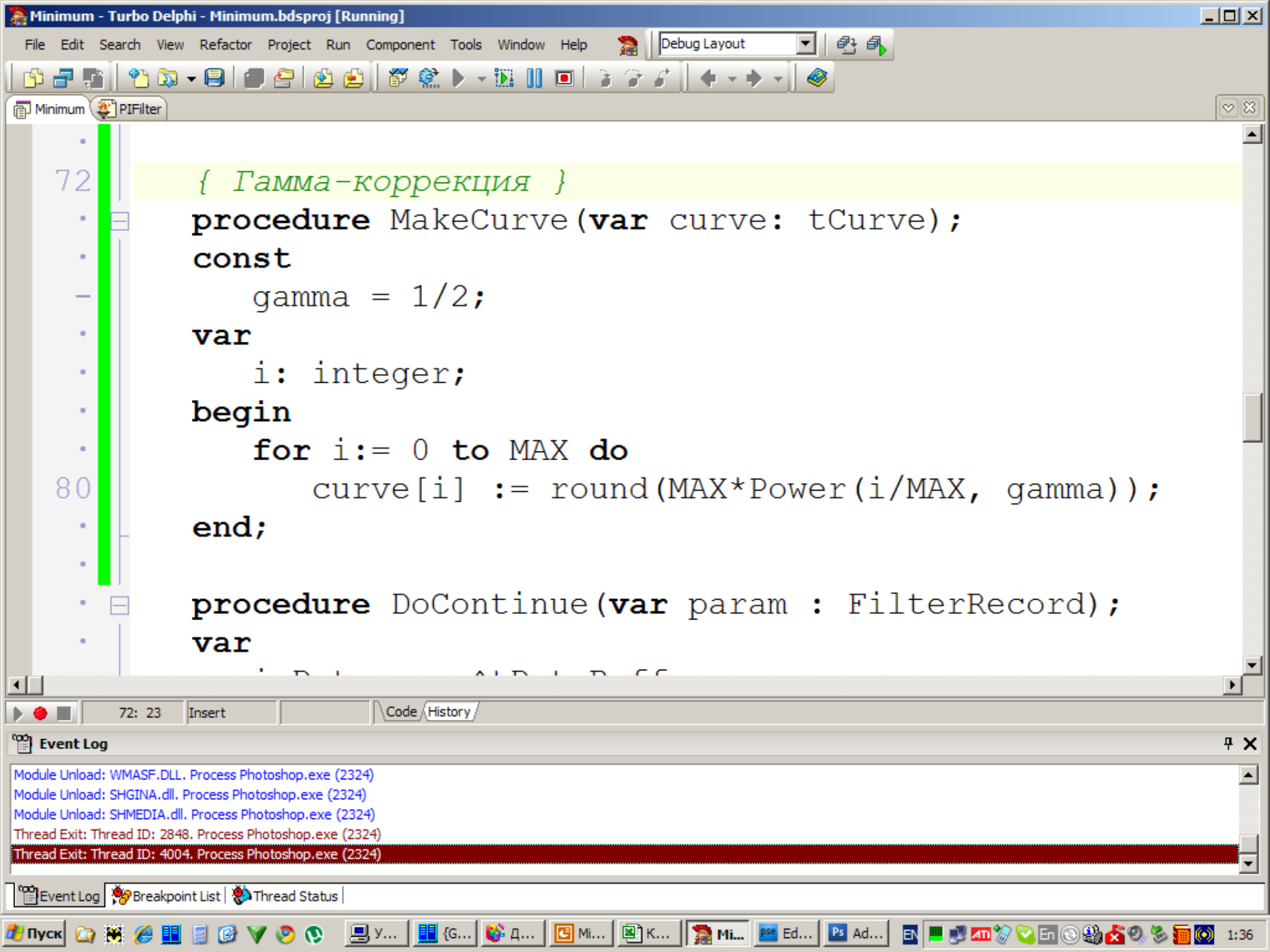


Осветление



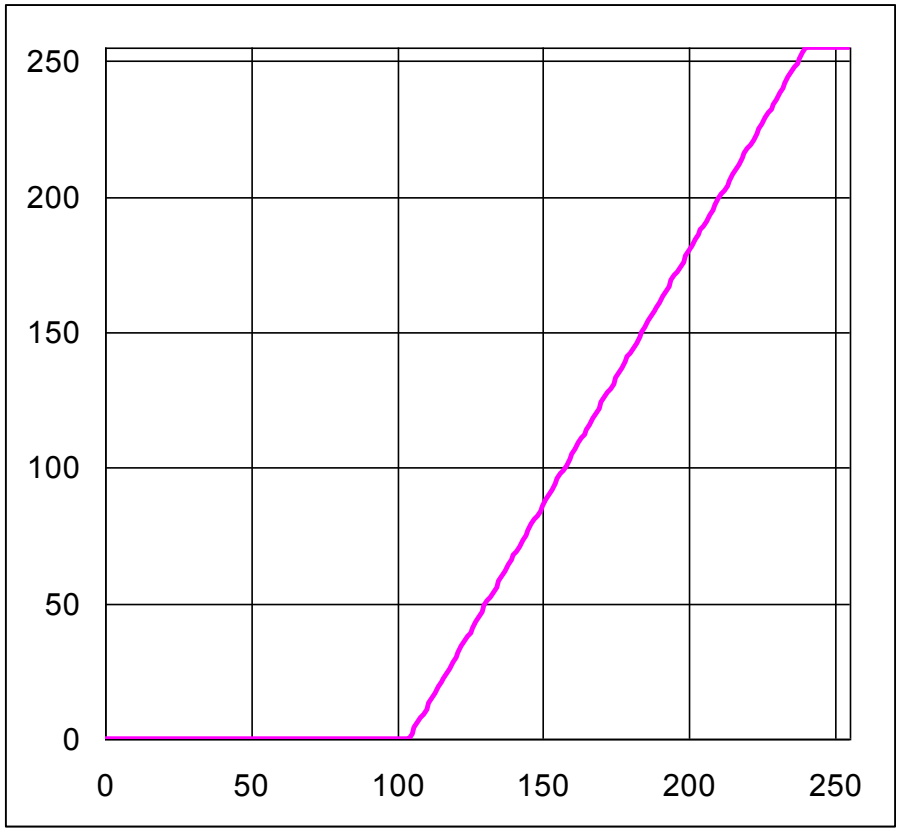
Затемнение

Гамма-коррекция

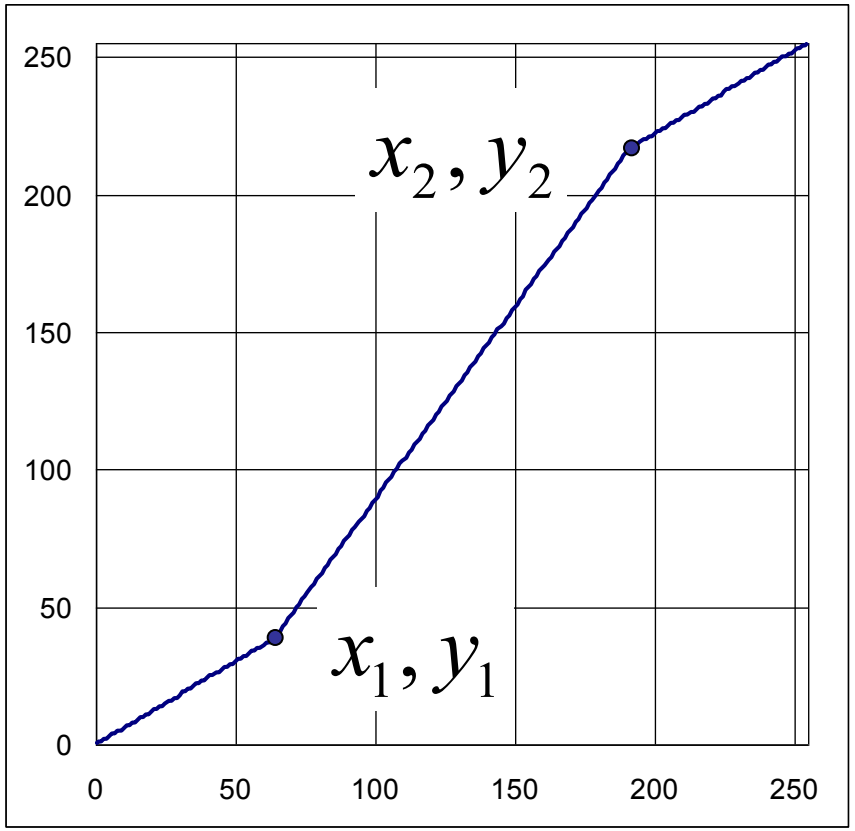

Minimum - Turbo Delphi - Minimum.bdsproj [Running]
File Edit Search View Refactor Project Run Component Tools Window Help
Debug Layout
Minimum
PIFilter
72 { Гамма-коррекция }
procedure MakeCurve(var curve: tCurve);
const
gamma = 1/2;
var
i: integer;
begin
for i:= 0 to MAX do
80 curve[i] := round(MAX*Power(i/MAX, gamma));
end;
procedure DoContinue(var param : FilterRecord);
var
72: 23
Insert
Code
History
Event Log
Module Unload: WMASF.DLL. Process Photoshop.exe (2324)
Module Unload: SHGINA.dll. Process Photoshop.exe (2324)
Module Unload: SHMEDIA.dll. Process Photoshop.exe (2324)
Thread Exit: Thread ID: 2848. Process Photoshop.exe (2324)
Thread Exit: Thread ID: 4004. Process Photoshop.exe (2324)
Event Log
Breakpoint List
Thread Status
Пуск
1:36

$\gamma = 1/2$

Варианты кривых



Уровни



Контрастность

Контрастность

Minimum - Turbo Delphi - Minimum.bdsproj

File Edit Search View Refactor Project Run Component Tools Window Help

Debug Layout

Minimum | PIFilter

```
procedure MakeCurve(var curve: tCurve);
var
   i: integer;
   c: real;
begin
   for i:= 0 to MAX do begin
      c := i/MAX;
      if c<=0.04045 then
         c := c/12.92
      else
         c := Power((c+0.055)/(1+0.055), 2.4);
      curve[i] := round(MAX*c);
   end;
end;
```

Линеаризация sRGB

92: 4 | Insert | Modified | Code | History

Пуск | Списо... | {ZIP:... | Micros... | Adobe... | Editor ... | Minim... | 22:14

Ресурсы плагина

В файле ресурса задаются:

- Название плагина в меню Photoshop.
- Название группы плагинов, в которую он включается в меню Filter.
- Режимы изображения, которые поддерживает плагин.
- ...

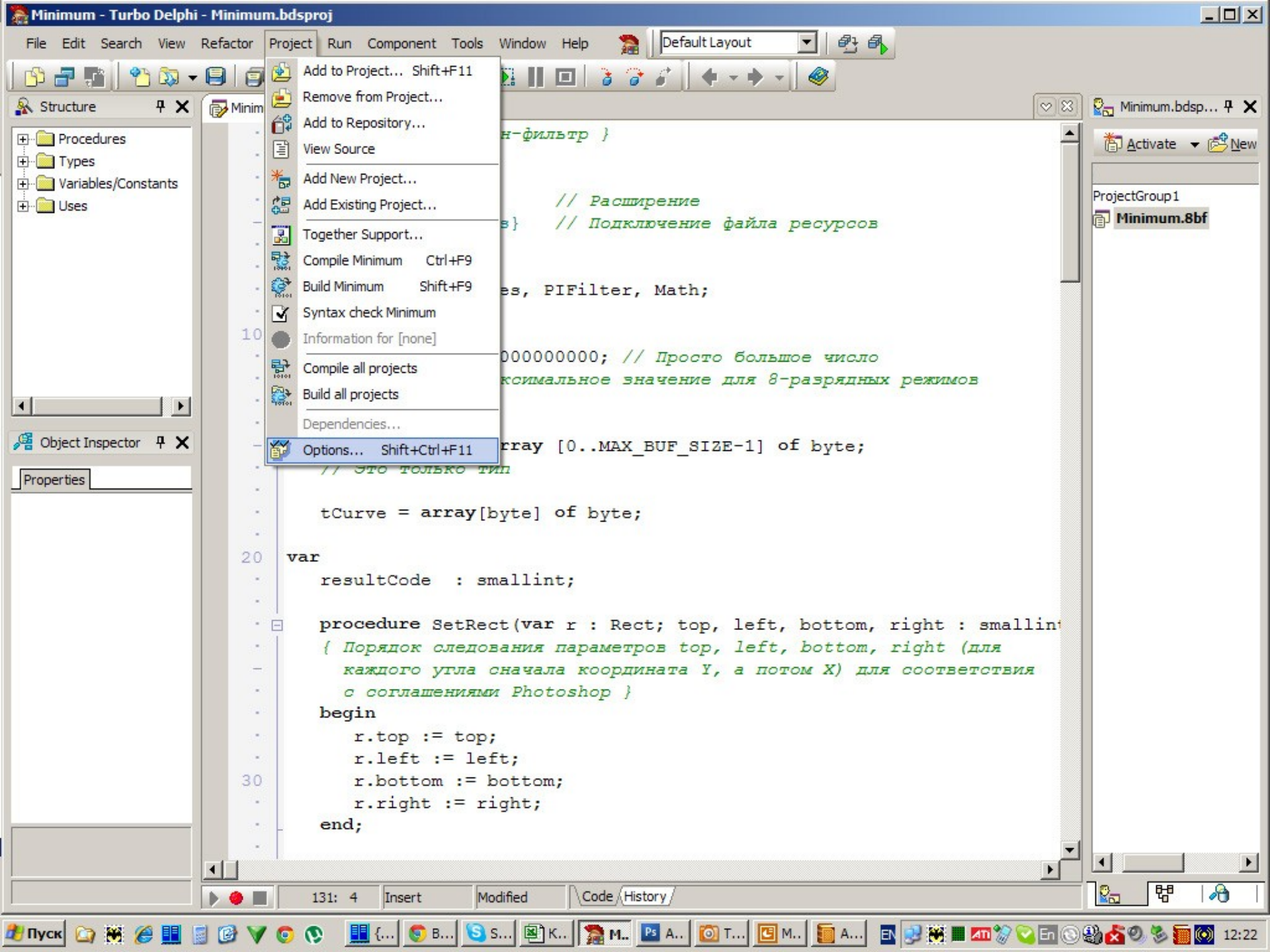
```
/* Ресурсы плагина для Photophop
   Здесь описываются свойства (properties) плагина
*/

...

/* Свойство 'name' или PINameProperty - название плагина,
   которое будет меню Filter редактора. */
   "MIB8",
   "eman",  /* 'name' */
   0L,
   9L,
   "\x08Negative\0\0\0",

/* Свойство 'catg' или PICategoryProperty - пункт подменю
   меню Filter, в котором будет отображаться имя плагина. */
   "MIB8",
   "gtac",  /* 'catg' */
   0L,
   12L,
   "\x0BС3С filters",
...

/* Свойство 'enbl' или PIEnableInfoProperty - показывает, когда
   будет активно меню вызова плагина. */
   "MIB8",
   "lbne",  /* 'enbl' */
   0L,
   29L,
   "in (PSHOP_ImageMode, RGBMode)\0\0\0",
END
```

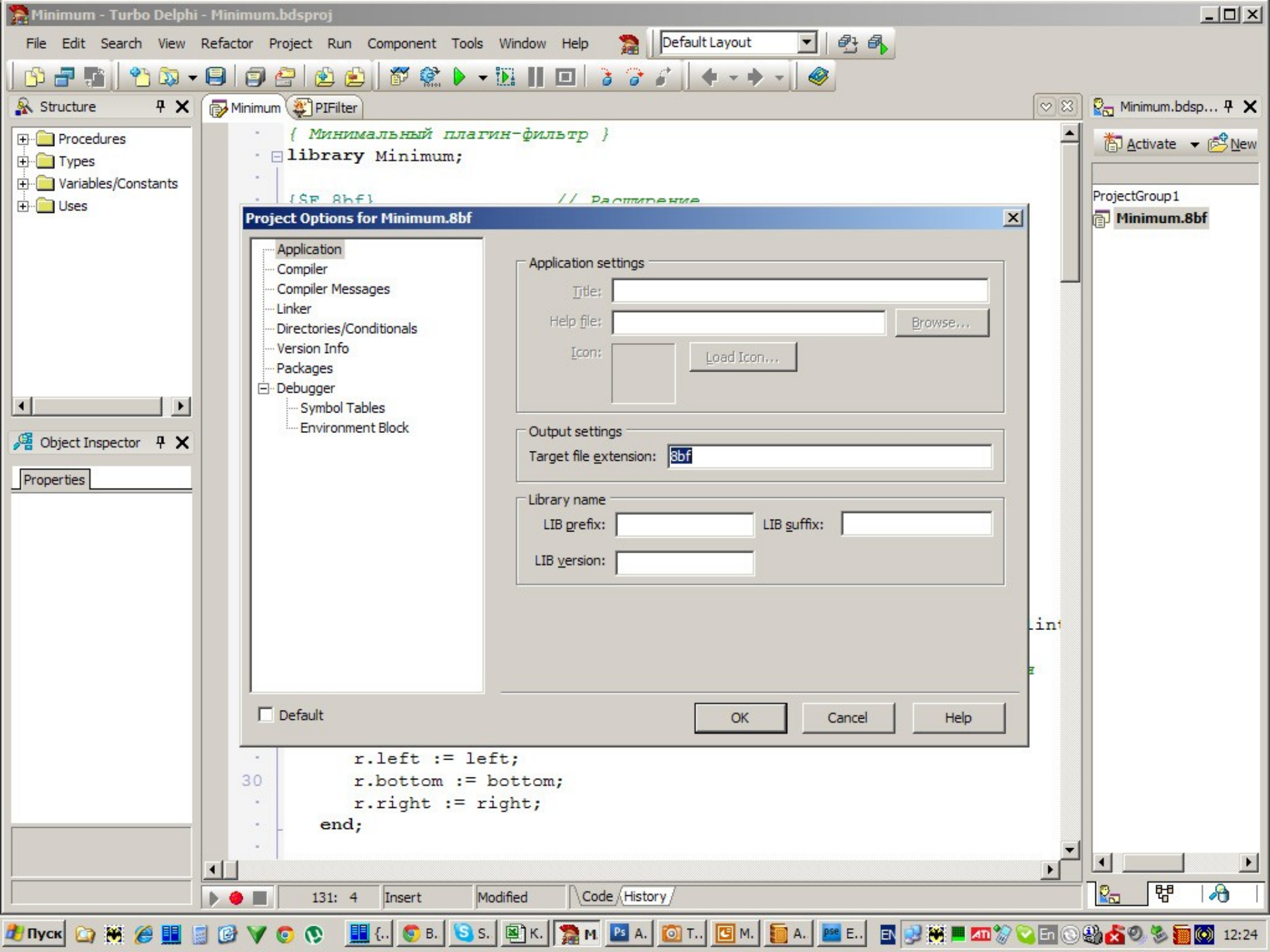
Файл Negative_prop.rc

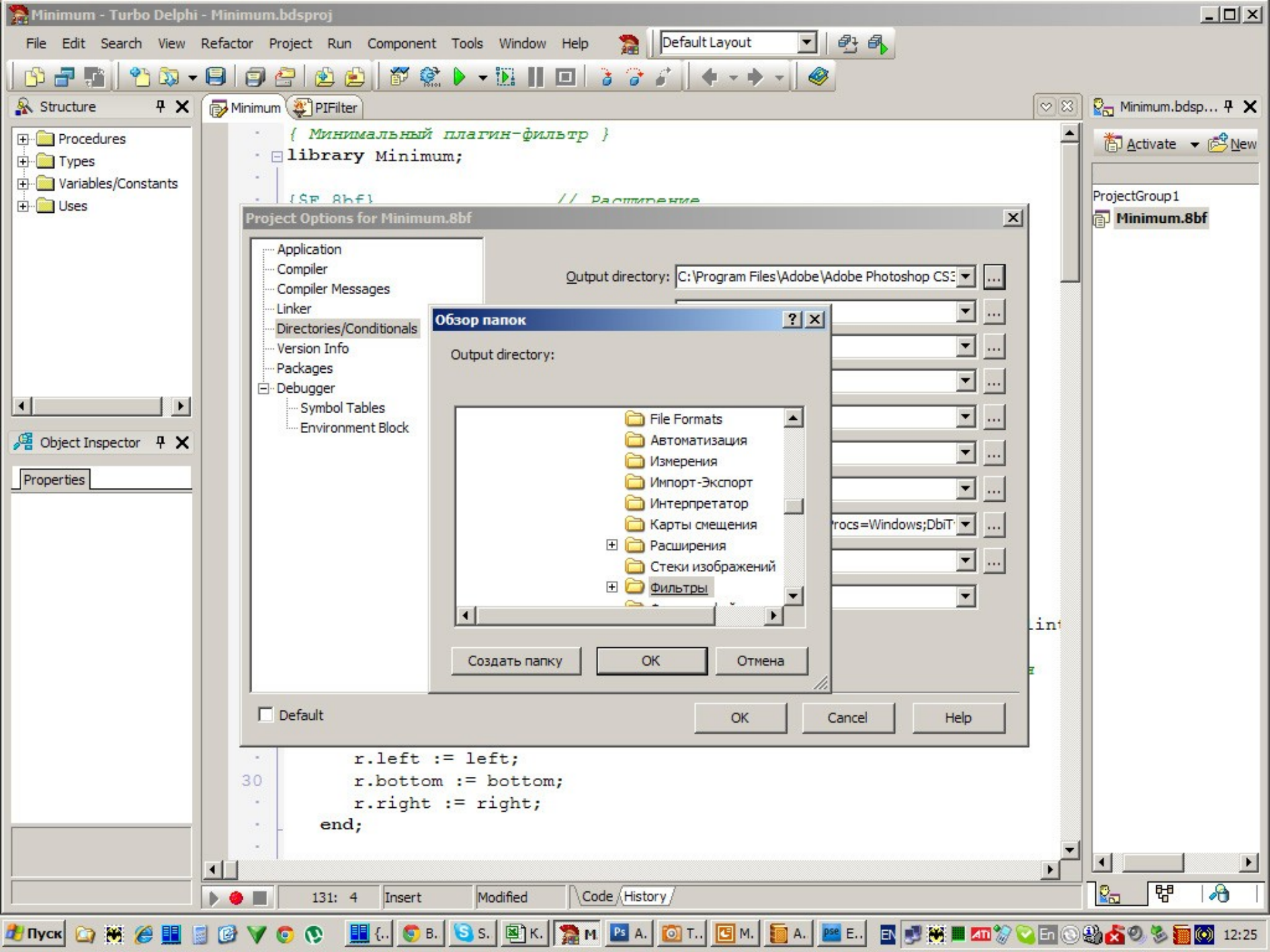
Компиляция и отладка плагина

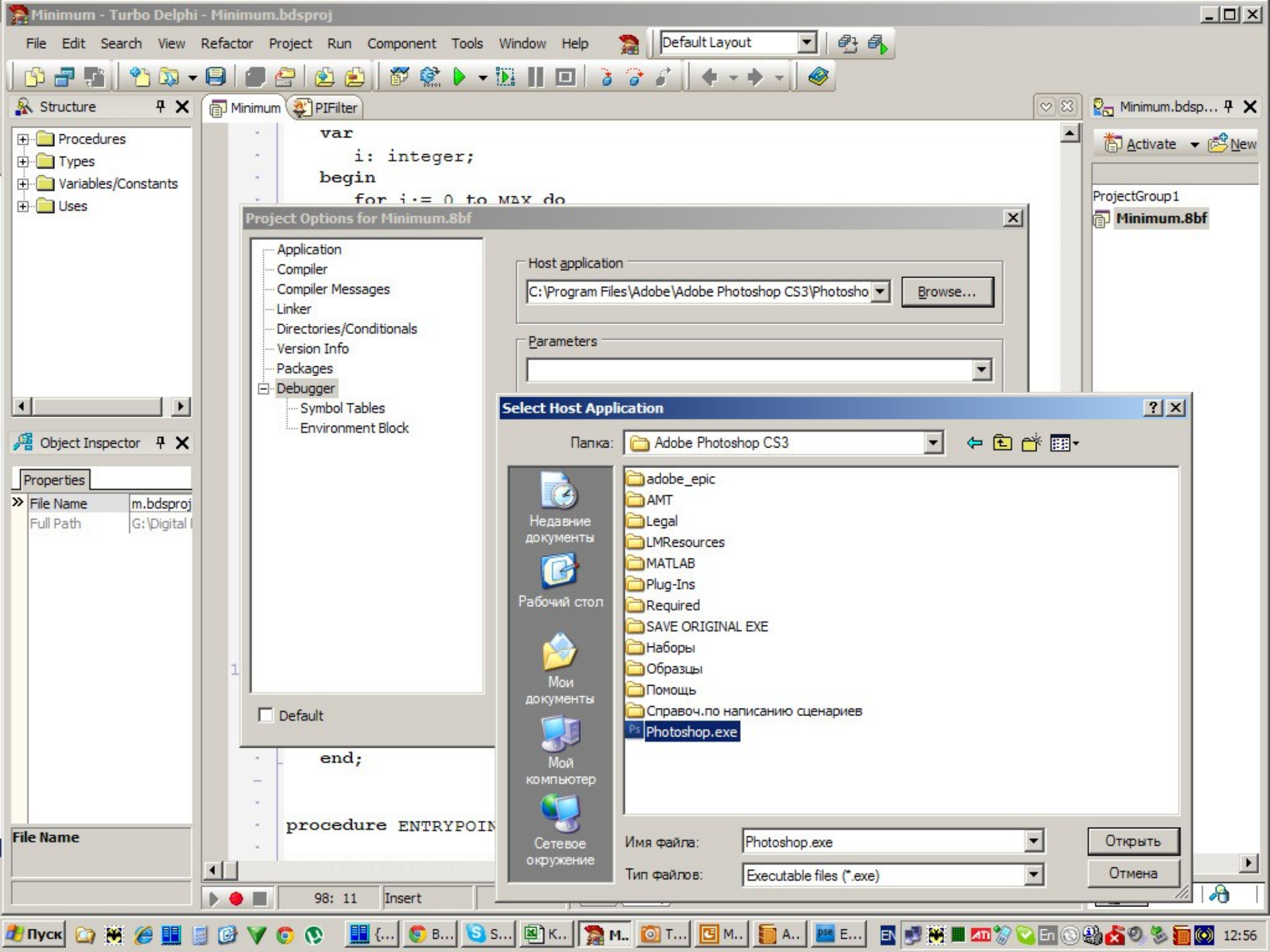
Настройки:

- Project/Options/Application/Target file extension
- Project/Options/Directories/Output Directory
- Project/Options/Debugger/Host application

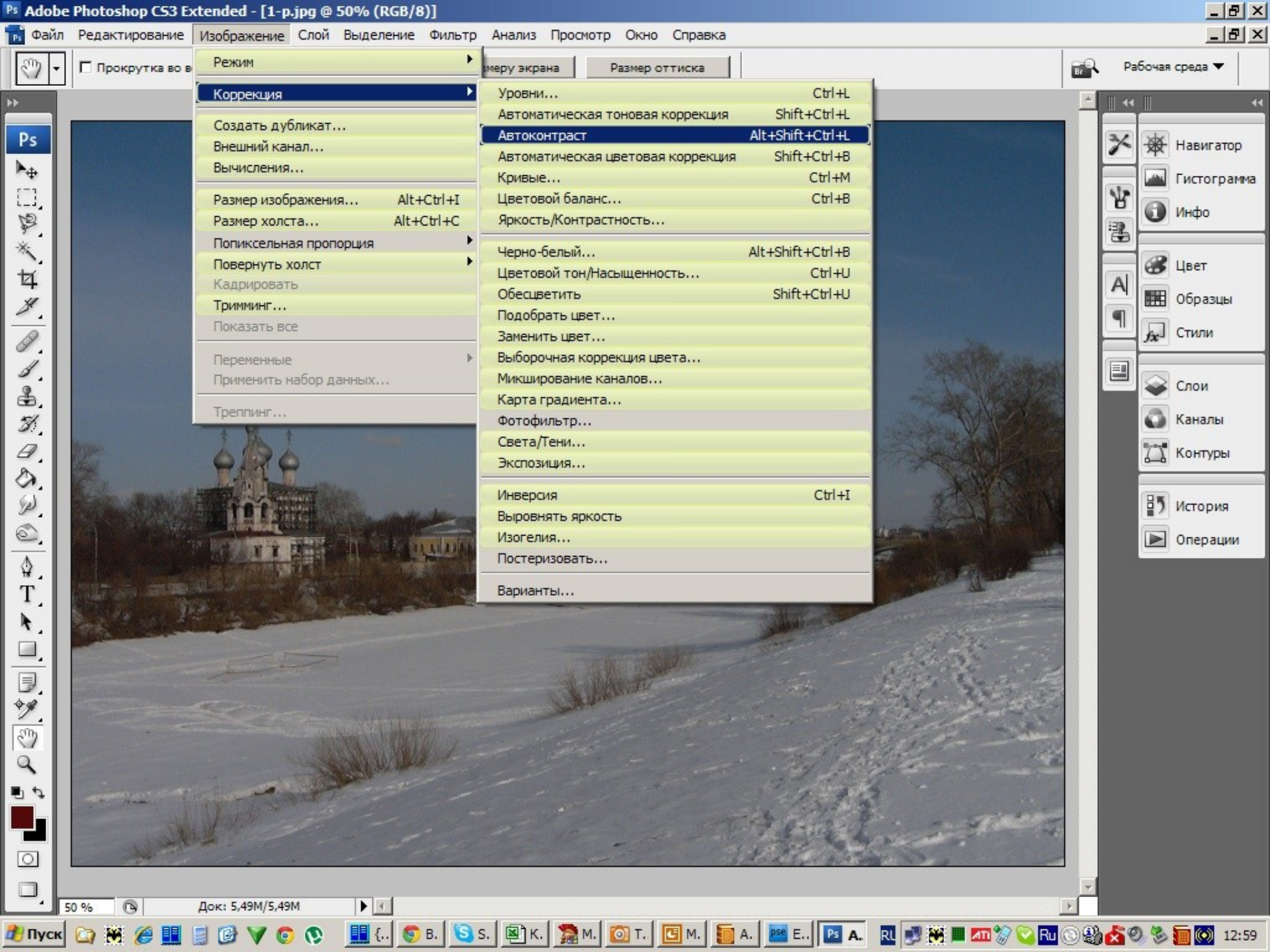

Minimum - Turbo Delphi - Minimum.bdsproj
File Edit Search View Refactor Project Run Component Tools Window Help
Default Layout
Add to Project... Shift+F11
Remove from Project...
Add to Repository...
View Source
Add New Project...
Add Existing Project...
Together Support...
Compile Minimum Ctrl+F9
Build Minimum Shift+F9
Syntax check Minimum
Information for [none]
Compile all projects
Build all projects
Dependencies...
Options... Shift+Ctrl+F11
Structure
Procedures
Types
Variables/Constants
Uses
Object Inspector
Properties
Minimum.bdsp...
Activate
New
ProjectGroup1
Minimum.8bf
// Расширение
// Подключение файла ресурсов
es, PIFilter, Math;
000000000; // Просто большое число
ксимальное значение для 8-разрядных режимов
rray [0..MAX_BUF_SIZE-1] of byte;
// Это только тип
tCurve = array[byte] of byte;
var
resultCode : smallint;
procedure SetRect(var r : Rect; top, left, bottom, right : smallint
{ Порядок следования параметров top, left, bottom, right (для
каждого угла сначала координата Y, а потом X) для соответствия
с соглашениями Photoshop }
begin
r.top := top;
r.left := left;
r.bottom := bottom;
r.right := right;
end;
131: 4
Insert
Modified
Code
History
Пуск
12:22

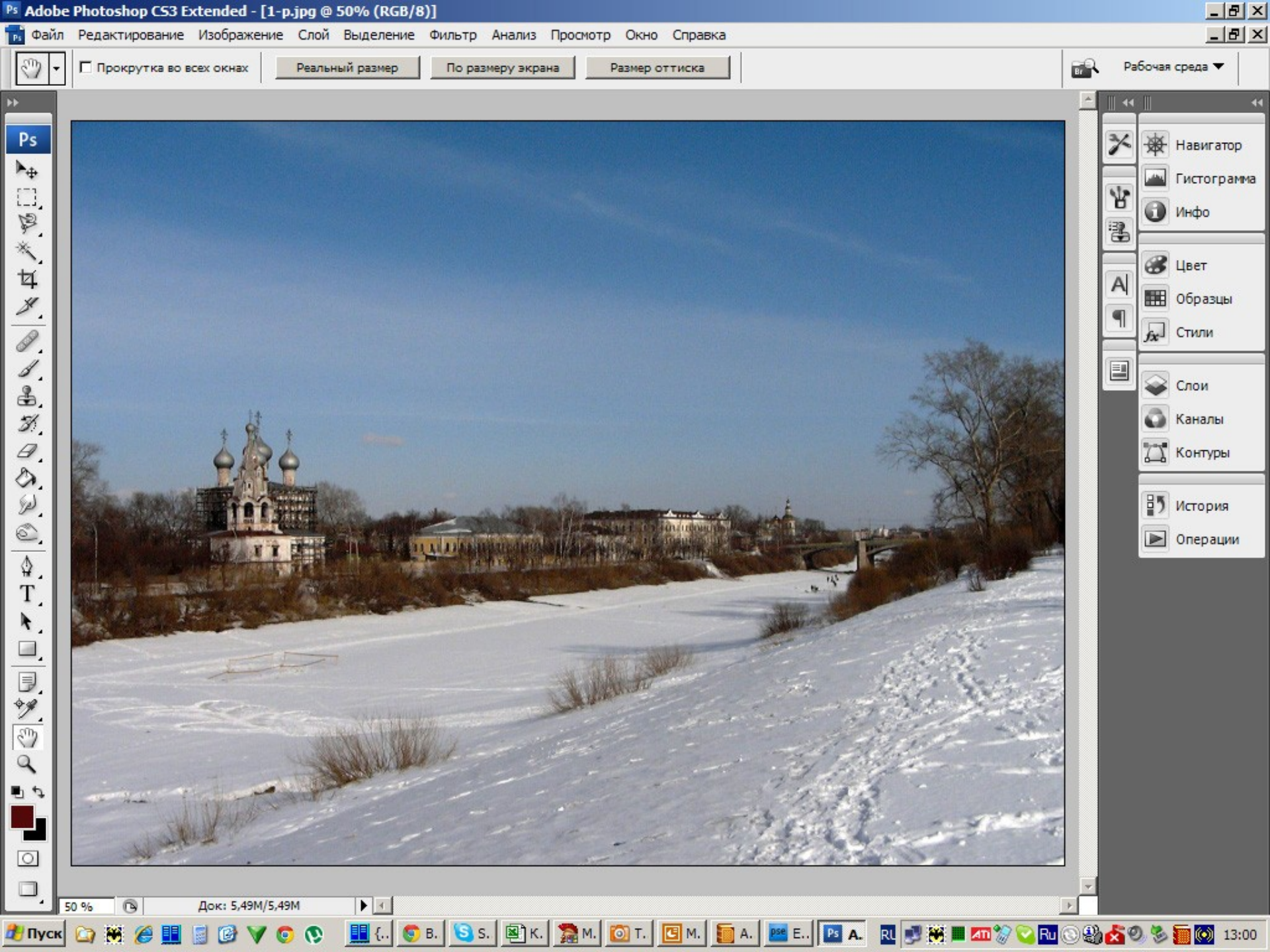

Minimum - Turbo Delphi - Minimum.bdsproj
File Edit Search View Refactor Project Run Component Tools Window Help
Default Layout
Structure
Procedures
Types
Variables/Constants
Uses
Object Inspector
Properties
Minimum
PIFilter
{ Минимальный плагин-фильтр }
library Minimum;
{$R 8bf}
// Расширение
Project Options for Minimum.8bf
Application
Compiler
Compiler Messages
Linker
Directories/Conditionals
Version Info
Packages
Debugger
Symbol Tables
Environment Block
Application settings
Title:
Help file:
Browse...
Icon:
Load Icon...
Output settings
Target file extension:
8bf
Library name
LIB prefix:
LIB suffix:
LIB version:
Default
OK
Cancel
Help
r.left := left;
r.bottom := bottom;
r.right := right;
end;
131: 4
Insert
Modified
Code
History
Minimum.bdsp...
Activate
New
ProjectGroup1
Minimum.8bf
Пуск
12:24


Minimum - Turbo Delphi - Minimum.bdsproj
File Edit Search View Refactor Project Run Component Tools Window Help
Default Layout
Structure
Procedures
Types
Variables/Constants
Uses
Object Inspector
Properties
Minimum
PIFilter
{ Минимальный плагин-фильтр }
library Minimum;
{$R 8bf}
// Расширение
Project Options for Minimum.8bf
Application
Compiler
Compiler Messages
Linker
Directories/Conditionals
Version Info
Packages
Debugger
Symbol Tables
Environment Block
Output directory:
C:\Program Files\Adobe\Adobe Photoshop CS3
Обзор папок
Output directory:
File Formats
Автоматизация
Измерения
Импорт-Экспорт
Интерпретатор
Карты смещения
Расширения
Стеки изображений
Фильтры
Создать папку
OK
Отмена
Default
OK
Cancel
Help
r.left := left;
r.bottom := bottom;
r.right := right;
end;
131: 4
Insert
Modified
Code
History
Minimum.bdsp...
Activate
New
ProjectGroup1
Minimum.8bf
Пуск
12:25


Minimum - Turbo Delphi - Minimum.bdsproj
File Edit Search View Refactor Project Run Component Tools Window Help
Default Layout
Structure
Procedures
Types
Variables/Constants
Uses
Minimum
PIFilter
var
i: integer;
begin
for i:= 0 to MAX do
Minimum.bdsp...
Activate
New
ProjectGroup1
Minimum.8bf
Project Options for Minimum.8bf
Application
Compiler
Compiler Messages
Linker
Directories/Conditionals
Version Info
Packages
Debugger
Symbol Tables
Environment Block
Default
Host application
C:\Program Files\Adobe\Adobe Photoshop CS3\Photosho
Browse...
Parameters
Select Host Application
Папка:
Adobe Photoshop CS3
Недавние документы
Рабочий стол
Мои документы
Мой компьютер
Сетевое окружение
adobe_epic
AMT
Legal
LMResources
MATLAB
Plug-Ins
Required
SAVE ORIGINAL EXE
Наборы
Образцы
Помощь
Справоч.по написанию сценариев
Photoshop.exe
Имя файла:
Photoshop.exe
Тип файлов:
Executable files (*.exe)
Открыть
Отмена
Object Inspector
Properties
File Name
m.bdsproj
Full Path
G:\Digital
File Name
end;
procedure ENTRYPOIN
98: 11
Insert
Пуск
12:56

Автоконтраст


Adobe Photoshop CS3 Extended - [1-p.jpg @ 50% (RGB/8)]
Файл
Редактирование
Изображение
Слой
Выделение
Фильтр
Анализ
Просмотр
Окно
Справка
Прокрутка во в
Размер экрана
Размер оттиска
Рабочая среда
Режим
Коррекция
Создать дубликат...
Внешний канал...
Вычисления...
Размер изображения... Alt+Ctrl+I
Размер холста... Alt+Ctrl+C
Попиксельная пропорция
Повернуть холст
Кадрировать
Тримминг...
Показать все
Переменные
Применить набор данных...
Треппинг...
Уровни... Ctrl+L
Автоматическая тоновая коррекция Shift+Ctrl+L
Автоконтраст Alt+Shift+Ctrl+L
Автоматическая цветовая коррекция Shift+Ctrl+B
Кривые... Ctrl+M
Цветовой баланс... Ctrl+B
Яркость/Контрастность...
Черно-белый... Alt+Shift+Ctrl+B
Цветовой тон/Насыщенность... Ctrl+U
Обесцветить Shift+Ctrl+U
Подобрать цвет...
Заменить цвет...
Выборочная коррекция цвета...
Микширование каналов...
Карта градиента...
Фотофильтр...
Света/Тени...
Экспозиция...
Инверсия Ctrl+I
Выровнять яркость
Изогелия...
Постеризовать...
Варианты...
Навигатор
Гистограмма
Инфо
Цвет
Образцы
Стили
Слои
Каналы
Контуры
История
Операции
50 %
Док: 5,49M/5,49M
Пуск
12:59


Adobe Photoshop CS3 Extended - [1-p.jpg @ 50% (RGB/8)]
Файл Редактирование Изображение Слой Выделение Фильтр Анализ Просмотр Окно Справка
Прокрутка во всех окнах
Реальный размер
По размеру экрана
Размер оттиска
Рабочая среда
Навигатор
Гистограмма
Инфо
Цвет
Образцы
Стили
Слои
Каналы
Контуры
История
Операции
50 %
Док: 5,49M/5,49M
Пуск
13:00

Алгоритм 1

- Найти максимальное и минимальное значения RGB
- Линейно «растянуть» значения RGB, чтобы минимальное было 0, а максимальное 255:

$$RGB_{нов} = MAX \frac{(RGB - MinRGB)}{MaxRGB - MinRGB}$$

Autocontrast - Turbo Delphi - Autocontrast.bdsproj

File Edit Search View Refactor Project Run Component Tools Window Help

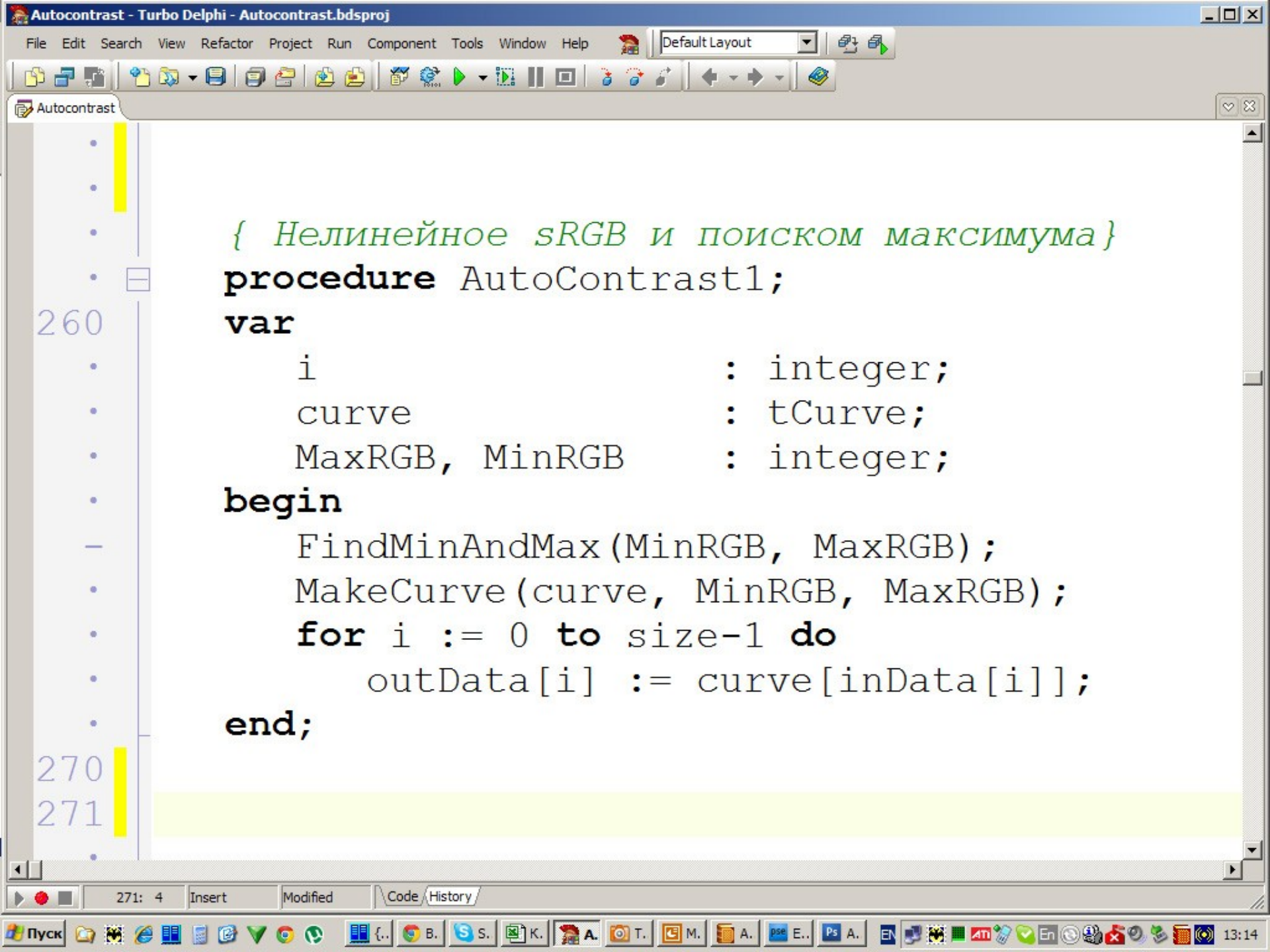
Default Layout

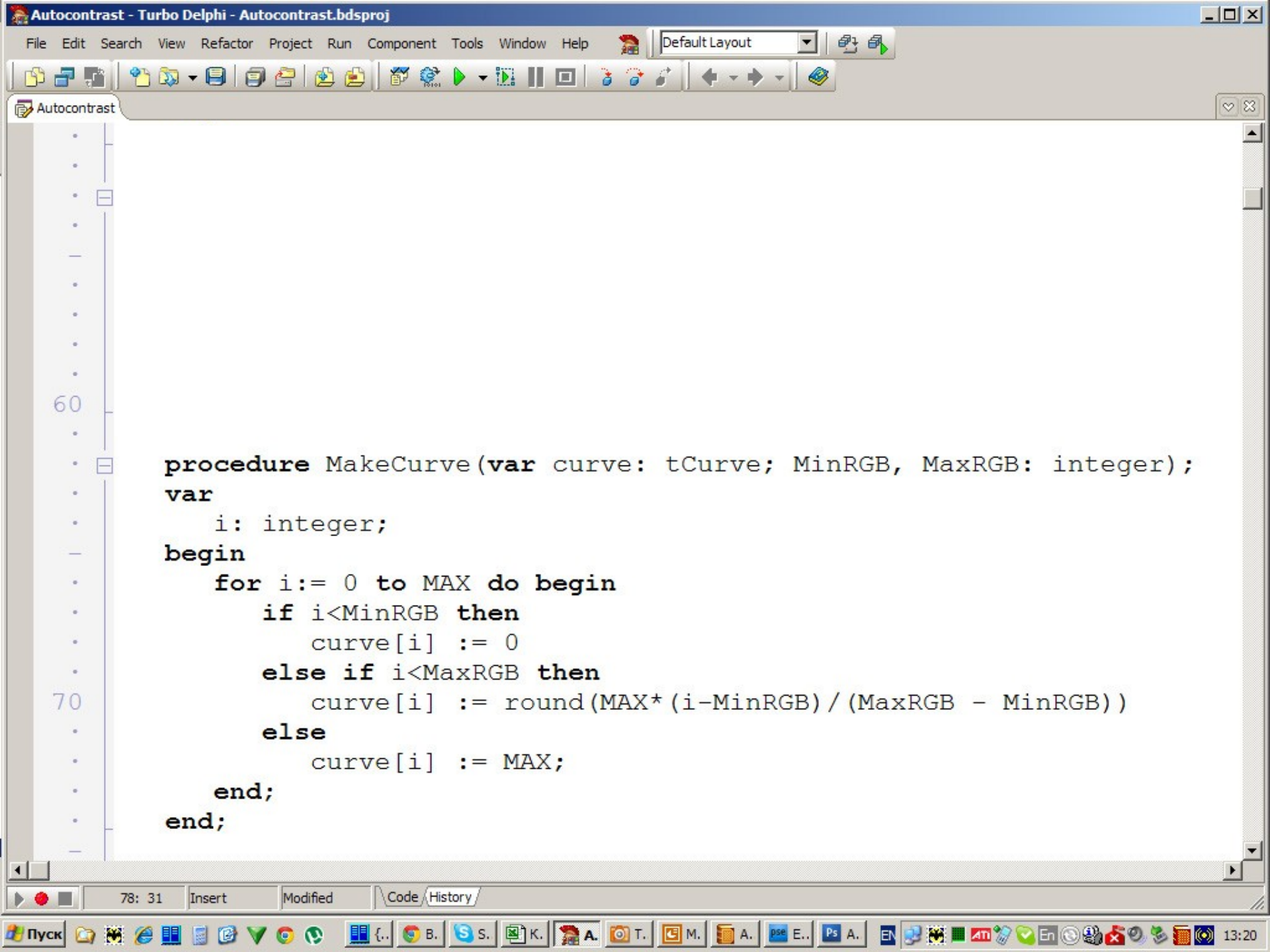
Autocontrast

```
procedure DoContinue(var param: FilterRecord);
begin
   inData := param.inData;
   outData := param.outData;
   w := param.imageSize.h;
   h := param.imageSize.v;
   planes := param.planes;
   size := w*h*planes;

   AutoContrast1;

   SetRect(param.inRect, 0, 0, 0, 0);
   param.outRect := param.inRect;
end;


procedure ENTRYPOINT(selector  : smallint;      { Код запраши
                     var param : FilterRecord;  { Параметры из
                     var data  : integer;
                     var result: smallint); cdecl;
begin
```

730: 52 | Insert | Modified | Code | History

Пуск 20:47


Autocontrast - Turbo Delphi - Autocontrast.bdsproj
File Edit Search View Refactor Project Run Component Tools Window Help
Default Layout
Autocontrast
{ Нелинейное sRGB и поиском максимума}
procedure AutoContrast1;
var
i : integer;
curve : tCurve;
MaxRGB, MinRGB : integer;
begin
FindMinAndMax(MinRGB, MaxRGB);
MakeCurve(curve, MinRGB, MaxRGB);
for i := 0 to size-1 do
outData[i] := curve[inData[i]];
end;
271: 4
Insert
Modified
Code
History
Пуск
13:14


Autocontrast - Turbo Delphi - Autocontrast.bdsproj
File Edit Search View Refactor Project Run Component Tools Window Help
Default Layout
Autocontrast
60
70
procedure MakeCurve(var curve: tCurve; MinRGB, MaxRGB: integer);
var
i: integer;
begin
for i:= 0 to MAX do begin
if i<MinRGB then
curve[i] := 0
else if i<MaxRGB then
curve[i] := round(MAX*(i-MinRGB)/(MaxRGB - MinRGB))
else
curve[i] := MAX;
end;
end;
78: 31
Insert
Modified
Code
History
Пуск
13:20

Алгоритм 1

Исходный

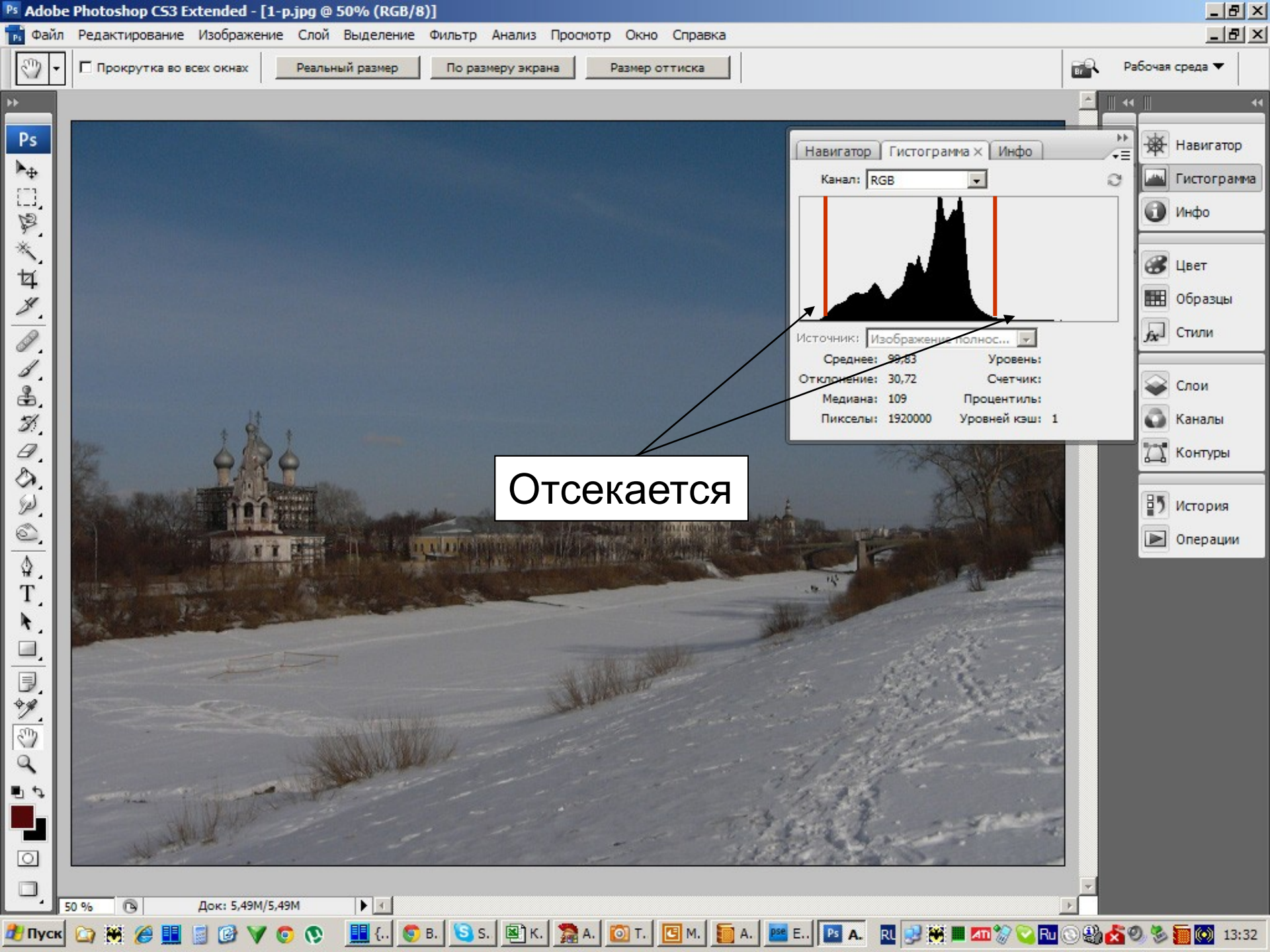
Алгоритм 1

Алгоритм 10

Алгоритм 2

- Построить гистограмму.
- Найти по гистограмме значения RGB, отсекающие заданную долю (`LostLevel=0.001`) самых больших и самых малых значений.
- Линейно «растянуть» значения RGB, чтобы минимальное было 0, а максимальное 255:

$$RGB_{нов} = MAX \frac{(RGB - MinRGB)}{MaxRGB - MinRGB}$$


Adobe Photoshop CS3 Extended - [1-p.jpg @ 50% (RGB/8)]
Файл Редактирование Изображение Слой Выделение Фильтр Анализ Просмотр Окно Справка
Прокрутка во всех окнах
Реальный размер
По размеру экрана
Размер оттиска
Рабочая среда
Навигатор
Гистограмма
Инфо
Канал: RGB
Источник: Изображение полнос...
Среднее: 99,83
Отклонение: 30,72
Медиана: 109
Пикселы: 1920000
Уровень:
Счетчик:
Процентиль:
Уровней кэш: 1
Отсекается
Цвет
Образцы
Стили
Слои
Каналы
Контуры
История
Операции
50 %
Док: 5,49M/5,49M
Пуск
13:32

Autocontrast - Turbo Delphi - Autocontrast.bdsproj

File Edit Search View Refactor Project Run Component Tools Window Help

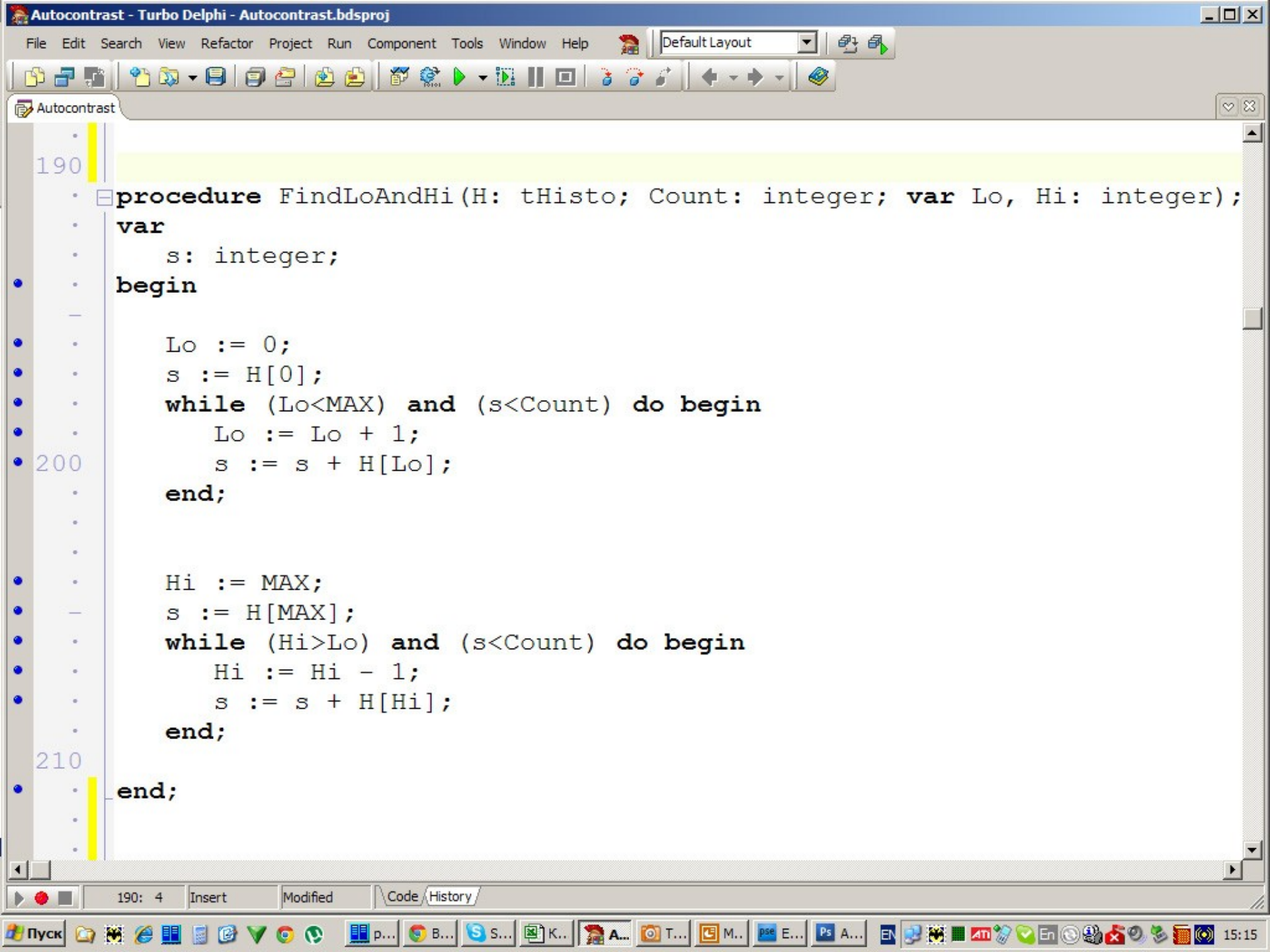
Default Layout

Autocontrast

```
{ Нелинейное sRGB с гистограммой}
procedure AutoContrast2;
const
   LostLevel = 0.001;
var
   i                    : integer;
   curve                : tCurve;
   histo                : tHisto;
   Lo, Hi              : integer;
begin
   BuildHistogram(histo);
   FindLoAndHi(histo,  round(size*LostLevel), Lo, Hi);
   MakeCurve(curve, Lo, Hi);
   for i := 0 to size-1 do
      outData[i] := curve[inData[i]];
   WriteCurve(curve, 'AC2.txt');
end;
```

299: 46 Insert Modified Code History

Пуск 15:07


Autocontrast - Turbo Delphi - Autocontrast.bdsproj
File Edit Search View Refactor Project Run Component Tools Window Help
Default Layout
Autocontrast
190
procedure FindLoAndHi(H: tHisto; Count: integer; var Lo, Hi: integer);
var
s: integer;
begin
Lo := 0;
s := H[0];
while (Lo<MAX) and (s<Count) do begin
Lo := Lo + 1;
200
s := s + H[Lo];
end;
Hi := MAX;
s := H[MAX];
while (Hi>Lo) and (s<Count) do begin
Hi := Hi - 1;
s := s + H[Hi];
end;
210
end;
190: 4
Insert
Modified
Code
History
Пуск
15:15

Алгоритм 2

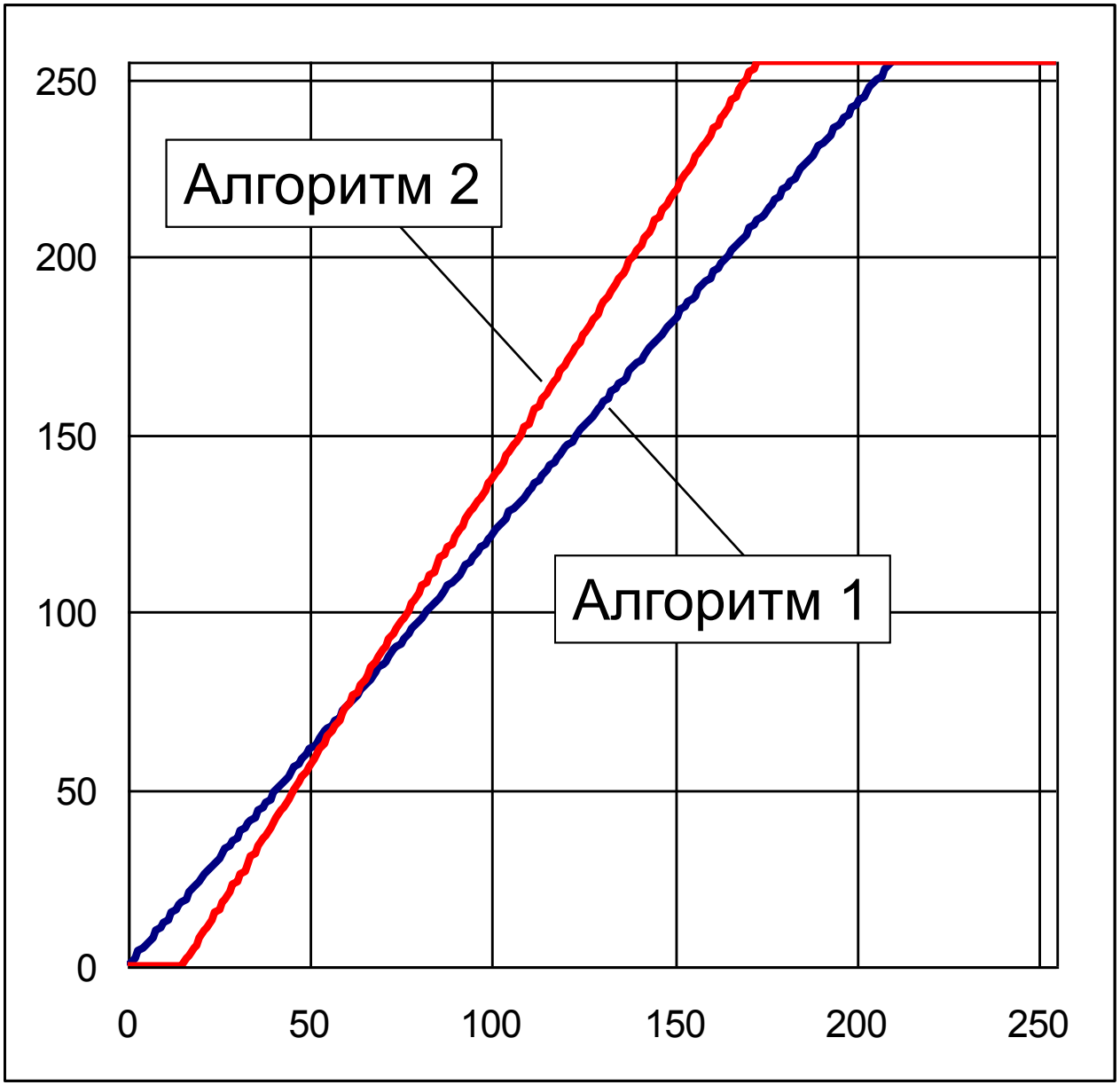
Алгоритм 1

Дополнительные главы компьютерной графики

Лекция 6

Copyright © С. Свердлов, 2009-2017
http://www.uni-vologda.ac.ru/~c3c

Кривые



Алгоритм 2

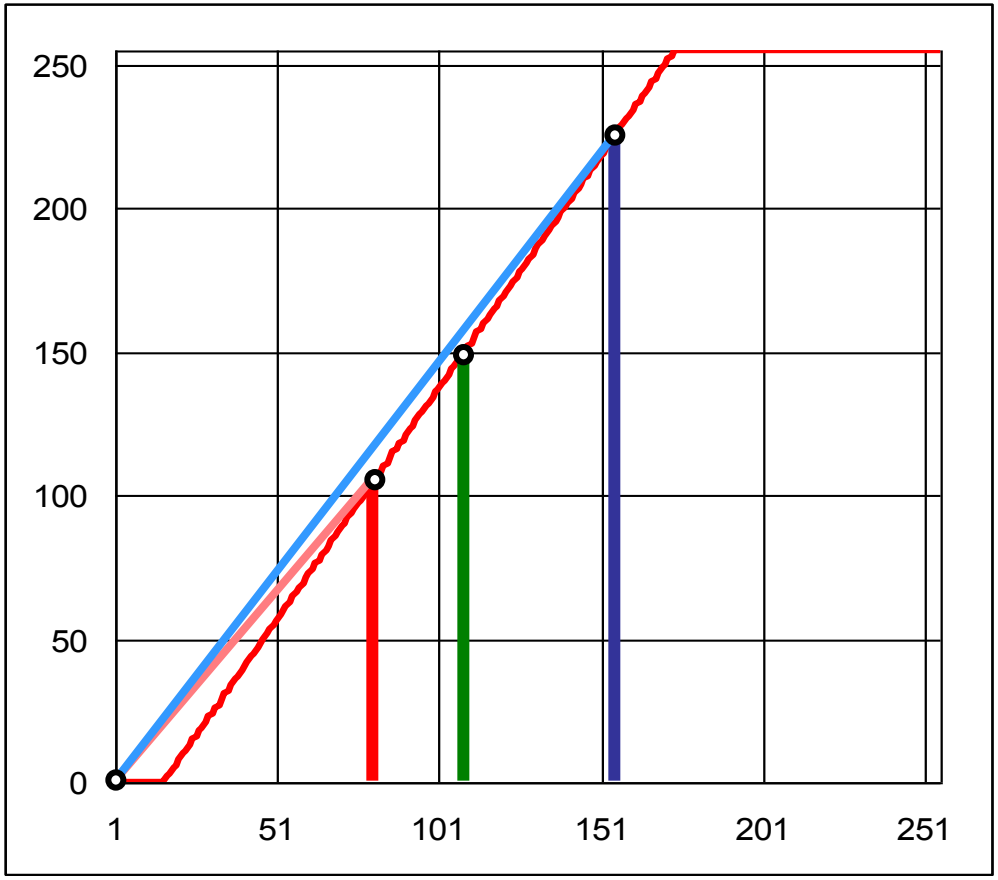
Алгоритм 10

Проблемы

- Вычисления выполняются в нелинейном sRGB (умножать можно)

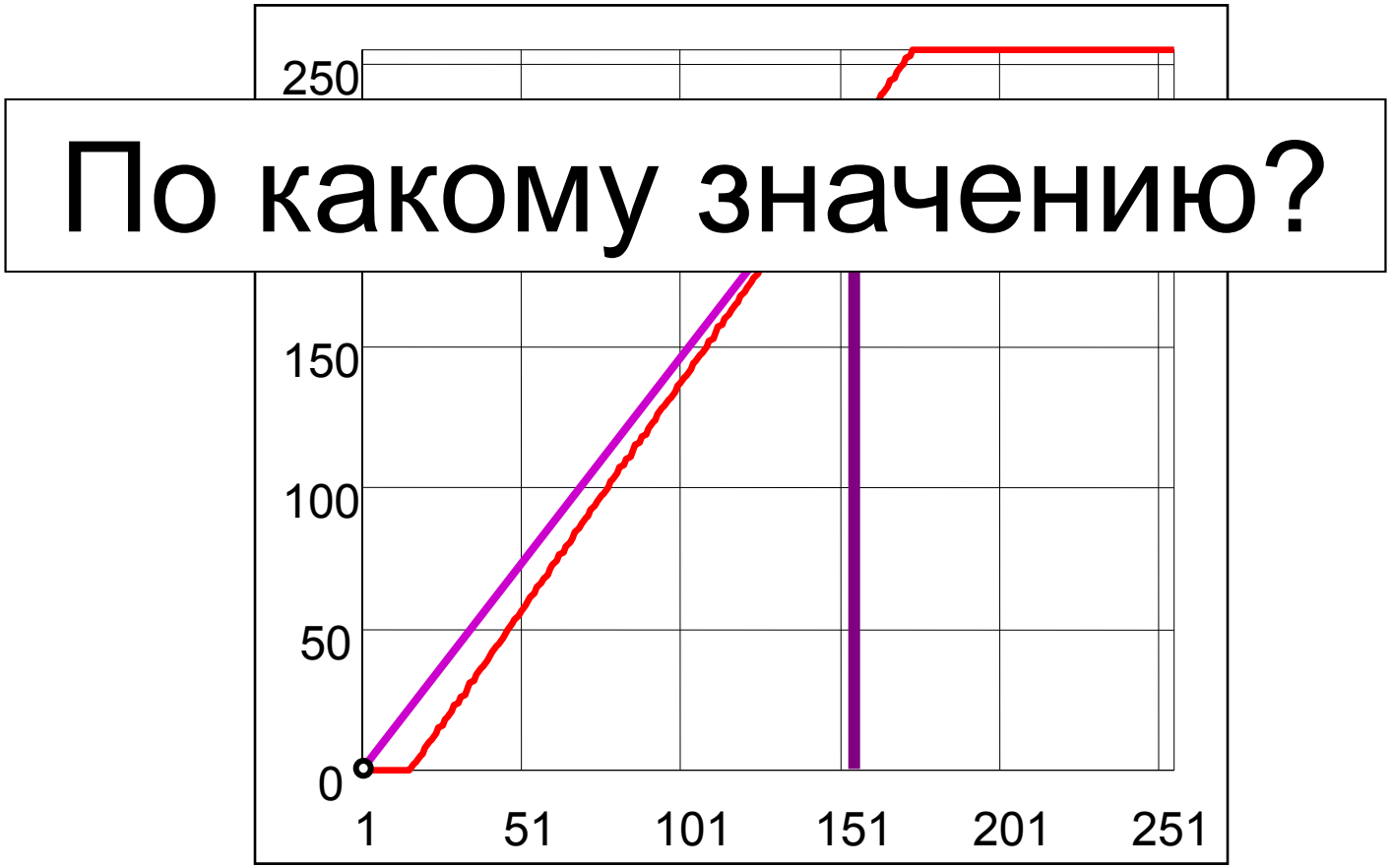
$$kC^{\gamma} = \left(k^{1/\gamma}\right)^{\gamma} C^{\gamma} = \left(k^{1/\gamma} C\right)^{\gamma}$$

- Не сохраняется цветность, поскольку R, G и B меняются в разной пропорции (в случае кривой Алгоритма 2).



Решение

- sRGB → Линейное sRGB → Вычисления → sRGB
- Все каналы преобразуются по одному значению аргумента кривой.



Варианты

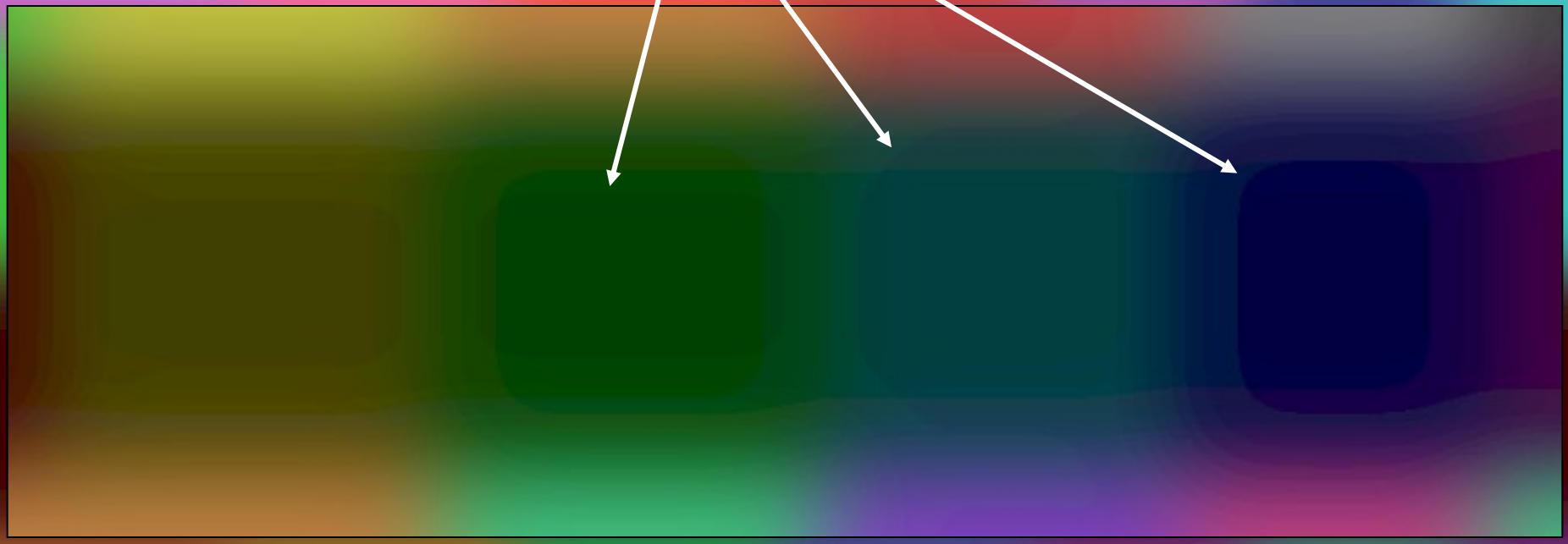
- R
- G
- B
- R+G+B
- (R+G+B)/3
- $\sqrt{R^2 + G^2 + B^2}$
- Max(R, G, B) √
- Y(R, G, B)

Алгоритм 9

1. sRGB → Линейное sRGB (кривая)
2. Построить гистограмму (по линейным значениям).
3. Найти уровни отсечения и построить кривую преобразования.
4. Преобразовать для каждой триады Max(R, G, B). Остальные два канала – в той же пропорции.
5. Линейное sRGB → sRGB (кривая).

Проблемы
Исходный

Постеризация из-за нескольких округлений (3 кривые)



Алгоритм 9

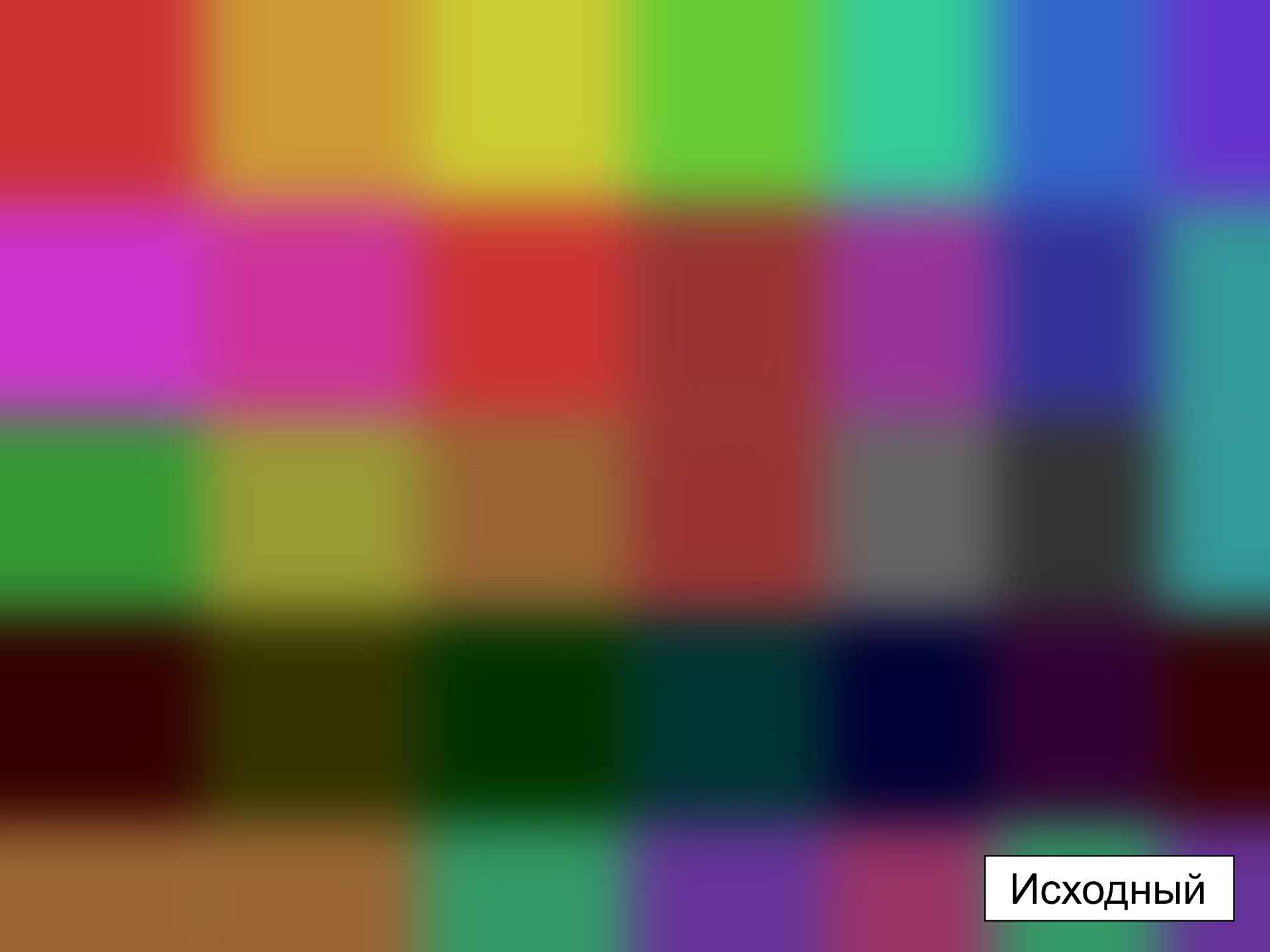
Алгоритм 10

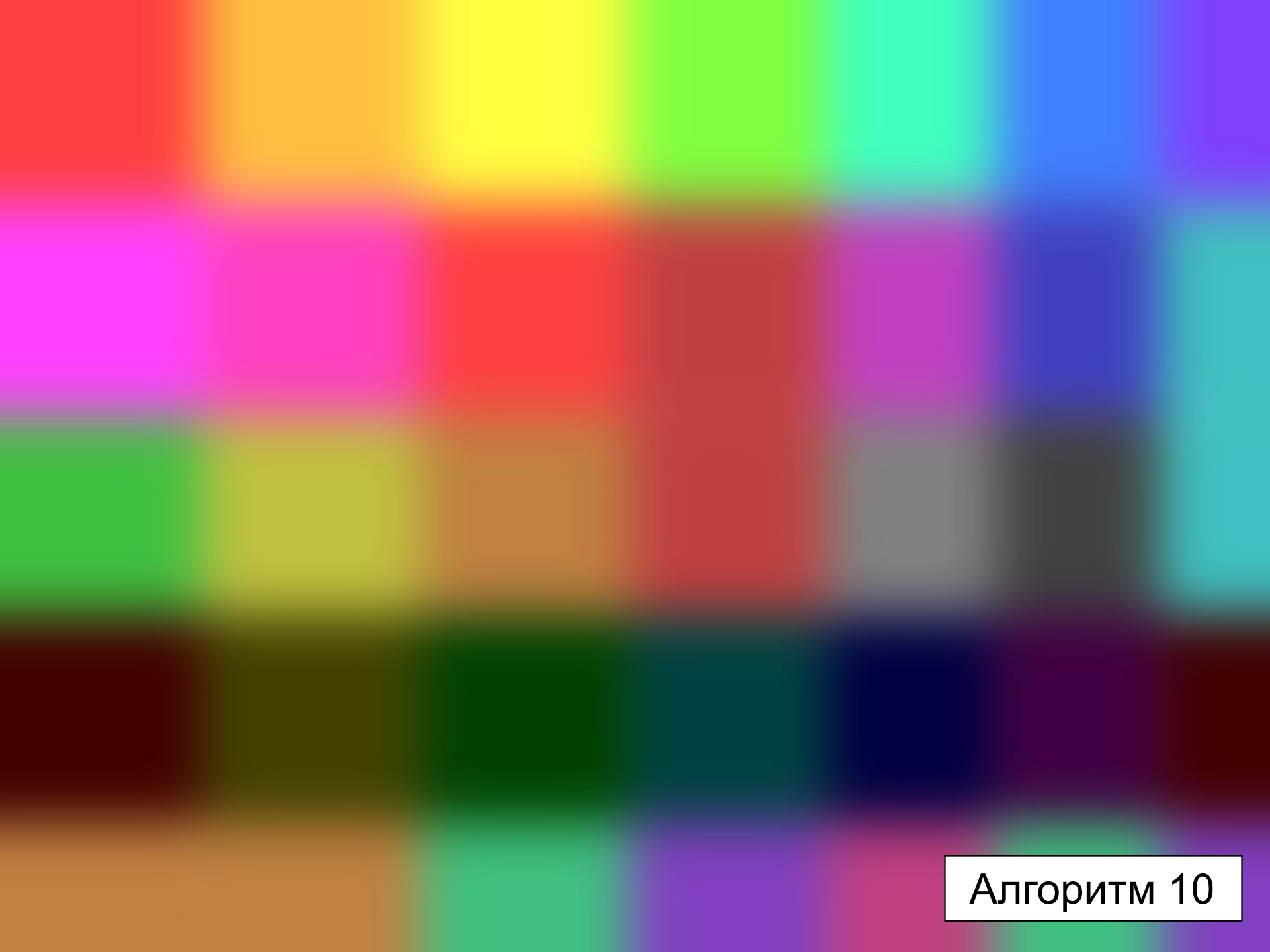
1. Построить гистограмму по линейным значениям, используя кривую sRGB → Линейное sRGB.

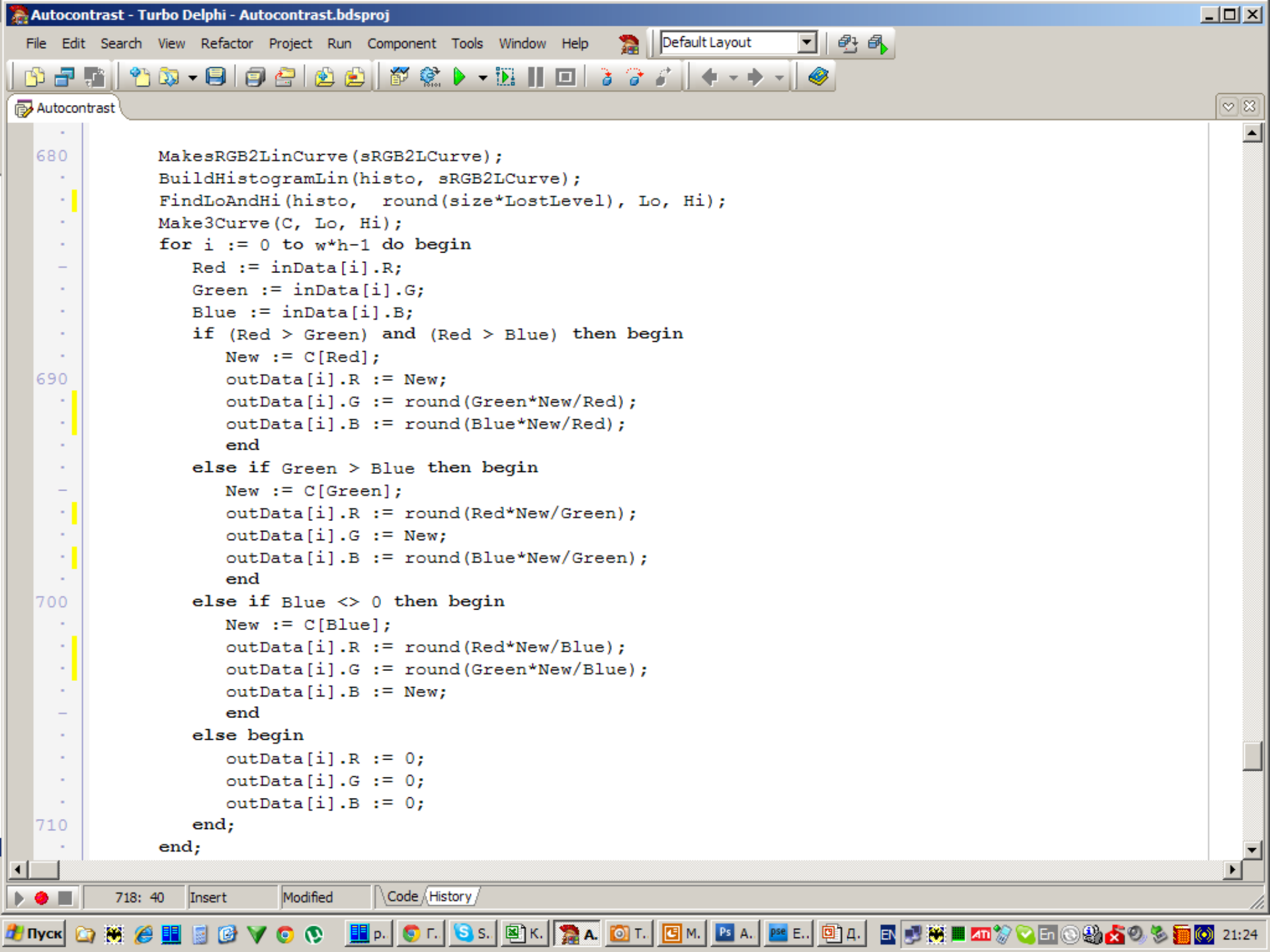
2. Найти уровни отсечения.

3. Построить единую кривую комплексного преобразования: sRGB → Линейное sRGB → Преобразование контраста → sRGB.

4. Преобразовать для каждой триады Max(R, G, B). Остальные два канала – в той же пропорции.


Исходный


Алгоритм 10


Autocontrast - Turbo Delphi - Autocontrast.bdsproj
File Edit Search View Refactor Project Run Component Tools Window Help
Default Layout
Autocontrast
680
690
700
710
MakesRGB2LinCurve(sRGB2LCurve);
BuildHistogramLin(histo, sRGB2LCurve);
FindLoAndHi(histo, round(size*LostLevel), Lo, Hi);
Make3Curve(C, Lo, Hi);
for i := 0 to w*h-1 do begin
Red := inData[i].R;
Green := inData[i].G;
Blue := inData[i].B;
if (Red > Green) and (Red > Blue) then begin
New := C[Red];
outData[i].R := New;
outData[i].G := round(Green*New/Red);
outData[i].B := round(Blue*New/Red);
end
else if Green > Blue then begin
New := C[Green];
outData[i].R := round(Red*New/Green);
outData[i].G := New;
outData[i].B := round(Blue*New/Green);
end
else if Blue <> 0 then begin
New := C[Blue];
outData[i].R := round(Red*New/Blue);
outData[i].G := round(Green*New/Blue);
outData[i].B := New;
end
else begin
outData[i].R := 0;
outData[i].G := 0;
outData[i].B := 0;
end;
end;
718: 40
Insert
Modified
Code
History
Пуск
21:24

Autocontrast - Turbo Delphi - Autocontrast.bdsproj

File Edit Search View Refactor Project Run Component Tools Window Help

Default Layout

Autocontrast

```
procedure Make3Curve(var curve: tCurve; MinRGB, MaxRGB: integer);
var
   i: integer;
   c: real;
begin
   for i:= 0 to MAX do begin
      c := i/MAX;
      if c<=0.04045 then
         c := c/12.92
      else
         c := Power((c+0.055)/1.055, 2.4);

      if c < MinRGB/MAX then
         c := 0
      else if c < MaxRGB/MAX then
         c := (c-MinRGB/MAX)/(MaxRGB/MAX - MinRGB/MAX)
      else
         c := 1.0;

      if c<=0.0031308 then
         c := 12.92*c
      else
         c := 1.055*Power(c, 1/2.4)-0.055;
      curve[i] := round(MAX*c);
   end;

end;
```

105: 13 Insert Modified Code History

Пуск 20:53

Исходный

Алгоритм 10

Adobe Photoshop Автоконтраст

Исходный

Adobe Photoshop Автоконтраст

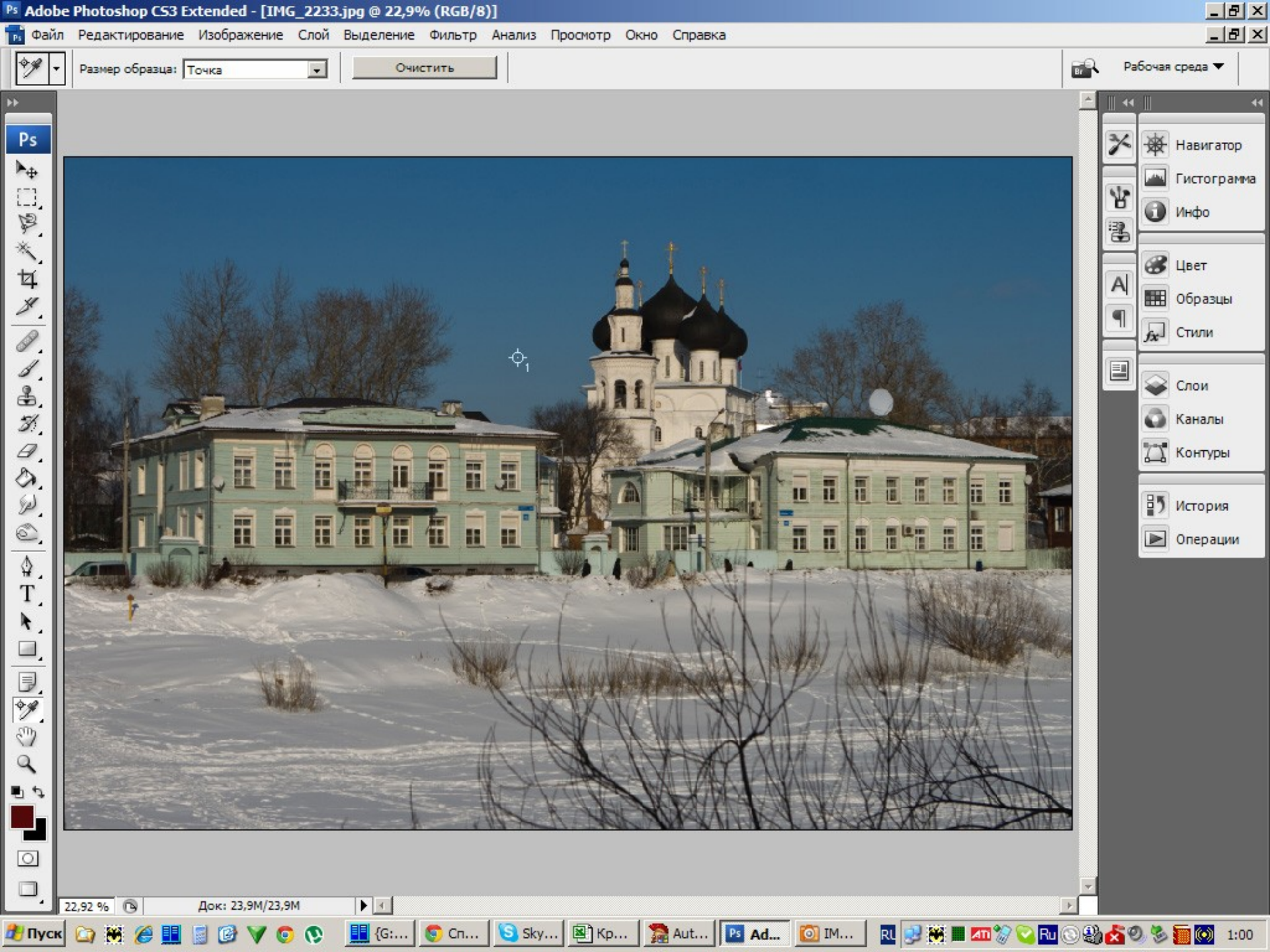
Исходный

Алгоритм 10

Исходный

Adobe Photoshop Автоконтраст

Алгоритм 10


Adobe Photoshop CS3 Extended - [IMG_2233.jpg @ 22,9% (RGB/8)]
Файл Редактирование Изображение Слой Выделение Фильтр Анализ Просмотр Окно Справка
Размер образца: Точка
Очистить
Рабочая среда
Навигатор
Гистограмма
Инфо
Цвет
Образцы
Стили
Слои
Каналы
Контуры
История
Операции
22,92 %
Док: 23,9M/23,9M
Пуск
1:00

Цветность контрольной точки

	R	G	B
Исходный	66	101	121
Алгоритм 10	80	123	147
Adobe Photoshop	70	115	141
	r	g	b
Исходный	0.229	0.351	0.420
Алгоритм 10	0.229	0.351	0.420
Adobe Photoshop	0.215	0.353	0.433

Базовые (линейные) цветовые преобразования

Базовые (линейные) цветовые преобразования

Все вычисления в линейном (RGB) пространстве

1. **Изменение экспозиции.** Увеличение или уменьшение количества световой энергии при неизменном спектре. Изменяется светлота. Не изменяются тон и насыщенность (saturation).

2. **Удаление вуали (паразитной засветки).** Вычитание постоянной составляющей (установка уровня черного). Меняются светлота и насыщенность при неизменном цветовом тоне.

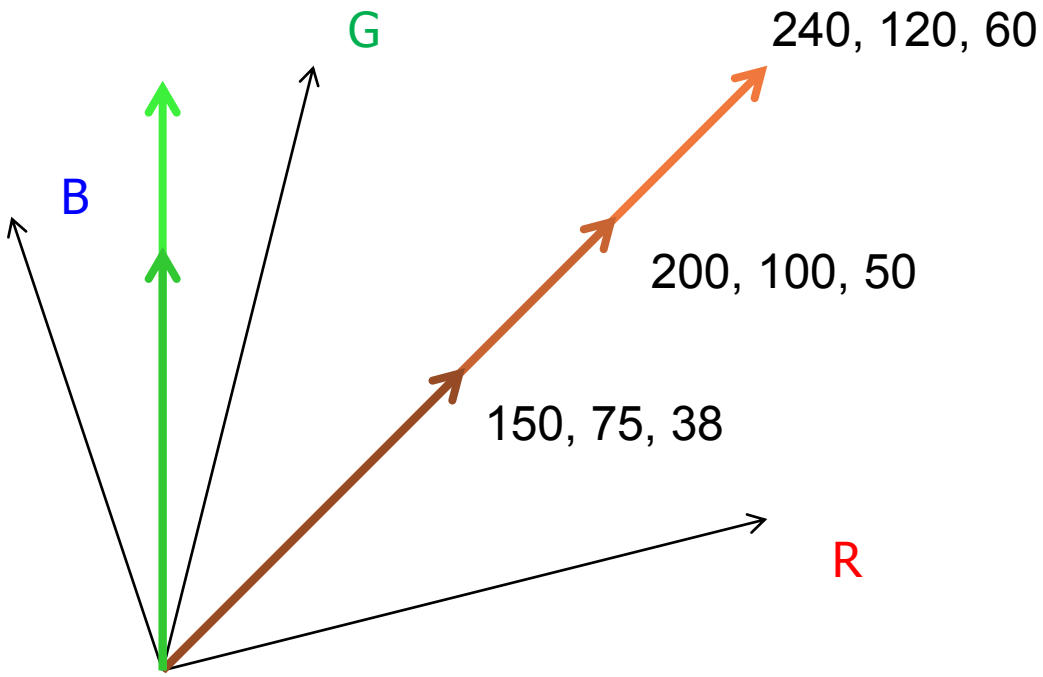
3. **Цветовой баланс (баланс белого).** Изменение цветового тона и насыщенности с помощью преобразований, имеющих физический смысл.

4. **Изменение насыщенности**. Увеличение или уменьшение насыщенности (saturation, chroma) при неизменном тоне и светлоте.

5. **Изменение светлоты.** Сохраняются тон и насыщенность (chroma).

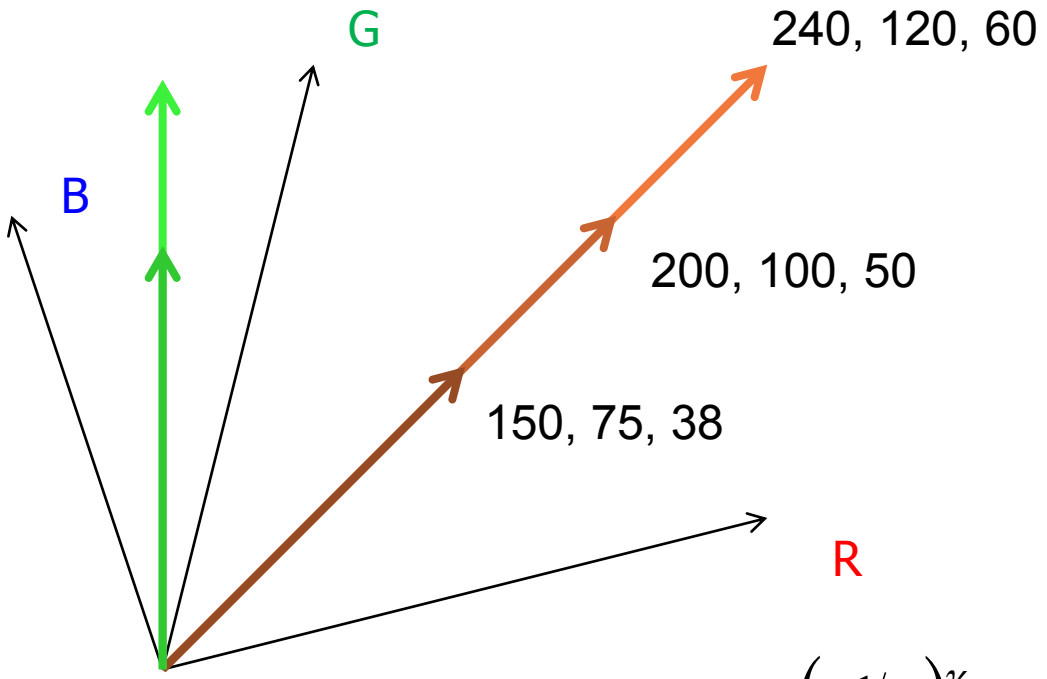
6. **Сдвиг тона.** Изменение (циклическое) цветового тона при неизменных насыщенности (chroma, saturation) и светлоты.

Экспозиция



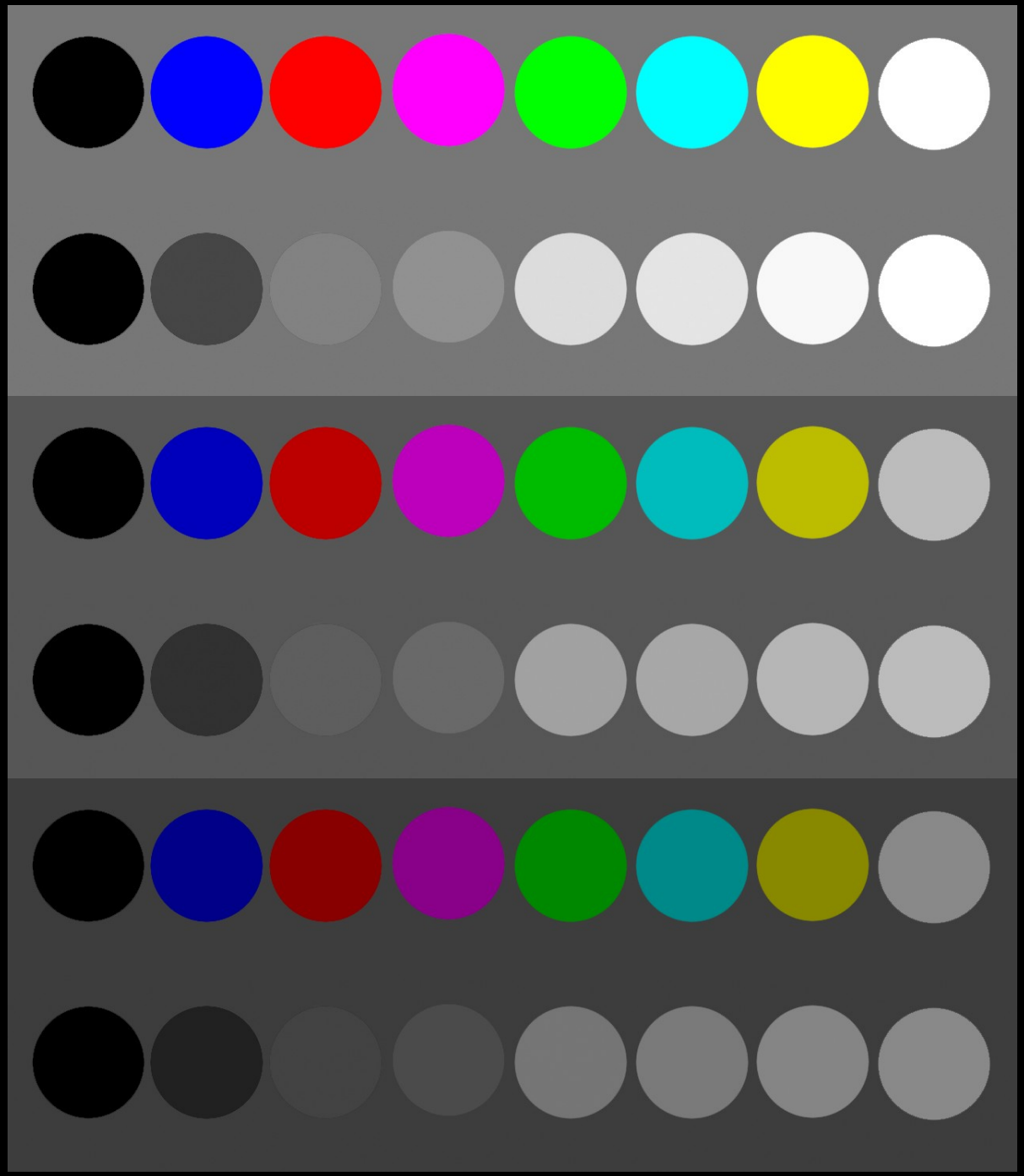
- Цветовой вектор меняется по длине, но не по направлению
- Умножение RGB-координат на одно и то же число в линейном цветовом пространстве

Экспозиция



$$kC^{\gamma} = \left(k^{1/\gamma}\right)^{\gamma} C^{\gamma} = \left(k^{1/\gamma} C\right)^{\gamma}$$

- Цветовой вектор меняется по длине, но не по направлению
- Умножение RGB-координат на одно и то же число в линейном цветовом пространстве (в порядке исключения ☺ гамма-корректированном)
- 1EV ~ умножение на 2 в линейном пространстве (1.37 при γ=2.2)



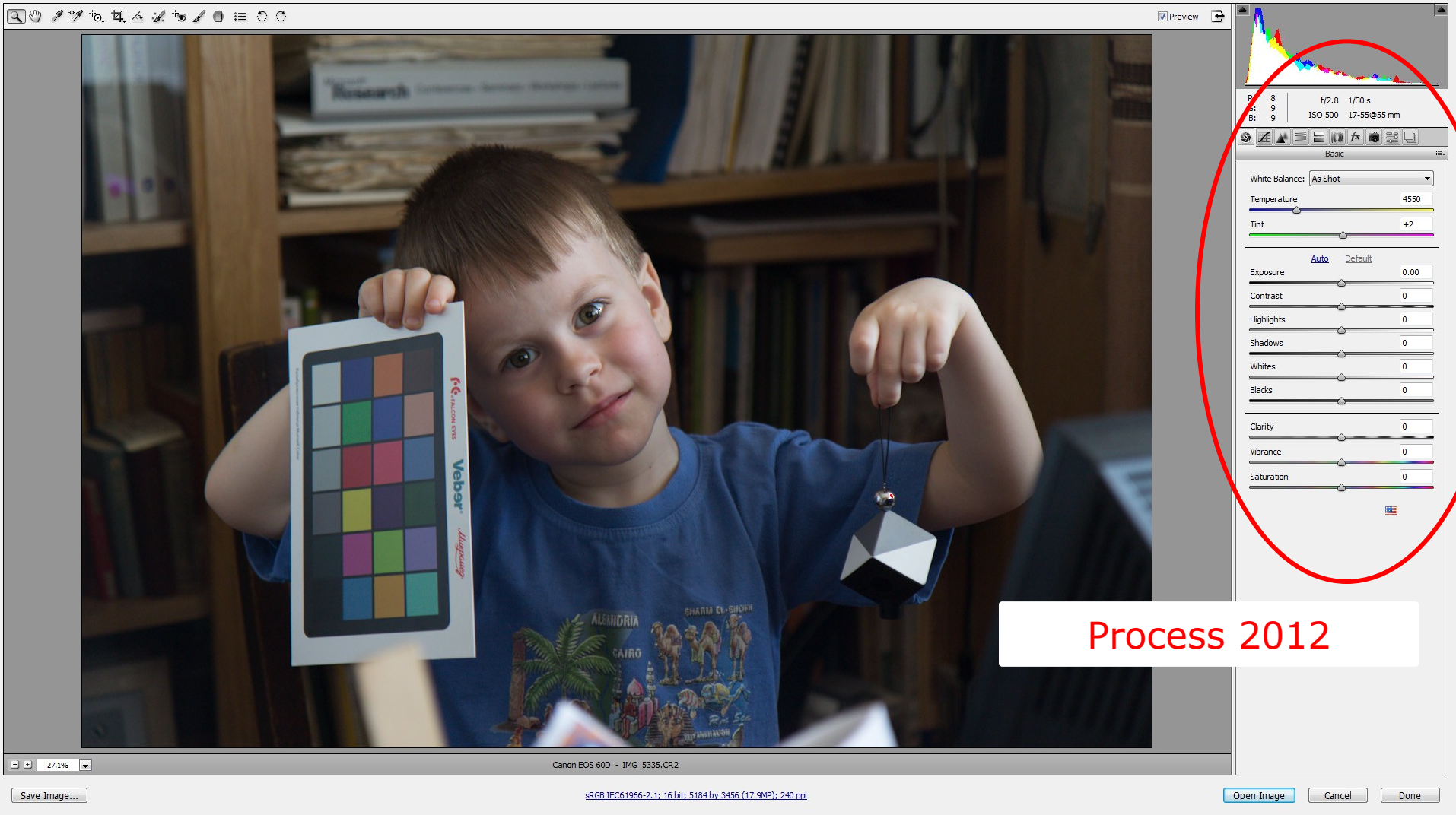
-1EV

-2EV

Adobe Camera Raw 7.0

Как снято

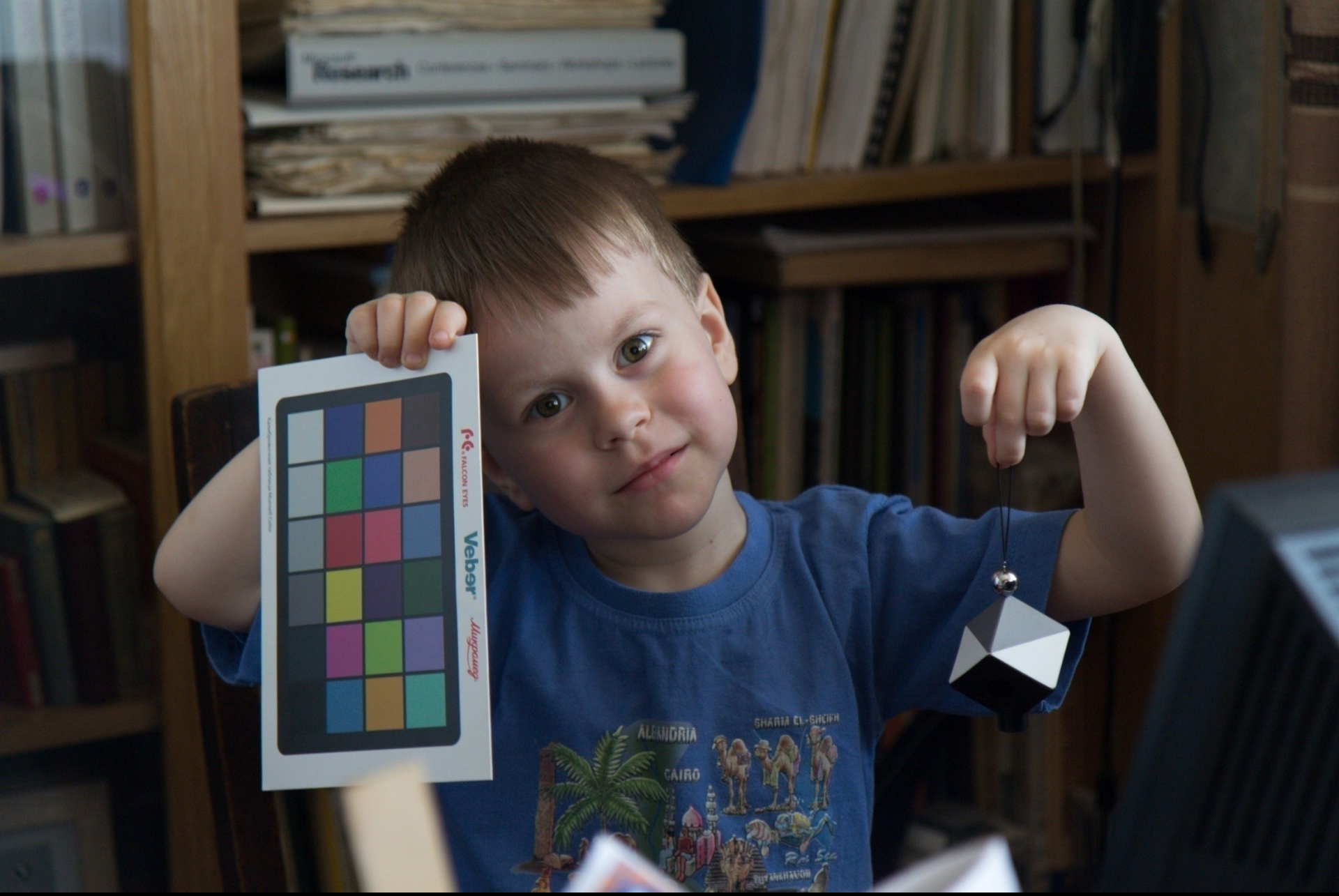
Canon 60D, Canon EF-S/2.8 17-55IS, f/2.8, 1:30, ISO 500, ФР=55мм (ЭФР 88мм)



Adobe Camera Raw 7.0

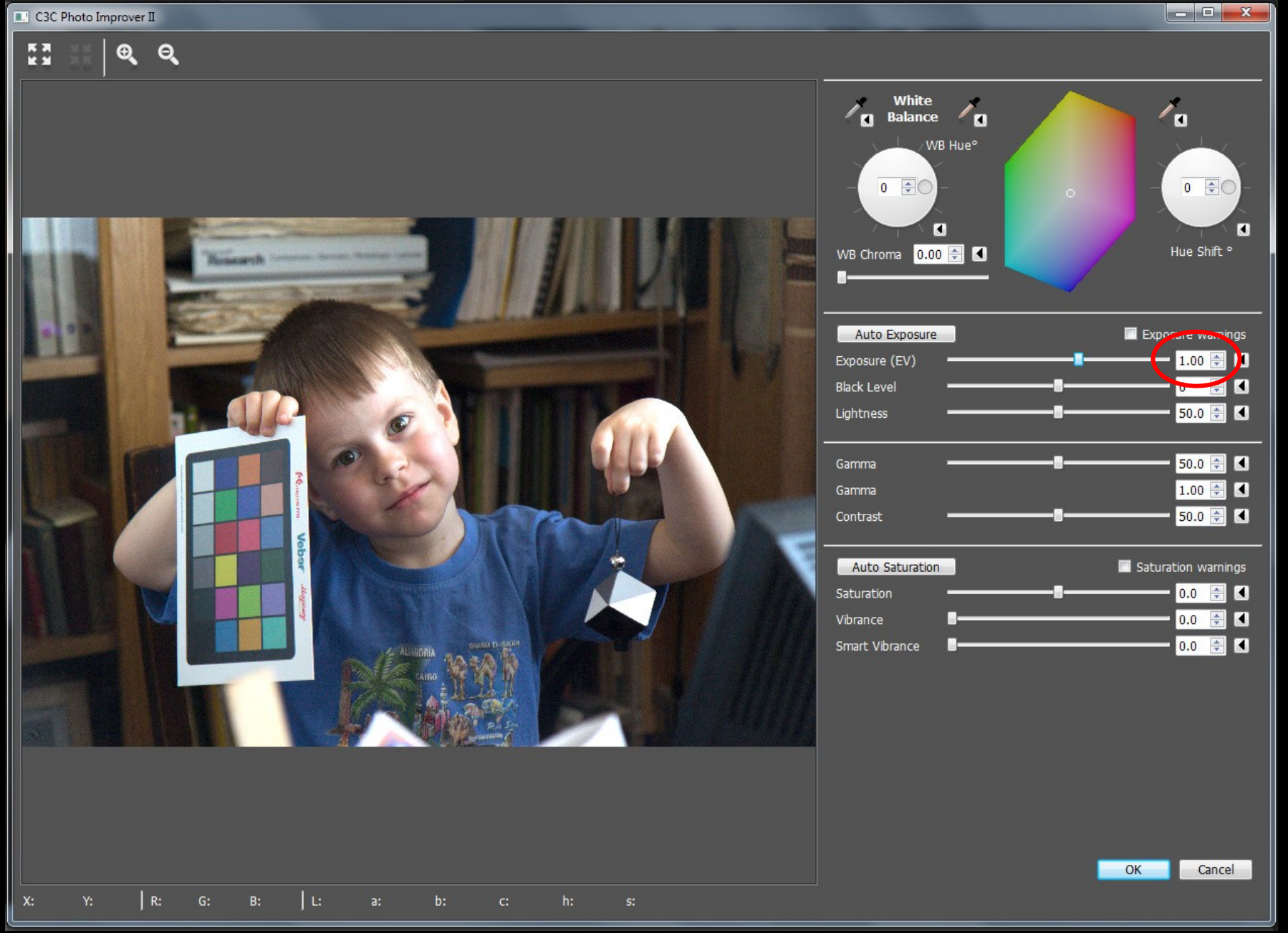
Как снято

Canon 60D, Canon EF-S/2.8 17-55IS, f/2.8, 1:30, ISO 500, ФР=55мм (ЭФР 88мм)

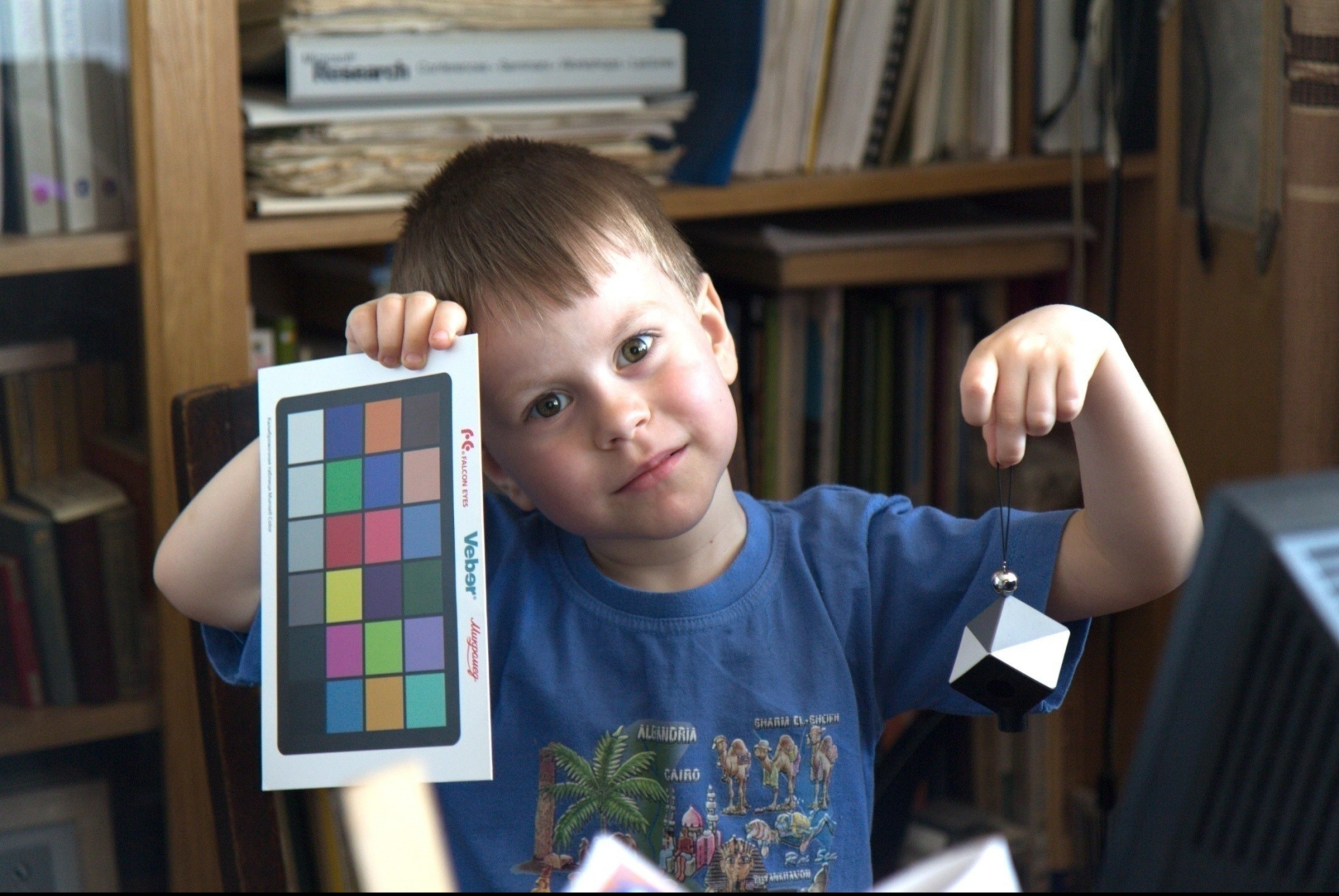


Как снято

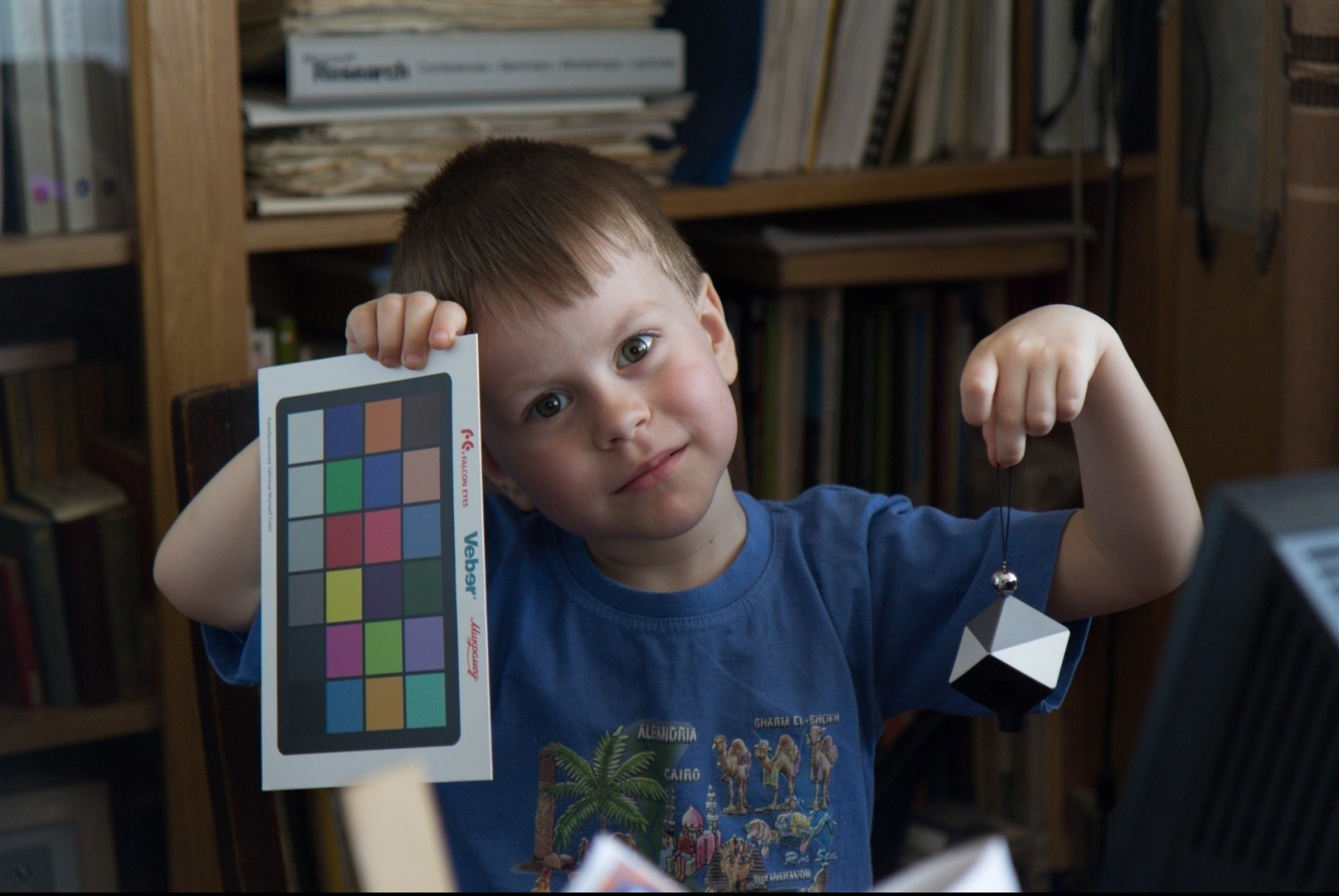
Canon 60D, Canon EF-S/2.8 17-55IS, f/2.8, 1:30, ISO 500, ФР=55мм (ЭФР 88мм)



C3C Color Wizard 0.22
+1EV



+1EV



Как снято

Canon 60D, Canon EF-S/2.8 17-55IS, f/2.8, 1:30, ISO 500, ФР=55мм (ЭФР 88мм)

-1EV

-2EV

C3C Photo Improver II

White Balance

WB Hue°

0

WB Chroma 0.00

0

Hue Shift °

Auto Exposure

Exposure warnings

Exposure (EV) 0.27

Black Level 0

Lightness 50.0

Gamma 50.0

Gamma 1.00

Contrast 50.0

Auto Saturation

Saturation warnings

Saturation 0.0

Vibrance 0.0

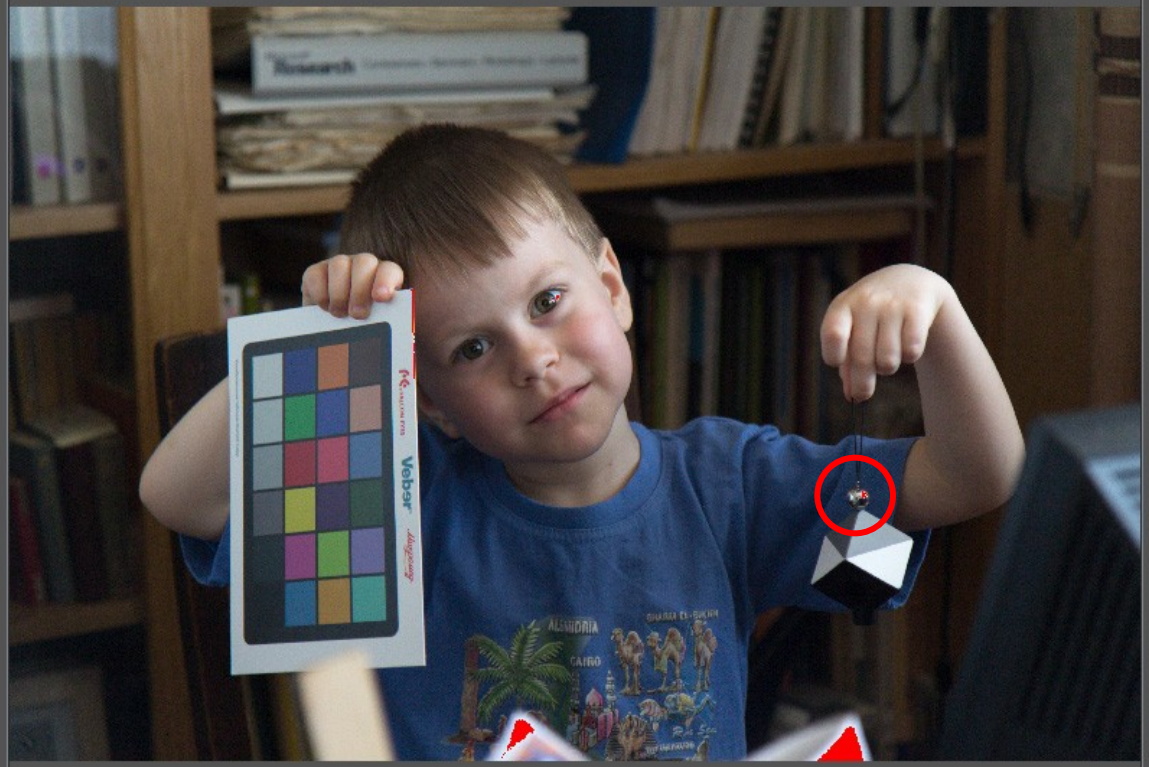
Smart Vibrance 0.0

OK

Cancel

X: Y: | R: G: B: | L: a: b: c: h: s:

C3C Photo Improver II



White Balance

WB Hue°

0

WB Chroma 0.00

Hue Shift °

0

Auto Exposure

Exposure warnings

Exposure (EV) 0.27

Black Level 0

Lightness 50.0

Gamma 50.0

Gamma 1.00

Contrast 50.0

Auto Saturation

Saturation warnings

Saturation 0.0

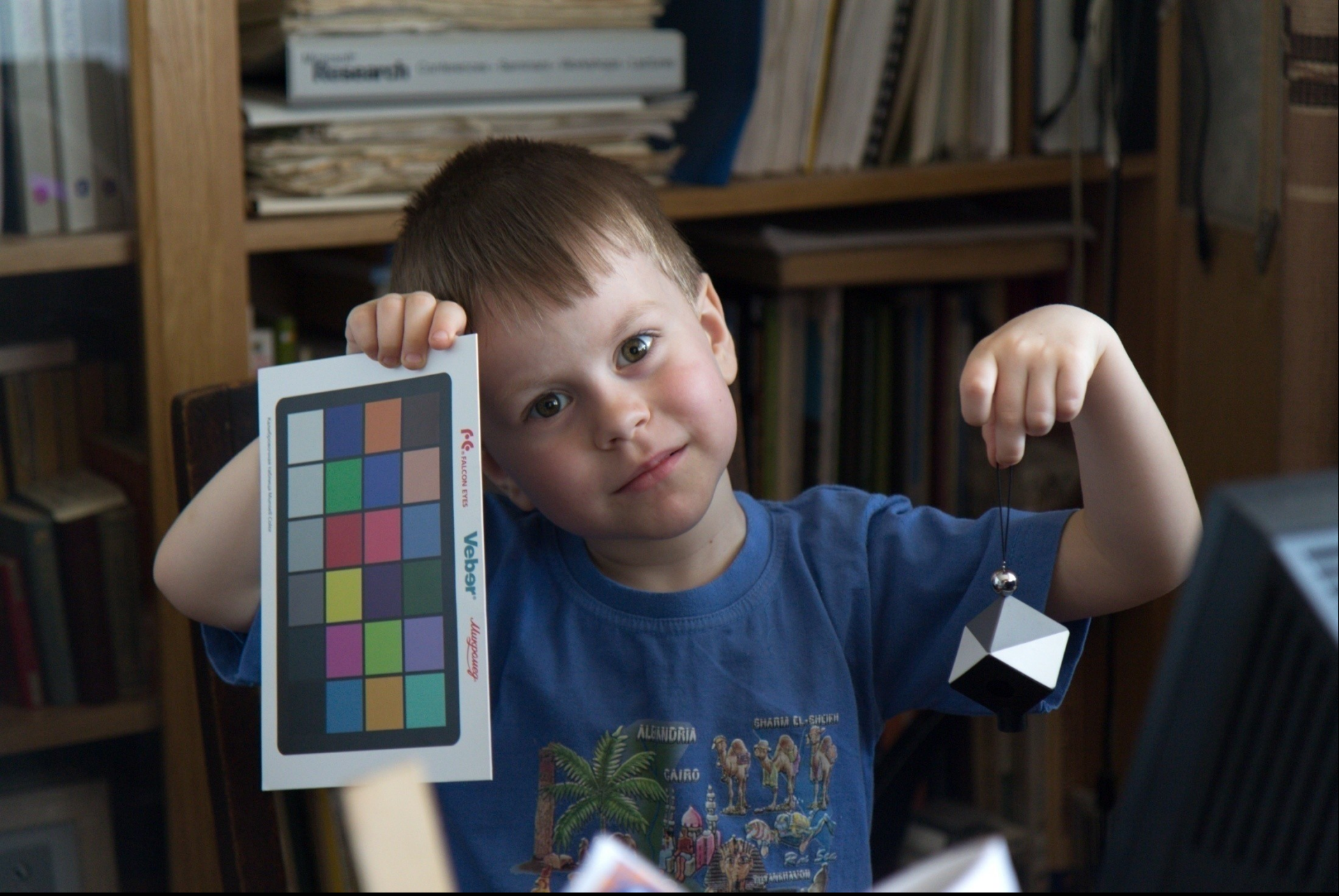
Vibrance 0.0

Smart Vibrance 0.0

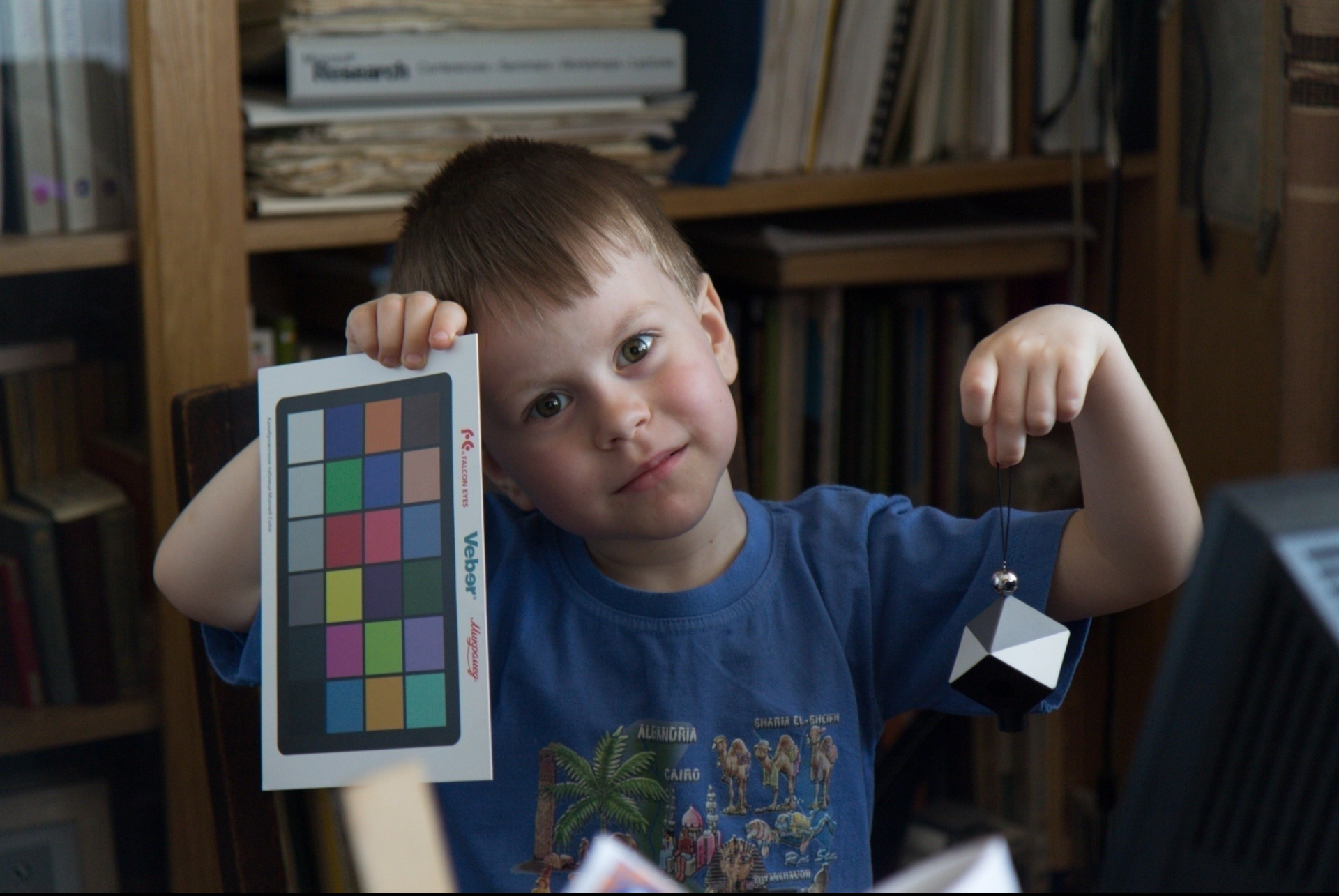
OK

Cancel

X: Y: | R: G: B: | L: a: b: c: h: s:



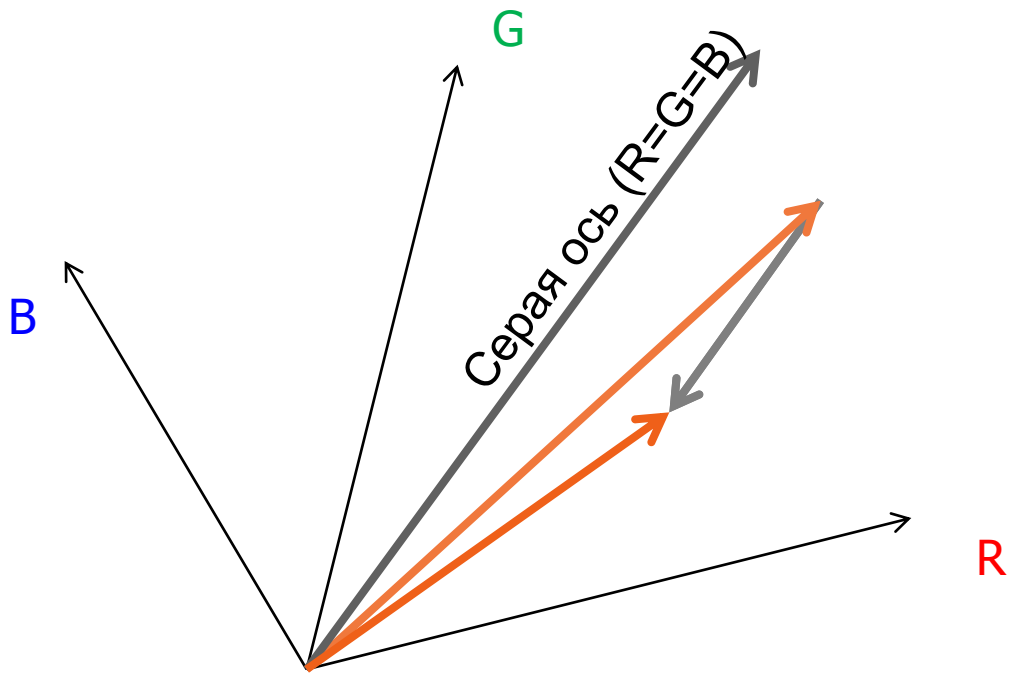
+0.27EV



Как снято

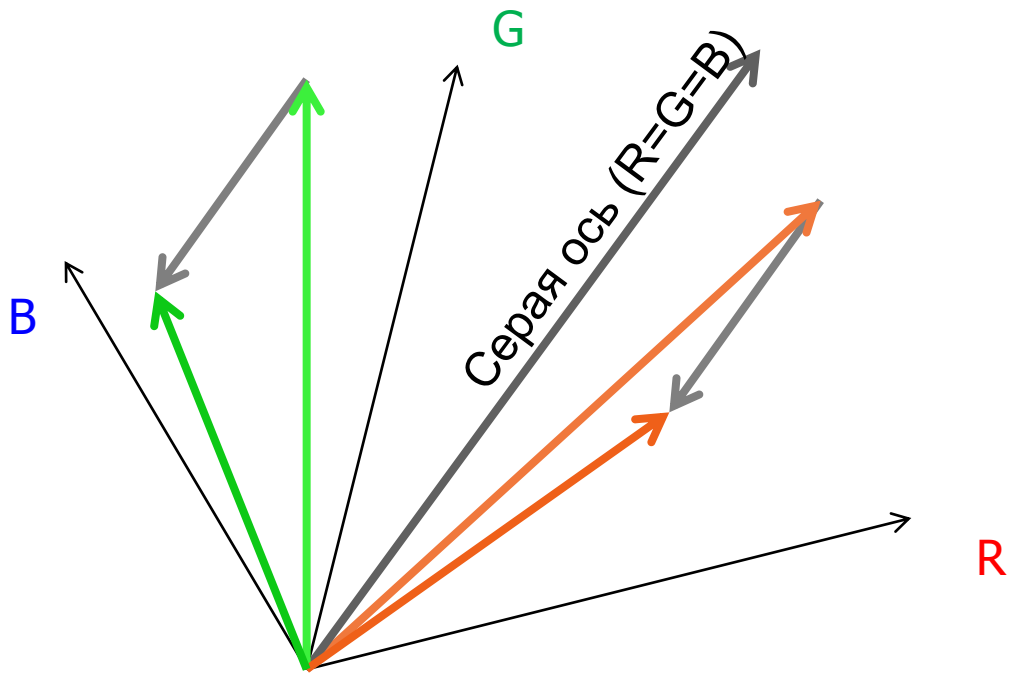
Canon 60D, Canon EF-S/2.8 17-55IS, f/2.8, 1:30, ISO 500, ФР=55мм (ЭФР 88мм)

Удаление вуали (уровень черного)



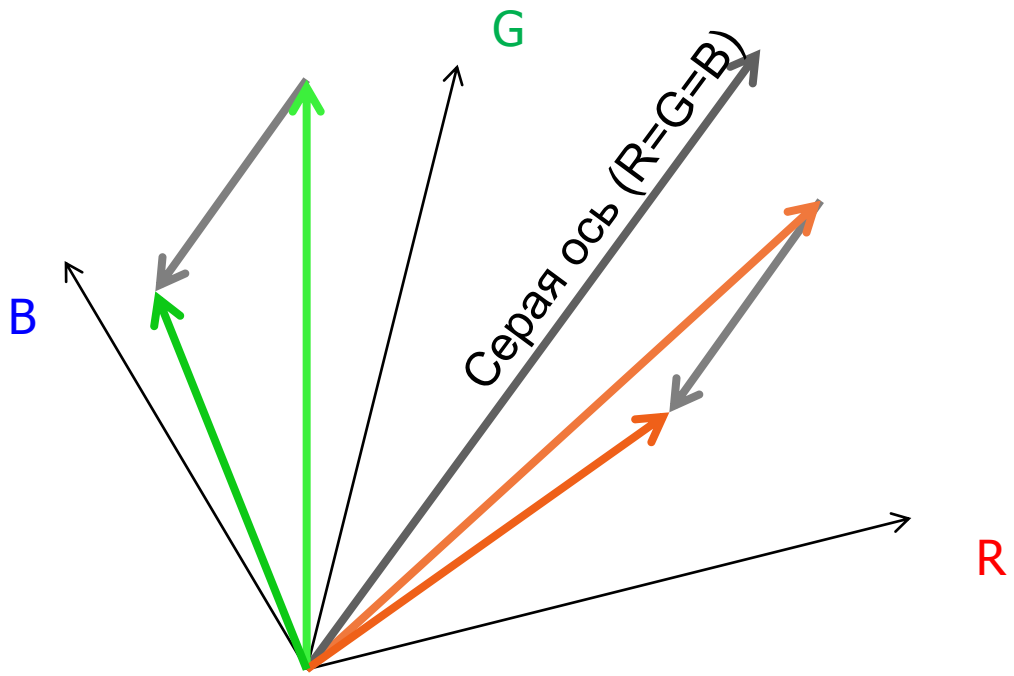
❖ RGB (линейные) изменяются НА одну и ту же величину, то есть вычитается (добавляется) ахроматическая (серая) компонента

Удаление вуали (уровень черного)



❖ RGB (линейные) изменяются НА одну и ту же величину, то есть вычитается (добавляется) ахроматическая (серая) компонента

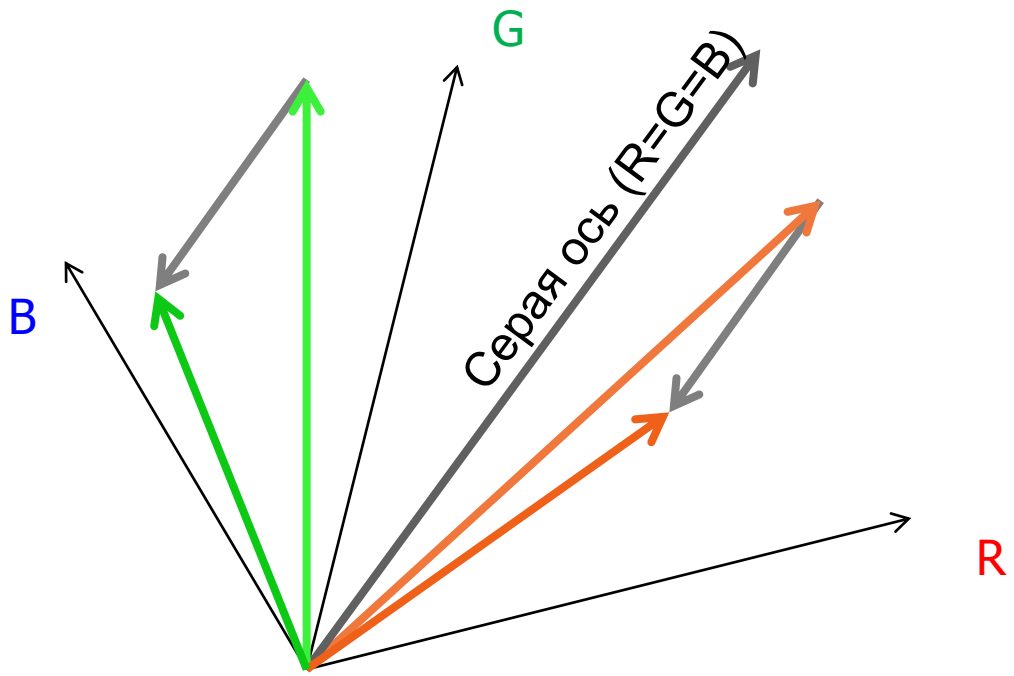
Удаление вуали (уровень черного)



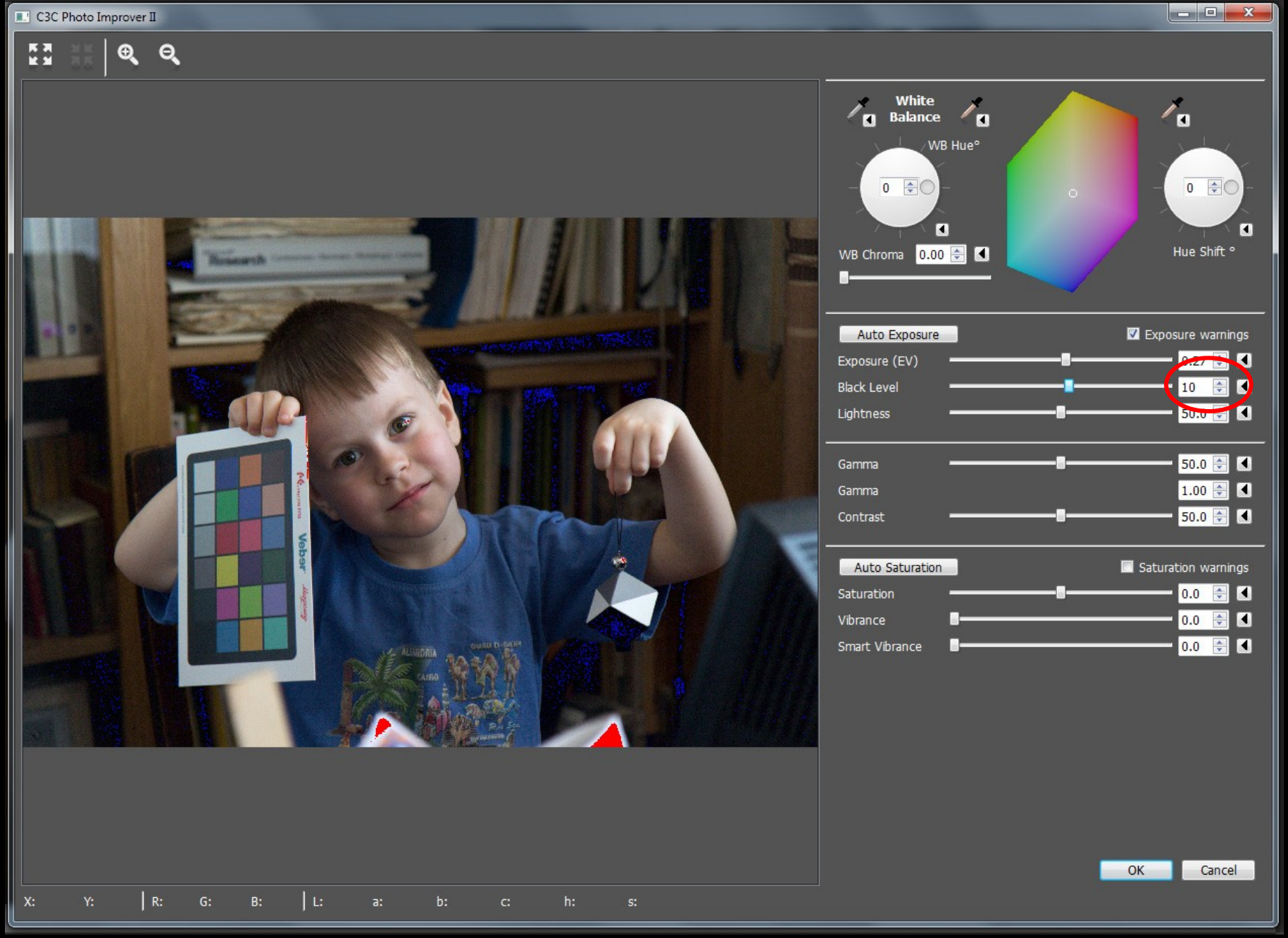
- RGB (линейные) изменяются НА одну и ту же величину, то есть вычитается (добавляется) ахроматическая (серая) компонента

- Меняются светлота и насыщенность (saturation), не меняется цветовой тон
  - Вычитание – рост насыщенности, уменьшение светлоты (вуаль уменьшается)
  - Добавление – уменьшение насыщенности, увеличение светлоты (вуаль увеличивается)

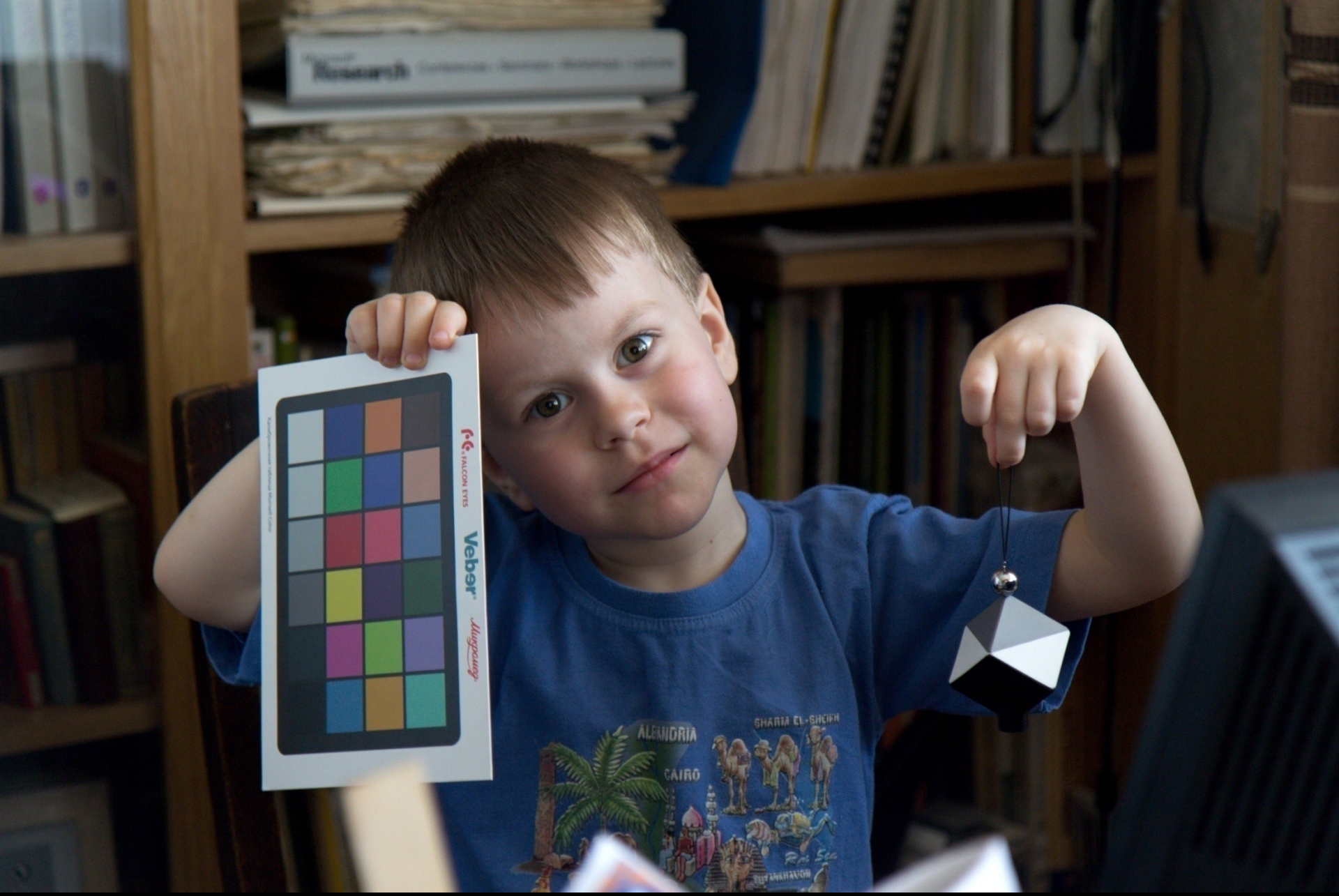
Удаление вуали (уровень черного)



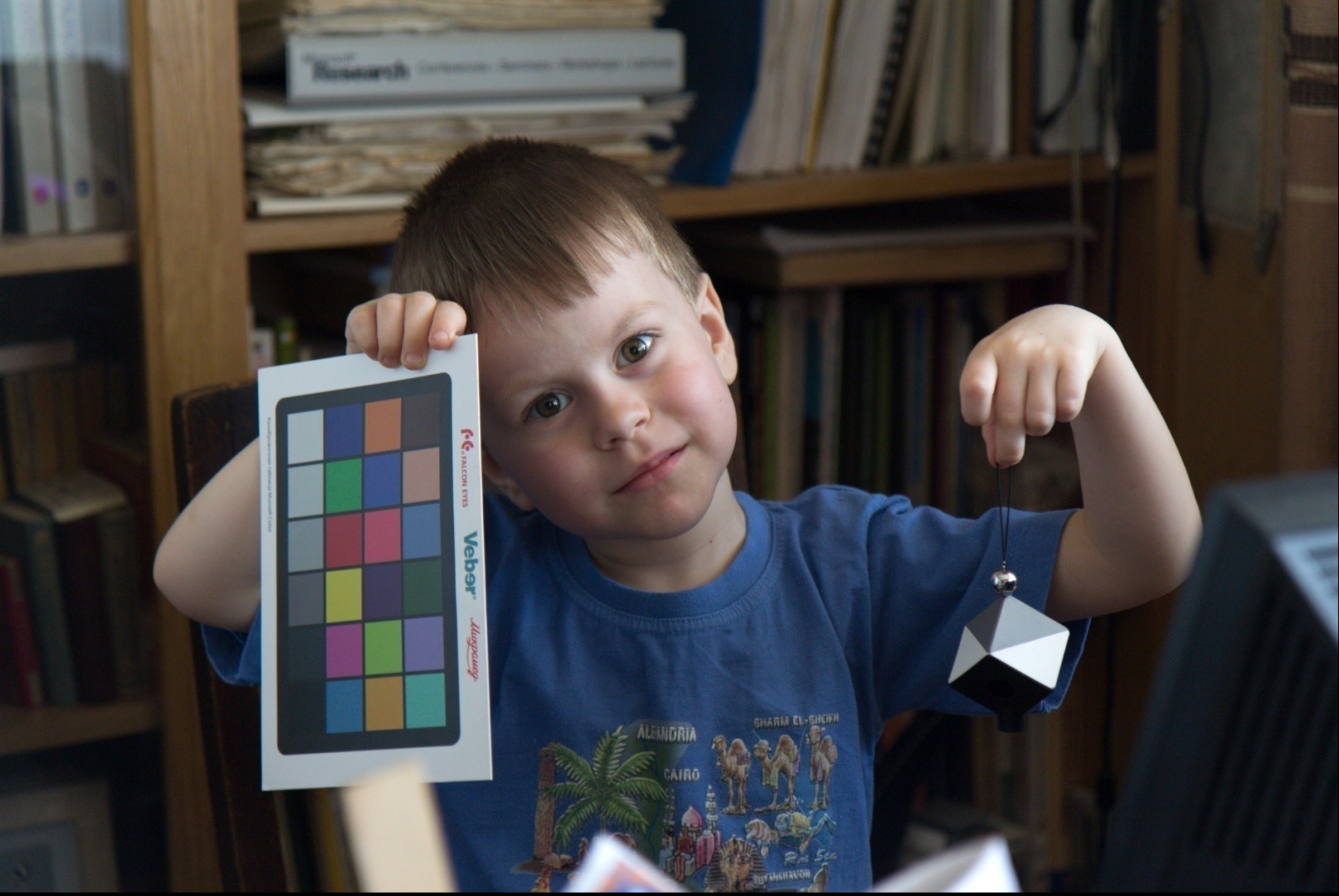
- RGB (линейные) изменяются НА одну и ту же величину, то есть вычитается (добавляется) ахроматическая (серая) компонента
- Меняются светлота и насыщенность (saturation), не меняется цветовой тон
  - Вычитание – рост насыщенности, уменьшение светлоты (вуаль уменьшается)
  - Добавление – уменьшение насыщенности, увеличение светлоты (вуаль увеличивается)
- Вычисления в нелинейном пространстве некорректны



Уровень черного 10 (10/255 в sRGB)



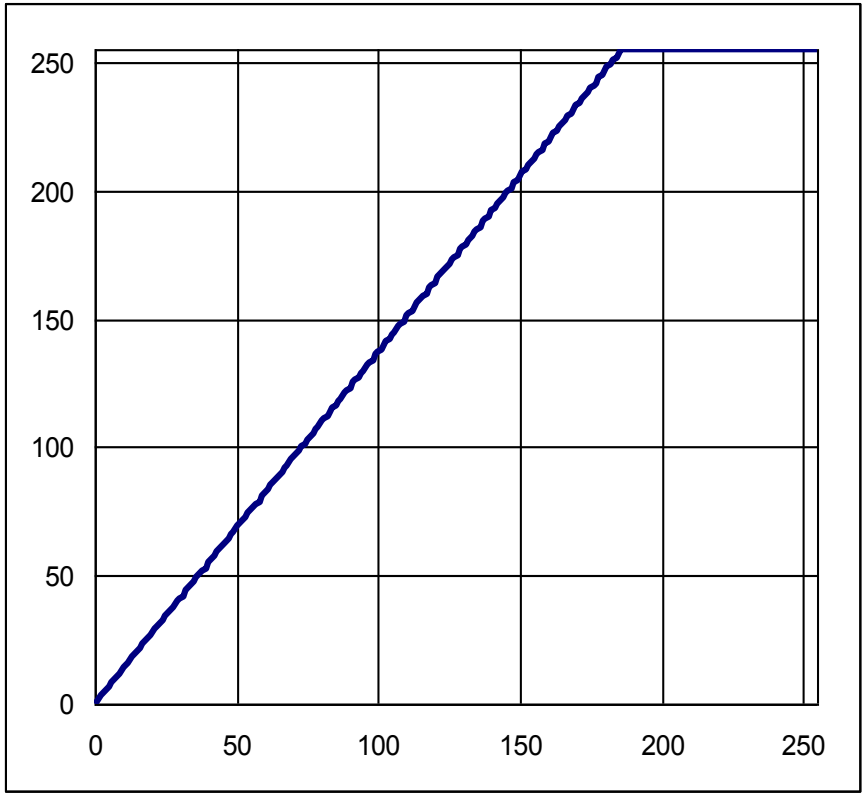
+0.27EV, уровень черного 10

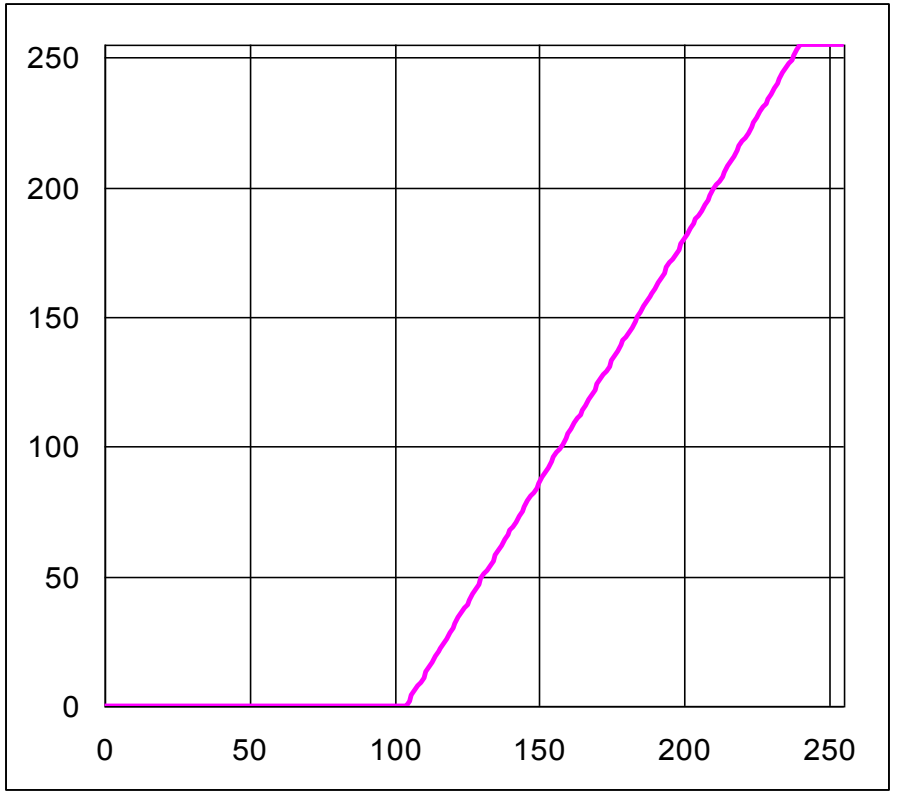


+0.27EV

Общий контраст
(экспозиция + уровень черного)

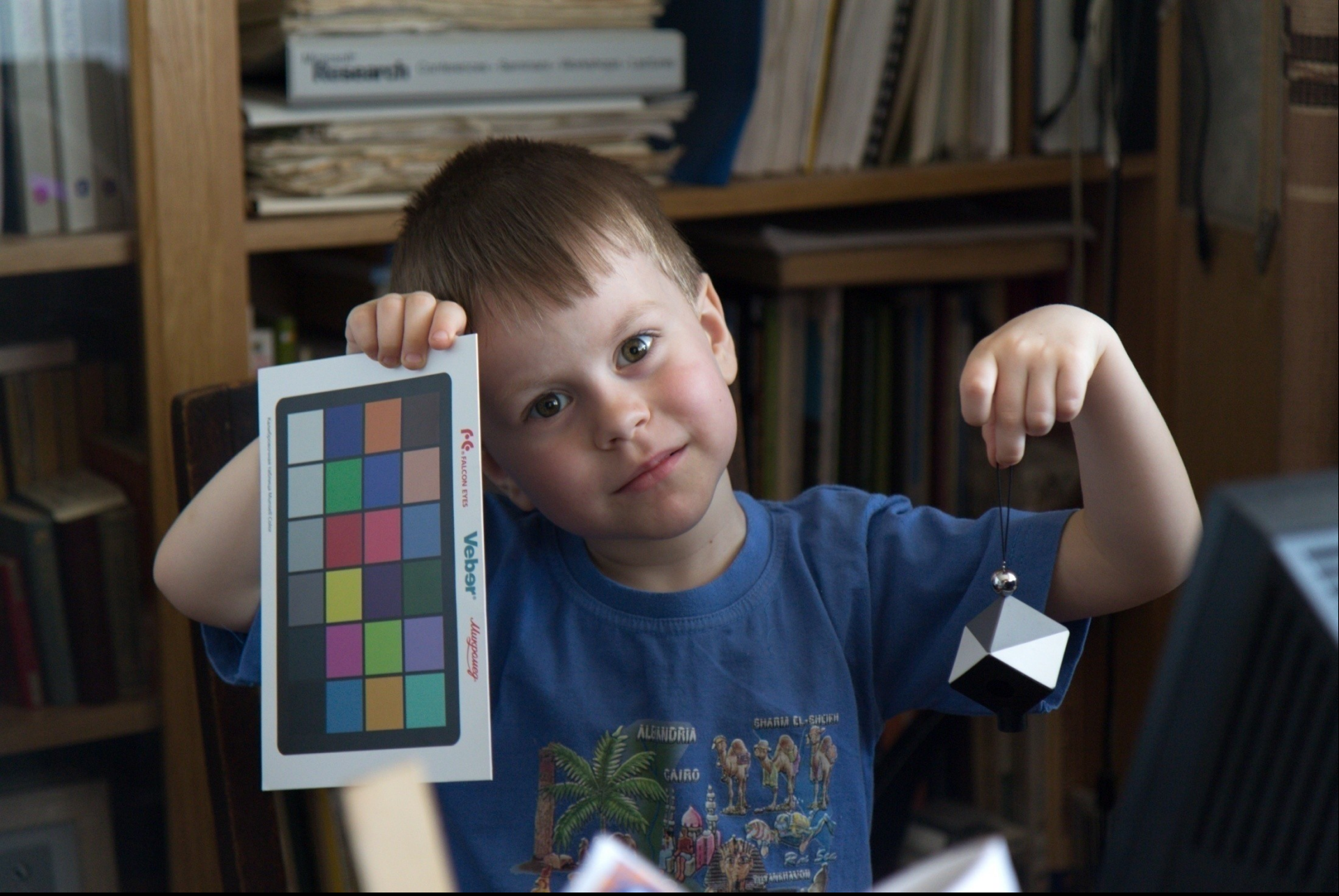
Кривые



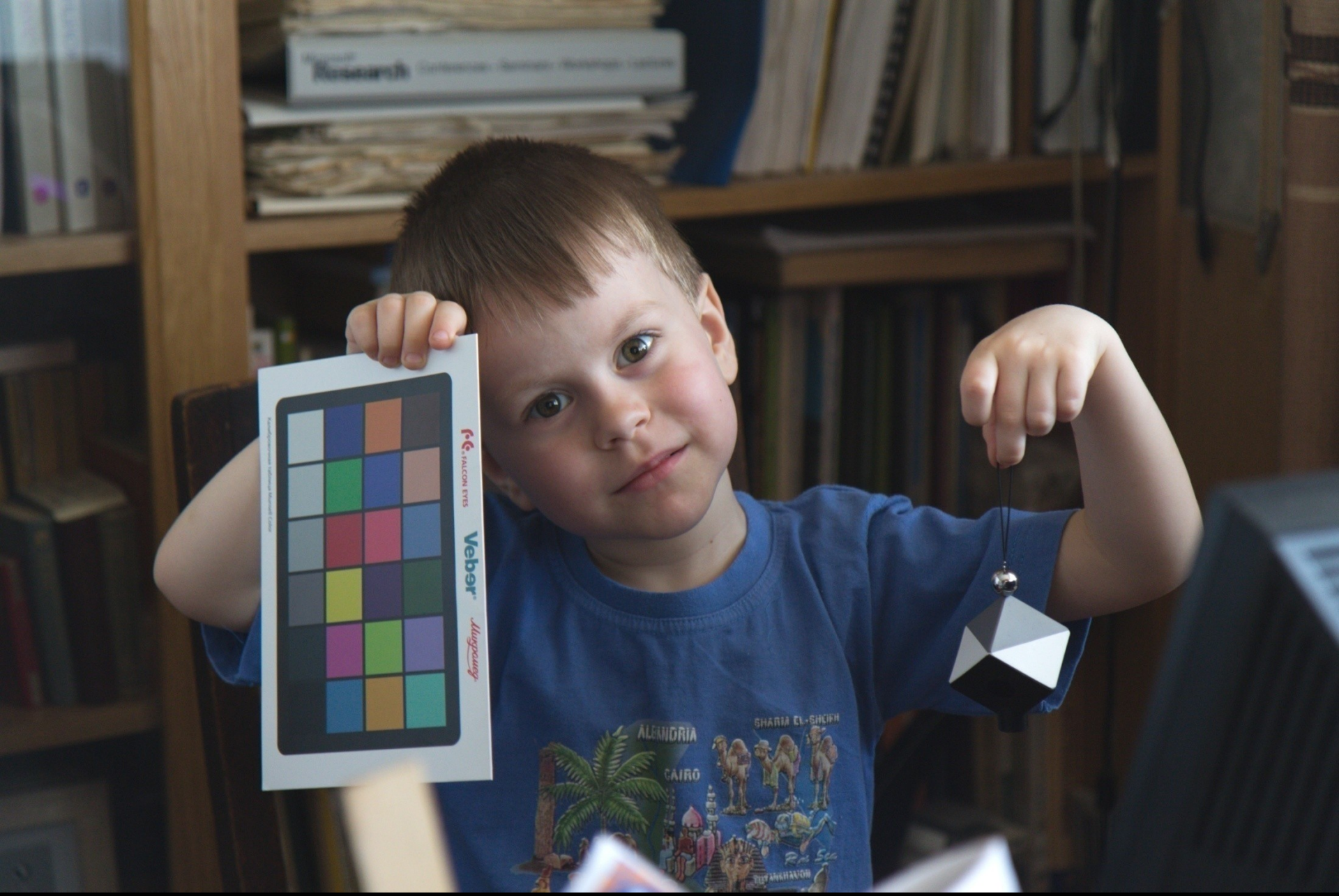


Экспозиция (уровень белого)

Уровни белого и черного



+0.27EV (сейчас будет вуаль)



Вуаль. Уровень черного -20

Снимок с вуалью

Источник: форум ixbt.com

Уровень черного 45

Цветовой баланс (баланс белого)

Цветовой баланс (баланс белого)

- Задача не имеет однозначного решения
  - Спектральная информация не сохраняется
  - Метамерия
  - …

Цветовой баланс (баланс белого)

- Задача не имеет однозначного решения
  - Спектральная информация не сохраняется
  - Метамерия
  - …
- Цели
  - Компенсация особенностей освещения
  - Компенсация «ошибок» внутрикамерной обработки

Цветовой баланс (баланс белого)

- Задача не имеет однозначного решения
  - Спектральная информация не сохраняется
  - Метамерия
  - …
- Цели
  - Компенсация особенностей освещения
  - Компенсация «ошибок» внутрикамерной обработки
- Критерии
  - Серое должно быть серым
  - Памятные цвета (телесный, …) должны быть правильными.
  - Чтобы нравилось

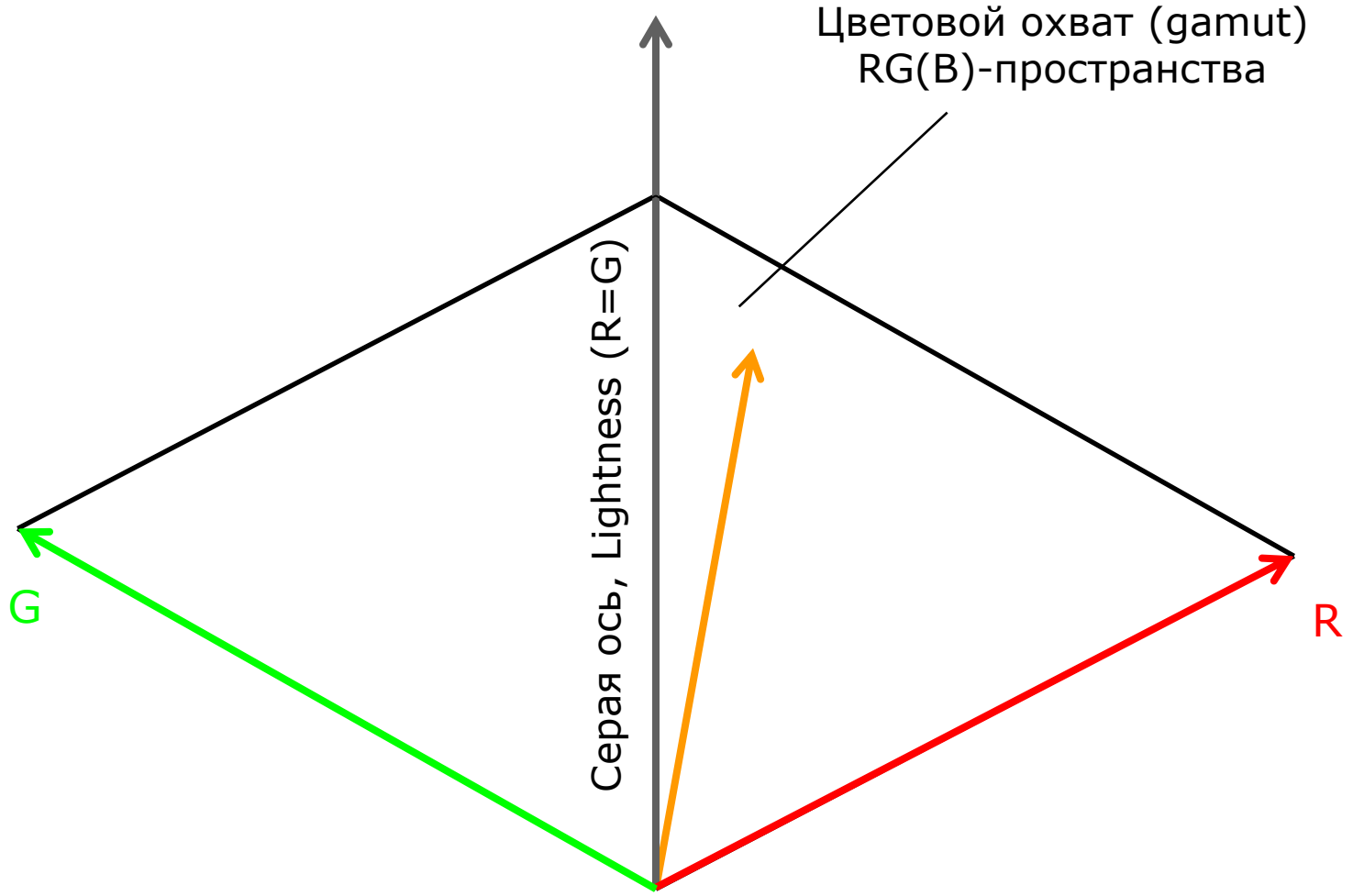
Цветовой баланс (баланс белого)

- Задача не имеет однозначного решения
  - Спектральная информация не сохраняется
  - Метамерия
  - …
- Цели
  - Компенсация особенностей освещения
  - Компенсация «ошибок» внутрикамерной обработки
- Критерии
  - Серое должно быть серым
  - Памятные цвета (телесный, …) должны быть правильными.
  - Чтобы нравилось
- Алгоритмы (преобразования в цветовом пространстве)

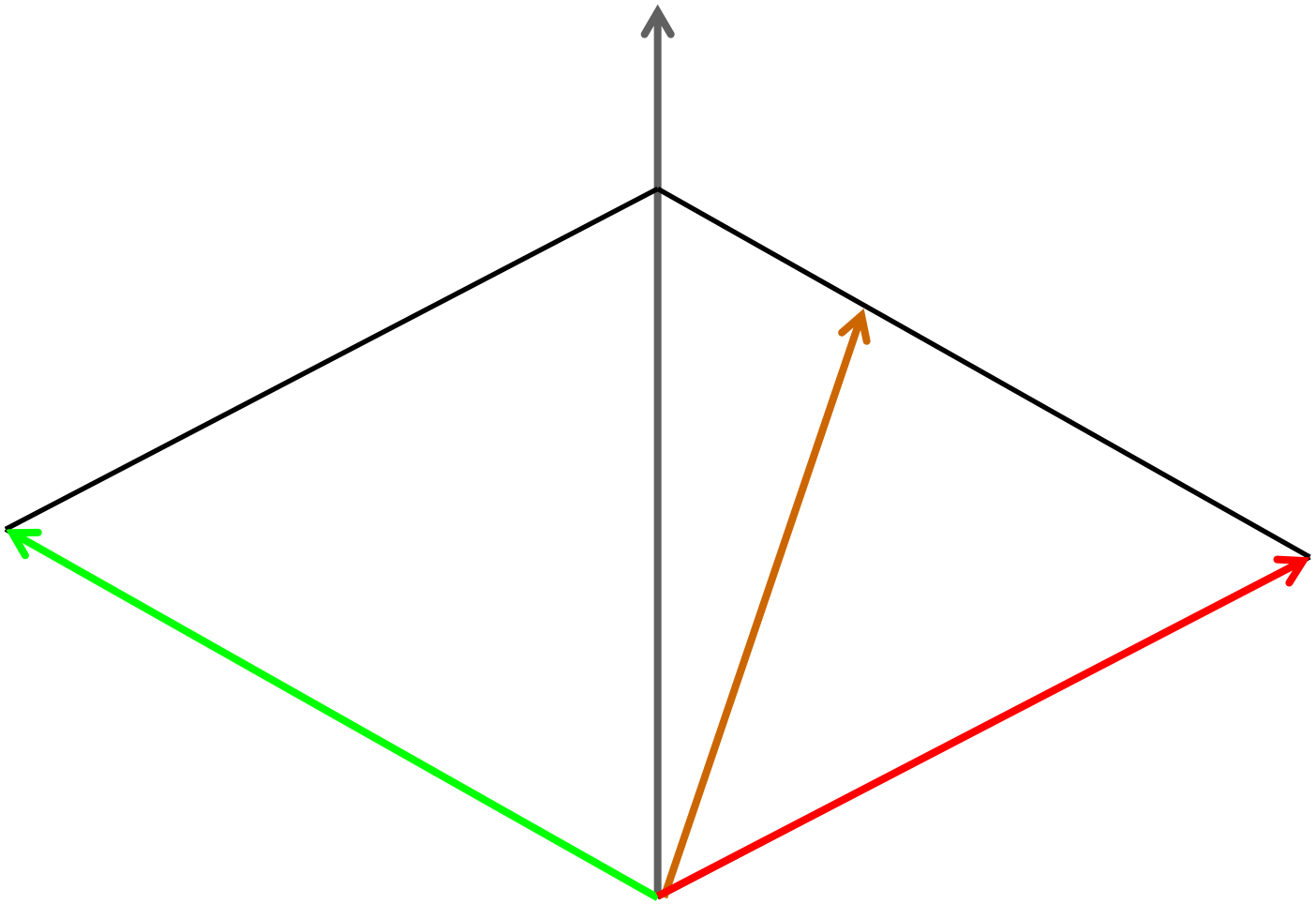
Цветовой баланс (баланс белого)

- Задача не имеет однозначного решения
  - Спектральная информация не сохраняется
  - Метамерия
  - …
- Цели
  - Компенсация особенностей освещения
  - Компенсация «ошибок» внутрикамерной обработки
- Критерии
  - Серое должно быть серым
  - Памятные цвета (телесный, …) должны быть правильными.
  - Чтобы нравилось
- Алгоритмы (преобразования в цветовом пространстве)
  - Сдвиг

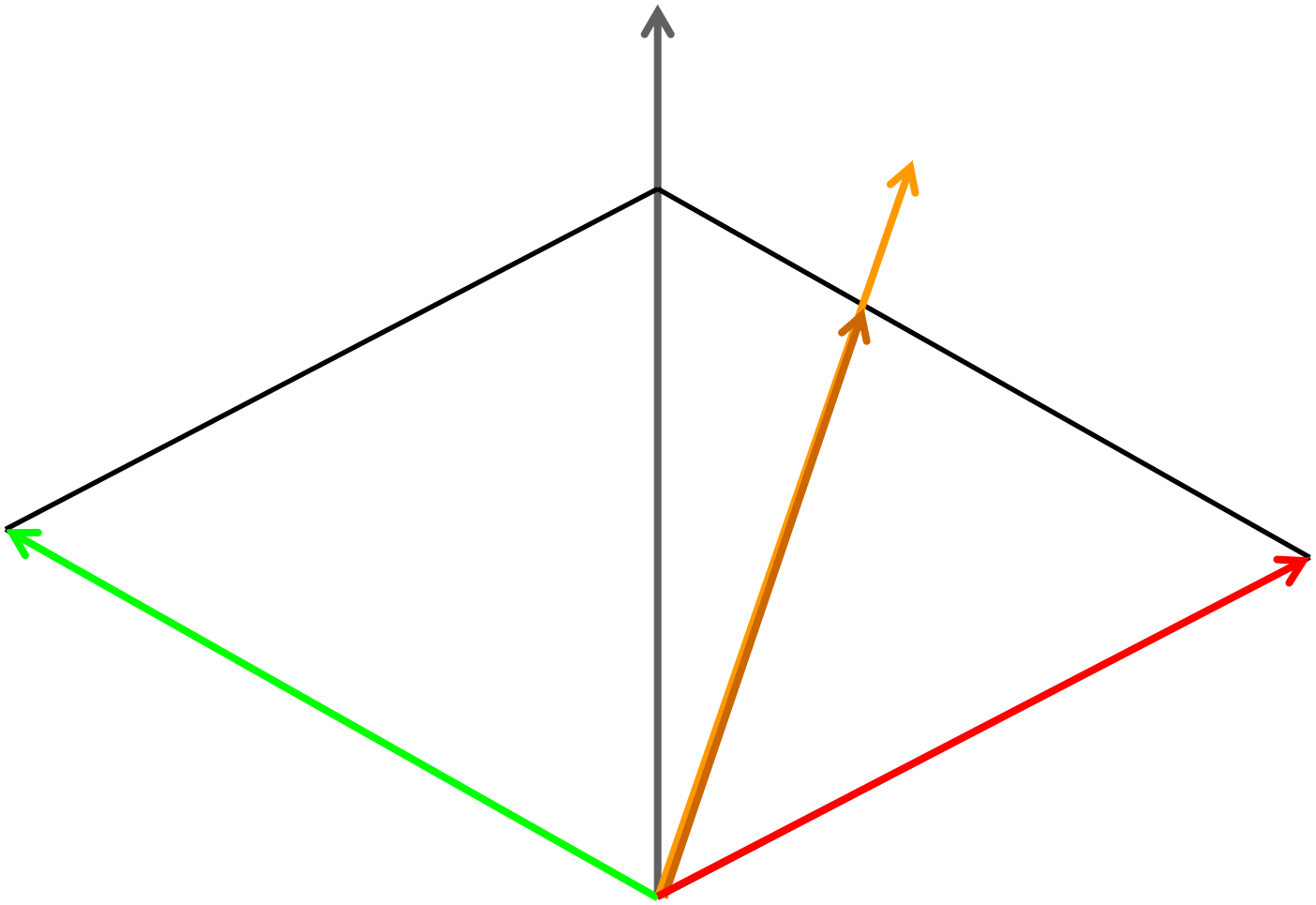
Красно-зеленый мир (RG)



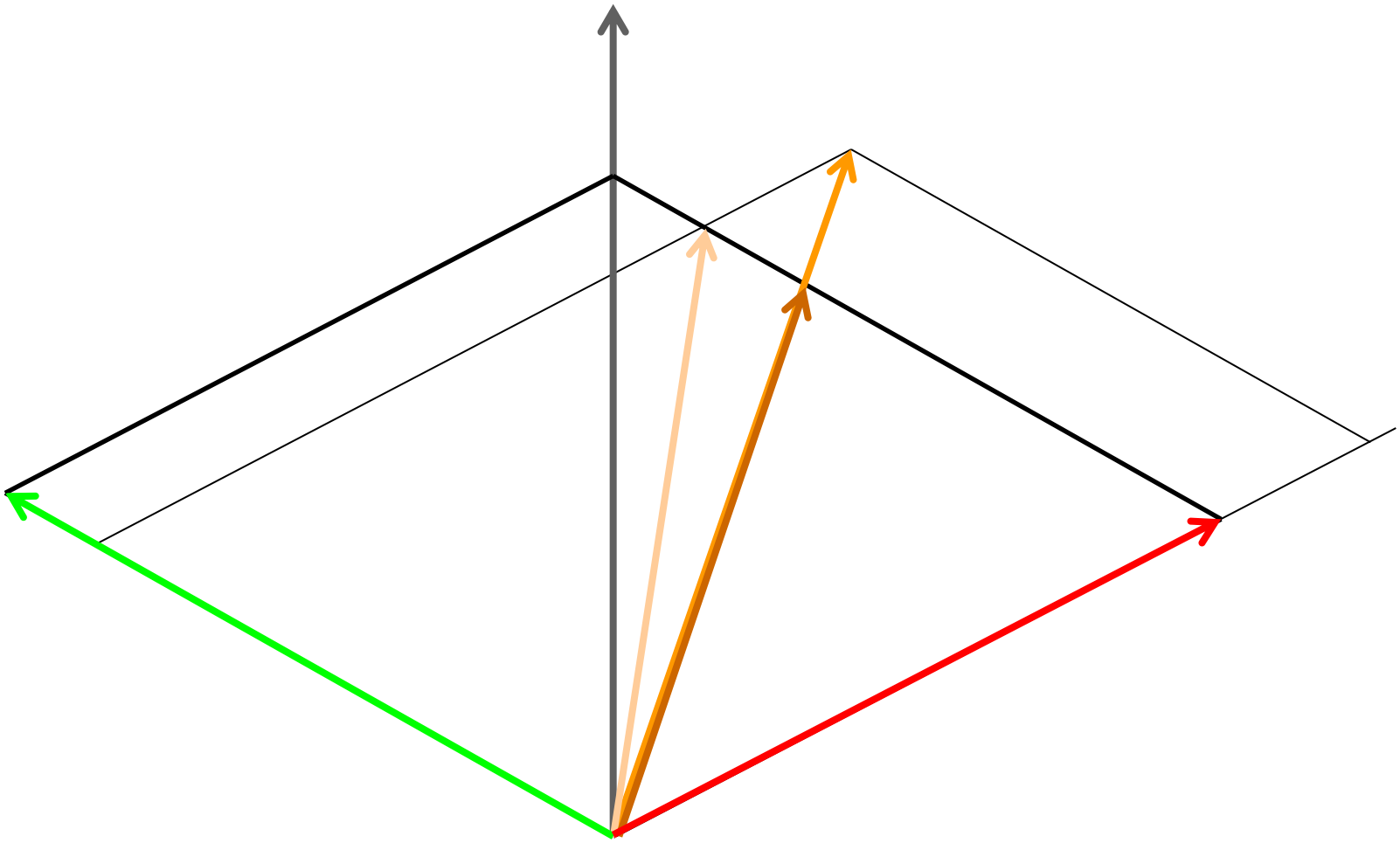
Выход за охват. Отсечение



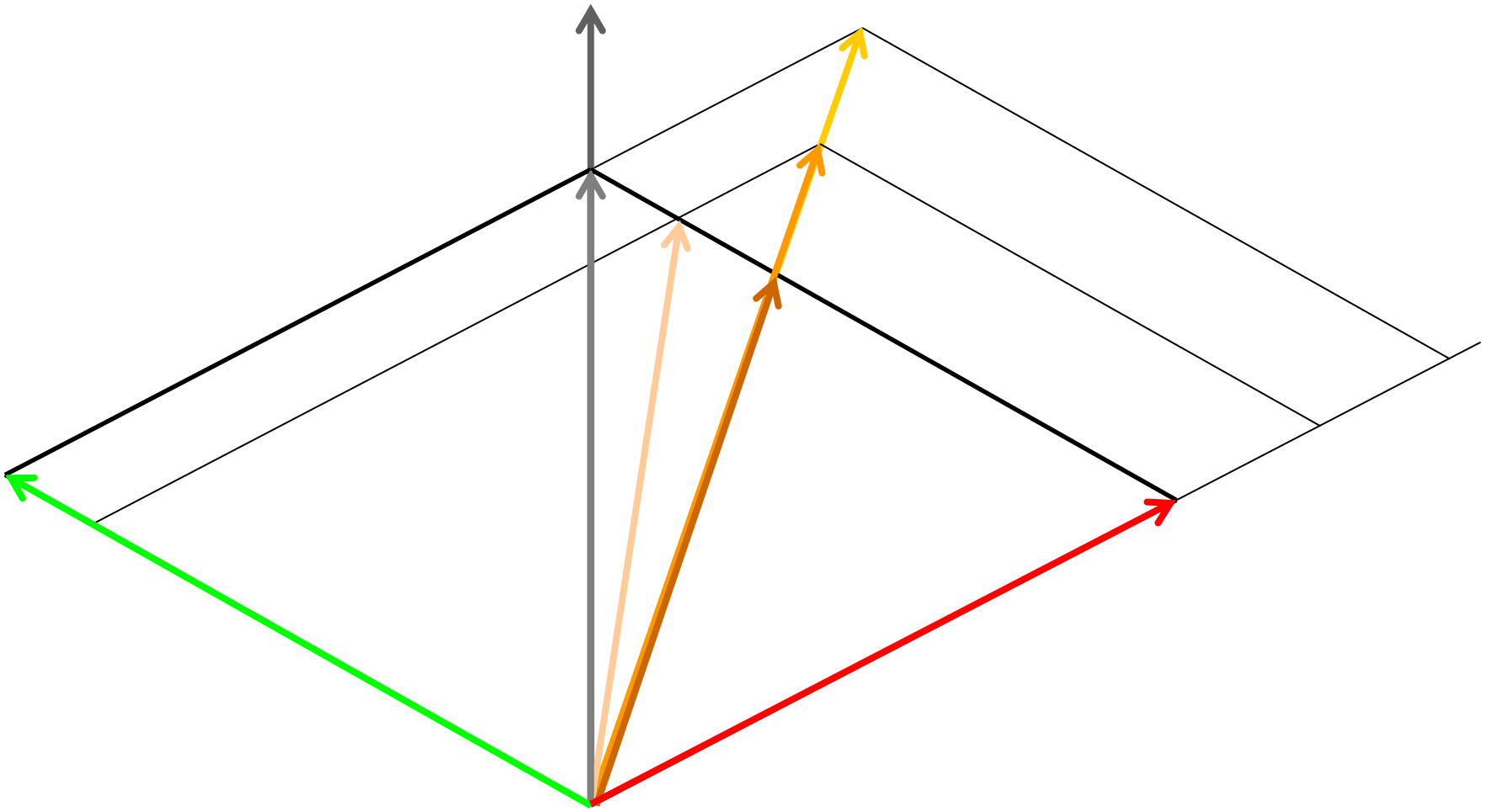
Выход за охват. Отсечение



Выход за охват. Отсечение



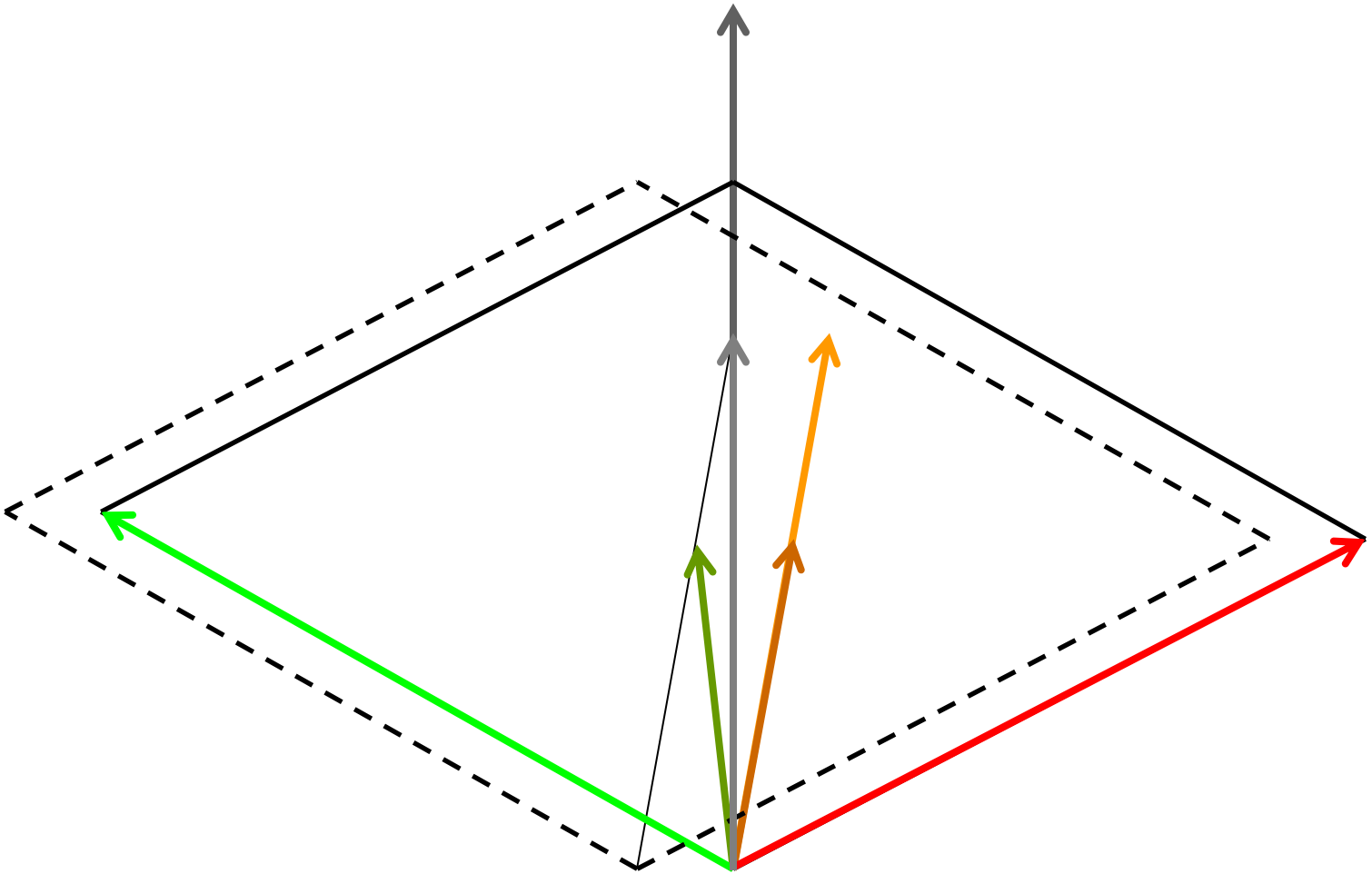
Выход за охват. Отсечение



Цветовой баланс (баланс белого)

- Задача не имеет однозначного решения
  - Спектральная информация не сохраняется
  - Метамерия
  - …
- Цели
  - Компенсация особенностей освещения
  - Компенсация «ошибок» внутрикамерной обработки
- Критерии
  - Серое должно быть серым
  - Памятные цвета (телесный, …) должны быть правильными.
  - Чтобы нравилось
- Алгоритмы (преобразования в цветовом пространстве)
  - Сдвиг

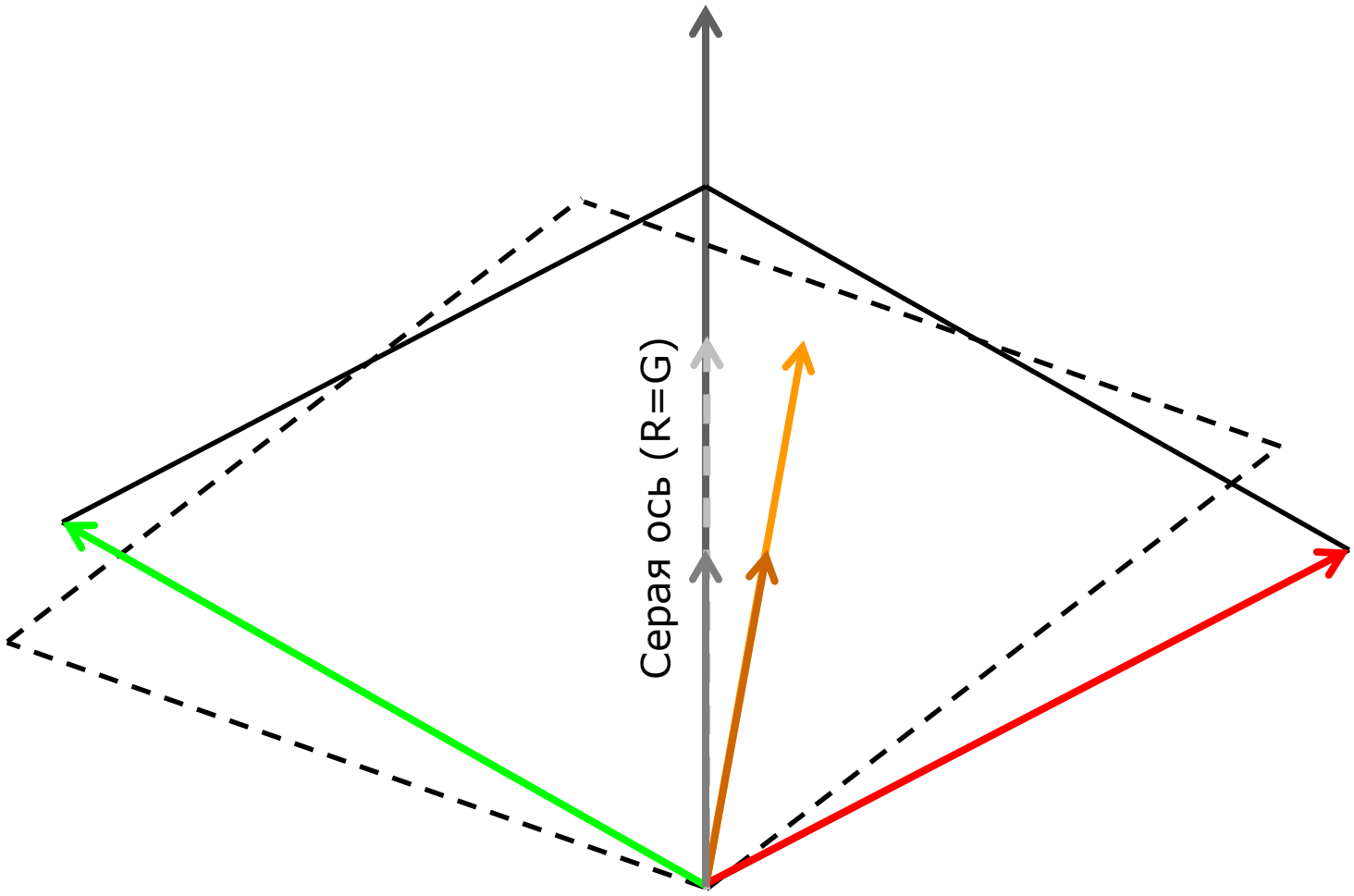
Сдвиг



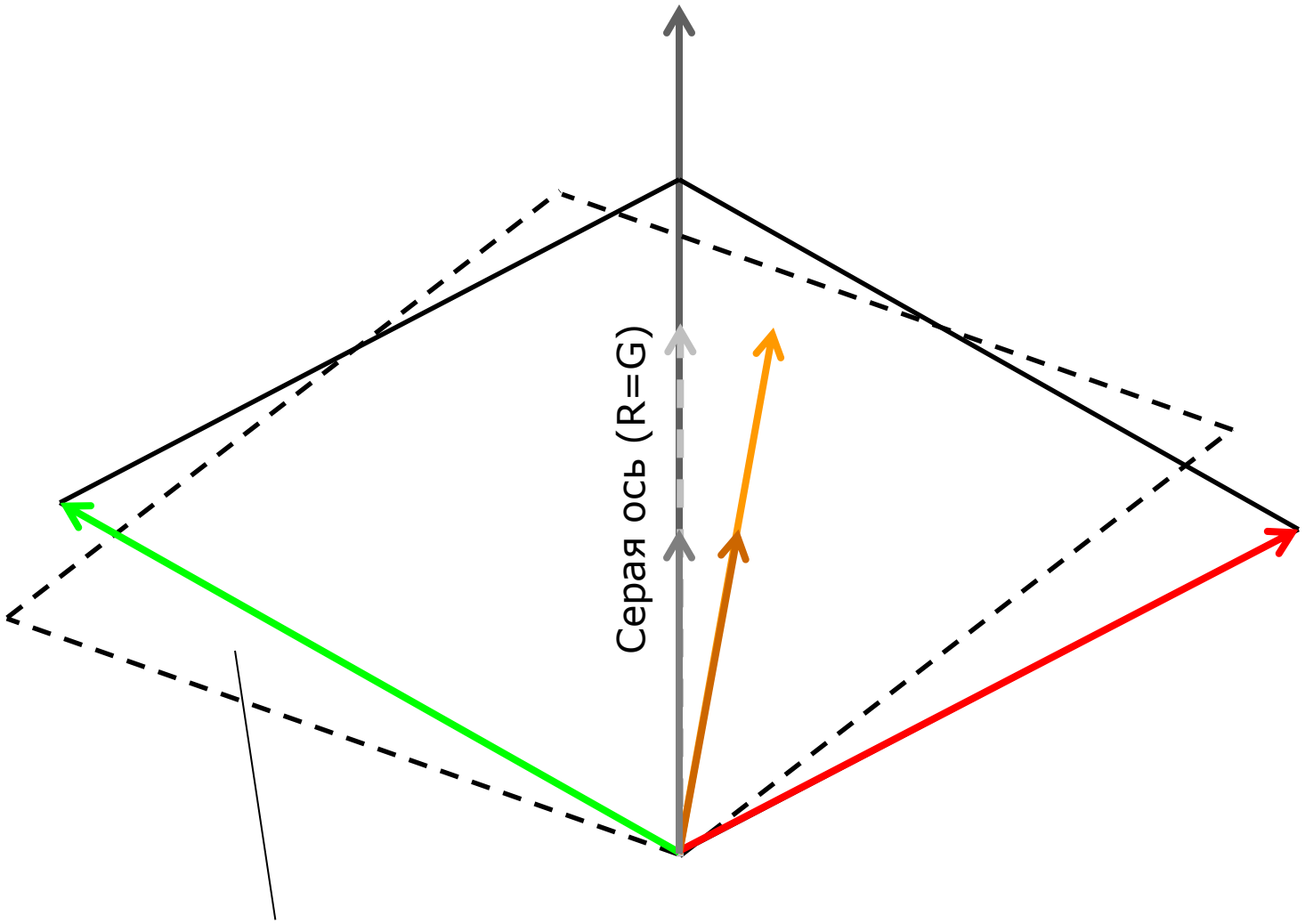
Цветовой баланс (баланс белого)

- Задача не имеет однозначного решения
  - Спектральная информация не сохраняется
  - Метамерия
  - …
- Цели
  - Компенсация особенностей освещения
  - Компенсация «ошибок» внутрикамерной обработки
- Критерии
  - Серое должно быть серым
  - Памятные цвета (телесный, …) должны быть правильными.
  - Чтобы нравилось
- Алгоритмы (преобразования в цветовом пространстве)
  - Сдвиг
  - **Поворот**

Поворот



Поворот

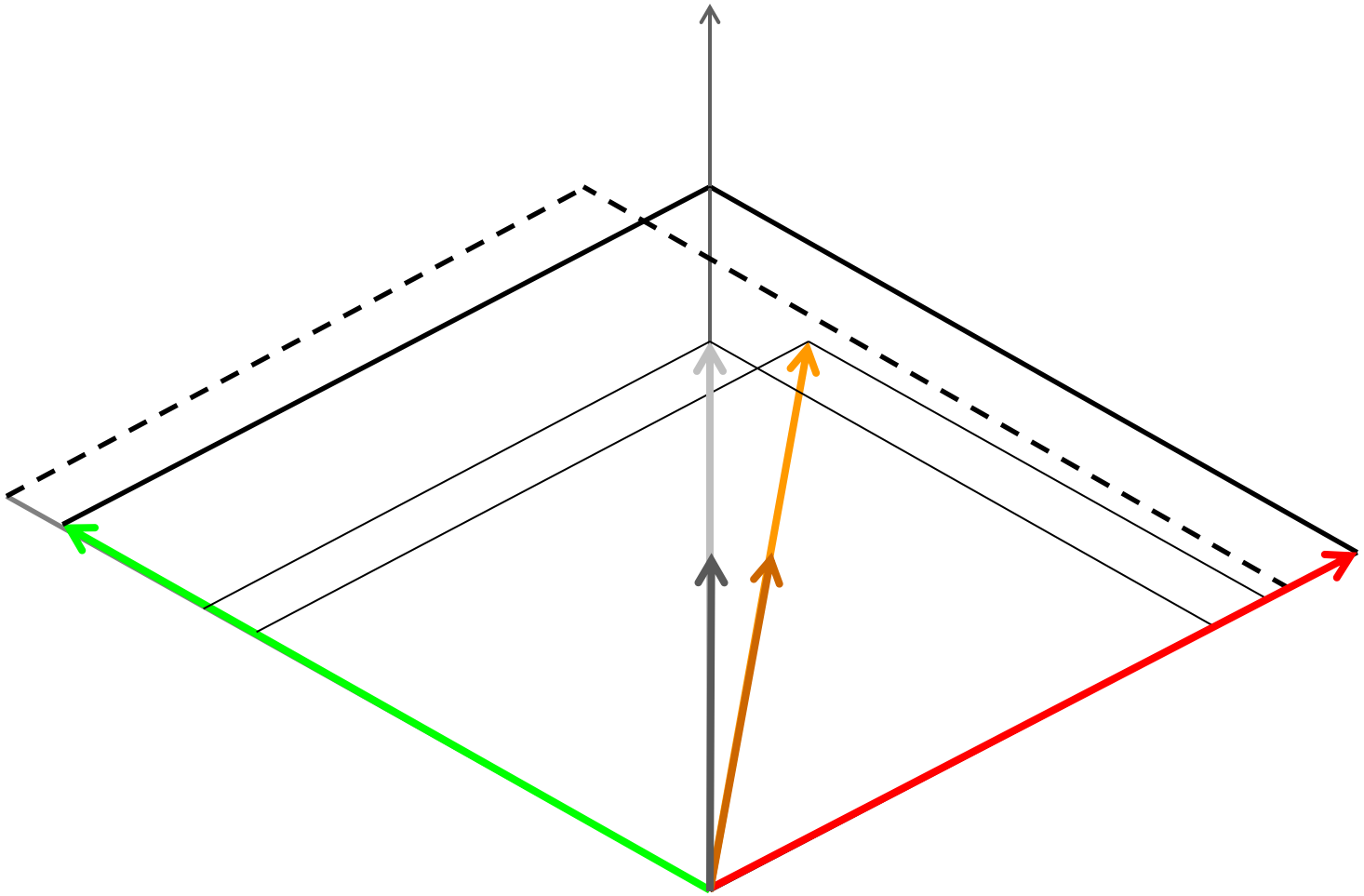


Образуются отрицательные значения RGB

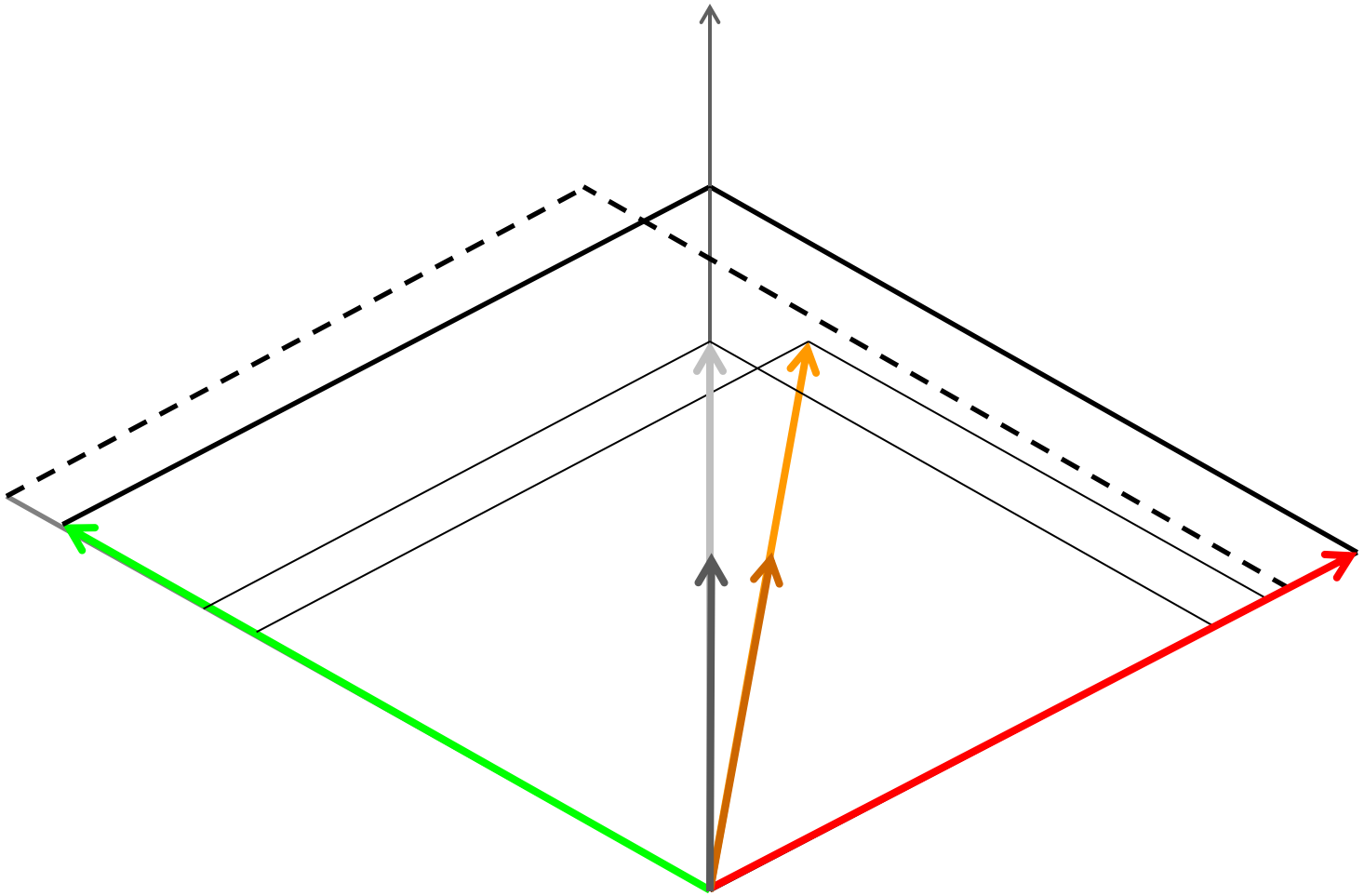
Цветовой баланс (баланс белого)

- Задача не имеет однозначного решения
  - Спектральная информация не сохраняется
  - Метамерия
  - …
- Цели
  - Компенсация особенностей освещения
  - Компенсация «ошибок» внутрикамерной обработки
- Критерии
  - Серое должно быть серым
  - Памятные цвета (телесный, …) должны быть правильными.
  - Чтобы нравилось
- Алгоритмы (преобразования в цветовом пространстве)
  - Сдвиг
  - Поворот
  - **Масштабирование RGB**

Масштабирование



Масштабирование



Возможно (последующее) уменьшение экспозиции
для возвращения в пределы охвата

Цветовой баланс (баланс белого)

- Задача не имеет однозначного решения
  - Спектральная информация не сохраняется
  - Метамерия
  - …
- Цели
  - Компенсация особенностей освещения
  - Компенсация «ошибок» внутрикамерной обработки
- Критерии
  - Серое должно быть серым
  - Памятные цвета (телесный, …) должны быть правильными.
  - Чтобы нравилось
- Алгоритмы (преобразования в цветовом пространстве)
  - Сдвиг
  - Поворот
  - √ **Масштабирование RGB**

Цветовой баланс (баланс белого)

- Задача не имеет однозначного решения
  - Спектральная информация не сохраняется
  - Метамерия
  - …
- Цели
  - Компенсация особенностей освещения
  - Компенсация «ошибок» внутрикамерной обработки
- Критерии
  - Серое должно быть серым
  - Памятные цвета (телесный, …) должны быть правильными.
  - Чтобы нравилось
- Алгоритмы (преобразования в цветовом пространстве)
  - Сдвиг
  - Поворот
  - √ **Масштабирование RGB**
  - Преобразование, приводящее к заданным цветам заданный набор точек (например, цвета мишени Color Checker).
  - …

Дополнительные главы компьютерной графики

Лекция 7

Copyright © С. Свердлов, 2009-2017
http://www.uni-vologda.ac.ru/~c3c

Почему масштабирование

- Масштабирование RGB – настройка чувствительности сенсоров (излучателей) – физично
- Соответствует упрощенной модели отражения (цветовые свойства поверхности задаются тремя коэффициентами отражения K_r, K_g, K_b)
  - «Чистые» (базисные) цвета (R, G, B) остаются чистыми
- Возможность остаться в пределах охвата

Масштабирование. Формулы

$$k_R R_0 = L_0;\ k_G G_0 = L_0;\ k_B B_0 = L_0$$

Масштабирование. Формулы

$$k_R R_0 = L_0;\ k_G G_0 = L_0;\ k_B B_0 = L_0$$

$$k_R = \frac{L_0}{R_0};\ k_G = \frac{L_0}{G_0};\ k_B = \frac{L_0}{B_0}$$

$$R_0 \neq 0;\ G_0 \neq 0;\ B_0 \neq 0;$$

Масштабирование. Формулы

$$k_R R_0 = L_0;\ k_G G_0 = L_0;\ k_B B_0 = L_0$$

$$k_R = \frac{L_0}{R_0};\ k_G = \frac{L_0}{G_0};\ k_B = \frac{L_0}{B_0}$$

$$R_0 \neq 0;\ G_0 \neq 0;\ B_0 \neq 0;$$

$$R' = k_R R$$

$$G' = k_G G$$

$$B' = k_B B$$

Вычисление светлоты

Вычисление светлоты

1. Светлота = Y

Вычисление светлоты

1. Светлота = Y

2. sRGB:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.4124 & 0.3576 & 0.1805 \\ 0.2126 & 0.7152 & 0.0722 \\ 0.0193 & 0.1192 & 0.9505 \end{bmatrix} \begin{bmatrix} R_{\text{linear}} \\ G_{\text{linear}} \\ B_{\text{linear}} \end{bmatrix}$$

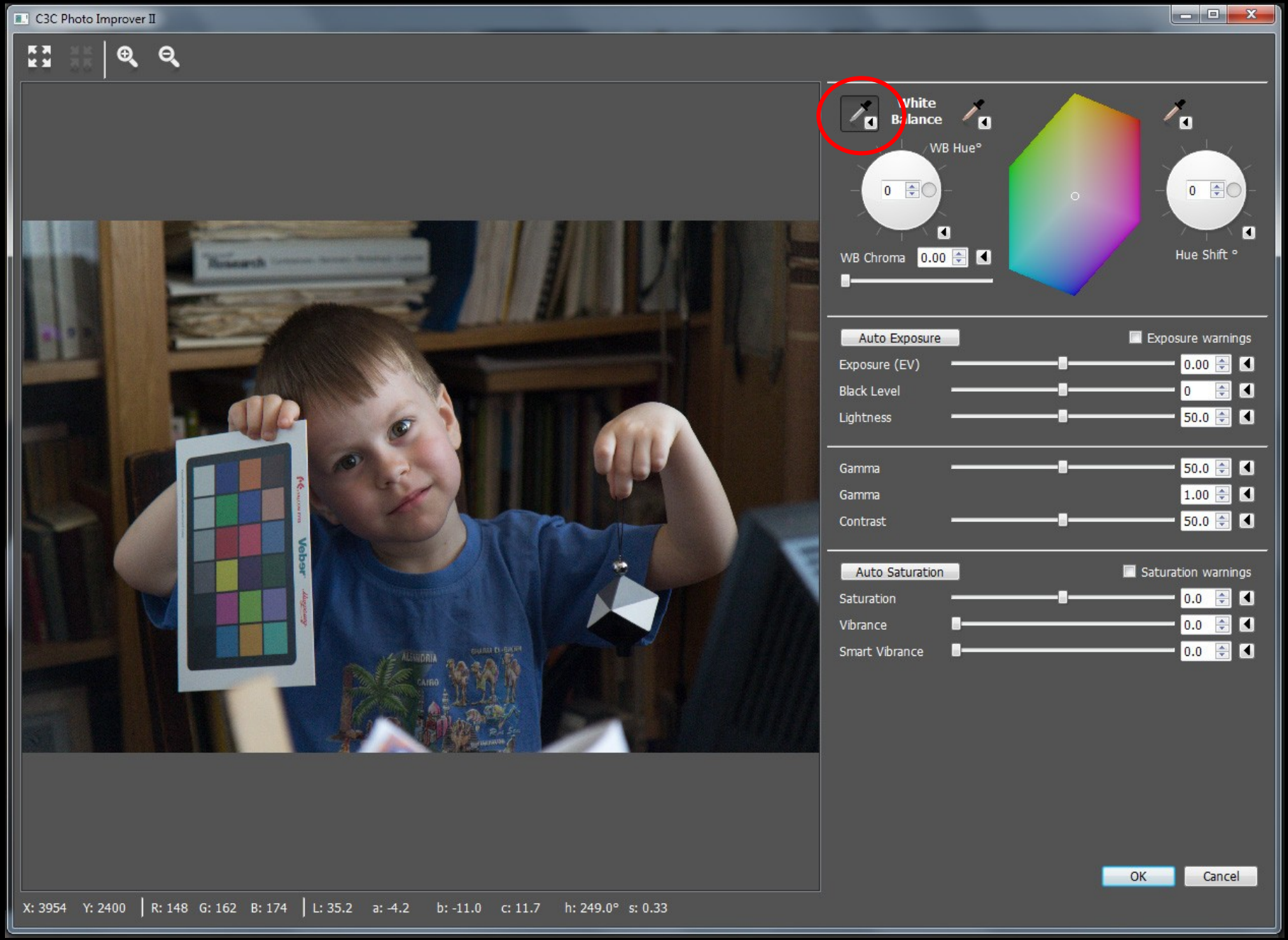
Вычисление светлоты

1. Светлота = Y

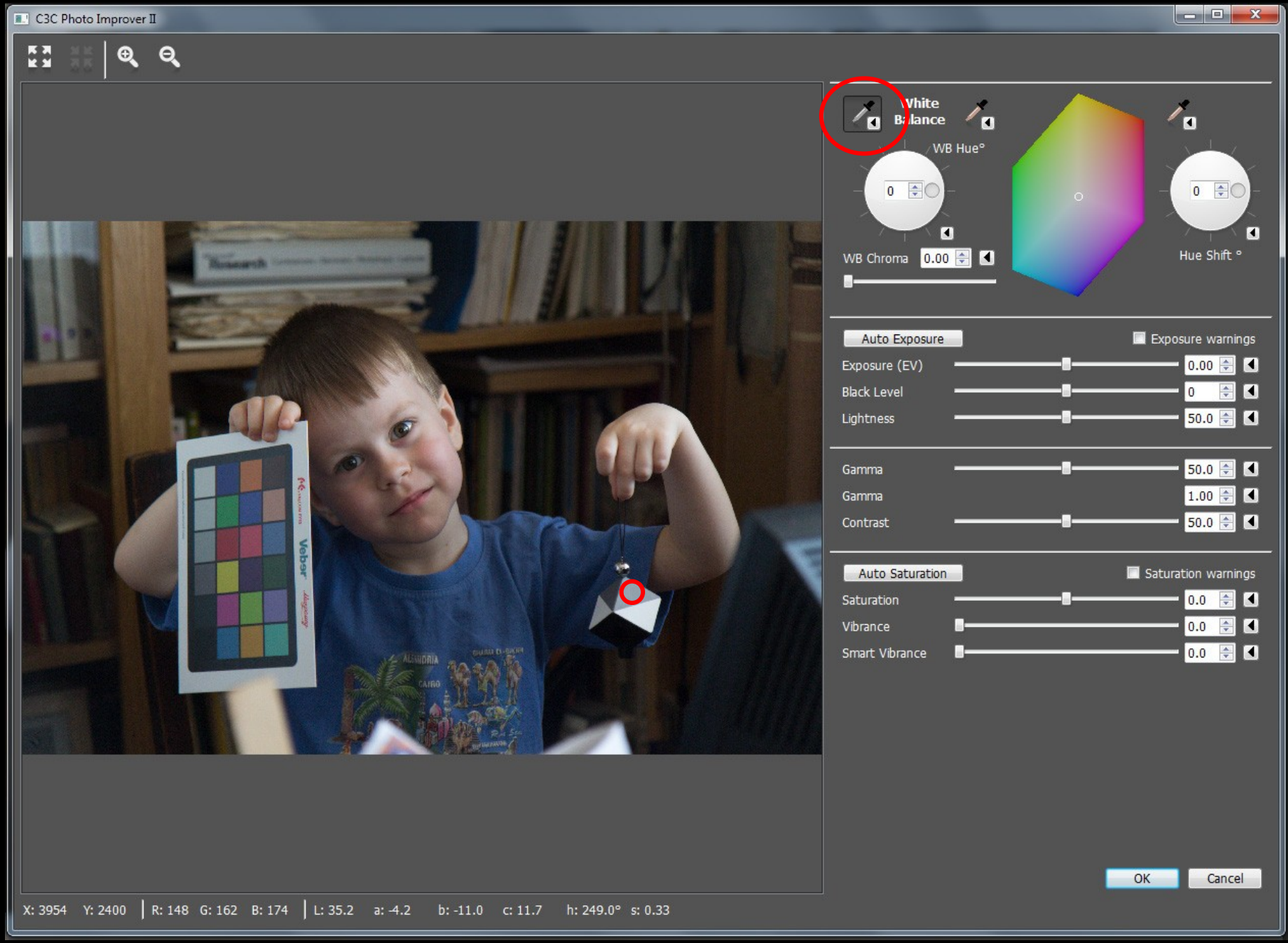
2. sRGB:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.4124 & 0.3576 & 0.1805 \\ 0.2126 & 0.7152 & 0.0722 \\ 0.0193 & 0.1192 & 0.9505 \end{bmatrix} \begin{bmatrix} R_{\text{linear}} \\ G_{\text{linear}} \\ B_{\text{linear}} \end{bmatrix}$$

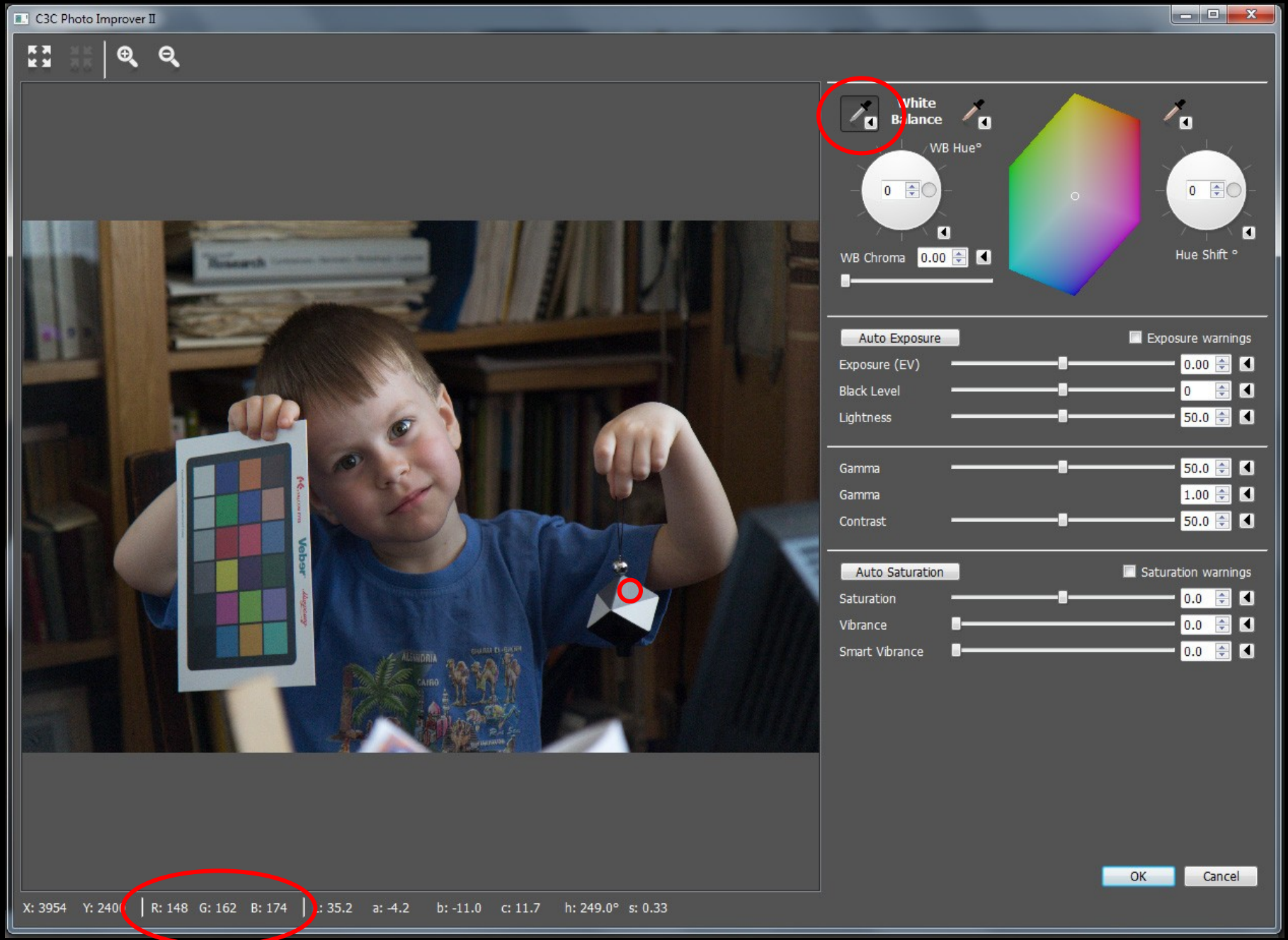
$$L = 0.2126R + 0.7152G + 0.0722B$$



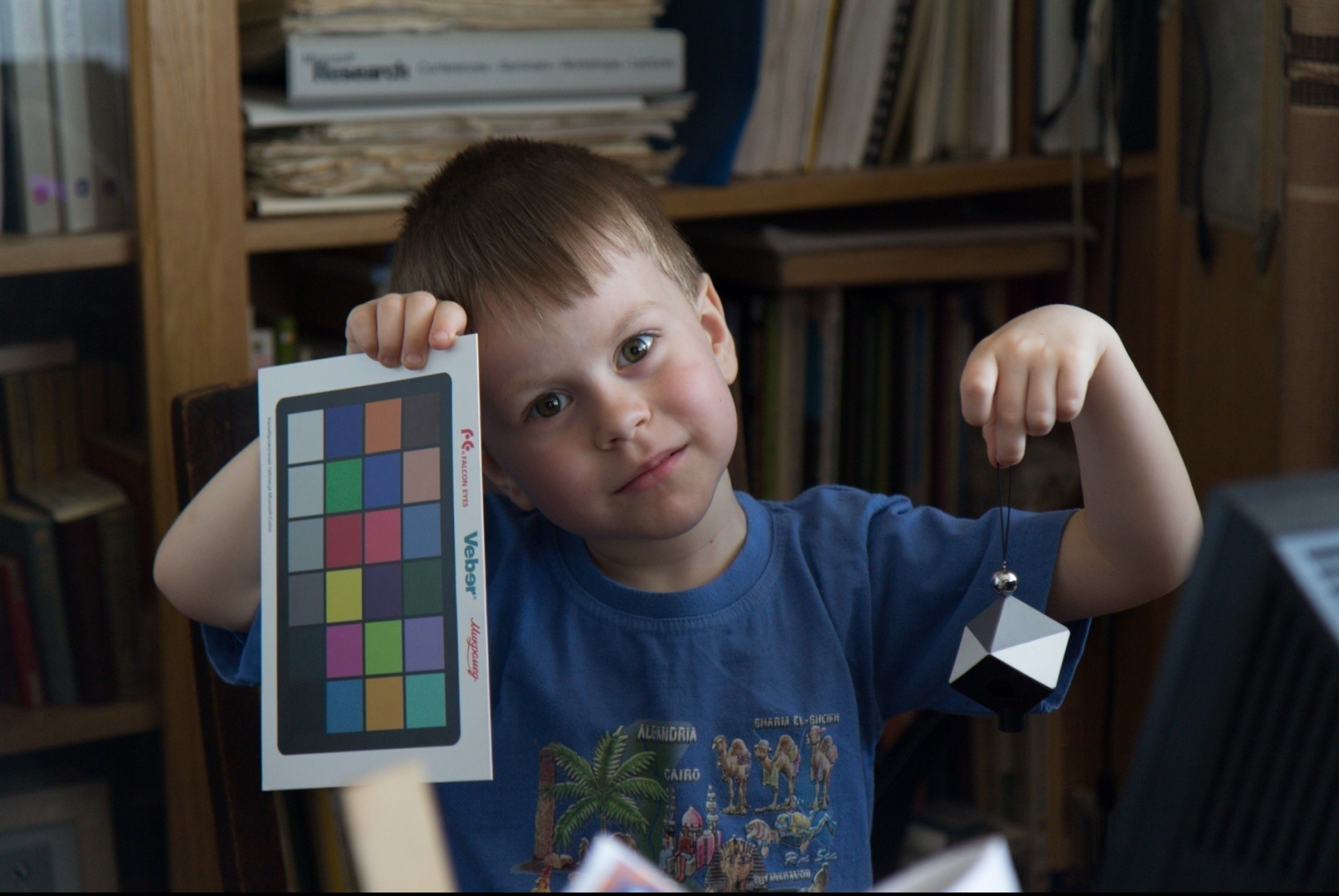
ББ по Spyder CUBE



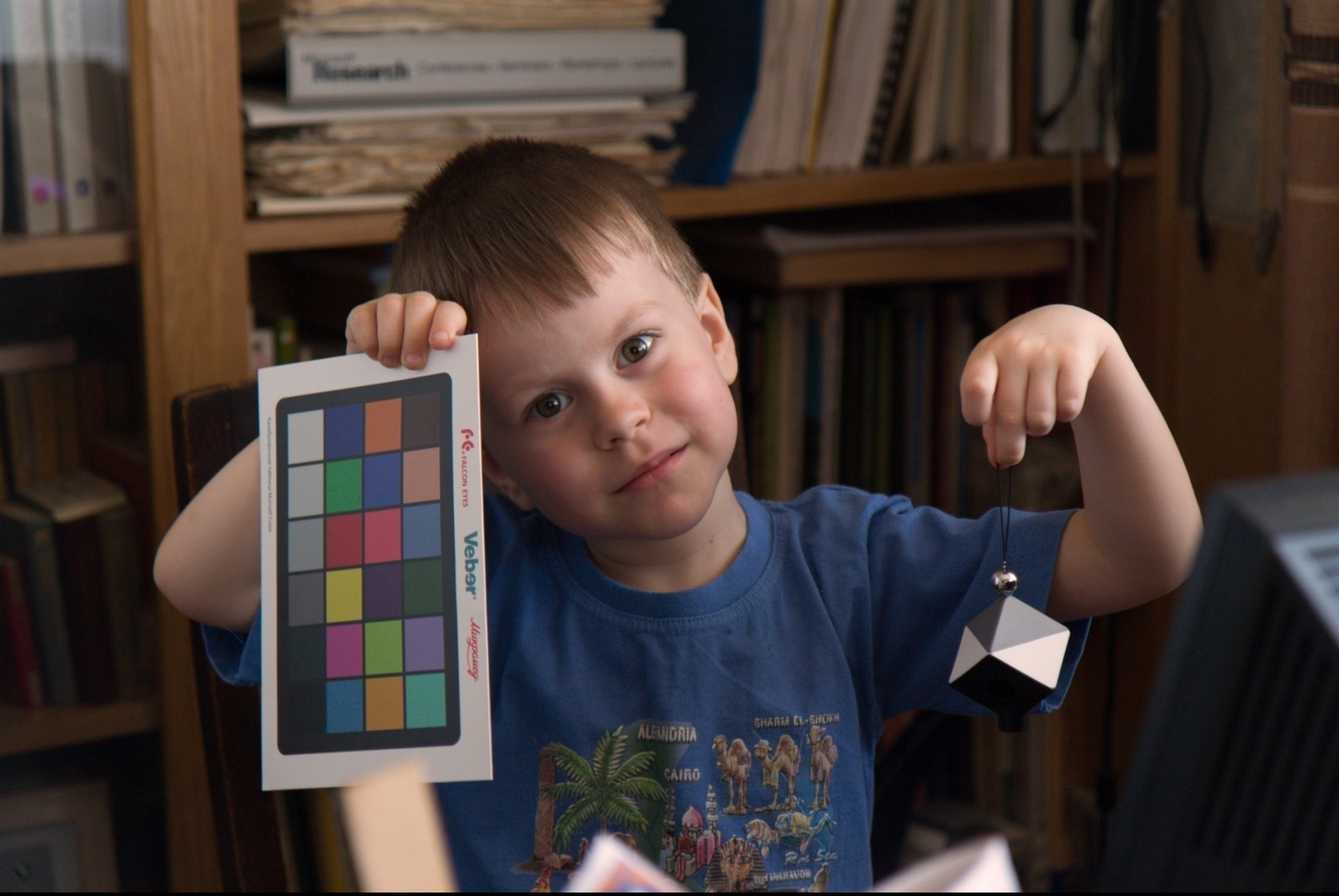
ББ по Spyder CUBE



ББ по Spyder CUBE



Как снято



Баланс белого по Spyder CUBE

Баланс белого по среднему

$$R_0 = R_{cp};\ G_0 = G_{cp};\ B_0 = B_{cp}$$

$$k_R = \frac{L_0}{R_0};\ k_G = \frac{L_0}{G_0};\ k_B = \frac{L_0}{B_0}$$

$$R_0 \neq 0;\ G_0 \neq 0;\ B_0 \neq 0;$$

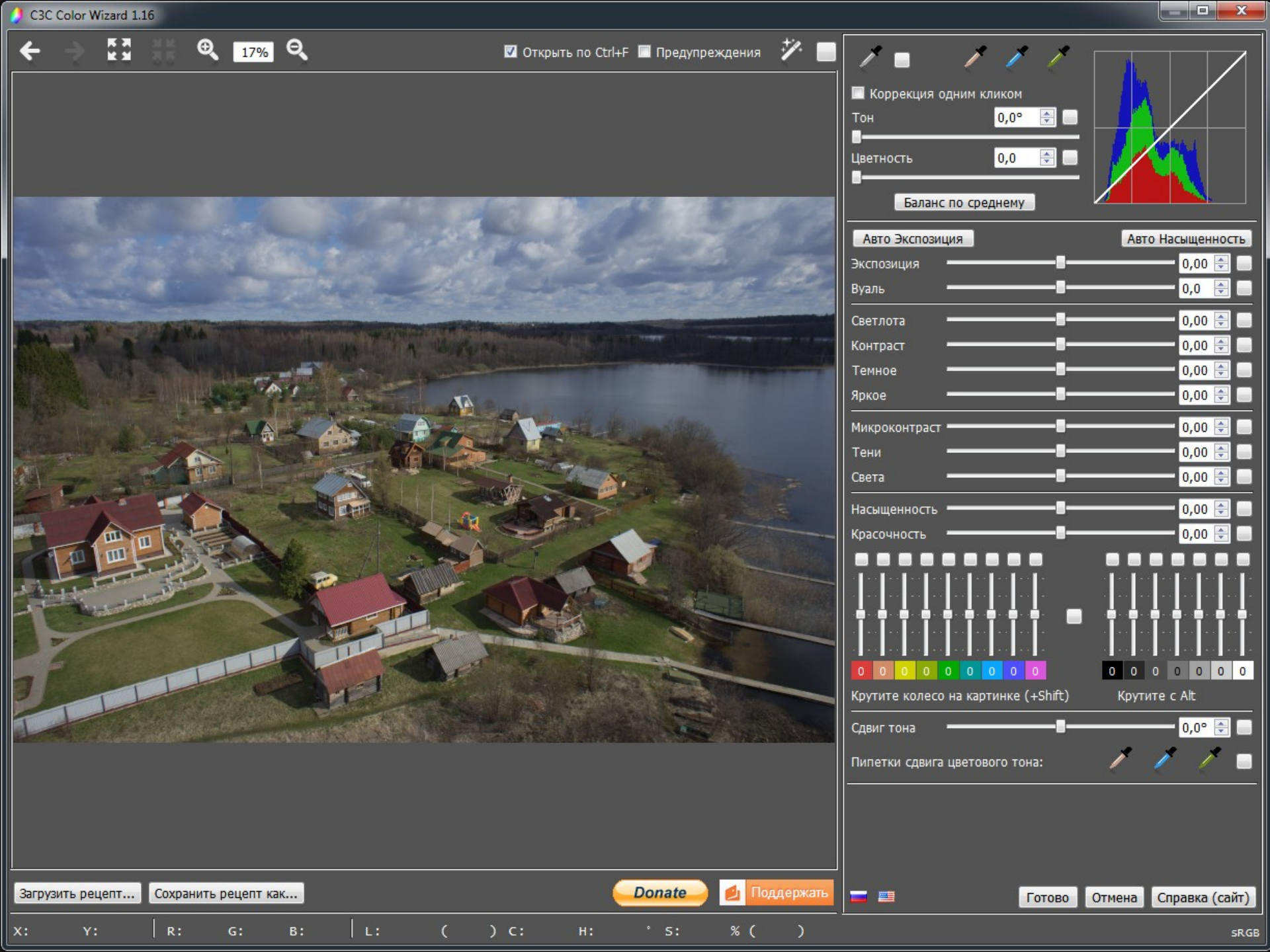
$$R' = k_R R$$

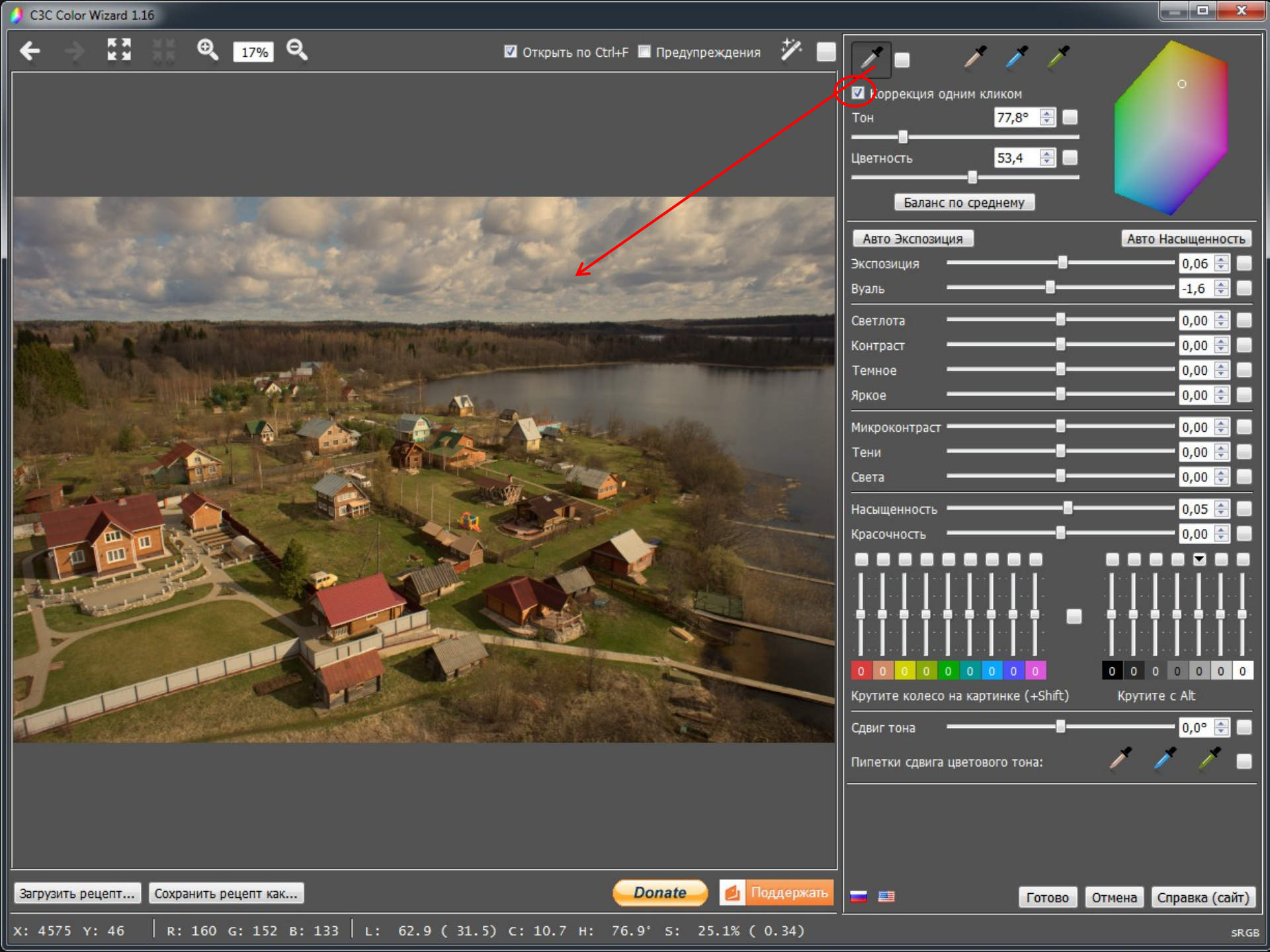
$$G' = k_G G$$

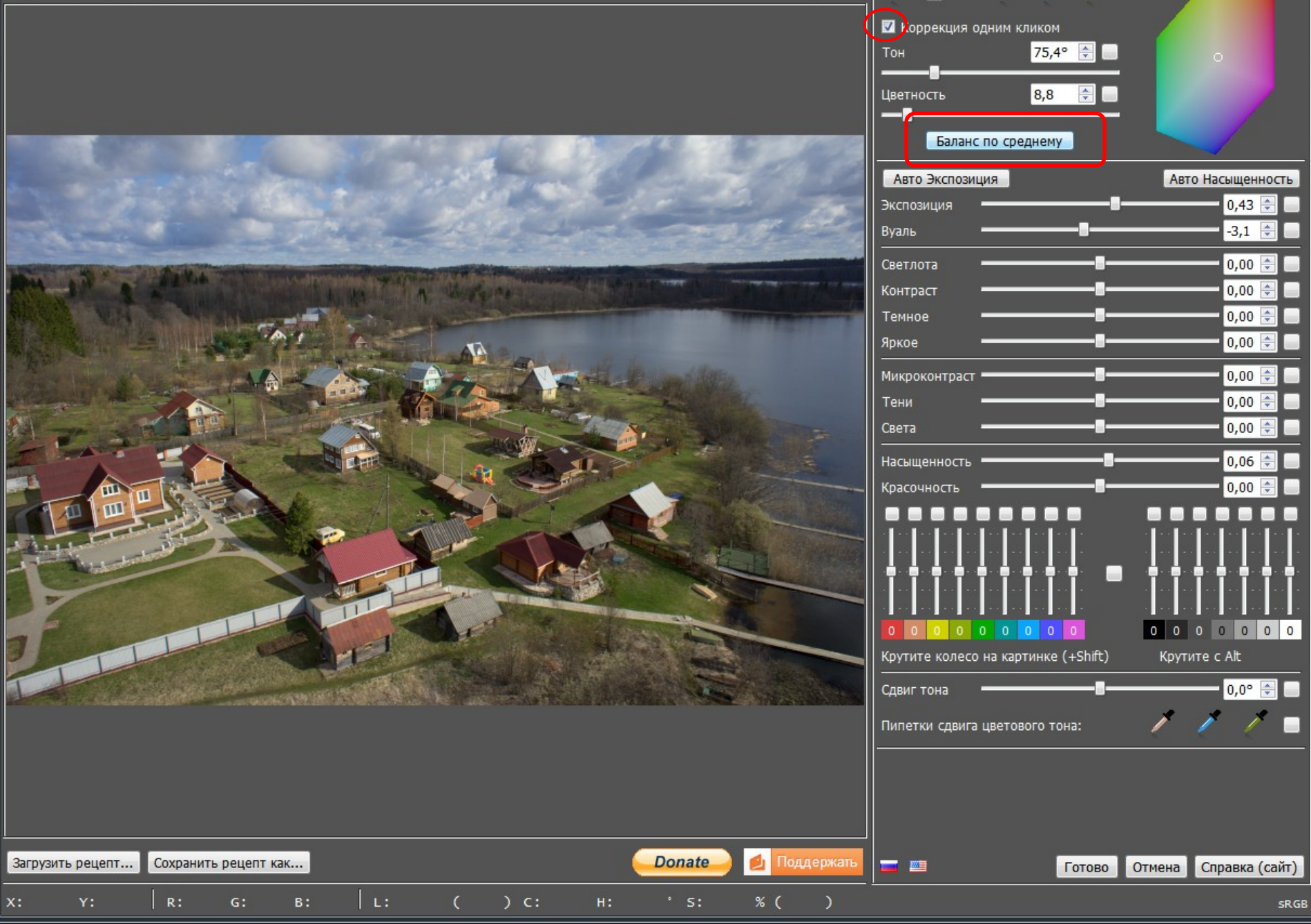
$$B' = k_B B$$

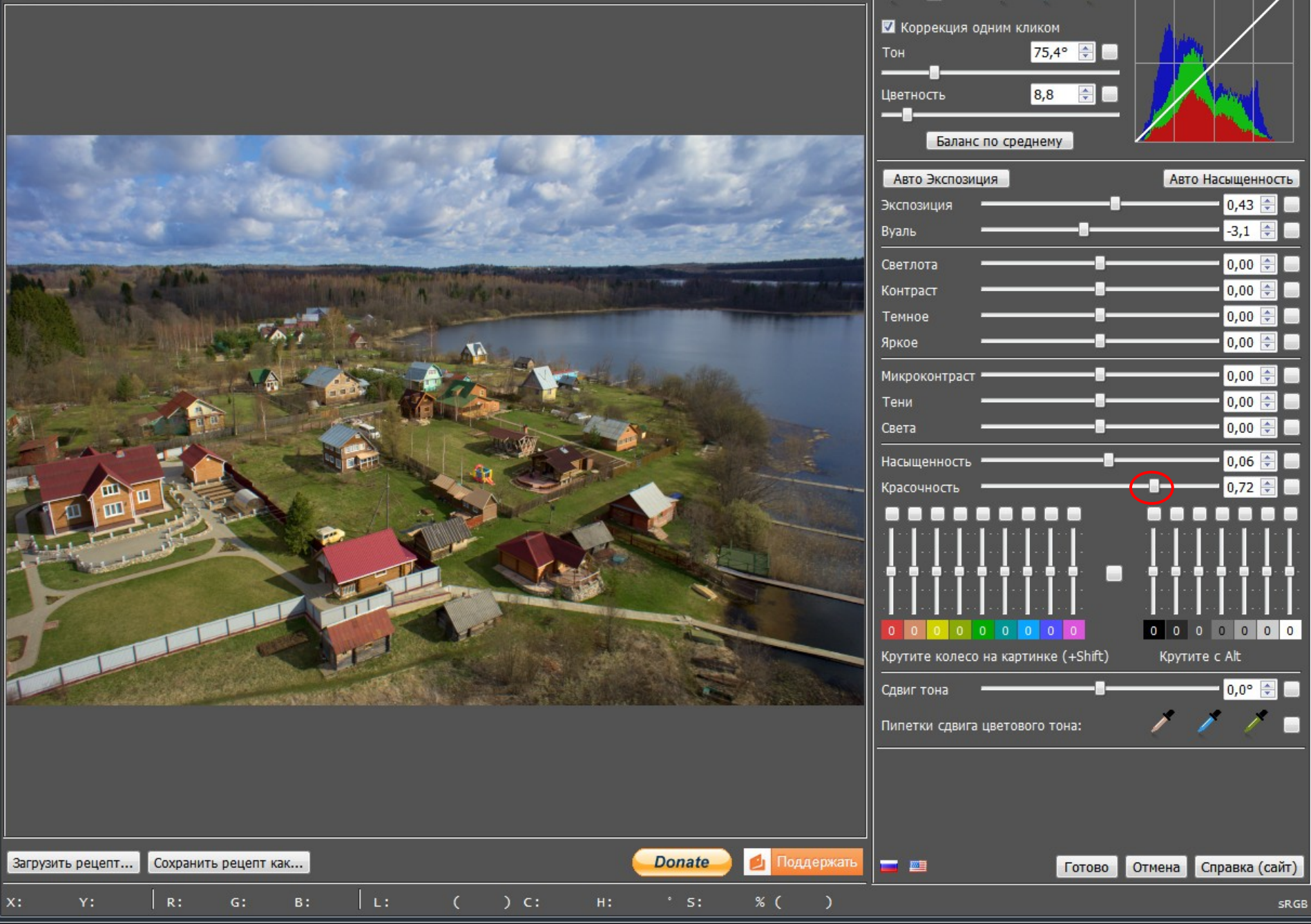
Ермаково

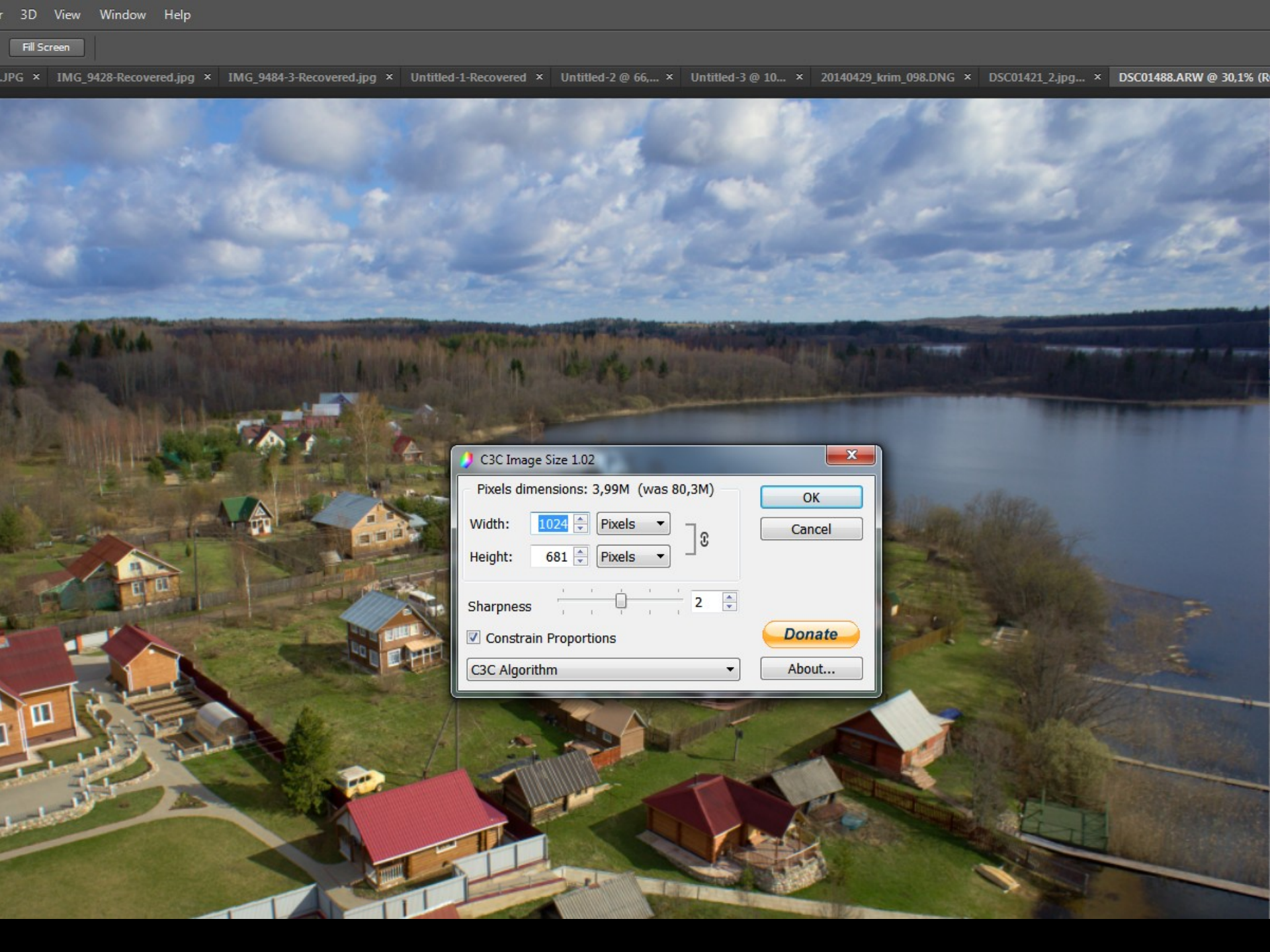
Sony NEX-5 + SEL1628, Диафрагма 5.6, выдержка 1:800с, ISO 200
Квадрокоптер 450мм с гиростабилизированным подвесом. Высота ~80м


C3C Color Wizard 1.16
17%
Открыть по Ctrl+F
Предупреждения
Коррекция одним кликом
Тон
0,0°
Цветность
0,0
Баланс по среднему
Авто Экспозиция
Авто Насыщенность
Экспозиция
0,00
Вуаль
0,0
Светлота
Контраст
Темное
Яркое
Микроконтраст
Тени
Света
Насыщенность
Красочность
Крутите колесо на картинке (+Shift)
Крутите с Alt
Сдвиг тона
Пипетки сдвига цветового тона:
Загрузить рецепт...
Сохранить рецепт как...
Donate
Поддержать
Готово
Отмена
Справка (сайт)
X:
Y:
R:
G:
B:
L:
C:
H:
S:
sRGB


C3C Color Wizard 1.16
17%
Открыть по Ctrl+F
Предупреждения
Коррекция одним кликом
Тон
77,8°
Цветность
53,4
Баланс по среднему
Авто Экспозиция
Авто Насыщенность
Экспозиция
0,06
Вуаль
-1,6
Светлота
0,00
Контраст
0,00
Темное
0,00
Яркое
0,00
Микроконтраст
0,00
Тени
0,00
Света
0,00
Насыщенность
0,05
Красочность
0,00
Крутите колесо на картинке (+Shift)
Крутите с Alt
Сдвиг тона
0,0°
Пипетки сдвига цветового тона:
Загрузить рецепт...
Сохранить рецепт как...
Donate
Поддержать
Готово
Отмена
Справка (сайт)
X: 4575 Y: 46
R: 160 G: 152 B: 133
L: 62.9 (31.5) C: 10.7 H: 76.9° S: 25.1% (0.34)
sRGB


C3C Color Wizard 1.16
17%
Открыть по Ctrl+F
Предупреждения
Коррекция одним кликом
Тон
75,4°
Цветность
8,8
Баланс по среднему
Авто Экспозиция
Авто Насыщенность
Экспозиция
0,43
Вуаль
-3,1
Светлота
0,00
Контраст
0,00
Темное
0,00
Яркое
0,00
Микроконтраст
0,00
Тени
0,00
Света
0,00
Насыщенность
0,06
Красочность
0,00
Крутите колесо на картинке (+Shift)
Крутите с Alt
Сдвиг тона
0,0°
Пипетки сдвига цветового тона:
Загрузить рецепт...
Сохранить рецепт как...
Donate
Поддержать
Готово
Отмена
Справка (сайт)
X:
Y:
R:
G:
B:
L:
C:
H:
S:
sRGB


C3C Color Wizard 1.16
17%
Открыть по Ctrl+F
Предупреждения
Коррекция одним кликом
Тон
75,4°
Цветность
8,8
Баланс по среднему
Авто Экспозиция
Авто Насыщенность
Экспозиция
0,43
Вуаль
-3,1
Светлота
0,00
Контраст
0,00
Темное
0,00
Яркое
0,00
Микроконтраст
0,00
Тени
0,00
Света
0,00
Насыщенность
0,06
Красочность
0,72
Крутите колесо на картинке (+Shift)
Крутите с Alt
Сдвиг тона
0,0°
Пипетки сдвига цветового тона:
Загрузить рецепт...
Сохранить рецепт как...
Donate
Поддержать
Готово
Отмена
Справка (сайт)
X:
Y:
R:
G:
B:
L:
C:
H:
S:
sRGB


3D
View
Window
Help
Fill Screen
IMG_9428-Recovered.jpg
IMG_9484-3-Recovered.jpg
Untitled-1-Recovered
Untitled-2 @ 66,...
Untitled-3 @ 10...
20140429_krim_098.DNG
DSC01421_2.jpg...
DSC01488.ARW @ 30,1% (R
C3C Image Size 1.02
Pixels dimensions: 3,99M (was 80,3M)
Width:
1024
Pixels
Height:
681
Pixels
Sharpness
2
Constrain Proportions
C3C Algorithm
OK
Cancel
Donate
About...

Вечерняя Москва

Баланс белого по среднему

Коррекция цветового баланса по памятным цветам

Памятные цвета. Устойчив цветовой тон

- Цвет кожи (h ≈ 60°, Lch(py))
- Цвет неба (h ≈ 255°)
- Цвет зелени (h ≈ 107°)

Преобразуем исходные RGB-координаты выбранного участка кожи в координатную систему Lch(py) . Для этого вначале получим координаты L_1, p_1 и y_1:

$$\begin{pmatrix} L_1 \\ p_1 \\ y_1 \end{pmatrix} = M \cdot \begin{pmatrix} R_1 \\ G_1 \\ B_1 \end{pmatrix}$$

здесь M — матрица преобразования из координатной системы RGB в координатную систему Lpy [4]. Вычислим цветность:

$$c_1 = \sqrt{p_1^{\,2} + y_1^{\,2}}$$

Цветовой тон после коррекции должен быть равен h_0. Находим новые значения p и y:

$$p_2 = c_1 \cos h_0$$
$$y_2 = c_1 \sin h_0$$

Выполняя обратное преобразование из Lpy в RGB, получаем новые значения RGB для выбранного участка кожи:

$$\begin{pmatrix} R_2 \\ G_2 \\ B_2 \end{pmatrix} = M^{-1} \cdot \begin{pmatrix} L_1 \\ p_2 \\ y_2 \end{pmatrix}$$

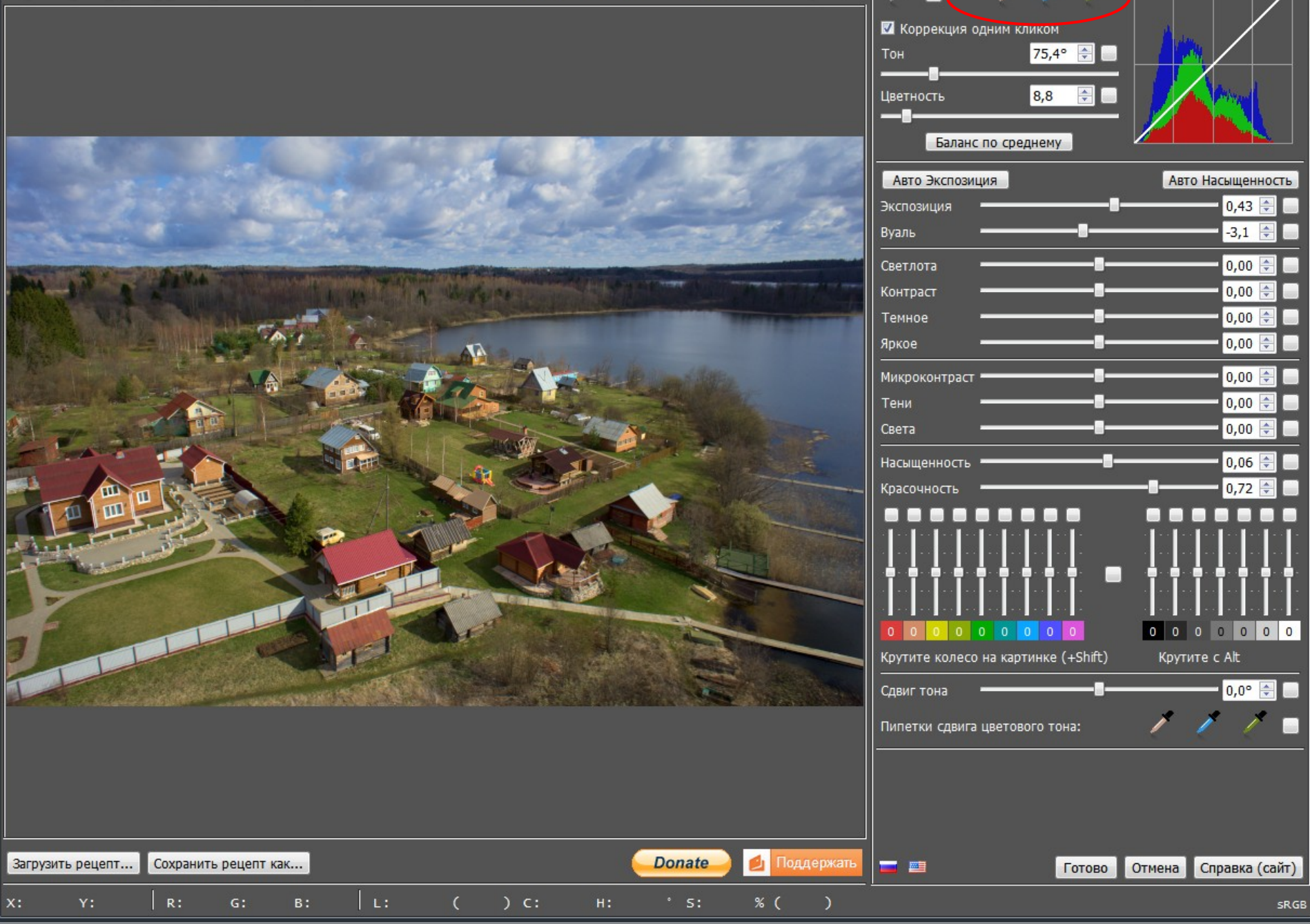
Коррекция цветового баланса по памятным цветам

$$k_R = \frac{R_2}{R_1};\ k_G = \frac{G_2}{G_1};\ k_B = \frac{B_2}{B_1}$$

$$R' = k_R R$$

$$G' = k_G G$$

$$B' = k_B B$$


C3C Color Wizard 1.16
17%
Открыть по Ctrl+F
Предупреждения
Коррекция одним кликом
Тон
75,4°
Цветность
8,8
Баланс по среднему
Авто Экспозиция
Авто Насыщенность
Экспозиция
0,43
Вуаль
-3,1
Светлота
0,00
Контраст
0,00
Темное
0,00
Яркое
0,00
Микроконтраст
0,00
Тени
0,00
Света
0,00
Насыщенность
0,06
Красочность
0,72
Крутите колесо на картинке (+Shift)
Крутите с Alt
Сдвиг тона
0,0°
Пипетки сдвига цветового тона:
Загрузить рецепт...
Сохранить рецепт как...
Donate
Поддержать
Готово
Отмена
Справка (сайт)
X:
Y:
R:
G:
B:
L:
C:
H:
S:
sRGB

Современная аэрофотосъемка

Поля тюльпанов в Голландии

Станислав Седов (проект AirPano)

Пилот Станислав Седов (проект AirPano)

Канны ночью

Станислав Седов (проект AirPano)

Баланс белого по среднему

$$R_0 = R_{cp};\ G_0 = G_{cp};\ B_0 = B_{cp}$$

$$k_R = \frac{L_0}{R_0};\ k_G = \frac{L_0}{G_0};\ k_B = \frac{L_0}{B_0}$$

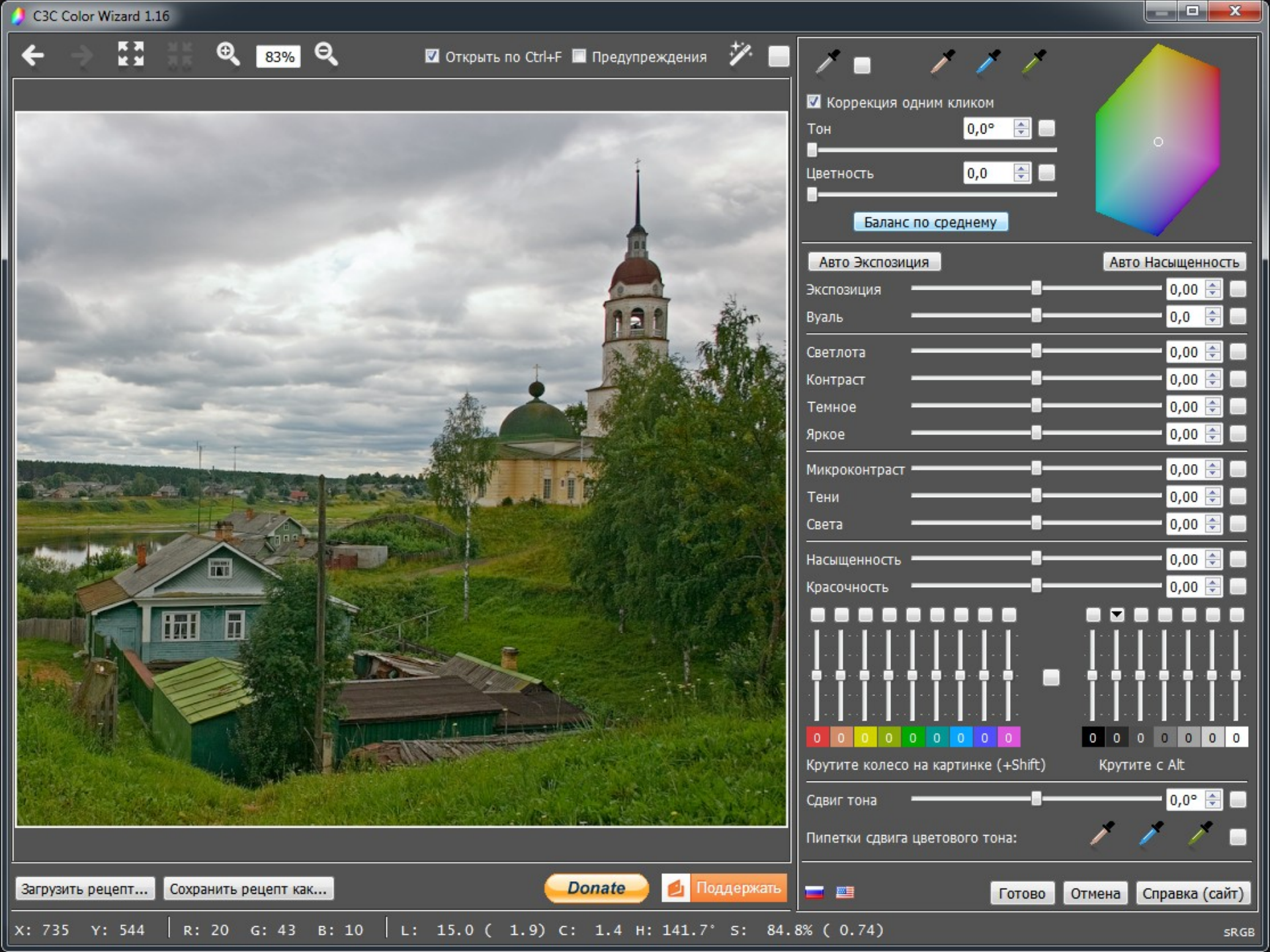
$$R_0 \neq 0;\ G_0 \neq 0;\ B_0 \neq 0;$$

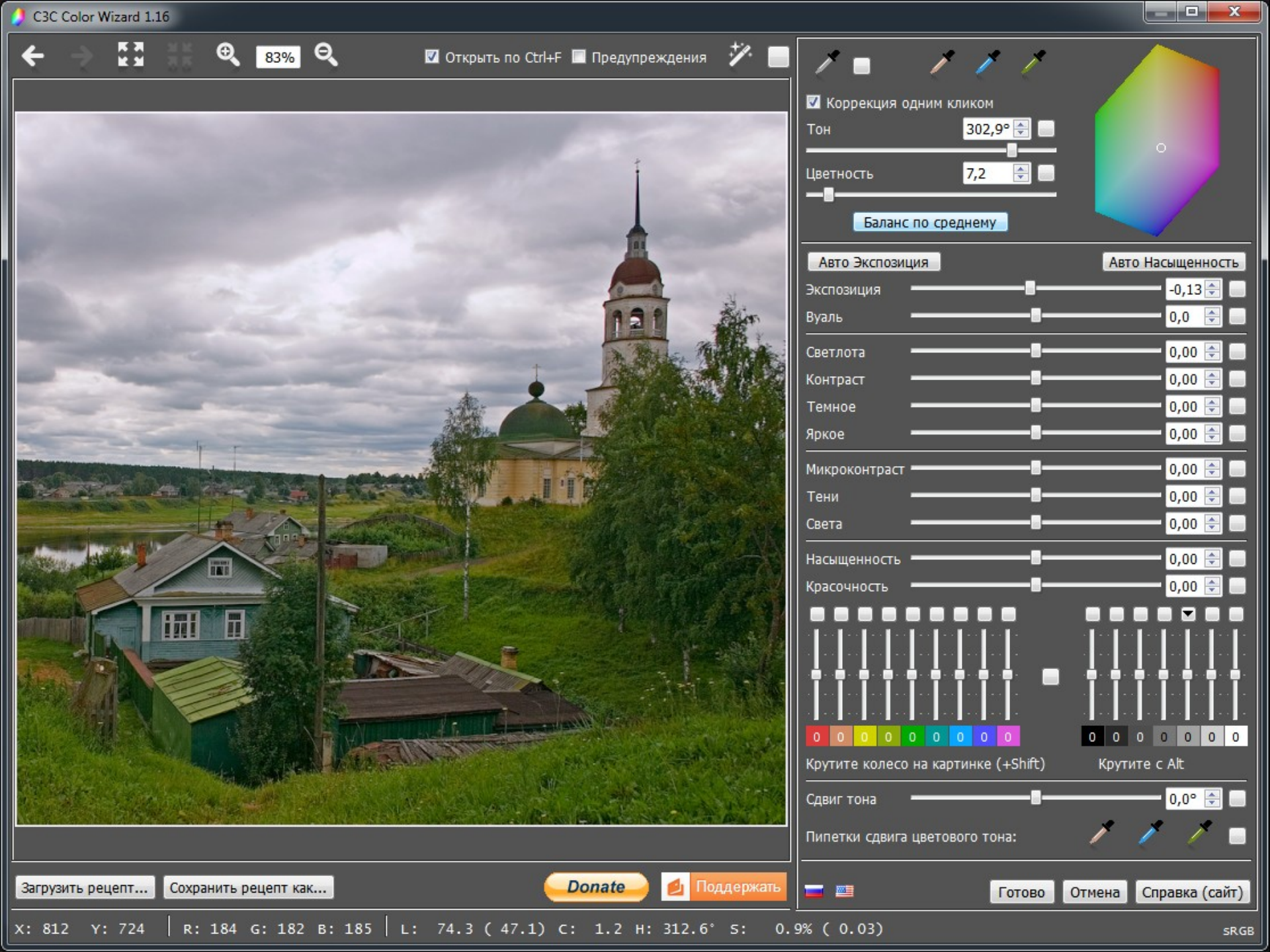
$$R' = k_R R$$

$$G' = k_G G$$

$$B' = k_B B$$

В Тотьме


C3C Color Wizard 1.16
83%
Открыть по Ctrl+F
Предупреждения
Коррекция одним кликом
Тон
0,0°
Цветность
0,0
Баланс по среднему
Авто Экспозиция
Авто Насыщенность
Экспозиция
0,00
Вуаль
0,0
Светлота
Контраст
Темное
Яркое
Микроконтраст
Тени
Света
Насыщенность
Красочность
Крутите колесо на картинке (+Shift)
Крутите с Alt
Сдвиг тона
0,0°
Пипетки сдвига цветового тона:
Загрузить рецепт...
Сохранить рецепт как...
Donate
Поддержать
Готово
Отмена
Справка (сайт)
X: 735 Y: 544
R: 20 G: 43 B: 10
L: 15.0 (1.9) C: 1.4 H: 141.7° S: 84.8% (0.74)
sRGB


C3C Color Wizard 1.16
83%
Открыть по Ctrl+F
Предупреждения
Коррекция одним кликом
Тон
302,9°
Цветность
7,2
Баланс по среднему
Авто Экспозиция
Авто Насыщенность
Экспозиция
-0,13
Вуаль
0,0
Светлота
0,00
Контраст
0,00
Темное
0,00
Яркое
0,00
Микроконтраст
0,00
Тени
0,00
Света
0,00
Насыщенность
0,00
Красочность
0,00
Крутите колесо на картинке (+Shift)
Крутите с Alt
Сдвиг тона
0,0°
Пипетки сдвига цветового тона:
Загрузить рецепт...
Сохранить рецепт как...
Donate
Поддержать
Готово
Отмена
Справка (сайт)
X: 812 Y: 724
R: 184 G: 182 B: 185
L: 74.3 (47.1) C: 1.2 H: 312.6° S: 0.9% (0.03)
sRGB

В Тотьме

В Тотьме

Насыщенность

Степень отличия от ахроматического (серого)

Насыщенность

Степень отличия от ахроматического (серого)

- **Saturation** is the attribute of a visual sensation which permits a judgment to be made of the degree to which a chromatic stimulus differs from an achromatic stimulus regardless of their brightness[1].

  - **Насыщенность** – атрибут зрительного ощущения, который позволяет судить о степени отличия хроматических стимулов от ахроматических, независимо от их яркости.

[1] *G. Wyszecki and W.S. Stiles* Color science: Concepts and methods, quantitative data and formulae, New York: Wiley, 1982 (2nd ed.), стр. 487

Насыщенность

Степень отличия от ахроматического (серого)

- **Saturation** is the attribute of a visual sensation which permits a judgment to be made of the degree to which a chromatic stimulus differs from an achromatic stimulus regardless of their brightness[1].

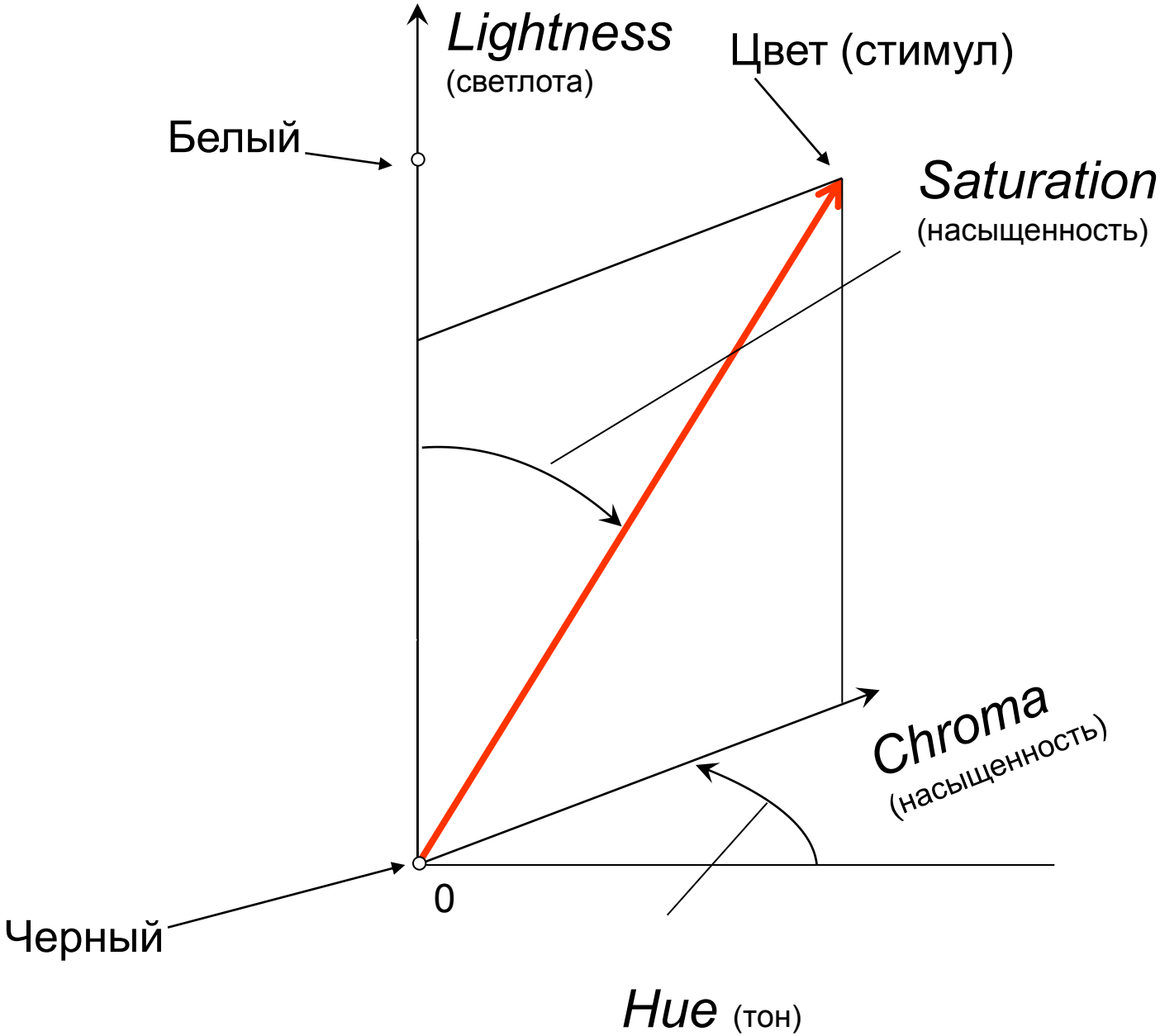
  - **Насыщенность** – атрибут зрительного ощущения, который позволяет судить о степени отличия хроматических стимулов от ахроматических, независимо от их яркости.

- **Chroma** is the attribute of a visual sensation which permits a judgment to be made of the degree to which a chromatic stimulus differs from an achromatic stimulus of the same brightness[1].

  - **Chroma (чистота цвета, цветность, насыщенность)** – атрибут зрительного ощущения, который позволяет судить о степени отличия хроматических стимулов от ахроматических той же яркости.

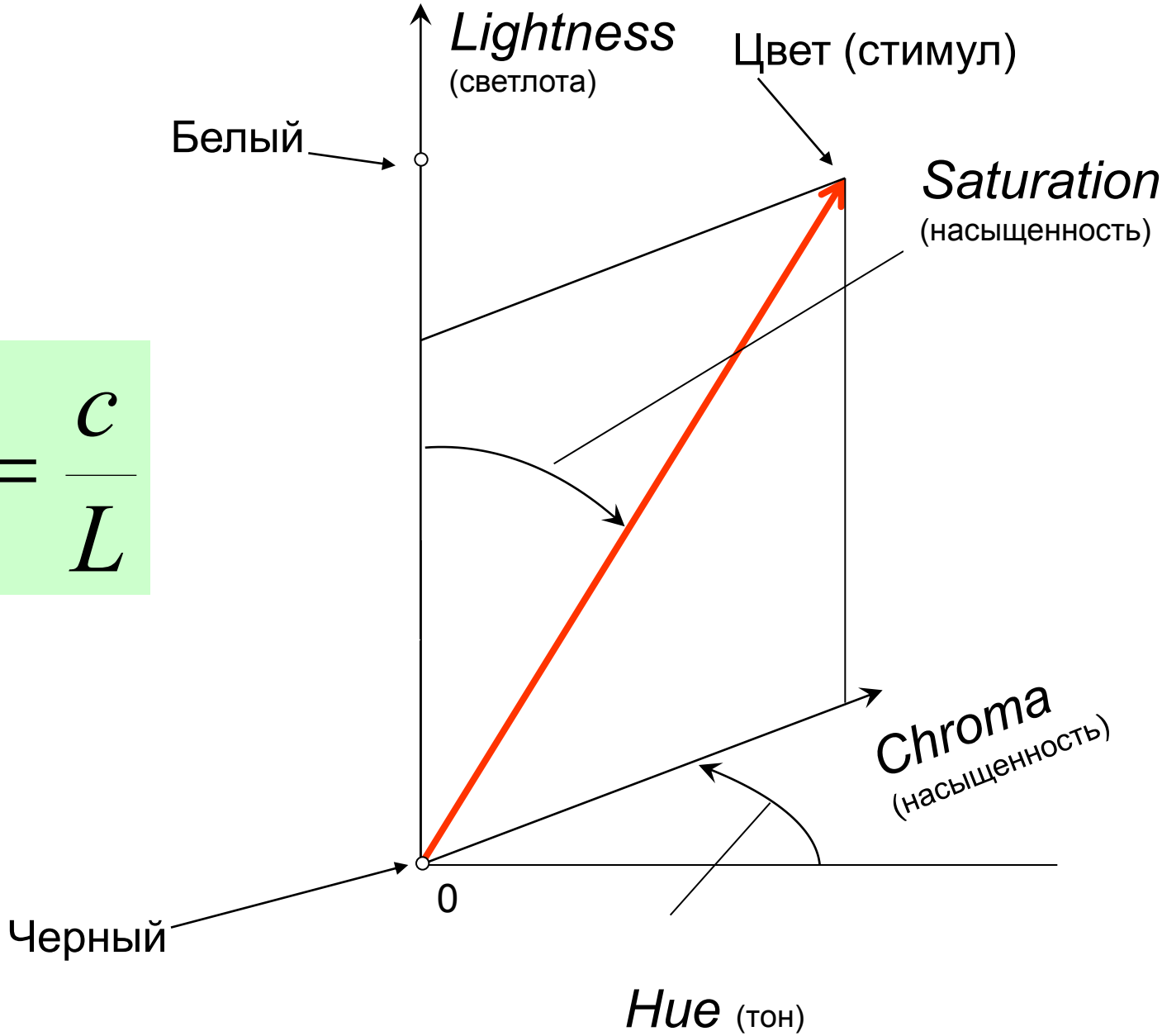
[1] *G. Wyszecki and W.S. Stiles* Color science: Concepts and methods, quantitative data and formulae, New York: Wiley, 1982 (2nd ed.), стр. 487

Насыщенность: saturation и chroma



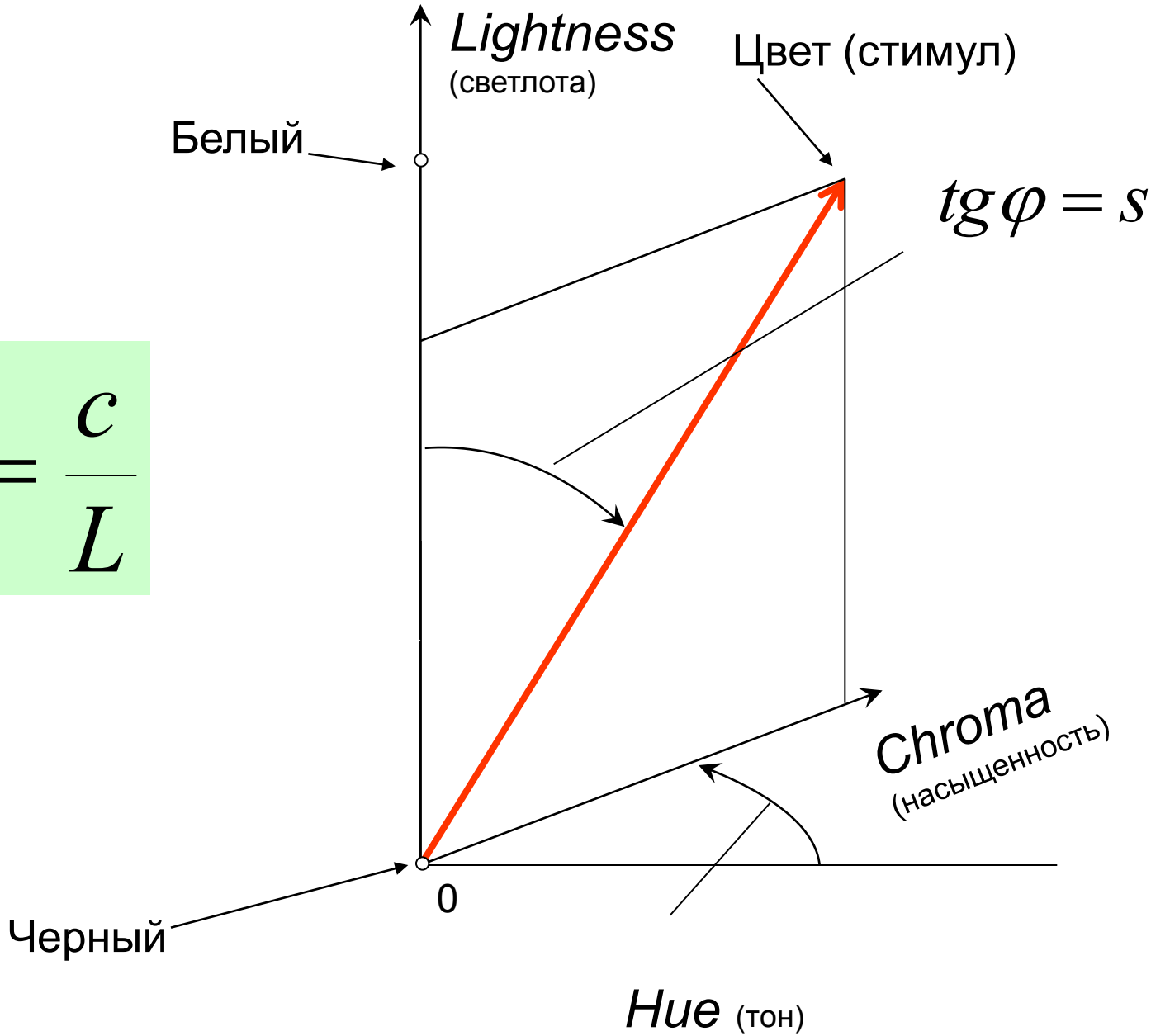
Насыщенность: saturation и chroma

$$s = \frac{c}{L}$$

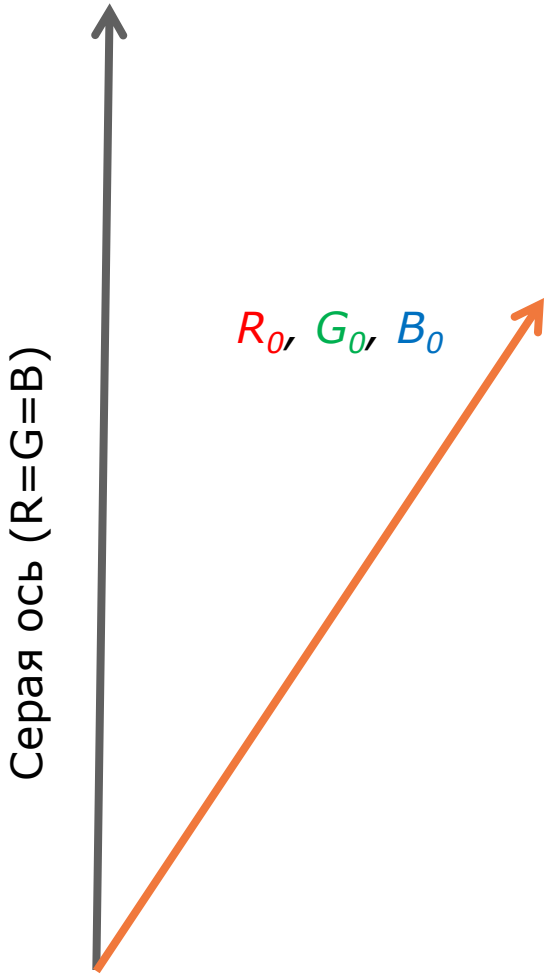


Насыщенность: saturation и chroma

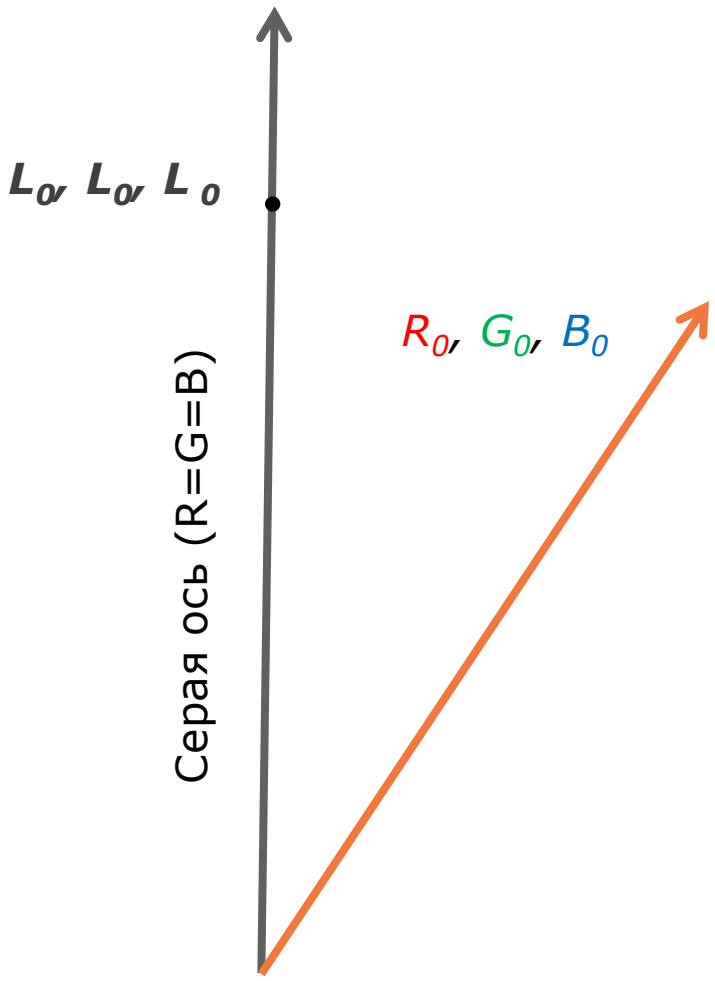
$$s = \frac{c}{L}$$



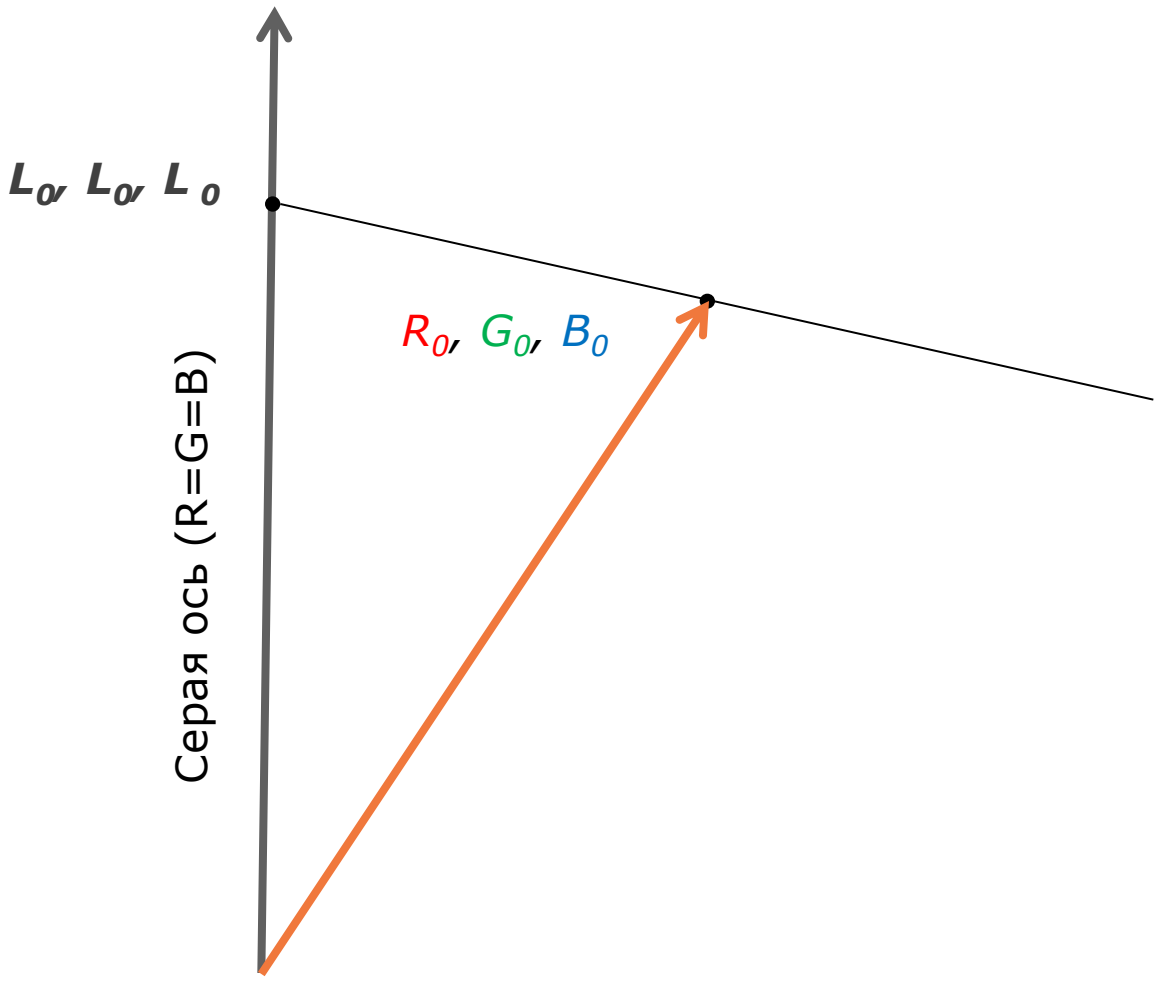
Изменение насыщенности



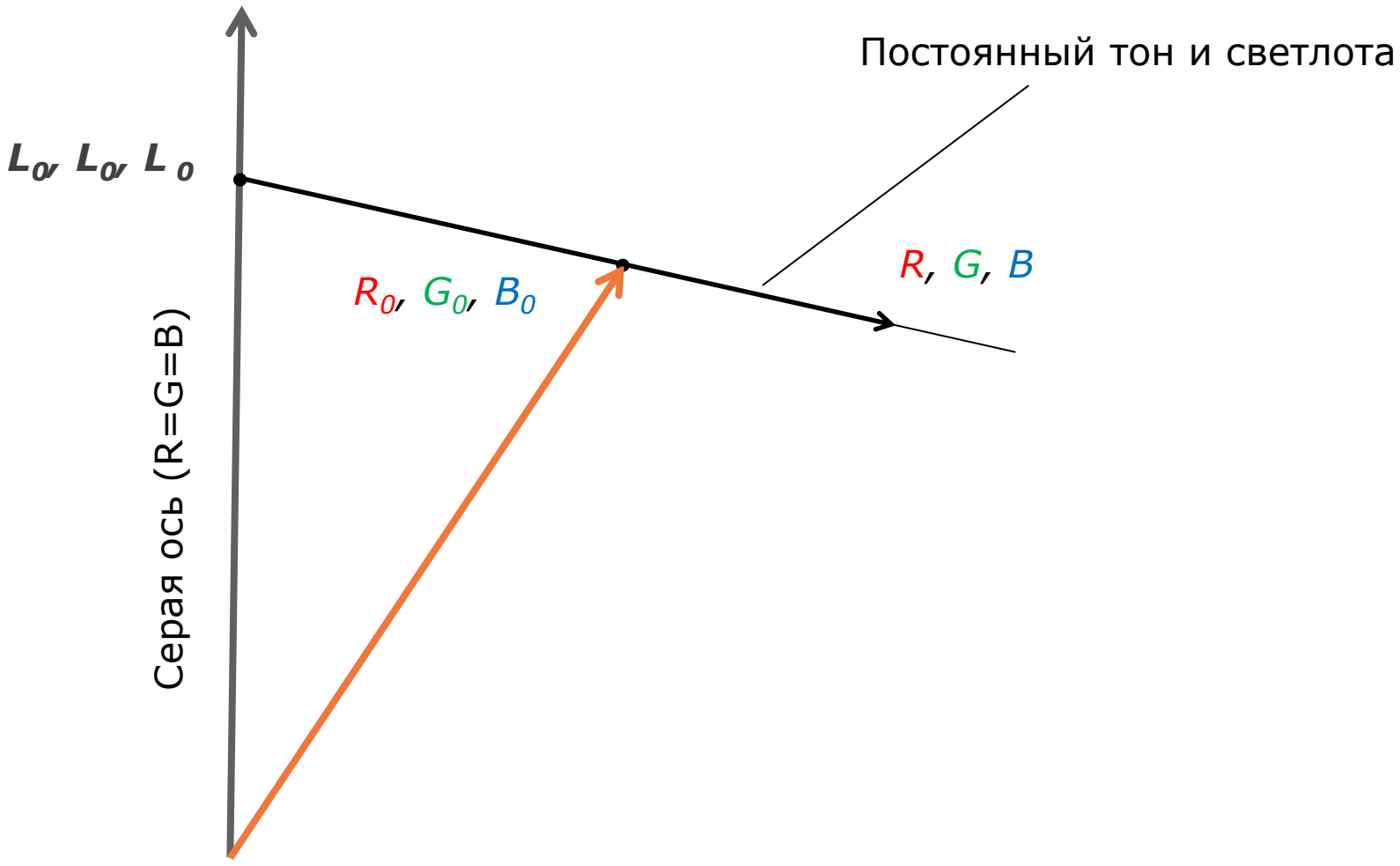
Изменение насыщенности



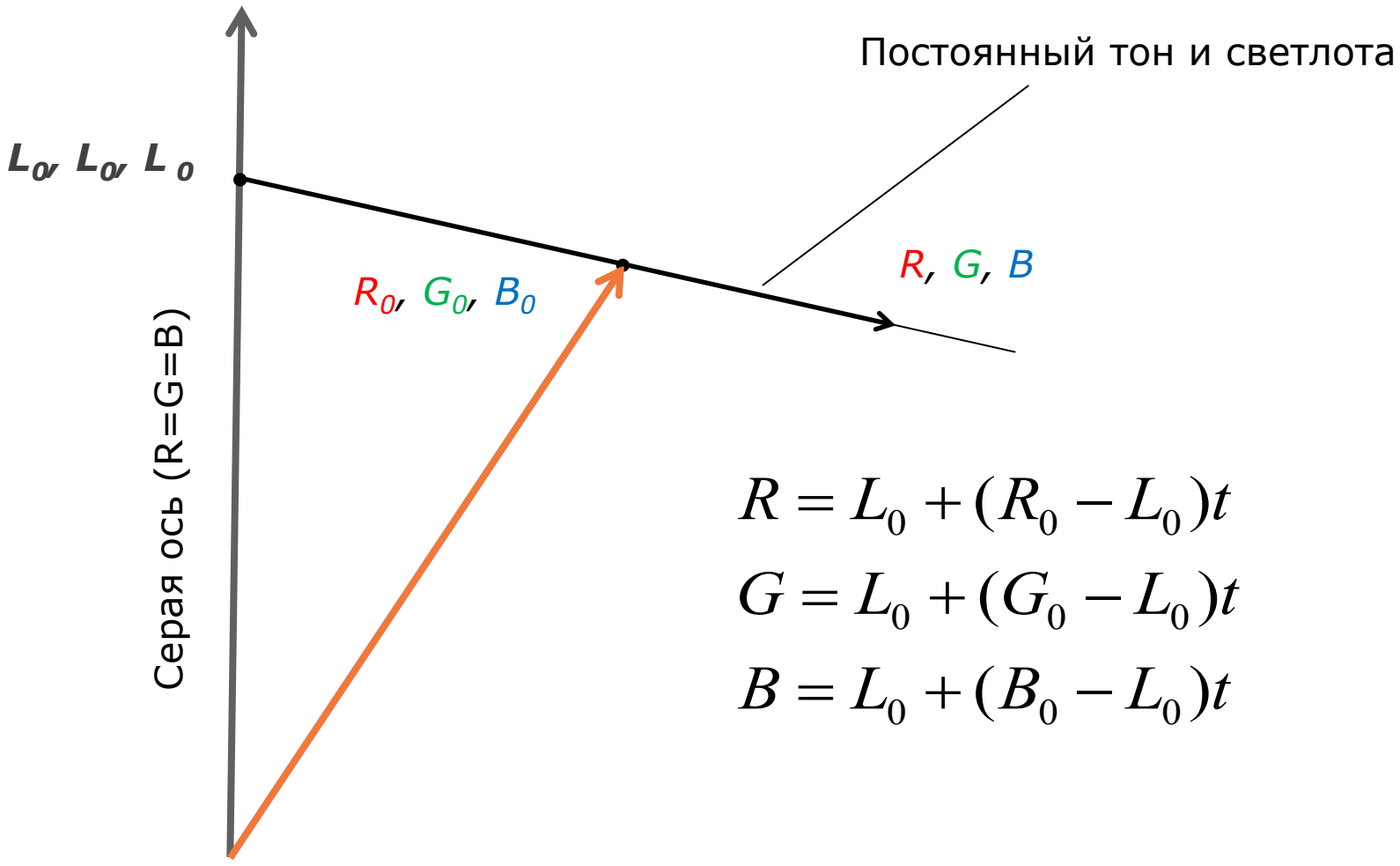
Изменение насыщенности



Изменение насыщенности

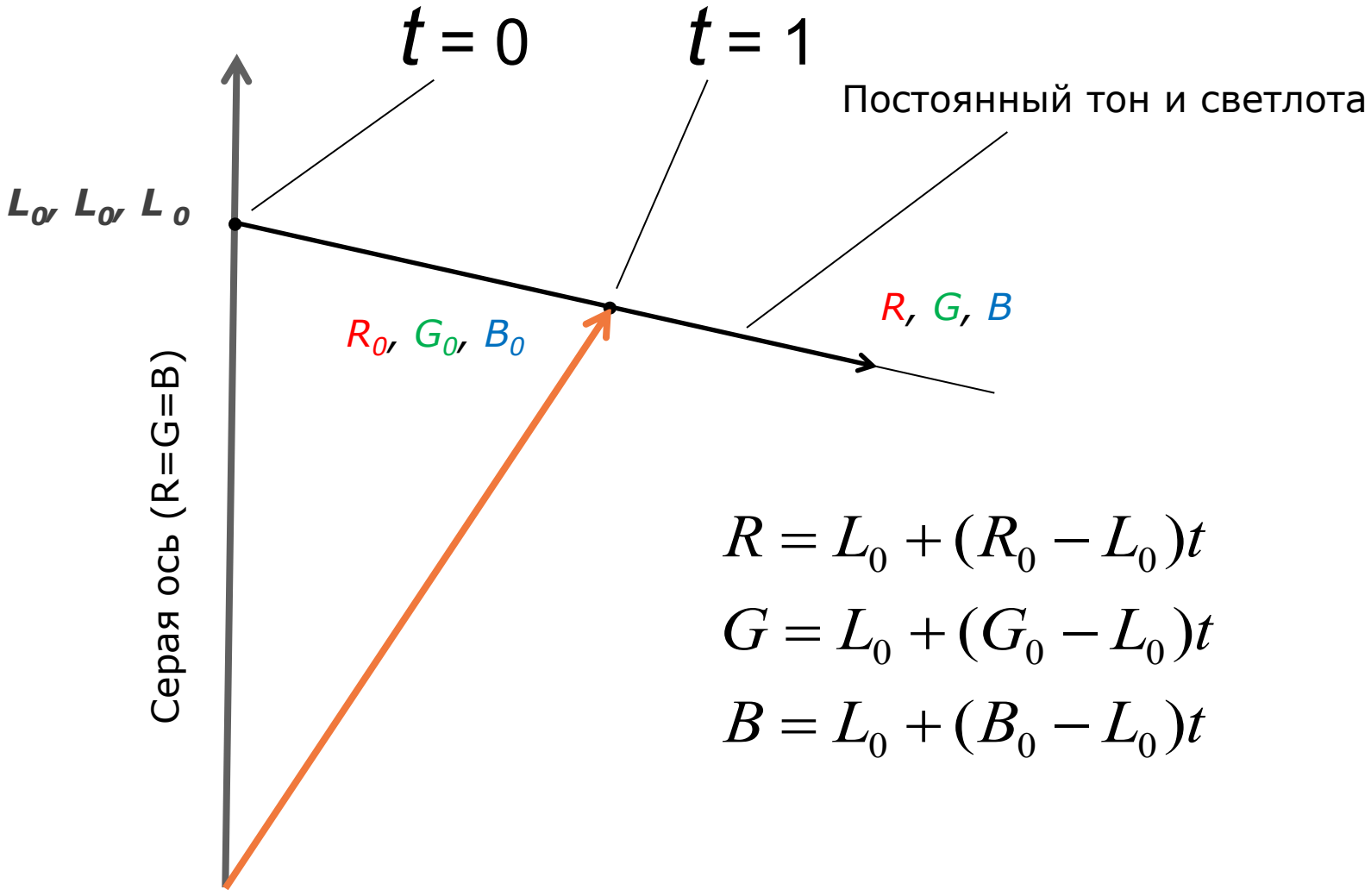


Изменение насыщенности



$$R = L_0 + (R_0 - L_0)t$$
$$G = L_0 + (G_0 - L_0)t$$
$$B = L_0 + (B_0 - L_0)t$$

Изменение насыщенности

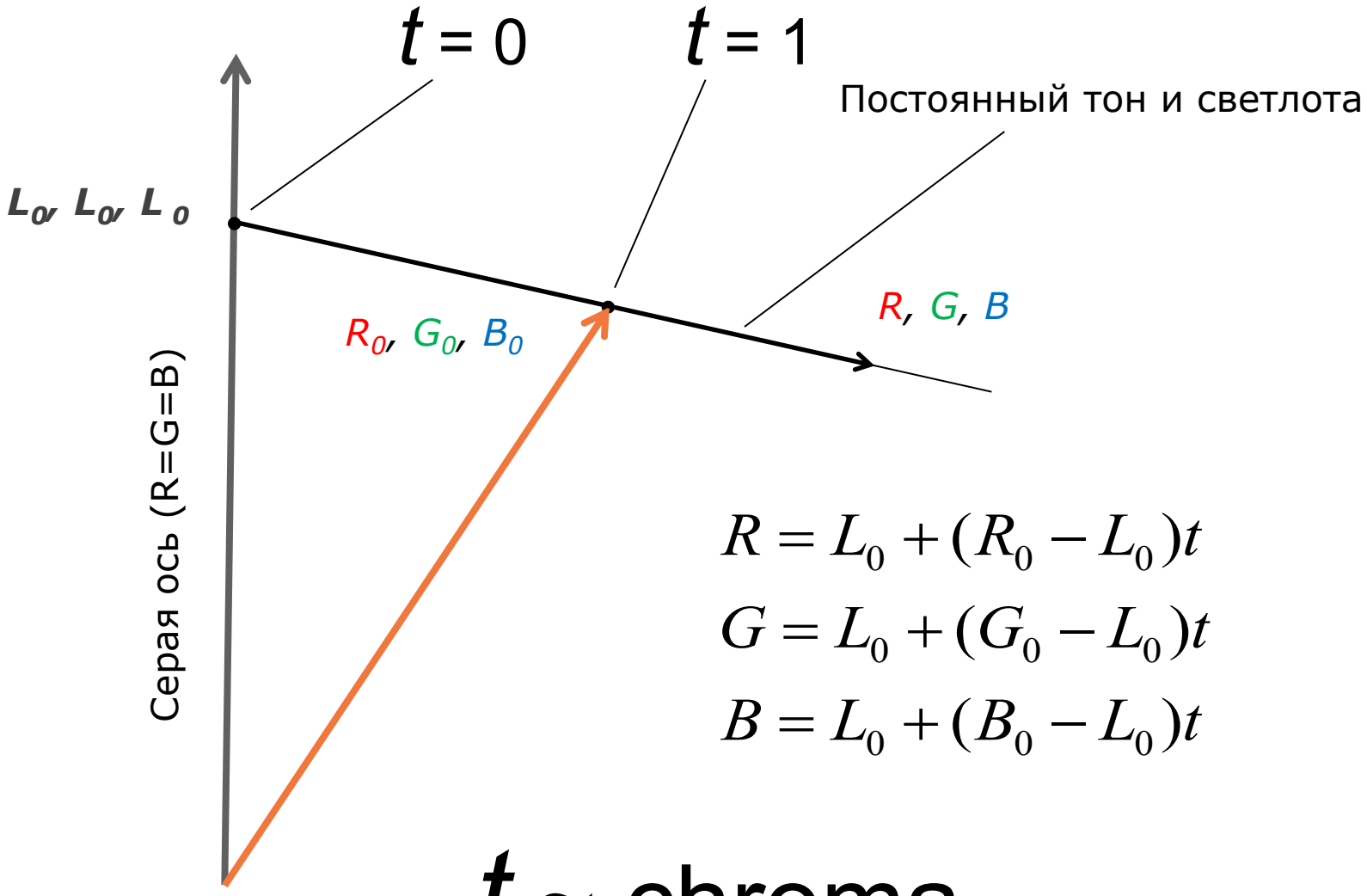


$$R = L_0 + (R_0 - L_0)t$$

$$G = L_0 + (G_0 - L_0)t$$

$$B = L_0 + (B_0 - L_0)t$$

Изменение насыщенности



$$R = L_0 + (R_0 - L_0)t$$
$$G = L_0 + (G_0 - L_0)t$$
$$B = L_0 + (B_0 - L_0)t$$

t ~ chroma

Максимальная и относительная насыщенность

$$R = L_0 + (R_0 - L_0)t$$

$$G = L_0 + (G_0 - L_0)t$$

$$B = L_0 + (B_0 - L_0)t$$

Максимальная и относительная насыщенность

$$R = L_0 + (R_0 - L_0)t$$

$$G = L_0 + (G_0 - L_0)t$$

$$B = L_0 + (B_0 - L_0)t$$

$$t > 0,\ t \uparrow$$

$$R = 1.0 \text{ или } G = 1.0 \text{ или } B = 1.0$$

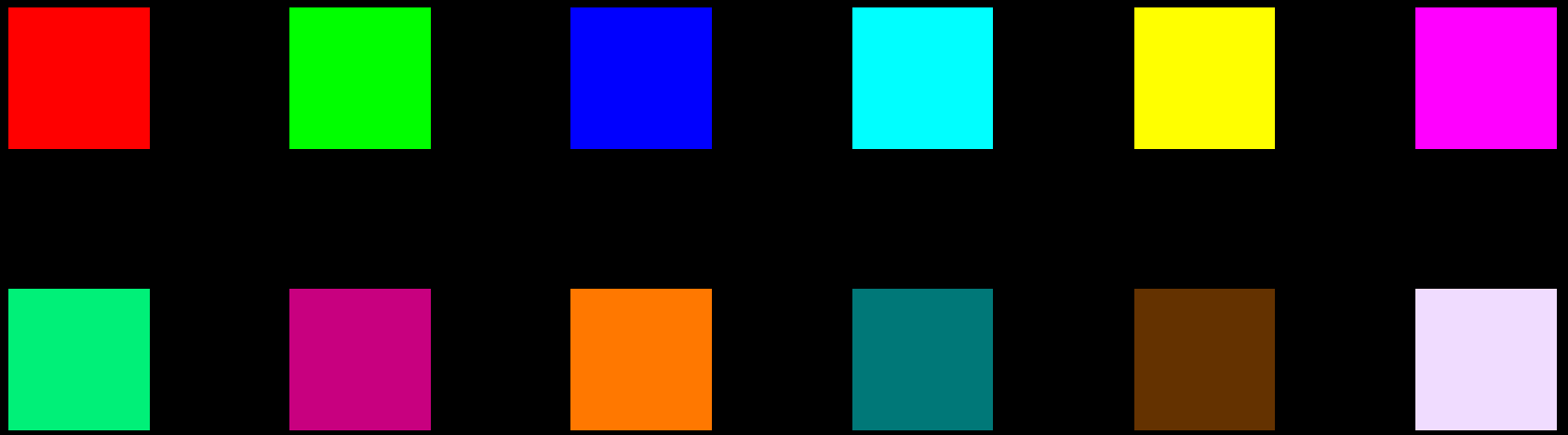
или

$$R = 0.0 \text{ или } G = 0.0 \text{ или } B = 0.0$$

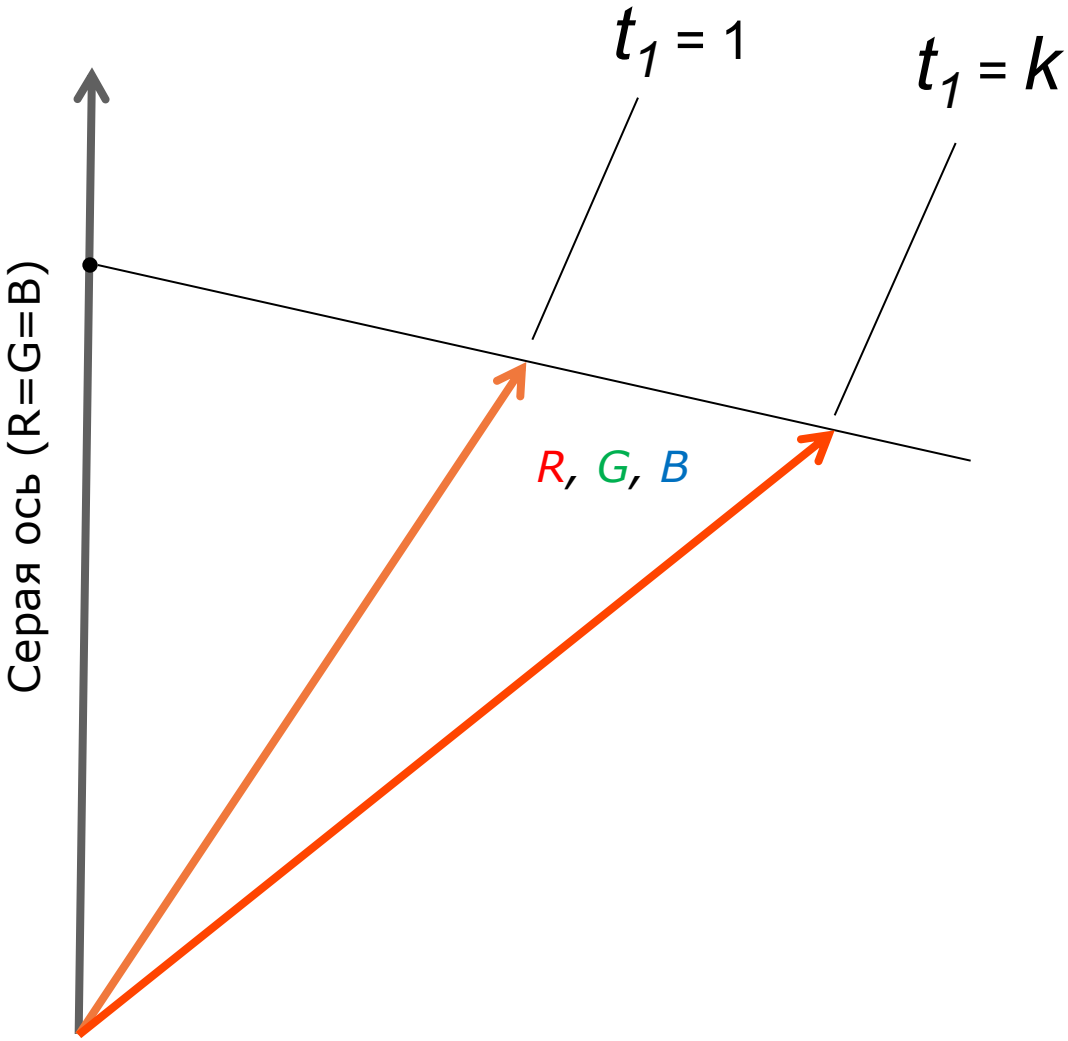
$$t = t_{\max}$$

$$chroma_{отн} = \frac{1}{t_{\max}}$$

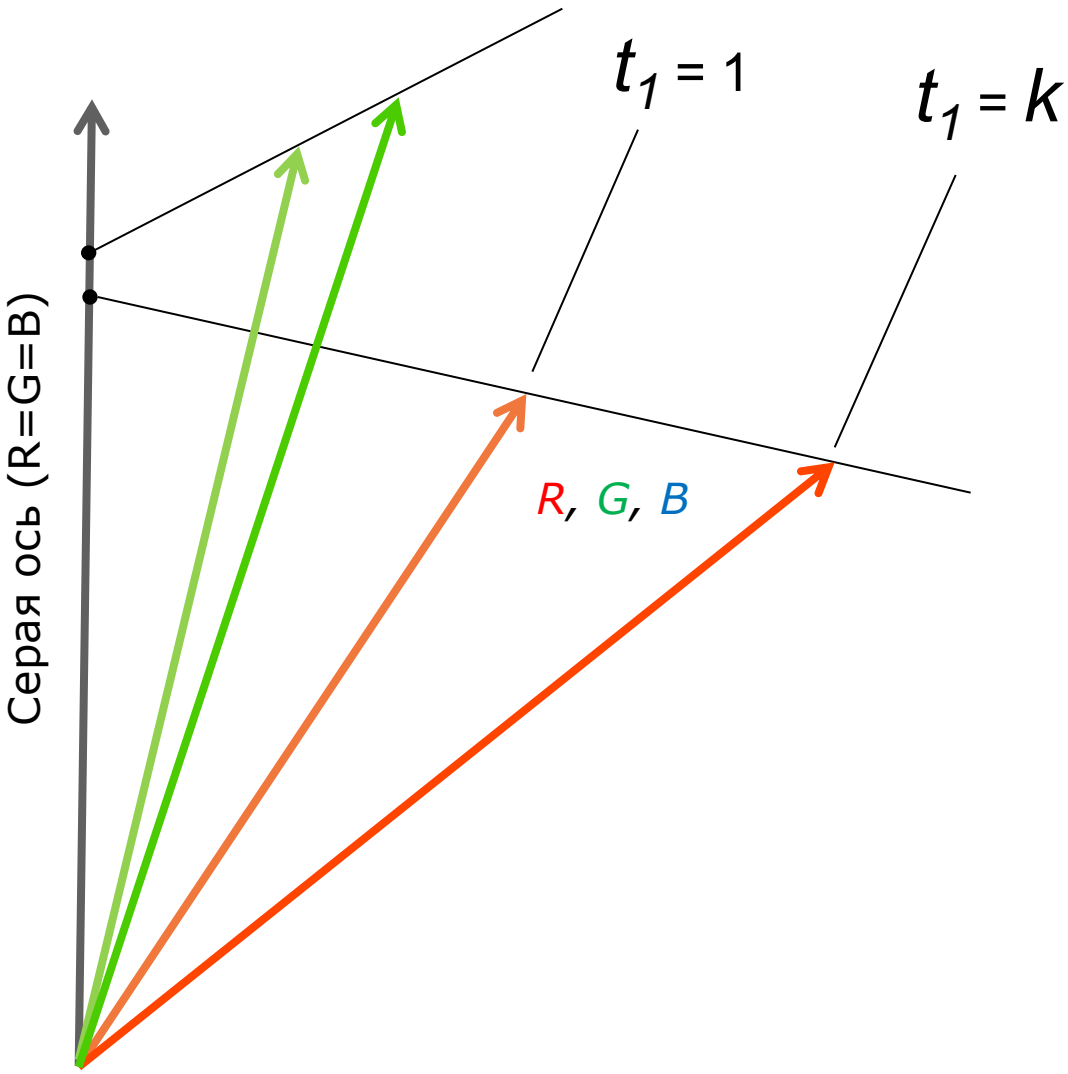
Максимальной насыщенностью обладают цвета, у которых равны нулю или максимуму одна или две цветовые координаты (из RGB)



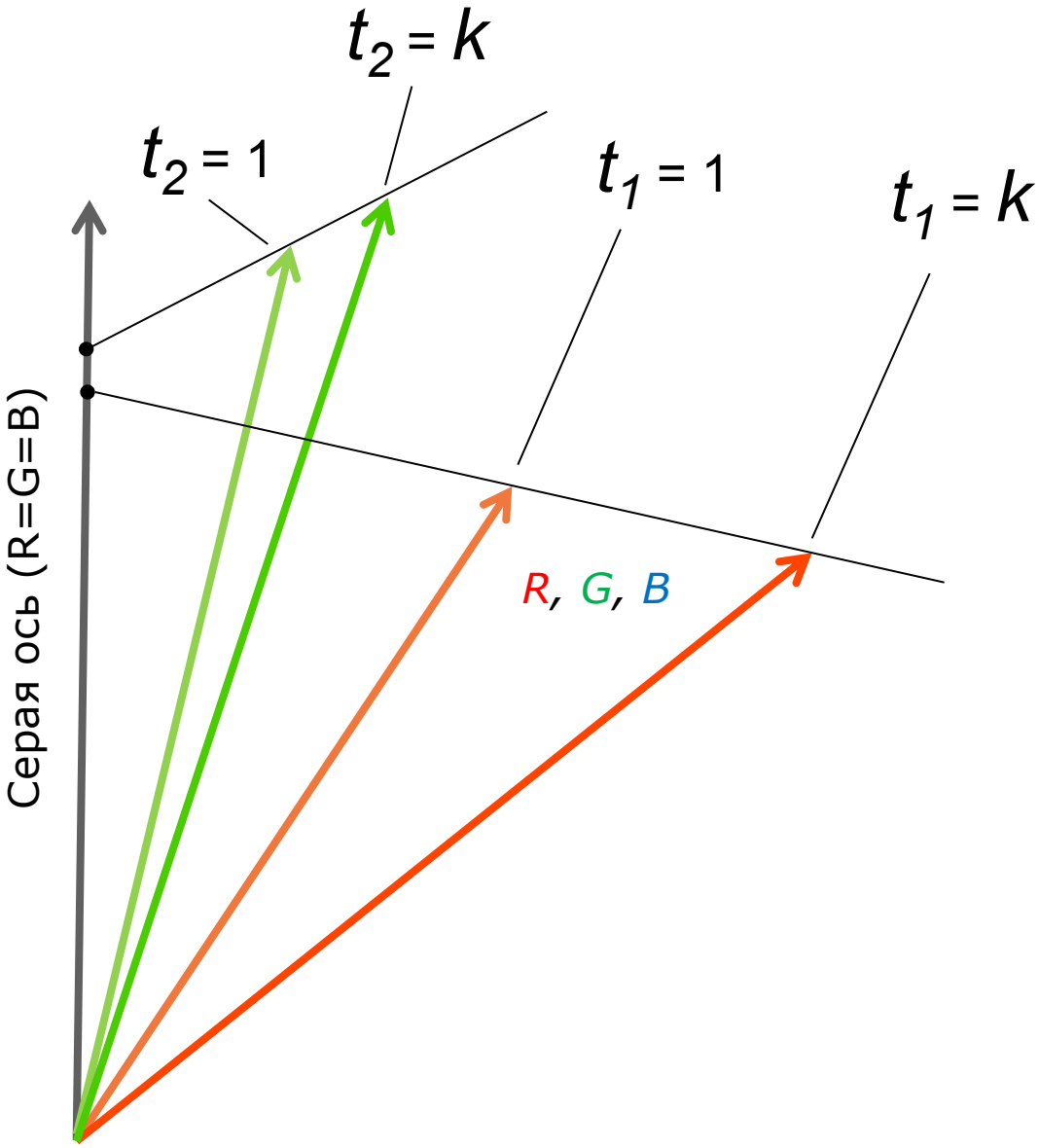
Изменение насыщенности в k раз

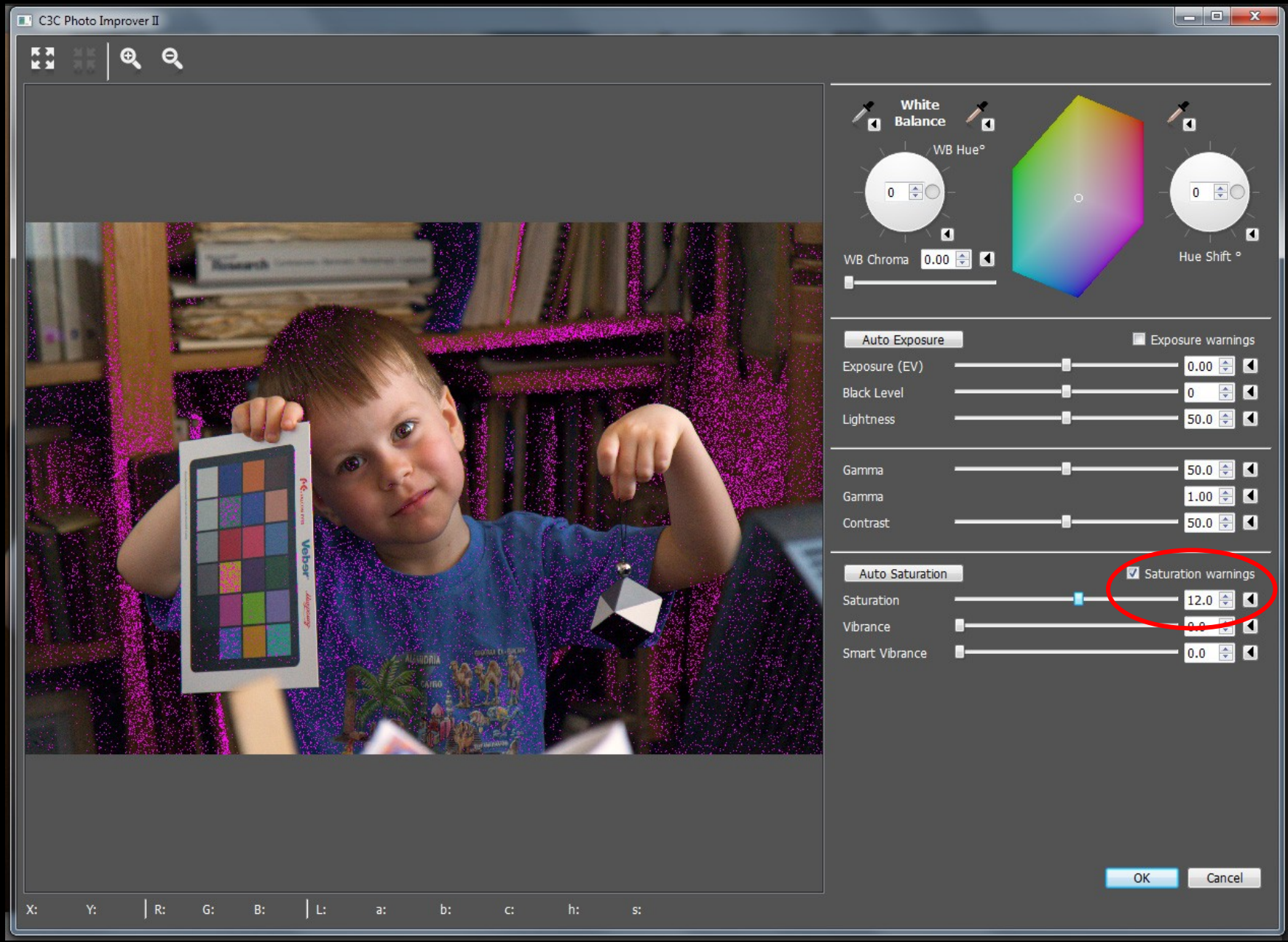


Изменение насыщенности в k раз

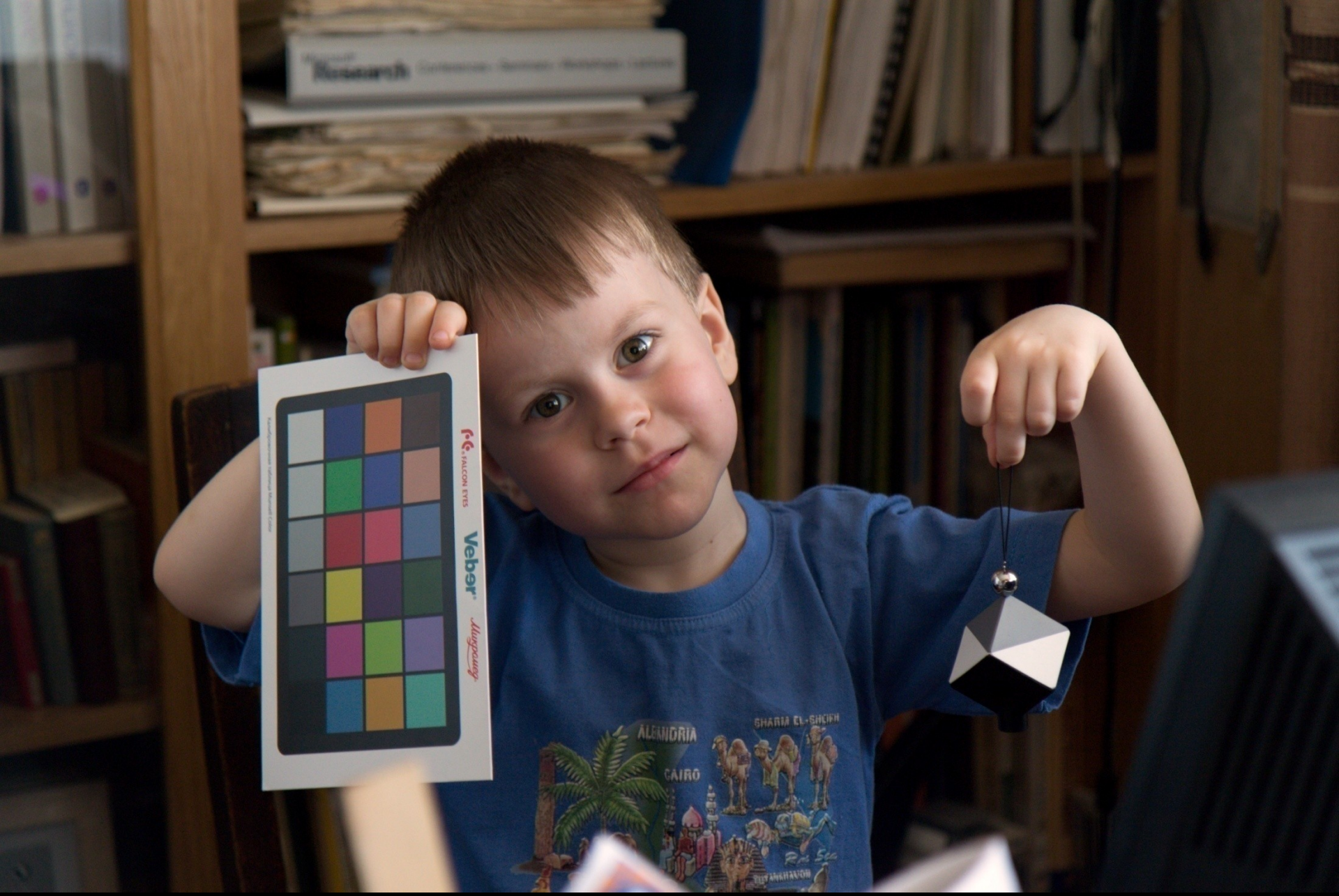


Изменение насыщенности в k раз

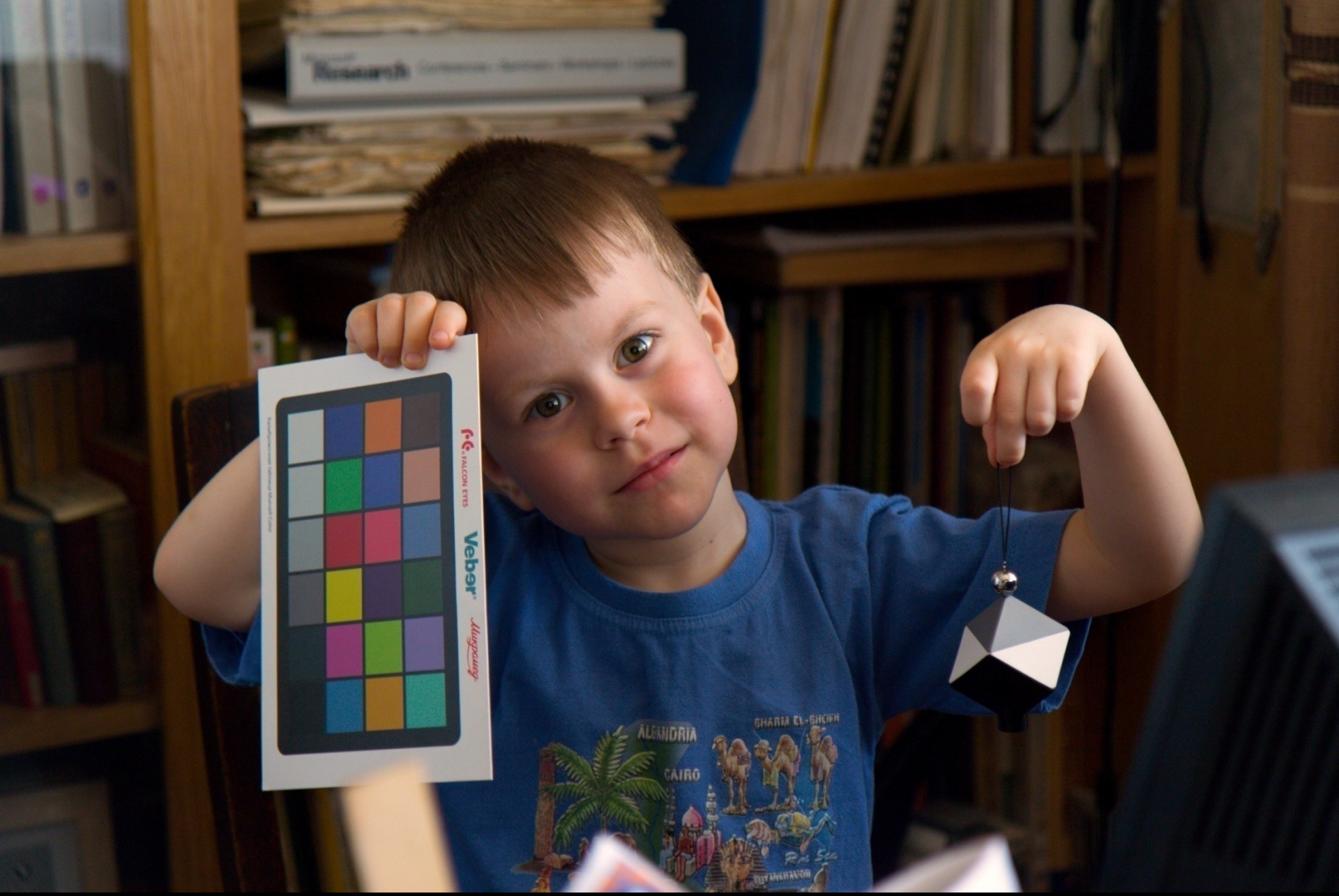




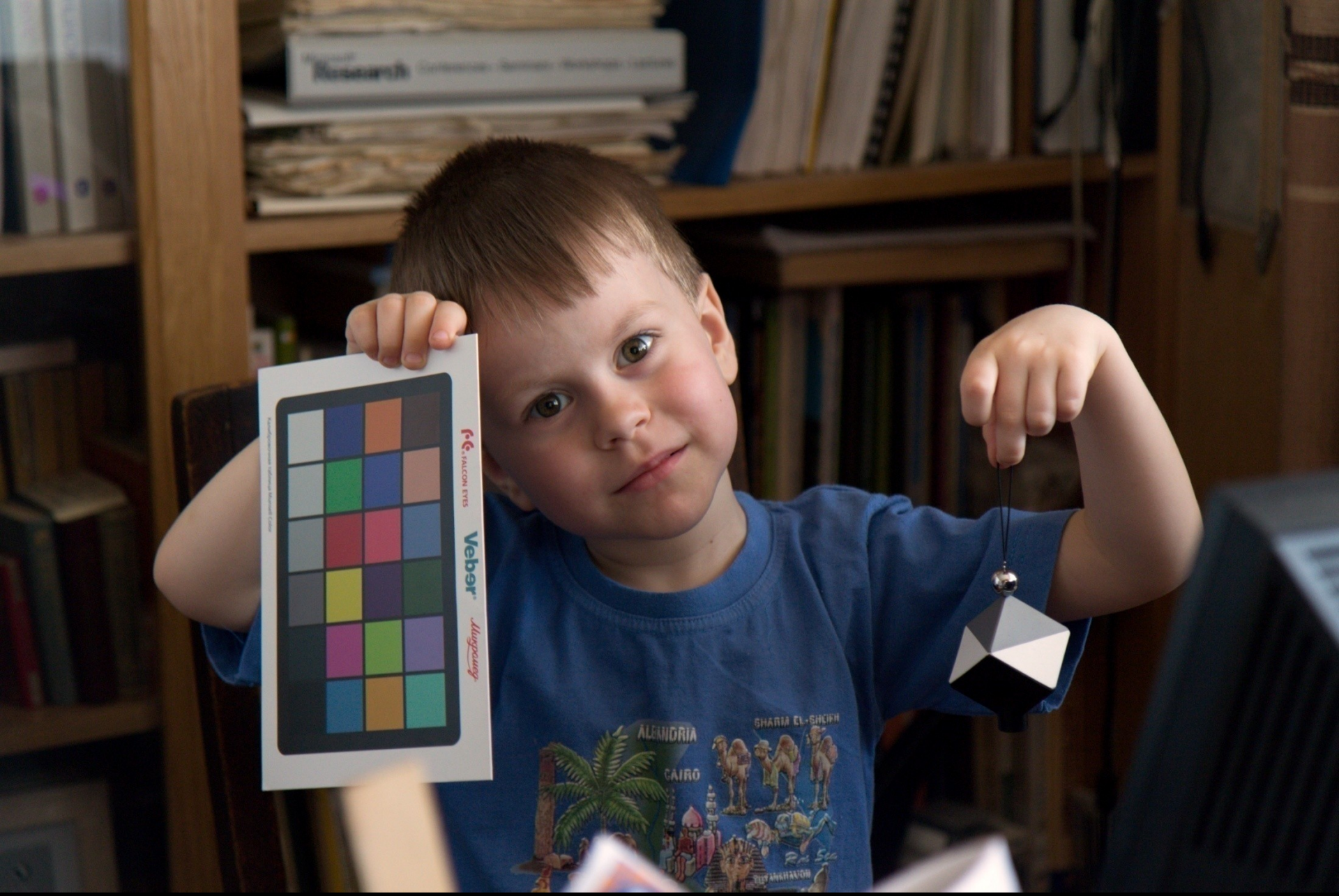
Увеличение насыщенности



Баланс белого по Spyder CUBE, +0.15EV, Black Level +8



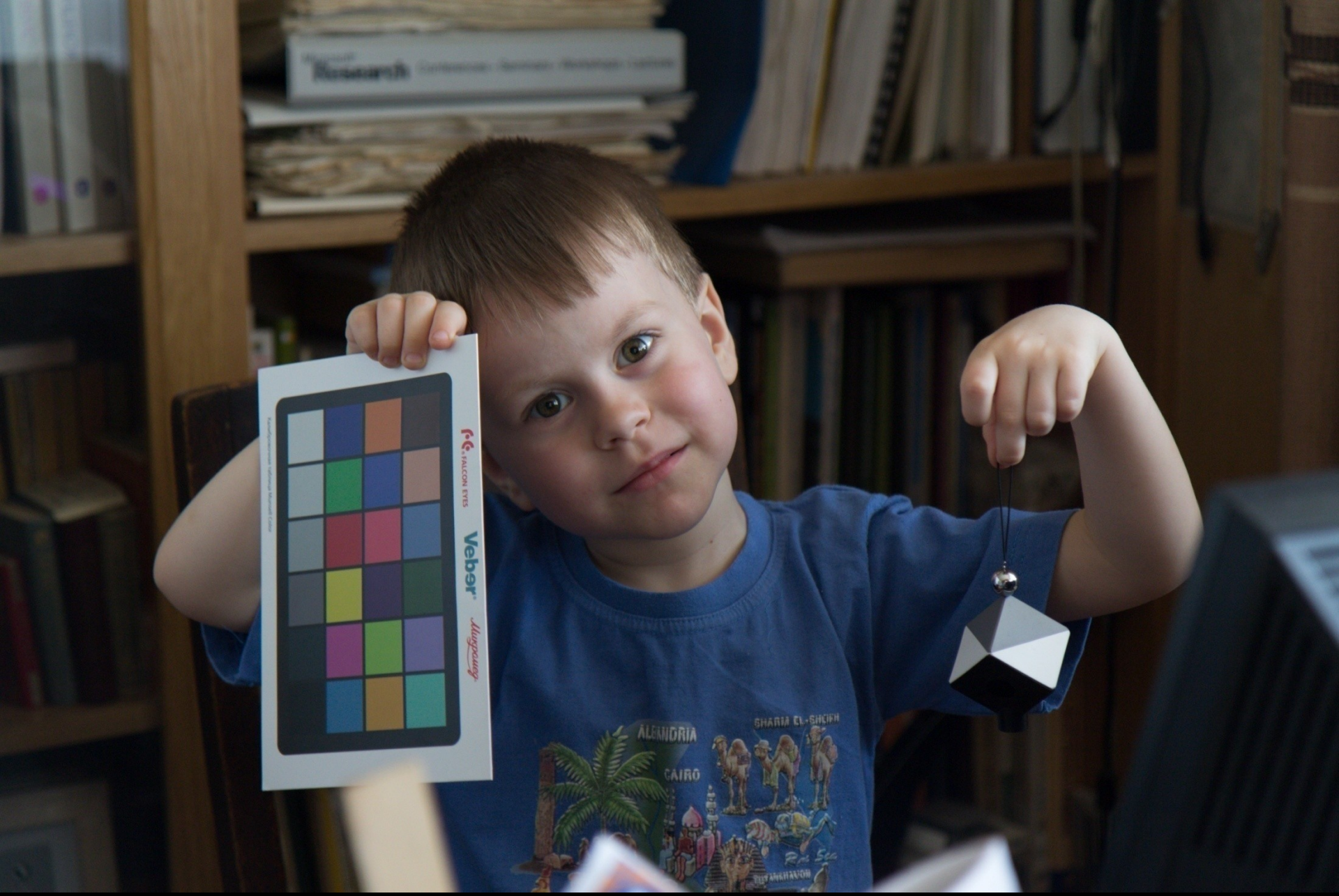
Баланс белого по Spyder CUBE, +0.15EV, Black Level +8, Saturation +12 (x1.23)



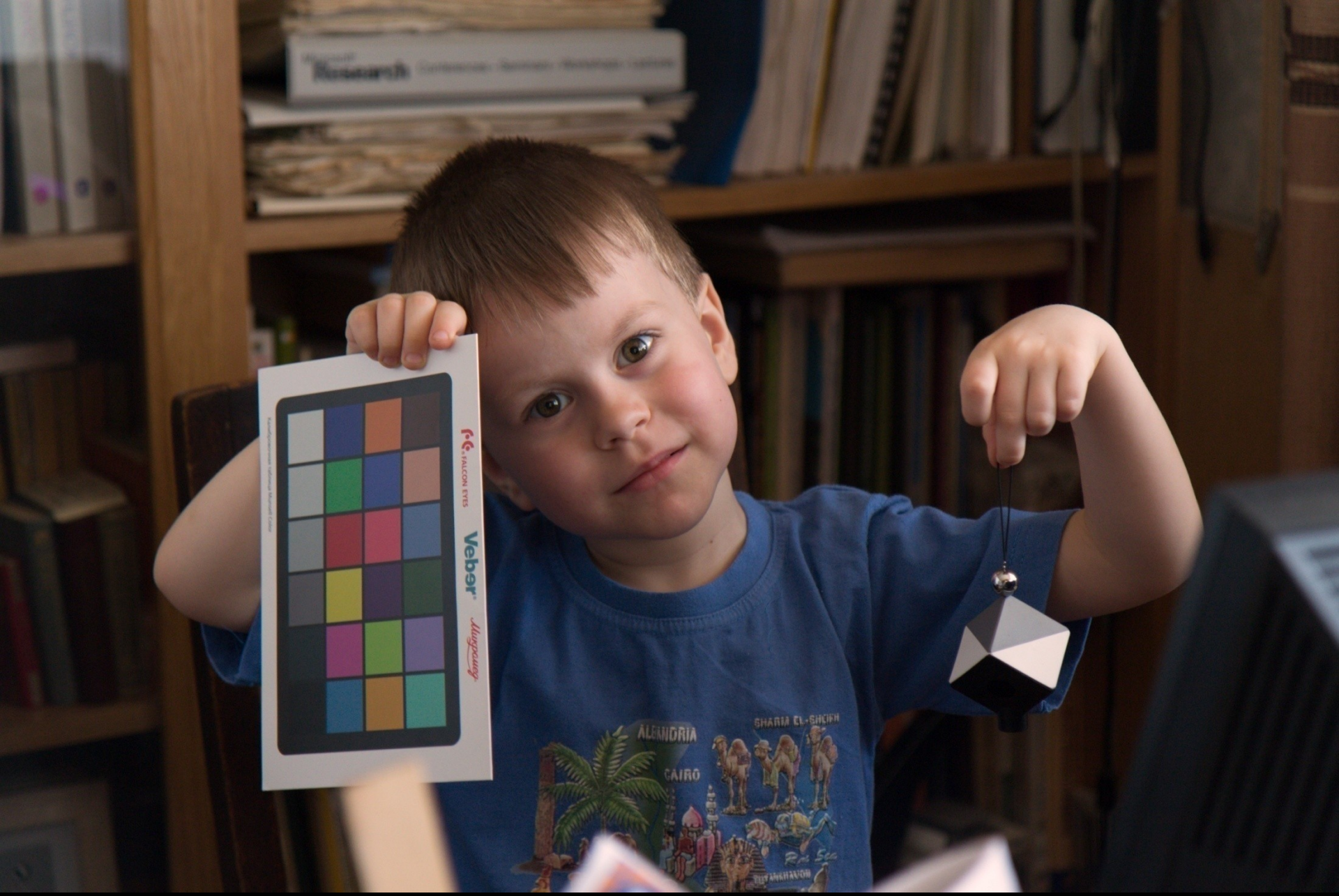
Баланс белого по Spyder CUBE, +0.15EV, Black Level +8

Баланс белого по Spyder CUBE, +0.15EV, Black Level +8, Saturation -20 (x1/1.41)

Баланс белого по Spyder CUBE, +0.15EV, Black Level +8, Saturation -40 (x1/2)



Как снято



Баланс белого по Spyder CUBE

Линейные цветовые преобразования

1. **Изменение экспозиции.** Масштабирование по осям *R, G, B* или *L, p, y* (одинаковый масштаб по осям). Меняются *L*, chroma.

2. **Удаление вуали (паразитной засветки).** Вычитание одной и той же величины из значений *R, G, B*. Уменьшение *L* на постоянную величину. Меняются *L*, *saturatio*n.

3. **Цветовой баланс (баланс белого).** Масштабирование по осям *R, G, B* (разный масштаб по осям). Меняются *тон*, *chroma, saturation*.

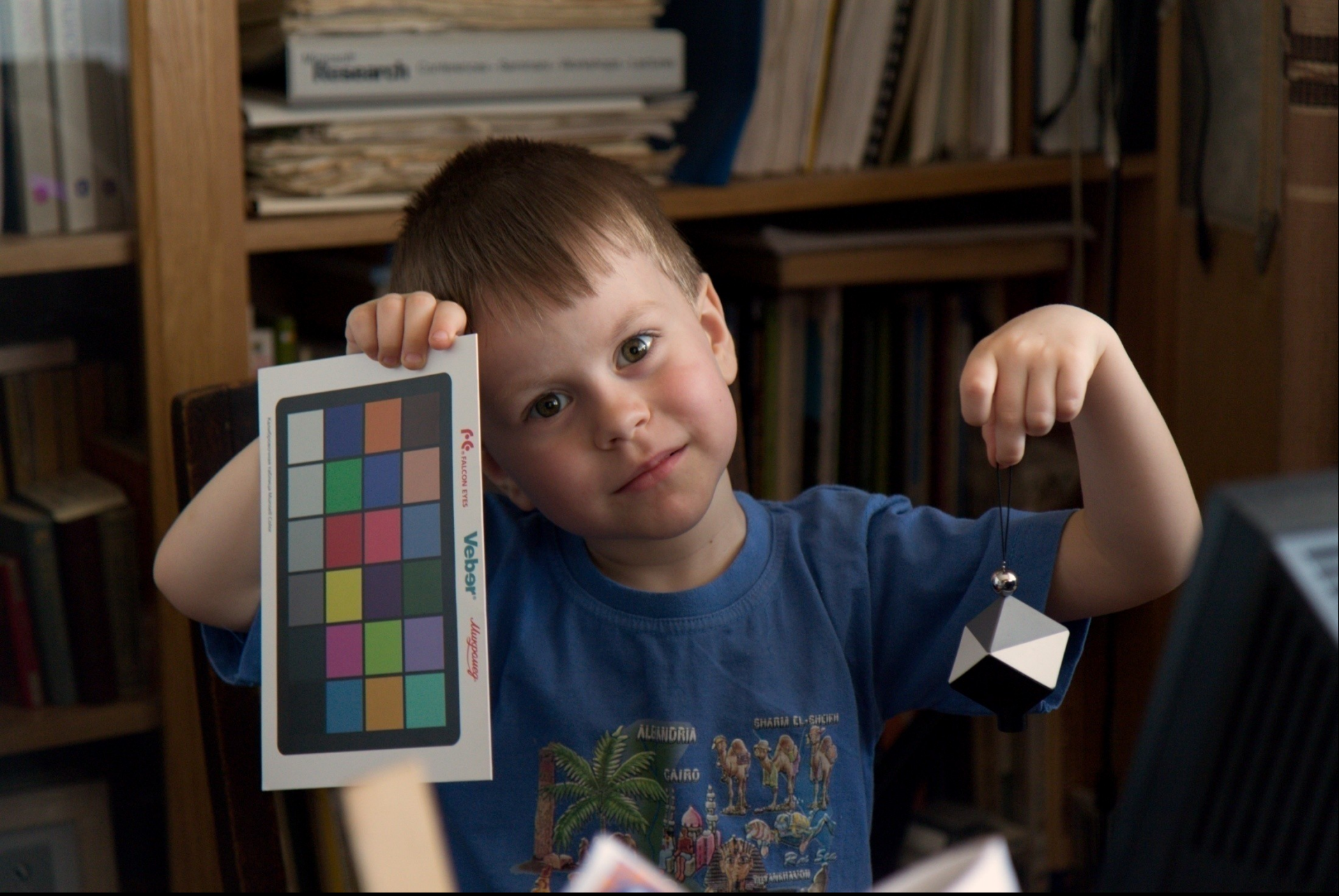
4. **Изменение насыщенности**. Масштабирование по осям *p, y* (одинаковый масштаб по осям). Меняются *chroma* и *saturation*.

5. **Изменение светлоты.** Масштабирование по оси *L*. Изменяются *L, saturation.*

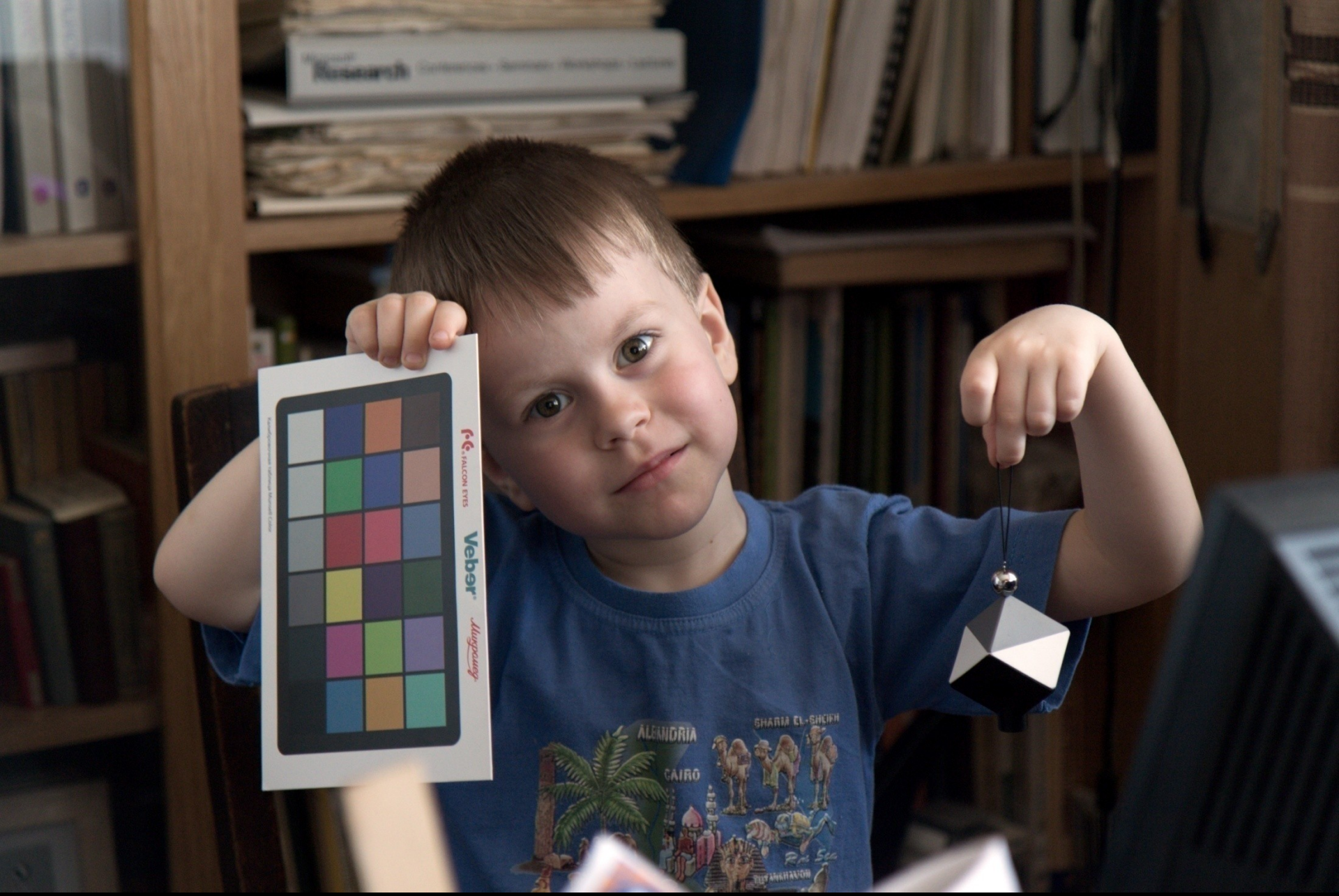
6. **Сдвиг тона.** Поворот вокруг оси *L*. Изменяется *тон*.

Lightness

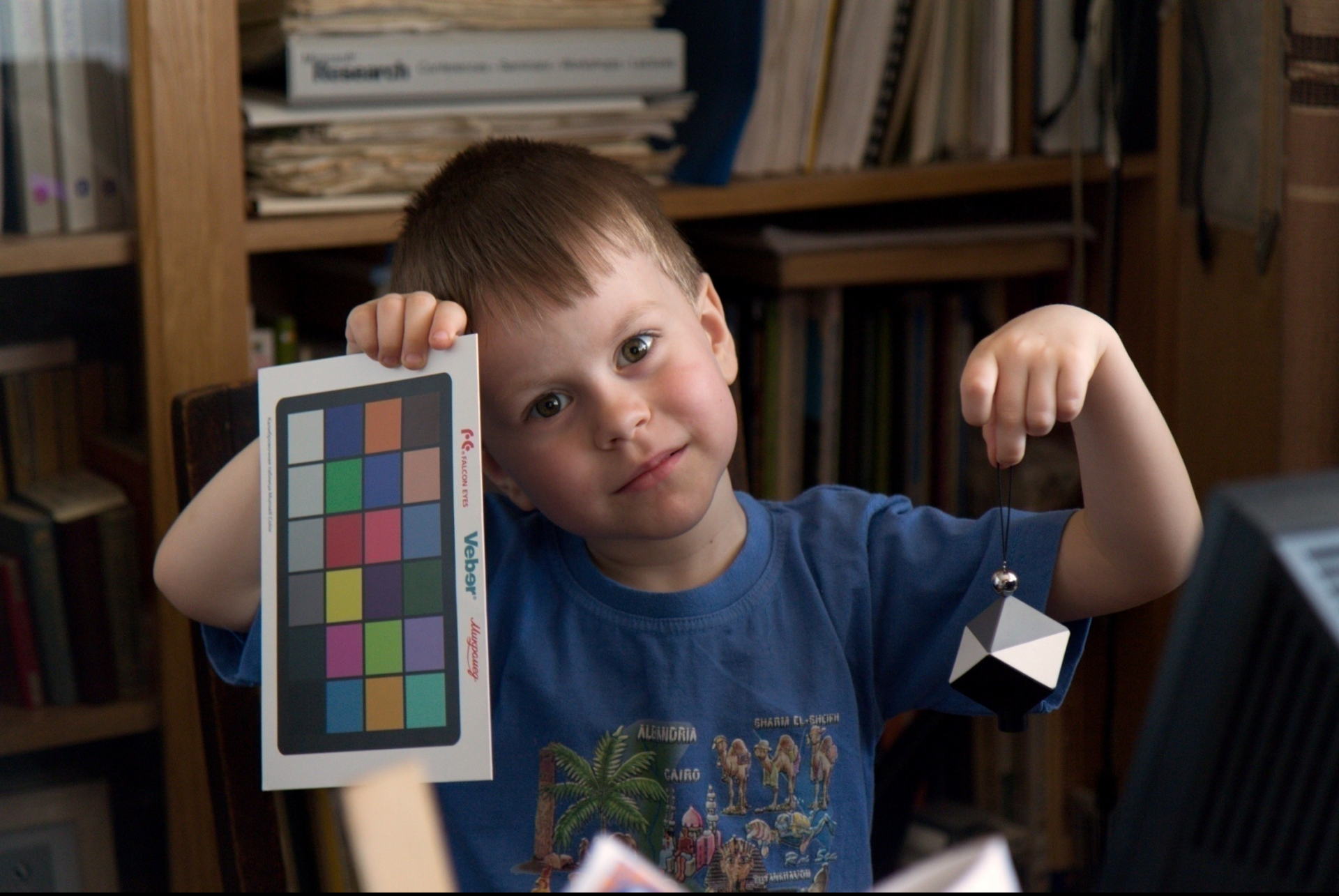
Изменение светлоты. Умножение *L* на постоянный коэффициент ≥ 0. Изменяются *L, saturation.* Не меняется *chroma.*



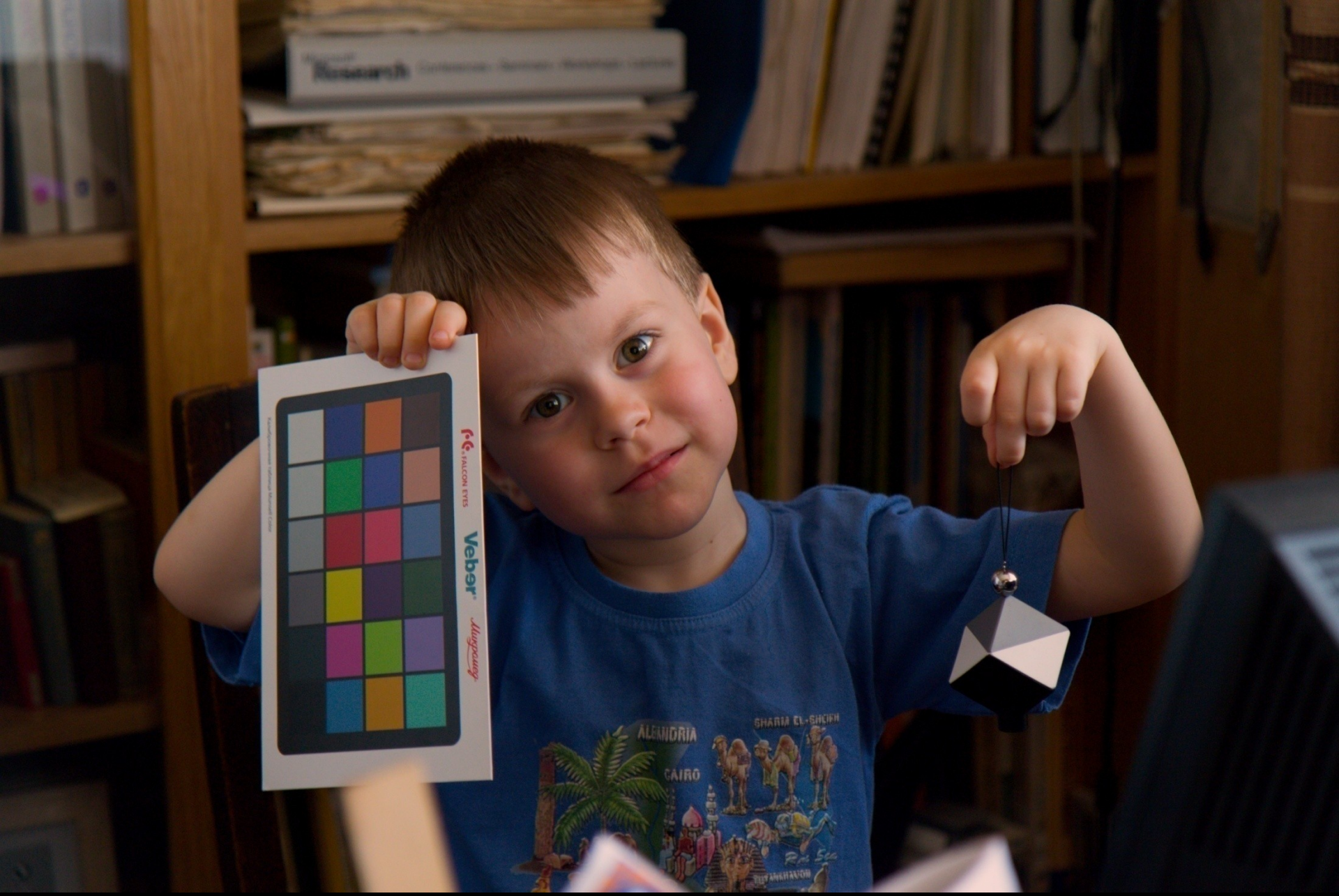
Lightness 50



Lightness 60



Lightness 50



Lightness 40

Exposure, Lightness, Saturation

Exposure, Lightness, Saturation

Exposure+ ~ Lightness+ &
Saturation+

Exposure, Lightness, Saturation

Exposure+ ~ Lightness+ & Saturation+

Exposure- ~ Lightness- & Saturation-

Exposure, Lightness, Saturation

Exposure+ ~ Lightness+ & Saturation+

Exposure- ~ Lightness- & Saturation-

Третий лишний ☺

Сдвиг тона
(поворот вокруг оси *L*)

В усадьбе Н. А. Некрасова Карабиха
Исходный

В усадьбе Н. А. Некрасова Карабиха
Сдвиг тона 180°

Базовые (линейные) цветовые преобразования

Все вычисления в линейном (RGB) пространстве

1. **Изменение экспозиции.** Увеличение или уменьшение количества световой энергии при неизменном спектре. Изменяется светлота. Не изменяются тон и насыщенность (saturation).

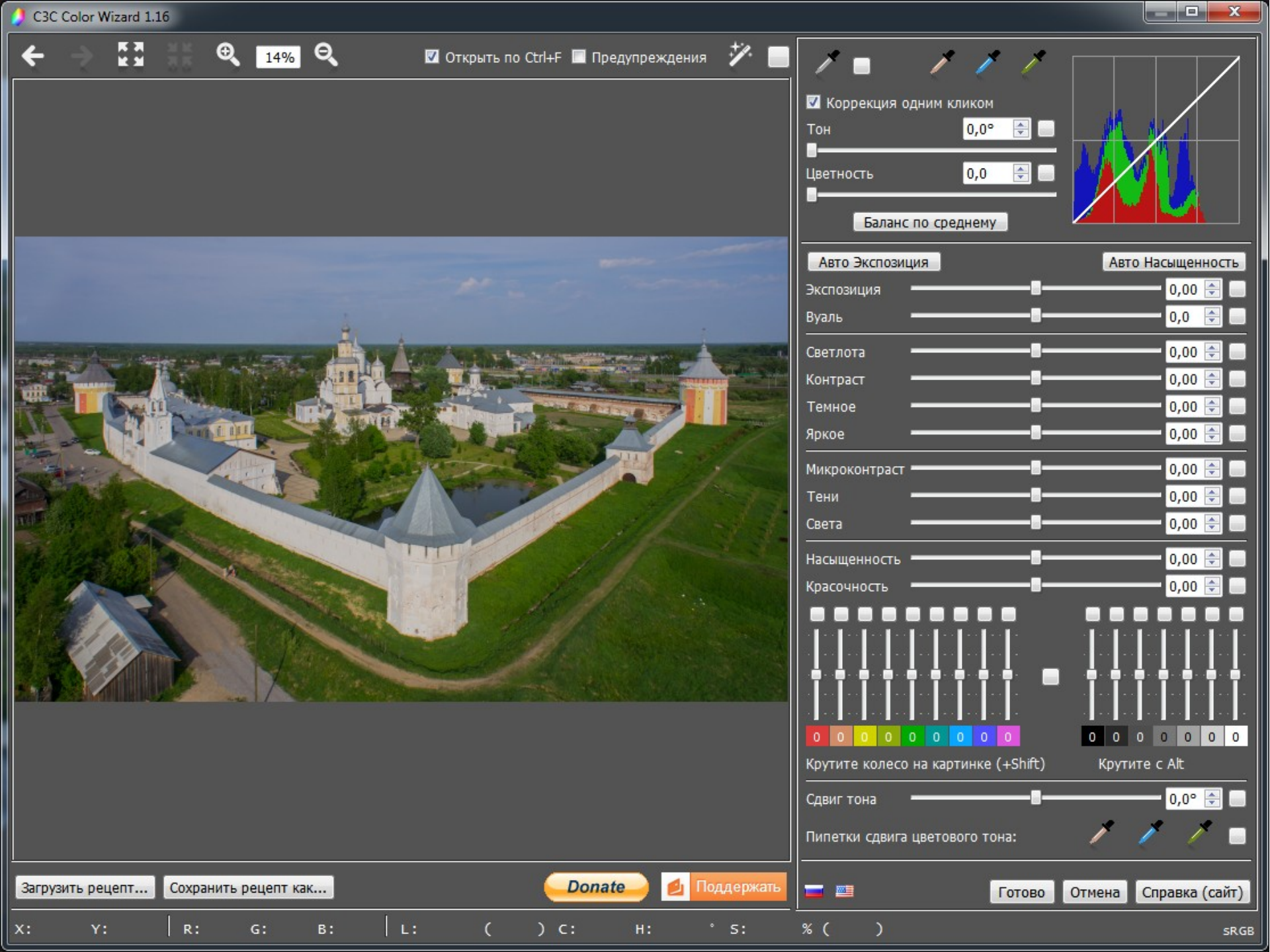
2. **Удаление вуали (паразитной засветки).** Вычитание постоянной составляющей (установка уровня черного). Меняются светлота и насыщенность при неизменном цветовом тоне.

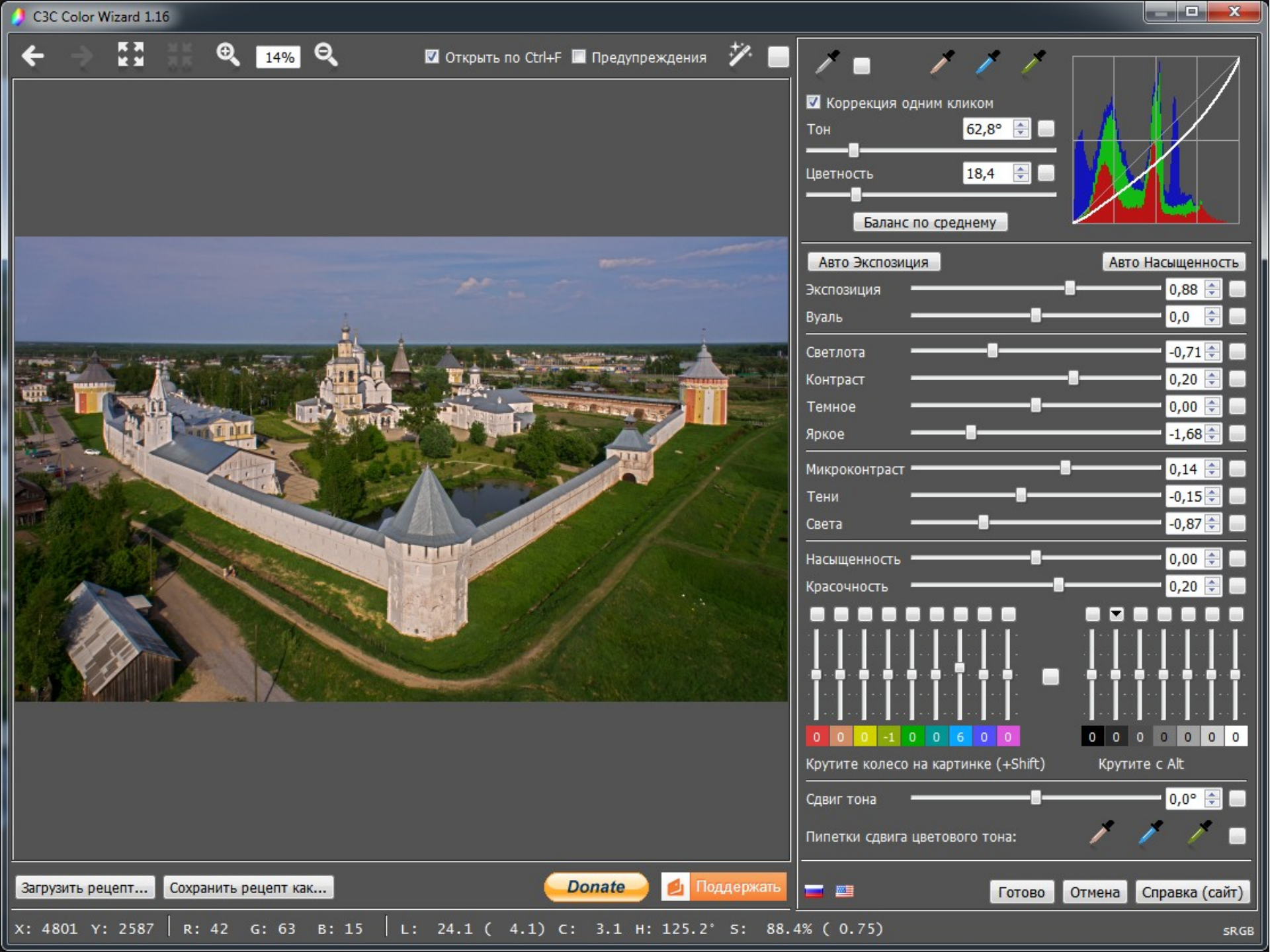
3. **Цветовой баланс (баланс белого).** Изменение цветового тона и насыщенности с помощью преобразований, имеющих физический смысл.

4. **Изменение насыщенности**. Увеличение или уменьшение насыщенности (saturation, chroma) при неизменном тоне и светлоте.

5. **Изменение светлоты.** Сохраняются тон и насыщенность (chroma).

6. **Сдвиг тона.** Изменение (циклическое) цветового тона при неизменных насыщенности (chroma, saturation) и светлоты.


C3C Color Wizard 1.16
14%
Открыть по Ctrl+F
Предупреждения
Коррекция одним кликом
Тон
0,0°
Цветность
0,0
Баланс по среднему
Авто Экспозиция
Авто Насыщенность
Экспозиция
0,00
Вуаль
0,0
Светлота
Контраст
Темное
Яркое
Микроконтраст
Тени
Света
Насыщенность
Красочность
Крутите колесо на картинке (+Shift)
Крутите с Alt
Сдвиг тона
0,0°
Пипетки сдвига цветового тона:
Загрузить рецепт...
Сохранить рецепт как...
Donate
Поддержать
Готово
Отмена
Справка (сайт)
X:
Y:
R:
G:
B:
L:
C:
H:
S:
sRGB


C3C Color Wizard 1.16
14%
Открыть по Ctrl+F
Предупреждения
Коррекция одним кликом
Тон
62,8°
Цветность
18,4
Баланс по среднему
Авто Экспозиция
Авто Насыщенность
Экспозиция
0,88
Вуаль
0,0
Светлота
-0,71
Контраст
0,20
Темное
0,00
Яркое
-1,68
Микроконтраст
0,14
Тени
-0,15
Света
-0,87
Насыщенность
0,00
Красочность
0,20
0 0 0 -1 0 0 6 0 0
0 0 0 0 0 0 0
Крутите колесо на картинке (+Shift)
Крутите с Alt
Сдвиг тона
0,0°
Пипетки сдвига цветового тона:
Загрузить рецепт...
Сохранить рецепт как...
Donate
Поддержать
Готово
Отмена
Справка (сайт)
X: 4801 Y: 2587
R: 42 G: 63 B: 15
L: 24.1 (4.1) C: 3.1 H: 125.2° S: 88.4% (0.75)
sRGB

C3C Color Wizard 1.16

14% Открыть по Ctrl+F Предупреждения

Линейные преобразования

Тон 62,8°

Цветность 18,4

Баланс по среднему

Авто Экспозиция Авто Насыщенность

Экспозиция 0,88

Вуаль 0,0

Светлота -0,71

Контраст 0,20

Темное 0,00

Яркое -1,68

Микроконтраст 0,14

Тени -0,15

Света -0,87

Насыщенность 0,00

Красочность 0,20

0 0 0 -1 0 0 6 0 0

0 0 0 0 0 0 0

Крутите колесо на картинке (+Shift) Крутите с Alt

Сдвиг тона 0,0°

Пипетки сдвига цветового тона:

Загрузить рецепт... Сохранить рецепт как... Donate Поддержать

Готово Отмена Справка (сайт)

X: 4801 Y: 2587 | R: 42 G: 63 B: 15 | L: 24.1 (4.1) C: 3.1 H: 125.2° S: 88.4% (0.75) sRGB

Суперпозиция базовых преобразований

Суперпозиция базовых преобразований

- Последовательность (суперпозиция) базовых преобразований может быть сведена к:

Суперпозиция базовых преобразований

- Последовательность (суперпозиция) базовых преобразований может быть сведена к:

  - Вычисление матрицы суммарного преобразования (перемножение матриц преобразований)

Суперпозиция базовых преобразований

- Последовательность (суперпозиция) базовых преобразований может быть сведена к:

  - Вычисление матрицы суммарного преобразования (перемножение матриц преобразований)

  - Умножение цветового вектора каждой точки (пикселя) на матрицу преобразования

Матрицы преобразований

- Матрица баланса белого (пространство *RGB*)

$$\begin{bmatrix} \frac{L_0}{R_0} & 0 & 0 & 0 \\ 0 & \frac{L_0}{G_0} & 0 & 0 \\ 0 & 0 & \frac{L_0}{B_0} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- Изменение насыщенности (пространство *Lpy*)

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & k_S & 0 & 0 \\ 0 & 0 & k_S & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Матрицы преобразований

- Удаление вуали (пространство *RGB*)

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -L_0 & -L_0 & -L_0 & 1 \end{bmatrix}$$

- Удаление вуали (пространство *Lpy*)

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -L_0 & 0 & 0 & 1 \end{bmatrix}$$

Матрицы преобразований

- Преобразование *RGB* → *Lpy*

$$\begin{bmatrix} 0.2126 & 0.7152 & 0.0722 & 0 \\ 0.6468 & -0.6468 & 0.0000 & 0 \\ 0.7325 & 0.2649 & -0.9974 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- …

Матрица произвольного поворота

(Википедия)

Выражение матрицы поворота через угол поворота θ и единичный вектор оси вращения $\hat{\mathbf{v}} = (x, y, z)$

В декартовых координатах матрица поворота имеет вид:

$$M(\hat{\mathbf{v}}, \theta) = \begin{pmatrix} \cos\theta + (1-\cos\theta)x^2 & (1-\cos\theta)xy - (\sin\theta)z & (1-\cos\theta)xz + (\sin\theta)y \\ (1-\cos\theta)yx + (\sin\theta)z & \cos\theta + (1-\cos\theta)y^2 & (1-\cos\theta)yz - (\sin\theta)x \\ (1-\cos\theta)zx - (\sin\theta)y & (1-\cos\theta)zy + (\sin\theta)x & \cos\theta + (1-\cos\theta)z^2 \end{pmatrix}$$

Матрица произвольного поворота

(Википедия)

Выражение матрицы поворота через угол поворота θ и единичный вектор оси вращения $\hat{\mathbf{v}} = (x, y, z)$

В **декартовых** координатах матрица поворота имеет вид:

$$M(\hat{\mathbf{v}}, \theta) = \begin{pmatrix} \cos\theta + (1-\cos\theta)x^2 & (1-\cos\theta)xy - (\sin\theta)z & (1-\cos\theta)xz + (\sin\theta)y \\ (1-\cos\theta)yx + (\sin\theta)z & \cos\theta + (1-\cos\theta)y^2 & (1-\cos\theta)yz - (\sin\theta)x \\ (1-\cos\theta)zx - (\sin\theta)y & (1-\cos\theta)zy + (\sin\theta)x & \cos\theta + (1-\cos\theta)z^2 \end{pmatrix}$$

Преобразование

$$\begin{bmatrix} R' \\ G' \\ B' \end{bmatrix} = M \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

Преобразование

```
struct Mat3D {
    float x2x, y2x, z2x;
    float x2y, y2y, z2y;
    float x2z, y2z, z2z;
    float dx, dy, dz;
};
…
inline void trans3D(float x, float y, float z, Mat3D m,
float &x2, float &y2, float &z2)
{
    x2 = m.x2x*x + m.y2x*y + m.z2x*z + m.dx;
    y2 = m.x2y*x + m.y2y*y + m.z2y*z + m.dy;
    z2 = m.x2z*x + m.y2z*y + m.z2z*z + m.dz;
}
```

Преобразование

```
struct Mat3D {
    float x2x, y2x, z2x;
    float x2y, y2y, z2y;
    float x2z, y2z, z2z;
    float dx, dy, dz;
};
```

9 умножений и 9 сложений на пиксель

```
inline void trans3D(float x, float y, float z, Mat3D m,
float &x2, float &y2, float &z2)
{
    x2 = m.x2x*x + m.y2x*y + m.z2x*z + m.dx;
    y2 = m.x2y*x + m.y2y*y + m.z2y*z + m.dy;
    z2 = m.x2z*x + m.y2z*y + m.z2z*z + m.dz;
}
```

Преобразование

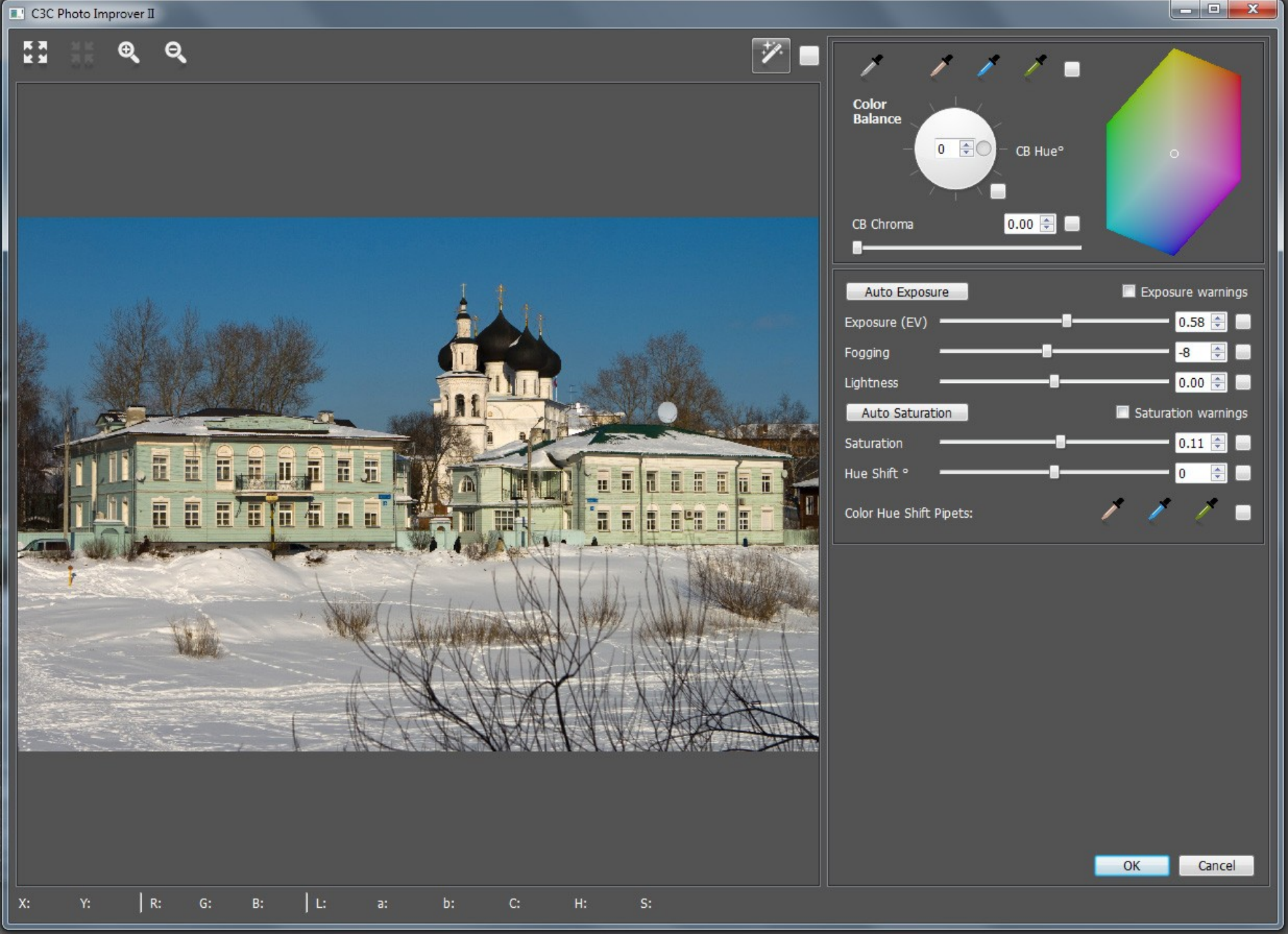
```
struct Mat3D {
    float x2x, y2x, z2x;
    float x2y, y2y, z2y;
    float x2z, y2z, z2z;
    float dx, dy, dz;
};
```

9 умножений и 9 сложений на пиксель

```
inline void trans3D(float x, float y, float z, Mat3D m,
float &x2, float &y2, float &z2)
{
```

+ преобразование в линейное RGB и обратно по таблице 3*(3 индексации, округление)

```
    z2 = m.x2z*x + m.y2z*y + m.z2z*z + m.dz;
}
```



Линейные преобразования

Было

Стало

Нелинейные контекстно-независимые преобразования

Нелинейные контекстно-независимые преобразования

- Контекстная независимость – цвет пикселя после преобразования зависит только от его цвета до обработки.

Нелинейные контекстно-независимые преобразования

- Контекстная независимость – цвет пикселя после преобразования зависит только от его цвета до обработки.

- В общем случае

$$R' = f_R(R, G, B)$$
$$G' = f_G(R, G, B)$$
$$B' = f_B(R, G, B)$$

Нелинейные контекстно-независимые преобразования

- Контекстная независимость – цвет пикселя после преобразования зависит только от его цвета до обработки.

- В общем случае

$$R' = f_R(R, G, B)$$
$$G' = f_G(R, G, B)$$
$$B' = f_B(R, G, B)$$

- Табулировать функцию преобразования
  - 256 уровней: 256^3 = 16 777 216 – возможно, но уже на грани разумного. 256 уровней – мало.
  - Если аргументов 1 или 2, можно табулировать

Нелинейные контекстно-независимые преобразования

- Контекстная независимость – цвет пикселя после преобразования зависит только от его цвета до обработки.

- В общем случае

$$R' = f_R(R, G, B)$$
$$G' = f_G(R, G, B)$$
$$B' = f_B(R, G, B)$$

- Табулировать функцию преобразования
  - 256 уровней: 256^3 = 16 777 216 – возможно, но уже на грани разумного. 256 уровней – мало.
  - Если аргументов 1 или 2, можно табулировать

- Табулировать с интерполяцией

Нелинейные контекстно-независимые преобразования

- Контекстная независимость – цвет пикселя после преобразования зависит только от его цвета до обработки.

- В общем случае

$$R' = f_R(R, G, B)$$
$$G' = f_G(R, G, B)$$
$$B' = f_B(R, G, B)$$

- Табулировать функцию преобразования
  - 256 уровней: 256^3 = 16 777 216 – возможно, но уже на грани разумного. 256 уровней – мало.
  - Если аргументов 1 или 2, можно табулировать

- Табулировать с интерполяцией

- Прямое вычисление

Нелинейные контекстно-независимые преобразования

Нелинейные контекстно-независимые преобразования

- Варианты

Нелинейные контекстно-независимые преобразования

- Варианты
  - Exposure = f(Exposure)
    - Осветление (gamma < 1)
    - Затемнение (gamma > 1)
    - Контрастность (S-образная кривая)
    - …

Нелинейные контекстно-независимые преобразования

- Варианты
  - Exposure = f(Exposure)
    - Осветление (gamma < 1)
    - Затемнение (gamma > 1)
    - Контрастность (S-образная кривая)
    - …
  - Lightess = f(Lightess)
    - Осветление (gamma < 1)
    - Затемнение (gamma > 1)
    - Контрастность (S-образная кривая)
    - …

Нелинейные контекстно-независимые преобразования

- Варианты
  - Exposure = f(Exposure)
    - Осветление (gamma < 1)
    - Затемнение (gamma > 1)
    - Контрастность (S-образная кривая)
    - …
  - Lightess = f(Lightess)
    - Осветление (gamma < 1)
    - Затемнение (gamma > 1)
    - Контрастность (S-образная кривая)
    - …
  - Chroma = f(Chroma)
    - Vibrance

Нелинейные контекстно-независимые преобразования

- Варианты
  - Exposure = f(Exposure)
    - Осветление (gamma < 1)
    - Затемнение (gamma > 1)
    - Контрастность (S-образная кривая)
    - …
  - Lightess = f(Lightess)
    - Осветление (gamma < 1)
    - Затемнение (gamma > 1)
    - Контрастность (S-образная кривая)
    - …
  - Chroma = f(Chroma)
    - Vibrance
  - Chroma = f(Chroma, Hue)
    - Эквалайзер Тон – Насыщенность

Нелинейные контекстно-независимые преобразования

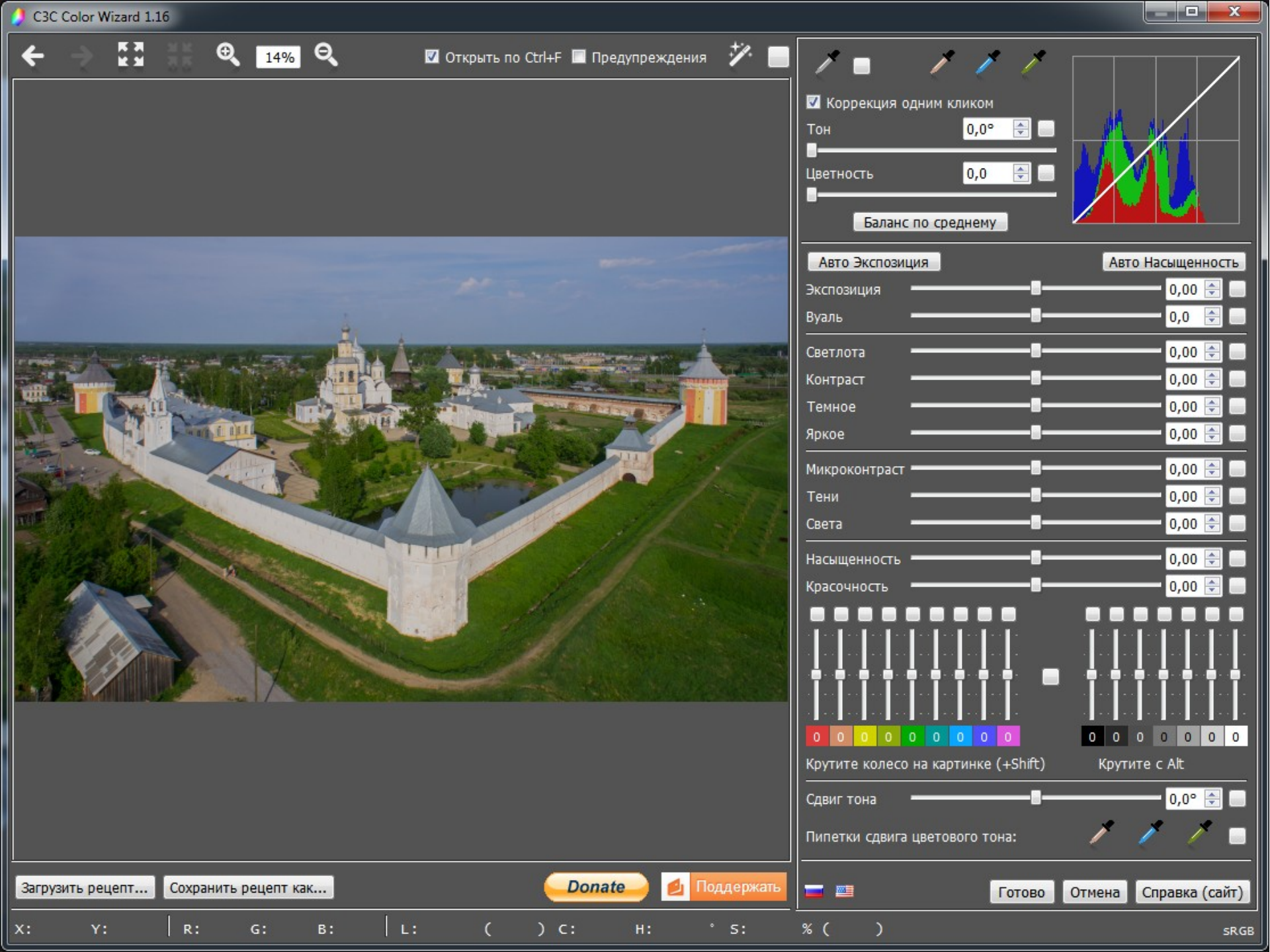
- Варианты
  - Exposure = f(Exposure)
    - Осветление (gamma < 1)
    - Затемнение (gamma > 1)
    - Контрастность (S-образная кривая)
    - …
  - Lightess = f(Lightess)
    - Осветление (gamma < 1)
    - Затемнение (gamma > 1)
    - Контрастность (S-образная кривая)
    - …
  - Chroma = f(Chroma)
    - Vibrance
  - Chroma = f(Chroma, Hue), Chroma = Chroma*f(Hue)
    - Эквалайзер Тон – Насыщенность

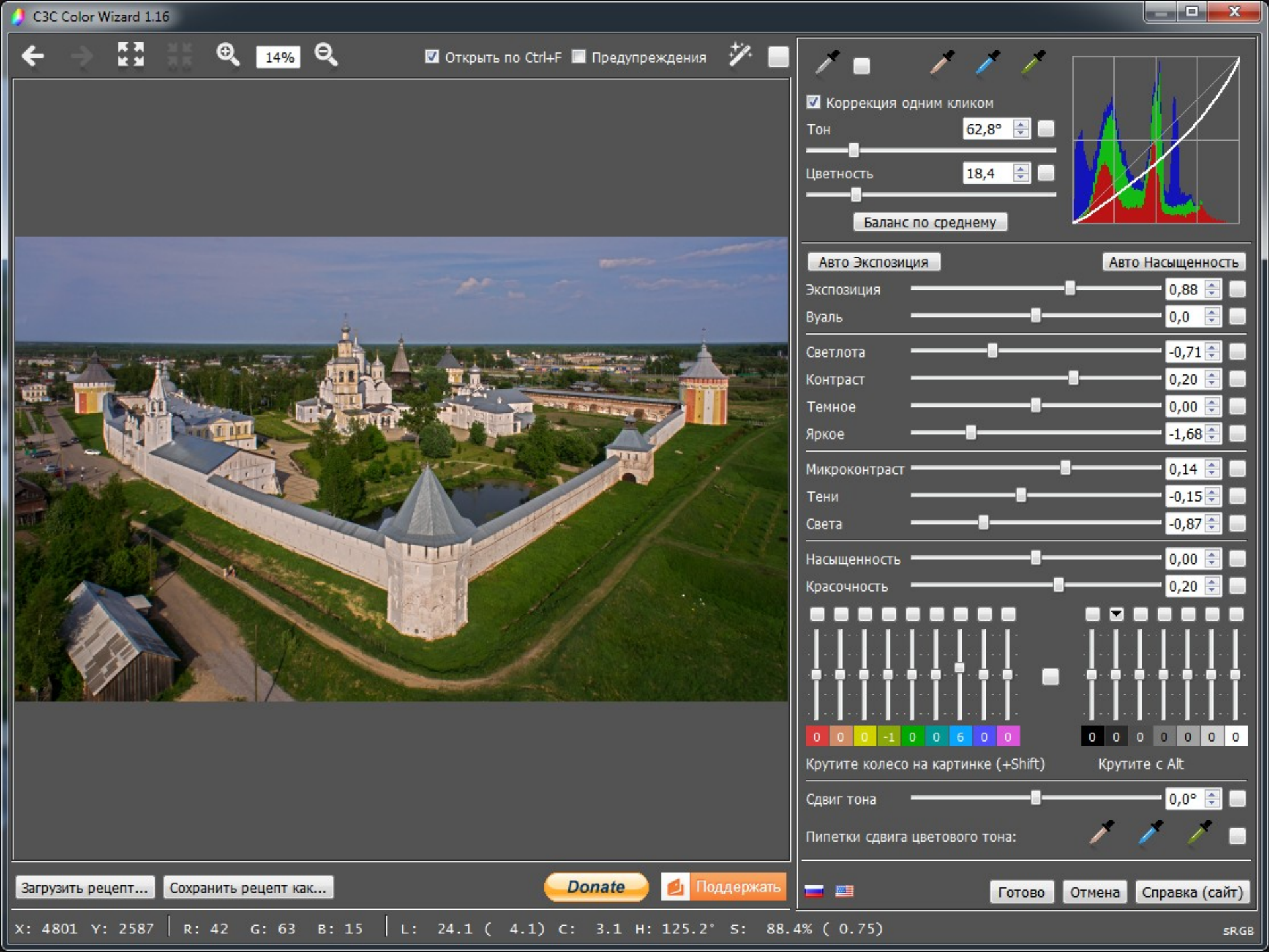
Нелинейные контекстно-независимые преобразования

- Варианты
  - Exposure = f(Exposure)
    - Осветление (gamma < 1)
    - Затемнение (gamma > 1)
    - Контрастность (S-образная кривая)
    - …
  - Lightess = f(Lightess)
    - Осветление (gamma < 1)
    - Затемнение (gamma > 1)
    - Контрастность (S-образная кривая)
    - …
  - Chroma = f(Chroma)
    - Vibrance
  - Chroma = f(Chroma, Hue), Chroma = Chroma*f(Hue)
    - Эквалайзер Тон – Насыщенность
  - Chroma = f(Chroma, Lightness), Chroma = Chroma*f(Lightness)
    - Эквалайзер Светлота – Насыщенность

Нелинейные контекстно-независимые преобразования

- Варианты
  - Exposure = f(Exposure)
    - Осветление (gamma < 1)
    - Затемнение (gamma > 1)
    - Контрастность (S-образная кривая)
    - …
  - Lightess = f(Lightess)
    - Осветление (gamma < 1)
    - Затемнение (gamma > 1)
    - Контрастность (S-образная кривая)
    - …
  - Chroma = f(Chroma)
    - Vibrance
  - Chroma = f(Chroma, Hue), Chroma = Chroma*f(Hue)
    - Эквалайзер Тон – Насыщенность
  - Chroma = f(Chroma, Lightness), Chroma = Chroma*f(Lightness)
    - Эквалайзер Светлота – Насыщенность
  - …


C3C Color Wizard 1.16
14%
Открыть по Ctrl+F
Предупреждения
Коррекция одним кликом
Тон
0,0°
Цветность
0,0
Баланс по среднему
Авто Экспозиция
Авто Насыщенность
Экспозиция
0,00
Вуаль
0,0
Светлота
Контраст
Темное
Яркое
Микроконтраст
Тени
Света
Насыщенность
Красочность
Крутите колесо на картинке (+Shift)
Крутите с Alt
Сдвиг тона
0,0°
Пипетки сдвига цветового тона:
Загрузить рецепт...
Сохранить рецепт как...
Donate
Поддержать
Готово
Отмена
Справка (сайт)
X:
Y:
R:
G:
B:
L:
C:
H:
S:
sRGB


C3C Color Wizard 1.16
14%
Открыть по Ctrl+F
Предупреждения
Коррекция одним кликом
Тон
62,8°
Цветность
18,4
Баланс по среднему
Авто Экспозиция
Авто Насыщенность
Экспозиция
0,88
Вуаль
0,0
Светлота
-0,71
Контраст
0,20
Темное
0,00
Яркое
-1,68
Микроконтраст
0,14
Тени
-0,15
Света
-0,87
Насыщенность
0,00
Красочность
0,20
0 0 0 -1 0 0 6 0 0
0 0 0 0 0 0 0
Крутите колесо на картинке (+Shift)
Крутите с Alt
Сдвиг тона
0,0°
Пипетки сдвига цветового тона:
Загрузить рецепт...
Сохранить рецепт как...
Donate
Поддержать
Готово
Отмена
Справка (сайт)
X: 4801 Y: 2587
R: 42 G: 63 B: 15
L: 24.1 (4.1) C: 3.1 H: 125.2° S: 88.4% (0.75)
sRGB

C3C Color Wizard 1.16

14% Открыть по Ctrl+F Предупреждения

Линейные преобразования

Тон 62,8°

Цветность 18,4

Баланс по среднему

Авто Экспозиция Авто Насыщенность

Экспозиция 0,88

Вуаль 0,0

Светлота -0,71

Контраст 0,20

Темное 0,00

Яркое -1,68

Микроконтраст 0,14

Тени -0,15

Света -0,87

Насыщенность 0,00

Красочность 0,20

0 0 0 -1 0 0 6 0 0

0 0 0 0 0 0 0

Крутите колесо на картинке (+Shift) Крутите с Alt

Сдвиг тона 0,0°

Пипетки сдвига цветового тона:

Загрузить рецепт... Сохранить рецепт как... Donate Поддержать

Готово Отмена Справка (сайт)

X: 4801 Y: 2587 | R: 42 G: 63 B: 15 | L: 24.1 (4.1) C: 3.1 H: 125.2° S: 88.4% (0.75) sRGB

C3C Color Wizard 1.16

14% Открыть по Ctrl+F Предупреждения

Нелинейные КС-преобразования

Коррекция одним кликом

Тон 62,8°

Цветность 18,4

Баланс по среднему

Авто Экспозиция Авто Насыщенность

Экспозиция 0,88

Вуаль 0,0

Светлота -0,71

Контраст 0,20

Темное 0,00

Яркое -1,68

Микроконтраст 0,14

Тени -0,15

Света -0,87

Насыщенность 0,00

Красочность 0,20

0 0 0 -1 0 0 6 0 0

0 0 0 0 0 0 0

Крутите колесо на картинке (+Shift) Крутите с Alt

Сдвиг тона 0,0°

Пипетки сдвига цветового тона:

Загрузить рецепт... Сохранить рецепт как... Donate Поддержать

Готово Отмена Справка (сайт)

X: 4801 Y: 2587 | R: 42 G: 63 B: 15 | L: 24.1 (4.1) C: 3.1 H: 125.2° S: 88.4% (0.75) sRGB

Дополнительные главы компьютерной графики

Лекция 8

Copyright © С. Свердлов, 2009-2017
http://www.uni-vologda.ac.ru/~c3c

Контекстно-зависимые (пространственно-зависимые) преобразования

Контекстно-зависимые (пространственно-зависимые) преобразования

- Цвет пикселя после преобразования зависит от его цвета до обработки и местоположения пикселя.

$$R' = f_R(R, G, B, x, y)$$
$$G' = f_G(R, G, B, x, y)$$
$$B' = f_B(R, G, B, x, y)$$

Контекстно-зависимые (пространственно-зависимые) преобразования

- Цвет пикселя после преобразования зависит от его цвета до обработки и местоположения пикселя.

$$R' = f_R(R, G, B, x, y)$$
$$G' = f_G(R, G, B, x, y)$$
$$B' = f_B(R, G, B, x, y)$$

  - Во многих случаях:

$$R' = f_R(R, G, B, C_R(x, y, \ldots))$$
$$G' = f_G(R, G, B, C_G(x, y, \ldots))$$
$$B' = f_B(R, G, B, C_B(x, y, \ldots))$$

Контекстно-зависимые (пространственно-зависимые) преобразования

- Цвет пикселя после преобразования зависит от его цвета до обработки и местоположения пикселя.

$$R' = f_R(R, G, B, x, y)$$
$$G' = f_G(R, G, B, x, y)$$
$$B' = f_B(R, G, B, x, y)$$

  - Во многих случаях:

$$R' = f_R(R, G, B, C_R(x, y, \ldots))$$
$$G' = f_G(R, G, B, C_G(x, y, \ldots))$$
$$B' = f_B(R, G, B, C_B(x, y, \ldots))$$

где $C_{R,\,G,\,B}(x, y, \ldots)$ – некоторая характеристика изображения в точке (x, y)

Контекстно-зависимые (пространственно-зависимые) преобразования

- Например

Контекстно-зависимые (пространственно-зависимые) преобразования

- Например
  - $C(x, y, r)$ – средняя по окрестности размера r светлота изображения в точке (x, y)

Контекстно-зависимые (пространственно-зависимые) преобразования

- Например
  - $C(x, y, r)$ – средняя по окрестности размера r светлота изображения в точке (x, y)
    - **Резкость** (USM – нерезкая маска в «канале» L). Малый размер окрестности

Контекстно-зависимые (пространственно-зависимые) преобразования

- Например
  - $C(x, y, r)$ – средняя по окрестности размера r светлота изображения в точке (x, y)
    - **Резкость** (USM – нерезкая маска в «канале» L). Малый размер окрестности
    - **Локальный контраст** (Clarity в «канале» L). Большой размер окрестности

Контекстно-зависимые (пространственно-зависимые) преобразования

- Например
  - $C(x, y, r)$ – средняя по окрестности размера r светлота изображения в точке (x, y)
    - **Резкость** (USM – нерезкая маска в «канале» L). Малый размер окрестности
    - **Локальный контраст** (Clarity в «канале» L). Большой размер окрестности
    - **Тени/Света**. Большой размер окрестности

Контекстно-зависимые (пространственно-зависимые) преобразования

- Например
  - $C(x, y, r)$ – средняя по окрестности размера r светлота изображения в точке (x, y)
    - **Резкость** (USM – нерезкая маска в «канале» L). Малый размер окрестности
    - **Локальный контраст** (Clarity в «канале» L). Большой размер окрестности
    - **Тени/Света**. Большой размер окрестности
  - $C_{R, G, B}(x, y, r)$ – средние по окрестности размера r значения R, G, B в точке (x, y)
    - **Резкость** (USM – нерезкая маска). Малый размер окрестности
    - **Локальный контраст** (Clarity). Большой размер окрестности

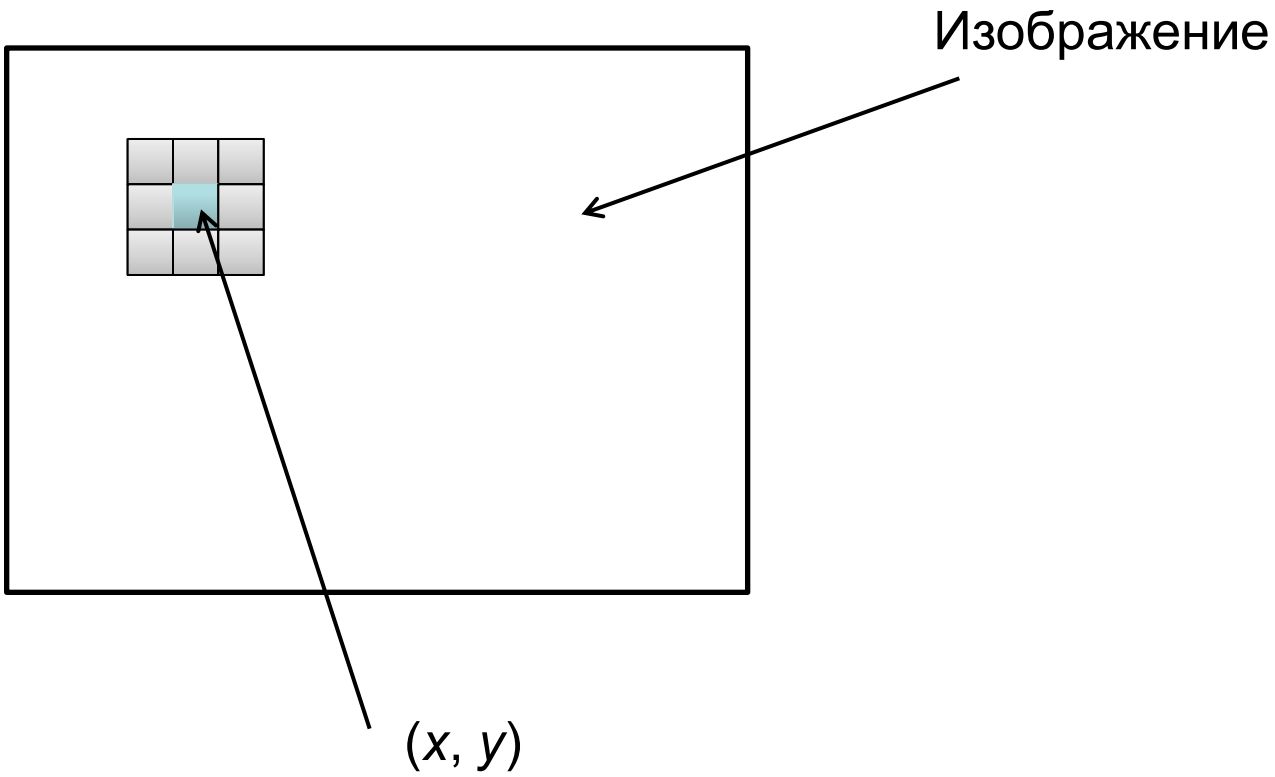
Вычисление средних

Вычисление средних

- Найдем среднее значение (например, светлоты) в точке x, y.
  - Усреднение по 9 соседним точкам.

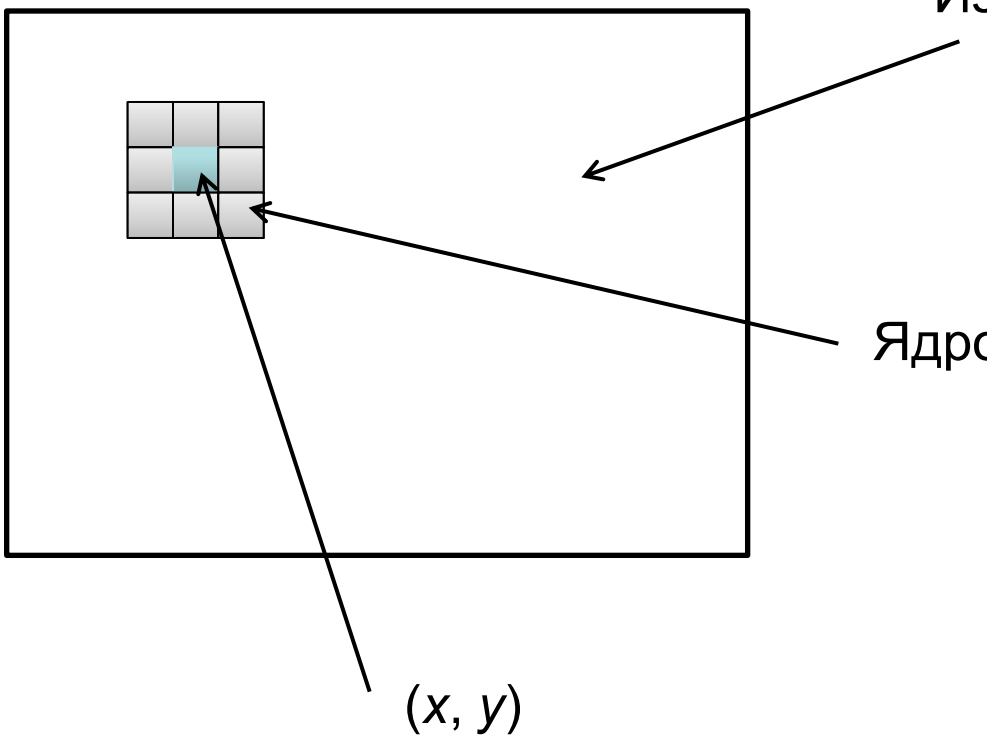
Вычисление средних

- Найдем среднее значение (например, светлоты) в точке x, y.
  - Усреднение по 9 соседним точкам.



Вычисление средних

- Найдем среднее значение (например, светлоты) в точке x, y.
  - Усреднение по 9 соседним точкам.

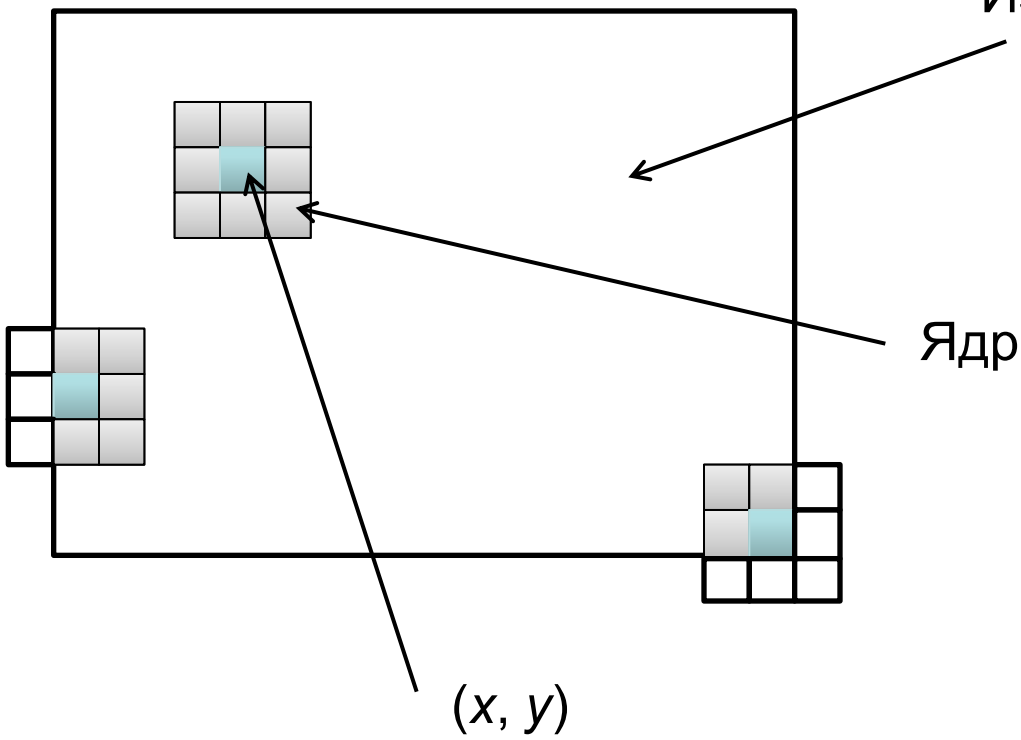


Ядро
(весовые коэффициенты)

$$\begin{bmatrix} \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \mathbf{\frac{1}{9}} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \end{bmatrix}$$

Вычисление средних

- Найдем среднее значение (например, светлоты) в точке x, y.
  - Усреднение по 9 соседним точкам.



Ядро
(весовые коэффициенты)

$$\begin{bmatrix} \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \mathbf{\frac{1}{9}} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \end{bmatrix}$$

  - На краях участвует меньше точек

Двумерная свёртка

$$P'[i,j] = \sum_{k=1}^{k=n} \sum_{l=1}^{l=n} P[i+k-\frac{n+1}{2}, j+l-\frac{n+1}{2})] \cdot K[k,l]$$

$$K[n \times n] - \textit{ядро свертки};$$

$$n - \textit{нечётное}$$

$$K[3 \times 3] = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & \mathbf{1} & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Фильтры

- Прямоугольный усредняющий (сглаживающий) фильтр

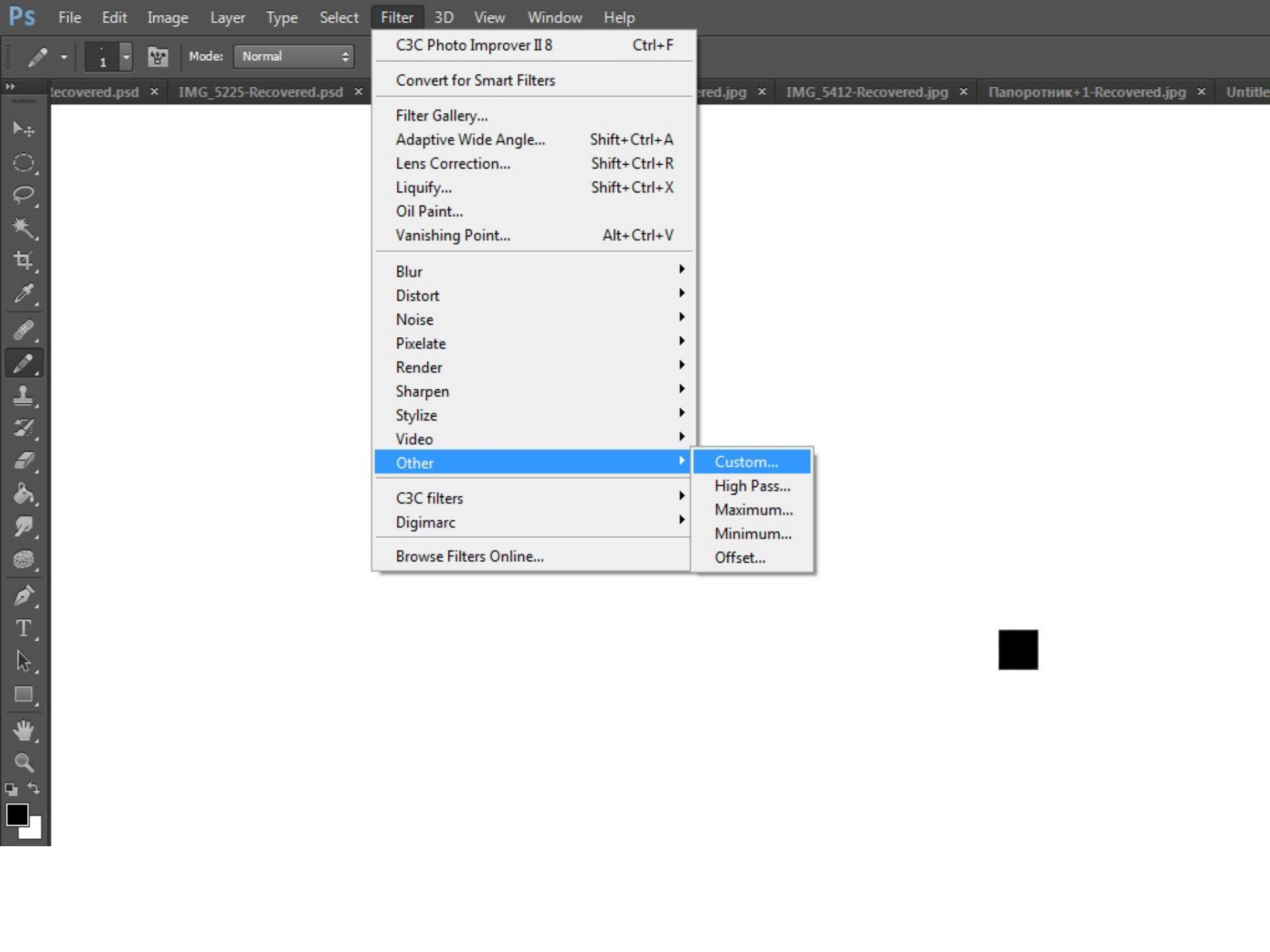
$$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & \mathbf{1} & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

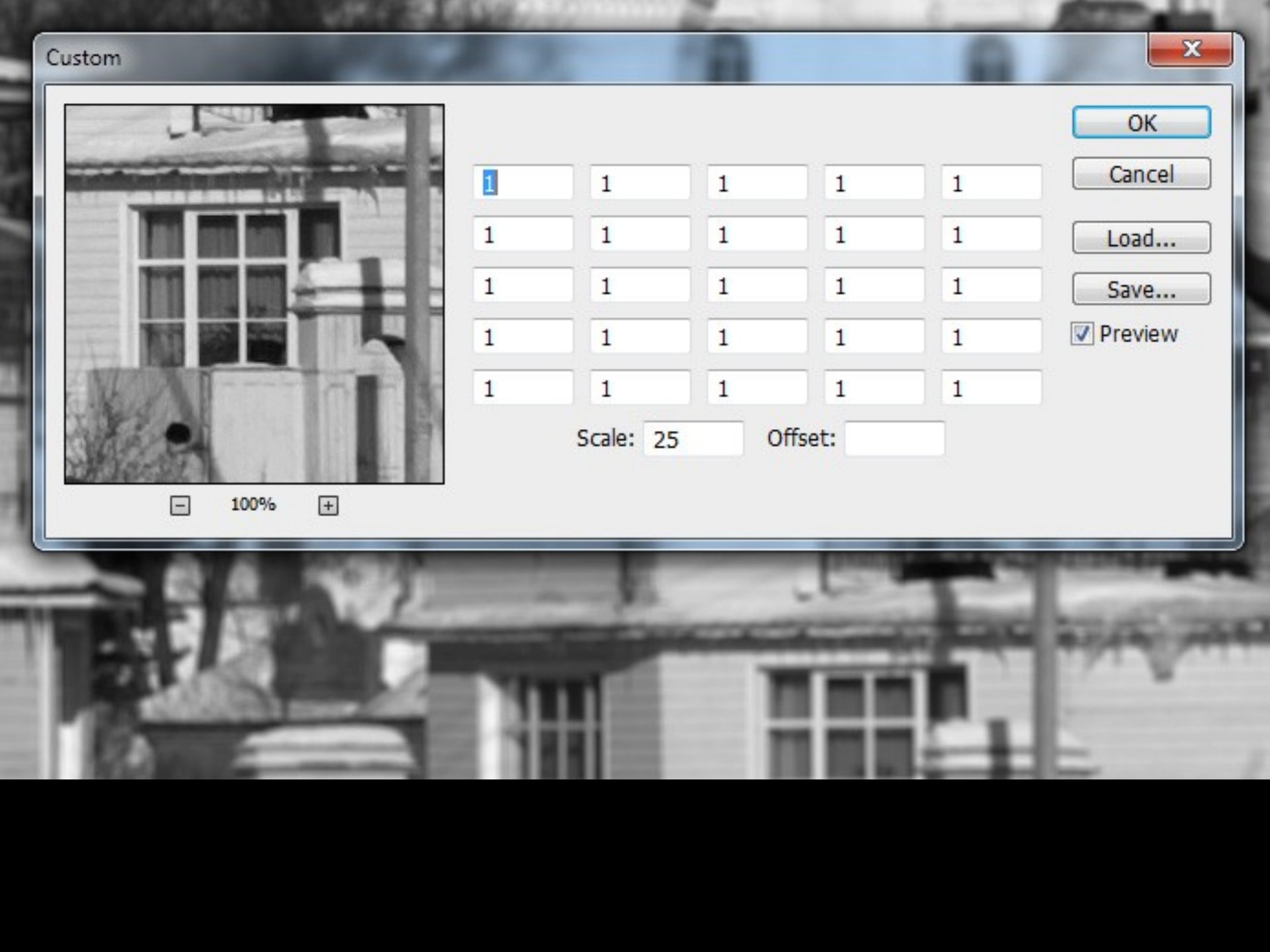
Фильтры

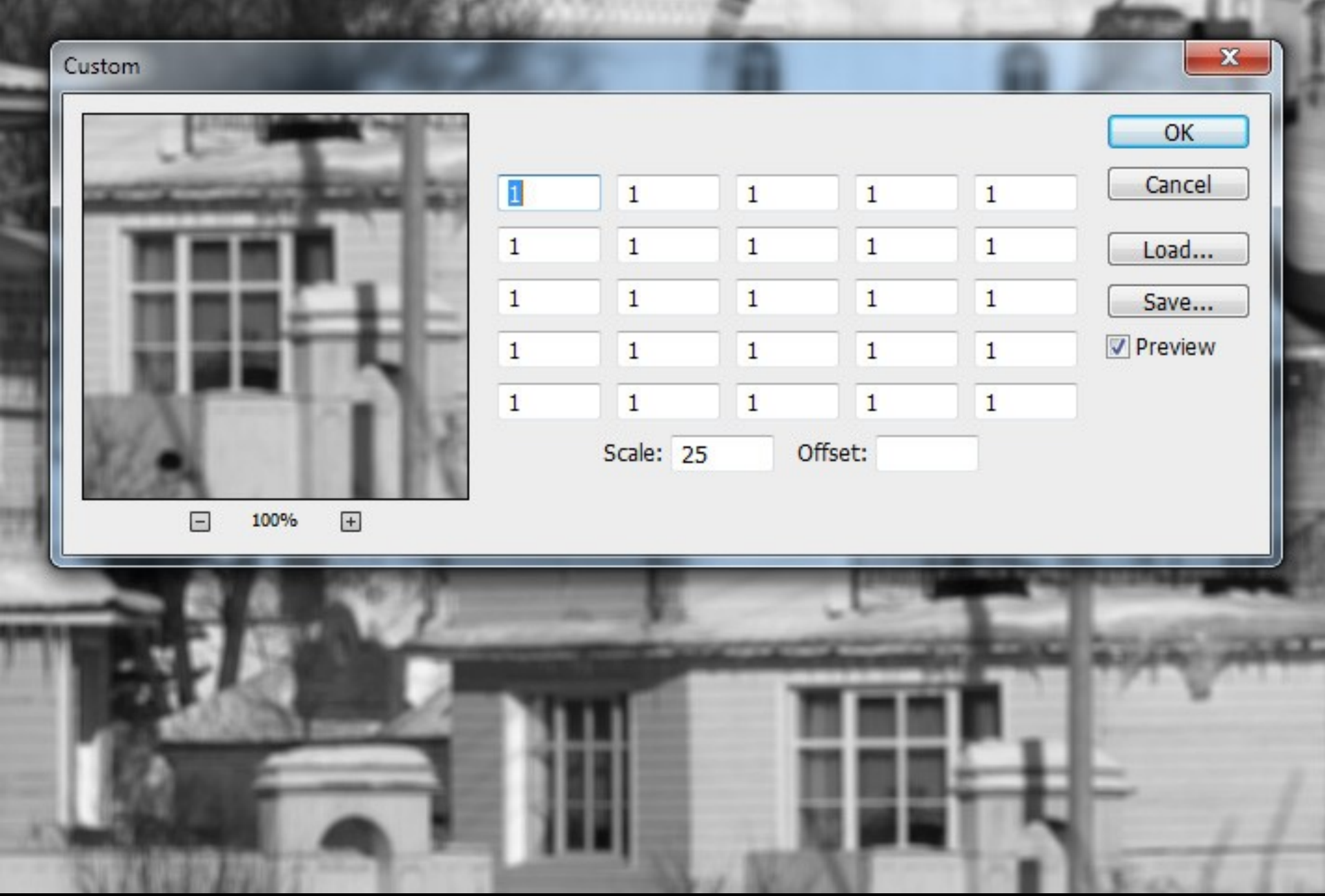
- Прямоугольный усредняющий (сглаживающий) фильтр

$$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & \mathbf{1} & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

- Усреднение = сглаживание = размытие


Ps
File
Edit
Image
Layer
Type
Select
Filter
3D
View
Window
Help
Mode:
Normal
ecovered.psd
IMG_5225-Recovered.psd
red.jpg
IMG_5412-Recovered.jpg
Папоротник+1-Recovered.jpg
Untitle
C3C Photo Improver II 8 Ctrl+F
Convert for Smart Filters
Filter Gallery...
Adaptive Wide Angle... Shift+Ctrl+A
Lens Correction... Shift+Ctrl+R
Liquify... Shift+Ctrl+X
Oil Paint...
Vanishing Point... Alt+Ctrl+V
Blur
Distort
Noise
Pixelate
Render
Sharpen
Stylize
Video
Other
C3C filters
Digimarc
Browse Filters Online...
Custom...
High Pass...
Maximum...
Minimum...
Offset...


Custom
OK
Cancel
Load...
Save...
Preview
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
Scale: 25
Offset:
100%


Custom
OK
Cancel
Load...
Save...
Preview
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
Scale: 25
Offset:
100%

Размытие одиночного пикселя (импульсная характеристика)

Увеличение 3200%. Исходный

Размытие одиночного пикселя (импульсная характеристика)

Увеличение 3200%. Прямоугольный сглаживающий фильтр

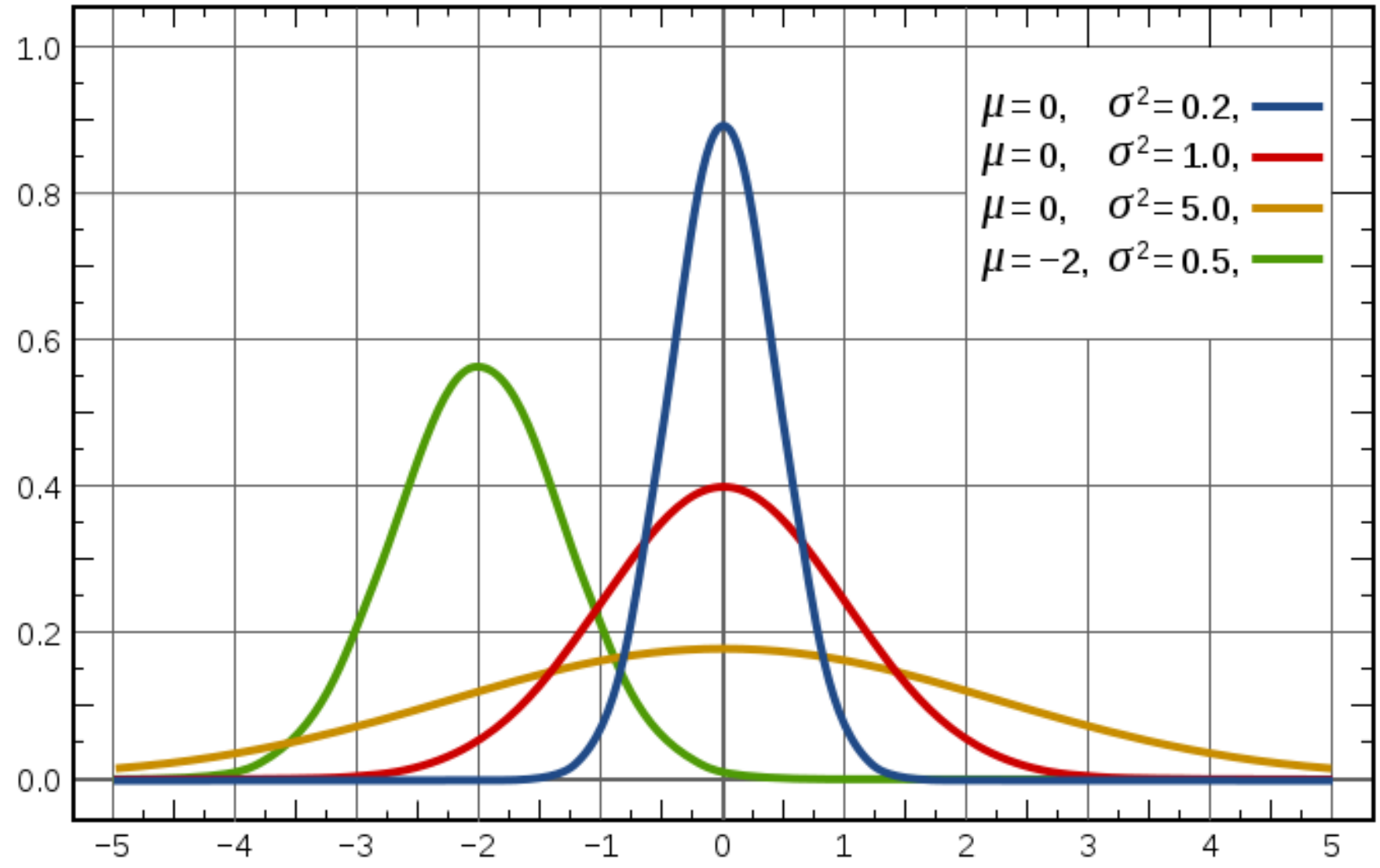
Размытие одиночного пикселя (импульсная характеристика)

Увеличение 3200%. Исходный

Размытие одиночного пикселя (импульсная характеристика)

Увеличение 3200%. Фильтр Гаусса σ = 0.5

Функция Гаусса



Ядро фильтра Гаусса

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

$$x, y \in [-\infty, \infty]$$

Двумерный

Ядро фильтра Гаусса

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

$$x, y \in [-\infty, \infty]$$

Двумерный

Одномерный

$$G(r) = \frac{1}{2\pi\sigma^2} e^{-\frac{r^2}{2\sigma^2}}$$

$$r \in [-3\sigma, 3\sigma]$$

Ядро фильтра Гаусса

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

$$x, y \in [-\infty, \infty]$$

Двумерный

Одномерный

$$G(r) = \frac{1}{2\pi\sigma^2} e^{-\frac{r^2}{2\sigma^2}}$$

$$r \in [-3\sigma, 3\sigma]$$

Исходное изображение

Размытие по Гауссу, σ=10 (Adobe Photoshop)

Box Blur, радиус 15 (Adobe Photoshop)

Реализация усредняющих (сглаживающих) фильтров

Реализация усредняющих (сглаживающих) фильтров

- Сепарабельность

Реализация усредняющих (сглаживающих) фильтров

- Сепарабельность
  - Применение двумерного (прямоугольного и гауссова) фильтра к изображению эквивалентно последовательному применению одномерного фильтра к строкам, а затем к столбцам.

Реализация усредняющих (сглаживающих) фильтров

- Сепарабельность
  - Применение двумерного (прямоугольного и гауссова) фильтра к изображению эквивалентно последовательному применению одномерного фильтра к строкам, а затем к столбцам.

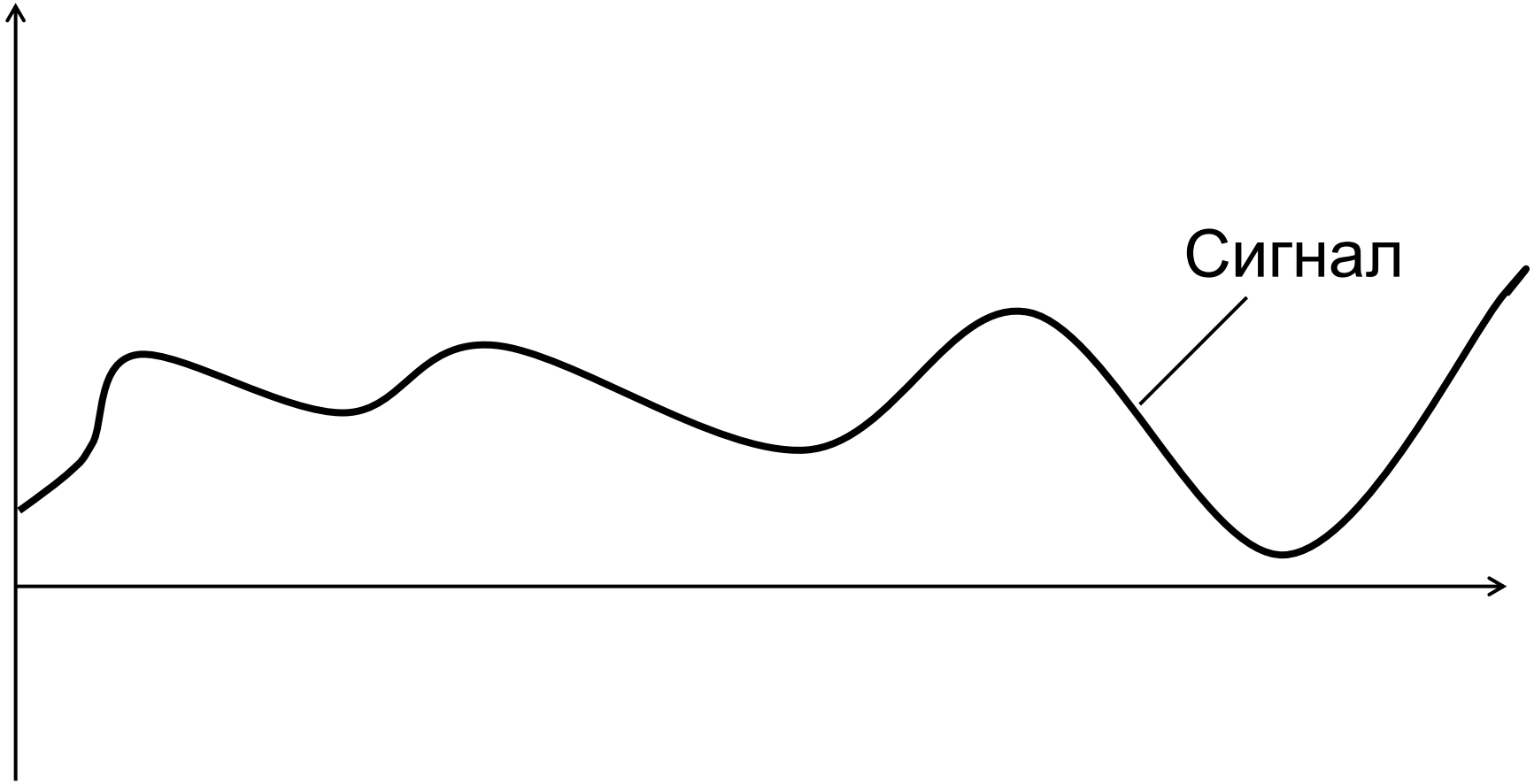
    Nx**N** – размер изображения;

    nx**n** – размер ядра.

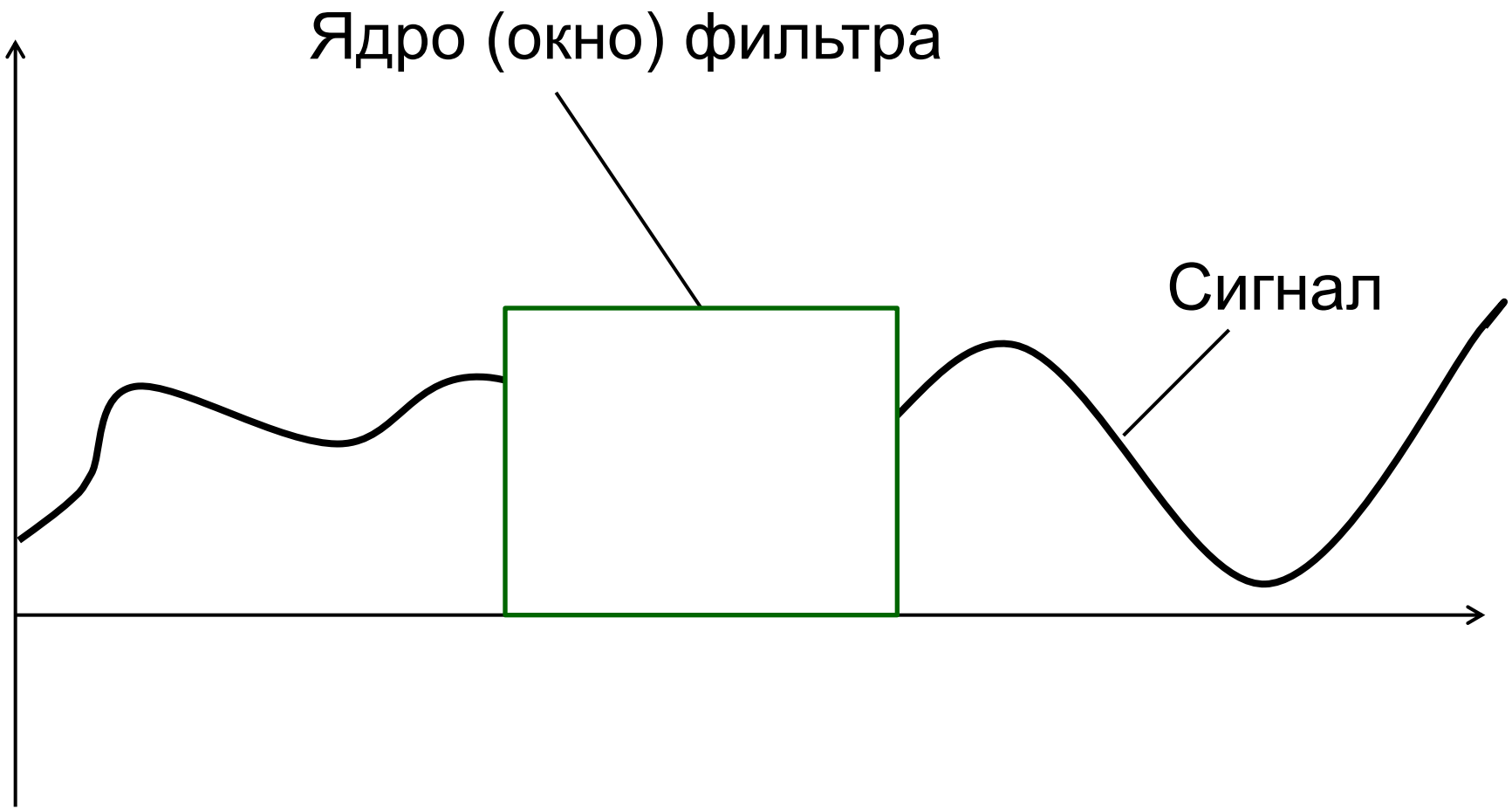
    - Несепарабельный фильтр (прямое вычисление): **$O(N^2*n^2)$**
    - Сепарабельный фильтр: **$O(2N^2*n)$**

Прямоугольный фильтр

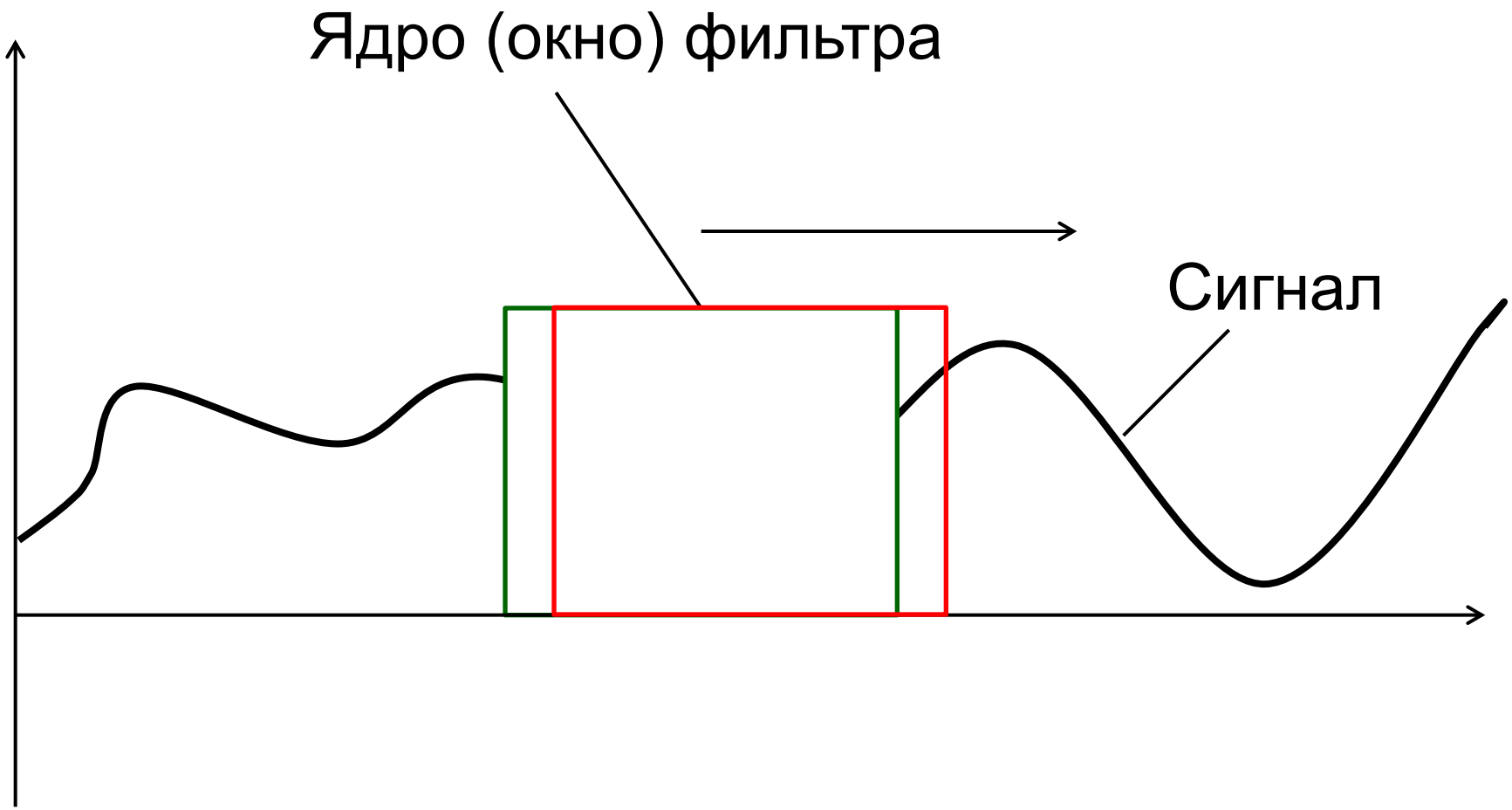
Прямоугольный фильтр



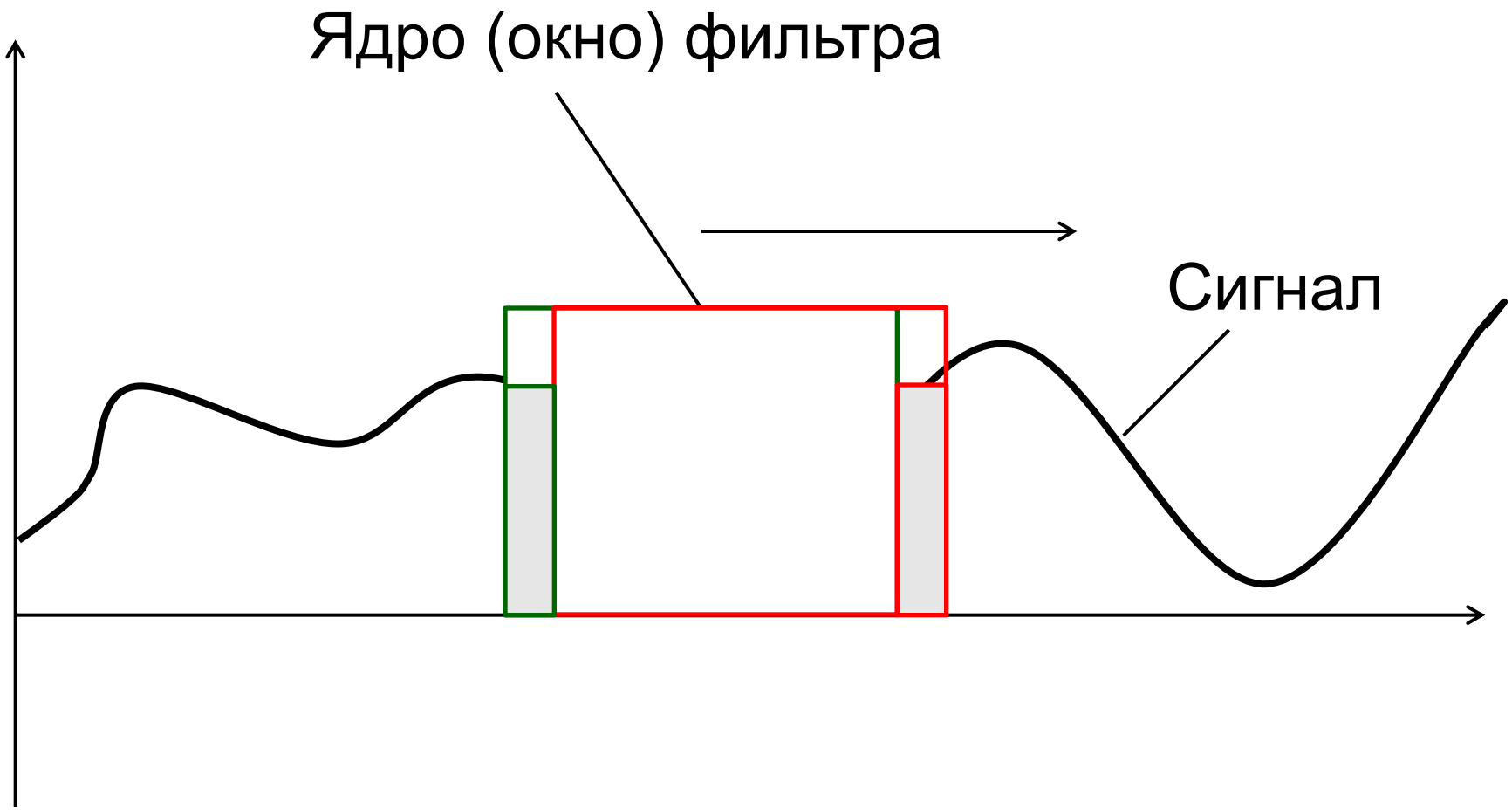
Прямоугольный фильтр



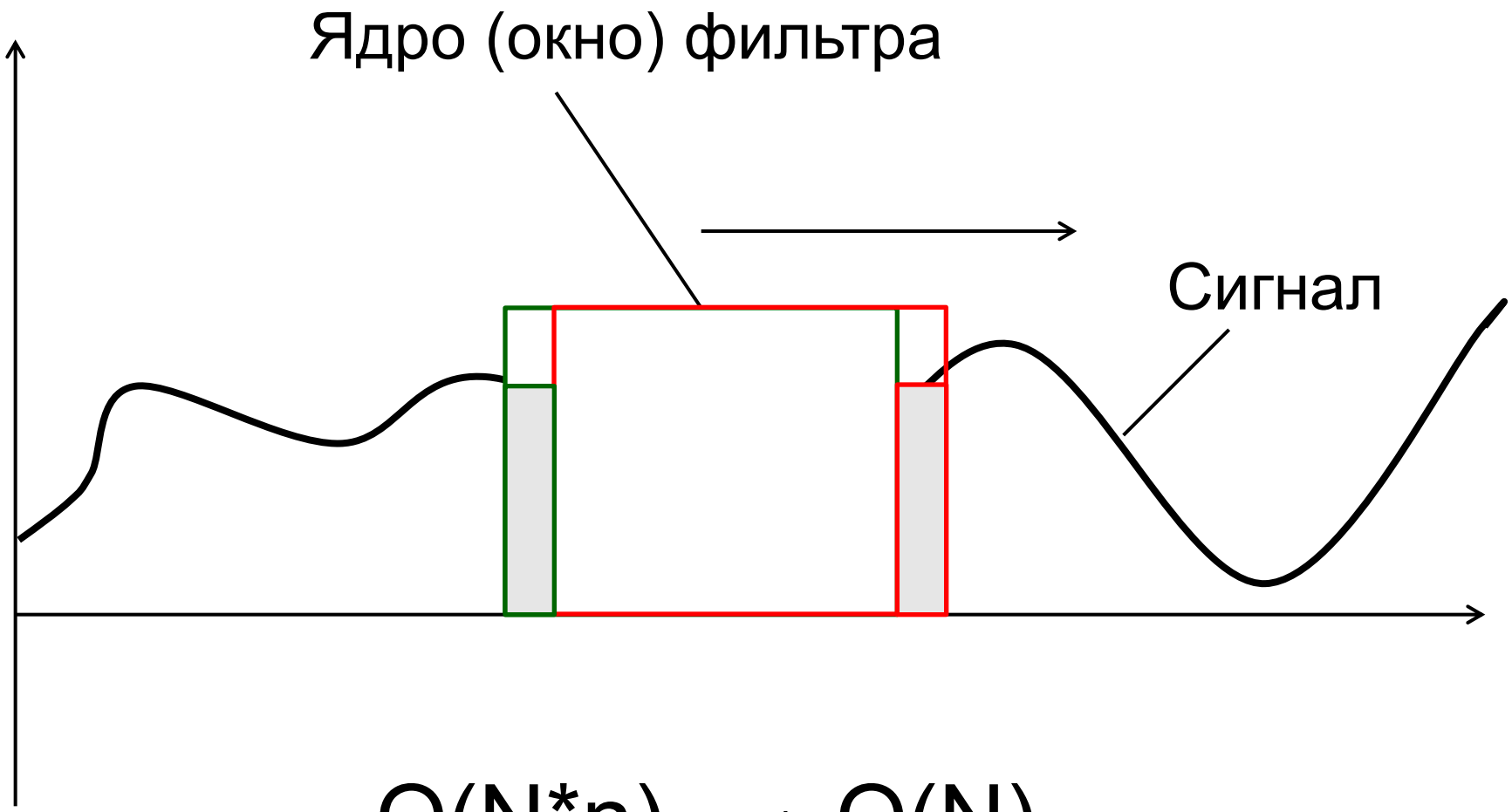
Прямоугольный фильтр



Прямоугольный фильтр



Прямоугольный фильтр



$$O(N*n) \rightarrow O(N)$$

Фильтр Гаусса

Фильтр Гаусса

- Прямая реализация (преобразование в пространственной области)
  - $[-3\sigma, +3\sigma]$, $n = 6\sigma + 1$
  - Затраты пропорциональны σ. При больших σ неприемлемо. $O(N*n)$ (одномерный фильтр)

Фильтр Гаусса

- Прямая реализация (преобразование в пространственной области)
  - $[-3\sigma, +3\sigma]$, $n = 6\sigma + 1$
  - Затраты пропорциональны σ. При больших σ неприемлемо. $O(N*n)$ (одномерный фильтр)
- Использование БПФ

Фильтр Гаусса

- Прямая реализация (преобразование в пространственной области)
  - $[-3\sigma, +3\sigma]$, $n = 6\sigma + 1$
  - Затраты пропорциональны σ. При больших σ неприемлемо. $O(N*n)$ (одномерный фильтр)
- Использование БПФ
  - БПФ (преобразование из пространственной в частотную область) $O(N \log N)$

Фильтр Гаусса

- Прямая реализация (преобразование в пространственной области)
  - $[-3\sigma, +3\sigma]$, $n = 6\sigma + 1$
  - Затраты пропорциональны σ. При больших σ неприемлемо.
    $O(N*n)$ (одномерный фильтр)
- Использование БПФ
  - БПФ (преобразование из пространственной в частотную область)
    $O(N \log N)$
  - Применение фильтра (свертка в пространственной области ~ умножение в частотной)
    $O(N)$

Фильтр Гаусса

- Прямая реализация (преобразование в пространственной области)
  - $[-3\sigma, +3\sigma]$, $n = 6\sigma + 1$
  - Затраты пропорциональны σ. При больших σ неприемлемо.
    $O(N*n)$ (одномерный фильтр)
- Использование БПФ
  - БПФ (преобразование из пространственной в частотную область)
    $O(N \log N)$
  - Применение фильтра (свертка в пространственной области ~ умножение в частотной)
    $O(N)$
  - Обратное БПФ (преобразование из частотной в пространственную область)
    $O(N \log N)$

Фильтр Гаусса

- Прямая реализация (преобразование в пространственной области)
  - $[-3\sigma, +3\sigma]$, $n = 6\sigma + 1$
  - Затраты пропорциональны σ. При больших σ неприемлемо.
    $O(N*n)$ (одномерный фильтр)
- Использование БПФ
  - БПФ (преобразование из пространственной в частотную область)
    $O(N \log N)$
  - Применение фильтра (свертка в пространственной области ~ умножение в частотной)
    $O(N)$
  - Обратное БПФ (преобразование из частотной в пространственную область)
    $O(N \log N)$
  - Трудоемкость $O(N \log N)$, не зависит от σ

Фильтр Гаусса

- Прямая реализация (преобразование в пространственной области)
  - $[-3\sigma, +3\sigma]$, $n = 6\sigma + 1$
  - Затраты пропорциональны σ. При больших σ неприемлемо.
    $O(N*n)$ (одномерный фильтр)
- Использование БПФ
  - БПФ (преобразование из пространственной в частотную область)
    $O(N \log N)$
  - Применение фильтра (свертка в пространственной области ~ умножение в частотной)
    $O(N)$
  - Обратное БПФ (преобразование из частотной в пространственную область)
    $O(N \log N)$
  - Трудоемкость $O(N \log N)$, не зависит от σ
- Рекурсивный фильтр (приближенное решение)
  - Трудоемкость $O(N)$, не зависит от σ

Преобразование в пространственной области (Java, фрагмент курсовой работы)

```
//по строкам
        for (int i = 0; i < img.height; i++) {
            int addr = i * img.width;
            for (int j = 0; j < img.width; j++) {
                sum = 0;
                newR = 0;
                newG = 0;
                newB = 0;
                for (int k = 0; k < 2 * n + 1; k++) {
                    l = j + k - n;
                    if (l >= 0 && l < img.width) {
                        color = img.array[addr + l];
                        newR += ((color & 0x00ff0000) >>> 16) * weight[k];
                        newG += ((color & 0x0000ff00) >>> 8) * weight[k];
                        newB += (color & 0x000000ff) * weight[k];
                        sum += weight[k];
                    }
                }
                newR = (int) (newR / sum);
                newG = (int) (newG / sum);
                newB = (int) (newB / sum);
                color = 0;
                color = (newR << 16) | (newG << 8) | (newB);
                tmp[j] = color;
            }
            for (int j = 0; j < img.width; j++) {
                img.array[addr + j] = tmp[j];
            }
        }
```

Рекурсивный фильтр

Рекурсивный фильтр

ELSEVIER

Signal Processing 44 (1995) 139–151

SIGNAL PROCESSING

Recursive implementation of the Gaussian filter

Ian T. Young*, Lucas J. van Vliet

Pattern Recognition Group, Faculty of Applied Physics, Lorentzweg 1, Delft University of Technology, NL-2628 CJ Delft, Netherlands

Received 29 September 1994; revised 25 January 1995

Abstract

In this paper we propose a recursive implementation of the Gaussian filter. This implementation yields an infinite impulse response filter that has six MADDs per dimension independent of the value of σ in the Gaussian kernel. In contrast to the Deriche implementation (1987), the coefficients of our recursive filter have a simple, closed-form solution for a desired value of the Gaussian σ. Our implementation is, in general, faster than (1) an implementation based upon direct convolution with samples of a Gaussian, (2) repeated convolutions with a kernel such as the uniform filter, and (3) an FFT implementation of a Gaussian filter.

Рекурсивный фильтр

ELSEVIER

Signal Processing 44 (1995) 139–151

SIGNAL PROCESSING

Recursive implementation of the Gaussian filter

Ian T. Young*, Lucas J. van Vliet

Pattern Recognition Group, Faculty of Applied Physics, Lorentzweg 1, Delft University of Technology, NL-2628 CJ Delft, Netherlands

Received 29 September 1994; revised 25 January 1995

Abstract

In this paper we propose a recursive implementation of the Gaussian filter. This implementation yields an infinite impulse response filter that has six MADDs per dimension independent of the value of σ in the Gaussian kernel. In contrast to the Deriche implementation (1987), the coefficients of our recursive filter have a simple, closed-form solution for a desired value of the Gaussian σ. Our implementation is, in general, faster than (1) an implementation based upon direct convolution with samples of a Gaussian, (2) repeated convolutions with a kernel such as the uniform filter, and (3) an FFT implementation of a Gaussian filter.

Recursive implementation of the Gaussian filter.pdf

Рекурсивный фильтр

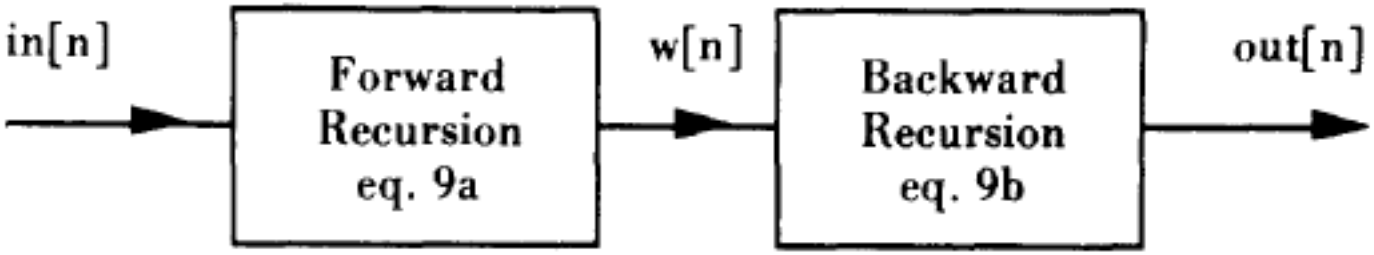


Fig. 7. Gaussian filter as concatenation of two filters.

Рекурсивный фильтр

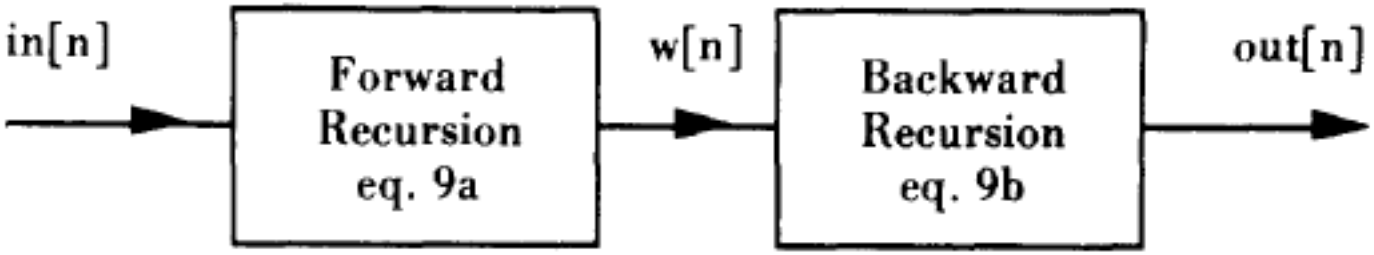


Fig. 7. Gaussian filter as concatenation of two filters.

forward: Слева направо

$$w[n] = B\,\mathrm{in}[n] + (b_1 w[n-1] + b_2 w[n-2] + b_3 w[n-3])/b_0 \qquad (9a)$$

and

backward: Справа налево

$$\mathrm{out}[n] = Bw[n] + (b_1 \mathrm{out}[n+1] + b_2 \mathrm{out}[n+2] + b_3 \mathrm{out}[n+3])/b_0 . \qquad (9b)$$

Рекурсивный фильтр

forward:

$$w[n] = B\,\mathrm{in}[n] + (b_1 w[n-1] + b_2 w[n-2] + b_3 w[n-3])/b_0 \qquad (9a)$$

Слева направо

and

backward:

$$\mathrm{out}[n] = Bw[n] + (b_1 \mathrm{out}[n+1] + b_2 \mathrm{out}[n+2] + b_3 \mathrm{out}[n+3])/b_0. \qquad (9b)$$

Справа налево

where

$$b_0 = 1.57825 + (2.44413q) + (1.4281q^2) + (0.422205q^3),$$

$$b_1 = (2.44413q) + (2.85619q^2) + (1.26661q^3), \qquad (8c)$$

$$b_2 = -((1.4281q^2) + (1.26661q^3)),$$

$$b_3 = 0.422205q^3.$$

$$B = 1 - ((b_1 + b_2 + b_3)/b_0). \qquad (10)$$

Рекурсивный фильтр

Вычисление q

$$q = \begin{cases} 0.98711\sigma_0 - 0.96330, & \sigma_0 \geqslant 2.5, \\ 3.97156 - 4.14554\sqrt{1 - 0.26891\sigma_0}, & \\ \quad 0.5 \leqslant \sigma_0 \leqslant 2.5. & \end{cases} \qquad (11b)$$

Рекурсивный фильтр

Вычисление q

$$q = \begin{cases} 0.98711\sigma_0 - 0.96330, & \sigma_0 \geqslant 2.5, \\ 3.97156 - 4.14554\sqrt{1 - 0.26891\sigma_0}, & \\ 0.5 \leqslant \sigma_0 \leqslant 2.5. & \end{cases} \qquad (11b)$$

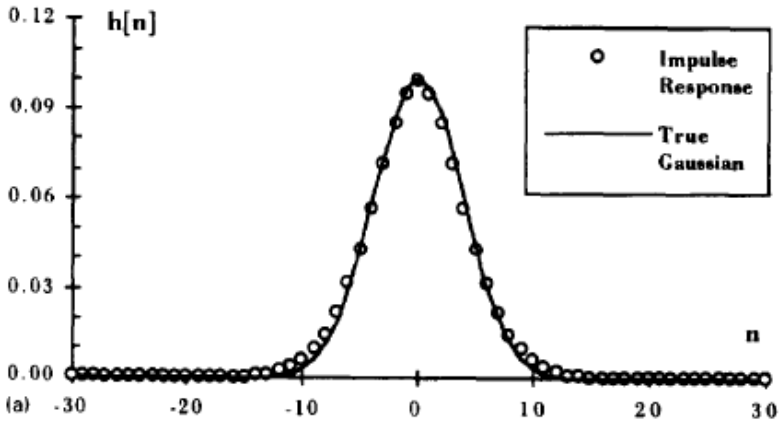


Fig. 10(a). Impulse response $h[n]$ resulting from the application of Eqs. (9a) and (9b) to the input signal $in[n] = \delta[n]$. The continuous curve is $g(x \mid \sigma = 4.0)$. The value of q used in Eq. (8c) is $q = 3.0$.

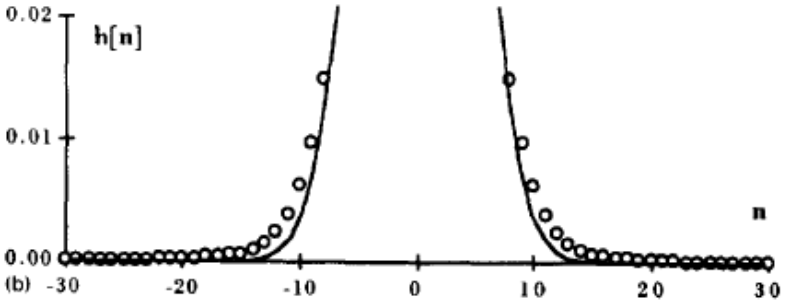
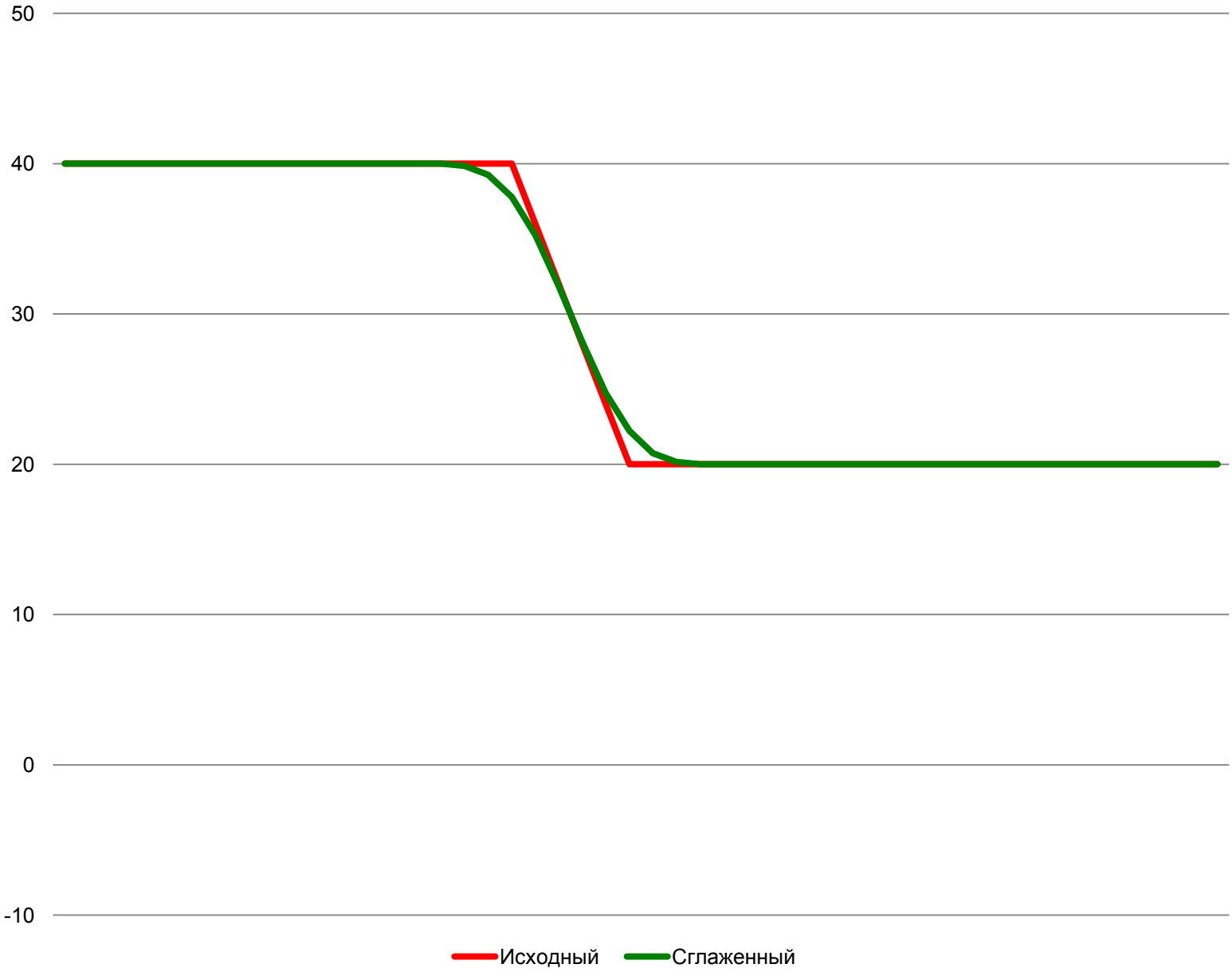


Fig. 10(b). Enlargement of $h[n]$ resulting from the application of Eqs. (9a) and (9b) to the input signal $in[n] = \delta[n]$ with $q = 3.0$. Note the values of the vertical scale.

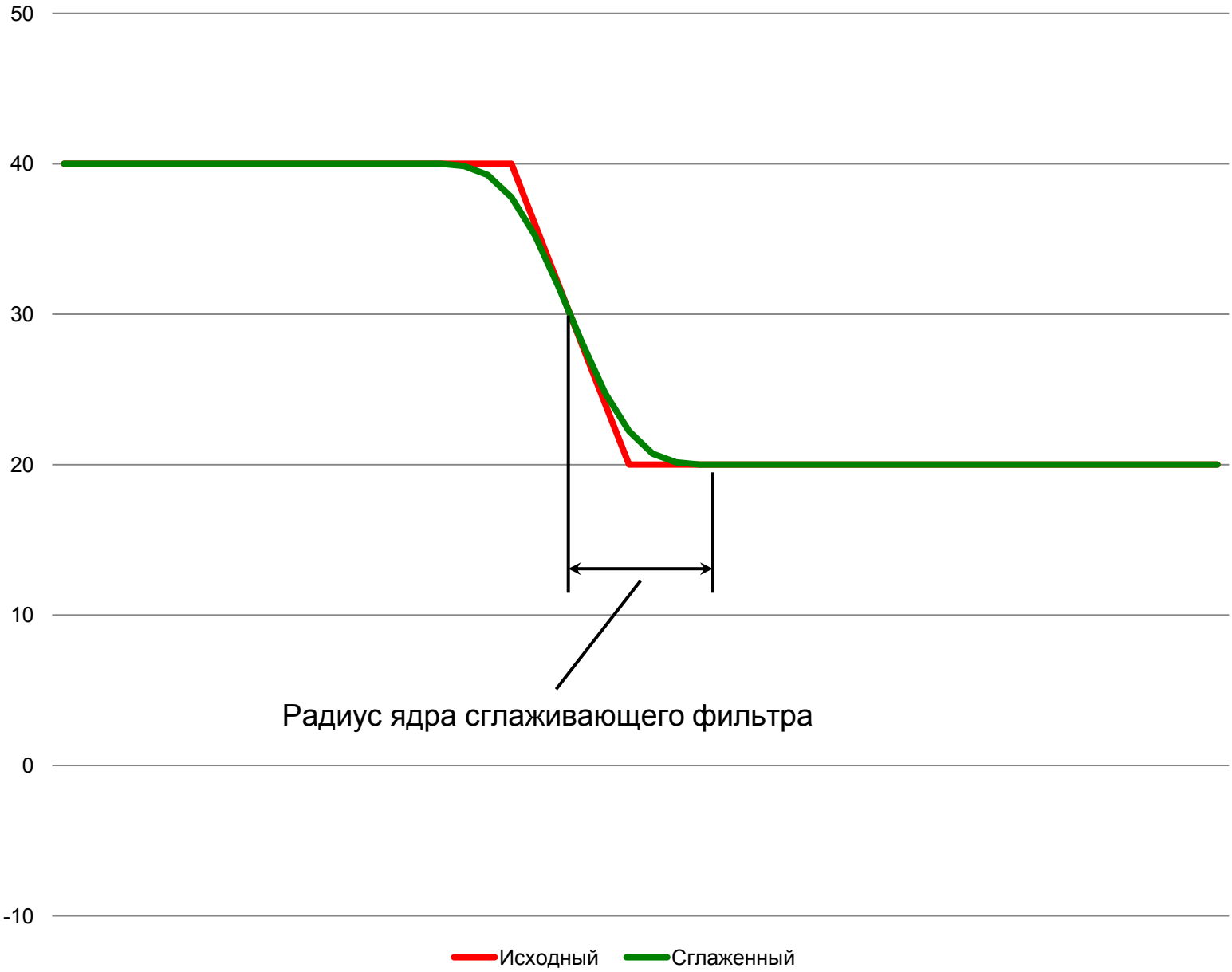
Нерезкая маска (USM)

Нерезкая маска (USM)

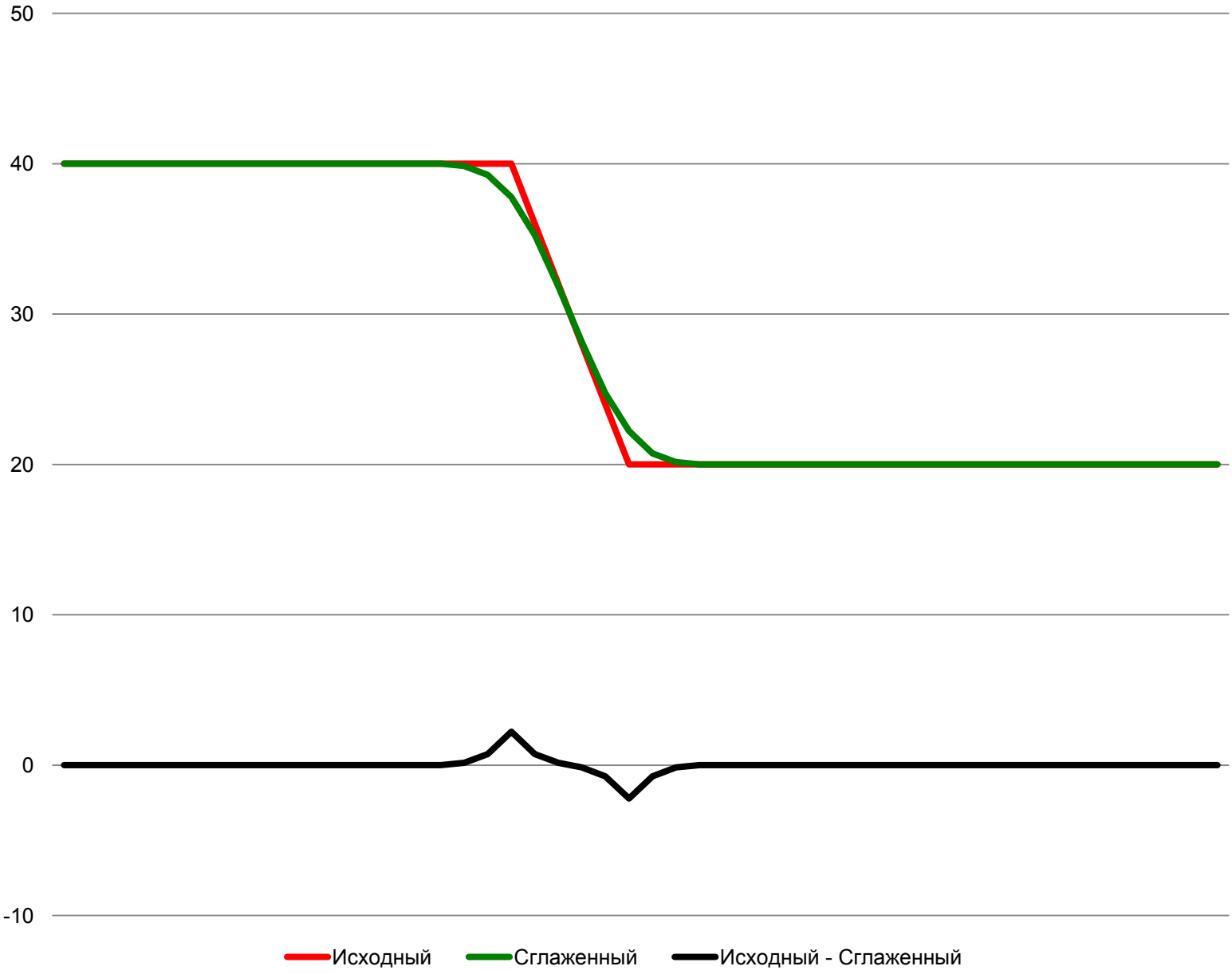
Нерезкая маска (USM)



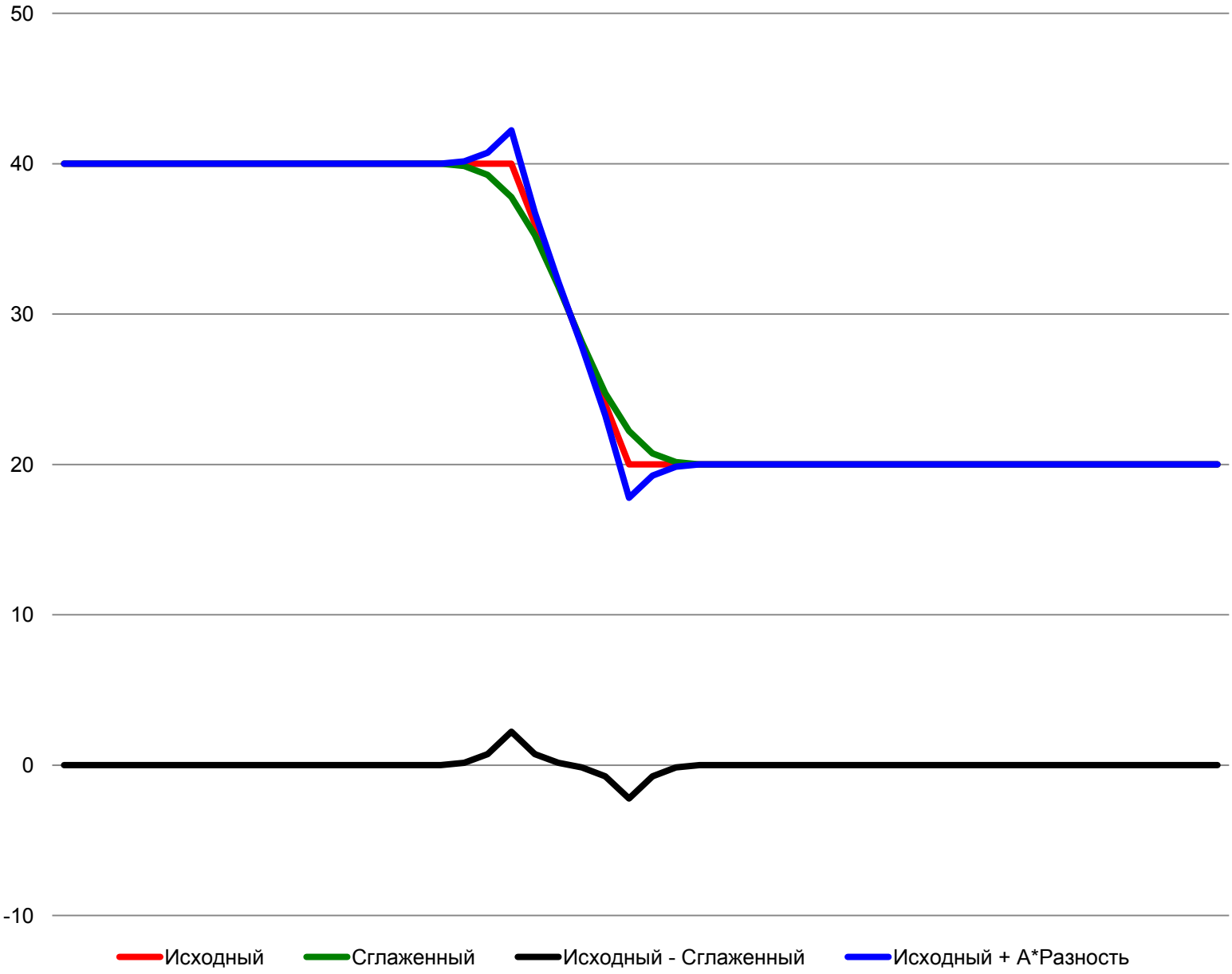
Нерезкая маска (USM)



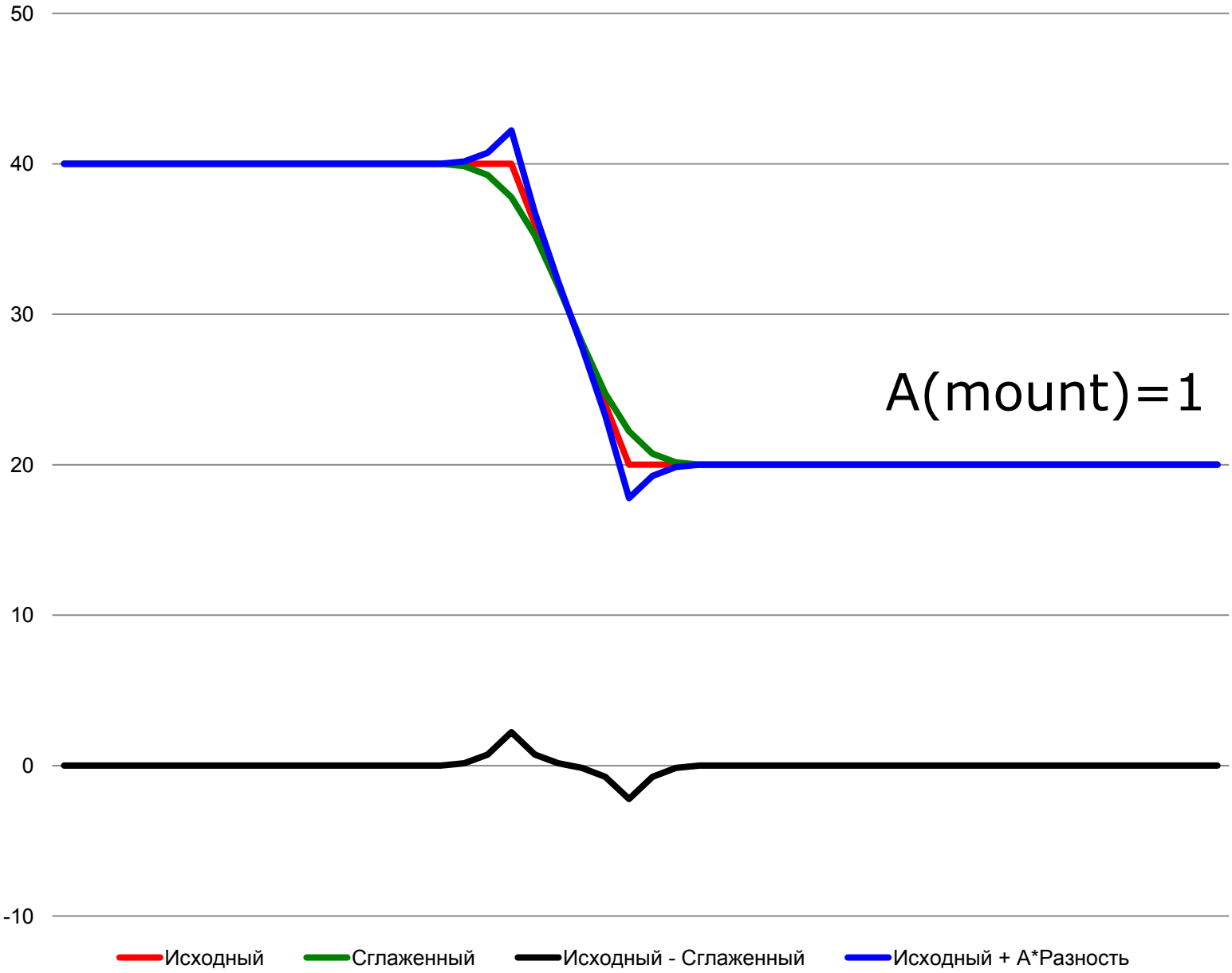
Нерезкая маска (USM)



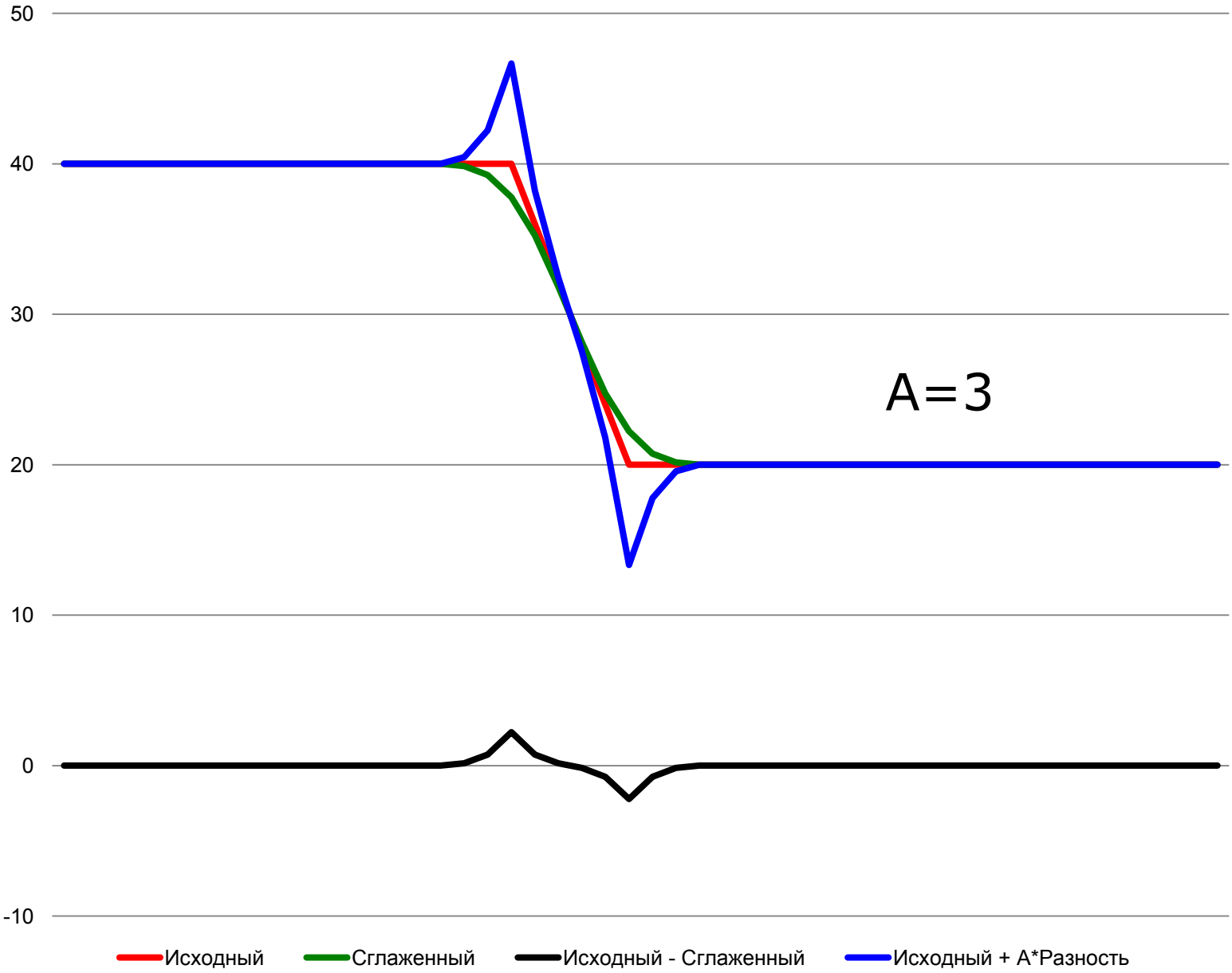
Нерезкая маска (USM)



Нерезкая маска (USM)



Нерезкая маска (USM)



Исходный (100%)

USM. Radius 1.0, Amount 100%

Резкость (простая реализация)

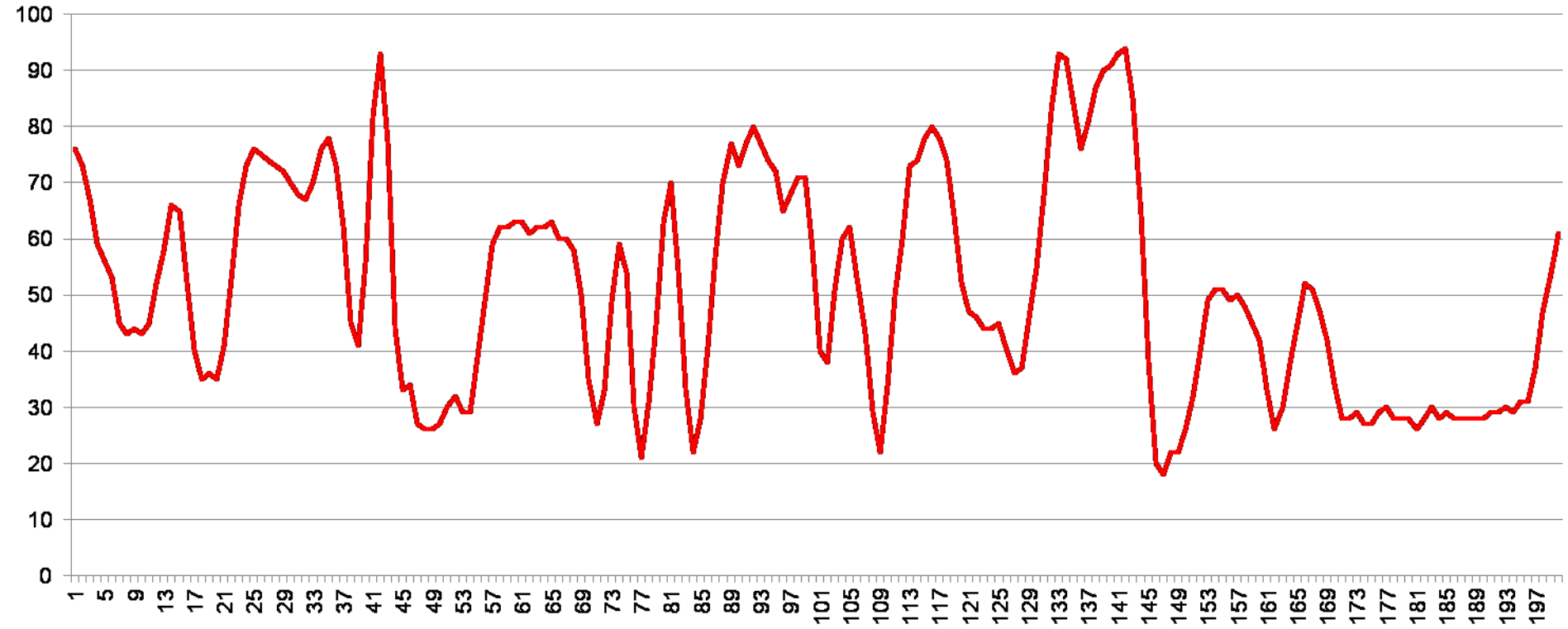
- Радиус размытия мал (обычно около 1 пикселя) – использовать «настоящий» фильтр Гаусса нет особого смысла.
- Усреднение по трем точкам:

$$x_i = Ax_i + (1-A)\cdot x_{cp};\ x_{cp} = \frac{x_{i-1} + x_i + x_{i+1}}{3};\ i = 2..n-1$$

- Сепарабельность
- Возможно усреднение по 5 точкам

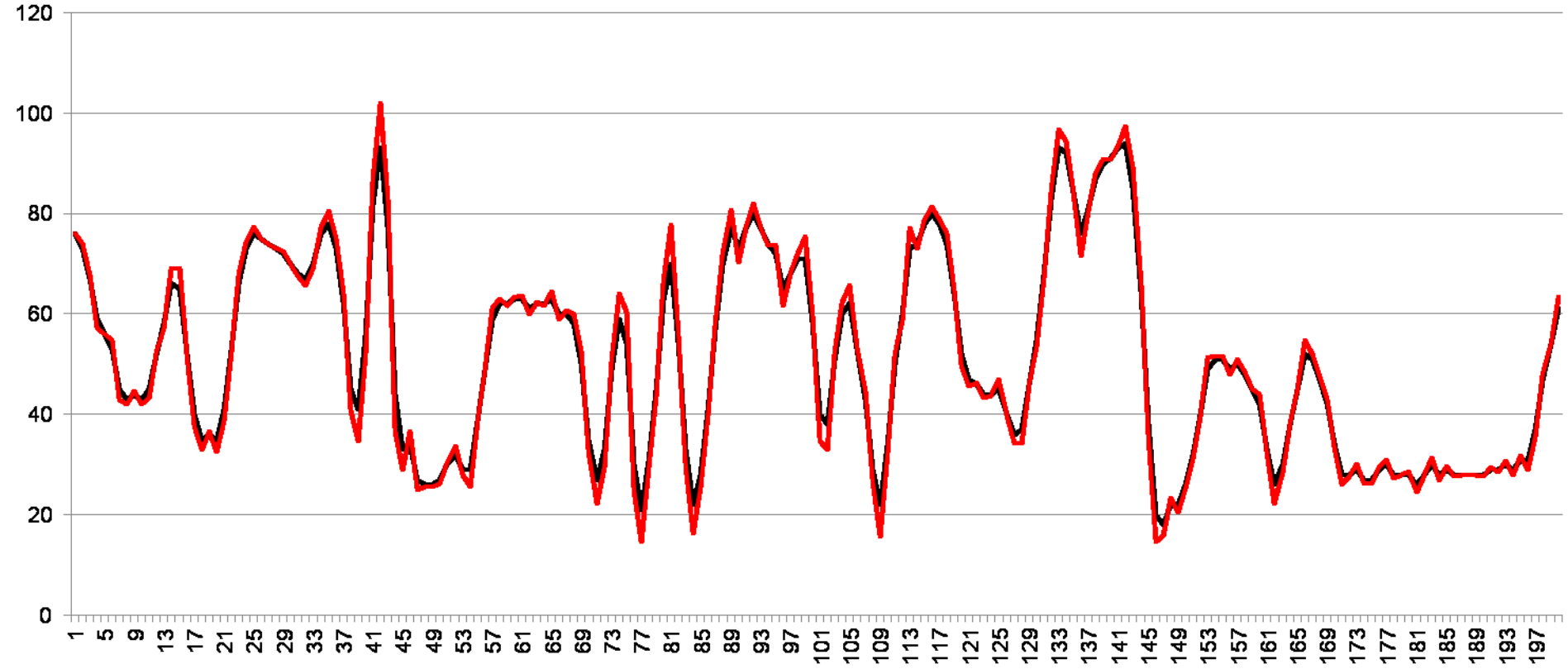
```
//По строкам
#pragma omp parallel for
for(int i = 1; i < h-1; i++) {
   float x = arr[i][0];
   for(int j = 1; j < w-1; j++) {
      float a = arr[i][j-1];
      float b = arr[i][j];
      float c = arr[i][j+1];
      float y = alpha*b + (1-alpha)*(a+b+c)/3.0f;
      arr[i][j-1] = x;
      x = y;
   }
   arr[i][w-2] = x;
}
```

Профиль строки реального снимка



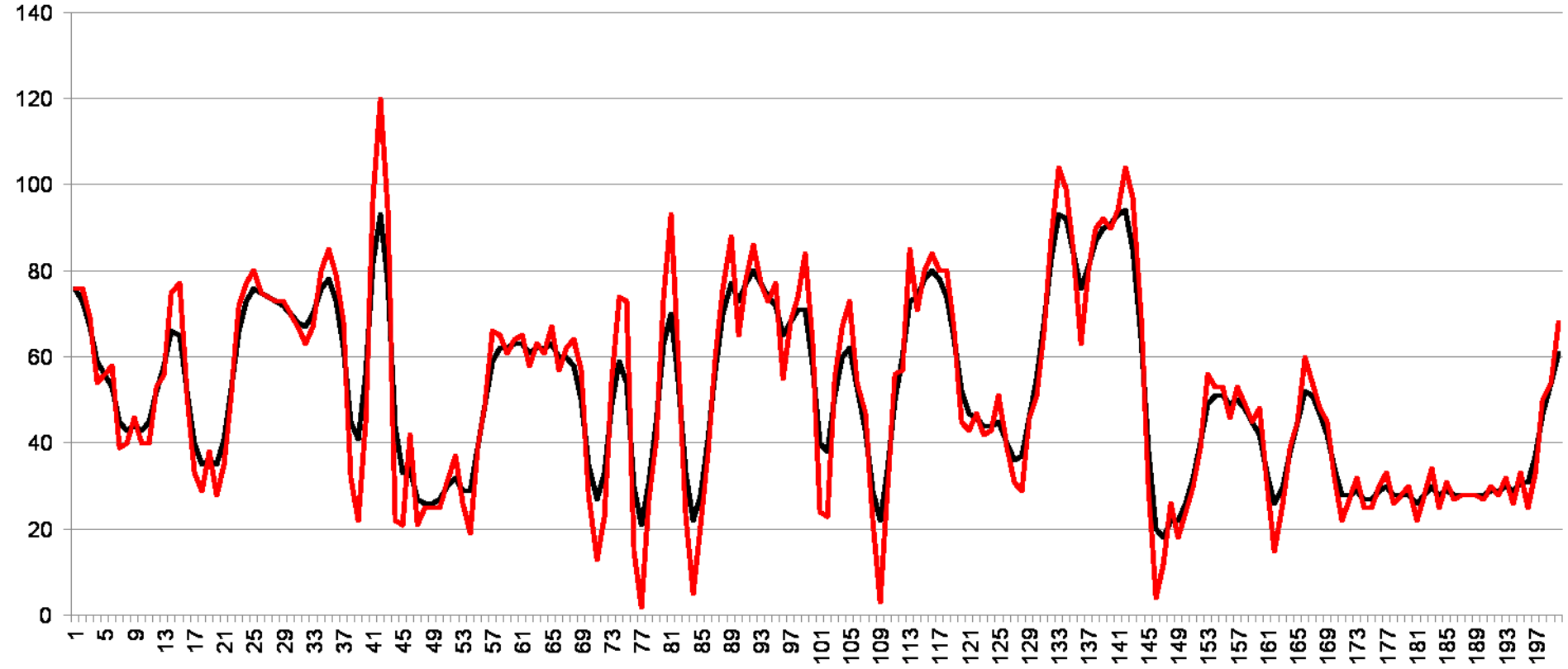
A = 1

Профиль строки реального снимка



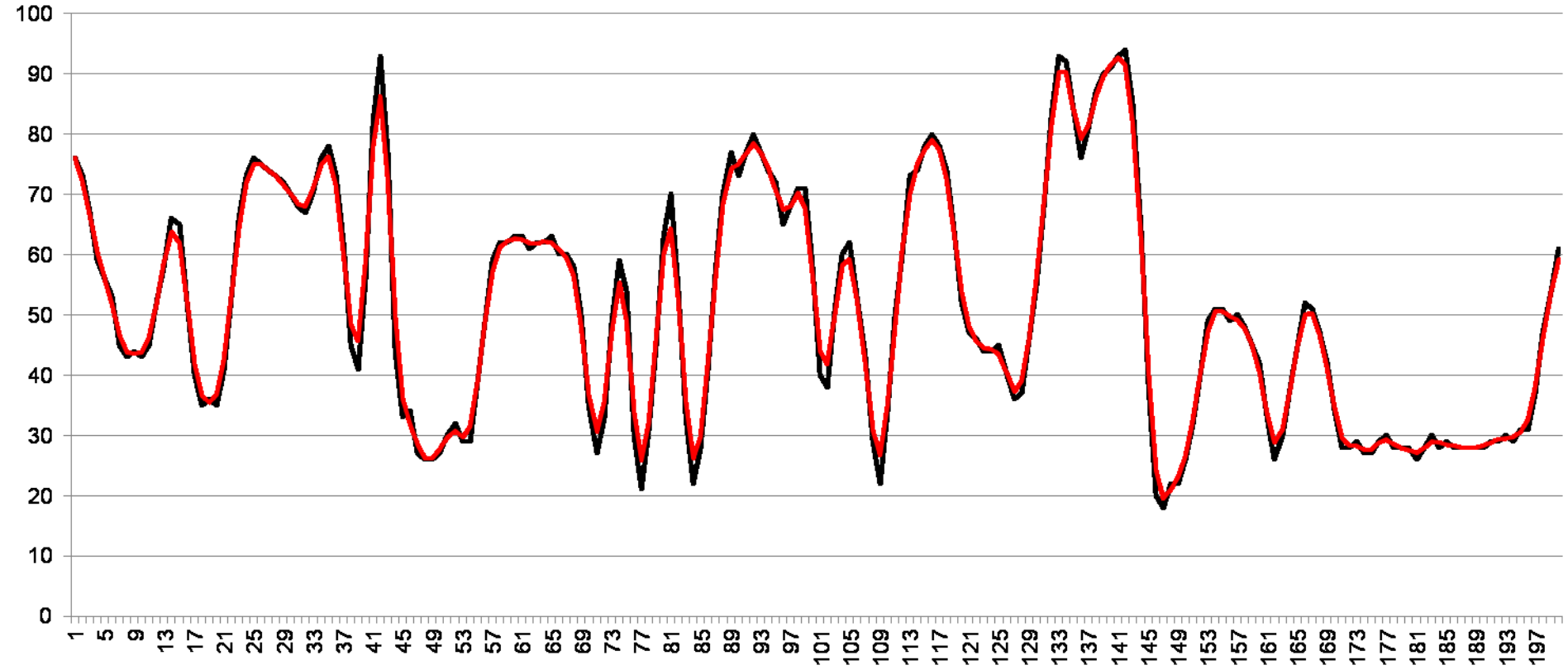
A = 2

Профиль строки реального снимка



A = 4

Профиль строки реального снимка



A = 0.25

Локальный контраст (Clarity)

Локальный контраст (Clarity)

- Степень отличия значения пикселя (например, светлоты) от среднего по окрестности

Локальный контраст (Clarity)

- Степень отличия значения пикселя (например, светлоты) от среднего по окрестности

- Clarity = USM с большим радиусом размытия

Локальный контраст (Clarity)

- Степень отличия значения пикселя (например, светлоты) от среднего по окрестности
- Clarity = USM с большим радиусом размытия
- «Hiraloam» © Д. Маргулис: High radius, low amount

Исходный (100%)

Clarity (USM. Radius 30.0, Amount 20%)

Тени/Света (Shadow/Highlight)

- В точках, где средняя по окрестности (σ~30) светлота больше заданной экспозиция (светлота) уменьшается

- В точках, где средняя по окрестности светлота меньше заданной, экспозиция (светлота) увеличивается

Дополнительные главы компьютерной графики

Лекция 9

Copyright © С. Свердлов, 2009-2017
http://www.uni-vologda.ac.ru/~c3c

Пространственные преобразования

Пространственные преобразования

- Изменение размера (resize)

Пространственные преобразования

- Изменение размера (resize)
- Исправление перспективных искажений

Пространственные преобразования

- Изменение размера (resize)
- Исправление перспективных искажений
- Исправление дисторсии

Пространственные преобразования

- Изменение размера (resize)
- Исправление перспективных искажений
- Исправление дисторсии
- Преобразование проекции для панорам

Пространственные преобразования

- Изменение размера (resize)
- Исправление перспективных искажений
- Исправление дисторсии
- Преобразование проекции для панорам
- Выпрямление линий (панорамы)

Пространственные преобразования

- Изменение размера (resize)
- Исправление перспективных искажений
- Исправление дисторсии
- Преобразование проекции для панорам
- Выпрямление линий (панорамы)
- Склейка при брекетинге

Пространственные преобразования

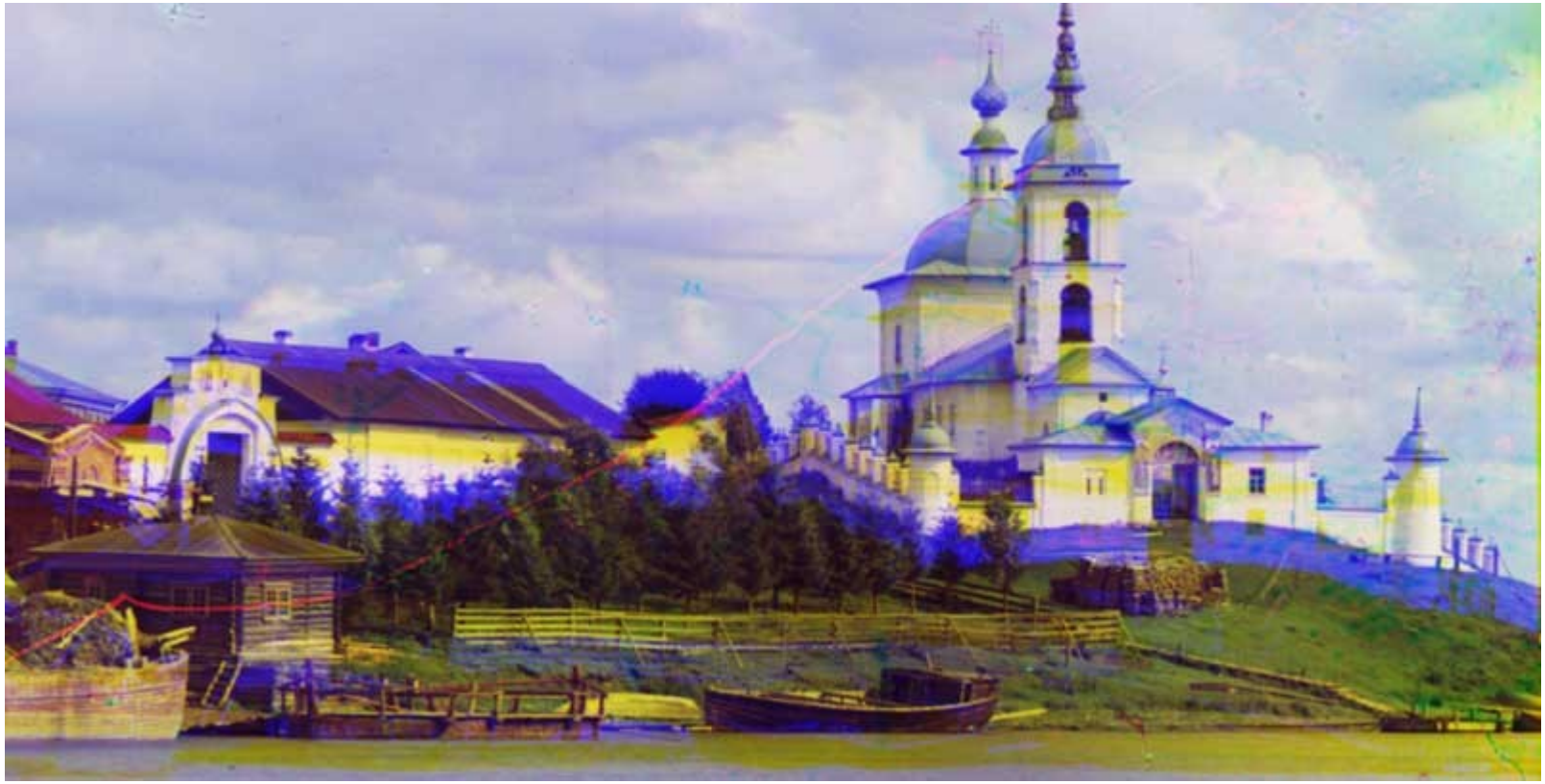
- Изменение размера (resize)
- Исправление перспективных искажений
- Исправление дисторсии
- Преобразование проекции для панорам
- Выпрямление линий (панорамы)
- Склейка при брекетинге
- Морфинг (панорамы)

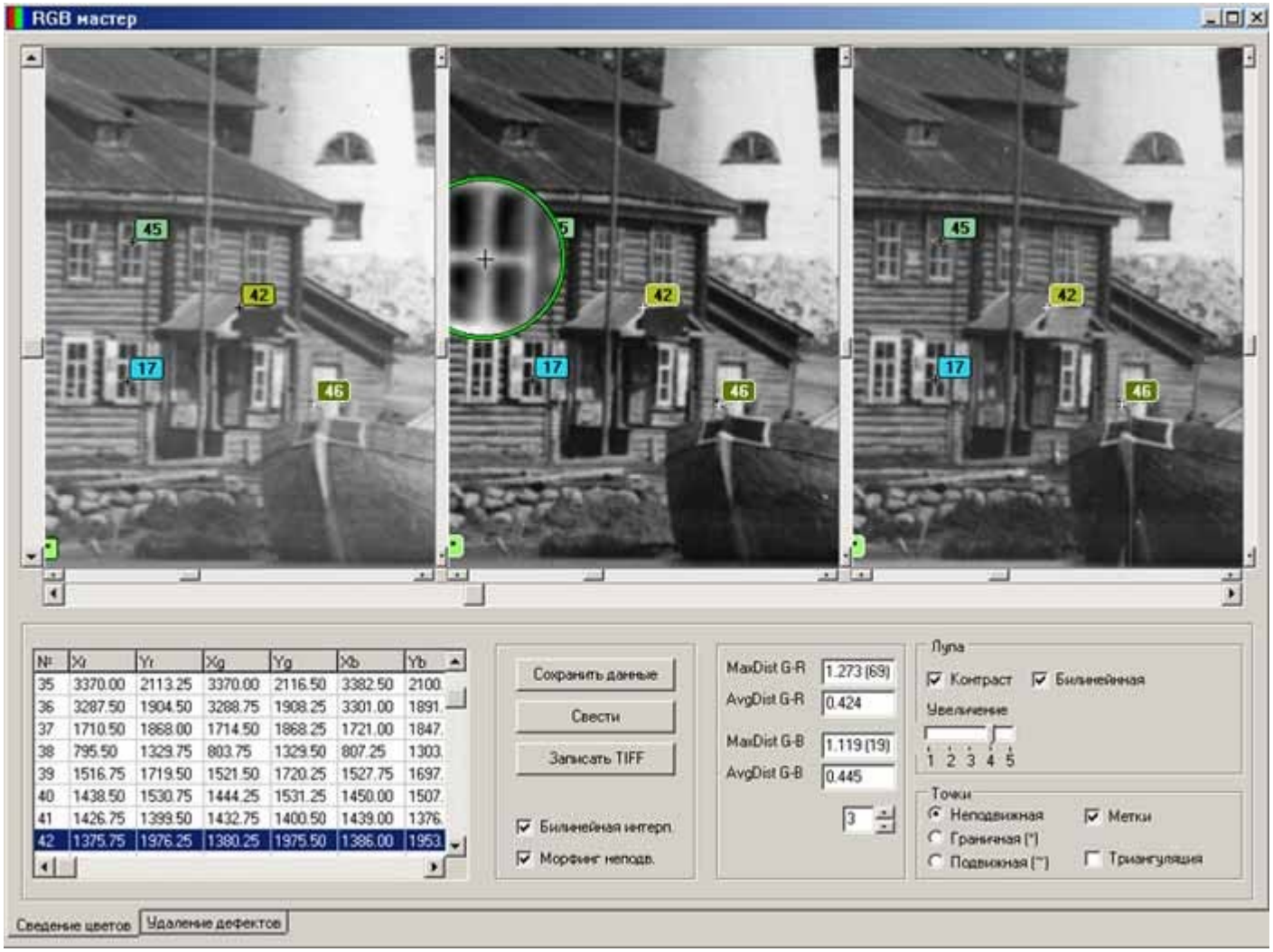
Пространственные преобразования

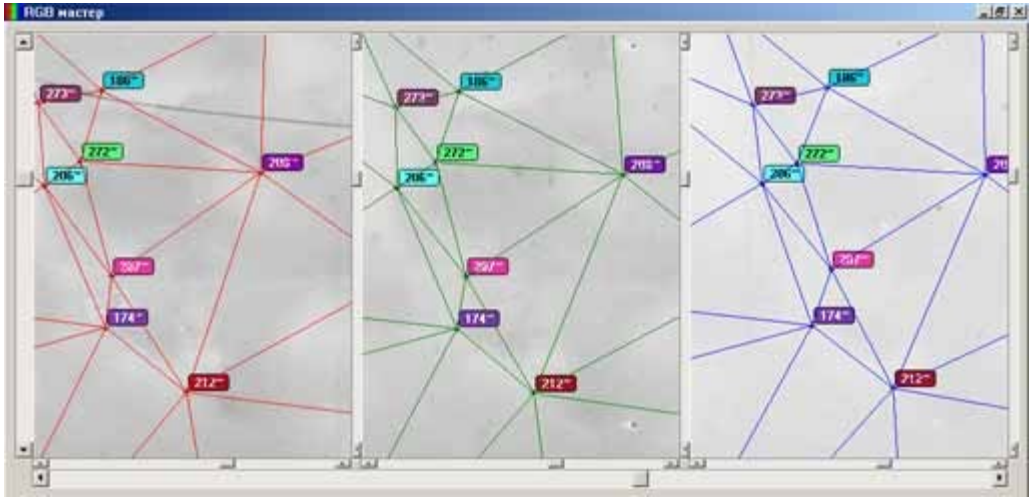
- Изменение размера (resize)
- Исправление перспективных искажений
- Исправление дисторсии
- Преобразование проекции для панорам
- Выпрямление линий (панорамы)
- Склейка при брекетинге
- Морфинг (панорамы)
- …

Пример. Реставрация фотографий С. М. Прокудина-Горского

Общий вид Горицкого монастыря, 1909




RGB мастер
45
42
17
46
№ Xr Yr Xg Yg Xb Yb
35 3370.00 2113.25 3370.00 2116.50 3382.50 2100.
36 3287.50 1904.50 3288.75 1908.25 3301.00 1891.
37 1710.50 1868.00 1714.50 1868.25 1721.00 1847.
38 795.50 1329.75 803.75 1329.50 807.25 1303.
39 1516.75 1719.50 1521.50 1720.25 1527.75 1697.
40 1438.50 1530.75 1444.25 1531.25 1450.00 1507.
41 1426.75 1399.50 1432.75 1400.50 1439.00 1376.
42 1375.75 1976.25 1380.25 1975.50 1386.00 1953.
Сохранить данные
Свести
Записать TIFF
Билинейная интерп.
Морфинг неподв.
MaxDist G-R 1.273 (69)
AvgDist G-R 0.424
MaxDist G-B 1.119 (19)
AvgDist G-B 0.445
3
Лупа
Контраст
Билинейная
Увеличение
1 2 3 4 5
Точки
Неподвижная
Граничная [*]
Подвижная [**]
Метки
Триангуляция
Сведение цветов
Удаление дефектов


RGB мастер

Общий принцип

- Исходное и преобразованное изображения – прямоугольные матрицы пикселей
- Нужно определить цвет каждого пикселя преобразованного изображения
- В ходе обработки надо пройти по матрице пикселей **преобразованного** изображения

$$x' = f_x(x, y)$$

$$y' = f_y(x, y)$$

x, y – координаты на исходном изображении

x', y' – координаты на преобразованном изображении

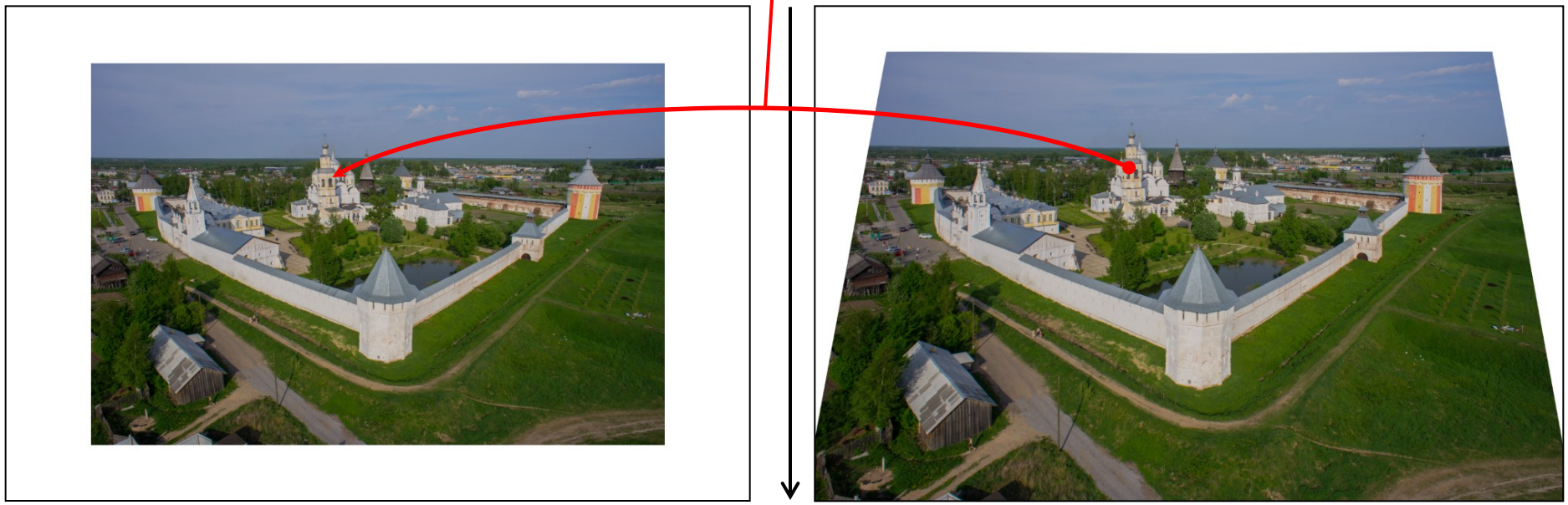
Решение системы уравнений

$$x = f_x^{-1}(x', y')$$

$$y = f_y^{-1}(x', y')$$

$$x = f_x^{-1}(x', y')$$

$$y = f_y^{-1}(x', y')$$



Исходное

Преобразованное

Изменение (уменьшение) размера. Resize (downscaling)

Сепарабельность

- Метод ближайшего соседа
- Интерполяция (билинейная, бикубическая)
- Фильтры (Гаусса, Ланцоша, …)
- Алгоритмы, основанные на площадях

Метод ближайшего соседа

$$x = f_x^{-1}(x', y')$$

$$y = f_y^{-1}(x', y')$$

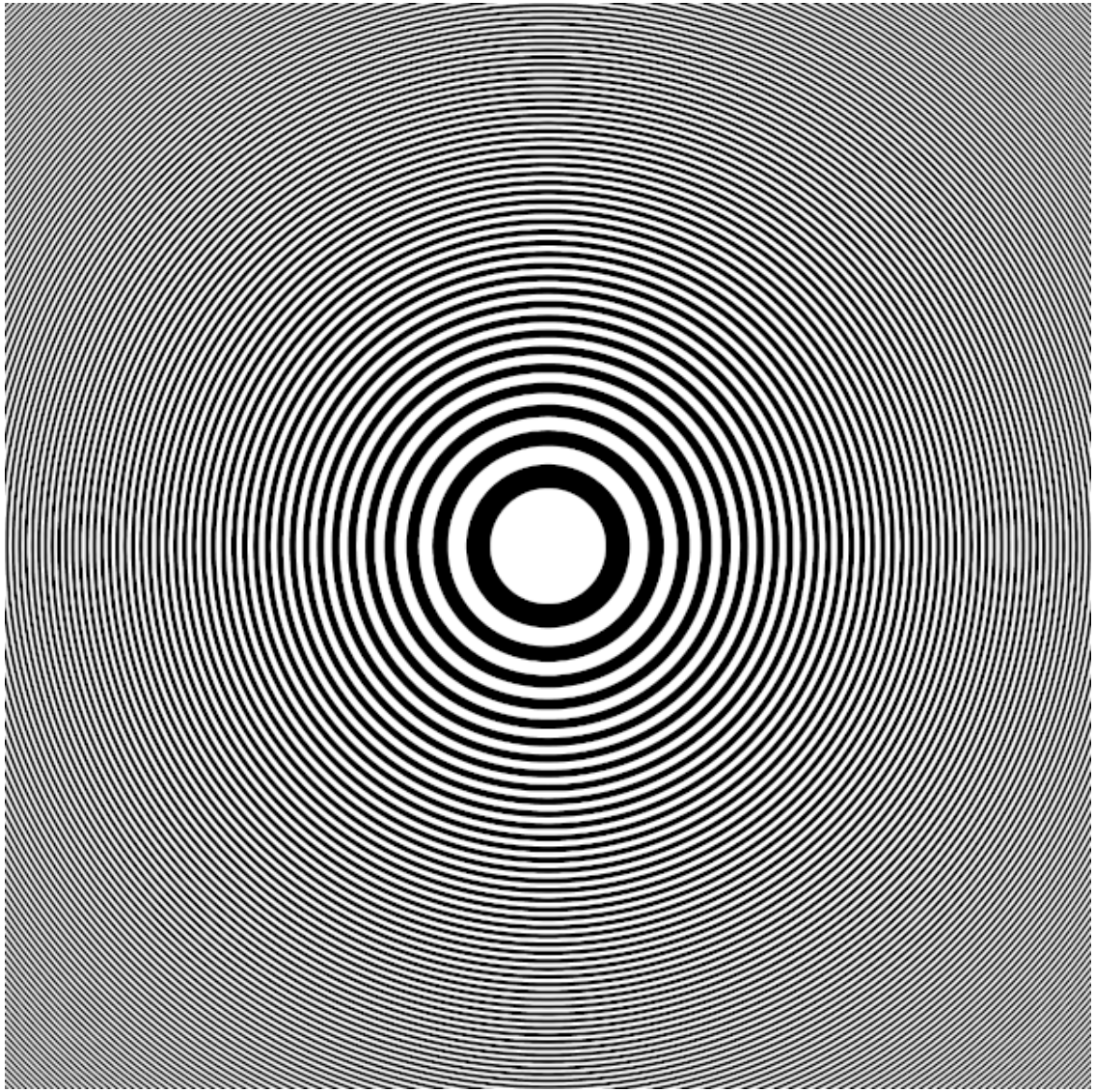
$$C'[y', x'] = C[round(y), round(x)]$$

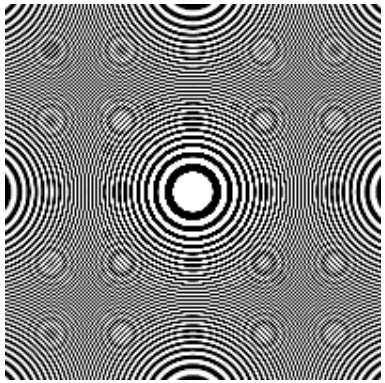
Учитывая сепарабельность, по строкам:

$$x = f_x^{-1}(x')$$

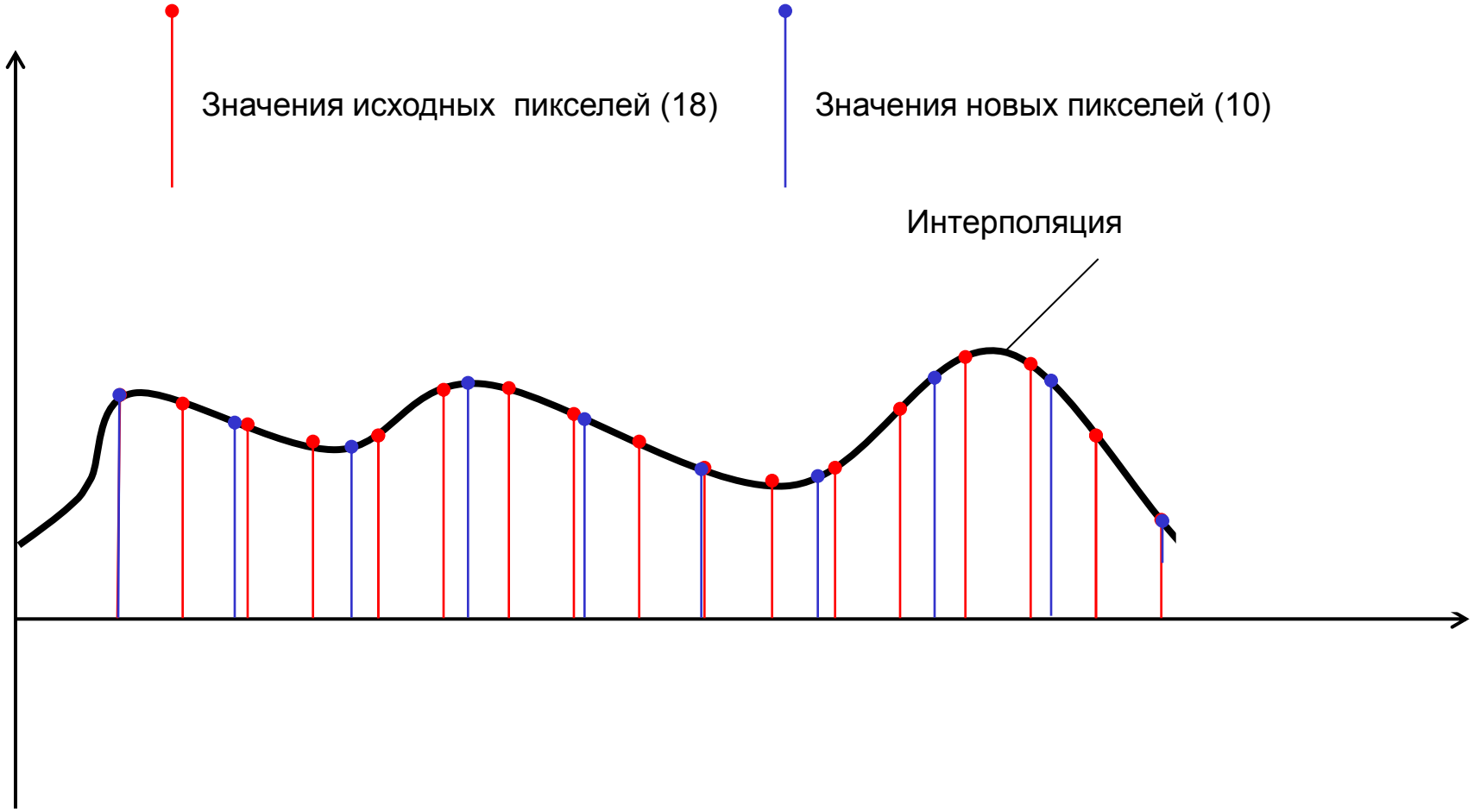
$$Line'[x'] = Line[round(x)]$$

Метод ближайшего соседа





Интерполяция.
Общая идея



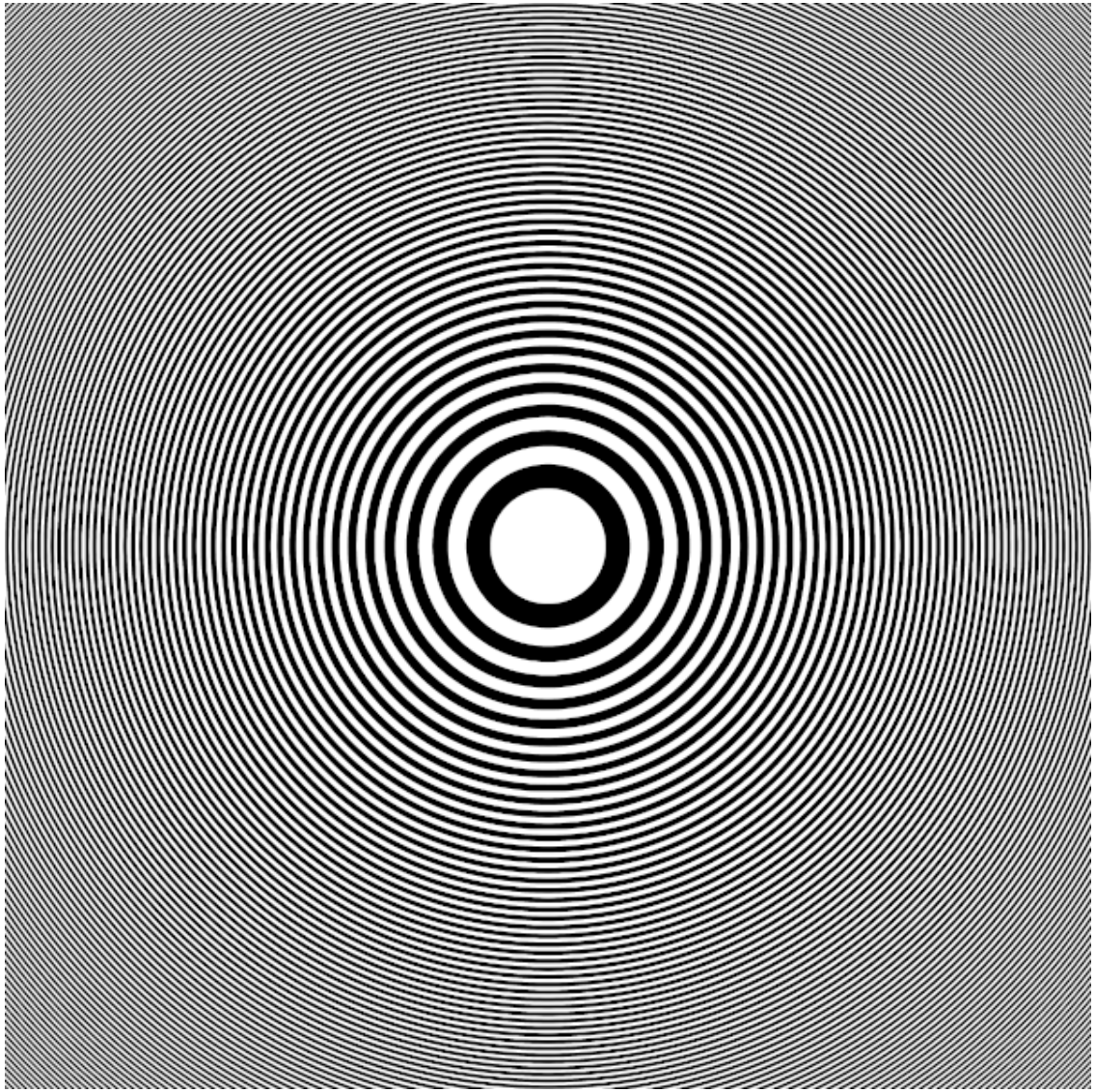
Билинейная интерполяция

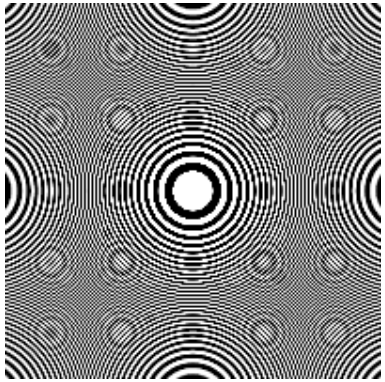
Учитывая сепарабельность, по строкам:

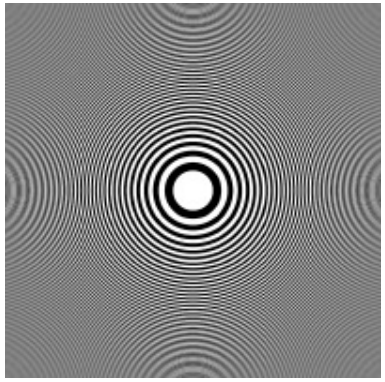
$$x = f_x^{-1}(x')$$

```
t := trunc(x);
f := x – t;
Line’[x’] :=(1-f)*Line[t] + f*Line[t+1];
```

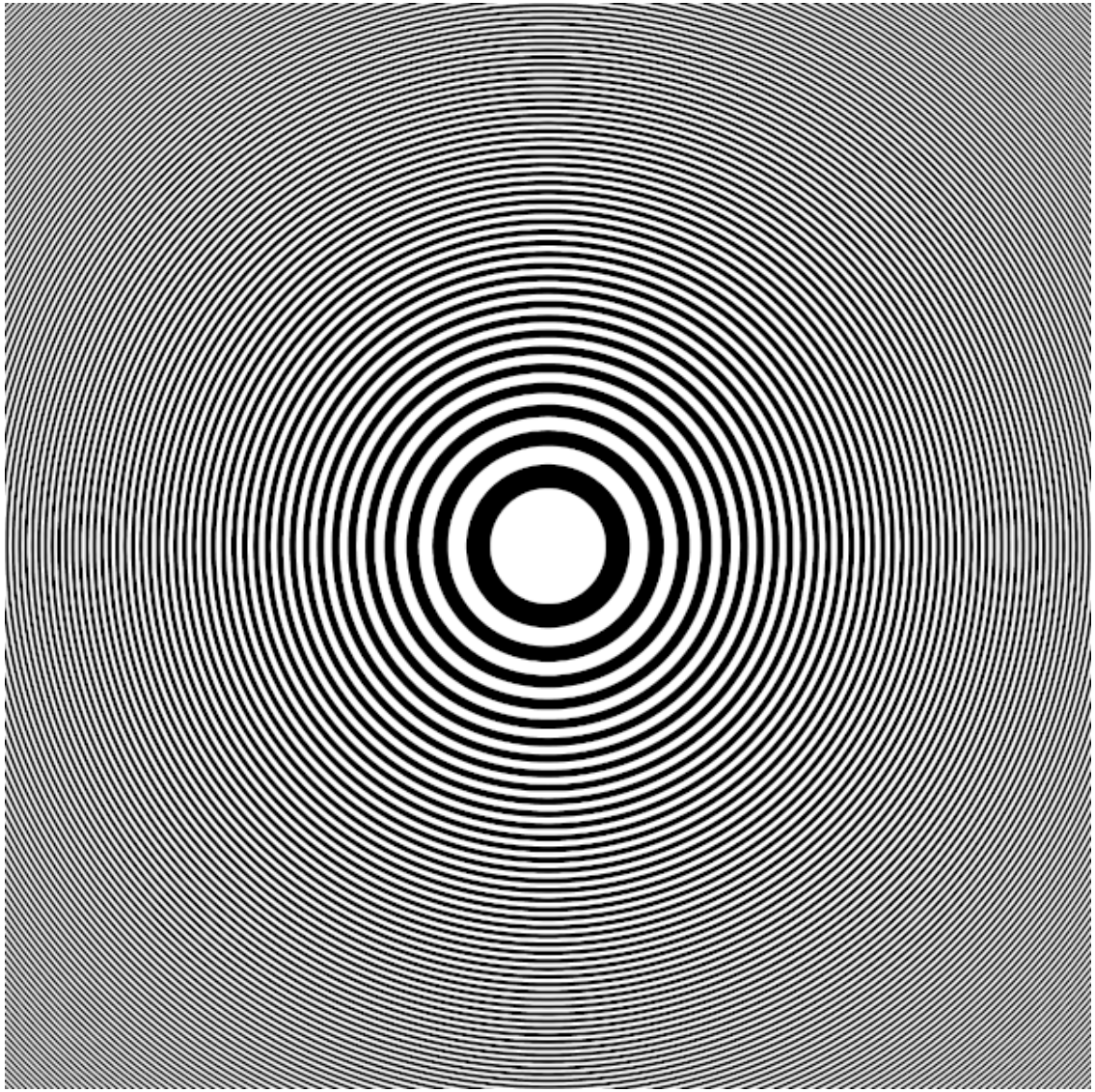
Билинейная интерполяция (PS – неправильная)

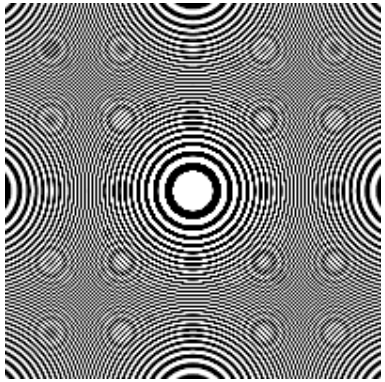


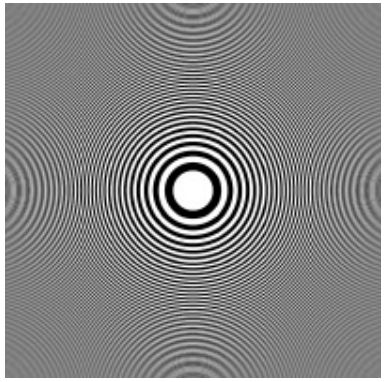


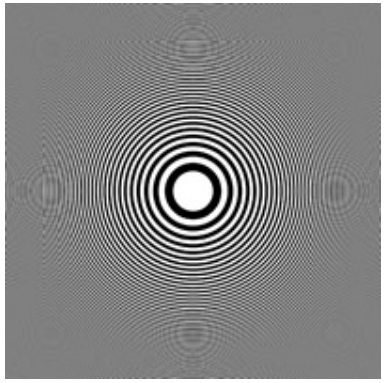


Бикубическая интерполяция (PS – неправильная)



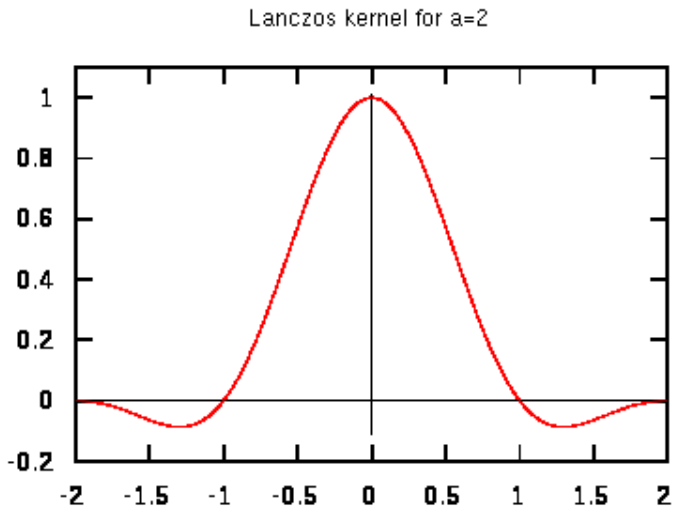


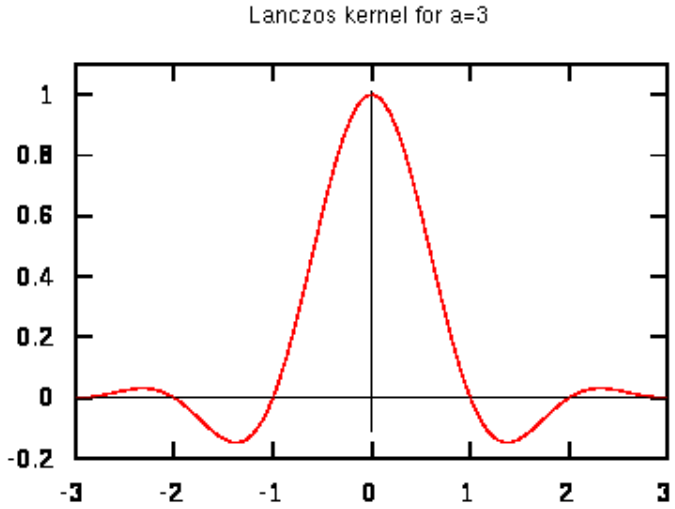




Использование фильтров

- Применение усредняющего фильтра с радиусом ядра, зависящим от коэффициента масштабирования (сильнее уменьшение – больше радиус)
  - Фильтр Гаусса
  - Фильтр Ланцоша
  - ...
- Выборка новых значений из размытого изображения.



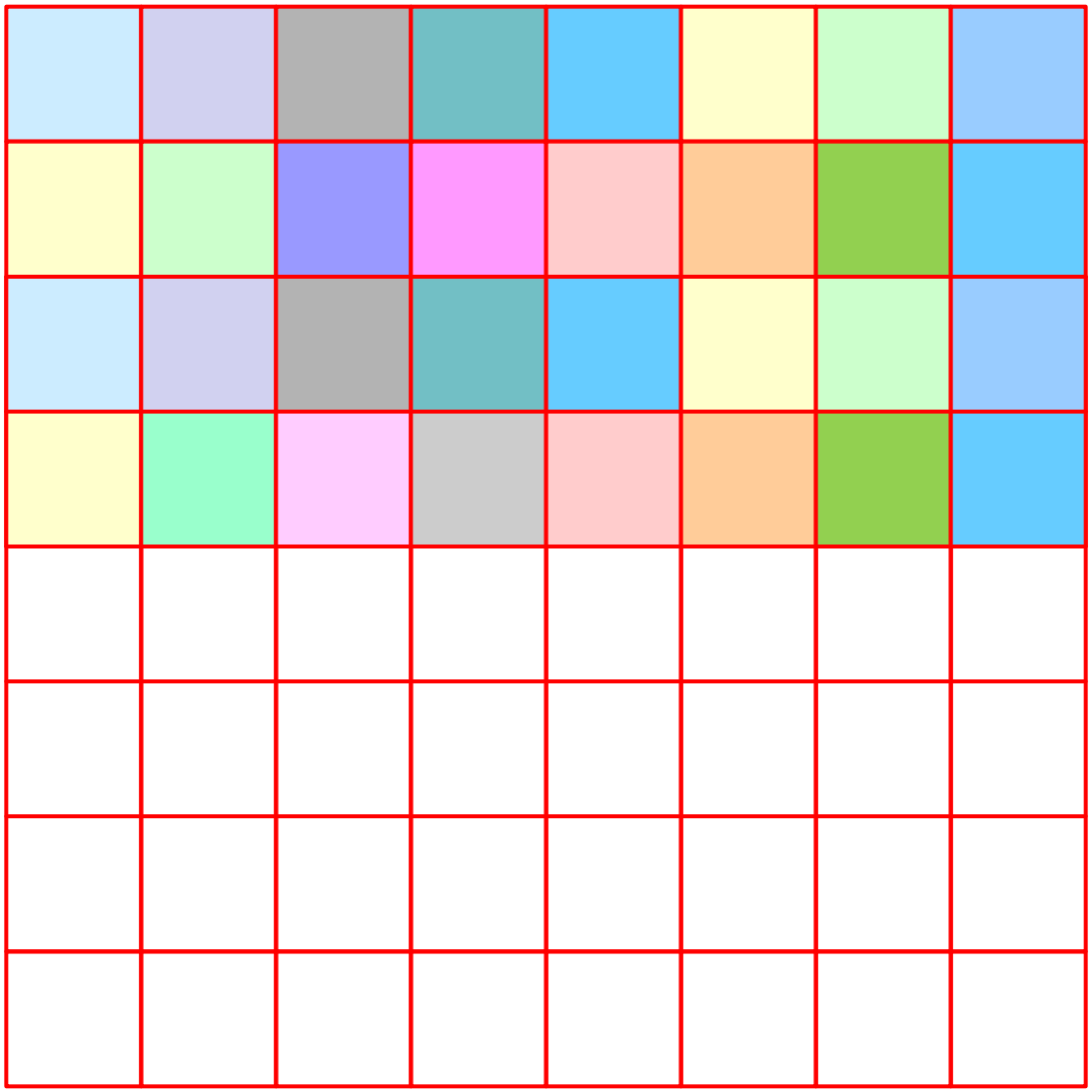


Ядро фильтра Ланцоша

Алгоритм, основанный на площадях

Алгоритм, основанный на площадях

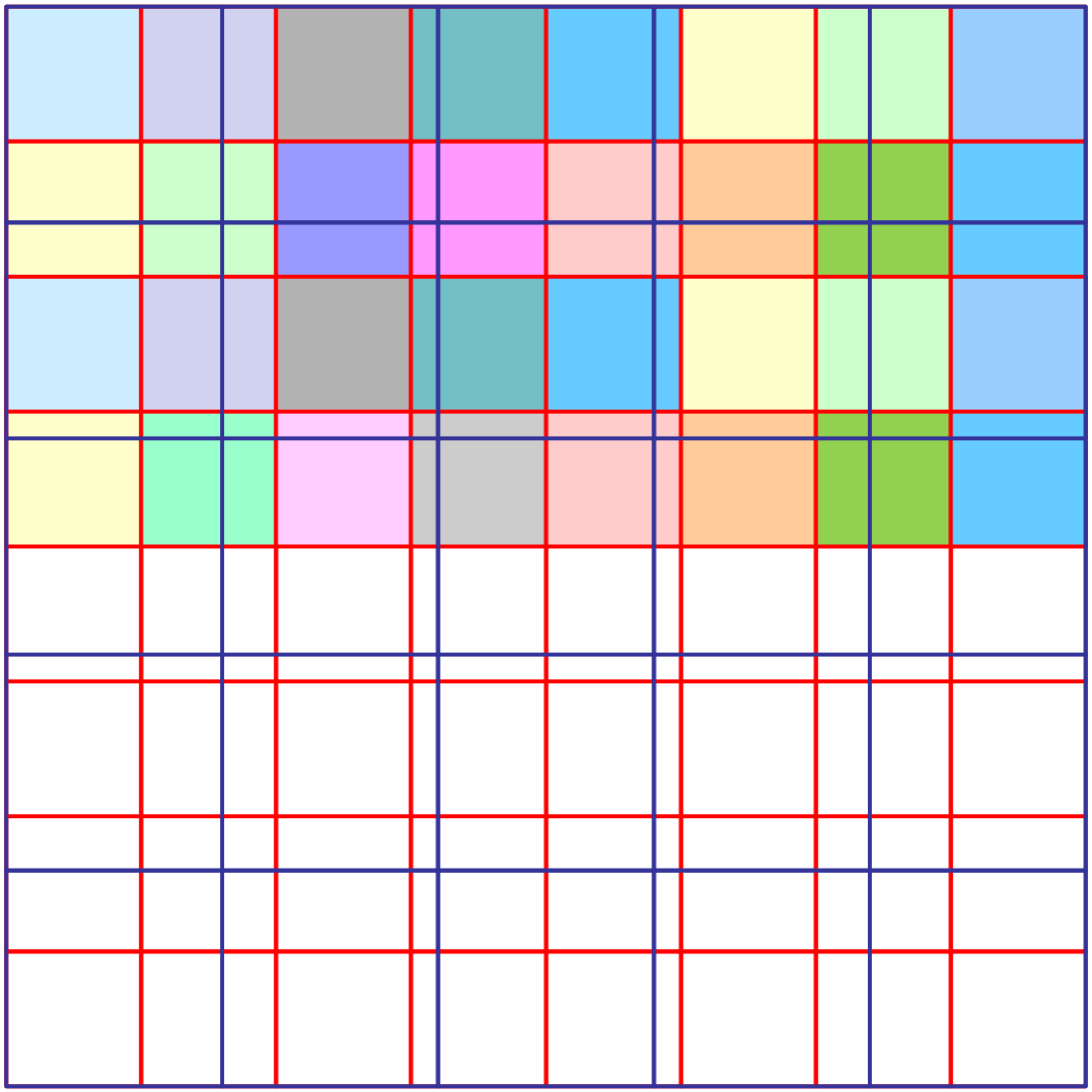
Исходный размер 8x8



Алгоритм, основанный на площадях

Исходный размер 8x8

Новый размер 5x5

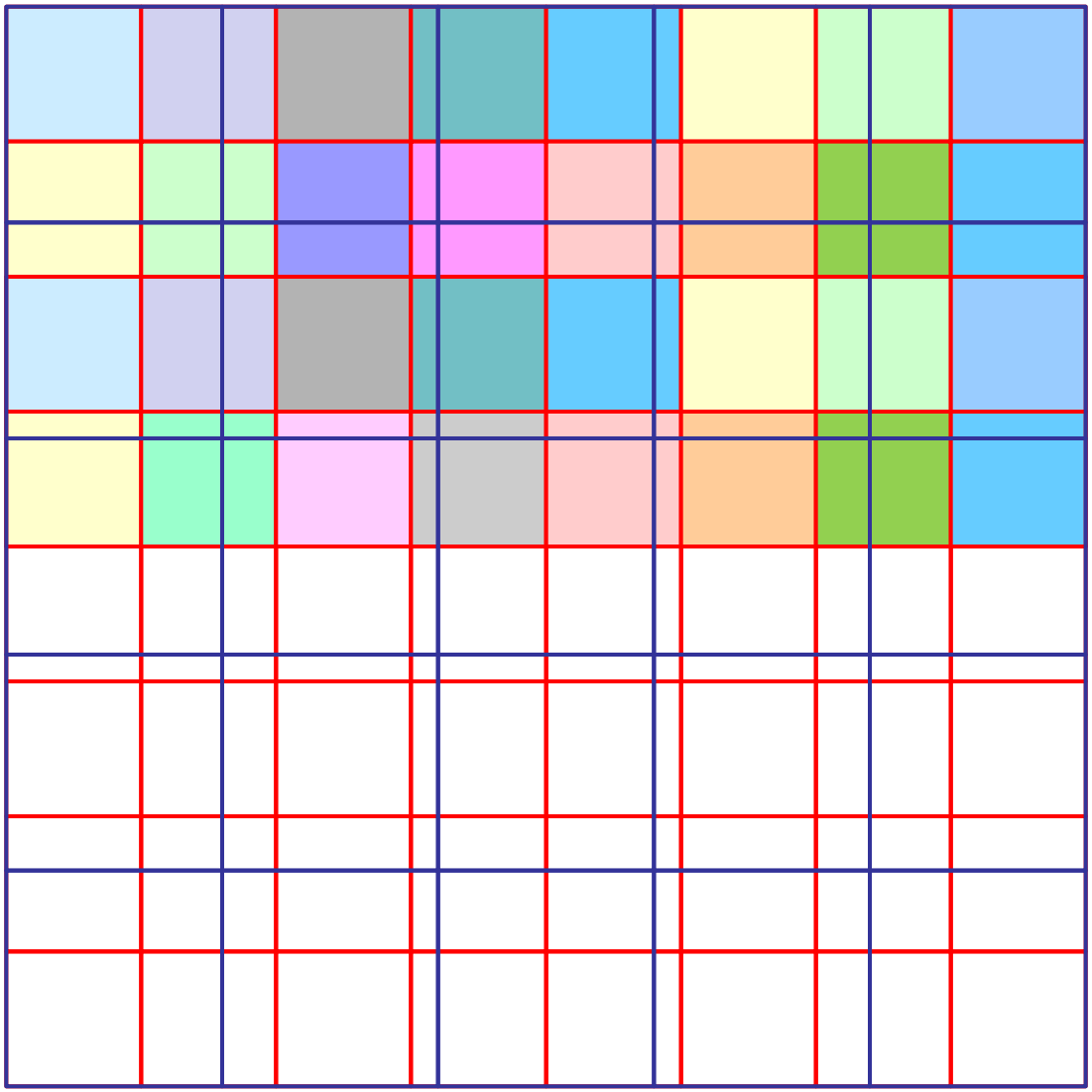


Алгоритм, основанный на площадях

Исходный размер 8x8

Новый размер 5x5

- Цвет пикселя – результат усреднения (интегрирования) по площади сенселя матрицы

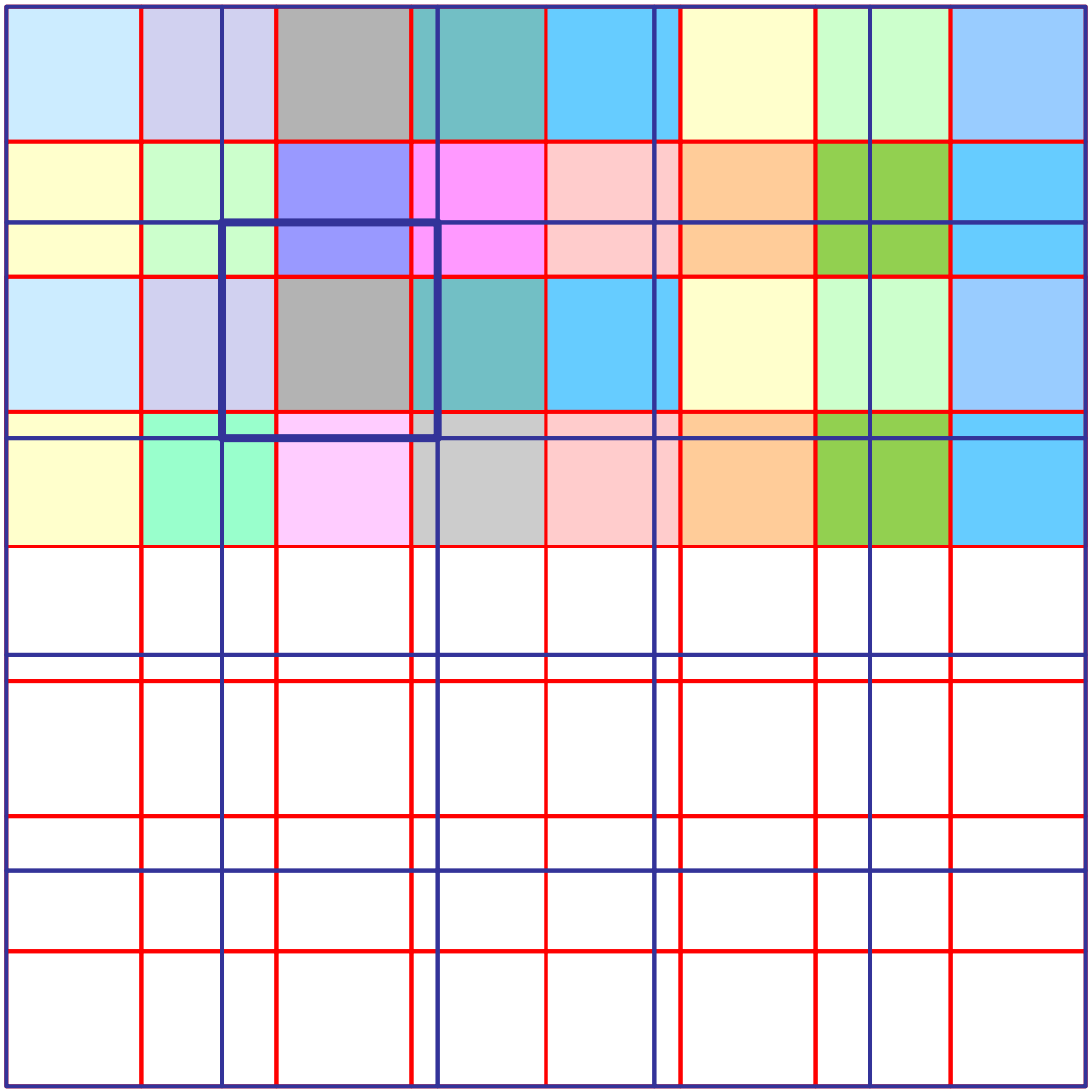


Алгоритм, основанный на площадях

Исходный размер 8x8

Новый размер 5x5

- Цвет пикселя – результат усреднения (интегрирования) по площади сенселя матрицы

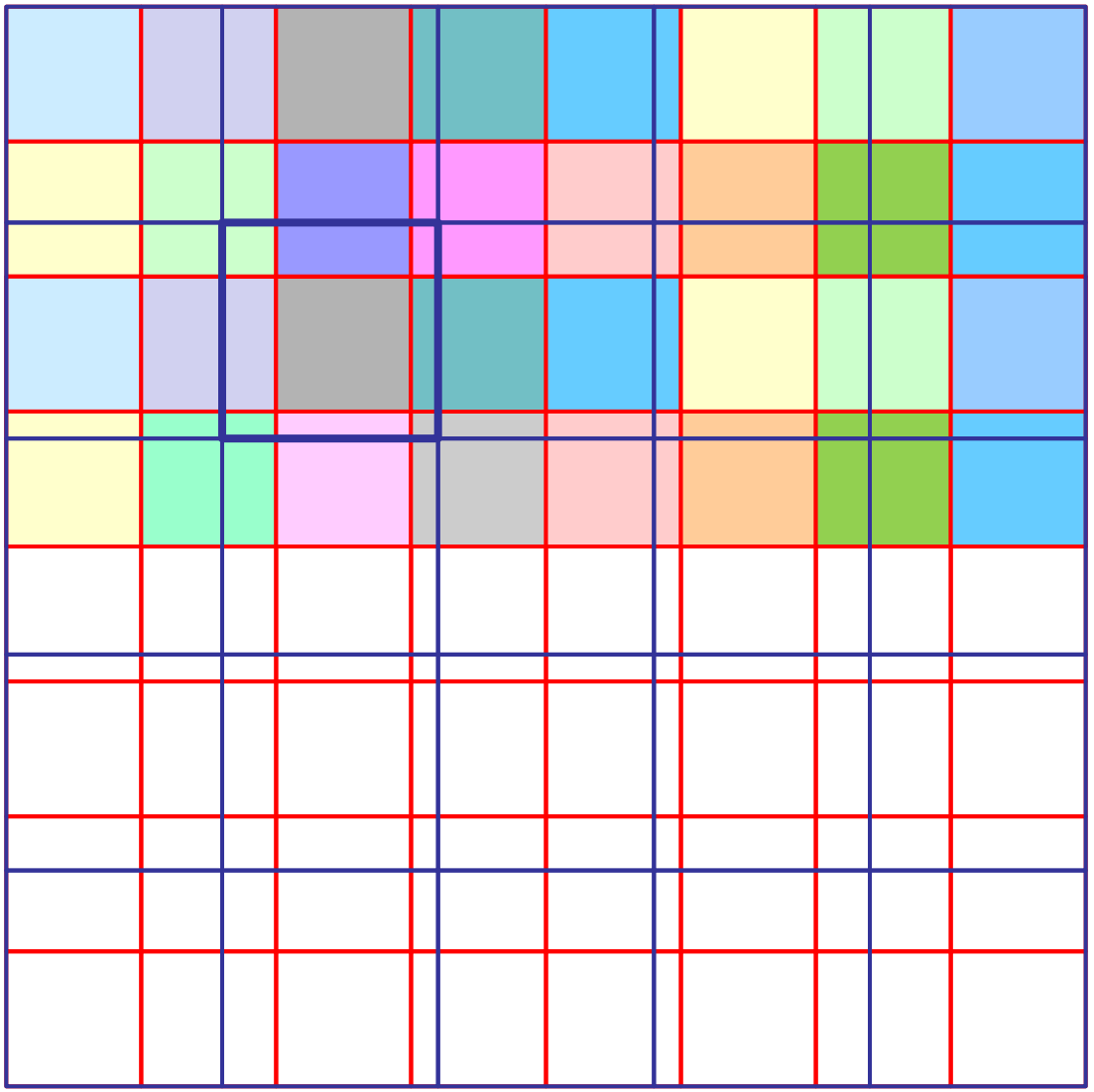


Алгоритм, основанный на площадях

Исходный размер 8x8

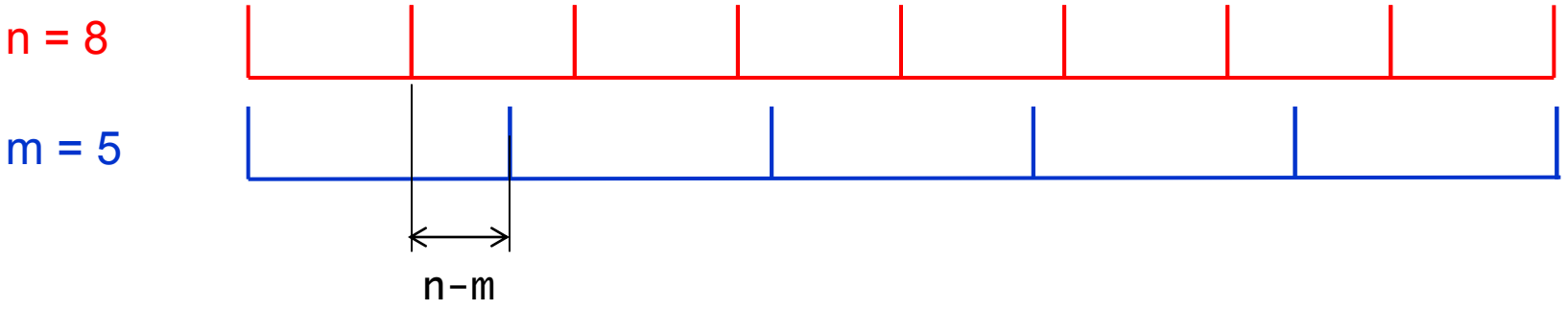
Новый размер 5x5

- Цвет пикселя – результат усреднения (интегрирования) по площади сенселя матрицы
- При увеличении размера пикселя его цвет получается усреднением (с учетом площадей) цветов перекрытых пикселей исходной картинки



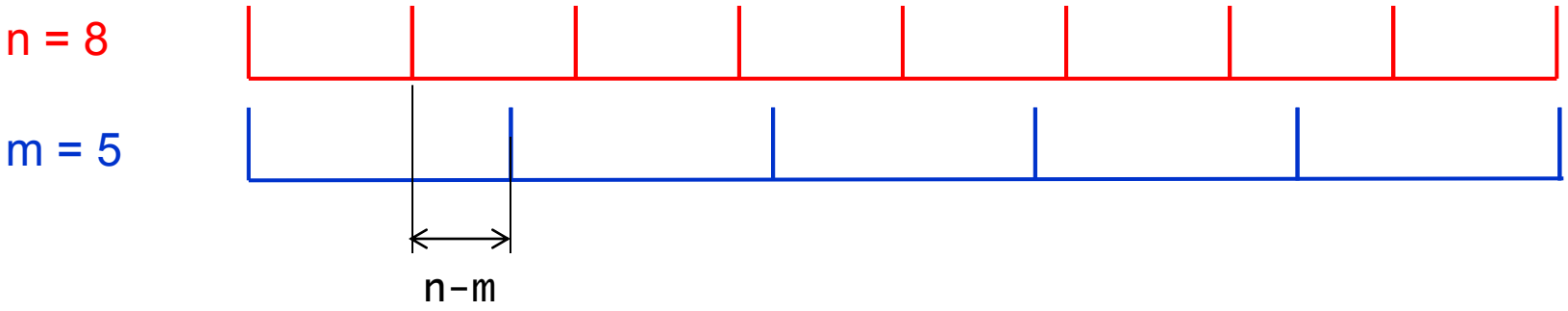
Реализация (C3C-алгоритм)

- Сепарабельность. Поканально (R, G, B).

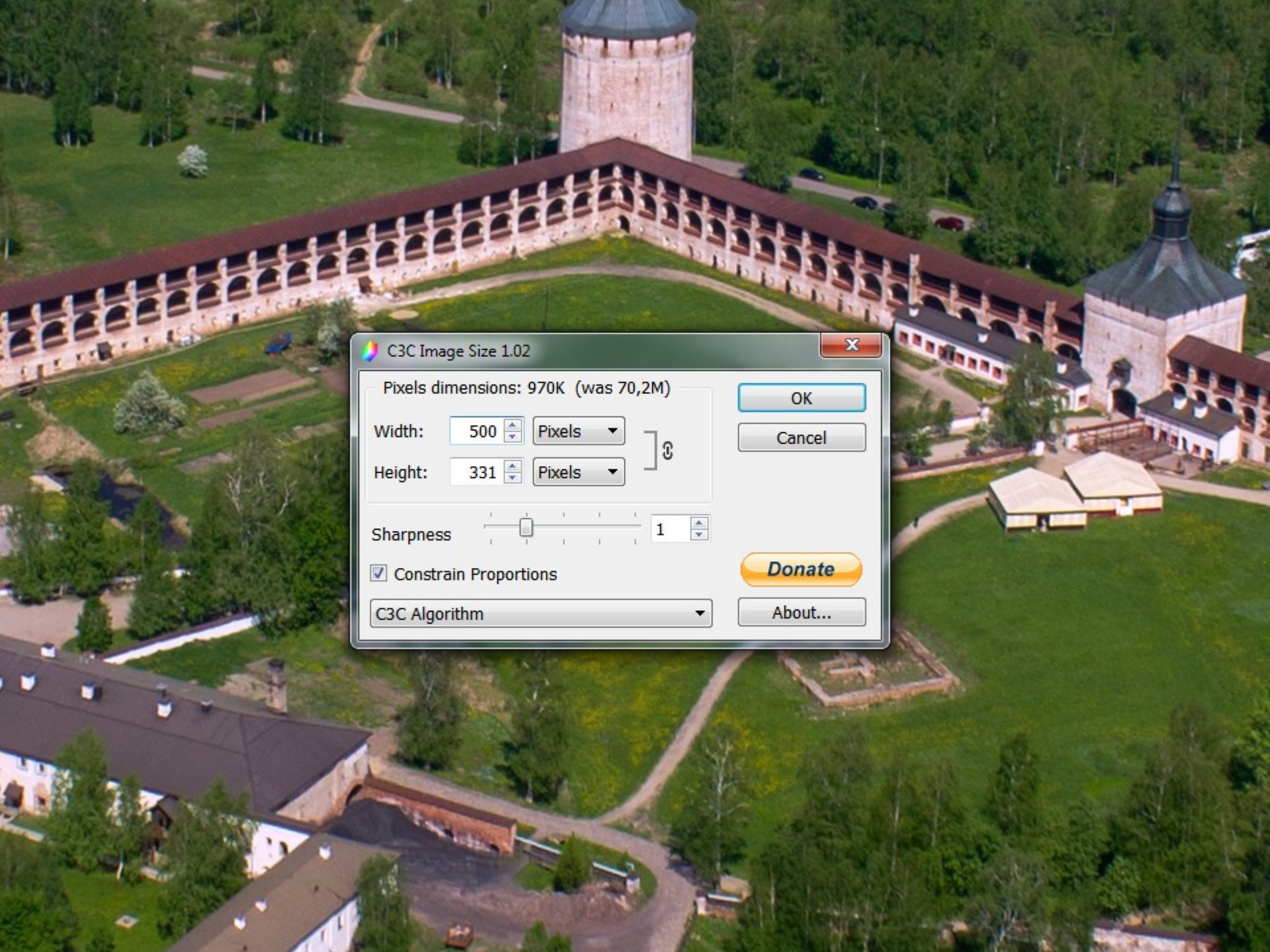


- Разобъем отрезки на n*m = 40 частей. Каждый «старый» пиксель содержит m частей, «новый» -- n частей.
- Делаем n шагов.
  - Цвет (значение R, G или B) каждого из n исходных пикселей или целиком «поглощается» одним из «новых» пикселей или разделяется между двумя соседними новыми.

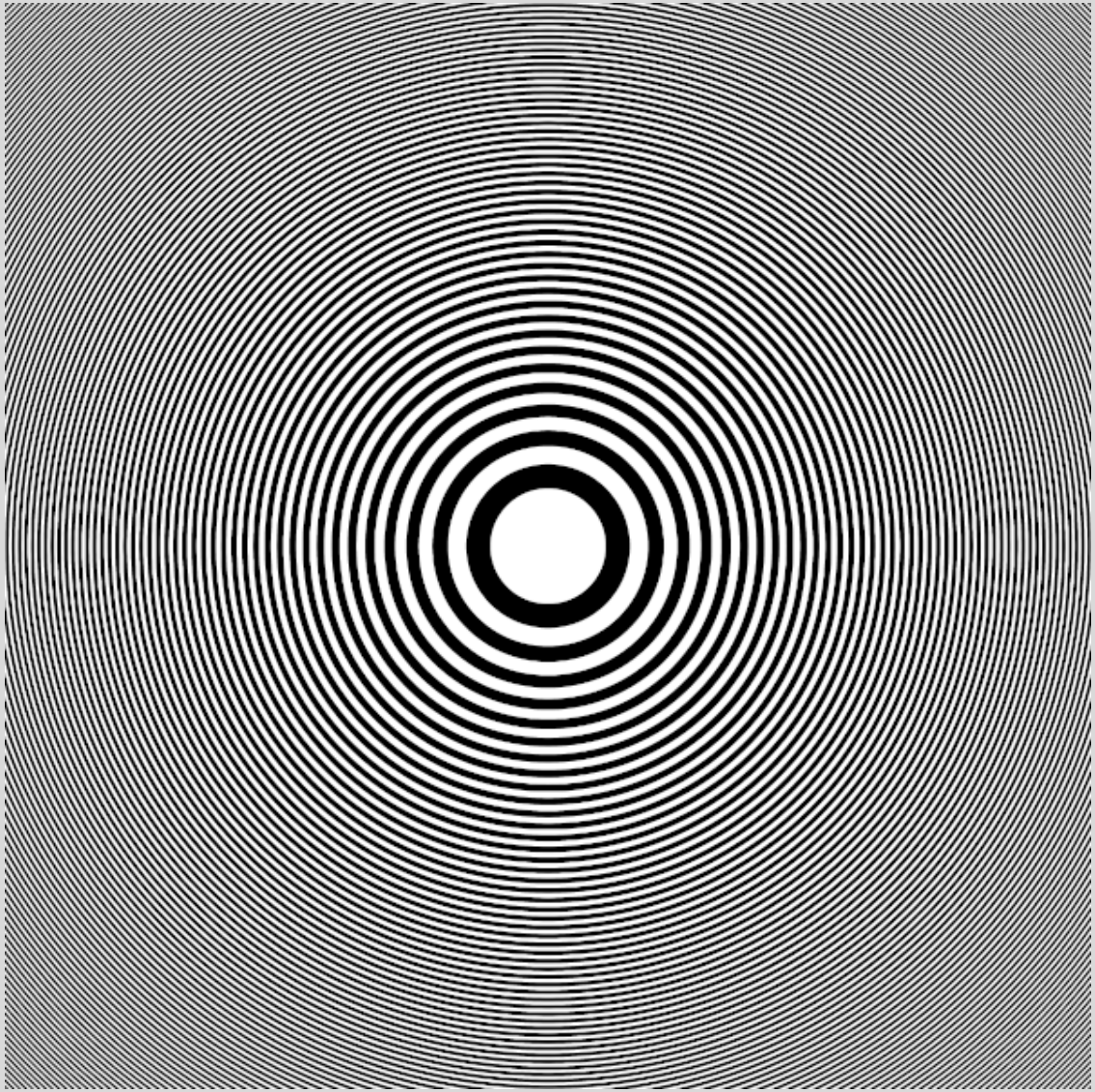
Реализация (C3C-алгоритм)

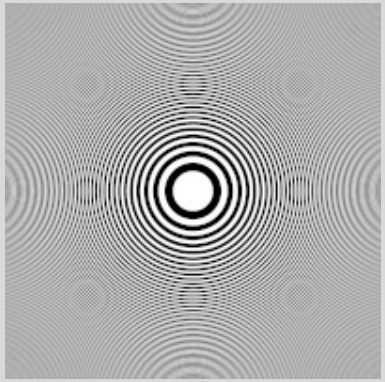


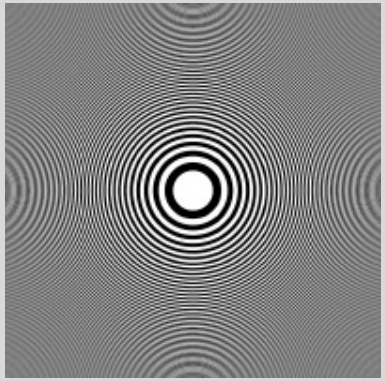

```
d := m-n;
k := 0;
for i := 0 to n-1 do begin
   if d < 0 then begin
      y[k] := y[k] + x[i]*m;
      d := d + m;
      end
   else begin
      y[k] := y[k] + (m-d)*x[i];
      k := k + 1;
      y[k] := y[k] + d*x[i];
      d := d + (m-n);
   end;
end;

for i := 0 to m-1 do
   y[i] := y[i]/n;
```

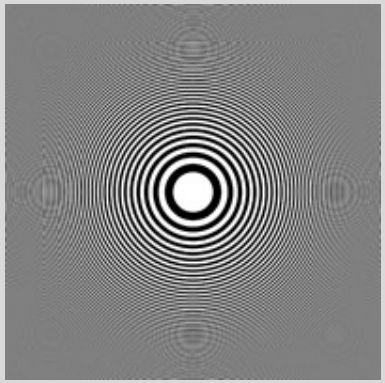

C3C Image Size 1.02
Pixels dimensions: 970K (was 70,2M)
Width:
500
Pixels
Height:
331
Pixels
OK
Cancel
Sharpness
1
Constrain Proportions
Donate
C3C Algorithm
About...

C3C-алгоритм









Резкость = 1

Резкость = 2

bicubic