**УДК 514.146.01**

***П. Г. Кононенко***

# АЛГОРИТМ КОМПЕНСАЦИИ ИСКАЖЕНИЙ ЦВЕТА ПРИ ПЕЧАТИ

***Аннотация.*** Предлагается вычислительный алгоритм компенсации искажений при печати, основанный на идее разбиения Делоне евклидова пространства.

***Ключевые слова:*** разбиение Делоне, точечная решетка.

***P. G. Kononenko***

# ALGORITHM FOR COMPENSATION OF COLOR DISTORTIONS DURING PRINTING

***Abstract.*** A computational algorithm is proposed to compensate for printing distortions based on the idea of dividing the Euclidean space by Delaunay.

***Key words:*** Delaunay partition, point lattice.

## 1. Введение

При создании художественных изображений на компьютере и их печати с помощью принтера на бумаге, на ткани или другом материале часто возникает такая проблема. Художник задумал и ввел в графический файл один цвет, а в результате печати неожиданно получил совсем другой. Могут ли такие разделы математики, как дискретная геометрия и методы вычислений, помочь в решении этой проблемы? Оказывается, могут.

## 2. Цветовое пространство $RGB$

Везде далее мы старались вести изложение максимально простым языком, и потому любые рассуждения следует воспринимать как имеющие «первое приближение» и нуждающиеся в уточнении.

Если в темноте смешать свет трех лучей: красного, зеленого, синего (англ. Red, Green, Blue — сокр. $RGB$) с яркостями, заданными соответствующими числовыми величинами $(x_R, x_G, x_B)$, которые могут независимо изменяться в диапазоне целых чисел $[0..255]$, то можно получить любой цвет, различаемый человеческим глазом. Таким образом, любой цвет — это точка с тремя координатами

$$\boldsymbol{x} = (x_R, x_G, x_B),$$

расположенная в 3-мерном пространстве $\mathbb{R}^3$, снабженном прямоугольной декартовой системой координат с осями $Ox_Rx_Gx_B$. А если быть более точным, цветовые точки расположены в кубе

$$C = [0..255]^3 = \\ = \{\boldsymbol{x} = (x_R, x_G, x_B) \in \mathbb{Z}^3 \mid 0 \leqslant x_w \leqslant 255,\ w = R, G, B\}.$$

## 3. Эмпирическая функция печати $f_\pi\colon \mathbb{R}^3 \longrightarrow \mathbb{R}^3$

*3.1. Определение эмпирической (табличной) функции $f_\pi$.*

Предположим, что художник имеет в своем распоряжении принтер под именем «$\pi$» и пытается с ним работать. Он рисует на компьютере деталь изображения и закрашивает ее цветом $\boldsymbol{x} = (x_R, x_G, x_B) \in \mathbb{R}^3$. А после печати на принтере получает другой реальный цвет $\boldsymbol{y} = (y_R, y_G, y_B) \in \mathbb{R}^3$. (Значения чисел $(y_R, y_G, y_B)$ можно получить, если сканировать полученное изображение на «очень хорошем» сканере, искажением которого мы пренебрегаем.) Таким образом, мы получаем эмпирически заданную функцию $f_\pi$, определенную работой принтера «$\pi$»:

$$f_\pi\colon C \subseteq \mathbb{R}^3 \longrightarrow C \subseteq \mathbb{R}^3,$$
$$\boldsymbol{x} = (x_R, x_G, x_B) \longmapsto \boldsymbol{y} = (y_R, y_G, y_B) = f_\pi(\boldsymbol{x}).$$

Формулу этой эмпирической функции мы не знаем. Однако некоторые ее значения можем найти, проводя серию экспериментов с печатью, сканированием и заполняя следующую таблицу.

*Таблица 1*

| Цвет, заданный художником в графическом файле | Цвет, полученный после печати на принтере «$\pi$» и сканирования |
|---|---|
| $\boldsymbol{x}_{(1)} = (x_{R_{(1)}}, x_{G_{(1)}}, x_{B_{(1)}})$ | $\boldsymbol{y}_{(1)} = (y_{R_{(1)}}, y_{G_{(1)}}, y_{B_{(1)}})$ |
| $\boldsymbol{x}_{(2)} = (x_{R_{(2)}}, x_{G_{(2)}}, x_{B_{(2)}})$ | $\boldsymbol{y}_{(2)} = (y_{R_{(2)}}, y_{G_{(2)}}, y_{B_{(2)}})$ |
| ... | ... |

*3.2. Мера искажения цвета при печати. Метрика.*

Мера искажения цвета при печати может быть определена как расстояние между точкой $\boldsymbol{x} = (x_R, x_G, x_B)$, представляющей цвет, заложенный художником в графическом файле, и цветовой точкой $\boldsymbol{y} = (y_R, y_G, y_B)$, полученной после печати и сканирования. В качестве расстояния можно взять всем привычную Евклидову метрику, вычисляемую по формуле:

$$d(\boldsymbol{x}, \boldsymbol{y}) = \sqrt{(x_R - y_R)^2 + (x_G - y_G)^2 + (x_B - y_B)^2}.$$

Очевидно, что $d(\boldsymbol{x}, \boldsymbol{y}) \geqslant 0$. При полном отсутствии искажения $d(\boldsymbol{x}, \boldsymbol{y}) = 0$, и увеличивается при росте искажений.

*3.3. Цветовой портрет принтера.*

Предположим, что, экспериментируя с принтером «$\pi$», мы перебрали все доступные цвета $\boldsymbol{x}_{(i)} = (x_{R_{(i)}}, x_{G_{(i)}}, x_{B_{(i)}})$ $(i = 1, 2, \ldots, N)$ и заполнили таблицу 1 соответствующими значениями $\boldsymbol{y}_{(i)} = (y_{R_{(i)}}, y_{G_{(i)}}, y_{B_{(i)}})$. Это потребовало бы проведения $N = 256^3 = 16\,777\,216 \approx 17$ *млн* испытаний.

Дадим геометрическую иллюстрацию гипотетически полученному результату. Все точки $\boldsymbol{x} = (x_R, x_G, x_B)$, как и прежде, заполняют цветовой куб $C = [0..255]^3 \subseteq \mathbb{R}^3$. А соответствующие им точки $\boldsymbol{y} = (y_R, y_G, y_B) = f(\boldsymbol{x})$ заполняют некоторую криволинейную область $C_\pi = f_\pi(C) \subseteq \mathbb{R}^3$, также расположенную внутри цветового куба $C \subseteq \mathbb{R}^3$.

Фигуру $C_\pi$ будем называть *областью печати принтера* «$\pi$» или *цветовым портретом принтера* «$\pi$», поскольку любые цвета $\boldsymbol{y} = (y_R, y_G, y_B)$, лежащие внутри области $C_\pi$, могут быть напечатаны на принтере «$\pi$». А цвета, лежащие снаружи, напечатаны быть не могут.

*3.4. Задача компенсации искажения цвета при печати.*

Предположим, что художник хочет получить на бумаге цвет $\boldsymbol{y} = (y_R, y_G, y_B)$ из допустимой области $C_\pi$. И желает узнать, какой для этого цвет $\boldsymbol{x} = (x_R, x_G, x_B)$ ему следует разместить в графическом файле. Он мог бы найти соответствующий $\boldsymbol{y} = (y_R, y_G, y_B)$ в правой колонке нашей гипотетической таблицы 1, содержащей около 17 млн строк, и взять соответствующий ему $\boldsymbol{x} = (x_R, x_G, x_B)$ в левой части этой строки. То есть воспользоваться эмпирически (или таблично) заданной обратной функцией

$$f_\pi^{-1}\colon (y_R, y_G, y_B) = \boldsymbol{y} \longmapsto f_\pi^{-1}(\boldsymbol{y}) = \boldsymbol{x} = (x_R, x_G, x_B).$$

Какие же проблемы мы видим при практическом использовании этого метода?

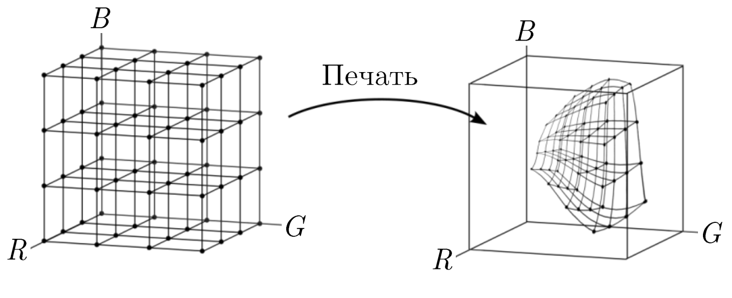
а) Хранение в компьютере таблицы из 17 млн строк на сегодняшний день не так уж обременительно (потребуется всего около 98 MB).

б) Проведение 17 млн испытаний для домашнего принтера (да и для производственного) было бы слишком дорогим.

в) Поиск в неупорядоченной таблице, содержащей 17 млн строк, при массовом использовании метода был бы слишком долгим.

## 4. Метод компенсации искажения цвета при печати

Следует составить сокращенную таблицу для меньшего количества точек $\boldsymbol{x}_{(i)} = (x_{R_{(i)}}, x_{G_{(i)}}, x_{B_{(i)}})$, *равномерно* распределенных по кубу $C = [0..255]^3$. Например, можно взять их в узлах кубической решетки. Назовем эту систему $L_X$. Соответствующие им точки $\boldsymbol{y}_{(i)} = (y_{R_{(i)}}, y_{G_{(i)}}, y_{B_{(i)}})$ образуют систему, которую назовем $L_Y$. В силу искажения при печати она выглядит как деформированная (согнутая и смятая) кубическая решетка. Если натянуть на каждую «клеточку» эластичную пленку, внутри получится выпуклая оболочка $conv\{L_Y\}$ множества $L_Y$. Эта оболочка представляет собой многогранник, близкий к области печати $C_\pi$ принтера «$\pi$».




Мы построим приближенную обратную функцию

$$\widehat{f_\pi^{-1}(\boldsymbol{y})}\colon conv\{L_X\} = C \longleftarrow conv\{L_Y\}$$

как линейную интерполяцию функции $f_\pi^{-1}$, заданную сокращенной таблицей на этих избранных точках $f_\pi^{-1} : \boldsymbol{y}_{(i)} \longmapsto \boldsymbol{x}_{(i)}$.

Теперь подробнее. Если художник хочет получить на бумаге (или ткани) цвет $\boldsymbol{y}_0 = (y_{R_0}, y_{G_0}, y_{B_0})$, то среди точек $\boldsymbol{y}_{(i)} \in L_Y$ найдем четыре точки $\boldsymbol{y}_{(0)}$, $\boldsymbol{y}_{(1)}$, $\boldsymbol{y}_{(2)}$, $\boldsymbol{y}_{(3)}$, наиболее близкие к точке $\boldsymbol{y}_0$, которые образуют тетраэдр (3-мерный симплекс), содержащий внутри эту точку. И пусть $\boldsymbol{x}_{(0)}$, $\boldsymbol{x}_{(1)}$, $\boldsymbol{x}_{(2)}$, $\boldsymbol{x}_{(3)}$ — соответствующие им точки сокращенной таблицы из множества $L_X$.

Составим барицентрическое разложение точки $\boldsymbol{y}_0$ по заданным точкам $\boldsymbol{y}_{(0)}$, $\boldsymbol{y}_{(1)}$, $\boldsymbol{y}_{(2)}$, $\boldsymbol{y}_{(3)}$:

$$\begin{gathered} \boldsymbol{y}_0 = \lambda_0 \cdot \boldsymbol{y}_{(0)} + \lambda_1 \cdot \boldsymbol{y}_{(1)} + \lambda_2 \cdot \boldsymbol{y}_{(2)} + \lambda_3 \cdot \boldsymbol{y}_{(3)}, \\ \lambda_0 + \lambda_1 + \lambda_2 + \lambda_3 = 1, \quad \lambda_0, \lambda_1, \lambda_2, \lambda_3 \geqslant 0 \end{gathered} \tag{1}$$

(такие числа $\lambda_0$, $\lambda_1$, $\lambda_2$, $\lambda_3$ существуют и вычисляются однозначно по известным формулам, см. [1, 3]).

Осталось лишь еще раз написать разложение (1), сохранив значения коэффициентов $\lambda_0$, $\lambda_1$, $\lambda_2$, $\lambda_3$ и заменив точки $\boldsymbol{y}_{(i)}$ на $\boldsymbol{x}_{(i)}$. И мы получим ответ к нашей задаче. Это цветовая точка $\widehat{\boldsymbol{x}}_0$, которую художник должен разместить в графическом файле, чтобы при печати на принтере «$\pi$» получить желаемый цвет $\boldsymbol{y}_0$ (с точностью до погрешности линейной интерполяции). Итак,

$$\widehat{\boldsymbol{x}}_0 = \lambda_0 \cdot \boldsymbol{x}_{(0)} + \lambda_1 \cdot \boldsymbol{x}_{(1)} + \lambda_2 \cdot \boldsymbol{x}_{(2)} + \lambda_3 \cdot \boldsymbol{x}_{(3)}. \tag{2}$$

Остается нерешенной лишь одна, как ни странно, вычислительно сложная задача: как выбрать подходящие точки $\boldsymbol{y}_{(0)}$, $\boldsymbol{y}_{(1)}$, $\boldsymbol{y}_{(2)}$, $\boldsymbol{y}_{(3)}$ в системе $L_Y$. Ее решение мы найдем в идее построения разбиения Делоне пространства $\mathbb{R}^3$, заданного конечной системой точек $L_Y \subseteq \mathbb{R}^3$.

## 5. Разбиение Делоне

Точки $\boldsymbol{y}_{(0)}$, $\boldsymbol{y}_{(1)}$, $\boldsymbol{y}_{(2)}$, $\boldsymbol{y}_{(3)}$ из конечной системы $L_Y \subseteq \mathbb{R}^3$ являются вершинами 3-мерного симплекса (тетраэдра) Делоне, если существует шар, который: а) содержит их на своей поверхности, б) не содержит внутри ни одной точки системы $L_Y$. В общем случае говорят о многогранниках Делоне, которые могут содержать более четырех вершин. Однако в нашем случае точки $\boldsymbol{y}_{(i)} \in L_Y$ случайны и попадание более четырех из них на одну сферу маловероятно. Поэтому мы будем считать все многогранники Делоне тетраэдрами.

Выпуклая оболочка множества точек $L_Y$ оказывается целиком заполнена такими тетраэдрами без пересечений. Соседние тетраэдры могут иметь лишь общую вершину, ребро или 2-мерную грань. Это и есть разбиение Делоне области $conv\{L_Y\}$.

Однако на практике вычислять и хранить в компьютере целиком все разбиение Делоне, заданное системой точек $L_Y$, было бы долго и неэффективно. Нам нужен лишь один тетраэдр, содержащий точку $\boldsymbol{y}_0$ (цвет, задуманный художником). Идею решения мы найдем в статье Б. Н. Делоне [2] за 1937 г., в которой он предлагает так называемый «метод пустого шара» для построения разбиения, впервые введенного им в этой же статье.

## 6. Поиск тетраэдра Делоне, содержащего точку $\boldsymbol{y}_0$

Начнем с цитаты из статьи [2].

«Рассмотрим шар, — увеличивающийся, уменьшающийся и как угодно передвигающийся между точками системы $L_Y$ [авторские обозначения изменены], — подчиненный лишь одному условию: не содержать внутри себя точек этой системы. Мы будем называть такой шар пустым. Начнем увеличивать радиус пустого шара, оставляя его центр на месте, пока шар не наткнется своей поверхностью на какую-нибудь точку системы $L_Y$. <...> Будем теперь дальше увеличивать радиус пустого шара, отодвигая его центр от этой точки или, если точек, на которые он наткнулся, было сразу несколько, — от того линейного подпространства, которое определяется этими точками. Продолжая так дальше, мы убедимся, что в системе $L_Y$ существует пустой шар, на поверхности которого лежит $n$-мерный [здесь 3-мерный] комплекс точек этой системы <...>, содержащий не менее $n+1$ точки [здесь четырех точек].»

Начав с точки $\boldsymbol{y}_0$ как центра пустого шара, мы получим первичный тетраэдр Делоне $S_{(0)}$, одна из вершин которого является ближайшей к $\boldsymbol{y}_0$ точкой системы $L_Y$.

Теперь вычислим коэффициенты $\lambda_0$, $\lambda_1$, $\lambda_2$, $\lambda_3$ барицентрического разложения (1) относительно вершин $S_{(0)}$. Если все они неотрицательны, то точка $\boldsymbol{y}_0$ уже лежит внутри тетраэдра $S_{(0)}$. Значит, мы получили желаемое, и процесс завершен.

В противном случае возьмем наименьшее $\lambda_i$ из отрицательных значений $\lambda_0$, $\lambda_1$, $\lambda_2$, $\lambda_3$. Значит, точка $\boldsymbol{y}_0$ лежит с внешней стороны от грани $F_i$ тетраэдра $S_{(0)}$, полученной выкидыванием $i$-й вершины $\boldsymbol{y}_{(i)}$. И притом наиболее удалена от плоскости грани по сравнению с другими гранями. Аналогично предыдущему будем смещать центр пустого шара в сторону внешней нормали грани $F_i$, опираясь на ее вершины, пока не наткнемся на вершину соседнего тетраэдра $S_{(1)}$.

Будем продолжать таким образом, переходя от одного тетраэдра Делоне к другому, приближаясь к точке $\boldsymbol{y}_0$, пока не «поймаем» точку $\boldsymbol{y}_0$ внутри очередного тетраэдра $S_{(k)}$. Либо не придем к такой ситуации, когда во внешнем полупространстве очередной грани $F_i$ не найдется ни одной точки системы $L_Y$.

Последний исход означает, что точка $\boldsymbol{y}_0$ лежит вне выпуклой оболочки $conv\{L_Y\} \approx C_\pi$ множества $L_Y$. То есть художник задумал цвет $\boldsymbol{y}_0$, лежащий вне области печати принтера «$\pi$», и он не может быть напечатан. В этом случае в качестве утешения мы можем предложить ему цвет $\widetilde{\boldsymbol{x}}_0$, полученный по формуле (2), в которой все отрицательные коэффициенты $\lambda_0$, $\lambda_1$, $\lambda_2$, $\lambda_3$ заменены нулями. То есть цветовую точку $\widetilde{\boldsymbol{x}}_0$, ближайшую желаемому цвету $\boldsymbol{y}_0$ из области печати $C_\pi$ принтера «$\pi$». Такой результат следует снабдить величиной предполагаемого искажения цвета $d = d(\widetilde{\boldsymbol{x}}_0, \boldsymbol{y}_0)$, которая выходит за рамки погрешности линейной интерполяции.

### *Библиографический список*


1. *Балк М. Б., Болтянский В. Г.* Геометрия масс. М.: Наука, 1987. 160 с. (Библиотечка «Квант». Вып. 61).
2. *Делоне Б. Н.* Геометрия положительных квадратичных форм // Успехи математических наук. 1937. Вып. 3. С. 16–62.
3. *Кострикин А. И., Манин Ю. И.* Линейная алгебра и геометрия: Учеб. пособ. для вузов. 2-е изд., перераб. М.: Наука, 1986. 304 с.


### *Информация об авторе / Information about the author*


***Кононенко Павел Геннадьевич*** – кандидат физико-математических наук, доцент кафедры фундаментальной математики, Ивановский государственный университет, г. Иваново, Россия, konon-pg@yandex.ru

***Kononenko Pavel Gennadievich*** – Candidate of Physical and Mathematical Sciences, Associate Professor of the Department of Fundamental Mathematics, Ivanovo State University, Ivanovo, Russia, konon-pg@yandex.ru