Гусев Н.В.

Бесшовное наложение изображений с помощью преобразований Пуассона

Воронежский государственный университет инженерных технологий
(Россия, Воронеж)

doi: 10.18411/lj-01-2019-118

idsp: ljjournal-01-2019-118

Аннотация

Рассматривается задача бесшовного наложения изображений друг на друга, так, чтобы не было видно границ между ними. Эта задача может быть решена с помощью преобразований Пуассона.

Ключевые слова: блендинг, преобразование, Пуассон, изображения, наложение, автоматизация

Abstract

This article deals with the problem of blending images on each other, so that the borders between them are not visible. This task can be sorted out using Poisson transformations.

Keywords: blending, transformation, Poisson, picture, overlay, automation

Во многих задачах автоматизированной обработки фотографий и изображений довольно часто встречается задача бесшовного наложения изображений или по-другому это называется блендингом.

Бесшовное наложение изображений называется такой метод, который допускает вставлять в одно изображение часть другого изображения, при этом, чтобы на границах вставки изображений не были заметны швы. Создается впечатление, что этот метод подстраивает и оптимизирует ту часть изображения, которую хотим вставить специально под то изображение, в которое мы хотим вставить часть без видимости швов. На самом деле, чтобы получить такой результат применяется только градиентное вычисление полей для вставляемой части и значения цвета и яркости данного пикселя исходного изображения на границе области, которая обрабатывается.

Прежде, чем рассказать про то, каким образом это происходит, стоит сделать постановку задачи. У нас имеются два изображения, назовем их изображение 1 и изображение 2. На изображение 1 накладывается изображение 2 в специальном месте.

Пример наложения изображений друг на друга

Для того, чтобы наложить изображение 2 на изображение 1 с использованием бесшовного наложения, требуется, чтобы градиенты изображения 1 равнялись градиенту изображения 2 в области их соединения. Восстановление цветовой палитры на области, где расположено изображение 2 будет происходить на основе градиента в этой области и цветовой палитры изображения 1 на границе.

Таким образом, это означает, что требуется для получения итогового изображения восстановить изображение по градиенту, то есть можно применить задачу Пуассона.

Решение задачи для восстановления градиента изображения приведено ниже. Для изображения с цветовой палитрой RGB метод используется к каждому каналу.

Следующим образом обозначим границу вставляемой области:

$$H_{(x,y)} = A_{(x,y)} \forall (x, y) \in \partial B$$

Вычисление градиента изображения 2:

$$|\nabla B_{(x,y)}| = 4B(x, y) - B(x - 1, y) - B(x + 1, y) - B(x, y - 1) - B(x, y + 1)$$

Применяя эти данные можно составить уравнение Пуассона:

$$\begin{aligned} & |N|H(x, y) - \sum_{(dx, dy) + (x, y) \in \Omega} H(x + dx, y + dy) \\ & - \sum_{(dx, dy) + (x, y) \in \partial \Omega} A(x + dx, y + dy) \\ & = \sum_{(dx, dy) + (x, y) \in (\Omega \cup \partial \Omega)} (B(x + dx, y + dy) - B(x, y)) \end{aligned}$$

где (x, y) – координаты данного пикселя, N – число пикселей, которые находятся рядом с текущим пикселем (не более 4), (dx, dy) – координаты пикселей, которые находятся рядом с текущим, могут принимать значения $\{(-1,0), (1,0), (0,-1), (0,1)\}$.

Для каждого нового пикселя составляется уравнение. Таким образом мы получаем систему с неизвестным числом пикселей вставляемого изображения.

Для примера возьмем два изображения. Изображение, из которого мы возьмем элемент слева и изображение, в которое вставим этот элемент расположено справа.

Исходные изображения

После вставки элемента изображения слева в изображение справа и настройки масштаба и угла наклона мы применяем преобразование Пуассона.

Изображения после применения преобразований Пуассона

Таким образом, с помощью преобразований Пуассона можно накладывать изображения друг на друга без видимости швов соединения.

1. Применение преобразований Пуассона для бесшовного наложения изображений [Электронный ресурс]: Хабр — Режим доступа: URL: <https://habr.com/post/213403> (12.01.2019).
 2. Kenny A. Hunt. The Art of Image Processing with Java: A.K. Peters. Ltd. Natick, Massachusetts, 2010. 300.
 3. Patrick Pérez, Michel Gangnet, Andrew Blake. Poisson Image Editing. ACM Transactions on Graphics (TOG) - Proceedings of ACM SIGGRAPH 2003: Volume 22 Issue 3, p 313-318.
-