

Лабораторная работа

23 сентября 2016 г.

Преобразование Хафа для детектирования линий

Теоретическая справка

В основе теории преобразования Хафа лежит утверждение, что любая точка двоичного изображения может быть частью некоторого набора возможных линий.

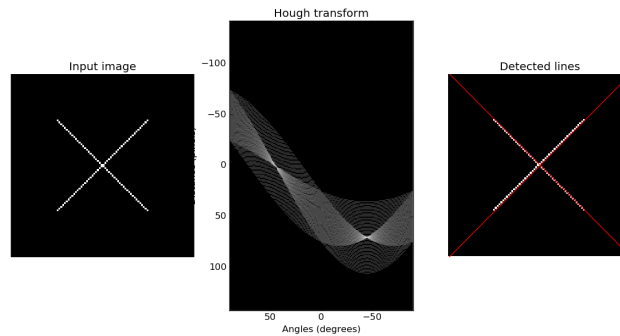


Рис. 1: Исходное изображение, соответствующее ему фазовое пространство и найденные линии

Линия задается уравнением в полярных координатах: $x \cos(f) + y \sin(f) = R$, где R — длина перпендикуляра, опущенного на линию из начала координат, а f — угол между перпендикуляром и осью абсцисс, находится в пределах $[0, 2\pi]$. Через каждую точку (x, y) изображения можно провести несколько прямых с разными R и f , то есть каждой точке (x, y) изображения соответствует набор точек в фазовом пространстве (R, f) , образующий синусоиду. Также каждой точке (R_0, f_0) пространства (R, f) можно поставить в соответствие счетчик, соответствующий количеству точек (x, y) , лежащих на прямой $x \cos(f_0) + y \sin(f_0) = R_0$. Непрерывное фазовое пространство переводится в дискретное вводом сетки на пространстве (R, f) , одной ячейке которой соответствует набор прямых с близкими значениями R и f .

Задание

Вам необходимо реализовать алгоритм поиска линий на изображении.

1. Выделите края изображения с помощью фильтра Canny или фильтра Sobel.
2. Создайте отдельную функцию `houghlines`, принимающую на вход изображение, шаг сетки, точность попадания точки на прямую и возвращающую матрицу фазового пространства Хафа. В рамках функции создайте матрицу для хранения фазового пространства Хафа. Для каждого контурного пикселя исходного изображения необходимо рассмотреть все возможные прямые, которые могут проходить через эту точку (угол наклона f и расстояние от начала координат R). Если $\left| (y \sin(f) + x \cos(f)) - r \right| < accuracy$, увеличиваем счетчик для этой точки фазового пространства.
3. Визуализируйте полученное фазовое пространство.
4. Преобразуйте точки фазового пространства с максимальным откликом в линии.
5. Визуализируйте линии на исходном изображении.

6. Подберите оптимальные параметры для тестовых изображений «line1.png», «line2.png», «line3.png».
7. Сравните работу написанного алгоритма с функцией `cvHoughLines2` из библиотеки OpenCV или `hough_line` из библиотеки `scikits_image`.

Преобразование Хафа для детектирования окружностей

Задание

Вам необходимо реализовать алгоритм окружностей на изображении.

1. Выделите края изображения с помощью фильтра Canny или фильтра Sobel.
2. Выпишите параметрическое уравнение окружности. Сколько параметров в него входит?
3. Создайте отдельную функцию `houghcircles`, по аналогии с `houghlines`.
4. Визуализируйте полученное фазовое пространство.
5. Преобразуйте точки фазового пространства с максимальным откликом в окружности.
6. Визуализируйте окружности на исходном изображении.
7. Подберите оптимальные параметры для тестовых изображений «circle1.png», «circle2.png», «circle3.png».
8. Сравните работу написанного алгоритма с функцией `hough_circle` из библиотеки `scikits_image`.