

МЕТОДИКА ПРОГРАММНОЙ ИДЕНТИФИКАЦИИ ПАРАМЕТРОВ НАБЛЮДАЕМОГО ОБЪЕКТА

Пархоменко И. Н., Господ А. В.

Кафедра автоматизации технологических процессов и производств

Могилевский государственный университет продовольствия

Могилев, Республика Беларусь

E-mail: gospod_andrei@mail.ru

Предложена новая методика программного определения площади исследуемого объекта, с помощью одной камеры, используя трансформацию цвета (функция I), пороговое преобразование (функция II) и определение контуров (функция III).

Необходимость отказа от ручного труда формируется современной необходимостью и одним из компонентов автоматизации является машинное зрение, позволяющее идентифицировать объекты [1]. Применение технического зрения в различных областях: в медицине - для диагностики заболеваний, в промышленности - автоматизация дает уменьшение себестоимости, в автомобилестроении - навигация беспилотников [2]. И одним из параметров наблюдаемых объектов, который необходимо идентифицировать является его площадь [3], для чего разработана методика её определения с помощью веб камеры и трех функций.

Для преобразования входного изображения из одного цветового пространства в другое используется функция I. Которая на входе получает 8-битное изображение без знака, 16-битное без знака или с плавающей точкой одинарной точности, код преобразования цветового пространства и количество каналов в целевом изображении, в результате имеем изображение в серых тонах, которое передается в функцию II.

Функция II применяет пороговое преобразование, для каждого пикселя применяется одинаковое пороговое значение. Если значение пикселя меньше порогового значения, оно устанавливается равным 0, в противном случае устанавливается максимальное значение. Первым аргументом является исходное изображение, которое должно быть изображением в градациях серого. Вторым аргумент - это пороговое значение, которое используется для классификации значений пикселей. Третьим аргумент - это максимальное значение, которое присваивается значениям пикселей, превышающим пороговое значение. На выходе имеем два параметра, первый - это порог, который был использован, а второй - пороговое изображение (см. рис. 1). Таким образом каждый пиксель, обнаруженный на изображении, превышающий пороговое значение содержания цвета соответствует единице в массиве под тем же порядковым номером, что и пиксель на изображении. Это необходимо для того, чтобы создать «очищенное» от лишних цветов 8-битное изображение, на котором будет находиться черный

фон и сам объект, который нам нужен и будет представлен собственным контуром в белом заполнении.

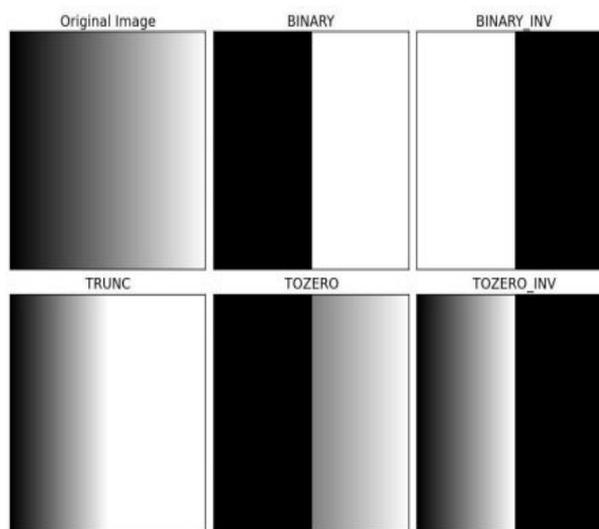


Рис. 1 – Пороговое преобразование

Но в глобальном пороговом значении использованы произвольно выбранное значение в качестве порога. Поэтому чтобы избежать необходимости выбирать значение, а определять его автоматически, рассмотрим изображение только с двумя различными значениями изображения (бимодальное изображение), где гистограмма будет состоять только из двух пиков. Хороший порог будет в середине этих двух значений. Точно так же определяется оптимальное глобальное пороговое значение из гистограммы изображения. Пороговое значение может быть выбрано произвольно. Затем алгоритм находит оптимальное пороговое значение, которое возвращается в качестве первого выхода. Входное изображение имеет шумы (см. рис.2). В первом случае применяется глобальное пороговое значение со значением 127. Во втором случае, пороговое значение применяется напрямую. В третьем случае изображение сначала фильтруется гауссовым ядром 5x5 для удаления шума, затем применяется пороговое значение. Поскольку идет работа с бимодальными изображениями, алгоритм пытается найти пороговое значение (t), которое мини-

мизирует взвешенную дисперсию внутри класса, определяемую соотношением:

$$\sigma_{\omega}^2(t) = q_1(t)\sigma_1^2(t) + q_2(t)\sigma_2^2(t)$$

тогда

$$q_1(t) = \sum_{i=1}^t P(i) \ \& \ q_2(t) = \sum_{i=t+1}^I P(i)$$

$$\mu_1(t) = \sum_{i=1}^t \frac{iP(i)}{q_1(t)} \ \& \ \mu_2(t) = \sum_{i=t+1}^I \frac{iP(i)}{q_2(t)}$$

$$\sigma_1^2(t) = \sum_{i=1}^t [i - \mu_1(t)]^2 \frac{P(i)}{q_1(t)} \ \& \ \sigma_2^2(t) = \sum_{i=t+1}^I [i - \mu_2(t)]^2 \frac{P(i)}{q_2(t)}$$

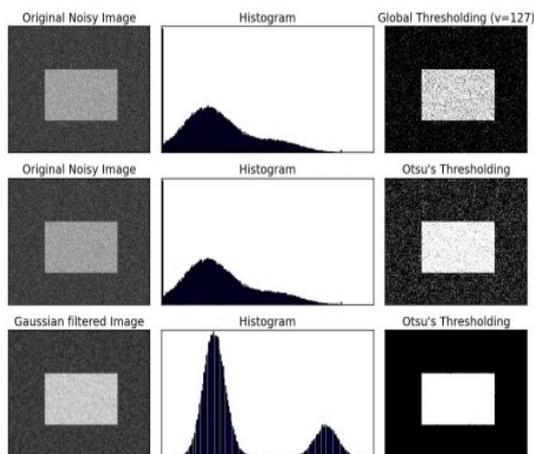


Рис. 2 – Фильтрация шума

Для определения контура не черного объекта используется функция Π , которая возвращает массив векторов точек, составляющий собой объект с изображения. На входе функция Π получает 8-битное одноканальное изображение, где ненулевые пиксели рассматриваются как 1, нулевые пиксели остаются 0, поэтому изображение рассматривается как двоичное. На выходе - обнаруженные контуры, которые сохраняются как векторы точек. Также получим выходной вектор, содержащий информацию о топологии изображения. Он имеет столько же элементов, сколько число контуров. Для каждого контура i -го контура $[i]$ иерархия элементов $[i] [0]$, $hiearchy [i] [1]$, $hiearchy [i] [2]$ и $hiearchy [i] [3]$ установлены в 0 - основные индексы в контурах следующего и предыдущего контуров на том же иерархическом уровне, первый дочерний контур и родительский контур, соответственно. Если для контура i нет следующих, предыдущих, родительских или вложенных контуров, соответствующие элементы иерархии $[i]$ будут отрицательными. Изображение рассматривается как массив

$$z(z_1, \dots, z_n, \dots, z_N)$$

где z – значения интенсивности пикселей, n – общее число пикселей. Для отделения объекта от

фона алгоритм определяет значения элементов массива прозрачности

$$a(a_1, \dots, a_n, \dots, a_N)$$

причем a_n может принимать два значения, если $a_n = 0$, значит пиксель принадлежит фону, если $a_n = 1$ то объекту. Внутренний параметр Θ содержит гистограмму распределения интенсивности переднего плана и гистограмму фона:

$$\Theta = \{h(z; a), a = 0, 1\}$$

Задача сегментации - нахождение неизвестных a_n . Рассматривается функция энергии:

$$E(a, \Theta, z) = U(a, \Theta, z) + V(a, z)$$

Причем минимум энергии соответствует наилучшей сегментации:

$$U(a, \Theta, z) = - \sum_n \log h(z_n, a_n)$$

$$V(a, z) = \sum_{(m,n)} \frac{1}{dis(m,n)} [a_n \neq a_m] \exp(-\beta(z_m - z_n)^2)$$

где $V(a, z)$ – слагаемое отвечает за связь между пикселями, сумма идет по всем парам пикселей, которые являются соседями, $dis(m, n)$ – евклидово расстояние, $a_n \neq a_m$ отвечает за участие пар пикселей в сумме, если $a_n = a_m$, то эта пара не будет учитываться, $U(a, \Theta, z)$ - отвечает за качество сегментации, т.е. разделение объекта от фона. Найдя глобальный минимум функции энергии E , получим массив прозрачности:

$$\hat{a} = argmin_a E(a, \Theta)$$

Для минимизации функции энергии, изображение описывается как граф и ищется минимальный разрез графа.

Таким образом разработано программное обеспечение определяющее площадь наблюдаемого объекта, что в перспективе позволит также находить расстояние до наблюдаемого объекта с использованием всего лишь одной камеры, т.е. без использования стерео пары.

СПИСОК ЛИТЕРАТУРЫ

1. Escriva, D. M., Joshi, P., Mendonca, V., Shilkrot, R. Building Computer Vision Projects with OpenCV 4 and C++ / D. M. Escriva, P. Joshi; V. Mendonca; R. Shilkrot // Packt Publishing. – Birmingham. – 2019. – p. 538.
2. Kaehler, A., Bradski, G. Learning opencv / A. Kaehler, G. Bradsk // O'Reilly Media. – Gravenstein Highway North. – 2017. – p. 1024.
3. Datta, S. Learning OpenCV 3 Application Development / S. Datta // Packt Publishing. – Birmingham. – 2016. – p. 310.