



Литература

1. СТО СГАУ 02068410-009-2007. Обработка и оформление результатов измерений [Текст]. - Взамен СТП КуАИ 144-5-88; введ. 2007-11-01. - Самара : СГАУ, 2007. - IV; 33 с. - (Комплексная система управления качеством деятельности вуза).
2. Осипович, И. Р. Интерферометрический метод контроля формы асферических поверхностей качения прецизионных подшипников [Текст] / И. Р. Осипович, Д. Т. Пуряев // Вестник Московского государственного технического университета. Сер. Приборостроение. 1999. Вып. 3. С. 65 – 75, 128.
3. Заякин, О. А. Получение профилей и контурных картин поверхностей вращения способом триангуляции с использованием зеркально отраженного излучения. [Текст] / О.А.Заякин // Вестн. Сам. гос. техн. ун-та. Сер.: Физ.-мат. науки. - 2007. - №2 (15). - С. 95 - 101.
4. Методика измерения отклонений от круглости и шероховатости поверхности деталей подшипников качения [Текст]. Руководящий документ РД 37.006.106.90 Согласован. - М.: НПО ВНИПП, 1990. - 40 листов.
5. ГОСТ 17353-89. Приборы для измерений отклонений формы и расположения поверхностей вращения. Типы. Общие технические требования [Текст]. - Введ. 1991 01 01; ИУС2-96, утв. в декабре 1995 г. - М.: ИПК Изд-во стандартов, 1999. - 9 с.
6. Белопухов, В. Н. Комплекс программных средств для исследования погрешности измерений экспериментального лазерного кругломера [Текст] / В. Н. Белопухов, О. А. Заякин, А. В. Манухин, А. А. Ростов // Программные продукты и системы. - 2018. - Т. 31, №1. - С. 561 - 566.

И.А. Рыбакова , В.Ю. Лабыскин

ИССЛЕДОВАНИЕ МЕТОДОВ И РАЗРАБОТКА АЛГОРИТМА АНАЛИЗА И РАСПОЗНАВАНИЯ ИЗОБРАЖЕНИЯ

(Самарский государственный технический университет)

Введение

В современном мире активно ведется автоматизация различных видов деятельности. Для выполнения ряда задач автоматизации широко применяются технологии машинного зрения и алгоритмы распознавания образов. Область применения таких программных средств достаточно широка: анализ изображения с камер видеонаблюдения, обработка данных с камер летательных аппаратов, автоматическое выделение приоритетных областей на разного рода снимках, разработка ПО автопилотирования беспилотных летательных аппаратов.

Для получения быстрого и точного результата анализа обрабатываемого изображения целесообразно использовать методы и принципы, решающие задачи распознавания изображения по выстроенному оптимальному алгоритму [5].



Методы и принципы распознавания образов

Для распознавания образов существует 3 основных метода:

1. Метод перебора вида объекта под различными углами, масштабами, смещениями и т. д. Для букв так же необходимо перебирать шрифт.
2. Метод поиска контура объекта и исследования его свойства (связность, наличие и количество углов и т. д.)
3. Использование искусственных нейронных сетей. Этот метод требует большого количества примеров задачи распознавания, на которых нейронная сеть сможет научиться правильно распознавать образы.

Так как основными ограничителями по качеству работы распознавания контуров являются скорость работы и достоверность результирующего контура, построение алгоритма распознавания изображения базируется на методе поиска контура объекта. Несмотря на значительную скорость работы нейросетей, они зависимы от обучающей базы, которая в свою очередь достаточно массивна [5].

Для удобства обработки изображения была взята библиотека OpenCV (Open Source Computer Vision Library). Данная библиотека имеет множество алгоритмов компьютерного зрения и является оснополагающей базой для разработки подобного рода ПО.

Построение алгоритма анализа и распознавания изображения на основе метода поиска контура объекта и исследования его свойств

Для правильного выделения контура необходимо выявить признаки, однозначно отличающие искомые пустоты объекта (являющиеся результатом поиска) от любых других возможных пустот. Таким образом, объект кадра должен иметь следующие свойства:

- являться четырёхугольником (4 грани): прямоугольник (в нормальном отображении окна) или трапеция (в перспективном отображении окна);
- наиболее высокую яркость области в сравнении с другими объектами;
- меньшую яркость соседних пикселей (по отношению к контуру).

Для удобства обработки используются несколько фреймов (кадров) из всего видеопотока с камеры. Каждый фрейм представляет собой RGB изображение с 8-битной глубиной цвета и разрешением 648 (ширина) на 486 (высота) пикселей.

Алгоритм разбивается на несколько этапов обработки изображения:

1. Нахождение размеров кадра (высота и ширина в пикселях).

Разрабатываемый алгоритм является универсальным, т.е. подходящим для изображений различного разрешения (для различных видеокамер). В связи с этим на первом шаге необходимо проверить размеры полученного изображения.

2. Преобразование изображения в оттенки серого.

Для облегчения бинаризации и получения более точного результата необходимо провести преобразование в оттенки серого. В качестве функции, преоб-



разующей изображение в градации серого, используется функция библиотеки «OpenCV» [3].

3. Размытие изображения по Гауссу.

С целью уменьшения резкости с границ объектов, в алгоритме используется размытие изображения по алгоритму Гаусса из библиотеки «OpenCV». В качестве параметров указывается размер ядра размытия [3].

4. Бинаризация изображения

Бинарное изображение - разновидность изображений, когда каждый пиксель может представлять только один из двух цветов, в данном случае – белый или черный.

Перевод изображения в бинарный формат используется с целью определения светлых и темных участков изображения. В качестве параметров указывается центр бинаризации (127) – граница яркости, по которой цвета разбиваются на черный и белый и верхний порог бинаризации (255) [4]. Значение данного параметра является верхней границей яркости.

5. Распознавание контуров фигур на изображении.

Обнаружение контуров основывается на алгоритмах, которые выделяют точки цифрового изображения. В таких точках резко изменяется яркость, или имеются другие неоднородности.

Подходящим алгоритмом является алгоритм Кэнни [1]. Применяется оператор обнаружения границ (контуров) изображения, полученного из видеопотока камеры.

6. Применение закрытия контуров на изображении.

В ходе предварительной обработки зачастую одним из результатов является неоднородность контура. Для того чтобы избавиться от подобного искажения, применяется алгоритм закрытия. Данная операция проводится на бинарных изображениях с целью избавления от разрывов в контурах фигур, тем самым убрав дефекты обнаружения.

7. Поиск контуров.

Выделение контуров в отдельный массив производится после того, как все контуры становятся закрытыми. Данная операция реализуется за счет использования функций библиотеки «OpenCV» [2].

8. Выбор контуров удовлетворяющих условиям

Для правильной выборки проводится анализ каждого найденного контура на соответствие заданным условиям.

Первоочередно выделяют из списка контуры с 4 гранями. В связи с тем, что контур представлен массивом, для продолжения анализа его предварительно необходимо аппроксимировать. Именно для этого используется функция `approxPolyDP` из библиотеки «OpenCV» [4]. В качестве параметров используется:

- контур, в виде объекта класса «`numpy array`»;
- максимальное расстояние между исходной кривой и ее аппроксимацией (позволяет задать точность приближения);



- параметр, определяющий закрытие контура. При задаваемом значении true, в приближенной кривой первая и последняя вершины будут соединены.

Результатом аппроксимации является список вершин контура. Таким образом, происходит проверка первого условия: контур является четырехугольником. Но если контур имеет более 4 граней, его анализ прекращается.

Последним действием анализа является сверка соседних с контуром пикселей. В соответствии с условием они должны быть черного цвета. С целью упрощения проверки данного условия, проверяются только те пиксели, которые находятся по координатам $(x-d, y+d)$ и $(x+d, y-d)$, где

- (x, y) координаты вершины,
- d – расстояние проверки.

Признак объекта нарушается, если пиксели, подвергшиеся проверке имеют иной цвет, отличный от черного. Если же все условия выполнены, искомым контур найден.

Данный алгоритм был протестирован на наборе изображений, в каждом из которых был безошибочно найден искомым контур. Результат тестирования представлен на рис 1.

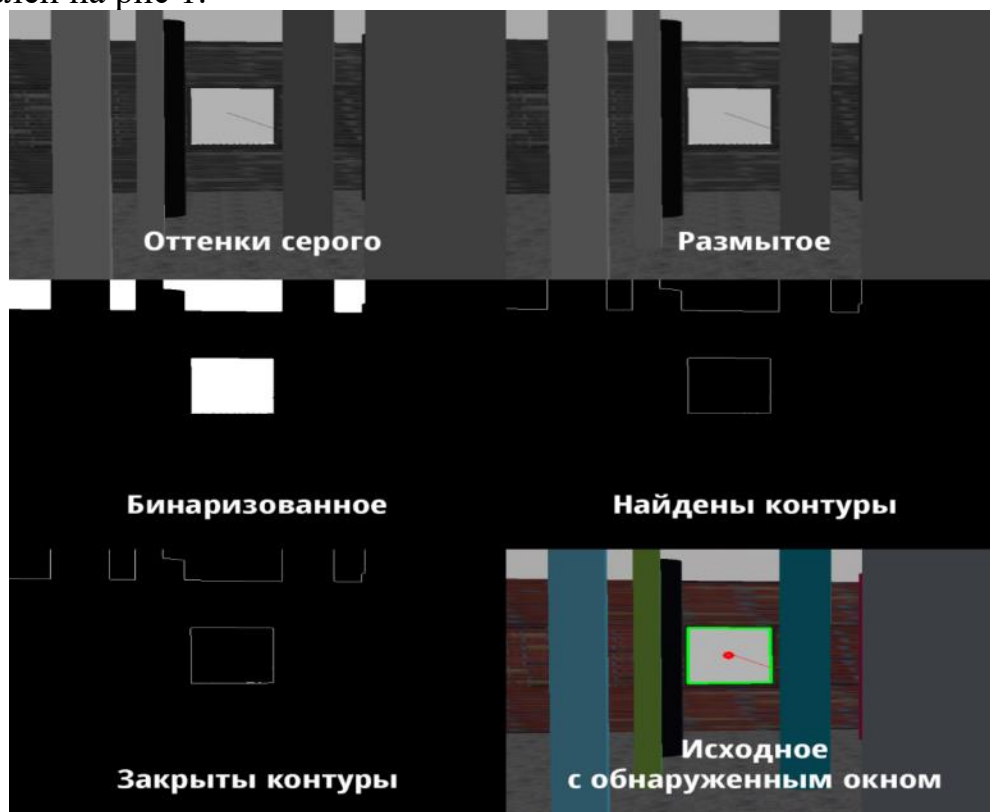


Рис. 1. Граница эффективности выхода Y3 к входу X3

Способ улучшения результата

В ходе тестирования алгоритма были выявлено, что проведение бинаризации по жесткому пороговому значению может привести к неточностям.

Для решения данных проблем было предложено проводить предварительный анализ с целью выявления пороговых значений для данного изображе-



ния и использование полученных значений при бинаризации. Данный факт позволит наиболее корректно проводить бинаризацию изображения.

Заключение

На основании проведенного анализа методов разработки и их основных принципов, разработан алгоритм распознавания объектов. Применение подобного алгоритма позволяет, с высокой точностью определять наличие пустот объекта. Алгоритм универсален и с легкостью может быть встроен в уже готовое ПО.

Литература

1. Лбов Г.С., Бериков В.Б. Устойчивость решающих функций в задачах распознавания образов и анализа разнотипной информации. - Новосибирск: Изд-во Ин-та математики, 2005. - 218 с.
2. Кэлер А., Брэдски Г. Изучаем OpenCV 3 = Learning OpenCV 3. - М.: ДМК-Пресс, 2017. - 826 с.
3. Буэно, Суарес, Эспиноса. Обработка изображений с помощью OpenCV = Learning Image Processing with OpenCV. - М.: ДМК-Пресс, 2016. 210 с.
4. Официальная документация к библиотеке OpenCV/ Интернет-ресурс. - Режим доступа: www/ URL: https://docs.opencv.org/- Загл. с экрана.
5. Теория распознавания образов Интернет-ресурс. - Режим доступа: www/ URL: https://ru.wikipedia.org/wiki - Загл. с экрана.

В.Б. Сахибазарова, М.А. Кудрина

ИССЛЕДОВАНИЕ АЛГОРИТМОВ ФРАКТАЛЬНОГО СЖАТИЯ ИЗОБРАЖЕНИЙ

(Самарский университет)

В настоящее время при передаче данных по сети учитываются два критерия: скорость передачи информации и объем передаваемых данных. Необходимо передать как можно больше информации в сообщении наименьшего размера. В случае передачи графической информации используются различные методы сжатия изображений для уменьшения объема передаваемых данных.

В данной работе рассматривается алгоритм фрактального сжатия изображений, основанный на том, что мы представляем изображение в более компактной форме – с помощью коэффициентов системы итерируемых функций Iterated Function System (IFS). IFS представляет собой набор трехмерных аффинных преобразований, переводящих одно изображение в другое. Преобразованию подвергаются точки в трехмерном пространстве (x_координата, y_координата, яркость) [1].

По своей сути, фрактальное сжатие (или фрактальная компрессия) – это процесс поиска самоподобных областей изображения и определения для них параметров аффинных преобразований.

Общий алгоритм фрактального сжатия представлен на рисунке 1.