

РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ РАСПОЗНАВАНИЯ ДОРОЖНОЙ ПОЛОСЫ НА ВИДЕОПОСЛЕДОВАТЕЛЬНОСТЯХ

*А.А. Друки, к.т.н., доц. ИШИТР ТПУ,
Д.С. Кружков, студент гр. 8ВМ91
Томский политехнический университет
E-mail: dsk30@tpu.ru*

Введение

Сегодня компьютерное зрение применяется для решения широкого спектра задач. В области транспортных средств компьютерное зрение имеет высокую степень интегрированности и является фундаментом решения всевозможных задач в системах активной и пассивной помощи водителю.

Одной из наиболее ярких спецификаций применения компьютерного зрения является расширение функционала и улучшение качества систем распознавания дорожной разметки [1].

Описание алгоритма

Разработка модуля по распознаванию дорожной разметки была выполнена на языке Python с использованием библиотеки OpenCV, с помощью которой был сформирован пул кадров из видеоряда, а также произведена их предобработка.

Первым шагом в процессе построения детектора является преобразование исходного цветового пространства изображения в «Grayscale». Такой подход позволит выделить границы разметки наиболее явно, а также отбросить лишнюю информацию, не имеющую отношения к разметке [2].

Далее необходимо затемнить полученное изображение, чтобы уменьшить контраст от обесцвеченных участков дороги и применить размытие по Гауссу, чтобы удалить лишние края на изображении в местах высокой контрастности, а также уменьшить шум.

Следующим шагом после предобработки к размытым снимкам применяется детектор Canny. Canny Edge Detection – алгоритм, который обнаруживает края на основе изменения градиента. Хотя первым его шагом по умолчанию является сглаживание изображения с размером ядра 5, наиболее корректным будет явное размытие по Гауссу на предыдущем этапе. Другие шаги работы детектора включают в себя: нахождение градиента интенсивности изображения, подавление максимумов, а также пороговый гистерезис [3].

Даже после детектора Canny на снимках остаётся много лишних линий и краёв, не являющихся разметкой. Область интереса – это многоугольник, охватывающий наиболее полезную часть изображения. Данный шаг является одним из ключевых в производстве анализа разметки, т.к. позволяет сформировать маску для входного изображения, тем самым отсеять все визуальные шумы и помехи, с которыми нет возможности справиться посредством предобработки кадров анализируемого видеопотока.

Заключительным шагом в работе детектора является анализ выделенной обработанной области на изображении и построение направляющих разметки с помощью функции библиотеки OpenCV – «HoughLinesP» [2].

Для финального вывода используется объединение начального и полученного кадров с определённой степенью прозрачности с помощью функции «addWeighted» библиотеки OpenCV. Результат анализа дорожной разметки в видеоряде представлен на рисунке 1.



Рис. 1. Поиск фрагмента сцены в видеофайле.

С помощью разработанного алгоритма можно осуществлять поиск дорожной разметки в видеоряде, в потоковом видео, а также на статичных изображениях. Никаких преобразований формата видеоряда при этом не требуется, однако, следует отметить, что при изменении положения источника видео требуется корректировка «области интереса».

Тестирование алгоритма распознавания дорожной разметки

В качестве тестовых примеров для проверки корректности выполнения алгоритма для определения дорожной разметки были использованы записи с видеорегистраторов с различной степенью освещённости, линейности трассы, различными погодными условиями, а также разного разрешения.

В процессе тестирования алгоритм показал наиболее стабильные и точные результаты на видеофрагментах с хорошим уровнем освещённости и наиболее линейной структурой дорожной полосы. Также была выявлена высокая степень зависимости от корректности настройки маски «области интереса изображения».

На кадрах с плохой освещённостью трассы или резкими градиентами «свет-тень» алгоритм выстраивал направляющие разметки менее точно, однако большая часть кадров (~80%) была обработана корректно, что подтверждает работоспособность разработанного модуля.

Заключение

Основываясь на результатах проведённого тестирования, можно сделать вывод о том, что разработанный алгоритм наиболее качественно работает на изображениях и видеорядах с хорошим освещением, чётко выделенной разметкой дорожной полосы.

Однако данный алгоритм имеет ряд недостатков, связанных с плохим ориентированием в среде с резкими световыми переходами (ночное время, тоннели, закрытые парковки).

Список использованных источников

1. Решение задачи по детектированию объектов на видео. [Электронный ресурс] – URL <https://newtechaudit.ru/detektirovanie-obektov/> (дата обращения: 17.01.2021).
2. Шпаргалка по OpenCV — Python. [Электронный ресурс] – URL <https://tproger.ru/translations/opencv-python-guide/> (дата обращения: 20.01.2021).
3. OpenCV шаг за шагом. Обработка изображения - детектор границ Канни (Canny). [Электронный ресурс] – URL <http://robocraft.ru/blog/computervision/484.html> (дата обращения: 17.01.2021).