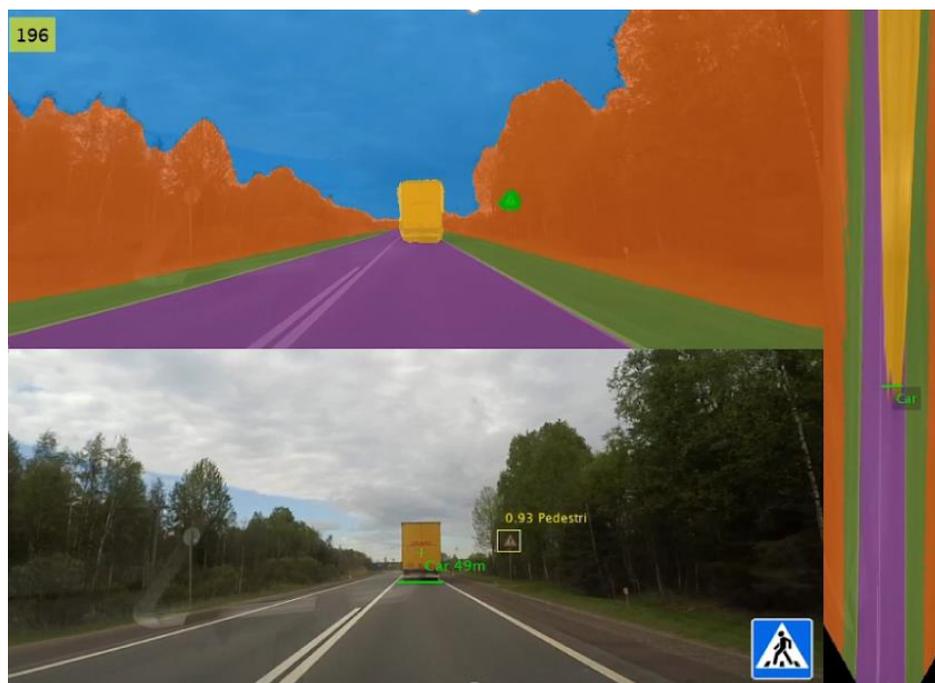


Семантическая сегментация изображения с камеры автомобиля для построения ADAS систем

Инженеры ЦИТМ Экспонента приняли участие во внутреннем пилотном проекте Экспоненты по семантической сегментации видео с камеры автомобиля для построения ADAS систем содействия водителю.

В результате проекта инженерами получено решение на основе нейронных сетей для сегментации исходного видео, распознавания автомобилей в кадре, вычисления расстояния до них и распознавания дорожных знаков, которое впоследствии может применяться как в ADAS системах содействия водителю, так и для построения автономных транспортных средств.

При необходимости данное решение может быть доработано под требования заказчика и смежные задачи: железнодорожный транспорт, навигация мобильных роботов, системы безопасности и т.д.



Проблема

Во многих задачах разработки систем содействия водителю и автономных транспортных средств на первом этапе требуется получить информацию о внешнем окружении объекта, определить другие объекты, классифицировать их и вычислить расстояние до них. Для детектирования объектов, как правило, применяют камеры, лидары и радары, а для классификации объектов используют изображение с камеры.

Решение

В нашем проекте мы ограничились использованием только одного датчика - камерой, ввиду широкого распространения камер уже в серийных автомобилях и возможностью выполнять детектирование и классификацию объектов по изображению с камеры автомобиля. Для сегментации исходного кадра с камеры применена дообученная сегментирующая нейронная сеть DeepLabV3+ с декодером на основе ResNet50. В пилотном проекте нейронная сеть сегментирует исходный кадр на следующие 6 классов: дорога, обочина, автомобили, дорожные знаки, небо и фон. Дальше, на основе данных с камеры и выполняя преобразование координат, вычисляется расстояние до автомобилей, и дополнительная нейронная сеть классифицирует изображения дорожных знаков на 14 классов. Демонстрация работы алгоритма:

[Демонстрация работы алгоритма.](#)

Этапы решения задачи

Сбор и предобработка данных

Исходные видео были записаны на экшен камеру Xiaomi Yi с разрешением 1920x1080 пикселей 30 кадров в секунду. Общая продолжительность записанного видео больше часа. Для обучения была выбрана пара видеофайлов продолжительностью 5 минут, далее из каждого видео был получен набор изображений – кадров. Так как между соседними кадрами происходит небольшое изменение сцены, то для обучения сохранялся только каждый 10-й кадр видео. Предварительно из исходного кадра вырезалась центральная область изображения и приводилась к размеру 850x355 пикселей (рисунок 1). Изображения для обучения специально сохранялись с немного большим разрешением 850x355 против 800x334 – разрешение первого слоя сети, чтобы при обучении иметь возможность выполнять аугментацию данных случайным сдвигом и кропом изображения.

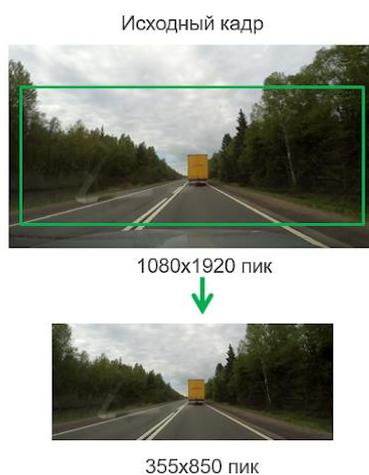


Рисунок 1. Предобработка исходного кадра

Разметка данных для сегментации и дообучение нейронной сети

На первом этапе для ручной разметки было отобрано 40 «сложных» изображений, содержащих пересечения со второстепенными дорогами, знаки, автобусные остановки. Данные изображения были вручную размечены в «Image Labeler App» на 6 классов: дорога, обочина, автомобили, дорожные знаки, небо, и фон (рисунок 2).

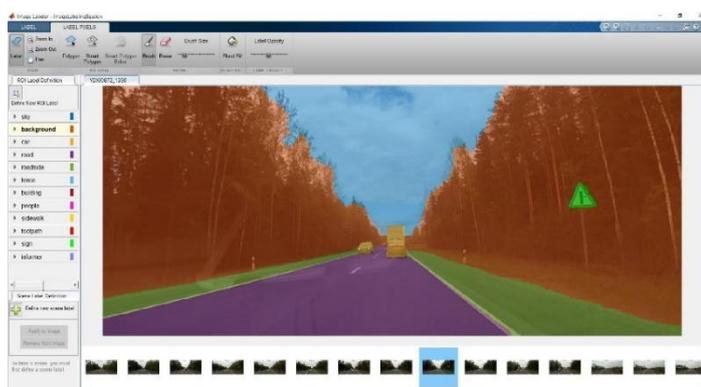


Рисунок 2. Разметка данных в «Image Labeler App»

Автоматизация процесса разметки данных

На следующем этапе для автоматизации разметки изображений, была выбрана и дообучена сеть SegNet на 40 размеченных изображениях с аугментацией на каждом шаге. В результате дообучения SegNet, при сегментации новых изображений уже верно сегментированы область неба, дороги, обочины, фона, но были проблемы с сегментацией автомобилей и дорожных знаков (рисунок 3).



Рисунок 3. Пример кадры с ошибками сегментации

Данную дообученную сеть можно использовать для автоматизации процесса разметки в «Image Labeler App» для предварительной сегментации кадра, а после вручную править те места, где есть ошибки с сегментацией. На новой расширенной базе изображений снова дообучается SegNet и, с ростом базы размеченных изображений, количество ошибок сегментации снижается (рисунок 4).



Рисунок 4. Пример сегментации кадра сетью SegNet дообученной по 200 (верхнее) и 400 (нижнее) изображениям

[Видео: Сегментация сетью SegNet дообученной по 400 изображениям.](#)

И так итерация за итерацией, в автоматизированном режиме есть возможность расширить количество размеченных изображений и повысить качество сегментации обучаемой сети. На финальном этапе выборка для обучения содержала 400 размеченных изображений, дообучение сети на которой давала хороший результат, в целом такой итерационный подход ускоряет процесс разметки исходных данных.

[Видео по разметке данных в MATLAB.](#)

Выбор и дообучение нейронной сети

При обучении размеченная выборка составляла 400 изображений и разделялась в соотношении 90/10: 90% - обучение, 10% - валидация. Изначально, в качестве сегментирующей сети, была выбрана сеть SegNet (декодер VGG16). Обучение происходило в MATLAB с использованием библиотеки Deep Learning Toolbox для обучения сети. Сеть обучалась на одной видеокарте GeForce GTX 1080, время обучения составляло около 1 часа. Наибольшую сложность для сети вызвала сегментация небольших объектов - дорожных знаков, для знаков параметр IoU (общая площадь сегментированной и размеченной области в исходных данных) равен 0.50, для остальных областей параметр IoU был не менее 0.97.

В дальнейшем были опробованы и другие архитектуры нейронных сетей: DeepLabV3+(декодер MobileNetV2) IoU=0.29 для знаков; DeepLabV3+(Xception) IoU=0.47; DeepLabV3+(InceptionResNetV2) IoU=0.35; U-Net IoU=0.18; DeepLabV3+(ResNet50) IoU=0.40, настроив параметры обучения удалось получить U=0.56 для знаков. Результаты тестирования для 2-х лучших сетей на тестовой выборке представлены на рисунке 5.

SegNet (VGG16)	Accuracy	IoU	MeanBFScore
sky	0.98578	0.97627	0.97924
background	0.97853	0.96693	0.96825
car	0.97543	0.88232	0.94554
road	0.98708	0.98031	0.98733
roadside	0.97683	0.91768	0.97826
sign	0.99488	0.50523	0.80634
DeepLabV3+ (ResNet50)	Accuracy	IoU	MeanBFScore
sky	0.98582	0.97993	0.9716
background	0.98259	0.96966	0.96958
car	0.99487	0.94839	0.96175
road	0.98855	0.98579	0.99515
roadside	0.98965	0.94422	0.98785
sign	0.99805	0.56501	0.9388

Рисунок 5. Результаты тестирования SegNet и DeepLabV3+ на тестовой выборке

[Видео: Сегментация сетью DeepLabV3+.](#)

Измерение расстояний до объектов

Следующей важной задачей является определение расстояния до обнаруженных объектов. При построении ADAS систем содействия водителю важно определять расстояние до обнаруженных автомобилей. Используя только изображение с камеры можно вычислять расстояние до объектов, при условии, что высота подвеса камеры постоянна и профиль дороги – прямой. Если дорога в конце начинает подниматься или спускаться вниз, то возникают ошибки с определением расстояния, но на практике это возникает для объектов на удалении более 150 метров и не является столь критичным, т.к. происходит для объектов в дальней зоне.

Для вычисления расстояния необходимо знать внутренние и внешние параметры камеры. Внутренние параметры камеры можно найти с помощью процедуры калибровки в «Single Camera Calibrator App», а внешние параметры камеры — это высота ее подвеса и угол наклона.

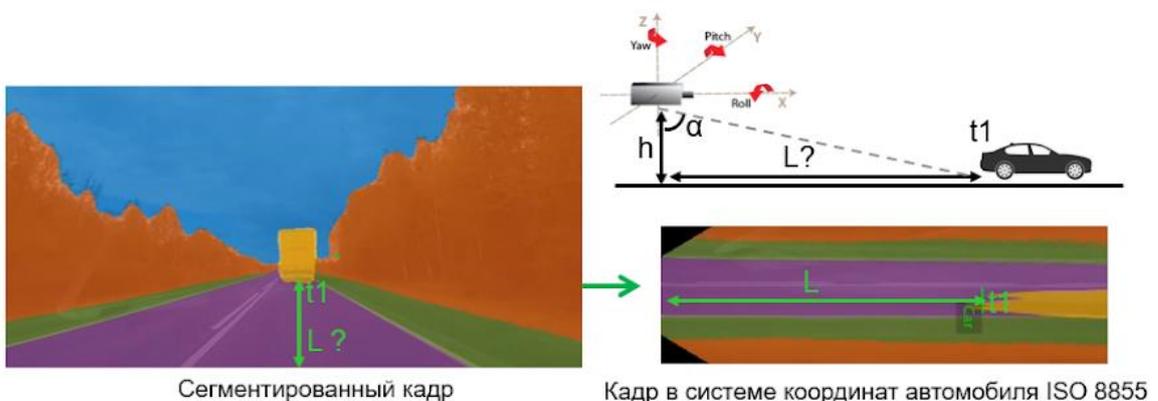


Рисунок 6. Вычисление расстояния до автомобиля.

Для нахождения расстояния до объекта на сегментированном кадре находятся координаты прямоугольной области (bbox) описывающей машину, вычисляются координаты точки t_1 , после, используя преобразование координат, вычисляются координаты точки t_1 в системе координат ISO 8855 «вид сверху», в новой системе, координата x – и есть расстояние до встречного автомобиля.

Для пересчета координат в систему ISO 8855 используются функции из библиотеки Automated Driving Toolbox. В этой системе координат эго-автомобиль находится в центре системы координат, ось y - направлена влево, ось x – прямо, ось z – вверх, расстояния измеряться в метрах.



Рисунок 7. Пример измерения расстояния до автомобилей.

[Видео: Пример измерения расстояния.](#)

Классификация дорожных знаков

Следующей задачей было научиться классифицировать дорожные знаки. После сегментации мы уже знали области на изображении, где находятся знаки, но не знали какие именно знаки перед нами. Для классификации знаков мы решили использовать небольшую сверточную сеть принимающую на вход изображения размером 40x40 пикселей. Архитектура нейронной сети для классификации дорожных знаков – рисунок 8.

```

layers = [
    imageInputLayer([40 40 3])
    convolution2dLayer(3,8,'padding','same')
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer(2,'stride',2)
    convolution2dLayer(3,16,'padding','same')
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer(2,'stride',2)
    convolution2dLayer(3,32,'padding','same')
    batchNormalizationLayer
    reluLayer
    fullyConnectedLayer(14)
    softmaxLayer
    classificationLayer1;

```

Рисунок 8. Архитектура нейронной сети для классификации дорожных знаков

Из всех записанных видеофайлов, используя дообученную сегментирующую сеть, была собрана база знаков. После сегментации кадра находилась прямоугольная область, описывающая знак и соответствующая область знака вырезалась из исходного кадра с большим разрешением и сохранялась в виде отдельного png – файла.



Рисунок 9. Примеры знаков для классификации

После собранная база изображений знаков была рассортирована на 14 классов, пример на рисунке 9 (включая класс изображений «не знак» – ветви деревьев и т.д.). При обучении сети каждый класс содержал от 200 изображений, на тестовой выборке точность классификации составила 99%. На рисунке 10 показан пример распознанных сетью знаков. При распознавании знаков в реальном видео иногда возникают случаи, когда группа знаков сегментируется как один объект: несколько знаков на столбе или несколько знаков рядом, что не является ошибкой сегментации. В таких случаях можно попытаться разделить знаки на основе информации о длине и ширине объекта (например, если несколько знаков на столбе расположены вертикально) или использовать сеть для «Instance detection».



Рисунок 10. Классы знаков и вероятность их распознавания

Для 9 классов знаков у которых нет подклассов предварительно сохранены иконки знаков (рисунок 11), а, например, для класса 5.15 «движение по полосам» иконки нет т.к. в него входит группа знаков 5.15.1-5.15.8. Для вывода информации о распознанных знаках используется следующая логика: если вероятность распознавания больше 50% то сверху над знаком выводится его класс, если вероятность больше 80% и для знака есть иконка, то, для большей наглядности, внизу кадра отображается иконка знака.



Рисунок 11. Иконки для 9 классов знаков

[Итоговое видео демонстрации работы алгоритма.](#)

Заключение

В результате пилотного проекта получен алгоритм содействия водителю, осуществляющий с сегментации видео, распознавания автомобилей в кадре, вычисления расстояния до них, и выполняющий распознавание дорожных знаков. В дальнейшем данный алгоритм можно расширить системой удержания автомобиля в полосе, адаптивным круиз контролем или системой экстренного торможения.

В ходе проекта инженерами Экспоненты получен опыт по сбору базы изображений, разметке данных, по выбору и дообучению нейронных сетей, распознаванию дорожных знаков, а также опыт по переносу алгоритмов на встраиваемые платформы NVIDIA Jetson.



Рисунок 12. Запуск алгоритма сегментации на Jetson AGX XAVIER

[Видео: Запуск алгоритма сегментации на Jetson AGX XAVIER.](#)

Использованные продукты:

[MATLAB](#)

[Deep Learning Toolbox](#)

[Parallel Computing Toolbox](#)

[Computer Vision Toolbox](#)

[Automated Driving Toolbox](#)

[Image Processing Toolbox](#)



ЭКСПОНЕНТА
ЦЕНТР ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ
И МОДЕЛИРОВАНИЯ

Контакты

exponenta.ru

E-mail: info@exponenta.ru

Тел.: +7 (495) 009 65 85

Адрес: 115088 г. Москва,

2-й Южнопортовый проезд, д. 31, стр. 4