Здравствуйте, хочу представить вам свою выпускную квалификационную работу на тему «Разработка автоматизированной системы обнаружения и анализа дорожных объектов с помощью искусственного интеллекта».

**Слайд 2.**

Вероятнее всего скоро мы будем наблюдать появление на наших дорогах большого числа беспилотных автомобилей. В процессе движения они должны непрерывно оценивать дорожную обстановку, а важнейшим ее элементом являются дорожные знаки, поэтому создание систем их распознавания, причем на основе современных средств и методов искусственного интеллекта, является актуальной задачей для специалистов в области ИТ.

**Слайд 3.**

Целью данной работы является повышение удобства эксплуатации автомобилей в части создания беспилотных систем управления, неотъемлемой частью которых является подсистема обнаружения и анализа дорожных объектов. Также на слайде приведены задачи работы, объект и предмет исследования.

**Слайд 4.**

На данном слайде представлена краткая характеристика самой задачи распознавания. При этом система может реализовывать функцию, допускающую всего лишь два разных значения: 0 – если объекта нет на изображении и 1 – если он там есть. Также функция может выдавать номер объекта, обнаруженного на изображении.

В работе также выделены основные этапы решения задачи распознавания, приведенные на слайде.

(Всю задачу распознавания образов можно разбить на отдельные этапы, некоторые из которых могут отсутствовать в каждом конкретном случае:

а) выделение отдельных участков изображения, которые могут представлять собой объекты *oi*, которые подлежат распознаванию. Здесь могут применяться методы кластерного анализа (для относительно однородных изображений). Также возможна разбивка на относительно крупные контрастные области (в случае, если таковые присутствуют на изображении);

б) очистка выделенных участков от посторонних шумов и других визуальных загрязнений, для чего целесообразно применять разнообразные методы цифровой фильтрации (часто – на базе обобщенного разложения Фурье, с отбрасыванием/ослаблением/усилением некоторых частотных составляющих);

в) линейное преобразование изображения, заключающееся в его повороте, что может выполняться с использованием формул преобразования координат типа «поворот» (при необходимости поворота на произвольный угол) или путем простой замены координаты *х* на *у* и наоборот.

г) линейное преобразование, заключающееся в масштабировании изображения и приведении к размеру, близкому к размеру эталонных изображений объектов *oi*;

д) непосредственно проведение процедуры сравнения с эталонами (эталоном);

)

Нейрон – базовая единица нейронной сети. У каждого нейрона есть определённое количество входов, куда поступают сигналы, которые суммируются с учётом значимости (веса) каждого входа. Далее сигналы поступают на входы других нейронов. Вес каждого такого «узла» может быть как положительным, так и отрицательным. Например, если у нейрона есть четыре входа, то у него есть и четыре весовых значения, которые можно регулировать независимо друг от друга.

**Слайд 5.**

Методы решения задачи распознавания объектов на изображении могут быть самыми разными, начиная от сравнения характеристик, проверяемого и эталонного изображений, и заканчивая современными методами искусственного интеллекта.

В частности, лучше всего с такими задачами справляются нейронные сети, причем в случае символьной информации вполне достаточно возможностей более простых сетей прямого распространения, а в случае с нерегулярными изображениями (как, например, дорожные знаки) для этого лучше применять сверточные нейронные сети, что и выполнено в данной работе.

**Свертывающие нейронные сети так называют потому, что в их структуре используется операция свертки (попарного умножения с последующим сложением всех произведений) элементов анализируемого изображения (обычно задаваемых окном небольшого размера) с некоторыми шаблонами (или фильтрами), характеризующими наличие на картинке тех, или иных признаков**

И рассмотрев существующие методы исредства для распознавания объектов, можно сделать вывод, что специализированные решения, нацеленные именно на определение дорожных объектов, отсутствуют. Это говорит о том, что целесообразной является разработка собственной автоматизированной системы обнаружения и анализа дорожных объектов с помощью искусственного интеллекта.

(Нейрон – базовая единица нейронной сети. У каждого нейрона есть определённое количество входов, куда поступают сигналы, которые суммируются с учётом значимости (веса) каждого входа. Далее сигналы поступают на входы других нейронов. Вес каждого такого «узла» может быть как положительным, так и отрицательным. Например, если у нейрона есть четыре входа, то у него есть и четыре весовых значения, которые можно регулировать независимо друг от друга.­)

Если что, смотри в 1.2 в ВКР

**Слайд 6.**

Были разработаны алгоритмы, необходимые для реализации рассмотренного метода в программных кодах: слева алгоритм работы всей программы, а справа – обучения нейронной сети, которое является самым алгоритмически сложным процессом во всей этой системе.

 Можно привести следующую последовательность действий:

а) устанавливаем веса синапсов сети на произвольные (случайные) значения;

б) пропускаем очередное изображение из обучающего дата-сета через сеть;

в) оцениваем выход, в частности, с помощью функции потерь – разницы между реальным выходом и требуемым значениями;

г) корректируем веса синаптических связей;

д) если не все изображения из обучающего набора (дата-сета) просмотрены, то переходим к пункту *б*;

е) обнуляем список просмотренных изображений (т.к. закончена очередная эпоха обучения);

ё) если максимальное рассогласование желаемого выхода с реальным наблюдаемым значением, которое наблюдалось на протяжении последней эпохи, больше точности обучения, то переходим к еще одной эпохе обучения, т.е. к пункту *б*;

ж) сеть обучена и ее можно использовать.

Сама нейронная сеть создается в виде следующего класса (следует отметить, что архитектура сети предложена самим разработчиком на основе анализа схожих примеров, но используются стандартные классы из библиотеки Keras/TensorFlow, что является общемировой практикой при программировании искусственного интеллекта на таком высокоуровневом языке, как Python)

**Слайд 7.**

Здесь можно видеть архитектуру сети, примененной для распознавания в данной работе. Она является комбинацией известных архитектур сетей глубокого обучения, часто применяемых для распознавания стандартных наборов типа MNIST, и что самое главное, как будет описано далее она показывает достаточно хорошие результаты распознавания.

**Свертывающие нейронные сети так называют потому, что в их структуре используется операция свертки (попарного умножения с последующим сложением всех произведений) элементов анализируемого изображения (обычно задаваемых окном небольшого размера) с некоторыми шаблонами (или фильтрами), характеризующими наличие на картинке тех, или иных признаков. Саму свертку выполняют слои типа Conv2D из библиотеки Keras. При этом результат свертки таким конволюционным слоем обычно направляют на еще один свертывающий слой, который и определяет наличие на частях изображения тех или иных признаков. Таким образом, конволюционные слои обычно используются последовательными парами.**

**Следующим фактором, который необходимо учесть, является то, что на изображении распознаваемый элемент (дорожный знак) может присутствовать не обязательно в центре и занимать всю площадь – он может быть относительно малым по размерам и занимать лишь часть изображения, но может и занимать полностью всё изображение. Чтобы распознавать объекты, занимающие произвольную часть изображения, необходимо применять операцию пулинга, позволяющего уменьшить изображение. Пулинг обычно выполняется по максимальному сценарию, то есть группа пикселей (например, 2х2) заменяется одним пикселем, яркость которого равна максимальному значению из всех пикселей, входящих в блок. После пулинга изображение снова пропускается через сдвоенный набор свертывающих слоев.**

**Наконец, еще одним приемом, который позволяет значительно улучшить характеристики получаемой нейронной сети, является dropout – отсечение на небольшое время (например, на одну эпоху обучения или даже ее часть) некоторых нейронов, что осуществляется с помощью некоторой заданной вероятности «выключения» нейрона *р*. Такой подход позволяет исключить слишком высокую специализацию нейронов, а значит и переобучение (overfitting), который может наблюдаться в тех случаях, когда количество нейронов в сети избыточно.**

**Результаты такой обработки входного изображения нужно подать на обширный распознающий слой из нескольких сотен нейронов, затем можно применить защиту от переобучения типа Dropout и наконец выходной результат считать с еще одного полносвязного слоя с числом нейронов, равным количеству классов классификации, т.е. 43.**

**Слайд 8.**

Далее был обоснован выбор средств разработки, а именно языка Python, который, в основном и применяется для решения задач, связанных с отраслью ИИ. В качестве среды разработки взята PyCharm – наиболее удобная среда, созданная на базе движка IntelliJ IDEA. Для реализации нейронных сетей использована библиотека KERAS, на сегодняшний день встроенная в пакет TensorFlow.

**Слайд 9.**

Как известно, для обучения любой нейронной сети требуется дата-сет и такой был найден на открытом ресурсе Kaggle.Был взят дата-сет, который был создан для конкурса по вопросам распознавания изображений, проводимого на Международной объединенной конференции по нейронным сетям. Всего там 50 тыс. изображений, из которых для обучения используется около 40 тыс., а остальные – для валидации в процессе обучения и для тестирования, которое является заключительным этапом в данной работе.

(Так, по рассматриваемой задаче достаточно качественный дата-сет был получен из открытых источников, а именно из наибольшего хранилища всех возможных наборов данных – всемирно известного ресурса Kaggle – рисунок 3.2, посвященного проблемам Big Data. На этом ресурсе можно найти дата-сет по дорожным знакам общим объемом около 640 МБ, который был создан для конкурса по вопросам распознавания изображений, проводимого в 2011 году на Международной объединенной конференции по нейронным сетям в Сан-Хосе, Калифорния. Также следует упомянуть тот факт, что изображения дорожных знаков являются одинаковыми по всему миру за некоторыми несущественными деталями, поэтому использование этого дата-сета в российских условиях может считаться вполне правомерным.)

**Слайд 10.**

В результате практической реализации было создано необходимое программное обеспечение. Слева сверху видно визуализацию дата-сета, разбитого по классам (всего 43 дорожных знака). Справа показана динамика процесс обучения сети. Видно, что уже за 5-7 эпох сеть выходит на более-менее значительную точность. Внизу видно, что на 25 эпохе точность распознавания знаков на валидирующем множестве составляет 99,21%, хотя потом на тестовом сеть выдает точность всего лишь около 95%. Следует отметить, что такая точность высокая для решения задачи распознавания образов в целом. Однако, для области управления беспилотным автомобилем такой точности явно недостаточно, поэтому рекомендуется применять ансамблевые методы, для повышения общей точности распознавания еще до более высоких значений.

**Слайд 11.**

На данном слайде представлена Обобщенная матрица ошибок созданной системы распознавания. И поскольку в ней 43 класса классификации, то условно принято отсутствие негативных реальных значений и любое реальное значение выхода считается положительным, а для сети негативный результат – это несовпадение с реальным значением

**Слайд 12.**

Таким образом, в работе проведена разработка автоматизированной системы обнаружения и анализа дорожных объектов с помощью ИИ, а именно нейронных сетей глубокого обучения. Эксперименты, проведенные на полученном продукте, показали точность распознавания на тестовом наборе данных около 95%. Разработка экономически обоснована, т.к. ее внедрение удорожает стоимость одного авто серии не более, чем на 0,01%. Таким образом, работу можно считать оконченной, а созданный продукт – рекомендовать к реальному практическому использованию, желательно в комплексе с другими методами распознавания.

**Слайд 13.**

**Спасибо за внимание**

Термином «искусственная нейронная сеть» или просто «нейронная сеть» часто обозначают математическую модель, очень упрощенно описывающую общий алгоритм работы человеческой нервной системы.

Центром нервной системы является мозг, представленный сетью нейронов. Специальные рецепторы преобразовывают сигналы от тела и из окружающей среды в электрические импульсы и передают ее в мозг. Мозг получает информацию от рецепторов в виде электрических импульсов, анализирует ее и выдает соответствующие решения эффекторам также в виде электрических импульсов. Эффекторы преобразовывают электрические импульсы, сгенерированные мозгом, в выходные сигналы.

Каждая нервная клетка, нейрон, отдельно работает по такой же трехступенчатой модели. У нервных клеток есть дендриты и аксоны. Дендриты - рецепторы, зоны приема - похожи на дерево, представляют собой неровную поверхность со множеством окончаний. Аксон же у каждой нервной клетки один и похож на кабель или линию электропередачи, по которой дальше передается некий сигнал, полученный в результате прохождения через клетку совокупности сигналов от дендритов. То есть все поступившие сигналы как бы смешиваются, «суммируются», и полученный «обработанный» сигнал передается дальше.

Синапсы – места соединения аксона одного нейрона с телом или отростком (дендритом или аксоном) другого нейрона. Через синапсы сигналы передаются от нейрона к нейрону.

У каждого синапса есть свой «вес», то есть как бы своя «важность», придаваемая значению сигнала, проходящего через данный синапс.

**Какая сеть используется:** Свертывающие нейронные сети (convolutional neural networks, CNN) и глубинные свертывающие нейронные сети (deep convolutional neural networks, DCNN) сильно отличаются от других видов сетей. Обычно они используются для обработки изображений, реже для аудио. Типичным способом применения CNN является классификация изображений: если на изображении есть кошка, сеть выдаст «кошка», если есть собака – «собака». Такие сети обычно используют сканер, не пропускающий все данные за один раз. Например, если есть изображение 200×200, сеть не будет сразу обрабатывать все 40 тысяч пикселей. Вместо этого сеть будет рассматривать квадраты размера 20 x 20 (обычно начиная с левого верхнего угла), затем сдвинется на 1 пиксель и будет рассматривать новый квадрат, и т.д. Эти входные данные затем передаются через свертывающие слои, в которых не все узлы соединены между собой. Эти слои обладают свойством сжиматься с глубиной, причем часто используются степени двойки: 32, 16, 8, 4, 2, 1. На практике к концу CNN прикрепляют FFNN для дальнейшей обработки данных. Такие сети называются глубинными (DCNN).

Сети данного типа, во-первых, чаще всего работают с изображениями, а, во-вторых, используются для классификации (например, по типу «есть определенный объект на картинке»/«этого объекта на картинке нет»). Поэтому свертывающие сети идеально подходят для задачи распознавания объектов.

важным аспектом проблемы является анализ комплекса дорожных объектов, которые могут способствовать лучшему пониманию дорожной обстановки автоматической системой управления. К таким объектам, в первую очередь, надо отнести:

- статические элементы, относящиеся собственно к особенностям дороги в данной местности: перекрестки, повороты, подъемы, спуски, сужения или расширения, одним словом все объекты, которые не могут менять свое состояние на дороге;

- динамические элементы, относящиеся к особенностям дороги, но могущие изменять свое состояние: светофоры, регулировщик, светофоры реверсивных полос, шлагбаумы железнодорожных переездов и охраняемых территорий и т.п.

- другие транспортные средства: движущиеся и остановленные;

- пешеходов;

- статические элементы дороги, которые не являются естественными ее составляющими, а нанесены специально для облегчения движения автомобилей по ней, в частности это дорожные знаки и разметка.

Именно последний случай является с одной стороны достаточно важным, т.к. дорожные знаки являются одним из основных регуляторных элементов в задаче управления автомобилем на дорогах общего пользования, а с другой стороны – запрограммировать их систему распознавания в целом значительно проще, чем, например, регулировщика. Следует отметить, что при ограниченных объемах и ресурсах, доступных при написании работы бакалавра, запрограммировать распознавание всех перечисленных выше объектов не представляется возможным, так как, например, в компании Яндекс над задачей создания беспилотного автомобиля в течение многих лет трудится команда из многих десятков программистов. Поэтому в дальнейшем решению в этой работе будет подлежать лишь задача распознавания дорожных знаков, как чрезвычайно важных объектов, но при этом достаточно надежно поддающихся распознаванию.