

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ рОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования

**«Дальневосточный федеральный университет»**

|  |  |
| --- | --- |
| **ШКОЛА ЕСТЕСТВЕННЫХ НАУК**  **кафедра информатики, математического и компьютерного моделирования** | |
| «Схема двуцикли» |

**КУРСОВАЯ РАБОТА**

по образовательной программе подготовки бакалавров

по направлению 01.03.02 «Прикладная математика и информатика»

|  |  |
| --- | --- |
| Работа защищена  с оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Регистрационный номер \_\_\_\_\_\_\_  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_\_г. | Студентка группы № Б8117-01.03.02  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Глушкова Д.И.  (подпись)  «\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_\_\_г.  Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (должность, ученое звание)  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись) (ФИО)  «\_\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_\_г. |

г. Владивосток

2020

Оглавление

[1. Введение 2](#_Toc43651769)

[Глоссарий. 3](#_Toc43651770)

# Аннотация

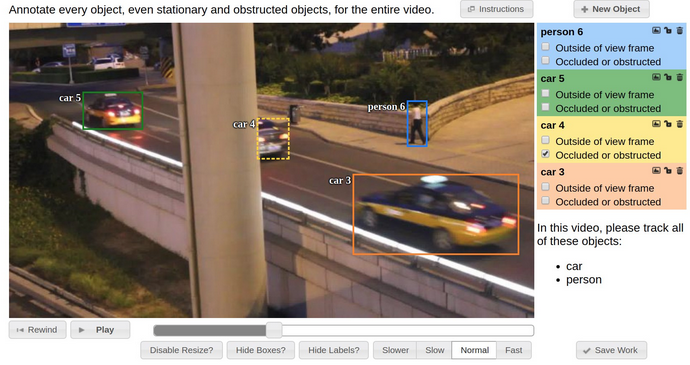
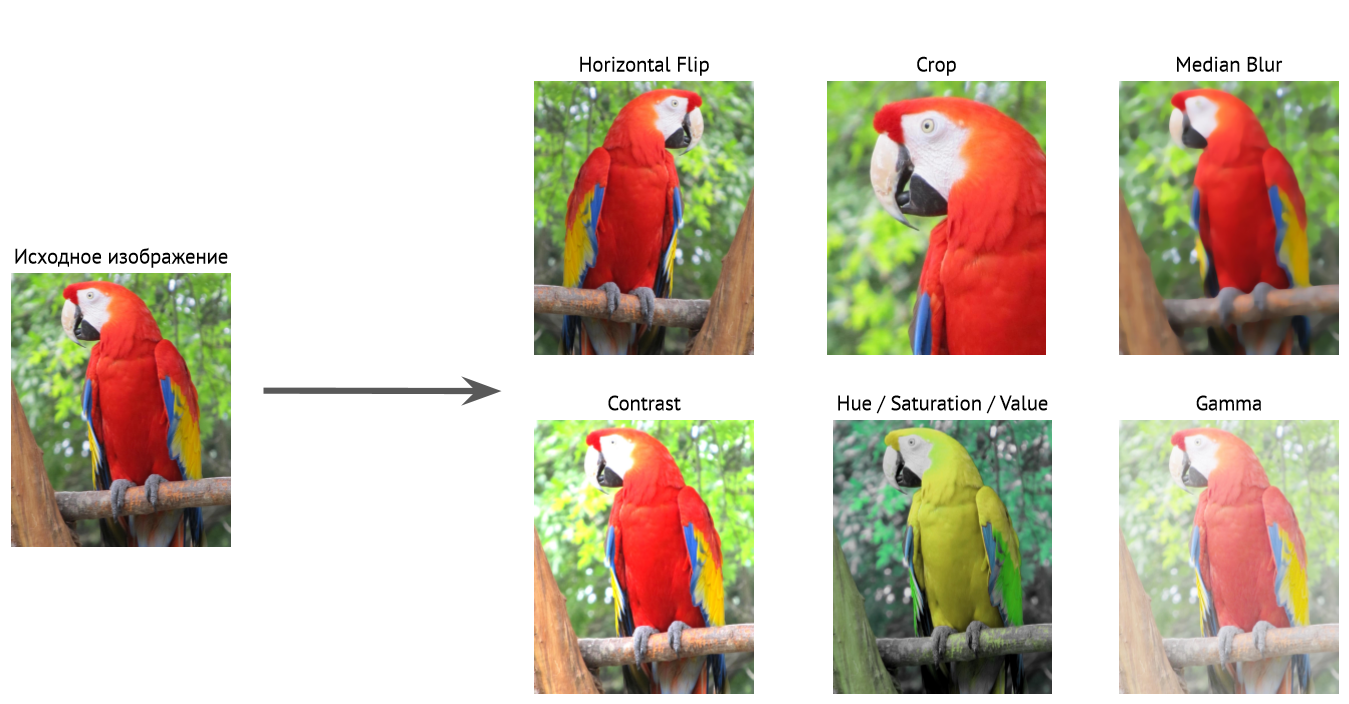
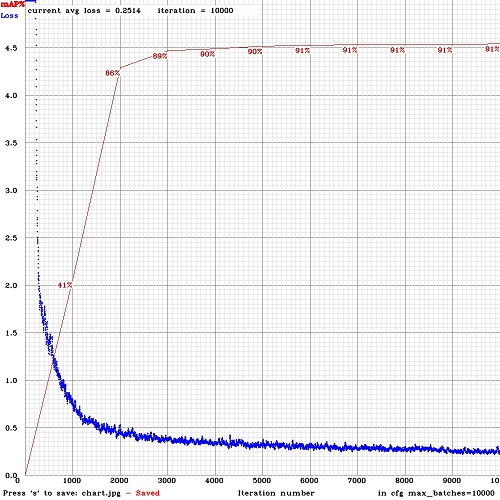
Целью данной работы является разработка веб-интерфейса для фреймворка нейронных сетей с функционалом для подготовки данных для обучения и расширения датасета, визуализации процесса обучения нейросети и с возможностью контроля версий результатов обучения в зависимости от входных данных.

# Введение

1.1 Глоссарий

1. average precisions –
2. CUDA
3. Вес -
4. Глубокое обучение – мощный набор техник обучения нейросетей.
5. Датасет -
6. Нейрон – это вычислительная единица, которая получает информацию, производит над ней простые вычисления и передает ее дальше. нейроны оперируют числами в диапазоне [0,1] или [-1,1].
7. Нейронная сеть — это последовательность нейронов, соединенных между собой синапсами.
8. Операторы управления потоком – повышают производительность и универсальность сивольной системы
9. Персептрон – это система, имитирующая все процессы внутри головного мозга.
10. Синапс – это связь между двумя нейронами. У синапсов есть 1 параметр — вес.
11. Функция потерь -
12. Эпоха -

# 1.2. Описание предметной области

* Искусственная нейронная сеть – математическая модель, а также её программная или аппаратная реализация, построенная по принципу организации и функционирования биологических нейронных сетей – сетей нервных клеток живого организма. Нейросетевой комплекс формируется из синапсов, имеющих свой вес, преобладание которого формирует исходящий из нейрона сигнал.
* Актуальность исследований в направлении нейронных сетей состоит в огромной предполагаемой сфере их полезности. Одним из таких направлений является распознавание и классификация объектов на изображении, что может быть полезно в развитии таких областей, как сельское хозяйство, безопасность, робототехники и т.д.
* Рассмотрим этапы работы с нейронными сетями.
* 1 этап – разметка данных. Все начинается с подготовки данных: сделаем по 2-3 фотографии объекта на разных фонах, с помощью Computer Vision Annotation Tool (cvat) создаем категорию под тип объекта и обводим объект, на котором хотим обучить нейронную сеть. Выглядит этот процесс следующим образом:
* 2 этап – аугментация. Она заключается она в том, что мы насильно портим изображение: добавляем размытие, зернистость, изменяем цвет и т.д. Аугментация нужна для расширения датасета, для того, чтобы сделать сеть инвариантной к некоторым преображением а также выступает в качестве регуляризатора и улучшают обобщающую способность сети.
* При составлении датасета важно помнить про явление переобучения, то есть, излишнее «подстраивание» под обучающую выборку из изображений. Чтобы избежать переобучения, нежно держать баланс между количеством различных данных по каждой категории. Не менее важно учитывать, что исходные данные должны быть однозначными, чтобы не возникали ситуации, когда нейронная сеть будет выдавать высокие вероятности принадлежности объекта к нескольким классам.
* 3 этап – обучение. Есть несколько видов машинного обучения: классическое и глубокое. Рассмотрим глубокое обучение. Оно предусматривает моделирование признаков данных с использование многослойных фильтров, позволяет строить более точную модель или сократить время на ее создание. Чем больше слоев, тем больше характеристик нейронная сеть может распознать. Недостатками глубинного обучения являются большие вычислительные мощности и сложность интерпретации получаемых моделей, а чем больше слоев, тем больше времени требуется для расчета, и тем сложнее будет обучение.
  + Обучение нейронной сети выполняется аналогично машинному обучению – берется датасет, сравниваются входные данные с желаемыми выходными, формируется вектор ошибок и на их основе применяются правки к сети.
* В конце обучения можем наблюдать похожий график:
  + Красной линией отмечено среднее значение average precisions. Обучение можно продолжать, пока эта величина будет увеличиваться.
  + Синей линией отмечена функция потерь, обучение нужно продолжать, пока ее значение != 0.24
  + Изображенный на графике процесс обучения прошел 100000 итераций, или эпох.
  + *График обучения нейронной сети.*
* Писать программы для глубокого обучения можно, конечно, с нуля, но гораздо проще и эффективнее использовать фреймворки, коих большое разнообразие в сети.
* В рамках проекта по распознаванию монет для сбербанка от ШЦЭ (2019 г.) в качестве фреймворка был использован даркнет и в работе с ним возникал ряд неудобств, из которых можно выделить:
  + Во время запущенного процесса обучения неочевидно, какое именно обучение запущено. При запуске с консоли эта проблема решается – отображается выполняемая команда;
  + Во время одновременного запуска нескольких процессов обучения с помощью графиков невозможно сравнить результаты, т.к. на графике нет пометок (см. График обучения нейронной сети);
  + При прерывании процесса обучения график, который был до остановки, не сохраняется и отображается уже с текущей точки;
    - Обучали нейронную сеть датасетом, там пишется все в файл chart.png и он один, там нет chart01.png, когда два процесса обучения, они пишут в один файл и если обучаешь, а потом запускаешь новое обучение с расширенным датасетом, старый chart.png не сохраняется, а перезаписывается на новый график. А мы хотим сравнивать эти графики.
* «На основе существующих решений выявлена потребность в разработке … » - 1 абзац

**1.3. Неформальная постановка задачи**

Система должна предоставлять пользователю следующие возможности:

* Подготовка данных для обучения
* Аугментация данных
* Балансировка данных
* Запуск процесса обучения на сервере
* Визуализация процесса обучения в realtime
* Контроль версий результатов обучения в зависимости от входных данных
  1. **Обзор существующих решений**
* Проведем сравнение между известными и использованным в данной работе.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Характеристика | Theano | TensorFlow | MXNET | Darknet |
| Автор | Монреальский университет | Команда Google Brain | Deep Machine Learning Community |  |
| Открытый исходный код | + | + | + | + |
| Платформа | Кросс-платформенное решение | Linux, Mac OS X, планируется поддержка Windows | Ubuntu, OS X, Windows, AWS, Android, iOS, JavaScript | Windows |
| Язык программирования | Python | C++  Python | C++, Python, Julia, Matlab, R, Scala | С |
| Поддержка CUDA | + | + | + | + |
| Наличие предварительно обученных моделей | + | - | + | + |
| Добавление новых операций | Можно добавлять операции на Python с поддержкой встроенных операторов С | Forward на C++, symbolic gradient на Python. | Можно добавлять операции на Python с поддержкой встроенных операторов С |  |
| Многократное использование кода | Lasagne — высокоуровневая платформа на основе Theano. В Lasagne очень просто использовать заранее обученные модели Caffe | Нет поддержки заранее обученных моделей | В MXNET предусмотрен инструмент caffe\_converter, предназначенный для преобразования заранее обученных моделей caffe в формат MXNET |  |
| Операторы управления потоком | + | В формате эксперимента | - |  |
| Высокоуровневая поддержка | Можно создавать высокоуровневые платформы в соответствии с требованиями. Среди успешных примеров — Keras, Lasagne, blocks | не ориентирован исключительно на нейронные сети. Можно использовать коллекции графов, очереди и дополнения образов в качестве составных компонентов для высокоуровневых оболочек | Помимо символьной части, в MXNET также предусмотрены все необходимые компоненты для классификации образов, от загрузки данных до построения моделей с методами для начала обучения |  |
| Производительность | Отлично | Средне | Великолепно |  |
| Память | Отлично | Средне | Великолепно |  |

Таблица (см. реплан) TensorBoard, darknet, weight&bias

«На основе существующих решений выявлена потребность в разработке … » - 1 абзац

// написать подробнее, чем в введении

* 1. **План работ**
  2. обзор существующих методов решений.

Список литературы.

1. https://habr.com/ru/company/microsoft/blog/313318/