# Реализации нейронной сети «Inception V3» в задаче распознавания типов летательных аппаратов.

## Сокращения

|  |  |
| --- | --- |
| Сокращение | **Наименование** |
| НС | * Нейронная сеть |
| СНС | * Сверточная нейронная сеть |
| ЛА | * Летательный аппарат |
| РГО | * Распознавание графических образов |

## Введение

Создание многофункциональных локационных систем путем введения в их состав алгоритмических средств распознавания графических образов типов летательных аппаратов (ЛА) является актуальной задачей, успешное решение которой непосредственно влияет на обороноспособность страны. В системах военного назначения распознавание типов ЛА необходимо для построения плана тактических и стратегических мероприятий и выработки алгоритма противодействия воздушным целям, исходя из анализа складывающейся воздушной обстановки. В настоящее время в связи с интенсивной теоретической разработкой и практическим применением технологий, маскирующих локационные характеристики целей, роль задачи распознавания типов ЛА существенно возрастает.

Использование новейших технологий в области распознавания графических образом может существенно повысить точность решения задачи распознавания, уменьшить требуемое время наблюдения ЛА до полного решения задачи.

Обзор современных подходов к задаче РГО дает однозначный ответ о лучших подходах – нейронные сети дают лучшие результаты во всех тестированиях. Из всего семейства НС следует выделить семейство сверточных НС (СНС) […].

В то же время практически отсутствую упоминания о применении таких сетей в авиационной области.

## Цель

В данной работе автор ставит перед собой 2 основные цели. Первая цель: применить новейшие разработки в области СНС в задаче РГО, оценить результат и перспективность использования сложных СНС в подобных задачах. Препятствием к достижению этой цели является отсутствие (или недостаточность) обучающей выборки изображений ЛА разных типов. Действительно, лучшая СНС «Inception V3», состоит более чем 40 слоев, для обучения такой сети необходима выборка из тысяч (как будет показано ниже) изображений каждого типа ЛА. Составить такую размеченную коллекцию изображений (размеченная в том смысле что каждому изображению необходимо указать класс объекта который на нем изображен), само по себе является отдельной ресурсоемкой задачей. Данная задача не может быть решена автоматически. Таким образом мы приходим к идеи генерировать изображения для обучения СНС, вместо того что бы брать реальные. Теперь цели номер два может быть сформулирована следующим образом: Получить ответ на вопрос, можно ли использовать сгенерированные изображения для обучения СНС, в частности СНС «Inception V3».

## План работы

1. Генерация изображений ЛА нескольких разных типов.
2. Обучение (а точнее дообучение) СНС «Inception V3» задаче классификации изображений этих типов ЛА.
3. Оценка результата классификатора по контрольным РЕАЛЬНЫМ (не сгенерированным) изображениям.

## Описание стенда

Данная работа выполнена в операционной среде Ubuntu14.04, использован язык программирования python2.7. Библиотеки для работы с НС Theano, keras. Полностью код проекта доступен по ссылке [5]. Генерация изображений осуществлялась в редакторе «3ds Max 2016».

## СНС «Inception V3» и доработка в части распознавания типов ЛА

СНС «Inception V3» разработана группой специалистов из корпорации Google в 2015 году и показала лучший результат в международном конкурсе классификации изображений[9]. Полное описание этой СНС доступно по ссылке[1], в данной статье приведена общая послойная архитектура «Inception V3».

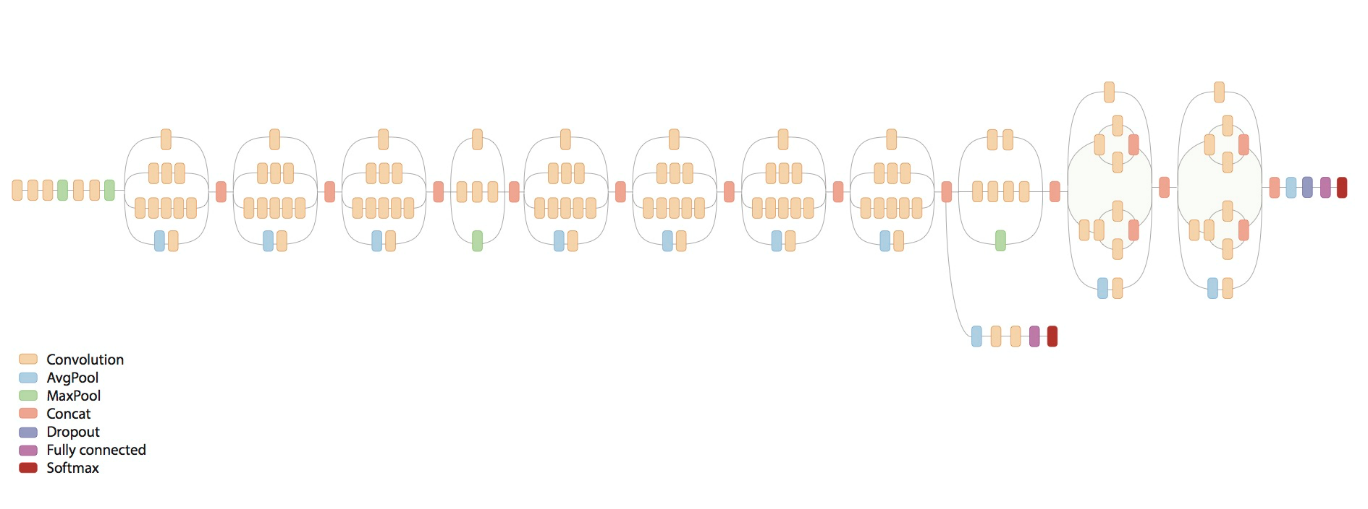


Рисунок 1. Архитектура нейронной сети

Обучение «Inception V3» само по себе является линейным процессом. В то же время количество нейронов делают эту задачу практически неподъёмной для домашних компьютеров. К счастью, существуют реализации этой НС уже обученные, т.е. дан вес каждого нейрона. Т.о. перед пользователями данной СНС не стоит задача проведения полного обучения, но стоит задача дообучения. Создатели «Inception V3» не предполагали использование данной сети в задаче распознавания различных типов ЛА. В случае если обученной «Inception V3» предложить классифицировать одно из изображение ЛА, представленных на рисунке Рисунок 2 то типовой ответ будет:

|  |  |
| --- | --- |
| Класс объекта | Вероятность присутствия объекта на изображении |
| «war aircraft» | 0.8 |
| «aircraft» | 0.05 |
| … | <0.01 |

Т.е. «Inception V3» имеет знание о классе «war aircraft», но не знает, на какие подклассы этот класс делиться. Использование «Inception V3» возможно двумя разными способами.

Первый и наиболее правильный – честное дообучение НС. Дообучение происходит путем подачи поочередно всех сгенерированные изображения ЛА на вход НС на выходе при этом каждому изображению соответствует метра класса. Этот способ довольно ресурсоемкий, для качественного обучения, нам необходимо…

Второй способ проще, не требует дообучения сети, и гораздо менее ресурсоемкий (насколько?) Дело в том что предпоследний слой «Inception V3» имеет размерность 1000х1 (т.е. вектор длинной 1000) и на последнем слое этот вектор преобразуется в метку класса объекта. Т.к. преобразование одного слоя линейно, то понятно что этот вектор НС до предпоследнего слоя выполнила работу по преобразованию входного изображения в вектор таким образом что бы все классы были линейно разделимы последним преобразованием (из предпоследнего слоя в последний). Т.о. мы можем использовать НС как инструмент отображение <изображение-вектор(1000)>. Далее в этом 1000-мерном пространстве для решения задачи классификации мы можем использовать Байесовский подход [ссылка] – построить функцию плотности вероятности и для каждого неклассифицированного типа ЛА считать его принадлежность к каждому классу использую эту функцию плотности. Возможно применение и других подходов, например, метода опорных векторов [ссылка].

## Генерация изображений для обучения НС

Для генерации исходных изображений для НС были использован редактор «3ds Max 2016». В качестве исходных моделей были использованы общедоступные модели [2, 3, 4]

Для обучения были использованы модели F15, F16, F117, Миг29 каждая модель ЛА была представлена в 3 различных вариантах раскраски. Для генерации изображений были использованы 3 разных фоновых изображения. Каждый ЛА снимался с различных углов с одинакового расстояния. При генерации изображения использовались фильтры для наложения шумов на изображения (параметры шумов – размытие, зернистость). Из всего множества возможных изображений случайным образом для каждого типа ЛА было выбрано 6000 изображений случайным образом. Эти изображения, дополненные меткой типа ЛА использовались в качестве обучающего множества. Типичная серия сгенерированных изображений показана на рисунке Рисунок 2. Все изображения сгенерированные и использованные в данной работе доступны[6].





Рисунок 2. Примеры серий сгенерированных изображений, использованные модели F16, F117

## Честное дообучение

….

## Использование предпоследнего слоя НС

Как мы писали выше возможно использование НС как инструмента отображения изображение в 1000-мерное пространство, в котором типы изображения различных типов ЛА должны быть хорошо линейно разделимы (вообще говоря не должны потому как нашу НС никто типам ЛА не учил, но обобщающая способность сети должна нас тут выручить).

В данной работе все сгенерированные изображения каждого типа ЛА были отображены в 1000-мерное пространство с помощью НС, далее к контрольным изображениям применялся один из подходов классификации

|  |  |  |
| --- | --- | --- |
|  | Ошибка классификации |  |
| Байесовский классификатор | … |  |
| Метод опорных векторов | … |  |
| GradienBoosting | … |  |

## Численный результат

Для анализа изображений были выбраны 120 общедоступных изображения ЛА типа 1 и типа 2. Изображения доступны и находятся в хранилище [4]. Все изображения были приведены к разрешения 300 пикселей по вертикали и горизонтали. Все изображения поочередно показывались «Inception V3», полученный усредненный результат представлен в таблице ниже.

|  |  |  |
| --- | --- | --- |
| Тип ЛА | F15 | F16 |
| F15 | 99 | 1 |
| F16 | 1 | 99 |

Таблица 1. Усредненные данные анализа изображений НС «Inception V3».

## Несколько соображений которые возникли у меня во время выполнения данной работы

…

Первое представленная выше техника легко обобщается на задачу распознавания изображения на видео – из видео-ряда необходимо сделать серию изображения, решить задачу классификации для каждого и далее, зная что на видео-кадрах представлен один и тот же объект воспользоваться теоремой о вероятности. Большое количество кадров даст нам высокую вероятность решения задачи классификации.

В рамках данной работы автор провел эксперимент с видео, доступным на youtube [12]. На видео снят полет самолета F16. На фоне городской застройки.

Результаты анализа видео.

… остальные соображения…

## Выводы

…

## Ссылки

1. Inception V3 main article. <https://arxiv.org/pdf/1512.00567v3.pdf>.
2. Модель f15 <http://tf3dm.com/3d-model/f15-aircraft-50507.html>
3. Модель f16 <https://www.cgstud.io/3d-model/-f-16e-block-60-8609?ref=tf3dm>
4. МодельF117
5. Модель Миг29
6. Git хранилище контрольных (проверочных изображений).
7. Хранилище исходного кода проекта.
8. Хранилище сгенерированных изображений ЛА.
9. Theano. <http://deeplearning.net/software/theano/install_windows.html>
10. Keras. <https://keras.io/>
11. Imagenet. <http://image-net.org/challenges/LSVRC/2016/index>
12. Видео f16. https://www.youtube.com/watch?v=aCmB3hz9UI0