

# Адаптация нейросети глубокого обучения к задаче распознавания типов летательных аппаратов.

## Сокращения

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

## Введение

Создание multifunctional локационных систем путем введения в их состав алгоритмических средств распознавания графических образов (РГО) различных типов летательных аппаратов (ЛА) является важной 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 РГО ЛА существенно возрастает. Решение задачи РГО ЛА так же может применяться для классификации «свой-чужой», и для предотвращения нанесения ударов по гражданским ЛА.

Интенсивное развитие вычислительной техники в последнее время и использование новейших технологий в области анализа графических образов может существенно повысить точность решения задачи классификации типов ЛА, уменьшить требуемое время наблюдения ЛА до решения задачи классификации с заданной точностью.

Обзор современных подходов к задаче РГО дает однозначный ответ о лучших методах решения задачи РГО. Во всех международных конкурсах по распознаванию и классификации изображений нейронные сети (НС) глубокого обучения дают наилучший результат [12]. Из всего семейства НС следует выделить семейство сверточных НС (СНС) [13]. В то же время практически отсутствуют упоминания о применении нейронных сетей в задачах РГО в авиационной области. Данная работа призвана несколько устранить этот пробел.

## Цель

В данной работе ставятся 2 основные цели. Первая цель: применить новейшие разработки в области СНС в задаче РГО ЛА, оценить результат и перспективность использования глубоких (многослойных) СНС в подобных задачах. Препятствием к достижению этой цели является отсутствие (недостаточность) обучающей выборки изображений ЛА разных типов. Действительно, СНС «ResNet», которая была выбрана в качестве базовой, при написании данной статьи состоит из 34 слоев, для обучения такой сети необходима выборка из, по крайней мере, тысяч, а лучше - десятков тысяч изображений каждого типа ЛА. Составление такой размеченной коллекции изображений (размеченная в том смысле что каждому изображению необходимо указать класс объекта, который на нем изображен или изображены), само по себе является отдельной ресурсоемкой задачей, которая остается нерешенной. Составление требуемой коллекции ручным образом, представляется подходом сомнительным: во-первых, ручная разметка всегда сопряжена

с риском допущения ошибок разметчиком, который (риск) по результатам последних исследований оценивается в 5% [16]. Во-вторых изображений может оказаться недостаточно для обучения СНС, это в первую очередь относится к перспективным, новейшим, ЛА, изображения которых и вовсе может быть засекречено, их страной-разработчиком. Таким образом мы приходим к идее генерировать необходимое множество изображений для обучения СНС, вместо того что бы брать реальные. Итак, вторая цель может быть сформулирована следующим образом: Выяснить, возможно ли использовать сгенерированные изображения ЛА (на основе трехмерной модели ЛА) для обучения СНС, в частности СНС «ResNet».

### 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 F15, F16 [2]. Каждая модель ЛА была представлена в 3 различных вариантах окраски. Окраски одинаковые для разных типов ЛА. Для генерации изображений были использованы 4 разных фоновых изображения. Каждый ЛА снимался с различных углов с одинакового расстояния. При генерации изображения использовались фильтры для наложения шумов на изображения, для получения «реалистичного» изображения ЛА. Из всего множества возможных изображений 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 1 . Все изображения сгенерированные и использованные в данной работе доступны [5].



Рисунок 1. Примеры серий сгенерированных изображений с помощью моделей F16, F15.

### Обзор СНС «ResNet»

СНС «ResNet» [1] разработана группой специалистов из лаборатории Microsoft Research в 2015 году и показала лучший результат в международном конкурсе классификации изображений [12]. В данной статье приведена общая послойная архитектура «ResNet-34».



|            |       |
|------------|-------|
| «aircraft» | 0.05  |
| ...        | <0.01 |

Таблица 1. Типовой ответ предобученной НС «ResNet» на запрос о классе ЛА (модель F15).

Т.е. «ResNet» имеет знание о классе «warplane» (боевой самолет), но не знает, на какие подклассы этот класс делиться. Т.о. прямое использование предобученной сети для нашей задачи классификации не представляется возможным.

### Исследование свойств предобученной НС для решения задачи классификации ЛА

Нам известно, что НС «ResNet» показывает высокие результаты на самых престижных соревнованиях по анализу изображений, будет логичным предположить, что даже без дообучения данная сеть может дать нам хорошее качество классификации. Т.к. НС не известны классы этих объектов, следовательно, мы не можем использовать выход НС для оценки качества классификации, но можем использовать предпоследний слой, который имеет размерность 1000. Т.к. преобразование однослойной сети линейно, то, ясно, что НС до предпоследнего слоя выполнила работу по преобразованию входного изображения в вектор признаков размерностью 1000, таким образом, что бы все классы были линейно разделимы последним преобразованием (из предпоследнего слоя в последний). Т.о. мы можем использовать НС как инструмент преобразования изображение в вектор размерностью 1000. Получив для каждого из изображений вектор, размерностью 1000, мы можем воспользоваться методом главных компонент [9, 10] для отображения 1000-мерного пространства в 2-х мерное для визуализации. Результат для 600 изображений ЛА (300 F15 и 300 F16) приведен на рисунке 3.

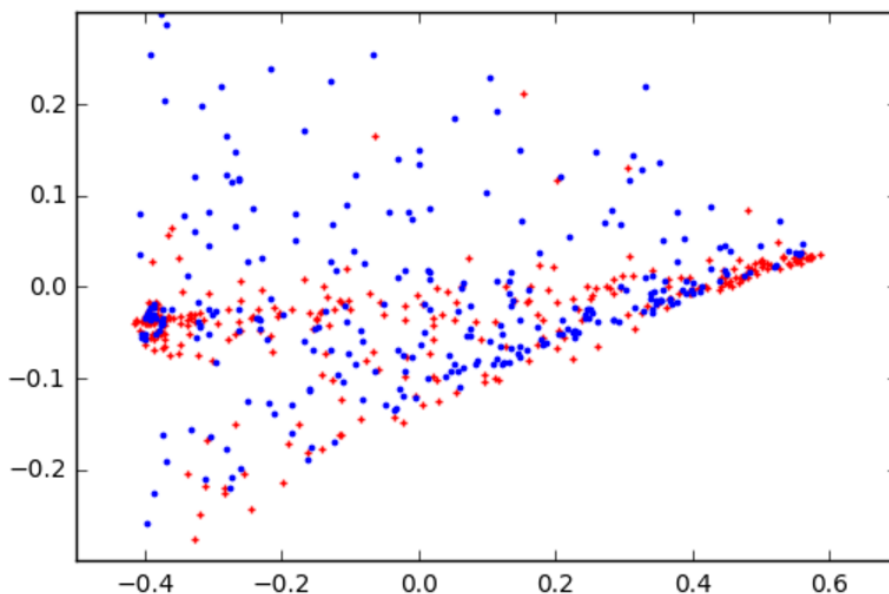


Рисунок 3. Проекция результатов выходного 1000-мерного слоя в 2-мерное пространство методом главных компонент. Красными ромбами обозначены выход для модели F16, синими точками – выход модели F15

Хотя из графика на рисунке 3 можно заметить некоторую корреляцию между типом ЛА и проекцией его изображения в двумерное пространство, все же использование последнего слоя «ResNet» не представляется перспективным, различить модели ЛА без некой доработки (адаптации) НС к задаче РГО типов ЛА не представляется возможным.

## Адаптация НС для решения задачи классификации типов ЛА

Адаптация НС «ResNet» состоит из двух задач: доработка архитектуры НС и дообучение НС (точная настройка весов нейронов для максимизации целевой функции решаемой задачи). Доработка архитектуры «ResNet» происходила следующим образом: был отрезан последний слой сети, и, вместо него, добавлен слой размерностью 2 (по числу исследованных в данной работе типов ЛА). Выход последнего слоя содержит вероятность принадлежности поданного на вход изображения соответствующему классу. Ответ выходного слоя (1,0) соответствует типу ЛА «F15» и (0,1) соответствует типу ЛА «F16». Такая форма выходного слоя была выбрана из-за ее неограниченной масштабируемости, при переходе к задаче мультиклассовой классификации мы просто увеличим количество нейронов в выходном слое. Процесс дообучения проходил в 2 этапа, на первом этапе были «заморожены» веса всех нейронов, кроме нейронов последнего, еще не обученного, добавленного нами, слоя. 80 эпох обучения позволили нам довольно точно найти значение нейронов, гарантирующее устойчивое решение задачи классификации, результаты оценки ошибки по эпохам приведены на рисунке 4. Каждая эпоха включала в себя обучение по 1000 случайно выбранным изображениям. При этом из каждого изображения (которые имеют размер 300x300 пикселей) брался случайный участок 224x224 пикселя и изображение поворачивалось на случайный угол (0-15°). Проверка (валидация) в каждой эпохе проводилась по 400 изображениям, которые так же выбирались случайно.



Рисунок 4. Снижение оценки ошибки по эпохам обучения последнего слоя

После обучения последнего слоя НС было выполнено дообучение всей сети целиком. Из-за высокой размерности задачи было выполнено всего 2 эпохи дообучения, каждая из которых включала в себя обучение на 10 000 изображений и проверка (валидация) на 1000 изображений. Результаты проведения полного обучения по эпохам представлены на рисунке 5.

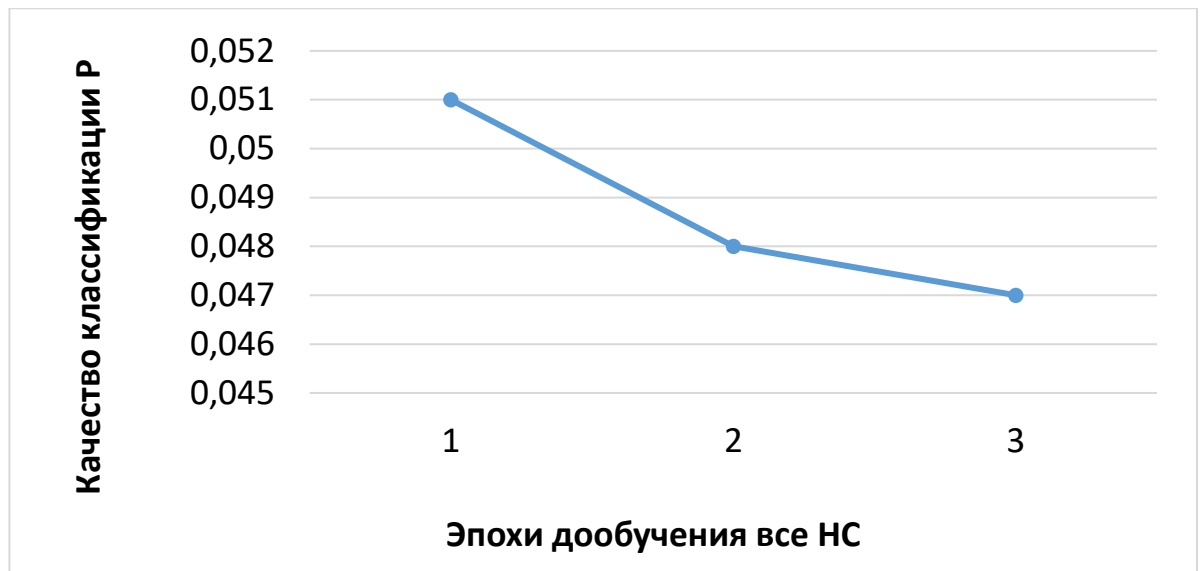


Рисунок 5. Снижение оценки ошибки по эпохам обучения всей сети

### Результаты численных исследований

После проведения дообучения и проведения валидации становится понятно, что НС получила способность различать сгенерированные искусственно изображения двух разных типов ЛА. Нас же интересует способность различать реальные изображения ЛА, для этого были взяты общедоступные фотографии F15, F16, типичные образцы которых приведены на рисунке 6. Всего в проверке участвовало 20 изображений F15 и 21 изображение F16. Результаты численных исследований на реальных изображениях приведены в таблице 2.



Рисунок 6. Проверочные (реальные) изображения ЛА

| Тип ЛА             | P(F15) | P(F16) |
|--------------------|--------|--------|
| 20 изображений F15 | 0.945  | 0.055  |
| 21 изображение F16 | 0.071  | 0.929  |

Таблица 2. Результаты численных исследований на реальных изображениях. В таблице представлено среднее значение вероятности принадлежности изображений ЛА F15, F16 к классу F15, F16.

Отметим, что качество решения просело ухудшилось примерно в два раза, по сравнению с оценкой, полученной на сгенерированных (искусственных) изображениях.

Для проведения обучения использовался домашний компьютер в конфигурации: Процессор: intel core i7-4700, 8 гигабайт оперативной памяти, видеокарта: NVidia GeForce GT 512 мегабайт. Полный цикл обучения (80 эпох обучения последнего слоя и 2 эпохи обучения всей сети) заняли 40 часов. Применение новых быстрых видеокарт (для задач обучения рекомендуется использовать графические процессоры из-за их высокой разрядности) позволило бы снизить это



время на порядок, к сожалению, при проведении исследований в распоряжении других ресурсов не было.

Заметим, что хоть обучение НС является задачей ресурсоемкой, решение задачи классификации для изображений задача очень простая, решается очень быстро. На домашнем компьютере мне удалось добиться скорости 0.3 секунды для классификации одного изображения. Так же мы знаем, что сети подобные «ResNet» легко «переживают» операцию «срезки» слабых связей [17], если мы хотим увеличить быстродействие сети, мы можем избавиться от 80% нейронов, потеряв менее 1% качества классификации.

## Выводы

Подводя итоги отметим, что обе поставленные в данной работе цели достигнуты, СНС применимы в задачах классификации типов ЛА по изображению. Результаты, полученные в данной работе кажутся довольно далеки от теоретического предела. Новейшие исследования СНС и прогресс в развитии вычислителей позволяет рассчитывать на решение этой задачи классификации в ближайшем будущем с вероятностью не менее 95% по одной фотографии. Если представить себе, что наблюдение за ЛА продолжается некоторое время и мы имеем несколько изображений, полученных независимо (во времени, пространстве) то воспользовавшись теоремой о сложении вероятностей и пронормировав результат мы разом можем улучшить вероятность правильной классификации с 0.9 до 0.98, используя два изображения.

Что же касается возможности применения сгенерированных данных для обучения, то и тут полученный результат можно считать удачным, обученная НС показывает приемлемый результат классификации реальных изображений (которые для обучения не использовались вовсе). В то же время необходимо отметить «проседание» результата при переходе к анализу реальных изображений. Для дальнейшего совершенствования НС представляется перспективным – наращивание искусственных данных для обучения (количество различных окрасов ЛА, типов фона, генерация изображений с различными типами боевой подвески и др.).

Стоит отметить что применяемая в данной работе СНС, так же, как и все самые лучшие СНС для задач классификации (Inception – GoogleLabs [14] (США), VGG19 – Oxford Visual Geometry Group [15] (Великобритания)) разработана за рубежом.

**Применение: попробовать на аэрокосмическом снимке!!!!**

**Добавить – обучение долгое, решить классификацию – мгновенно, за счет сокращения кол-ва нейронов**

**Обучение – мера**

## Ссылки

1. ResNet. Главная статья <https://arxiv.org/pdf/1512.00567v3.pdf>.
2. Модели F15, F16 <https://github.com/piero10/AircraftsRecognition/tree/master/models>
3. Реальные - проверочные изображений ЛА.  
<https://github.com/piero10/AircraftsRecognition/tree/master/images/realImages>
4. Хранилище исходного кода проекта.  
<https://github.com/piero10/AircraftsRecognition/tree/master>
5. Хранилище сгенерированных изображений ЛА.  
<https://github.com/piero10/AircraftsRecognition/tree/master/images/modelImages>

6. Библиотека низкого уровня для работы с НС - Theano. [http://deeplearning.net/software/theano/install\\_windows.html](http://deeplearning.net/software/theano/install_windows.html)
7. Библиотека высокого уровня для работы с НС - Keras. <https://keras.io/>
8. Соревнование по классификации изображений Imagenet. <http://image-net.org/challenges/LSVRC/2016/index>
9. Метод главных компонент. <http://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html>
10. Метод главных компонент. <http://setosa.io/ev/principal-component-analysis/>
11. Реализация сверточной нейронной сети Resnet34 <https://github.com/KaimingHe/deep-residual-networks>
12. ResNet first place in classification task <http://image-net.org/challenges/LSVRC/2015/>
13. Сверточная НС <http://cs231n.github.io/convolutional-networks/>
14. GoogleLabs. Inception. <https://research.googleblog.com/2015/06/inceptionism-going-deeper-into-neural.html>
15. Oxford. Visual Geometry Group. <http://www.robots.ox.ac.uk/~vgg/>
16. Ошибка ручной разметки <http://karpathy.github.io/2014/09/02/what-i-learned-from-competing-against-a-convnet-on-imagenet/>
17. Упрощение НС. <https://arxiv.org/pdf/1611.06211.pdf>