Обзор свёрточных нейронных сетей для задачи классификации изображений

Сикорский О.С., МГТУ им. Н.Э. Баумана
olegsik@gmail.com

Аннотация

В данной статье рассмотрена задача классификации изображений и дано краткое описание структуры свёрточных нейронных сетей. Проведён обзор свёрточных нейронных сетей для задачи классификации изображений и сделано сравнение их точности на примере аннотированной базы изображений ImageNet.

1 Введение

Технологии компьютерного зрения очень распространены. Они применяются для распознавания лиц, пешеходов, объектов, для медицинского анализа, навигации автономных автомобилей и во многих других сферах. В связи с ростом вычислительных мощностей и появлением больших баз изображений стало возможным обучать глубокие нейронные сети — нейронные сети с большим числом скрытых слоёв. В задаче распознавания образов особого успеха достигли свёрточные нейронные сети (Convolutional Neural Networks), которые каждый год с 2012 года выигрывали соревнование ImageNet Large Scale Visual Classification Challenge (ILSVRC) [Russakovsky et al., 2015].

Целью данной статьи является обзор архитектур свёрточных нейронных сетей для классификации изображений.

2 Задача классификации изображения

Одной из базовых задач в машинном зрении является задача классификации изображения — определения категорий объектов, который находится на изображении. В зависимости от конкретной задачи, на изображении может быть аннотирован как один объект, так и несколько.

Для оценки алгоритмов машинного обучения обычно используются аннотированные базы изображений, например, CIFAR-10 [Krizhevsky, 2009], ImageNet [Russakovsky et al., 2015], PASCAL VOC [Everingham et al., 2010].

Из-за того, что на изображениях в базе ImageNet может присутствовать несколько объектов, и лишь один из них аннотирован, в ImageNet основной оценкой ошибки является top-5 ошибка. При её использовании считается, что алгоритм не ошибся, если правильная категория объекта находится среди пяти категорий, выданных алгоритмом как наиболее вероятные. Вследствие этого многие нейронные сети для задачи классификации оцениваются именно с помощью top-5 ошибки.

3 Свёрточные нейронные сети

Свёрточная нейронная сеть — нейронная сеть, в которой присутствует слой свёртки (convolutional layer). Обычно в свёрточных нейронных сетях также присутствуют слой субдискретизации (pooling layer) и полносвязный слой (fully connected layer). Свёрточные нейронные сети применяются для оптического распознавания образов [LeCun et al., 1998], классификации

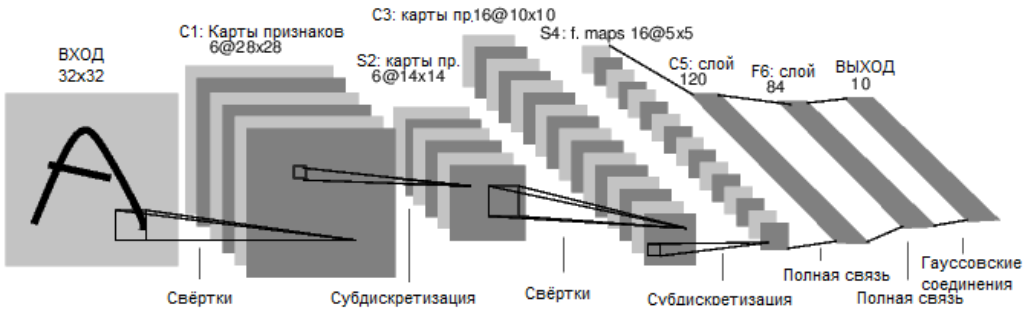

Рис. 1. LeNet-5

изображений [Russakovsky et al., 2015], детектирования предметов [Girshick et al., 2014], семантической сегментации [Long et al., 2015] и других задач [Gu et al., 2017].

Основы современной архитектуры свёрточных нейронных сетей были заложены в одной из первой широко известной свёрточной нейронной сети — LeNet-5 Яна ЛеКуна [LeCun et al., 1998], архитектура которой представлена на рисунке 1.

В свёрточных нейронных сетях слои свёртки и субдискретизации состоят из нескольких «уровней» нейронов, называемых картами признаков (feature maps), или каналами (channels). Каждый нейрон такого слоя соединён с небольшим участком предыдущего слоя, называемым рецептивным полем. В случае изображения, карта признаков является двумерным массивом нейронов, или просто матрицей. Другие измерения могут быть использованы, если на вход принимается другой вид данных, например, аудио данные (одномерный массив) или объёмные данные (трёхмерный массив).

В слое свёртки каждой карте признаков соответствует одно ядро свёртки, также называемое фильтром. Каждый нейрон в качестве своего выходного значения осуществляет операцию свёртки или взаимной корреляции со своим рецептивным слоем.

Стоит заметить, что эти две операции в контексте обучения свёрточных нейронных сетей взаимозаменяемы, вследствие чего во многих программных реализациях операция “свёртки” на самом деле является операцией взаимной корреляции.

Так как ядро свёртки для каждой карты признаков одно, это позволяет нейронной сети научиться выделять признаки вне зависимости от их расположения во входном изображении и также приводит к значительному уменьшению числа параметров.

Согласно устоявшейся нотации, говорят, что слой свёртки использует фильтр $W \times H$, если каждый фильтр в этом слое имеет размерность $W \times H \times C$, где C — число каналов в предыдущем слое.

Слой субдискретизации осуществляет уплотнение карт признаков предыдущего слоя и не изменяет количество карт. Каждая карта признаков слоя соединена с соответствующей картой признаков предыдущего слоя, каждый нейрон выполняет «сжатие» своего рецептивного поля посредством какой-либо функции.

Наиболее популярными видами этого слоя являются Max Pooling (из рецептивного слоя выбирается максимальное значение), Average Pooling (выбирается среднее значение) и L2 Pooling (выбирается норма L2) [Li et al., 2016]. С помощью слоя субдискретизации достигается устойчивость к небольшим сдвигам входного изображения, а также уменьшается размерность последующих слоёв [Goodfellow et al., 2016].

Полносвязный слой — обычный скрытый слой многослойного перцептрона, соединённый со всеми нейронами предыдущего слоя.

Таким образом, на вход свёрточной нейронной сети подаётся изображения, а на выходе получается класс, к которому принадлежит изобрежание.

4 Свёрточные нейронные сети для классификации изображеия

В данном разделе будут описаны архитектуры свёрточных нейронных сетей для задачи классификации изображения.

4.1 AlexNet

В 2012 году на конкурсе ILSVRC по классификации изображений впервые победила нейронная сеть — AlexNet [Krizhevsky, 2009], достигнув top-5 ошибки 15,31 % [Russakovsky et al., 2015]. Для сравнения, метод, не использующий свёрточные нейронные сети, получил ошибку 26,1 %. В AlexNet были собраны новейшие на тот момент техники для улучшения работы сети.

Обучение AlexNet из-за количества параметров сети происходило на двух GPU, что позволило сократить время обучения в сравнении с обучением на CPU. Также оказалось, что использование функции активации ReLU (Rectified Linear Unit) вместо более традиционных функций сигмоиды и гиперболического тангенса позволило снизить количество эпох обучения в шесть раз.

Формула ReLU следующая:

$$y(x) = \max(0, x).$$

ReLU позволяет побороть проблему затухания градиентов, свойственную другим функциям активации.

Помимо прочего, в AlexNet была применена техника отсева (Dropout) [Hinton et al., 2012]. Она заключается в случайном отключении каждого нейрона на заданном слое с вероятностью p на каждой эпохе. После обучения сети, на стадии распознавания, веса слоёв, к которым был применён dropout, должны быть умножены на 1/p. Dropout выступает в роли регуляризатора, не позволяя сети переобучаться.

Для объяснения эффективности данной техники существует несколько интерпретаций. Первая заключается в том, что dropout заставляет нейроны не полагаться на соседние нейроны, а обучаться распознавать более стойкие признаки. Вторая, более поздняя, состоит в том, что, обучение сети с dropout представляет собой аппроксимацию обучения ансамбля сетей, каждая из которых представляет сеть без некоторых нейронов [Srivastava et al., 2014]. Таким образом, конечное решение принимает не одна сеть, а ансамбль, каждая сеть которого обучена по-разному, тем самым снижается вероятность ошибки.

4.2 ZF Net

ZF Net — победитель ILSVRC 2013 с top-5 ошибкой 11,2 % [Zeiler, Fergus, 2014]. Основным достижением данной архитектуры является создание техники визуализации фильтров — сети развёртки (deconvolutional network), состоящей из операций, в каком-то смысле обратных операциям сети. В итоге сеть развёртки отображает скрытый слой сети на оригинальное изображение.

Чтобы изучить поведение фильтра на определённом изображении с помощью обученной нейронной сети, необходимо сначала осуществить вывод сетью, после чего в слое изучаемого фильтра обнулить все веса, кроме весов самого фильтра, и затем подать полученную активацию на слой сети развёртки. В сети развёртки последовательно применяются операции Unpooling, ReLU и фильтрации. Unpooling частично восстанавливает вход соответствующего слоя субдискретизации, запоминая координаты, которые выбрал слой субдискретизации. ReLU — обычный слой, применяющий функцию ReLU. Слой фильтрации выполняет операцию свёртки с весами соответствующего слоя свёртки, но веса каждого фильтра «перевёрнуты» вертикально и горизонтально. Таким образом, исходная активация фильтра движется в обратном направлении, пока не будет отображена в оригинальном пространстве изображения.

На рисунке 3 представлены некоторые фильтры сети и части изображения, наиболее их возбуждающие. Как можно увидеть, чем выше уровень фильтра, тем более сложные признаки он выделяет.

4.3 VGG Net

VGG Net — модель свёрточной нейронной сети, предложенная в [Simonyan, Zisserman, 2014]. В данной сети отказались от использования фильтров размером больее, чем 3x3.

Авторы показали, что слой с фильтром 7x7 эквивалентен трём слоям с фильтрами 3x3, причём в последнем случае используется на 55 % меньше параметров.

Аналогично слой с фильтром 5x5 эквивалентен двум слоям с фильтром 3x3, которые экономят 22 % параметров сети. Визуальное представление такой декомпозиции можно увидеть на рисунке 2.

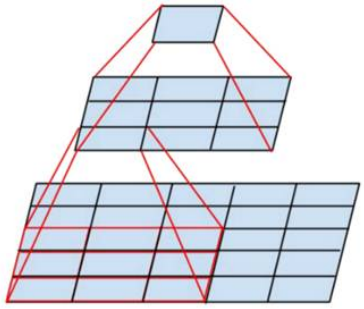

Рис. 2. Декомпозиция фильтра 5x5

На соревновании ILSVRC 2014 ансамбль из двух VGG Net получил top-5 ошибку 7,3 %. Хотя данная модель и не победила в соревновании, из-за её простоты она используется в более сложных сетях, предназначенных для детектирования предметов [Girshick, 2015; Ren et al., 2015], семантической сегментации [Noh et al., 2015] или маскирования объектов [Pinheiro et al., 2015].

4.4 Inception

Inveption-v1 — победитель ILSVRC 2014 с top-5 ошибкой 6,7 % [Szegedy et al., 2015], также известный как GoogLeNet. Создатели этой сети во главе с Christian Szegedy исходили из факта, что после каждого слоя сети необходимо сделать выбор — будет ли следующий слой свёрткой с фильтром 3x3, 5x5, 1x1 или же слоем субдискретизации.

Каждый из таких слоёв полезен — фильтр 1x1 выявляет корреляцию между каналами, в то время как фильтры большего размера реагируют на более глобальные признаки, а слой субдискретизации позволяет уменьшить размерность без больших потерь информации.

Вместо того чтобы выбирать, какой именно слой должен быть следующим, предлагается использовать все слои сразу, параллельно друг другу, а затем объединить полученные результаты в один. Чтобы избежать роста числа параметров, перед каждым слоем свёртки используется свёртка 1x1, которая уменьшает число карт признаков. Такой блок слоёв назвали модулем Inception. Он представлен на рисунке 3.

Также в GoogLeNet отказались от использования полносвязного слоя в конце сети, используя вместо него слой Average Pooling, благодаря чему резко уменьшилось число параметров в сети. Таким образом, GoogLeNet, состоящая из более чем ста базовых слоёв, имеет почти в 12 раз меньше параметров, чем AlexNet (около 7 миллионов параметров против 138 миллионов).

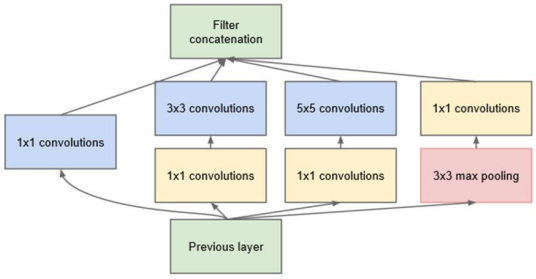

Рис. 3. Модуль Inception

4.5 Inception-v2 и Inception-v3

В следующей итерации модуля Inception, названной Inception-v2 [Szegedy et al., 2016], авторы, как было сделано в сети VGG, декомпозировали слой с фильтром 5x5 на два слоя 3x3. Далее была использована техника Batch Normalization [Ioffe, Szegedy, 2015], позволяющая многократно увеличить скорость обучения за счёт нормализации распределения выходов слоёв внутри сети.

В той же статье авторы предложили Inception-v3. В данной модели они развили идею декомпозиции фильтров, предложив декомпозировать фильтра NxN двумя последовательными фильтрами 1xN и Nx1.

Также в Inception-v3 используется RMSProp [Hinton, Srivasta, Swersky, 2012] вместо стандартного градиентного спуска и используется усечение градиентов [Pascanu et al., 2013] для повышения стабильности обучения. Ансамбль из четырёх Inception-v3 получил top-5 ошибку 3,58 % на ILSVRC 2015, уступив первенство ResNet.

4.6 ResNet

Победителем ILSVRC 2015 с top-5 ошибкой в 3,57 % стал ансамбль из шести сетей типа ResNet (Residual Network), разработанный в Microsoft Research [He et al., 2016].

Авторы ResNet заметили, что с повышением числа слоёв свёрточная нейронная сеть может начать деградировать — у неё понижается точность на валидационном множестве. Так как падает точность и на тренировочном множестве, можно сделать вывод, что проблема состоит не в переобучении сети.

Было сделано предположение, что если свёрточная нейронная сеть достигла своего предела точности на некотором слое, то все следующие слои должны будут выродиться в тождественное преобразование, но из-за сложности обучения глубоких сетей этого не происходит. Для того чтобы «помочь» сети, было предложено ввести пропускающие соединения (Shortcut Connections), изображённые на рисунке 4.

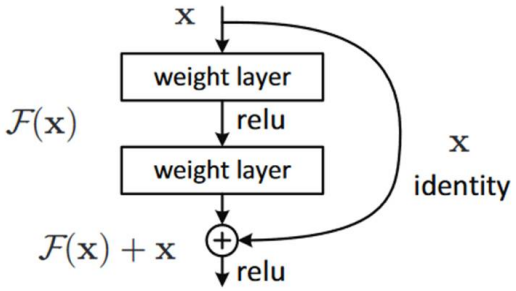

Рис. 4. Пропускающее соединение

Пускай оригинальная сеть должна вычислять функцию $H(X)$. Определим её остаточную функцию как $F(X) = H(x) - x$, которая, в теории, должна быть проще обучаема сетью. Добавив пропускающие соединения, как показано на рисунке 5, сеть учится остаточной функции, которая затем складывается с тождественным преобразованием.

Анализ в [Veit et al., 2016; Wu et al., 2016] показал, что глубокие остаточные нейронные сети можно считать ансамблем, состоящим из более мелких остаточных нейронных сетей,

чья эффективная глубина увеличивается в процессе обучения.

4.7 Inception-v4 и Inception-ResNet

После успеха ResNet, в [Szegedy, Ioffe et al., 2016] были представлены следующие версии сети Inception: Inception-v4 и Inception-ResNet. В обоих вариантах модуль Inception был разбит на модули A, B и C для входов размерностью 35x35, 17x17 и 8x8 соответственно. Также были выделены блоки редукции, в которых происходит понижение размерности и увеличение глубины данных внутри сети.

В Inception-v4 главными нововведениями являются замена Max Pooling на Average Pooling в самих модулях Inception.

Для Inception-ResNet в модули Inception были добавлены пропускающие соединения. Были сконструированы две версии сети — Inception-ResNet-v1, для которой требуется меньше вычислений, и Inception-ResNet-v2. В качестве примера на рисунке 5 представлен модуль Inception-ResNet-C для Inception-ResNet-v1.

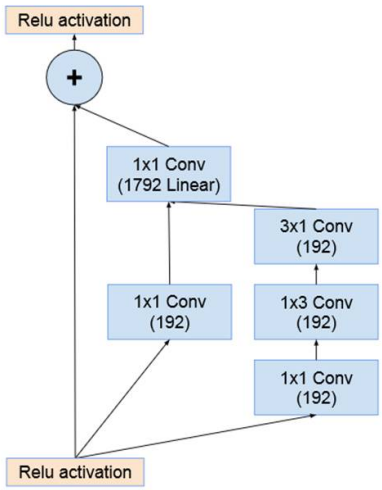

Рисунок 5. Модуль Inception-ResNet-C для Inception-ResNet-v1

5 Сравнение моделей

Для оценки моделей свёрточных нейронных сетей помимо вида ошибки обычно указывают количество моделей в ансамбле и количество вырезов изображения, которые подавались на вход каждой модели.

Например, 10 вырезов означает, что сделано четыре выреза по углам изображения, один вырез в центре, и каждый вырез дополнительно горизонтально перевёрнут.

В таблице 1 представлены результаты рассмотренных нейронных сетей с одной моделью и с одним вырезом на базе изображений ImageNet (кроме ResNet-151, для которой указан результат для 10 вырезов).

В таблице 2 представлены результаты использования ансамблей моделей со многими вырезами на базе изображений ImageNet.

Как видно из данных таблиц, за пять лет, с 2012 по 2016 годы, top-5 ошибка на ImageNet для одиночных моделей уменьшилась почти в четыре раза (с 17 % до 4,49 %), а для ансамбля — почти в пять раз (с 15,40 % до 3,10 %).

Таблица 1. Результаты для одной модели с одним вырезом

Нейронная сеть	Top-1	Top-5
AlexNet	39,00 %	17 %
ZF Net	37,50 %	16 %
VGG Net	25,60 %	8,10 %
GoogLeNet	29,00 %	9,20 %
Inception-v3	21,20 %	5,60 %
Inception-v4	20,00 %	5 %
Inception-ResNet-v2	19,90 %	4,90 %
ResNet-151	19,38 %	4,49 %

Таблица 1. Результаты для ансамблей, со многими вырезами

Нейронная сеть	Модели	Вырезы	Top-1	Top-5
AlexNet	7	1	36,70 %	15,31 %
ZF Net	6	10	36 %	14,70 %
VGG Net	2	150	23,70 %	6,80 %
GoogLeNet	7	144	—	6,67 %
Inception-v3	4	144	17,20 %	3,58 %
ResNet-151	6	144	—	3,57 %
Inception-v4 + 3x Inception-ResNet	4	144	16,50 %	3,10 %

Заключение

В данной работе были описаны одни из самых значимых архитектур свёрточных нейронных сетей для задачи классификации изображений. Они позволили сильно улучшить точность распознавания изображений и достичь результатов, не достигнутых классическими методами компьютерного зрения.

Список литературы

Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn, Andrew Zisserman. 2010. The pascal visual object classes (voc) challenge. *International journal of computer vision 88(2)*, pp. 303-338.

Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik. 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. *The IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580-587

Ross Girshick. 2015. Fast R-CNN. *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1440-1448.

Ian Goodfellow, Yoshua Bengio, Aaron Courville. 2016. Deep Learning. MIT Press, Massachusetts, US.

Jiuxiang Gu, Zhenhua Wang, Jason Kuen, Lianyang Ma, Amir Shahroudy, Bing Shuai, Ting Liu, Xingxing Wang, Gang Wang. 2017. Recent Advances in Convolutional Neural Networks. *arXiv preprint* arXiv:1512.07108

Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. 2016. Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770-778

Geoffrey Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, Ruslan R Salakhutdinov. 2012. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint* arXiv:1207.0580

Geoffrey Hinton, Nitish Srivasta, Kevin Swersky. 2012. "Lecture 6a Overview of Mini–Batch Gradient Descent." www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf. Accessed 21 Mar. 2017.

Sergey Ioffe, Christian Szegedy. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint* arXiv:1502.03167.

Alex Krizhevsky. 2009. Learning multiple layers of features from tiny images. URL: https://www.cs.toronto.edu/~kriz/cifar.html Accessed: 21 Mar. 2017

Yann LeCun, Leon Bottou, Yoshua Bengio, Patrick Haffner. 1998. Gradient-based learning applied to document recognition. *Proceedings of IEEE, 86(11)*, pp. 2278-2324

Fei-Fei Li, Andrej Karpathy, Justin Johnson. 2016. CS231n Convolutional Neural Networks for Visual Recognition. URL: https://cs231n.github.io/convolutional-networks. Accessed: 21 Mar. 2017

Jonathan Long, Evan Shelhamer, Trevor Darrell. 2015. Fully convolutional networks for semantic segmentation. *The IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3431-3440

Hyeonwoo Noh, Seunghoon Hong, Bohyung Han. 2015. Learning deconvolution network for semantic segmentation. *Proceedings of the IEEE International Conference on Computer Vision,* pp. 1520-1528

Razvan Pascanu, Tomas Mikolov, Yoshua Bengio. 2013. On the difficulty of training recurrent neural networks. *ICML* 28(3), pp. 1310-1318.

Pedro O. Pinheiro, Ronan Collobert, Piotr Dollar. 2015. Learning to segment object candidates. *Advances in Neural Information Processing Systems*, pp. 1990-1998

Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun. 2015. Faster R-CNN: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, pp. 91-99

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, Li Fei-Fei. 2015. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, *115*(3), pp. 211-252.

Karen Simonyan, Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint* arXiv:1409.1556.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1), pp. 211-252.

Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, Alex Alemi. 2016. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. *arXiv preprint* arXiv:1602.07261

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, Andrew Rabinovich. 2015. Going deeper with convolutions. *The IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1-9

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, Zbigniew Wojna. 2016. Rethinking the Inception Architecture for Computer Vision. *The IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2818-2826

Andreas Veit, Michael J. Wilber, Serge Belongie. 2016. Residual networks behave like ensembles of relatively shallow networks. *Advances in Neural Information Processing Systems,* pp.550-558

Zifeng Wu, Chunhua Shen, Anton van den Hengel. 2016. Wider or Deeper: Revisiting the ResNet Model for Visual Recognition *arXiv preprint* arXiv:1611.10080

Matthew D Zeiler, Rob Fergus. 2014. Visualizing and understanding convolutional networks. *European conference on computer vision*, pp. 818-833. Springer International Publishing.