**VGG 논문 리뷰**

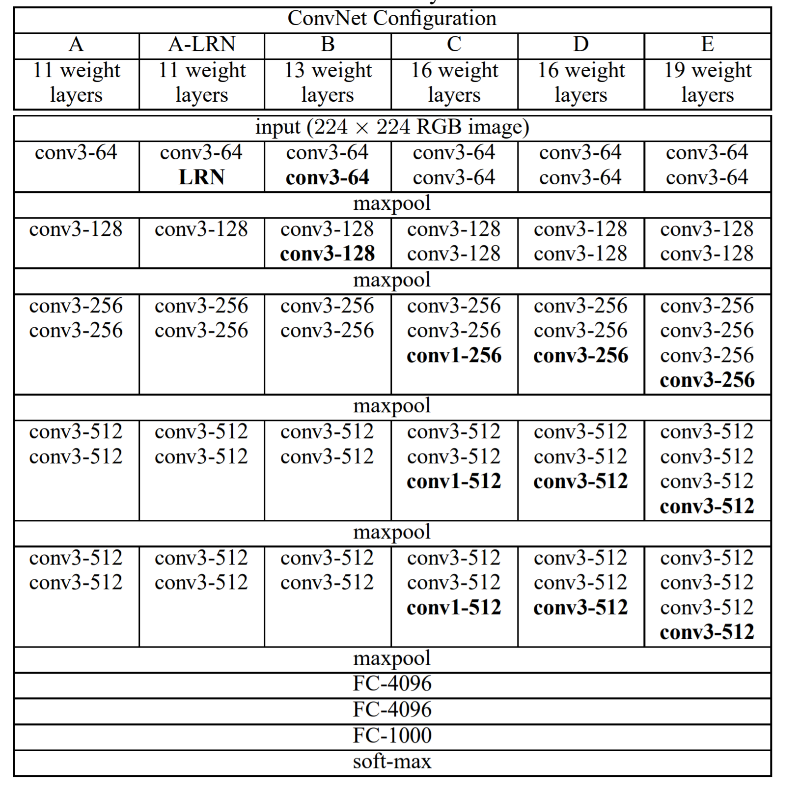
< [Very Deep Convolutional Networks For Large-Scale Image Recognition](https://arxiv.org/pdf/1409.1556.pdf%20http:/arxiv.org/abs/1409.1556.pdf) >

1. **개요**

VGG는 ImageNet Challenge 2014에서 Top-5 테스트 정확도를 92.7%까지 끌어 올렸다. 우승작인 GoogleNet과 0.1%의 근소한 차이로 2위에 입상했다. 하지만 GoogleNet보다 굉장히 단순한 구조를 가지고 있었고 이것은 신경망의 깊이가 정확도에 영향을 준다는 것을 잘 보여주었다.

1. **구조**

VGG모델은 신경망의 깊이가 딥러닝 정확도에 영향을 주는 지를 확인하기 위해서 11 -layer 부터 19 - layer까지 총 5개의 모델을 준비했다. 여기서 다룰 모델을 VGG - 16으로 16개의 layer를 가진 모델이다.



간략한 표현을 위해 Activation function을 생략함

1. Input

위의 표를 보면 알 수 있겠지만 입력 이미지의 크기는 224 x 224x 3 으로 고정이다.

1. Convolutional layer (conv layer)

Conv3 - 64가 의미하는 것은 3x3의 kernel size를 사용했고 64개의 filter를 가졌다는 의미이다. 여기서 3x3의 작은 kernel size를 사용한 이유는 3x3이 상하좌우, 중앙을 표현할 수 있는 가장 작은 단위이기 때문이다.

첫번째 conv layer 의 Filter개수는 64개이고Filter의 개수가 512에 도달할 때 까지 MaxPooling layer이후에 2배씩 계속 증가시켜준다.

Padding을 적용하여image size을 유지한다.

* Kernel size = 3 x 3
* Stride = 1
* Padding = same
* Activation Function = ReLu

1. MaxPooling2D

conv layer 이후 Maxpooling을 이용하여 출력의 크기를 절반으로 줄여준다.

이렇게 해주는 이유는 Feature map 에 있는 작고 사소한 특징이 출력에 큰 영향을 미치지 못하도록 하기 위함이다.

* Window size = 2 x 2
* Stride = 2

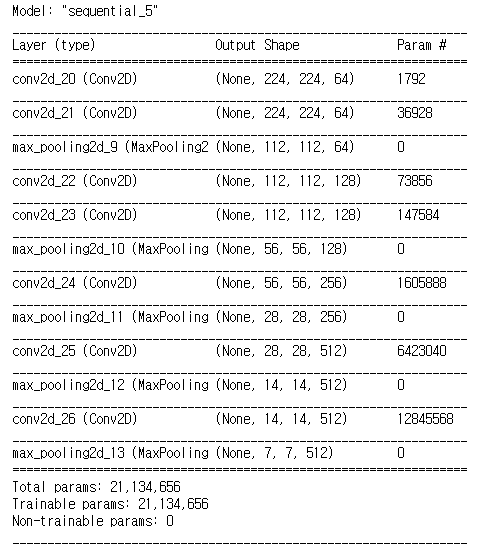
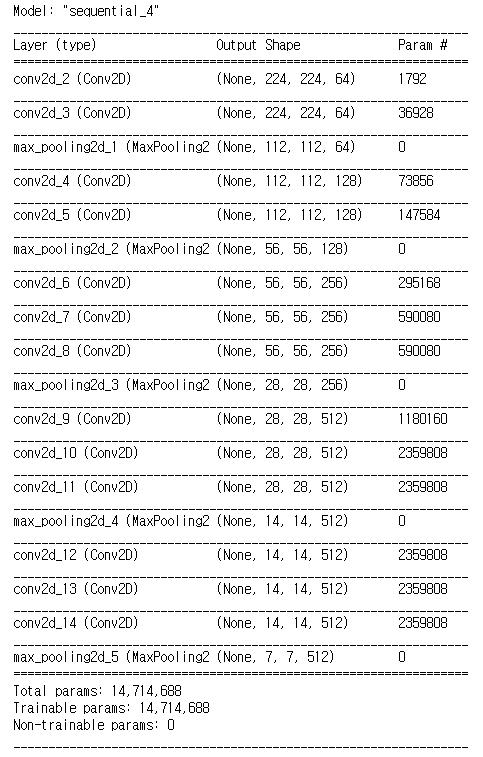
1. Fully Connected(FC) layer

1000개의 클래스를 분류하기 위해 3번째 FC layer에는 1000개의 channel을 가진다.

출력이 각 클래스에 속할 확률이어야 하므로soft-max function을 이용한다.

* First Two layers = 4096 channels each
* Third layers = 1000 channels
* Last layer = soft-max function

1. **7 x 7 conv layer 대신에 3 x 3 conv layer를 이용한 이유**



< Three 3 x 3 conv layer >

< Single 7 x 7 conv layer >

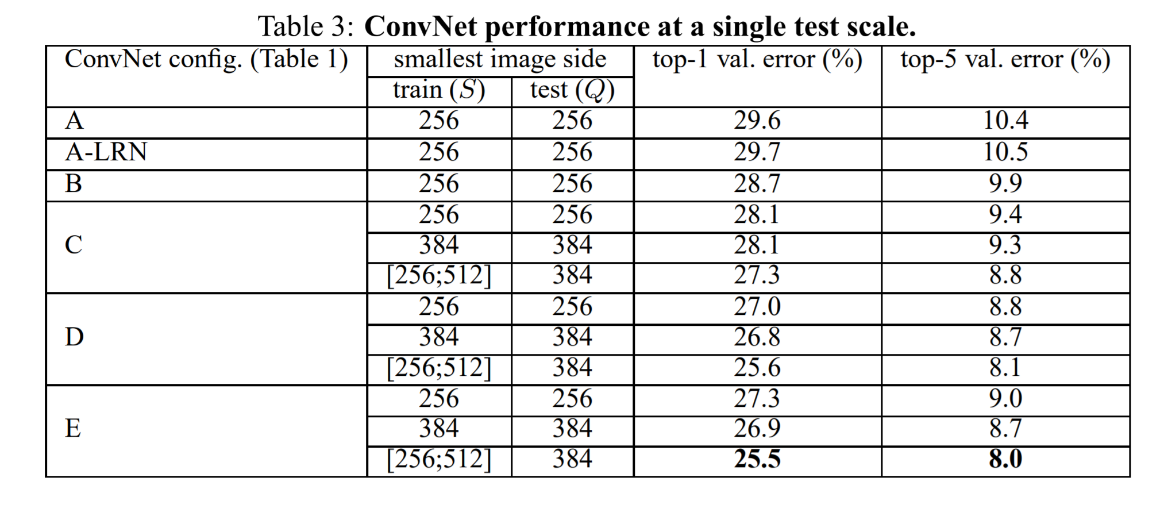
1. 위 표처럼 3개의 3 x 3 conv layer를 사용하면 7 x 7 conv layer 1개를 사용했을 때 보다 파라미터의 개수가 줄어 드는 것을 볼 수 있다. 파라미터 개수의 감소는 연산량 감소로 이어지므로 훈련 속도에 영향을 준다.
2. Single 7 x 7 conv layer를 사용했을 때의 feature map과 3개의 3 x 3 conv layer를 사용했을 때의 feature map의 크기가 동일하다. 예를 들어 10 x 10 image가 있다고 가정한다면 7 x 7 filter size(=receptive field)를 적용하면 4 x 4가 출력으로 나온다. 3 x 3 filter size를 3번 적용하면 8x8 -> 6x6 -> 4x4로 7 x 7 filter size와 동일하다.
3. 3번의 activation function을 사용하므로 1번 사용하는 것보다 비선형성이 높아지고 이것은 특징을 잘 식별할 수 있게한다.
4. **훈련**

* Optimizing the multinomial logistic regression
* Mini-batch gradient descent
* Momentum (0.9)
* L2 norm
* Dropout (0.5)
* Learning rate (0.01

Learning rate의 경우 validation accuracy의 증가가 멈출 때까지 10배씩 계속 감소시킨다.

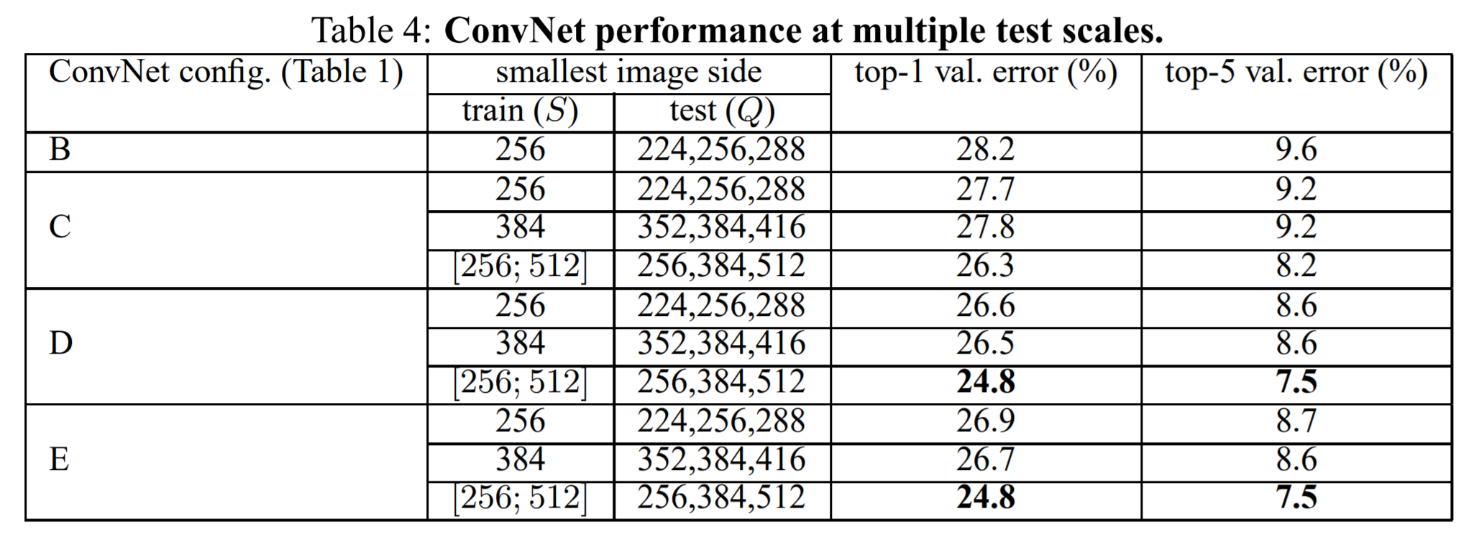
1. 훈련 전 이미지를 S의 크기로 rescale 한 후 224x224크리고 잘라낸다. 여기서 S는 하나의 고정된 값(Single scale)일 수도 있고 범위를 가진 값(Multi scale)일 수도 있다.
2. 입력 이미지를 잘라서 데이터 개수를 늘리므로 이것은 Data Augmentation으로 볼 수 있다.
3. 먼저 층이 얕은 모델 A를 적당히 훈련시킨 후 처음 4개의 conv layer와 마지막 3개의 FC layer를 다음에 학습할 네트워크의 초기값으로 활용했다. 모델A를 훈련 시킬 때 랜덤 초기화를 진행했는데 weight는 평균0, 분산은 0.01, 편향은 0으로 초기화 시켰다.
4. **테스트**

테스트 전 테스트 이미지를 Q의 크기로 rescale한다 S와 의미는 같지만 Q는 테스트에서만 사용하고 S는 훈련에서만 사용한다.

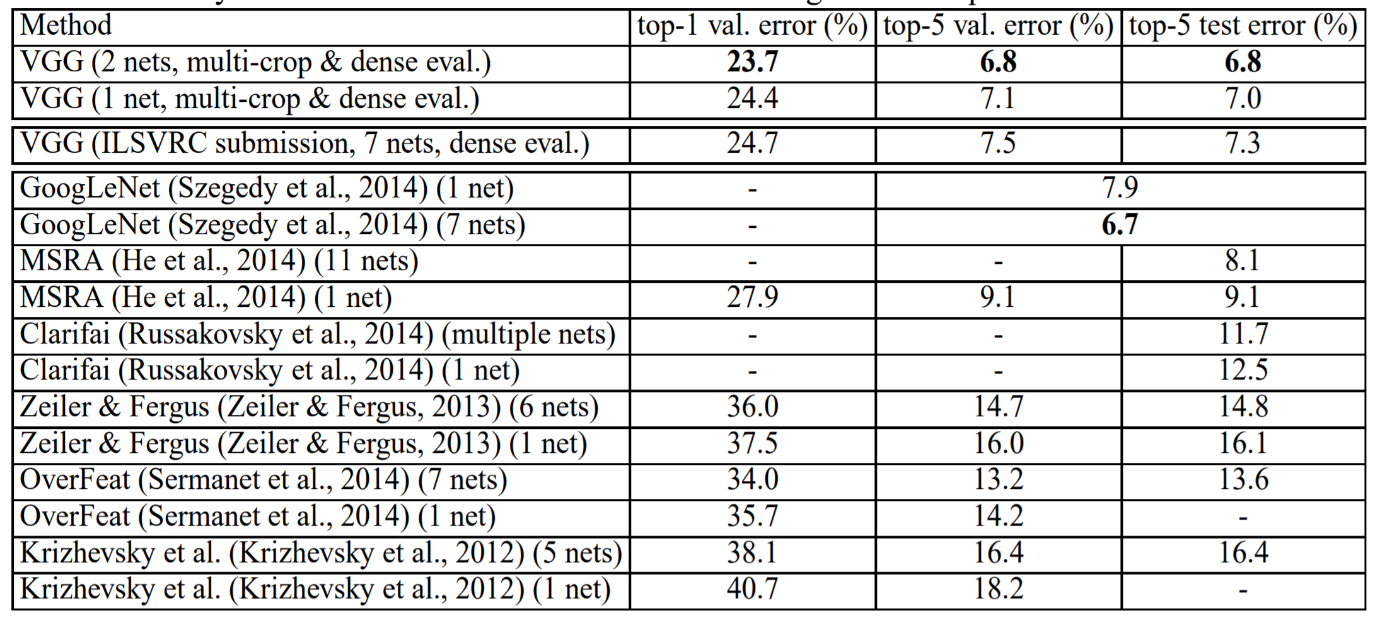


< Single Scale Evaluation>

1. 3개의 1x1 conv-layer가 더 추가됐다는 점을 제외하면 B와 C의 구조는 동일한데 C의 성능이 B보다 높다. 이것은 비선형성을 추가하는 것이 모델의 정확도를 높이는데 도움을 준다는 것을 의미한다.
2. 1x1 conv-layer 대신에 3x3 conv-layer이 들어있는 D가 C보다 성능이 높다. 이것은 kermel size가 정확도에 영향을 준다는 것을 의미한다.
3. Q가 single scale임에도 불과하고 S가 single scale때 보다 multi scale일 때 높은 성능을 보여준다.



< Multi Scale Evaluation>

1. 위 표를 보면 범위를 무작위로 변경하면서 훈련하는 것이(=scale jittering)이 정확도 상승에 도움을 준다는 것을 알 수 있다.
2. VGG 모델은 19 - layer에서 정확도가 수렴한다는 것을 알 수 있다.
3. **최종 성능**

모델 D와 E를 앙상블하여 multi-crop & dense evaluation이라는 메소드를 이용하여   
Top-5 에러가 6.8%로 1위 모델인 GoogleNet과 0.1%차를 보여주는 것을 알 수 있다.

1. **결론**

이 논문을 통해서 층의 깊이와 receptive field (=kernel size)가 정확도에 미치는 영향을 알 수 있었다. 층을 쌓을수록 정확도가 올라가는 것을 보아 데이터 셋이 더 큰 경우에는 더 깊은 층을 쌓을 수 있다는 것을 알 수 있다.  
논문에서 여러가지 평가방법을 언급하였고 이 평가방법과 앙상블을 사용하여 정확도를 올렸다. 이런 점을 본다면 평가방법이 정확도에 끼치는 영향과 앙상블에 대해서 조금 더 공부할 필요가 있다고 생각한다.

[**Reference**](http://endic.naver.com/enkrEntry.nhn?entryId=2d0ed8da08b144349735e26defc23f75)

* Very Deep Convolutional Networks For Large-Scale Image Recognition  
  (Karen Simonyan & Andrew Zisserman , University of Oxford)