# Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning

Inception-v4

# Introduction

- AlexNet에서 CNN이 여러가지 컴비전 task에 효과적임을 밝힘

- CNN과 관련해 ResNet, Inception architecture의 결합을 연구
  - Inception-ResNet
  - ResNet은 깊은 아키텍처 훈련에 도움을 줌
  - Inception은 엄청나게 깊은 아키텍처임

- Inception 아키텍처 스스로도 더 깊고 넓어질 수 있는지 연구
  - Inception-v4
  - Tensorflow의 등장

# Related work

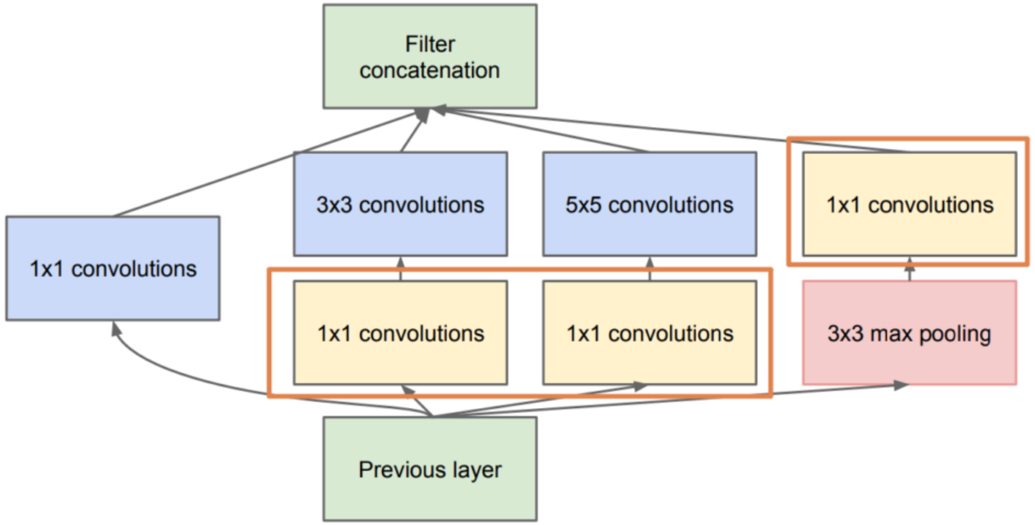
- ResNet은 깊은 네트워크 학습에서 필수라고 주장했었음
- 하지만 ResNet 없이도 학습이 잘 되는 것 확인 (Inception-v4)
- 그래도 ResNet 있으면 학습 속도가 크게 빨라짐 (Inception-ResNet)

# +) Inception-v1 (GoogLeNet)

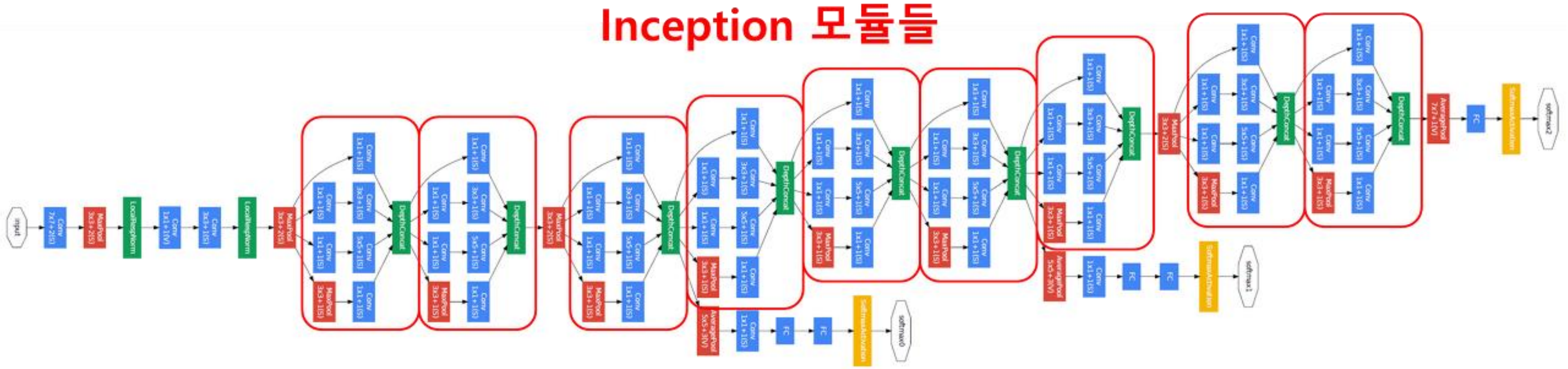
- 모델이 크고 깊으면 정확도가 높음
- But 과적합으로 이어질 수 있고 컴퓨팅 파워가 많이 필요함

- Inception은 AlexNet보다 파라미터 수가 12배 적지만 더 정확함

- 여러 크기의 conv filter
- 1x1 conv layer 사용을 통해 계산량 감소



# +) Inception-v1 (GoogLeNet)



# +) Inception-v2

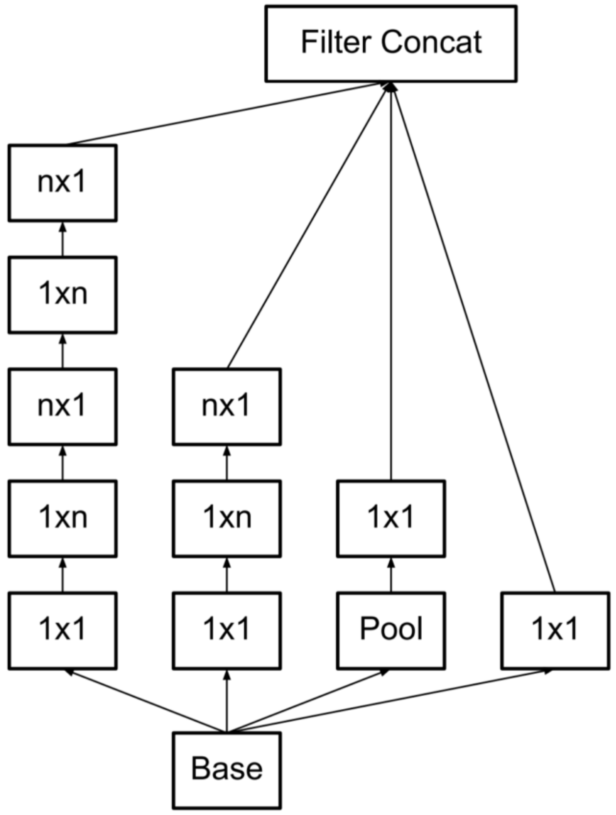



Figure 6. Inception modules after the factorization of the $n \times n$ convolutions. In our proposed architecture, we chose $n = 7$ for the $17 \times 17$ grid. (The filter sizes are picked using principle 3)

.

# Architectural Choices – Pure Inception

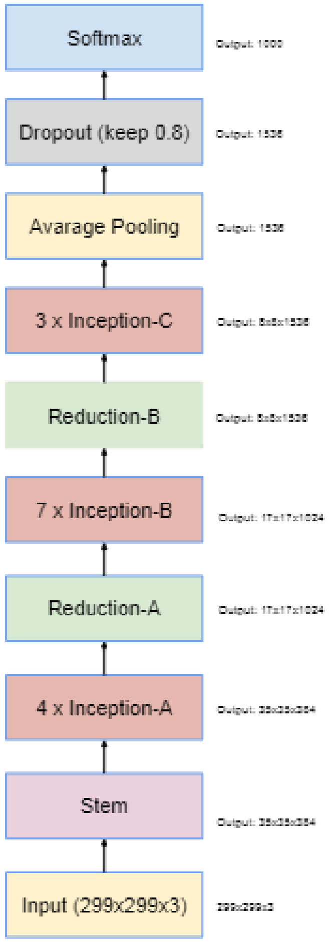



Figure 9. The overall schema of the Inception-v4 network. For the detailed modules, please refer to Figures 3, 4, 5, 6, 7 and 8 for the detailed structure of the various components.

# Architectural Choices – Pure Inception

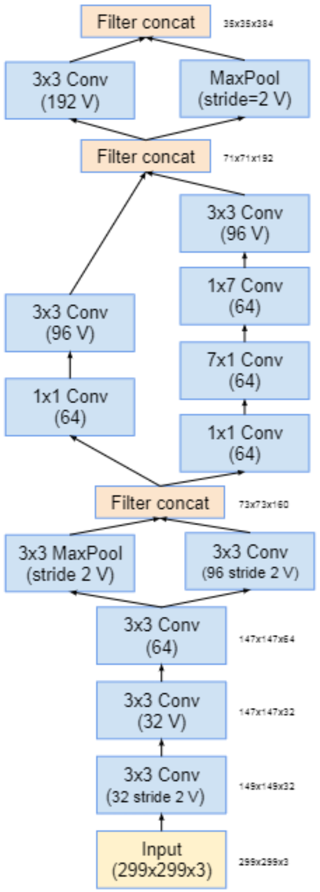



Figure 3. The schema for stem of the pure Inception-v4 and Inception-ResNet-v2 networks. This is the input part of those networks. Cf. Figures 9 and 15

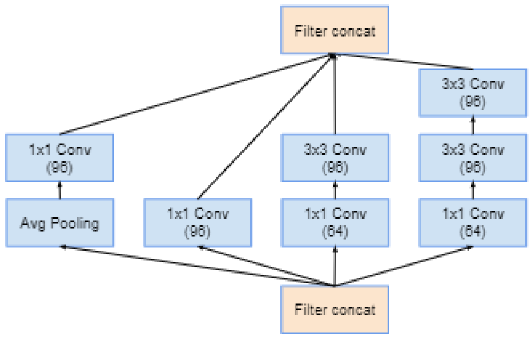



Figure 4. The schema for 35 × 35 grid modules of the pure Inception-v4 network. This is the Inception-A block of Figure 9.

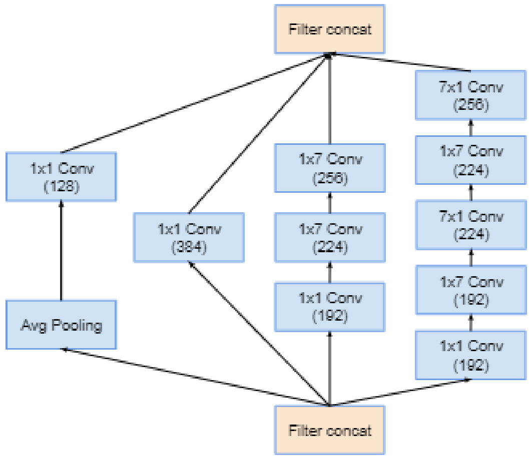



Figure 5. The schema for 17 × 17 grid modules of the pure Inception-v4 network. This is the Inception-B block of Figure 9.

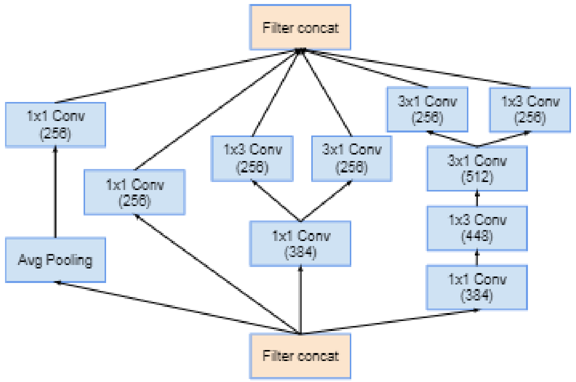



Figure 6. The schema for $8 \times 8$ grid modules of the pure Inception-v4 network. This is the Inception-C block of Figure 9.

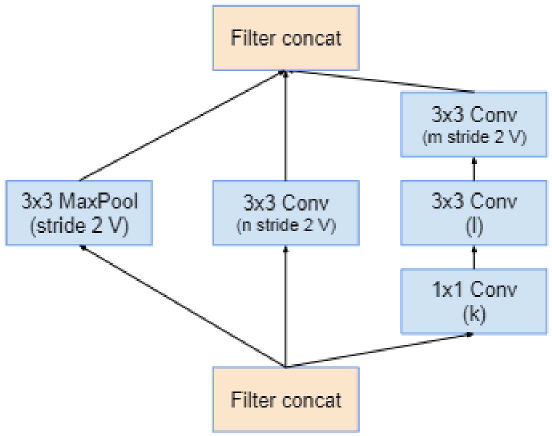



Figure 7. The schema for 35 × 35 to 17 × 17 reduction module. Different variants of this blocks (with various number of filters) are used in Figure 9, and 15 in each of the new Inception(-v4, -ResNet-v1, -ResNet-v2) variants presented in this paper. The $k$, $l$, $m$, $n$ numbers represent filter bank sizes which can be looked up in Table 1.

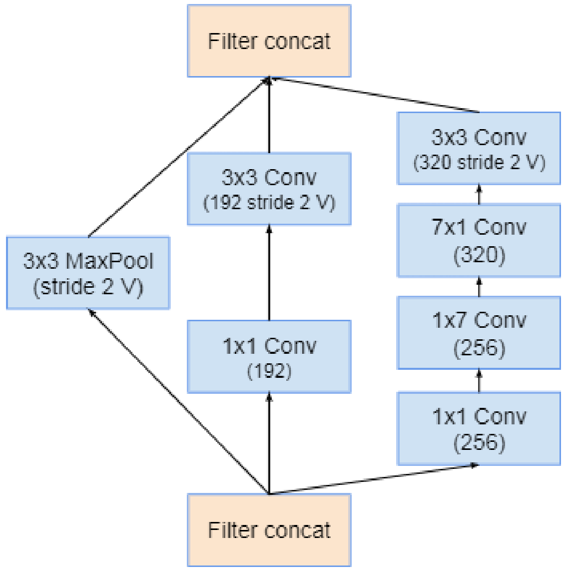



Figure 8. The schema for 17 × 17 to 8 × 8 grid-reduction module. This is the reduction module used by the pure Inception-v4 network in Figure 9.

# Architectural Choices – Residual

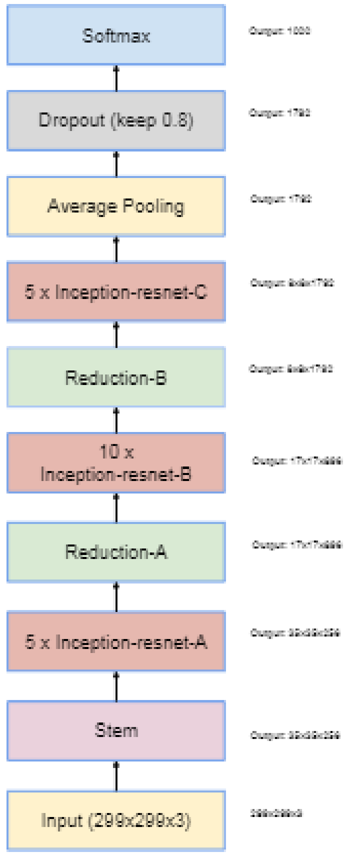



Figure 15. Schema for Inception-ResNet-v1 and Inception-ResNet-v2 networks. This schema applies to both networks but the underlying components differ. Inception-ResNet-v1 uses the blocks as described in Figures 14, 10, 7, 11, 12 and 13. Inception-ResNet-v2 uses the blocks as described in Figures 3, 16, 7,17, 18 and 19. The output sizes in the diagram refer to the activation vector tensor shapes of Inception-ResNet-v1.

# Architectural Choices – Residual

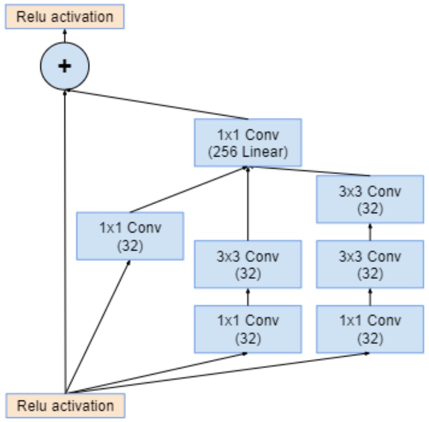



Figure 10. The schema for 35 × 35 grid (Inception-ResNet-A) module of Inception-ResNet-v1 network.

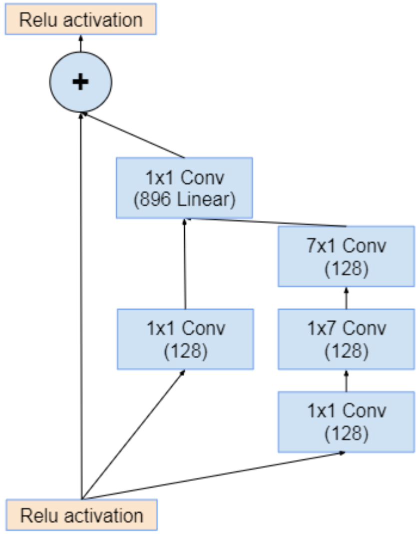



Figure 11. The schema for 17 × 17 grid (Inception-ResNet-B) module of Inception-ResNet-v1 network.

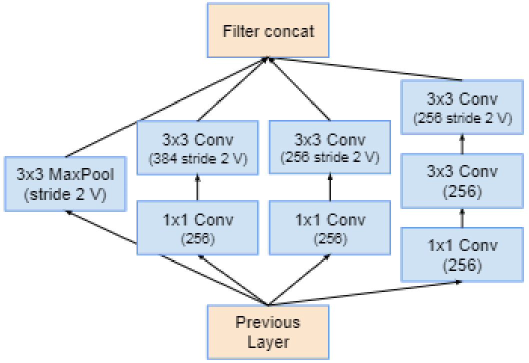



Figure 12. "Reduction-B" 17 × 17 to 8 × 8 grid-reduction module. This module used by the smaller Inception-ResNet-v1 network in Figure 15.

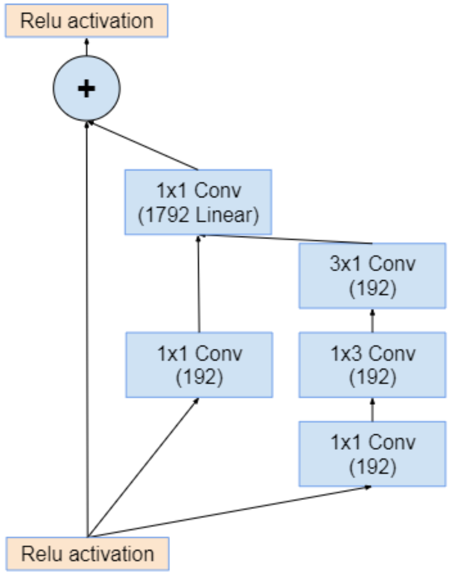
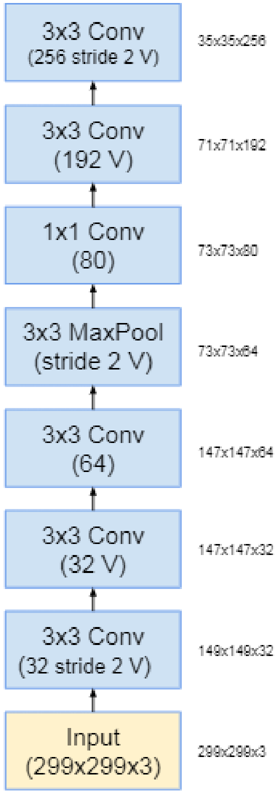



Figure 13. The schema for 8×8 grid (Inception-ResNet-C) module of Inception-ResNet-v1 network.




igure 14. The stem of the Inception-ResNet-v1 network.

# Scaling the Residual

- 필터 개수가 1000개가 넘어가면 훈련이 안됨
- Residual을 scaling down하는 게 훈련에 도움이 됨

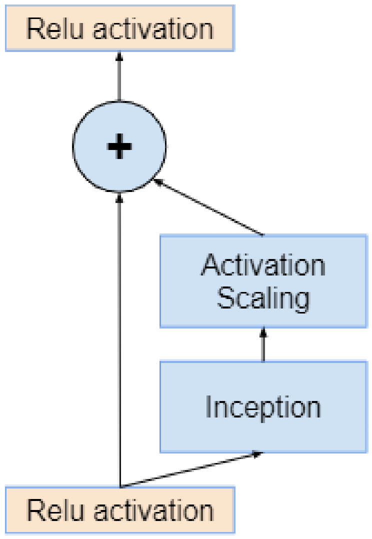



Figure 20. The general schema for scaling combined Inception-resnet moduels. We expect that the same idea is useful in the general resnet case, where instead of the Inception block an arbitrary subnetwork is used. The scaling block just scales the last linear activations by a suitable constant, typically around 0.1.

# Training Methodology

- SGD
- RMSProp (decay=0.9, e=1,0)
- Learning rate 0.045 (decay)

# Experimental Results

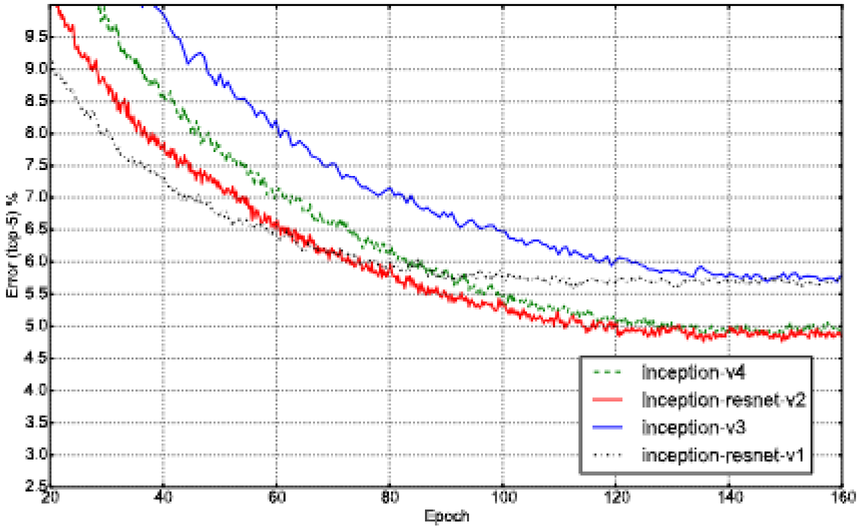



Figure 25. Top-5 error evolution of all four models (single model, single crop). Showing the improvement due to larger model size. Although the residual version converges faster, the final accuracy seems to mainly depend on the model size.

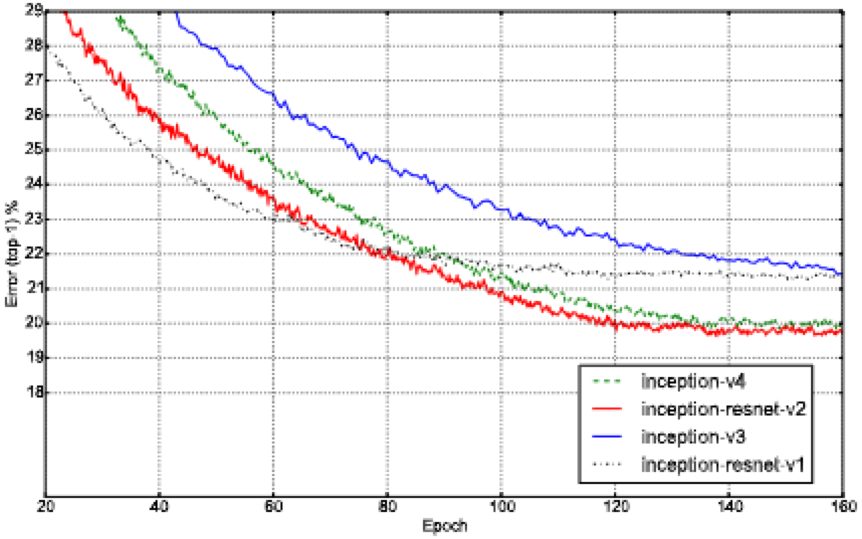



Figure 26. Top-1 error evolution of all four models (single model, single crop). This paints a similar picture as the top-5 evaluation.

# Experimental Results

| Network | Top-1 Error | Top-5 Error |
|---|---|---|
| BN-Inception [6] | 25.2% | 7.8% |
| Inception-v3 [15] | 21.2% | 5.6% |
| Inception-ResNet-v1 | 21.3% | 5.5% |
| Inception-v4 | 20.0% | 5.0% |
| Inception-ResNet-v2 | 19.9% | 4.9% |

Table 2. Single crop - single model experimental results. Reported on the non-blacklisted subset of the validation set of ILSVRC 2012.

| Network | Models | Top-1 Error | Top-5 Error |
|---|---|---|---|
| ResNet-151 [5] | 6 | – | 3.6% |
| Inception-v3 [15] | 4 | 17.3% | 3.6% |
| Inception-v4 + $3\times$ Inception-ResNet-v2 | 4 | 16.5% | 3.1% |

Table 5. Ensemble results with 144 crops/dense evaluation. Reported on the all 50000 images of the validation set of ILSVRC 2012. For Inception-v4(+Residual), the ensemble consists of one pure Inception-v4 and three Inception-ResNet-v2 models and were evaluated both on the validation and on the test-set. The test-set performance was 3.08% top-5 error verifying that we don't overfit on the validation set.

# Conclusions

- Inception-ResNet-v1: Inception-v3와 비슷한 cost
- Inception-ResNet-v2: recognition performance 향상
- Inception-v4: 위에 거랑 성능 같음

- Residual connection이 학습 속도 엄청 줄여줌
- 위의 모델들이 이전 모델들을 능가했음