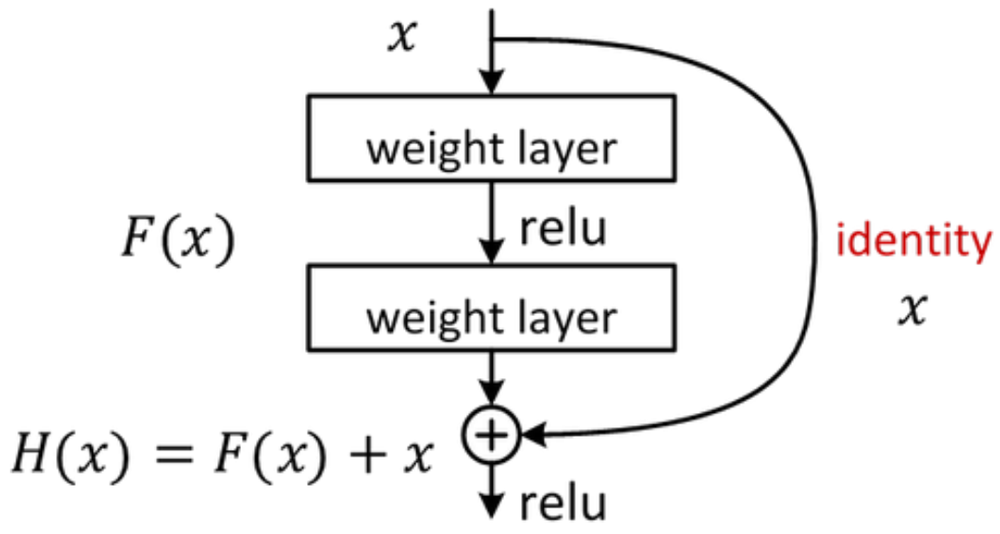
Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning

Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, Alex Alemi

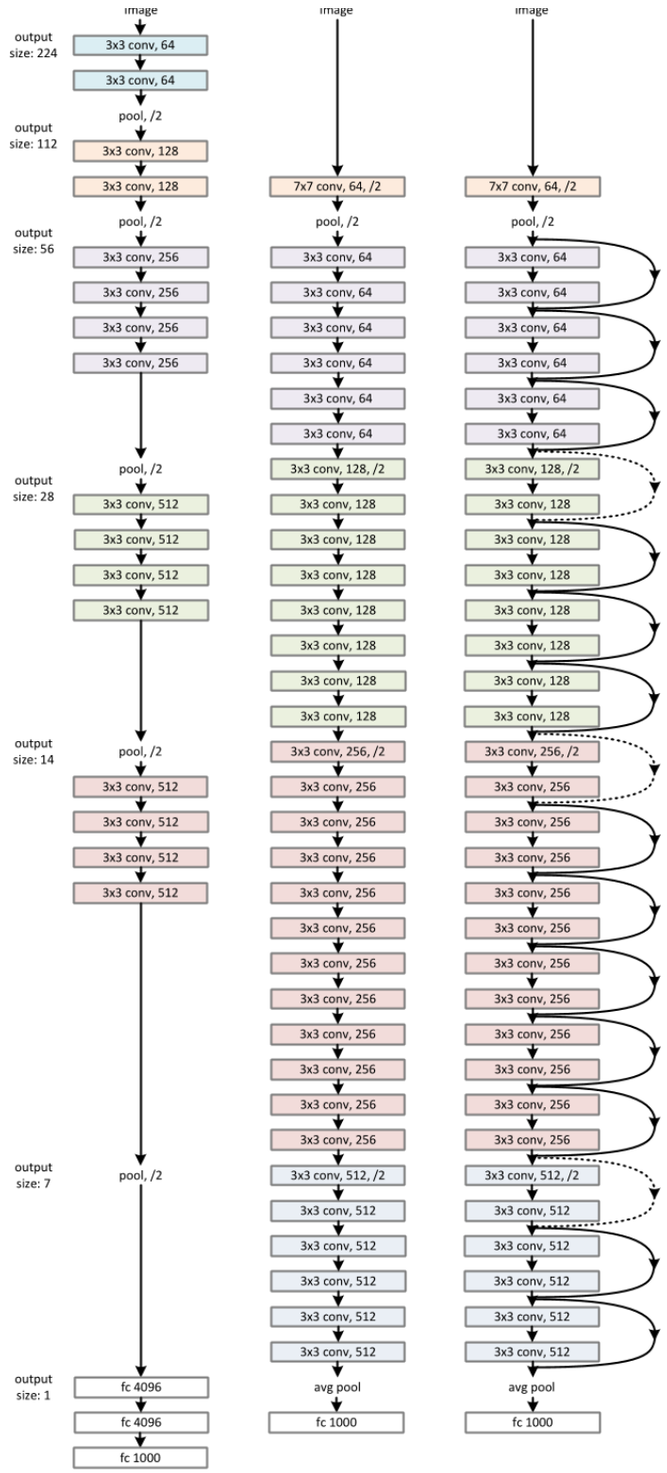
Presenter: Tianyi Jin, Haoran Liu

Residual Learning



Introduced by He et al., 2015.

- **Three advantages**:
- Easy to optimize
- Gain accuracy from considerably increased depth
- Speed up training



Inception Network

Inception-v1 (aka GoogLeNet)

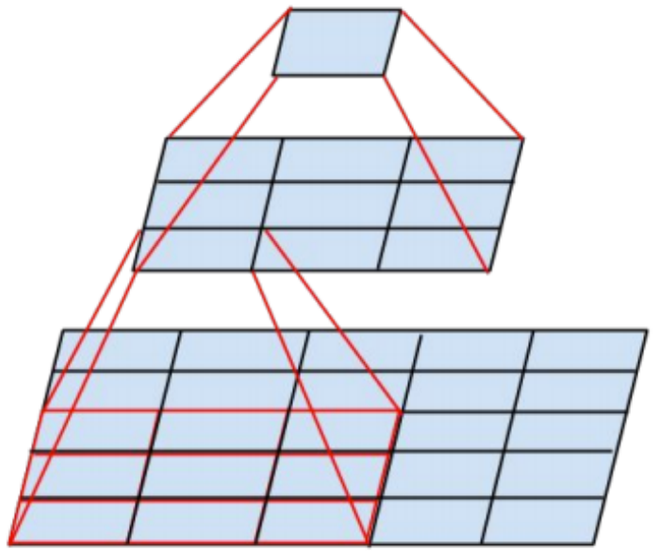


Figure 1. Mini-network replacing the 5×5 convolutions.

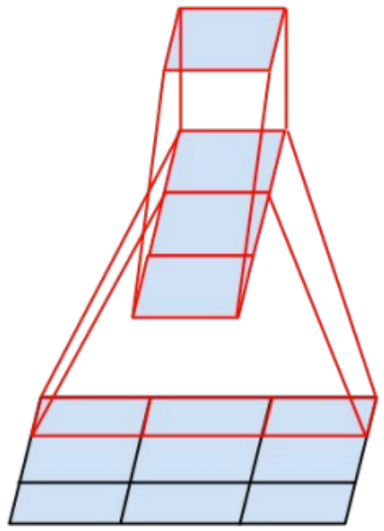


Figure 3. Mini-network replacing the 3×3 convolutions. The lower layer of this network consists of a 3×1 convolution with 3 output units.

Inception-v3: Factorization

Architectural Choices

Pure Inception blocks: Uniform choices for each grid size

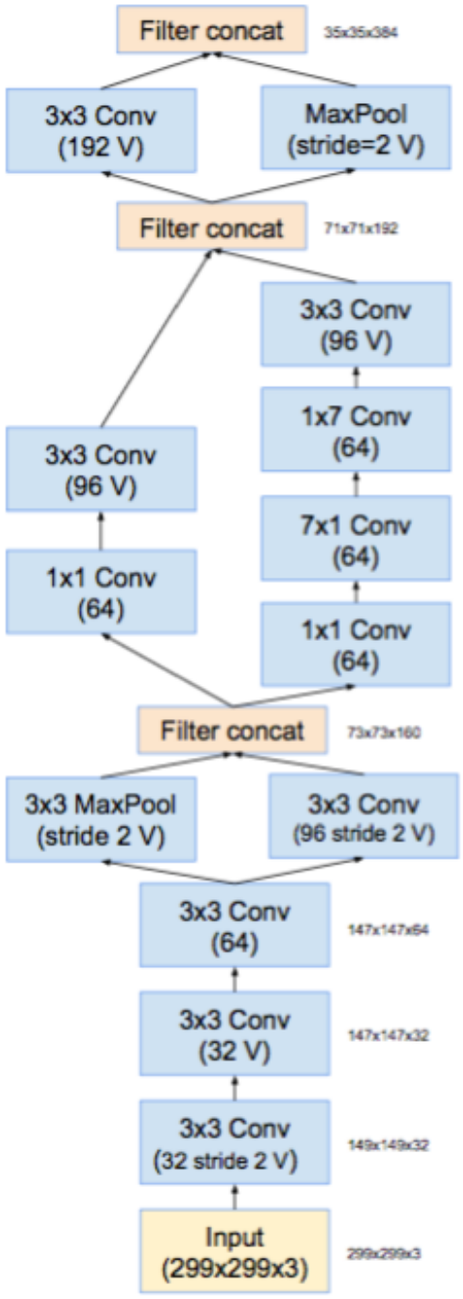


Figure 3. The schema for stem of the pure Inception-v4 and Inception-ResNet-v2 networks. This is the input part of those networks. Cf. Figures 9 and 15

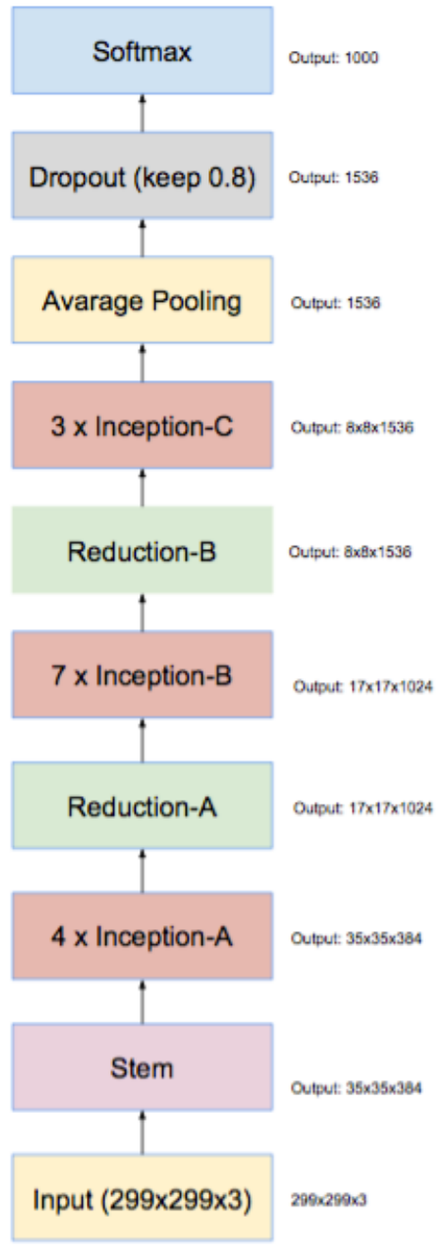


Figure 9. The overall schema of the Inception-v4 network. For the detailed modules, please refer to Figures 3, 4, 5, 6, 7 and 8 for the detailed structure of the various components.

Architectural Choices

Pure Inception blocks: Uniform choices for each grid size

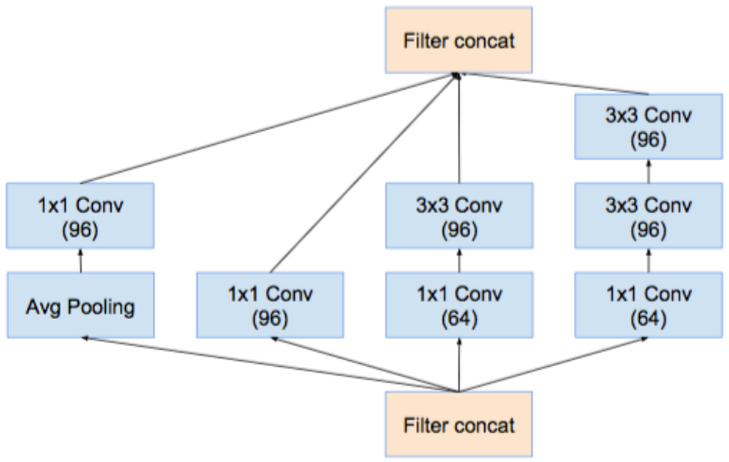


Figure 4. The schema for 35 × 35 grid modules of the pure Inception-v4 network. This is the Inception-A block of Figure 9.

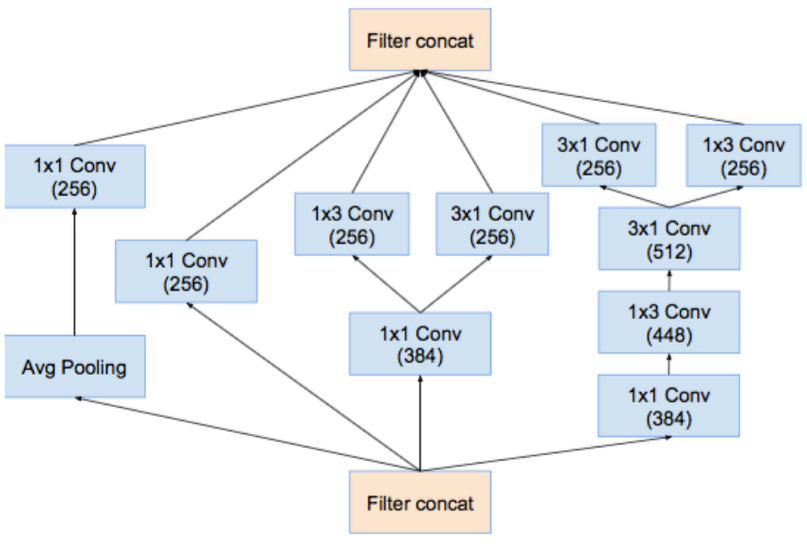


Figure 6. The schema for 8 × 8 grid modules of the pure Inception-v4 network. This is the Inception-C block of Figure 9.

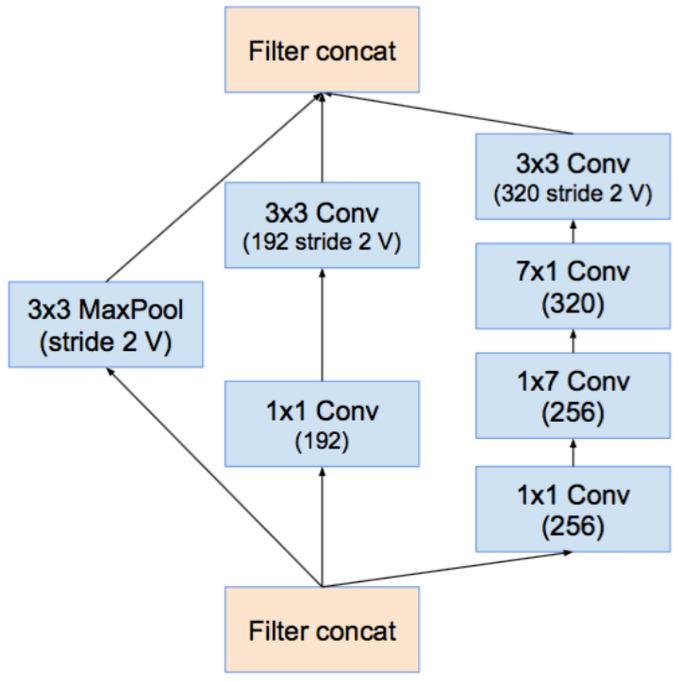


Figure 8. The schema for 17 × 17 to 8 × 8 grid-reduction module. This is the reduction module used by the pure Inception-v4 network in Figure 9.

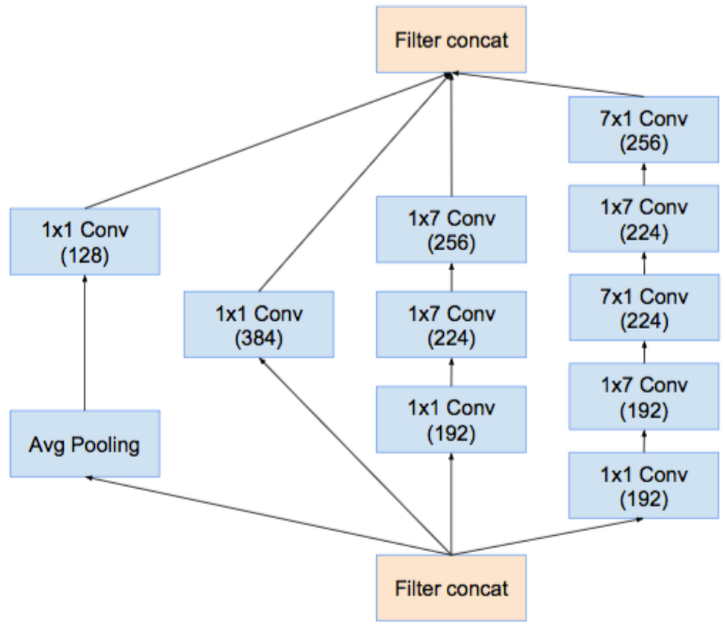


Figure 5. The schema for 17 × 17 grid modules of the pure Inception-v4 network. This is the Inception-B block of Figure 9.

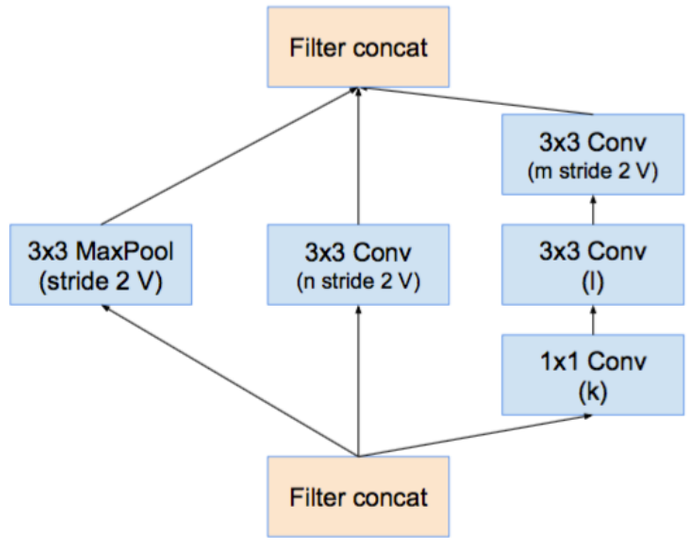


Figure 7. The schema for 35 × 35 to 17 × 17 reduction module. Different variants of this blocks (with various number of filters) are used in Figure 9, and 15 in each of the new Inception(-v4, -ResNet-v1, -ResNet-v2) variants presented in this paper. The k, l, m, n numbers represent filter bank sizes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

Architectural Choices

Residual Inception blocks: Cheap 1x1 convolution

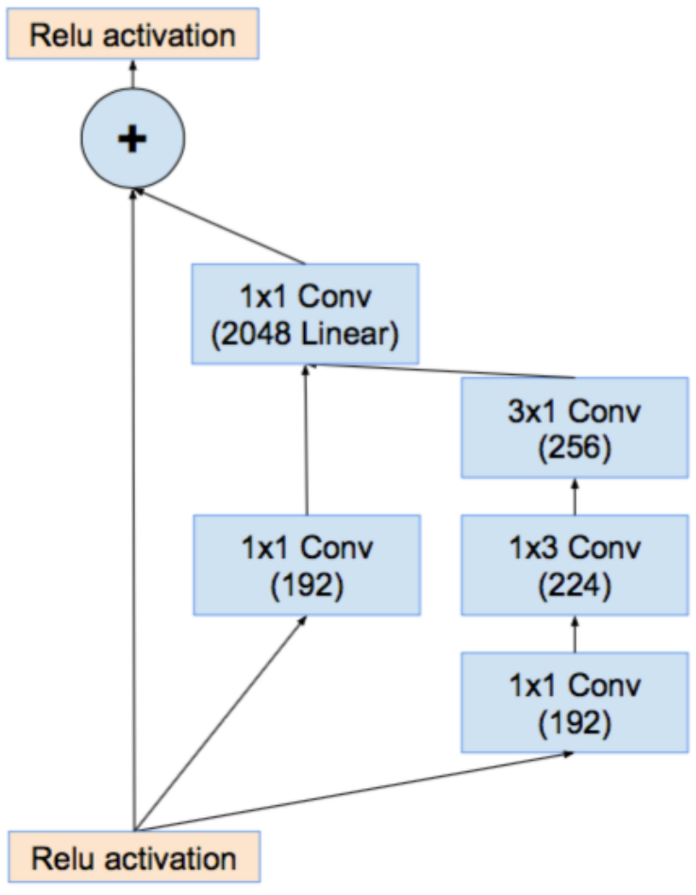


Figure 19. The schema for 8×8 grid (Inception-ResNet-C) module of the Inception-ResNet-v2 network.

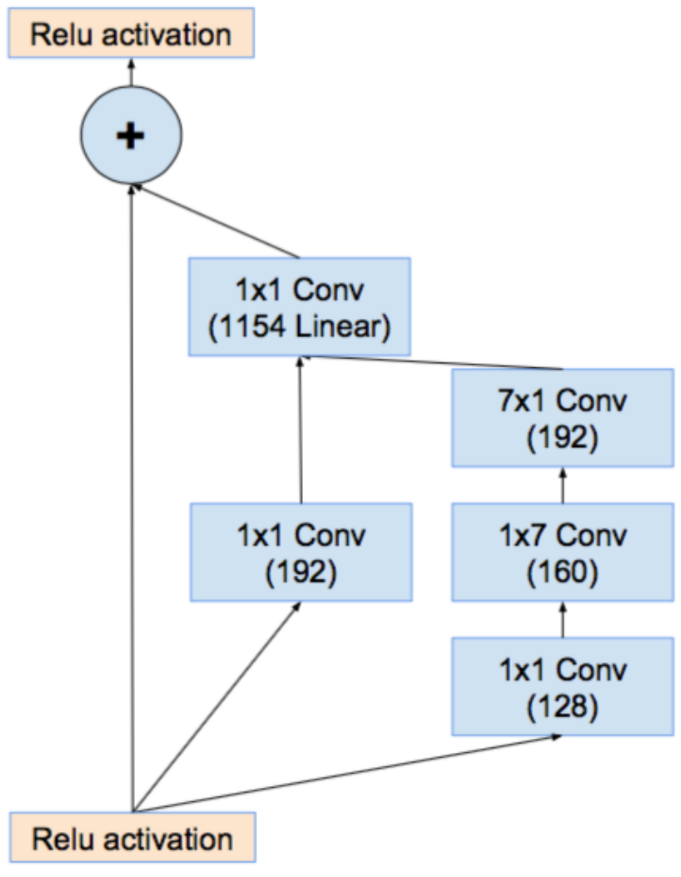


Figure 17. The schema for 17 × 17 grid (Inception-ResNet-B) module of the Inception-ResNet-v2 network.

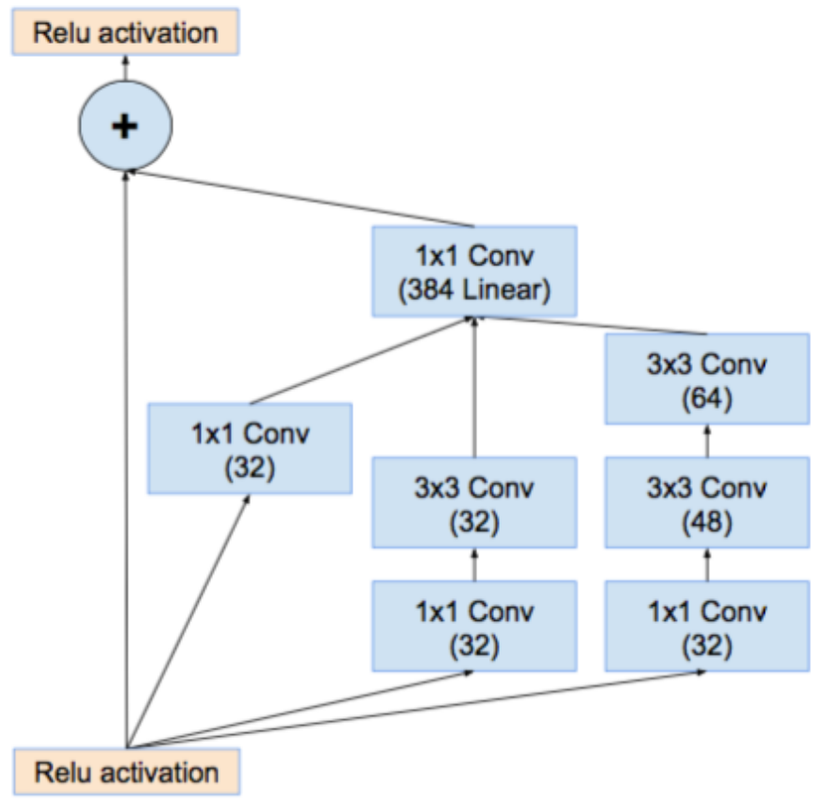


Figure 16. The schema for 35 × 35 grid (Inception-ResNet-A) module of the Inception-ResNet-v2 network.

Architectural Choices

Residual Inception blocks: Cheap 1x1 convolution

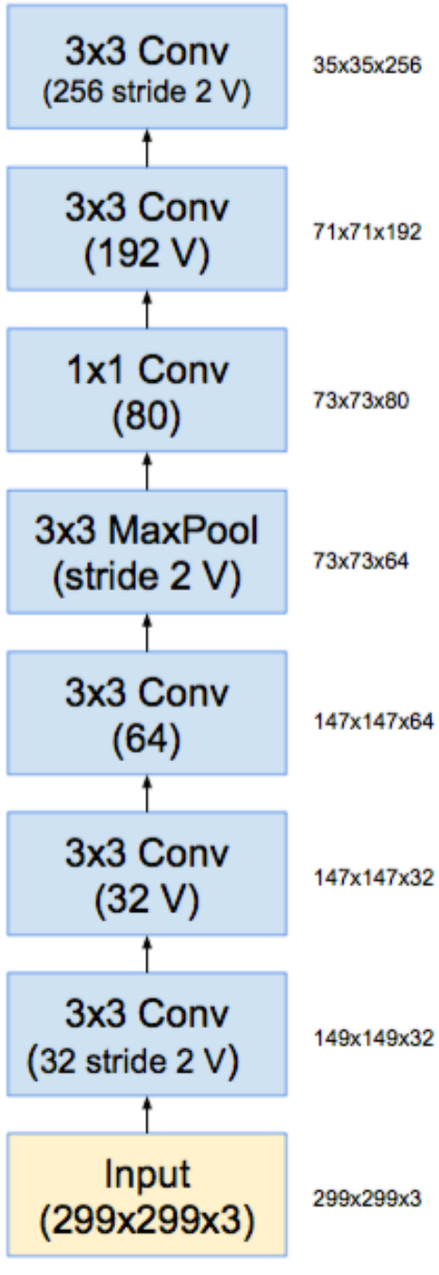


Figure 14. The stem of the Inception-ResNet-v1 network

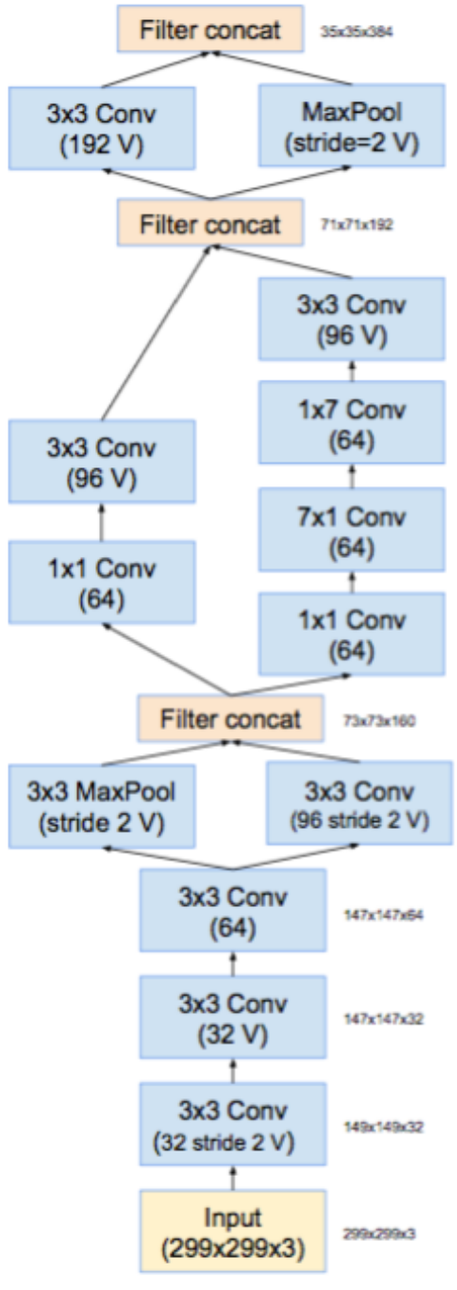


Figure 3. The schema for stem of the pure Inception-v4 and Inception-ResNet-v2 networks. This is the input part of those networks. Cf. Figures 9 and 15

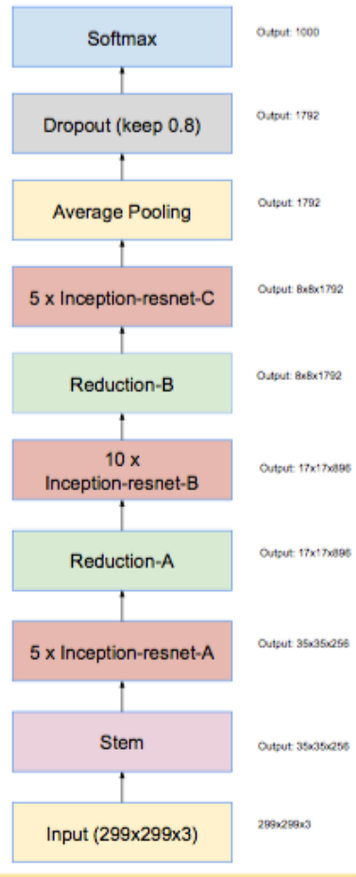


Figure 15. Schema for Inception-ResNet-v1 and Inception-ResNet-v2 networks. This schema applies to both networks but the underlying components differ. Inception-ResNet-v1 uses the blocks as described in Figures 14, 10, 7, 11, 12 and 13. Inception-ResNet-v2 uses the blocks as described in Figures 3, 16,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

Architectural Choices

Small tricks:

- Use batch-normalization only on top of layers (Inception Res-Net)
- Scaling down the residuals to stabilize training

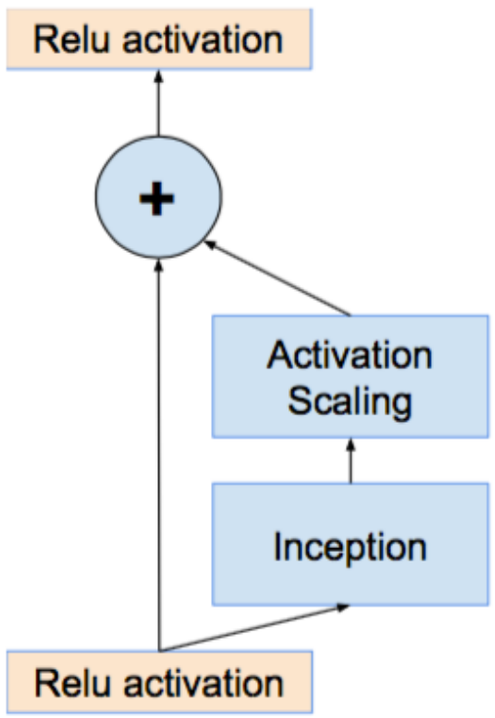


Figure 20. The general schema for scaling combined Inception-resnet moduels. We expect that the same idea is useful in the general resnet case, where instead of the Inception block an arbitrary subnetwork is used. The scaling block just scales the last linear activations by a suitable constant, typically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

Training

- Stochastic gradient utilizing the TensorFlow distributed machine learning system
- Using 20 replicas running each on a Nvidia Kepler GPU
- Momentum with a decay of 0.9
- Best models were achieved using RMSProp with decay of 0.9 and ϵ = 1.0.
- Learning rate of 0.045, decayed every two epochs using an exponential rate of 0.94.
- Model evaluations are performed using a running average of the parameters computed over time.

Experimental Results

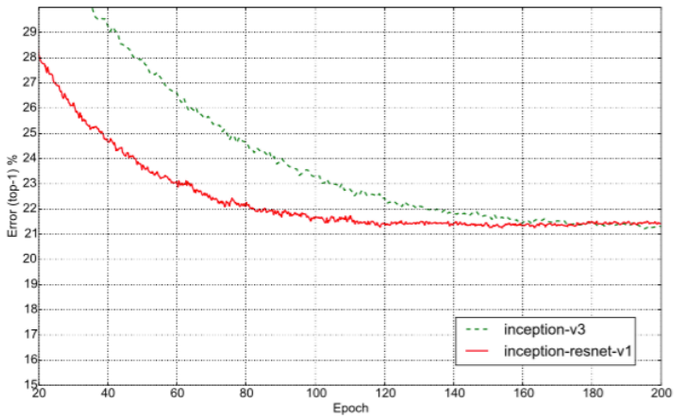


Figure 21. Top-1 error evolution during training of pure Inception-v3 vs a residual network of similar computational cost. The evaluation is measured on a single crop on the non-blacklist images of the ILSVRC-2012 validation set. The residual model was training much faster, but reached slightly worse final accuracy than the traditional Inception-v3.

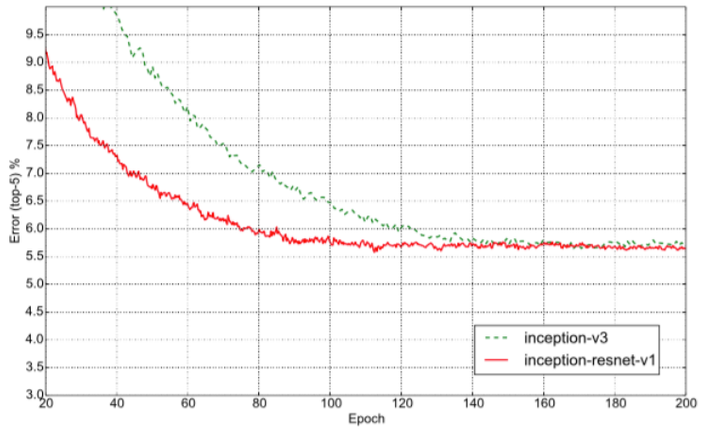


Figure 22. Top-5 error evolution during training of pure Inception-v3 vs a residual Inception of similar computational cost. The evaluation is measured on a single crop on the non-blacklist images of the ILSVRC-2012 validation set. The residual version has trained much faster and reached slightly better final recall on the validation set.

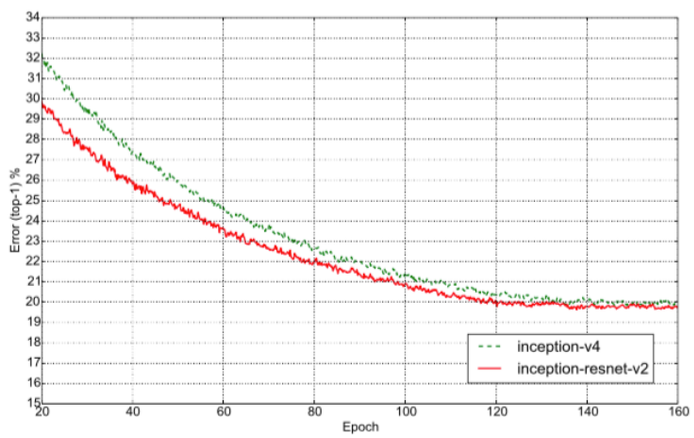


Figure 23. Top-1 error evolution during training of pure Inception-v3 vs a residual Inception of similar computational cost. The evaluation is measured on a single crop on the non-blacklist images of the ILSVRC-2012 validation set. The residual version was training much faster and reached slightly better final accuracy than the traditional Inception-v4.

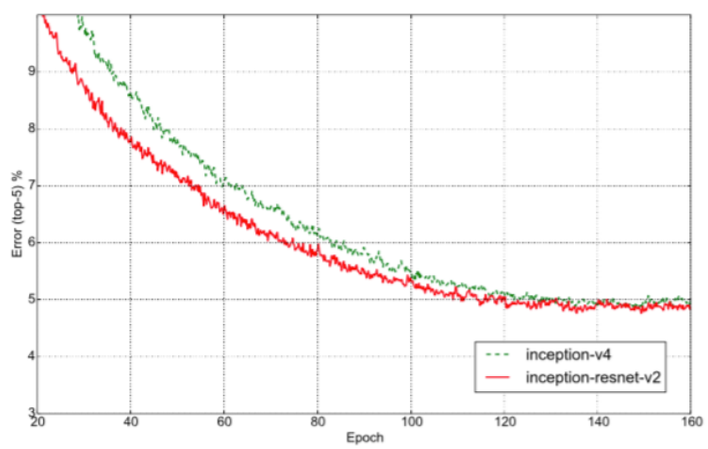


Figure 24. Top-5 error evolution during training of pure Inception-v4 vs a residual Inception of similar computational cost. The evaluation is measured on a single crop on the non-blacklist images of the ILSVRC-2012 validation set. The residual version trained faster and reached slightly better final recall on the validation set.

Experimental Results

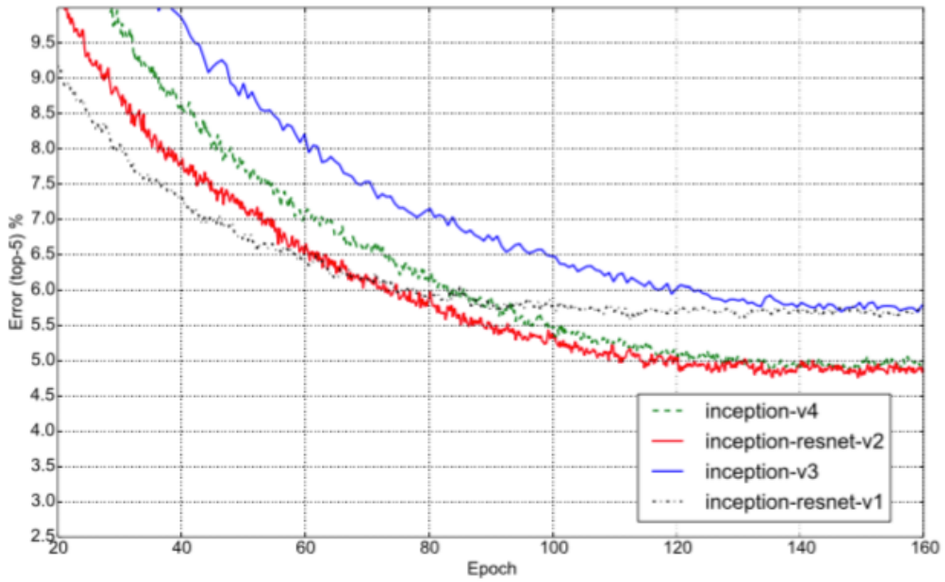


Figure 25. Top-5 error evolution of all four models (single model, single crop). Showing the improvement due to larger model size. Although the residual version converges faster, the final accuracy seems to mainly depend on the model size.

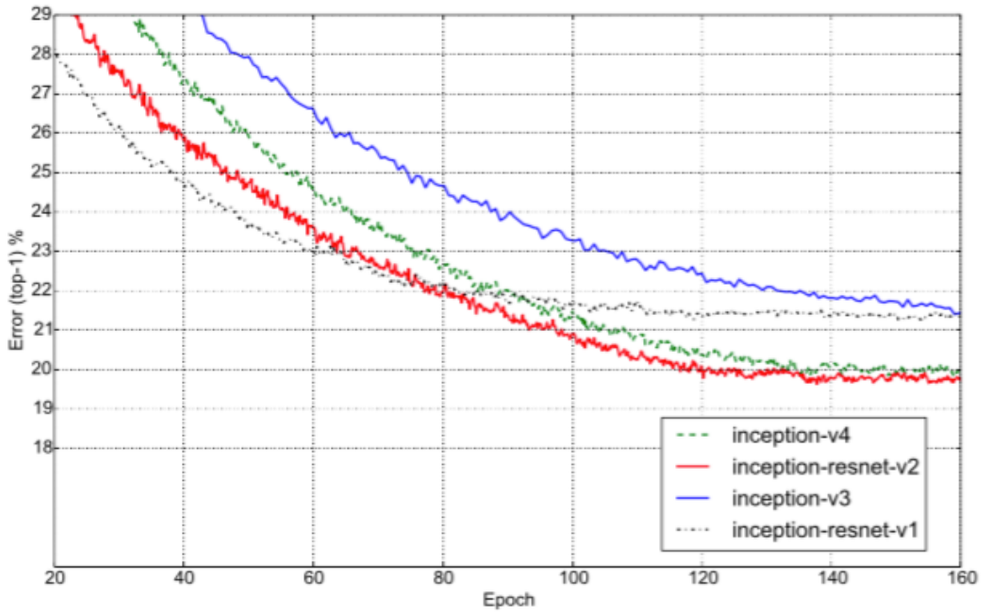


Figure 26. Top-1 error evolution of all four models (single model, single crop). This paints a similar picture as the top-5 evaluation.

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	Crops	Top-1 Error	Top-5 Error
ResNet-151 [5]	10	21.4%	5.7%
Inception-v3 [15]	12	19.8%	4.6%
Inception-ResNet-v1	12	19.8%	4.6%
Inception-v4	12	18.7%	4.2%
Inception-ResNet-v2	12	18.7%	4.1%

Table 3. 10/12 crops evaluations - single model experimental results. Reported on the all 50000 images of the validation set of ILSVRC 2012.

Network	Crops	Top-1 Error	Top-5 Error
ResNet-151 [5]	dense	19.4%	4.5%
Inception-v3 [15]	144	18.9%	4.3%
Inception-ResNet-v1	144	18.8%	4.3%
Inception-v4	144	17.7%	3.8%
Inception-ResNet-v2	144	17.8%	3.7%

Table 4. 144 crops evaluations - single model experimental results. Reported on the all 50000 images 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× 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.

Questions?

Thank you!