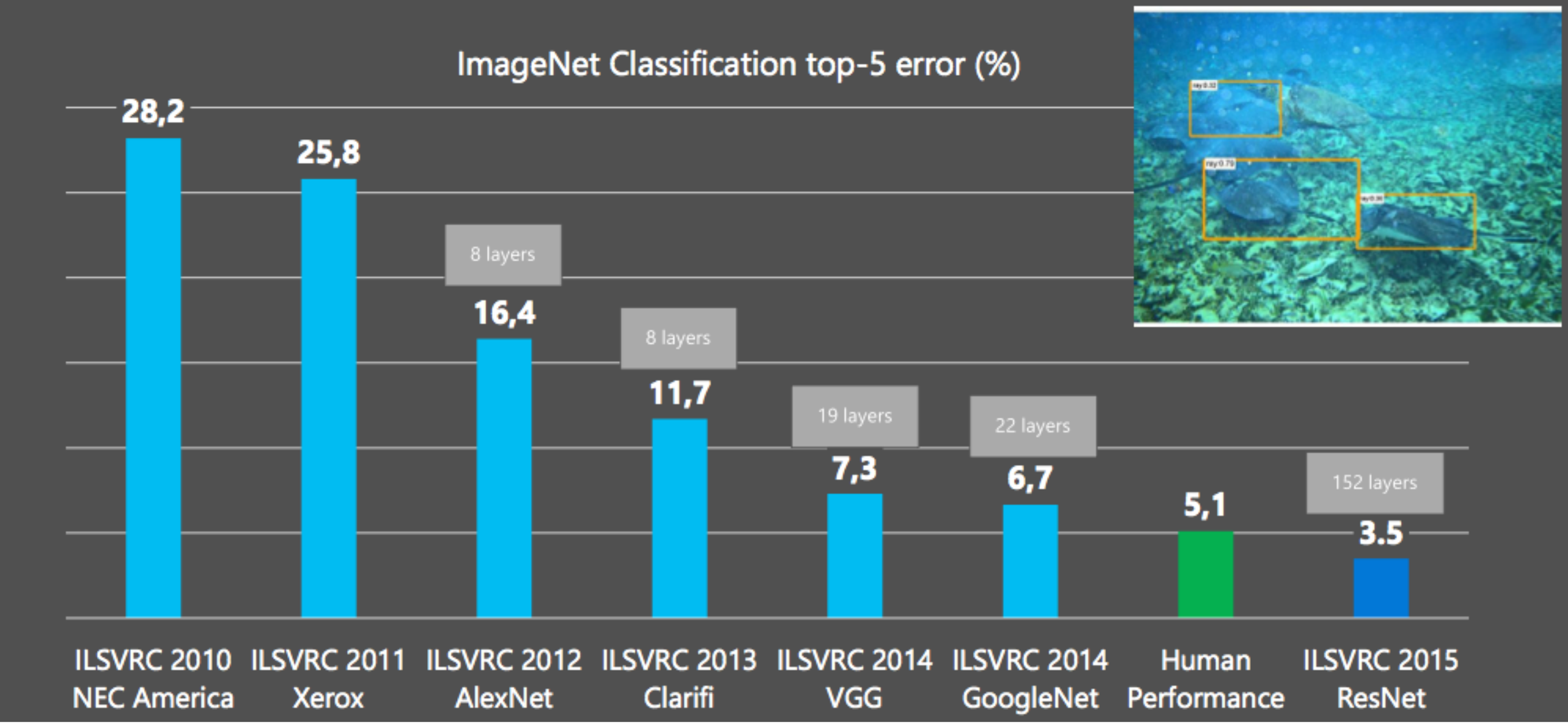
A Review of Inception(V1, V2 and V3)

©Kuang

Graduate School of Information, Production and Systems
早稲田大学　大学院情報生産システム研究科

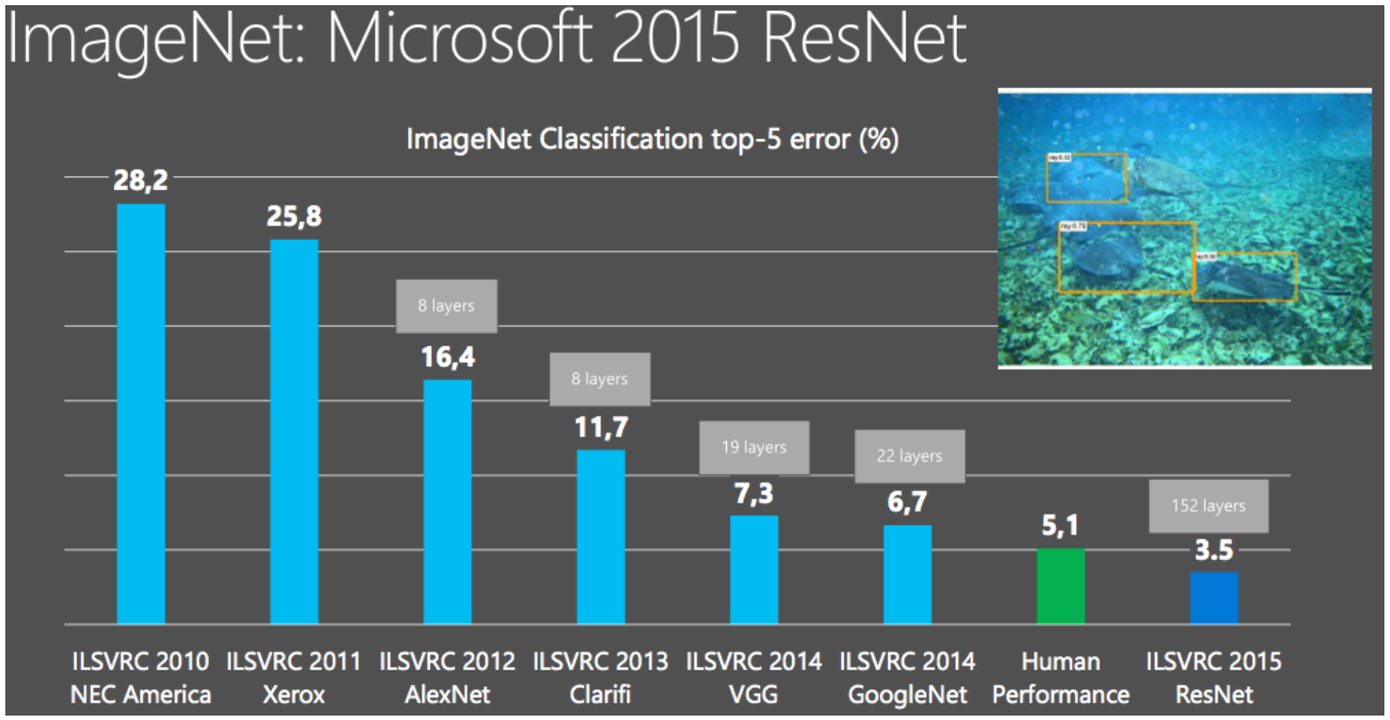
Introduction

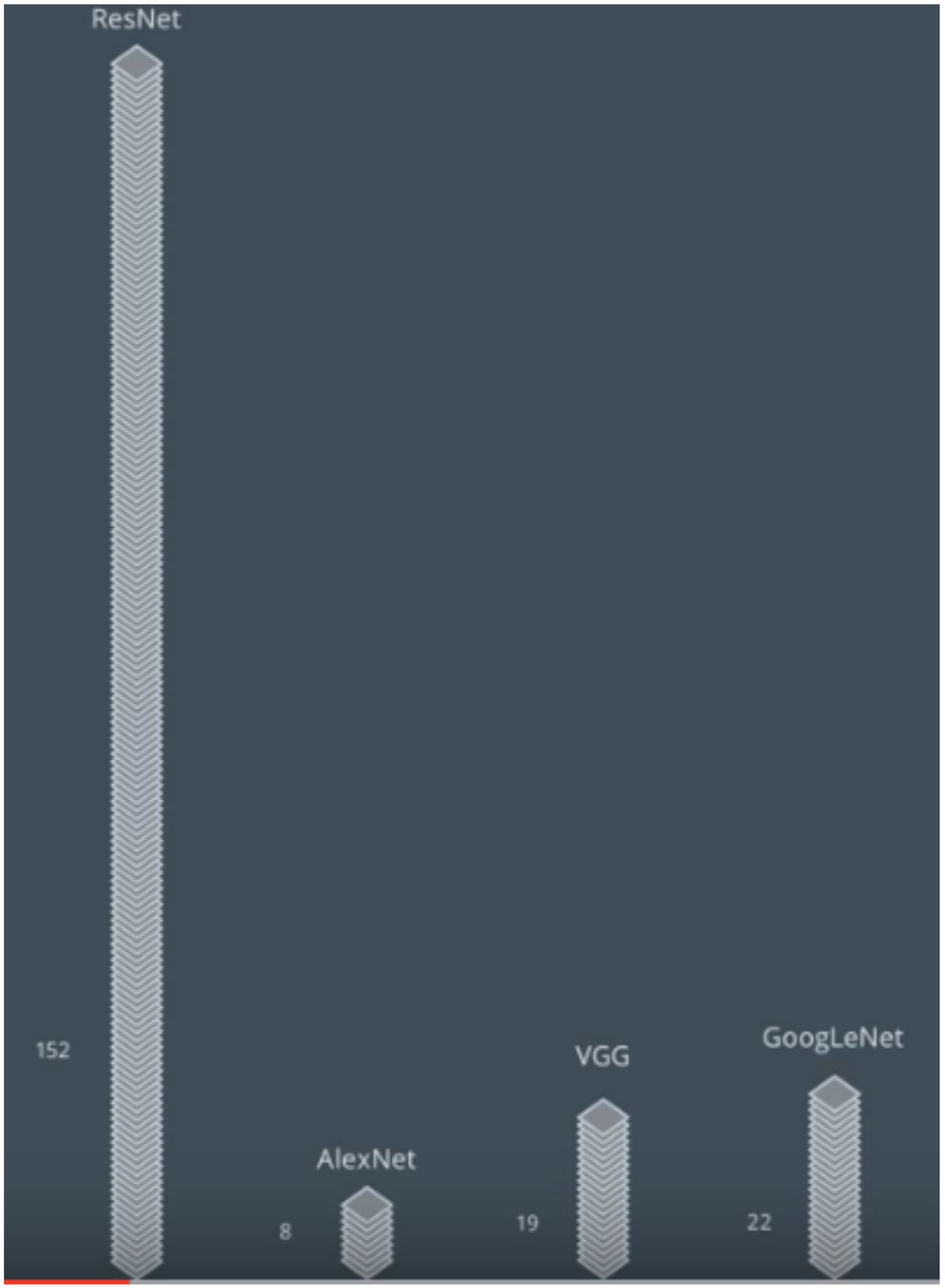
ImageNet: Microsoft 2015 ResNet



*Microsoft

Introduction





ILSVRC

Graduate School of Information, Production and Systems
早稲田大学　大学院情報生産システム研究科

Introduction

The Inception architecture of GoogLeNet [1] was also designed to perform well even under strict constraints on memory and computational budget.

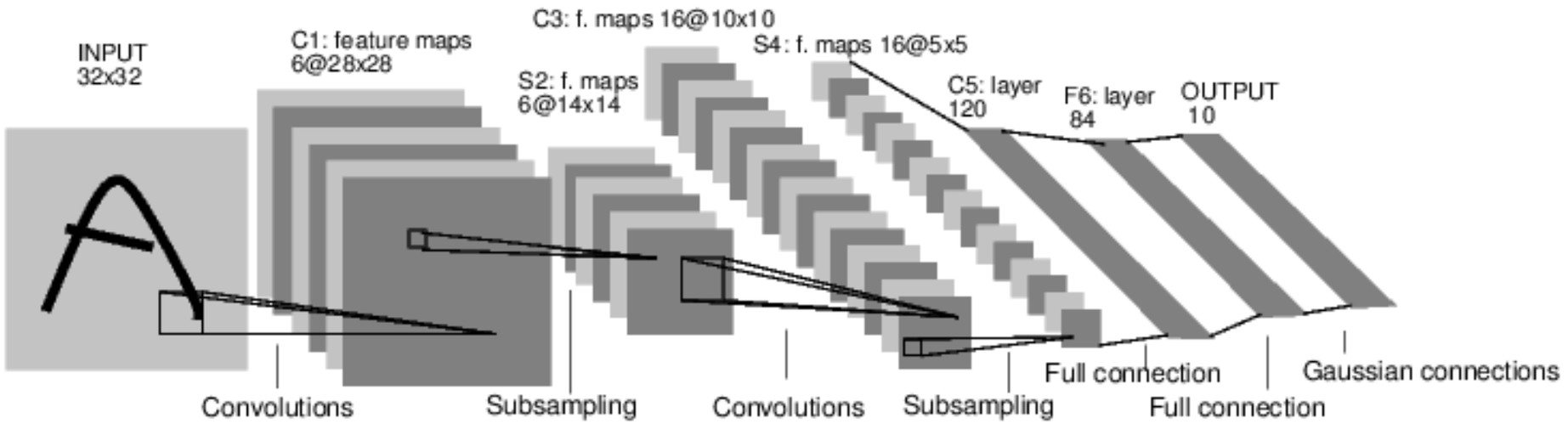
GoogleNet employed only **5 million** parameters, which represented a 12× reduction with respect to its predecessor AlexNet, which used **60 million** parameters. Furthermore, VGGNet employed about 3x more parameters(**180+ million**) than AlexNet. [2]

---

[1] Szegedy, Christian, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. "Going deeper with convolutions." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1-9. 2015.
[2] Szegedy, Christian, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. "Rethinking the inception architecture for computer vision." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2818-2826. 2016.

Related Work

LeNet-5

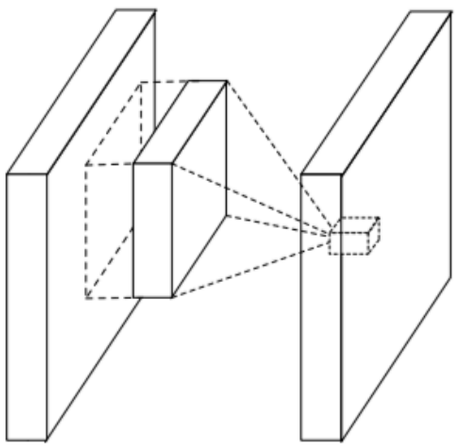


LeNet
GoogLeNet

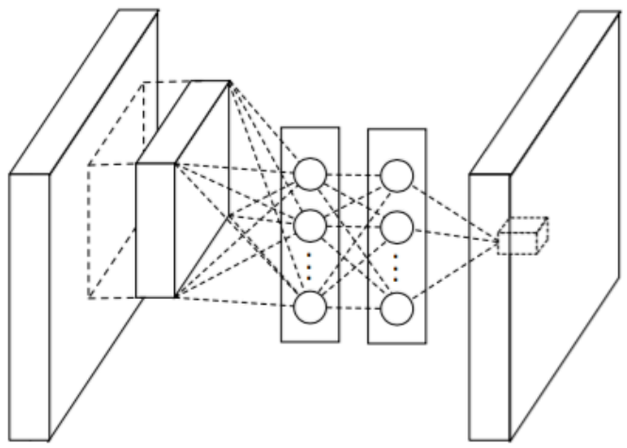
Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel.Backpropagation applied to handwritten zip code recognition. Neural Comput., 1(4):541–551, December 1989.

Related Work

Network in Network

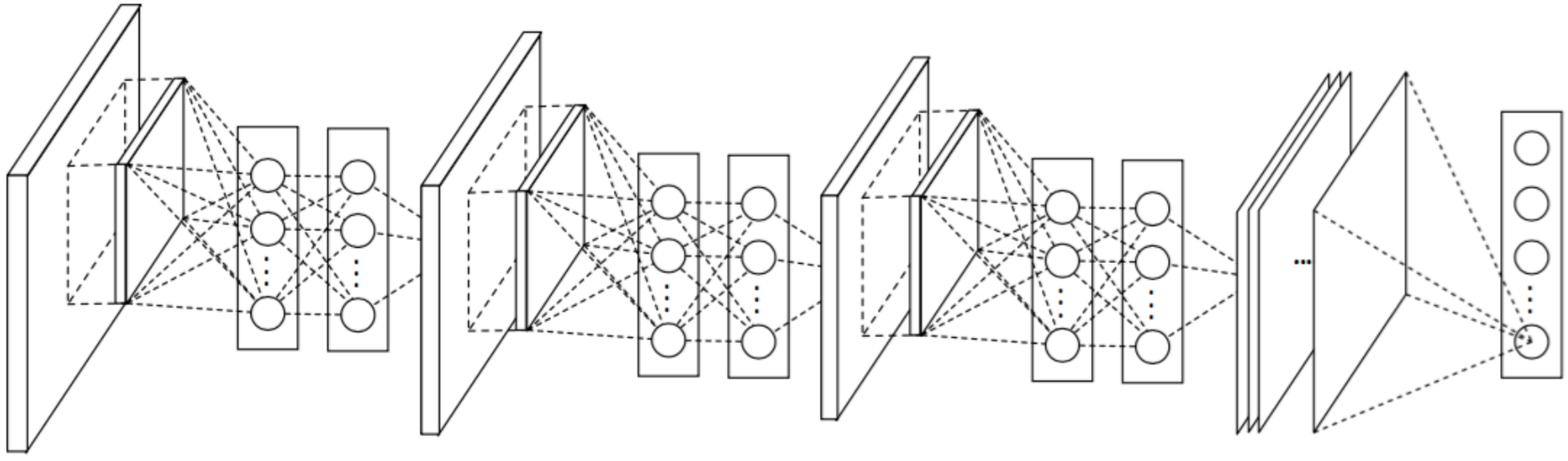


(a) Linear convolution layer



(b) Mlpconv layer

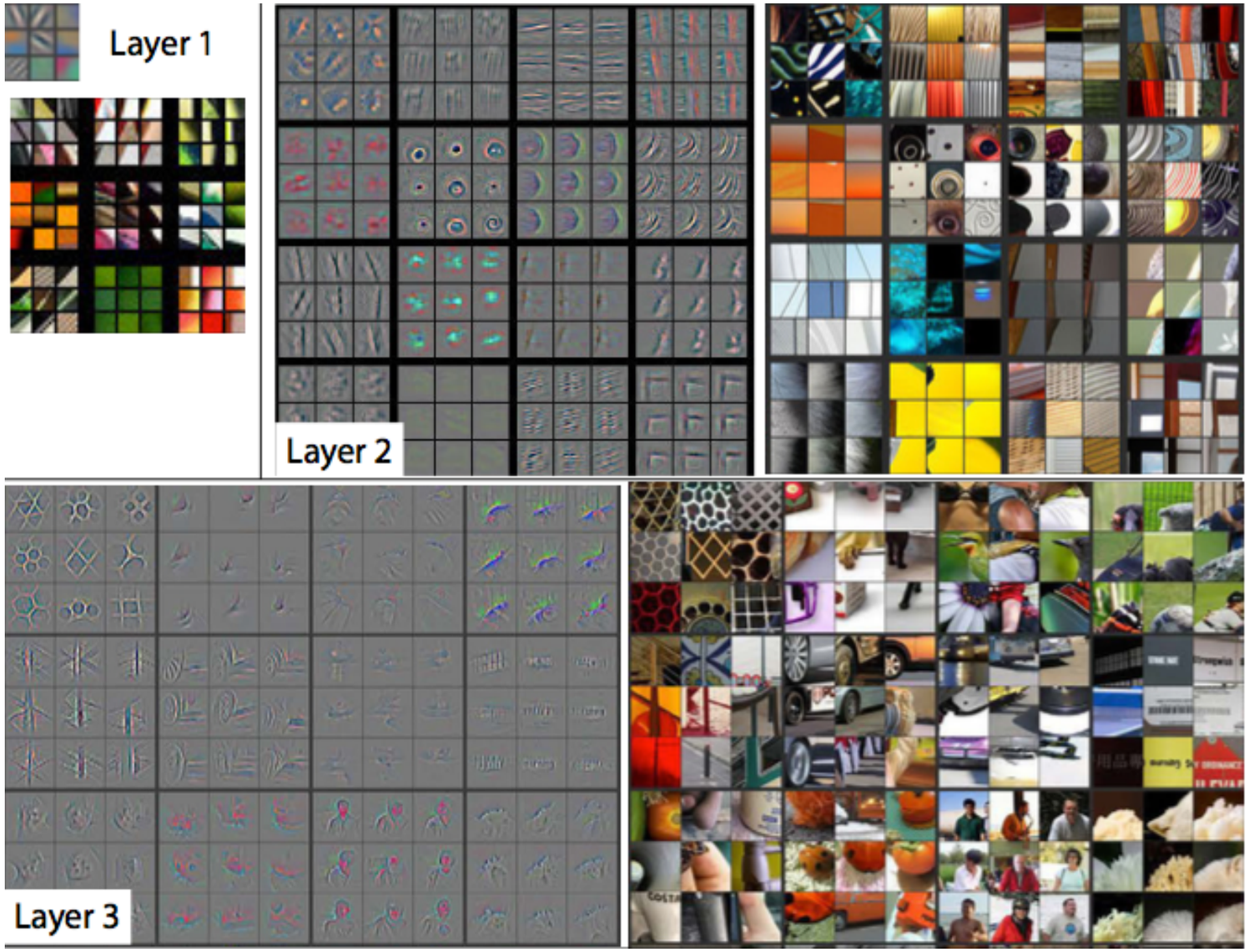
1x1 Convolutions

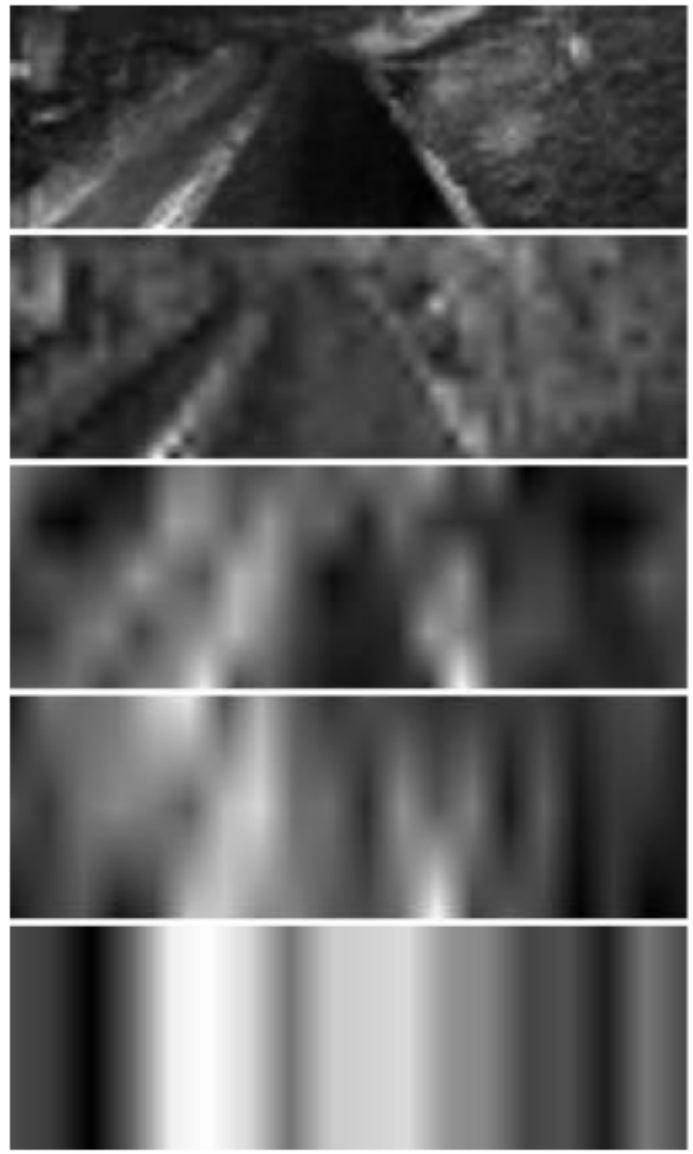


Min Lin, Qiang Chen, and Shuicheng Yan. Network in network. CoRR, abs/1312.4400, 2013.

Motivation

The most straightforward way of improving the performance of deep neural networks is by increasing their size.(**Depth**&**Width**)





---

[1] Zeiler, Matthew D., and Rob Fergus. "Visualizing and understanding convolutional networks." In *European conference on computer vision*, pp. 818-833. Springer, Cham, 2014.

[2] Bojarski, Mariusz, Philip Yeres, Anna Choromanska, Krzysztof Choromanski, Bernhard Firner, Lawrence Jackel, and Urs Muller. "Explaining How a Deep Neural Network Trained with End-to-End Learning Steers a Car." *arXiv preprint arXiv: 1704.07911* (2017).

Motivation

Two major backwards:

- the enlarged network more prone to overfitting
- increased use of computational resources

The fundamental way of solving both issues would be by ultimately moving from fully connected to **sparsely connected architectures**, even inside the convolutions.[1][2]

On the downside, todays computing infrastructures are very inefficient when it comes to numerical calculation on non-uniform sparse data structures.

---

[1] Szegedy, Christian, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. "Going deeper with convolutions." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1-9. 2015.

[2] Sanjeev Arora, Aditya Bhaskara, Rong Ge, and Tengyu Ma. Provable bounds for learning some deep representations. CoRR, abs/1310.6343, 2013

Motivation

Work before GoogLeNet

Most vision oriented machine learning systems before GoogLeNet utilize sparsity in the spatial domain just by the virtue of employing convolutions. However, convolutions are implemented as collections of dense connections to the patches in the earlier layer.

The proposed architecture—Inception

The Inception architecture tries to approximate a sparse structure implied by [1] for vision networks and covering the hypothesized outcome by dense, readily available components.

---

[1] Sanjeev Arora, Aditya Bhaskara, Rong Ge, and Tengyu Ma. Provable bounds for learning some deep representations. CoRR, abs/1310.6343, 2013.

Motivation

“We need go deeper” [1]

* By Google Search

---

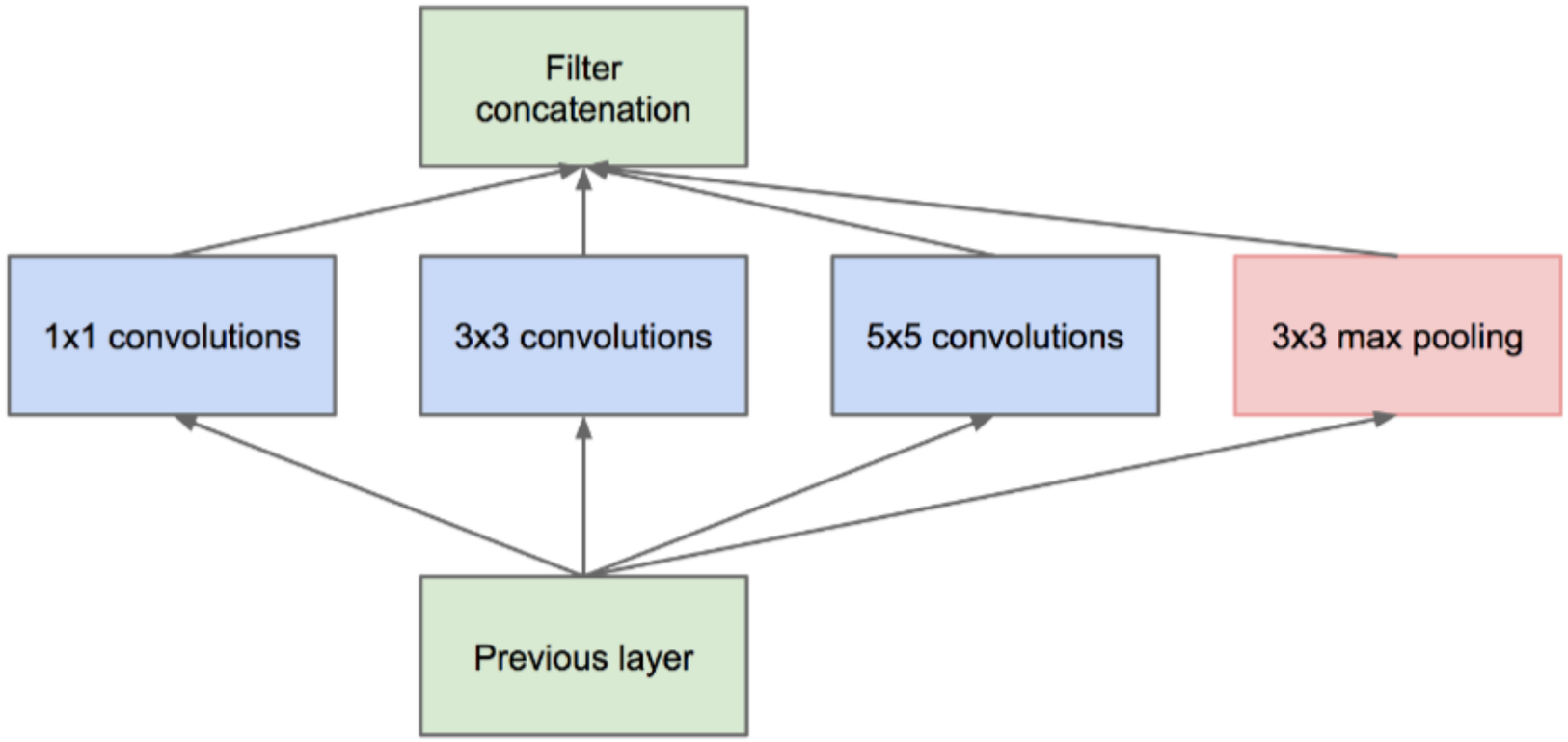
[1] Szegedy, Christian, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. "Going deeper with convolutions." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1-9. 2015.

Architectural Details

Inception V1

Graduate School of Information, Production and Systems
早稲田大学　大学院情報生産システム研究科

Architectural Details

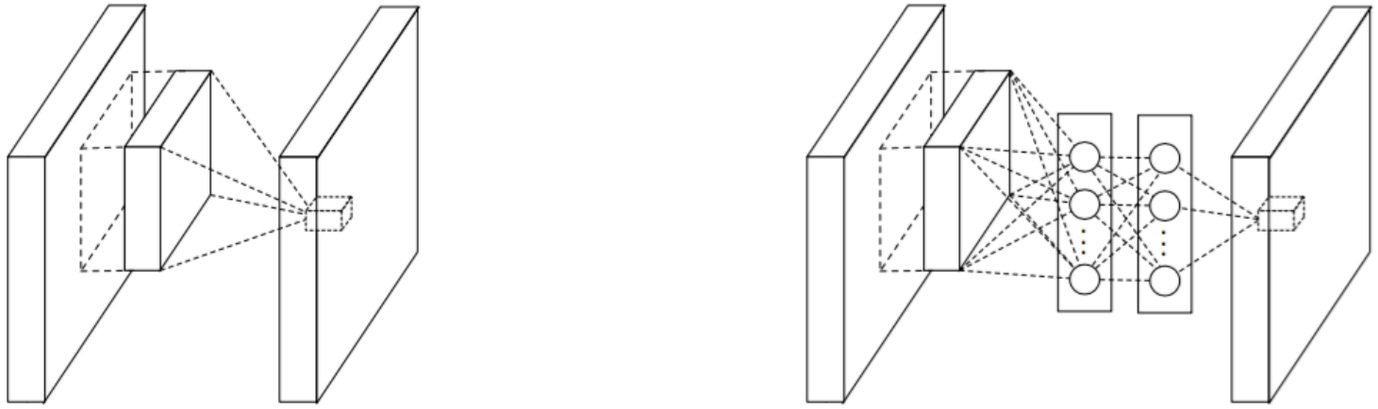


(a) Inception module, naïve version

Features:

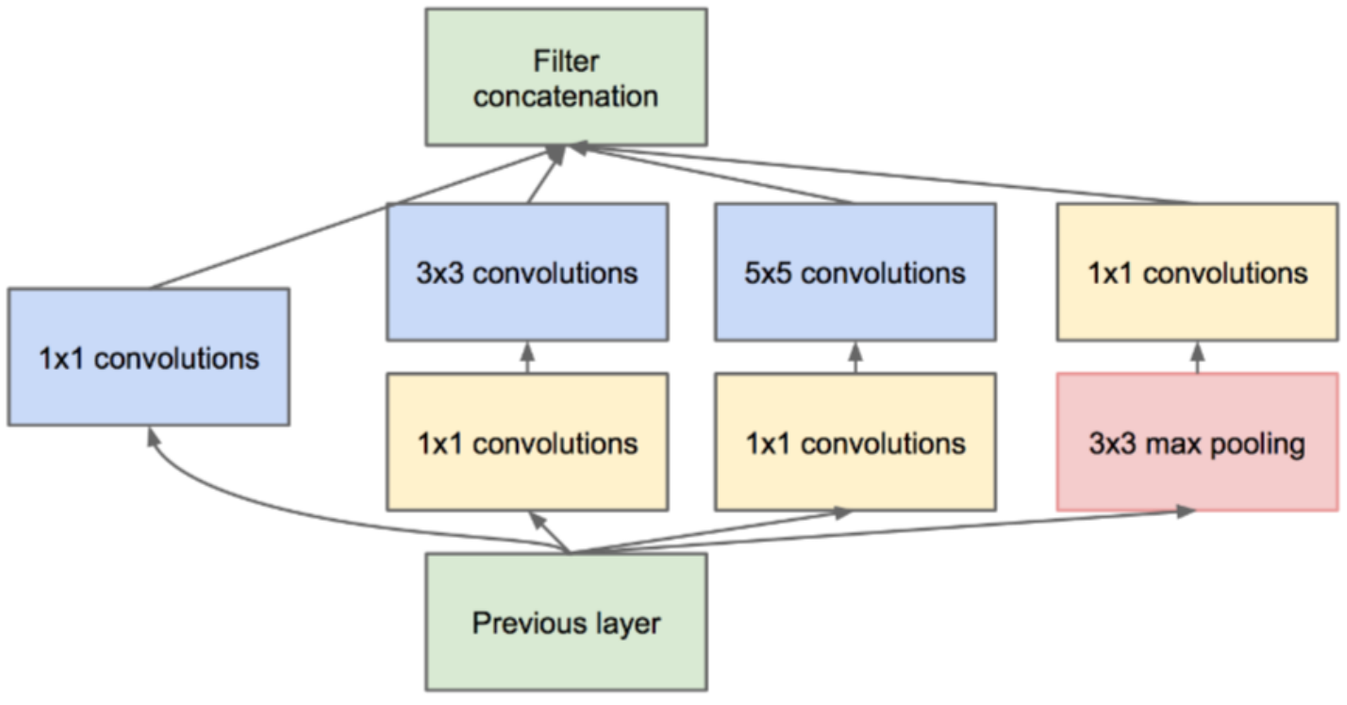
- Kernels with different size provide information in different scales;
- Kernel size 1x1,3x3 and 5x5 are easy to concatenation with pad= 0,1,2;
- The width of different kernel changing with depth;
- Pooling without Conv

Architectural Details



[1]

[2]



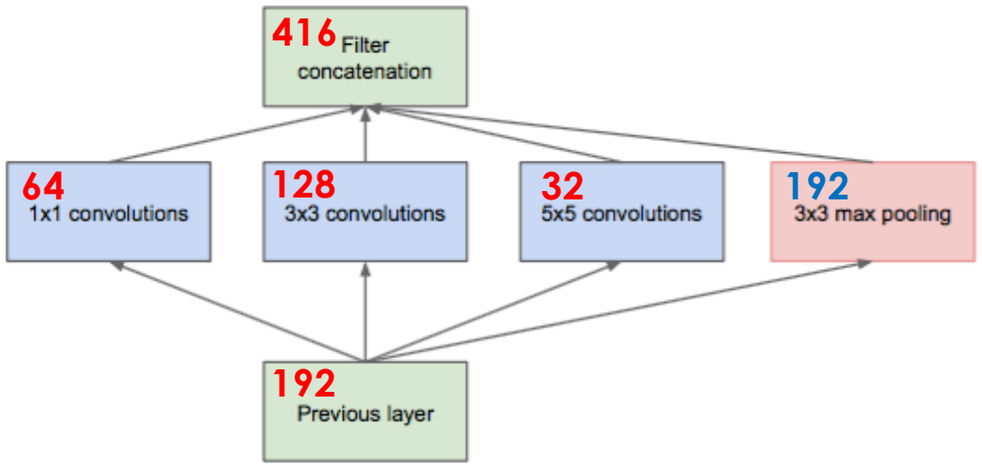
(b) Inception module with dimension reductions

---

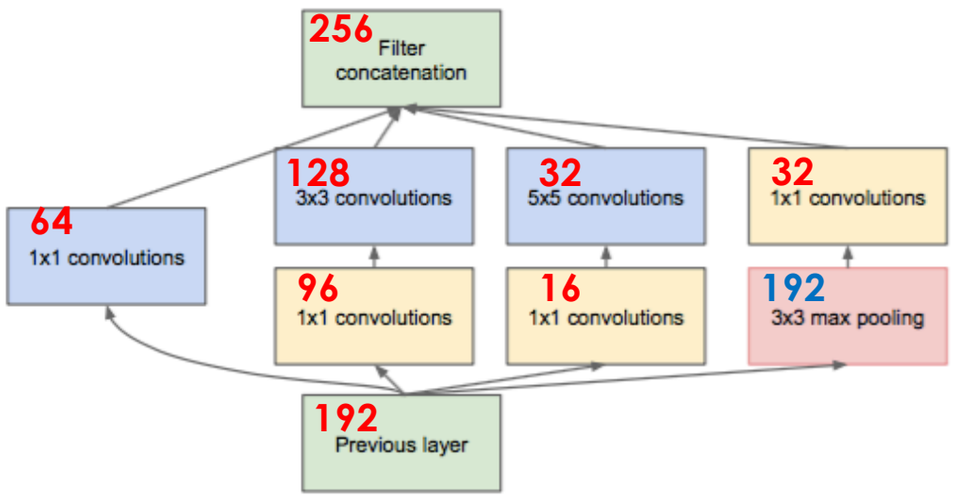
[1] Min Lin, Qiang Chen, and Shuicheng Yan. Network in network. CoRR, abs/1312.4400, 2013.

[2] Szegedy, Christian, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. "Going deeper with convolutions." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1-9. 2015.

Architectural Details



(a) Inception module, naïve version



(b) Inception module with dimensionality reduction

Number of weights:

(a) 64*192*1*1+128*192*3*3+32*192*5*5 = 387072 = **378K (without 1*1 convolution)**

(b)64*192*1*1+(96*192*1*1+128*96*3*3)+(16*192*1*1+32*16*5*5)+32*192*1*1 = 163328 = **159.5K(with 1*1 convolution)**

Min Lin, Qiang Chen, and Shuicheng Yan. Network in network. CoRR, abs/1312.4400, 2013.

Architectural Details

type	patch size/ stride	output size	depth	#1×1	#3×3 reduce	#3×3	#5×5 reduce	#5×5	pool proj	params	ops
convolution	7×7/2	112×112×64	1							2.7K	34M
max pool	3×3/2	56×56×64	0								
convolution	3×3/1	56×56×192	2		64	192				112K	360M
max pool	3×3/2	28×28×192	0								
inception (3a)		28×28×256	2	64	96	128	16	32	32	159K	128M
inception (3b)		28×28×480	2	128	128	192	32	96	64	380K	304M
max pool	3×3/2	14×14×480	0								
inception (4a)		14×14×512	2	192	96	208	16	48	64	364K	73M
inception (4b)		14×14×512	2	160	112	224	24	64	64	437K	88M
inception (4c)		14×14×512	2	128	128	256	24	64	64	463K	100M
inception (4d)		14×14×528	2	112	144	288	32	64	64	580K	119M
inception (4e)		14×14×832	2	256	160	320	32	128	128	840K	170M
max pool	3×3/2	7×7×832	0								
inception (5a)		7×7×832	2	256	160	320	32	128	128	1072K	54M
inception (5b)		7×7×1024	2	384	192	384	48	128	128	1388K	71M
avg pool	7×7/1	1×1×1024	0								
dropout (40%)		1×1×1024	0								
linear		1×1×1000	1							1000K	1M
softmax		1×1×1000	0								

Table 1: GoogLeNet incarnation of the Inception architecture

Architectural Details

The structure of GoogLeNet is as follow:

- An average pooling layer with 5×5 filter size and stride 3, resulting in an 4×4×512 output for the (4a), and 4×4×528 for the (4d) stage.
- A 1×1 convolution with 128 filters for dimension reduction and rectified linear activation.
- A fully connected layer with 1024 units and rectified linear activation.
- A dropout layer with 70% ratio of dropped outputs.
- A linear layer with softmax loss as the classifier (predicting the same 1000 classes as the main classifier, but removed at inference time).

Architectural Details

Inception V2 & V3

Graduate School of Information, Production and Systems
早稲田大学　大学院情報生産システム研究科

Architectural Details

[1] proposed a few design principles based on large-scale experimentation with various architectural choices with convolutional networks.

1, Avoid representational bottlenecks, especially early in the network.

2, Higher dimensional representations are easier to process locally within a network.

3, Spatial aggregation can be done over lower dimensional embeddings without much or any loss in representational power.

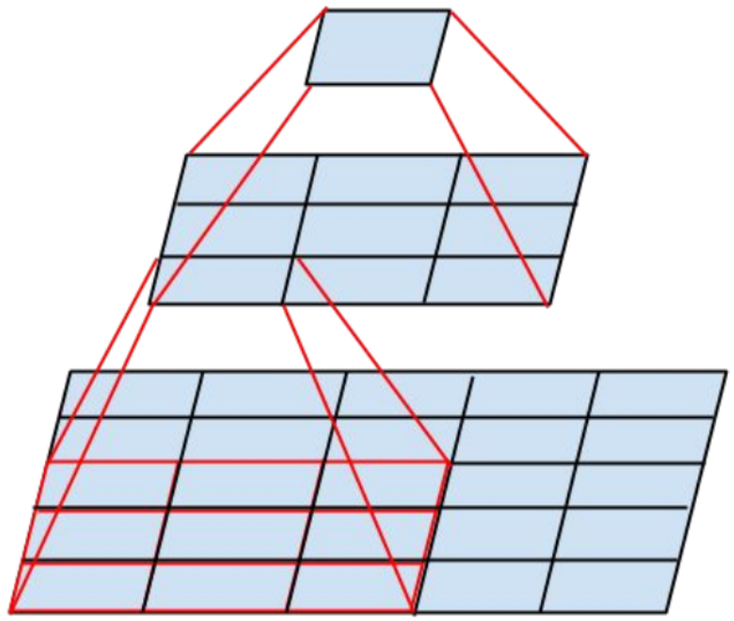
4, Balance the width and depth of the network. Optimal performance of the network can be reached by balancing the number of filters per stage and the depth of the network.

---

[1] Szegedy, Christian, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. "Rethinking the inception architecture for computer vision." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2818-2826. 2016.

Architectural Details

Factorization into smaller convolutions



Number of weights:

(a) 5x5 ConV

5 x 5 = **25**

(b) 3x3 ConV Concatenation

3 x 3 + 3 x 3 = **18**

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

---

[1] Szegedy, Christian, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. "Rethinking the inception architecture for computer vision." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2818-2826. 2016.

Architectural Details

Factorization into smaller convolutions

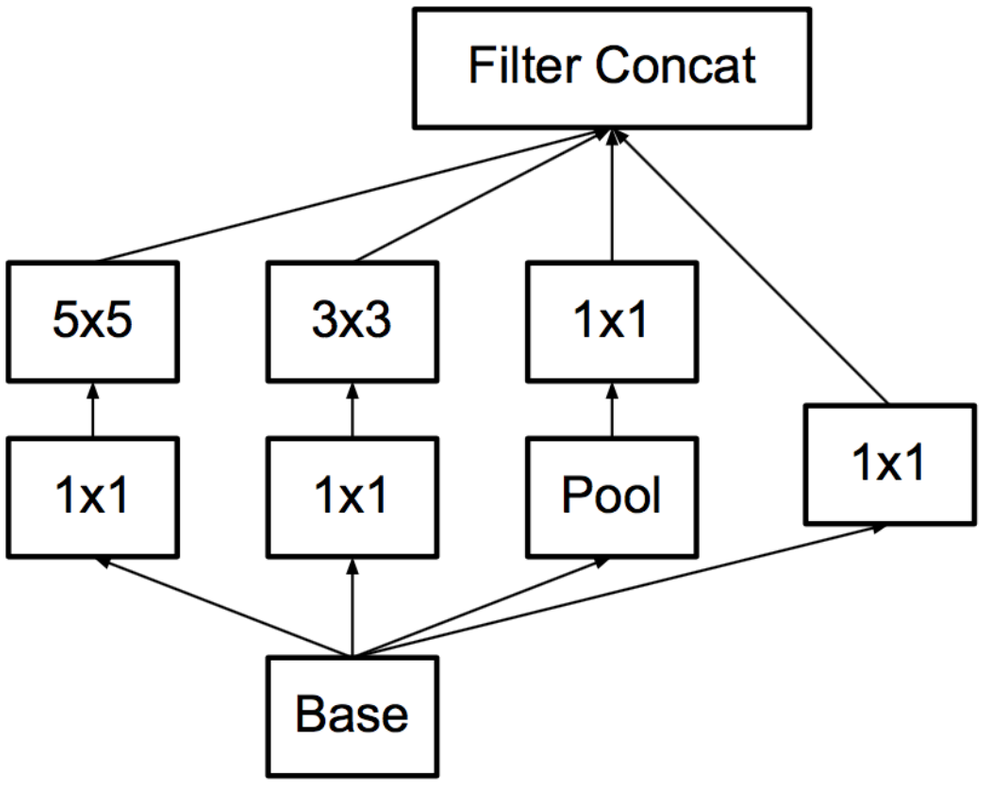


Figure 4. Original Inception module as described in [20].

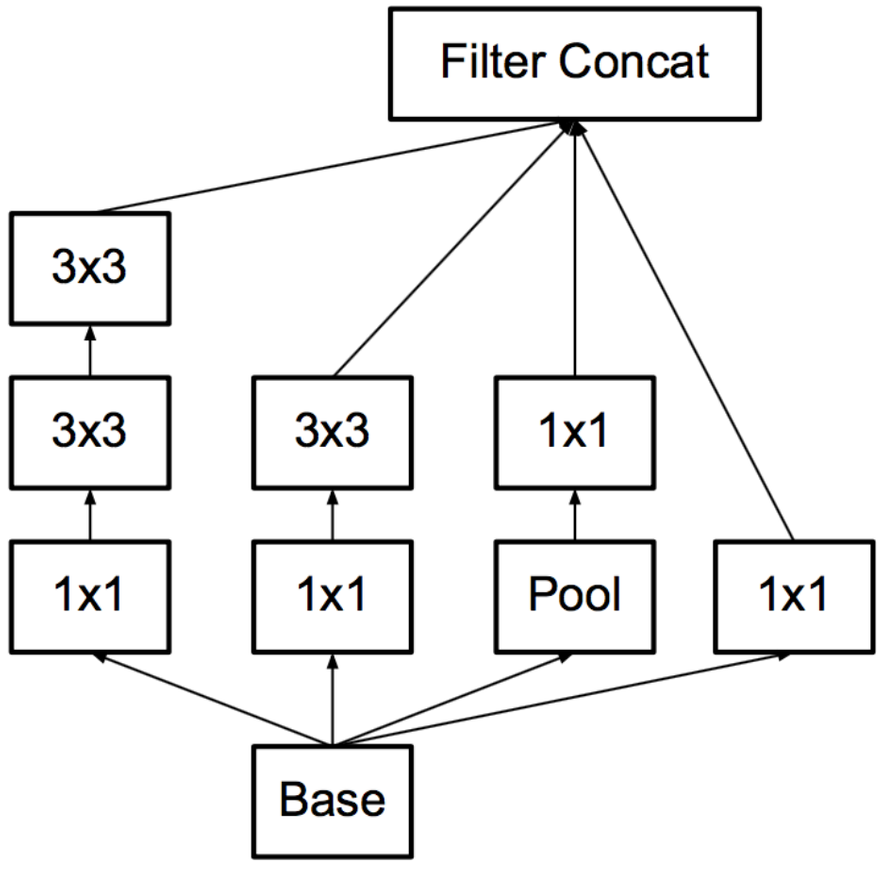


Figure 5. Inception modules where each 5×5 convolution is replaced by two 3×3 convolution, as suggested by principle 3 of Section 2.

---

[1] Szegedy, Christian, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. "Rethinking the inception architecture for computer vision." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2818-2826. 2016.

Architectural Details

Factorization into smaller convolutions

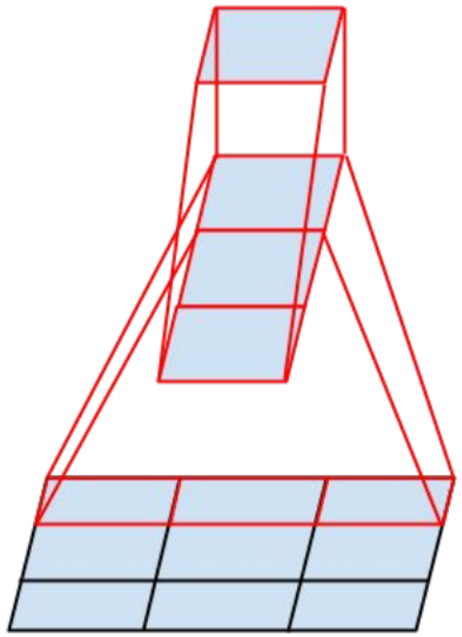


Figure 3. Mini-network replacing the 3×3 convolutions. The lower layer of this network consists of a 3×1 convolution with 3 output units.

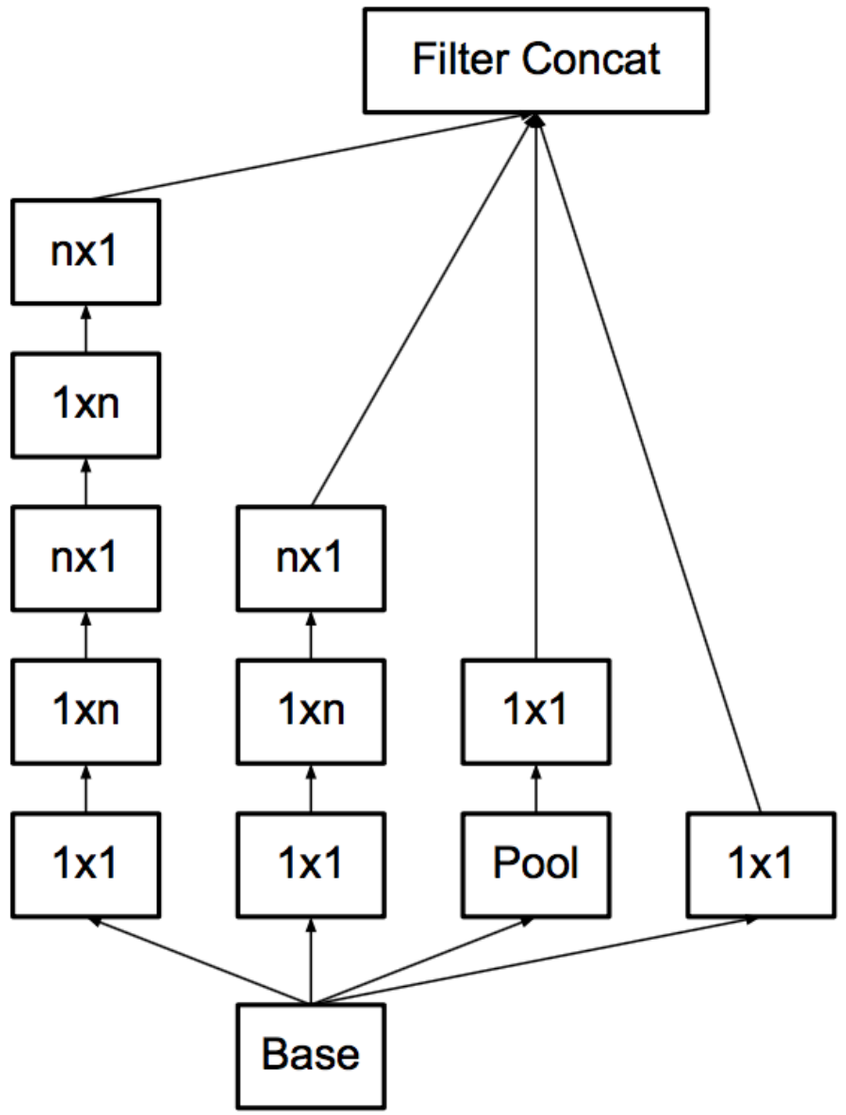


Figure 6. Inception modules after the factorization of the $n \times n$ convolutions. In our proposed architecture, we chose $n = 7$ for the 17×17 grid. (The filter sizes are picked using principle 3)

---

[1] Szegedy, Christian, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. "Rethinking the inception architecture for computer vision." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2818-2826. 2016.

Experimental Results and Comparisons

Network	Models Evaluated	Crops Evaluated	Top-1 Error	Top-5 Error
VGGNet [18]	2	-	23.7%	6.8%
GoogLeNet [20]	7	144	-	6.67%
PReLU [6]	-	-	-	4.94%
BN-Inception [7]	6	144	20.1%	4.9%
Inception-v3	4	144	**17.2%**	**3.58%** *

Table 5. Ensemble evaluation results comparing multi-model, multi-crop reported results. Our numbers are compared with the best published ensemble inference results on the ILSVRC 2012 classification benchmark. *All results, but the top-5 ensemble result reported are on the validation set. The ensemble yielded 3.46% top-5 error on the validation set.

---

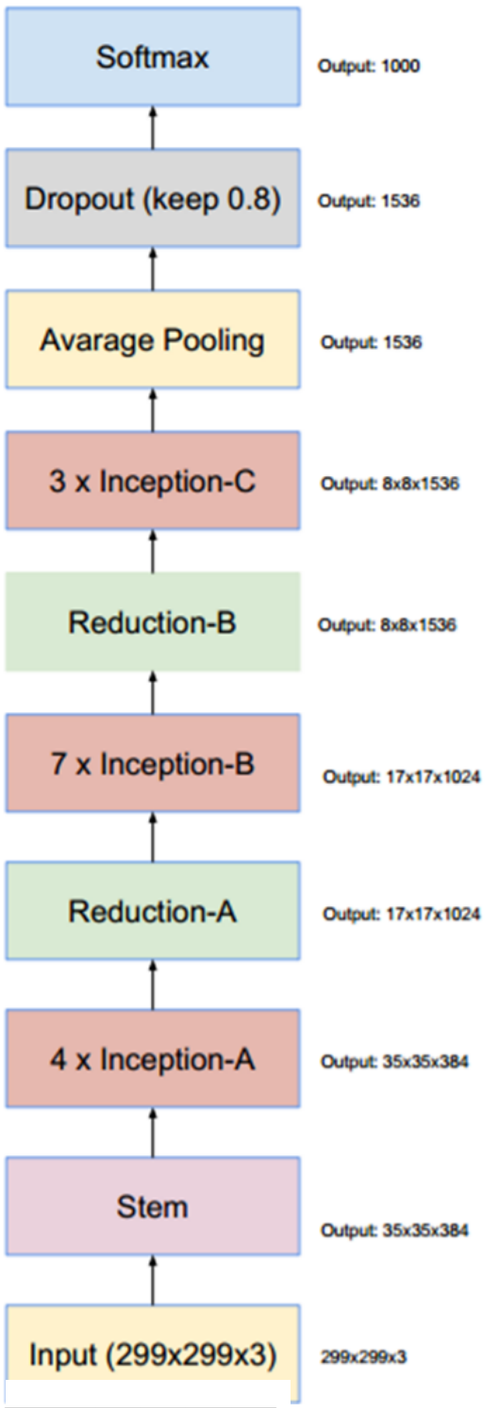
[1] Szegedy, Christian, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. "Rethinking the inception architecture for computer vision." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2818-2826. 2016.

Architectural Details

The Updating

Graduate School of Information, Production and Systems
早稲田大学　大学院情報生産システム研究科

The Updating



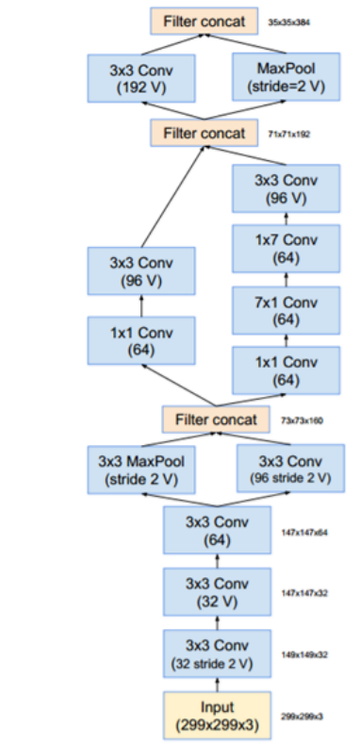


Figure 3. The schema for stem of the pure Inception-v4 and Inception-ResNet-v2 networks. This is the input part of those networks. Cf. Figures 9 and 15

Inception-V4

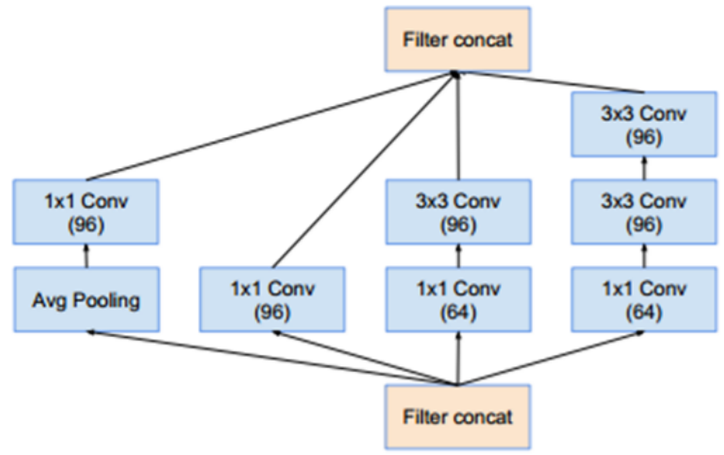


Figure 4. The schema for 35 × 35 grid modules of the pure Inception-v4 network. This is the Inception-A block of Figure 9.

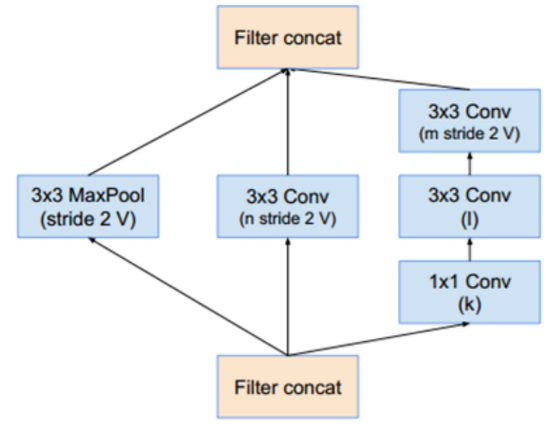


Figure 7. The schema for 35 × 35 to 17 × 17 reduction module. Different variants of this blocks (with various number of filters) are used in Figure 9, and 15 in each of the new Inception(-v4, -ResNet-v1, -ResNet-v2) variants presented in this paper. The k, l, m, n numbers represent filter bank sizes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

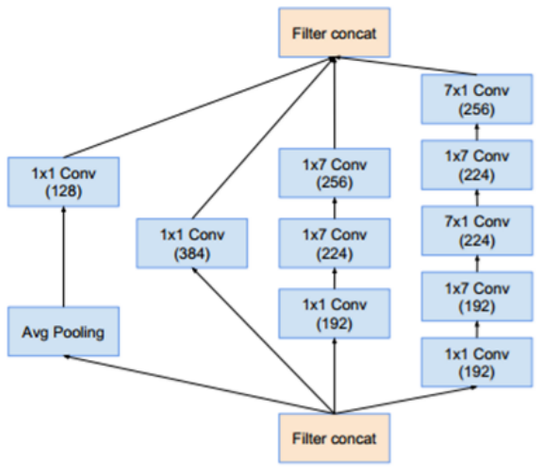


Figure 5. The schema for 17 × 17 grid modules of the pure Inception-v4 network. This is the Inception-B block of Figure 9.

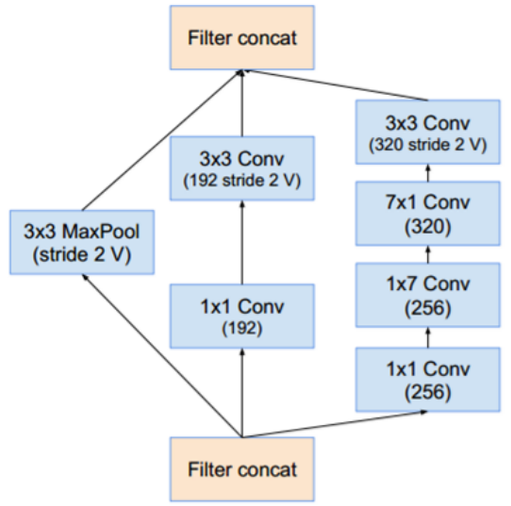


Figure 8. The schema for 17 × 17 to 8 × 8 grid-reduction module. This is the reduction module used by the pure Inception-v4 network in Figure 9.

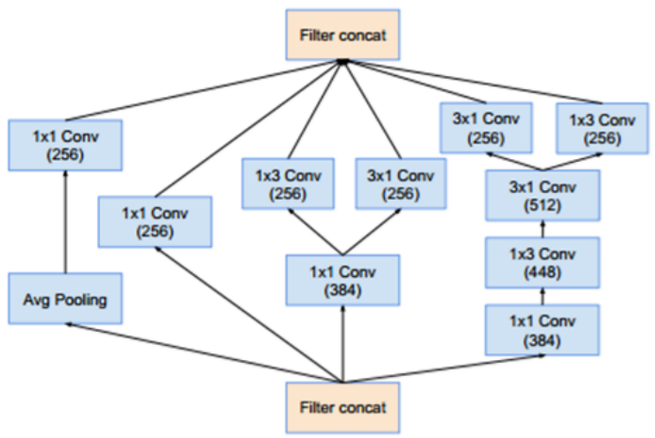
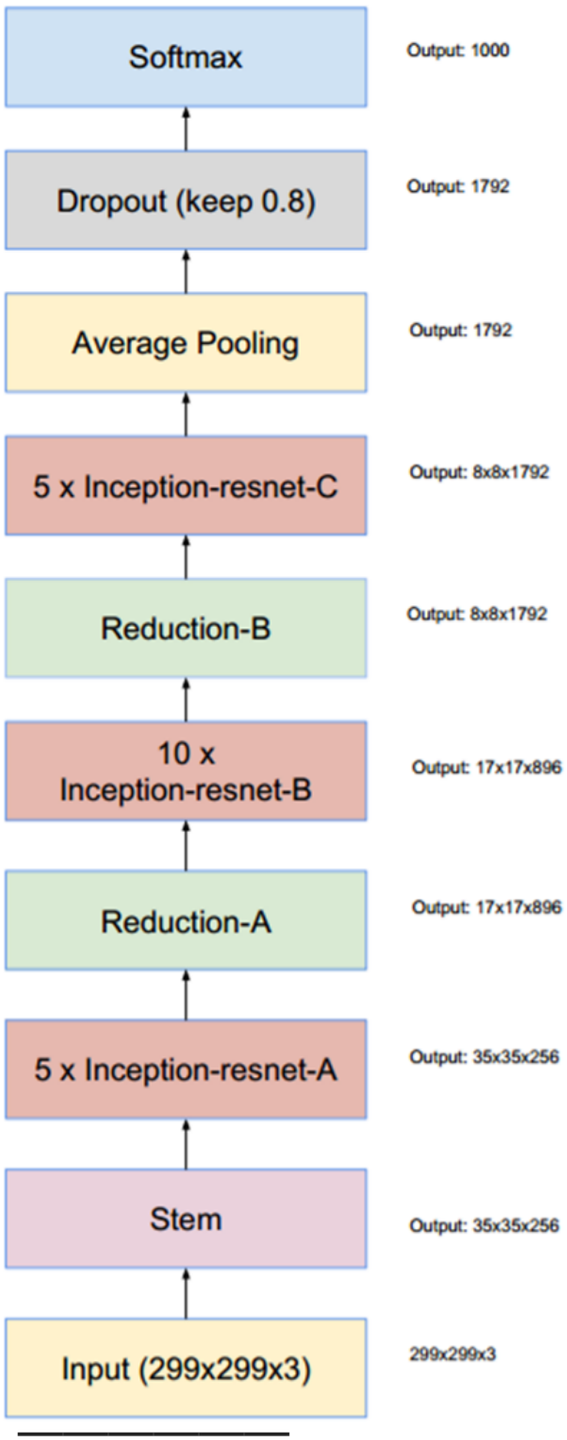


Figure 6. The schema for 8 × 8 grid modules of the pure Inception-v4 network. This is the Inception-C block of Figure 9.

Szegedy, Christian, Sergey Ioffe, Vincent Vanhoucke, and Alexander A. Alemi. "Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning." In *AAAI*, pp. 4278-4284. 2017.

The Updating



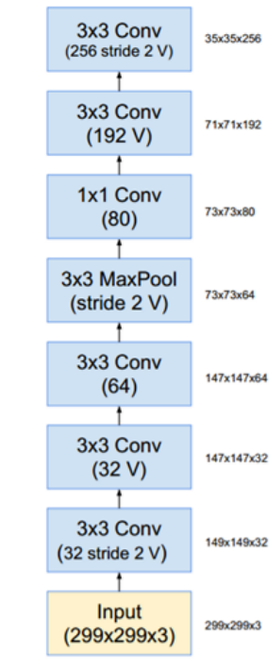


Figure 14. The stem of the Inception-ResNet-v1 network.

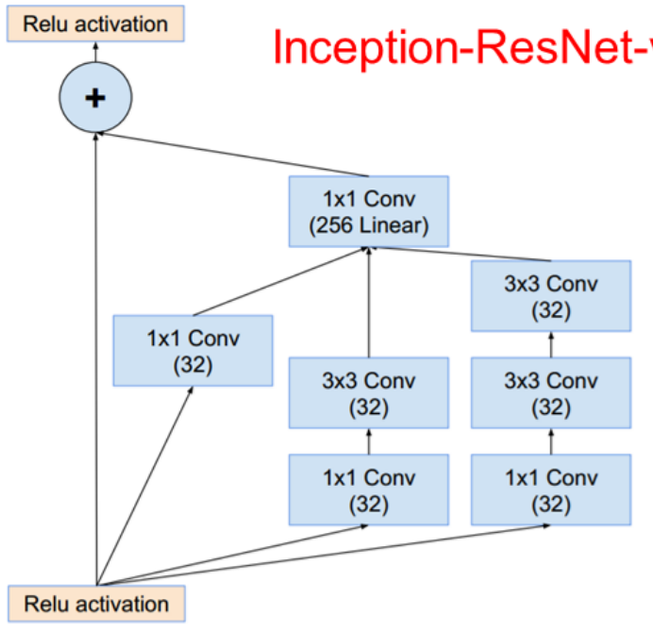


Figure 10. The schema for 35×35 grid (Inception-ResNet-A) module of Inception-ResNet-v1 network.

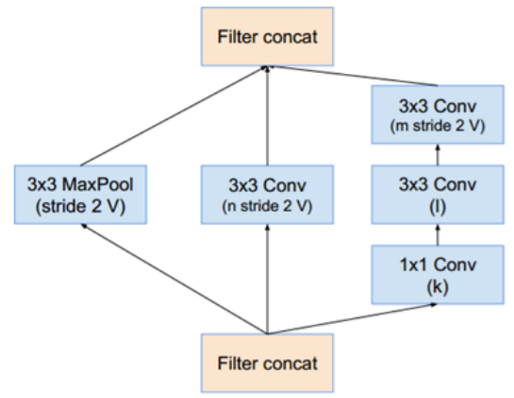


Figure 7. The schema for 35×35 to 17×17 reduction module. Different variants of this blocks (with various number of filters) are used in Figure 9, and 15 in each of the new Inception(-v4, -ResNet-v1, -ResNet-v2) variants presented in this paper. The k, l, m, n numbers represent filter bank sizes which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

Network	k	l	m	n
Inception-v4	192	224	256	384
Inception-ResNet-v1	192	192	256	384
Inception-ResNet-v2	256	256	384	384

Table 1. The number of filters of the Reduction-A module for the three Inception variants presented in this paper. The four numbers in the colums of the paper parametrize the four convolutions of Figure 7.

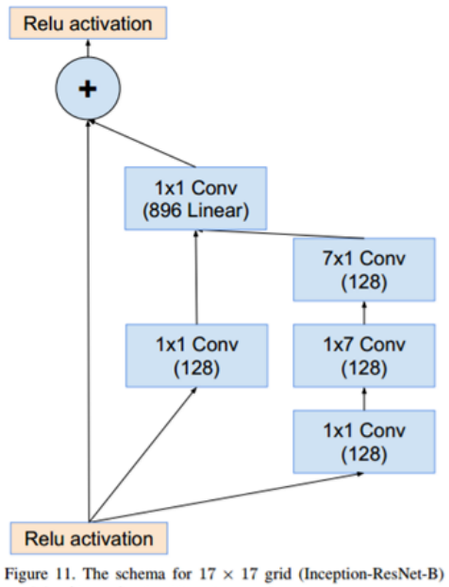


Figure 11. The schema for 17×17 grid (Inception-ResNet-B) module of Inception-ResNet-v1 network.

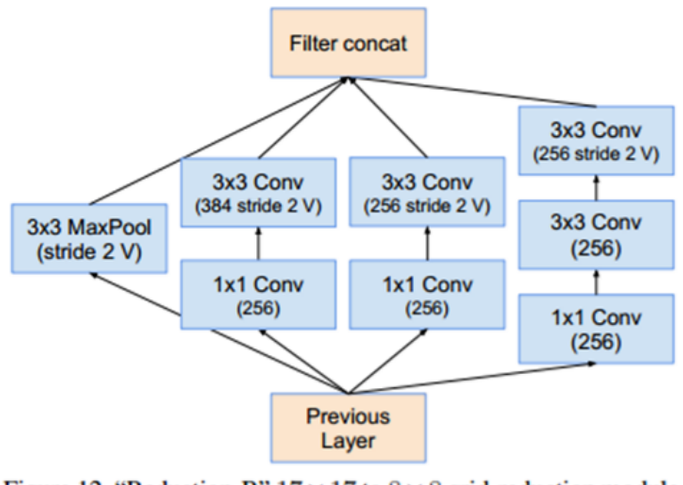


Figure 12. "Reduction-B" 17×17 to 8×8 grid-reduction module. This module used by the smaller Inception-ResNet-v1 network in Figure 15.

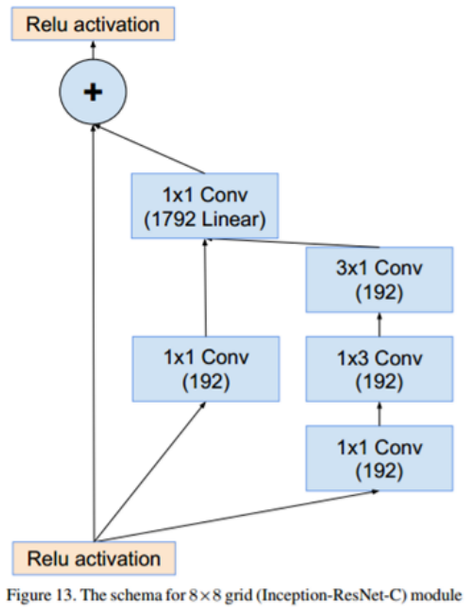
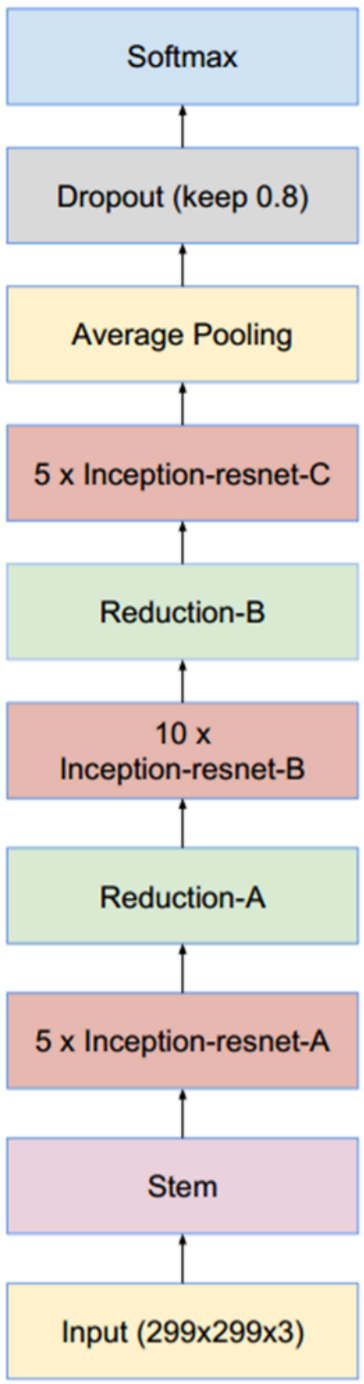


Figure 13. The schema for 8×8 grid (Inception-ResNet-C) module of Inception-ResNet-v1 network.

Szegedy, Christian, Sergey Ioffe, Vincent Vanhoucke, and Alexander A. Alemi. "Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning." In *AAAI*, pp. 4278-4284. 2017.

The Updating



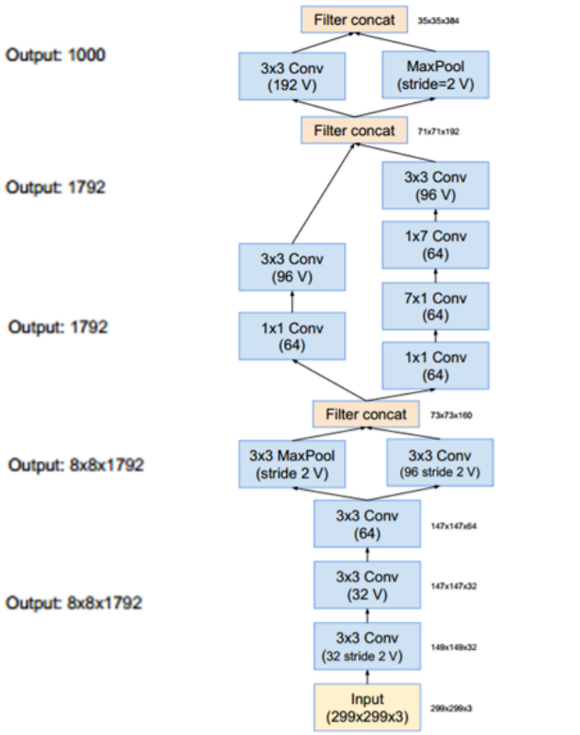


Figure 3. The schema for stem of the pure Inception-v4 and Inception-ResNet-v2 networks. This is the input part of those networks. Cf. Figures 9 and 15

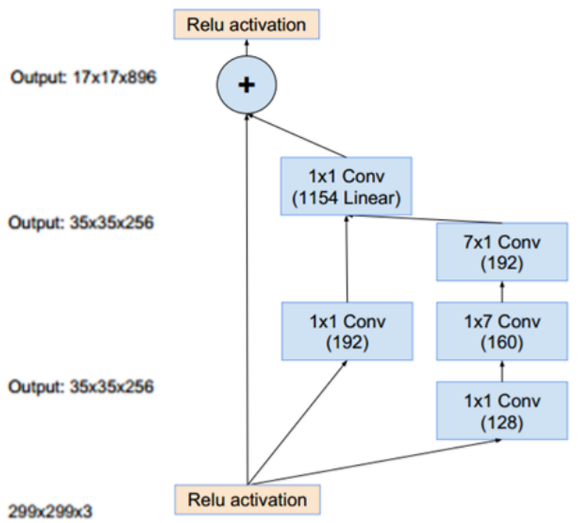


Figure 17. The schema for 17 × 17 grid (Inception-ResNet-B) module of the Inception-ResNet-v2 network.

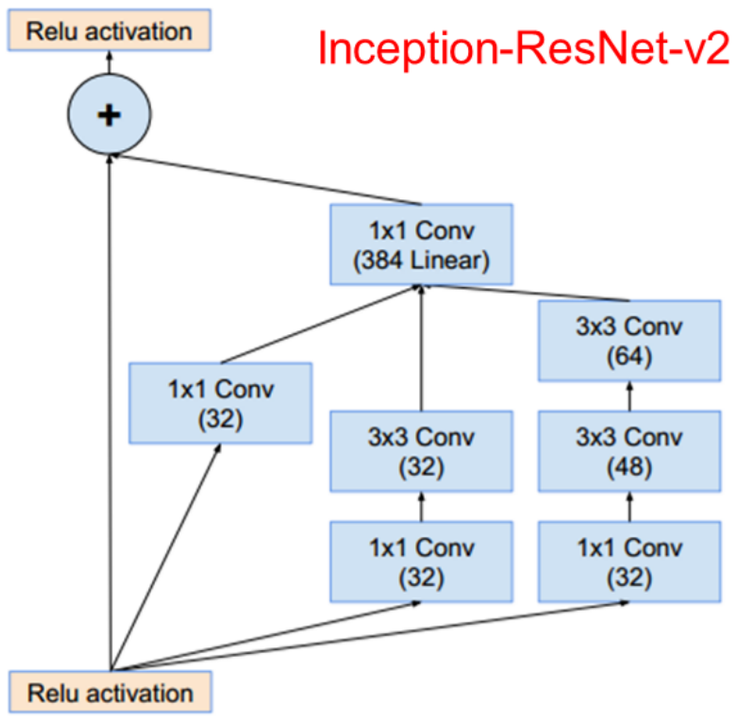


Figure 16. The schema for 35 × 35 grid (Inception-ResNet-A) module of the Inception-ResNet-v2 network.

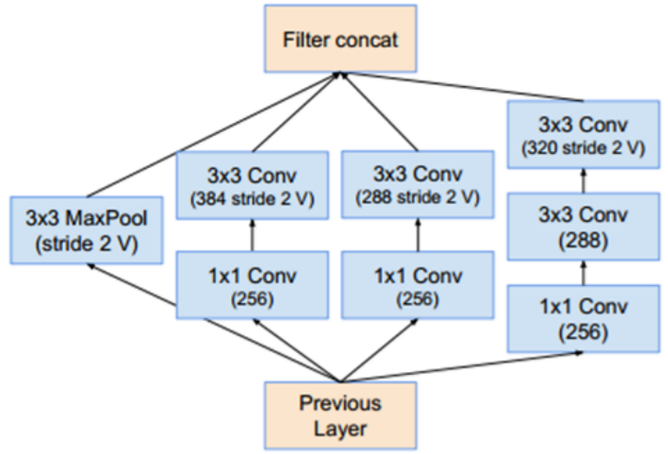


Figure 18. The schema for 17 × 17 to 8 × 8 grid-reduction module. Reduction-B module used by the wider Inception-ResNet-v1 network in Figure 15.

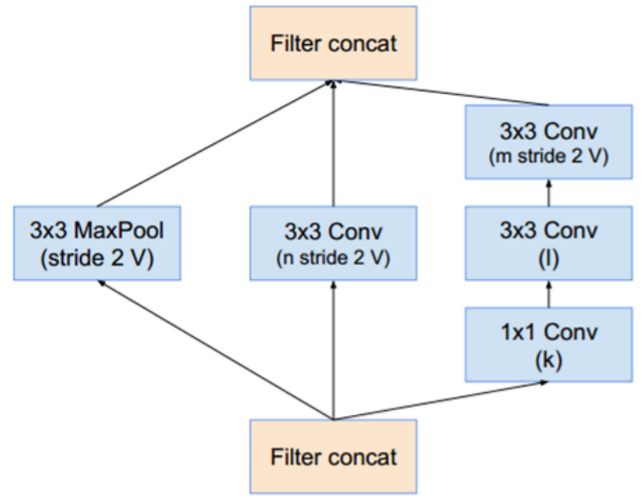


Figure 7. The schema for 35 × 35 to 17 × 17 reduction module. Different variants of this blocks (with various number of filters) are used in Figure 9, and 15 in each of the new Inception(-v4, -ResNet-v1, -ResNet-v2) variants presented in this paper. The k, l, m, n numbers represent filter bank sizes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

Network	k	l	m	n
Inception-v4	192	224	256	384
Inception-ResNet-v1	192	192	256	384
Inception-ResNet-v2	256	256	384	384

Table 1. The number of filters of the Reduction-A module for the three Inception variants presented in this paper. The four numbers in the colums of the paper parametrize the four convolutions of Figure 7.

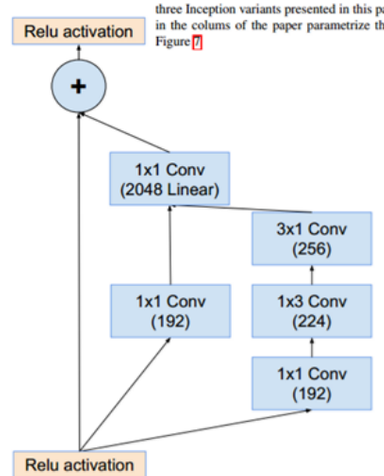


Figure 19. The schema for 8×8 grid (Inception-ResNet-C) module of the Inception-ResNet-v2 network.

Szegedy, Christian, Sergey Ioffe, Vincent Vanhoucke, and Alexander A. Alemi. "Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning." In *AAAI*, pp. 4278-4284. 2017.

The Updates

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.

Szegedy, Christian, Sergey Ioffe, Vincent Vanhoucke, and Alexander A. Alemi. "Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning." In *AAAI*, pp. 4278-4284. 2017.