Famous Image Classification Architectures

Presented By
Mohammad Hossein
Niki Maleki

Summer 2021

مرکز تحقیقات هوش مصنوعی پارت
هوشمندسازی فرایندهای زندگی

کالج تخصصی هوش مصنوعی پارت

Part1

Typical Image_Classification

1.1

Image Classification,

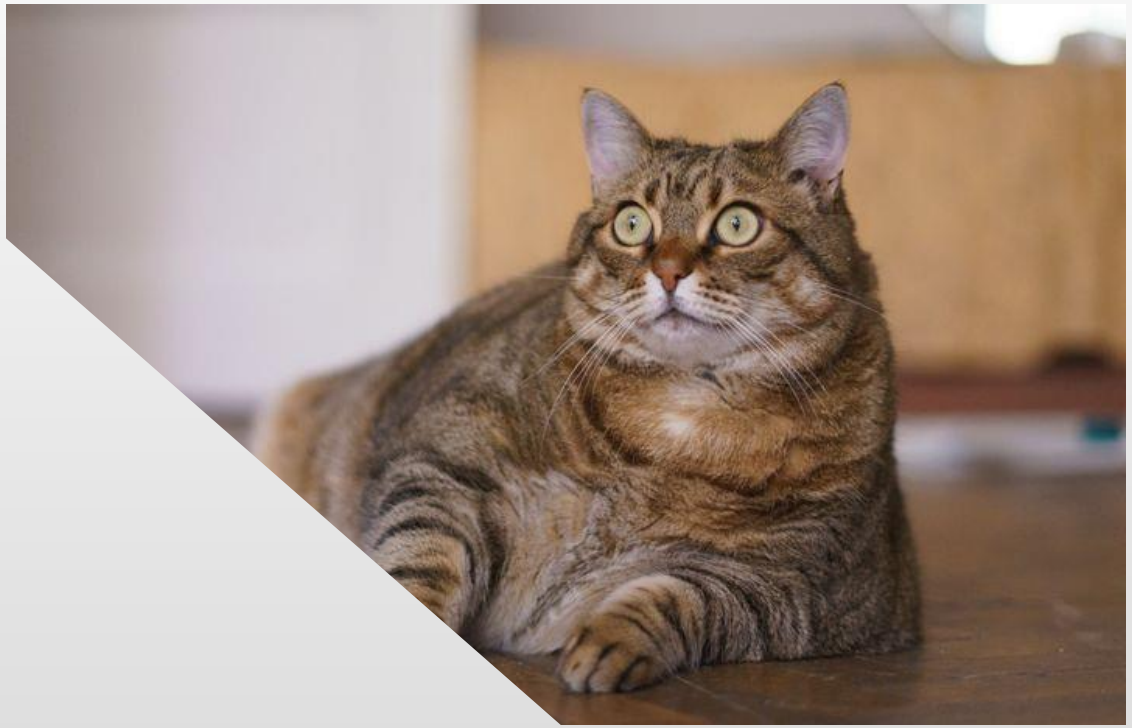
Overview,

History,

Challenges

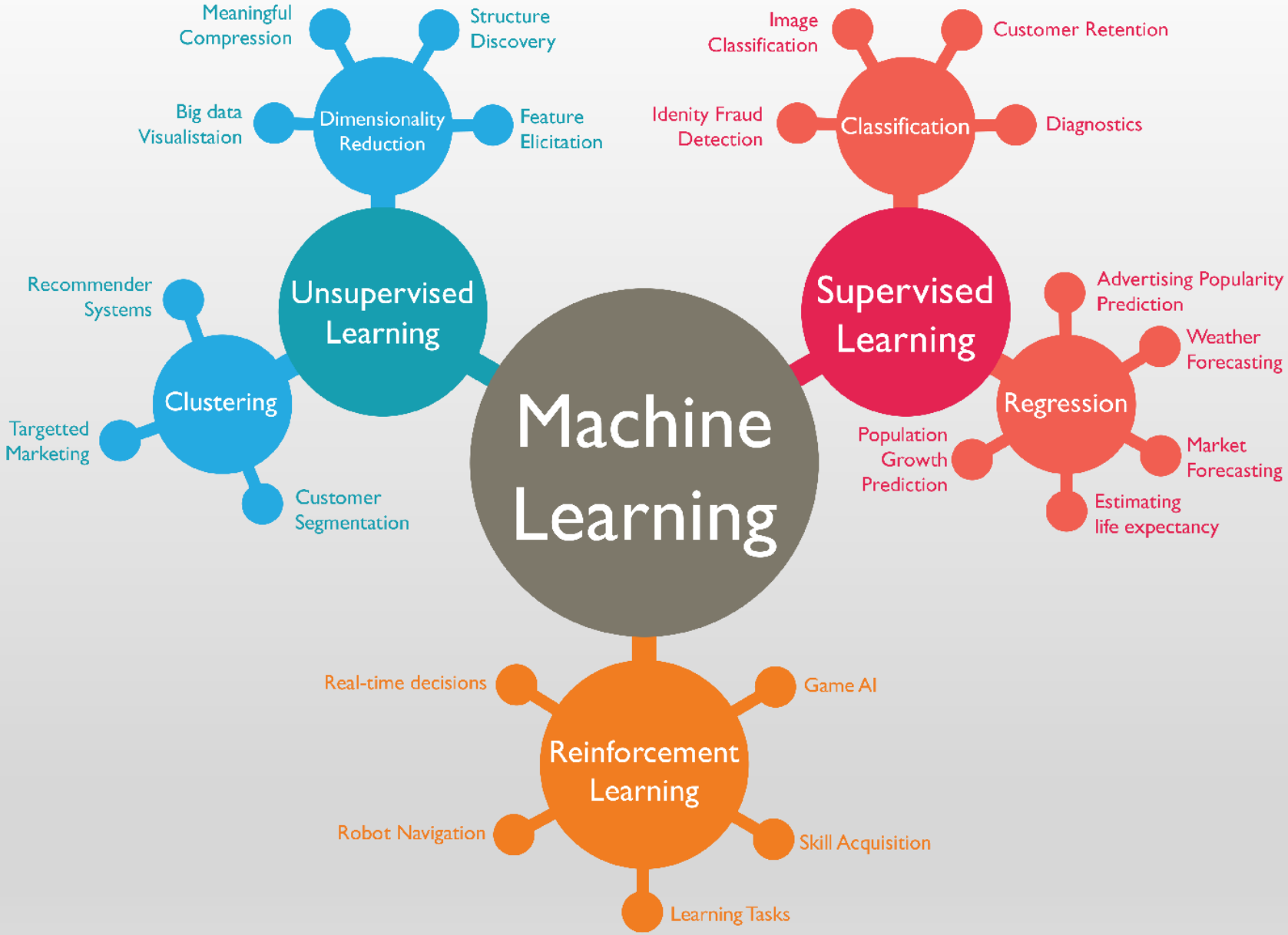
Image Classification

Application



- A core task in machine learning and computer vision.
- Assume given set of discrete labels and datasets; { dog, cat, truck, plane, ... }

" CAT"


Machine Learning
Unsupervised Learning
Dimensionality Reduction
Meaningful Compression
Structure Discovery
Big data Visualistaion
Feature Elicitation
Clustering
Recommender Systems
Targetted Marketing
Customer Segmentation
Supervised Learning
Classification
Image Classification
Customer Retention
Idenity Fraud Detection
Diagnostics
Regression
Advertising Popularity Prediction
Weather Forecasting
Market Forecasting
Estimating life expectancy
Population Growth Prediction
Reinforcement Learning
Real-time decisions
Game AI
Robot Navigation
Skill Acquisition
Learning Tasks

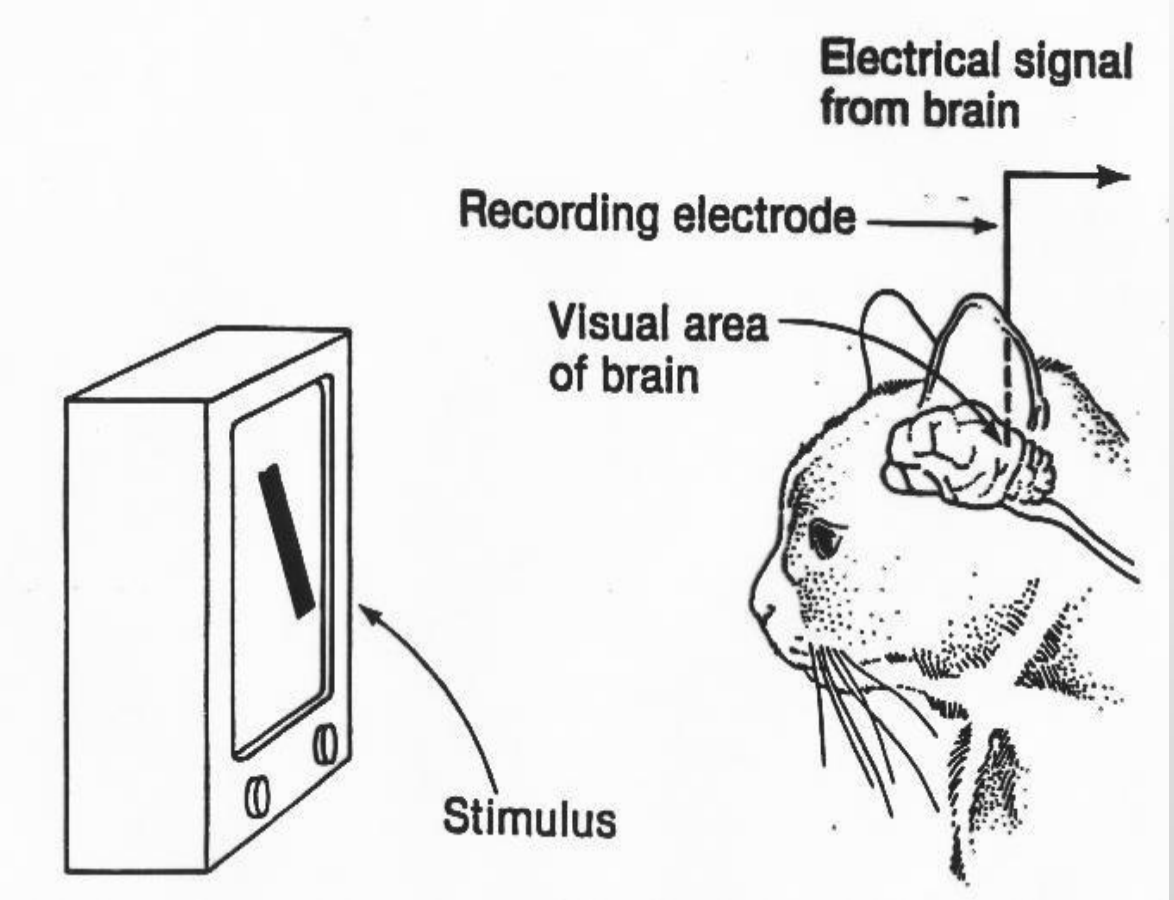
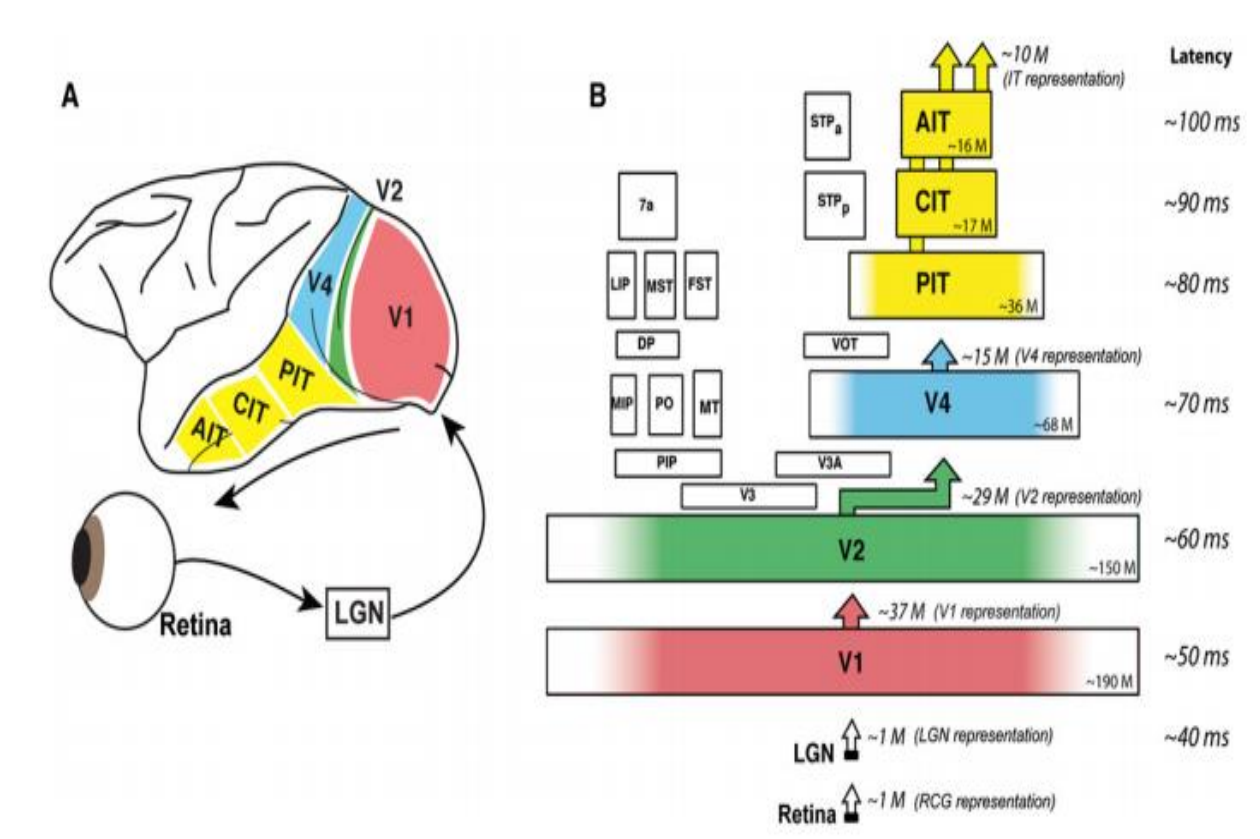


Image Classification Journey

- One of the most influential papers in Computer Vision was published by two neurophysiologists (David et al., 1959)
- They placed electrodes into the primary visual cortex area of an anesthetized cat's brain and observed, the neuronal activity in that region while showing the animal various images.
- There are simple and complex neurons in the primary visual cortex and that visual processing always starts with simple structures such as oriented edges. Sounds familiar?

Image Classification Journey

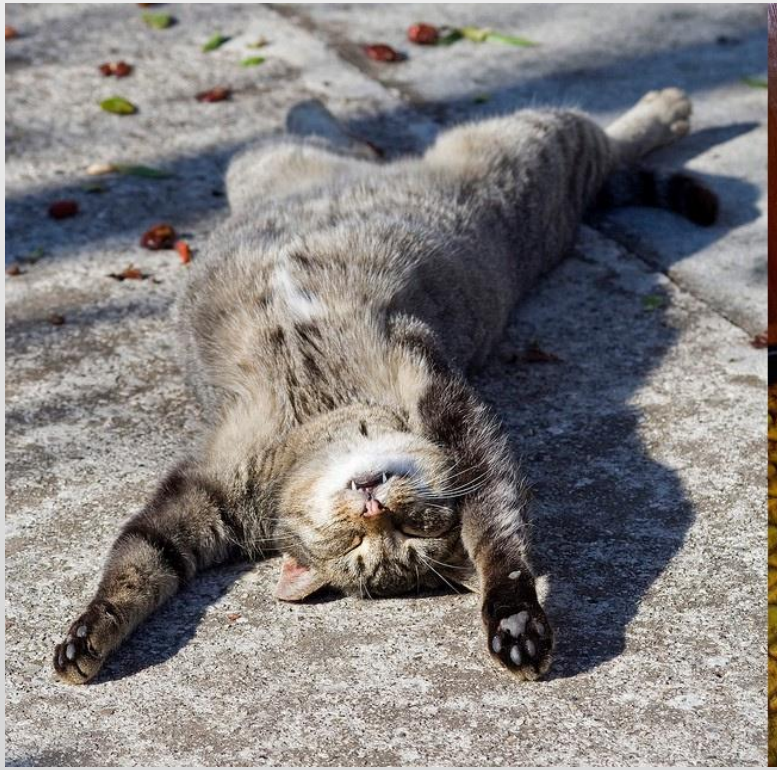


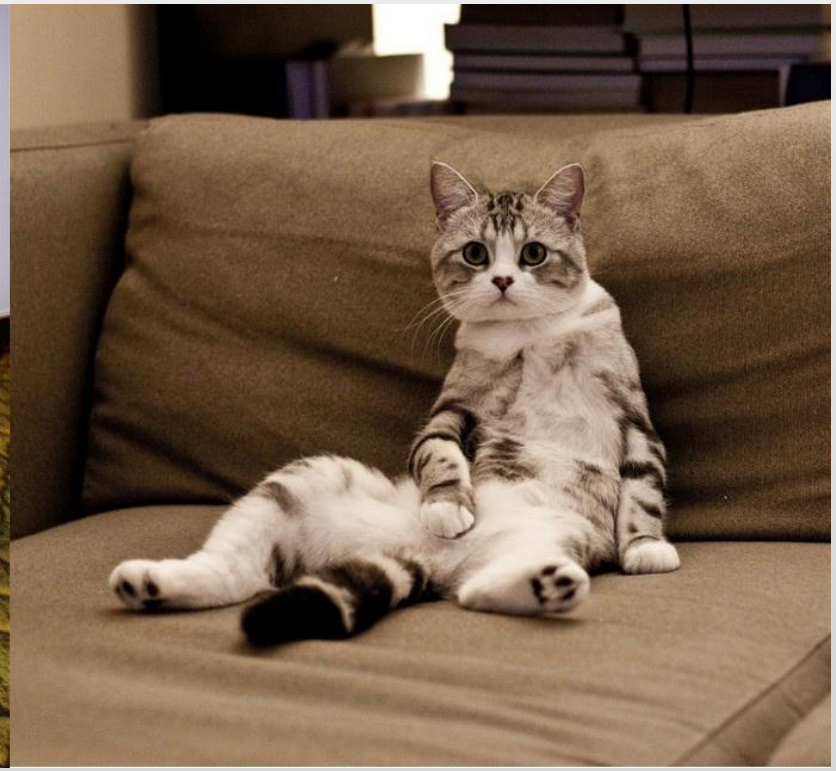
- Building on the ideas of Hubel and Wiesel David gave us the next important insight: He established that vision is hierarchical (David et al., 1983).
- Brain object recognition and visualize system (Dicarlo et al., 2012).

Image classification challenges

1. Illumination:

2. Deformation





3. Viewpoint Variation:

4. Occlusion:



5. Background Clutter:

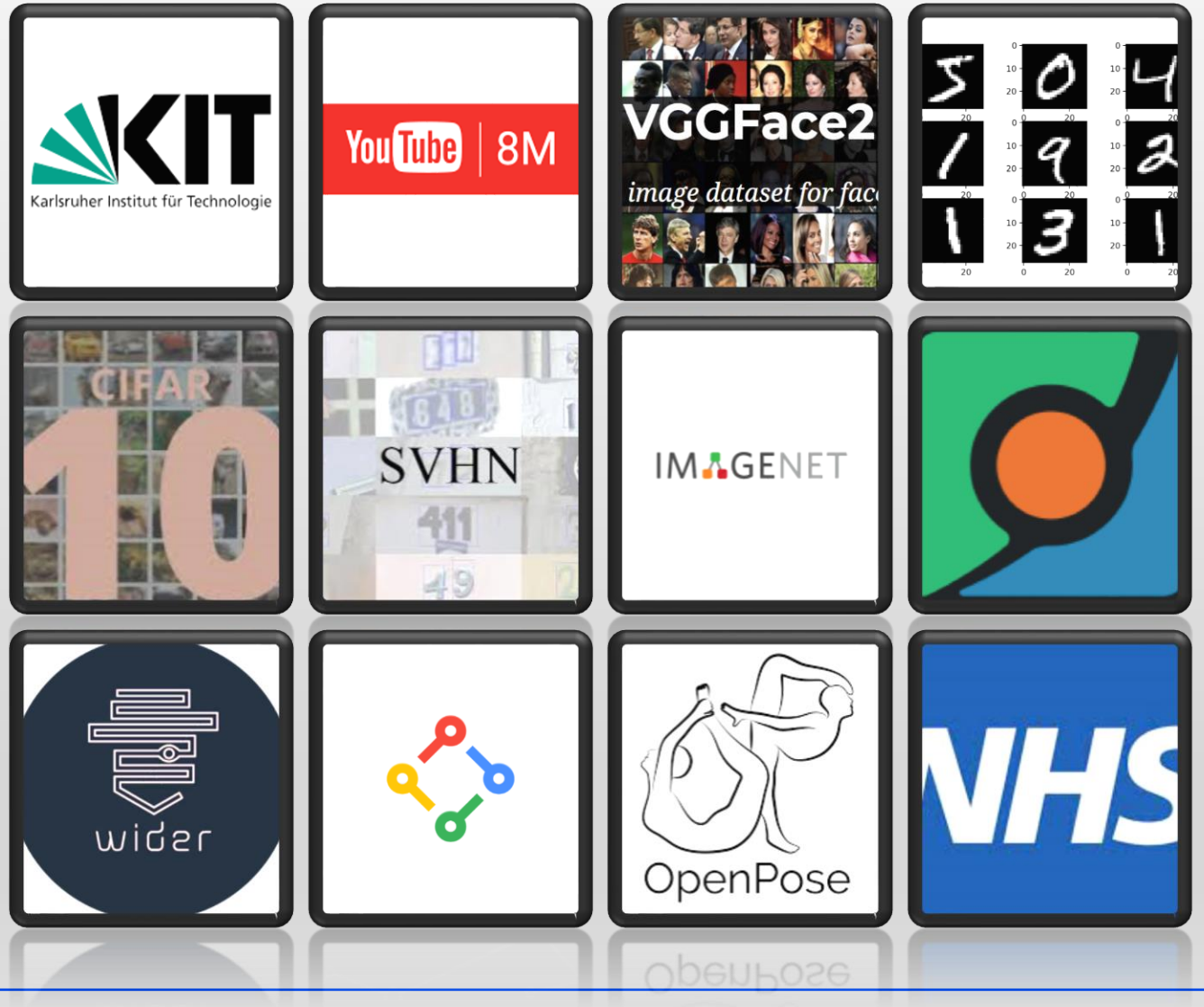
6. Scale Variation:

7. Intraclass Variation:

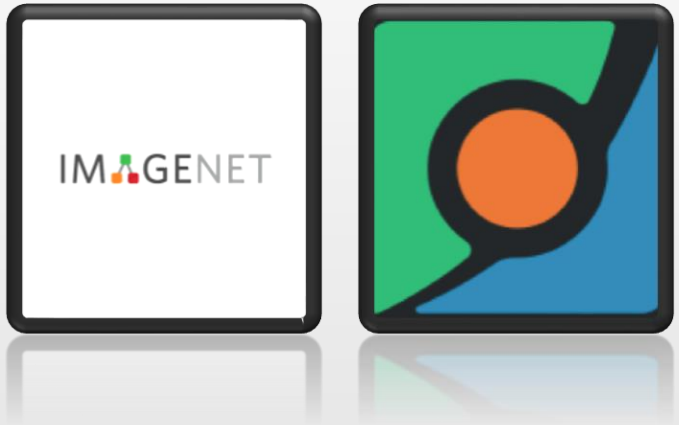
Part1

Typical Image_Classification

1.2 Image Classification Datasets



Object Recognition – Detection



- **MS-COCO**: Is a large-scale object detection, segmentation, and captioning dataset; 330K images, 80 object categories, 5 captions per image (Lin et al., 2014).
- **Imagenet**: Image classification and localization dataset. This dataset spans 1000 object classes and contains 1,281,167 training images, 50,000 validation images and 100,000 test images (Deng et al., 2009).

Object Recognition – Detection

- **CIFAR**: The CIFAR-10 dataset consists of 60000 32x32 colour images in 10 classes, with 6000 images per class. There are 50000 training images and 10000 test images; CIFAR-100 has 100 classes containing 600 images each (Krizhevsky et al., 2009).
- **OPEN IMAGES**: 9M images annotated with 36M image-level labels, bounding boxes, instance segmentations, and visual relationships (Kuznetsova et al., 2020).

Optical Character Recognition

- **IAM**: An English sentence database for offline handwriting recognition. The IAM database contains 13,353 images of handwritten lines of text created by 657 writers and 115'320 isolated and labeled words (Urs-Viktor Marti et al., 2002).

Optical Character Recognition

- **SVHN:** 600000 32×32 RGB images of printed digits from 0 to 9 (Netzer et al., 2011).
- **MNIST:** It has a training set of 60,000 examples, and a test set of 10,000 examples; the images were centered in a 28x28 (LeCun et al. 1998).

Face Detection

- **VGGFACE2:** New large scale face dataset with 9131 identities for face recognition across pose, age, gender and color (Cao et al., 2018).
- **WIDER FACE:** A face detection benchmark dataset, we choose 32,203 images and label 393,703 faces with a high degree of variability in scale, pose and occlusion, etc (Yang et al., 2016).

- Pose Detection
- Segmentation
- Image Captioning
- Action Recognition
- Medical Processing

Part1

Typical Image_Classification

1.3 Fisher Vector

In classical and statistical classification, two main approaches are called the **generative** approach and the **discriminative** approach (Jebara et al., 2004):

- Given an observable variable X and a target variable Y, a **generative model** is a statistical model of the joint probability distribution on $P(X|Y=y)$. **Generative adversarial networks** are examples of this, and are judged primarily by the similarity of particular outputs to potential inputs.

- A **discriminative model** is a model of the conditional probability of the target Y, given an observation x, symbolically $P(Y|X=x)$. Like **Neural Networks**.

Fisher Kernel

1.3

the **Fisher kernel,** is a function that measures the similarity of two objects on the basis of sets of measurements for each object and a statistical model.

→ It combines the advantages of generative statistical models and those of discriminative methods (like support vector machines) (Tommy et al., 1998):

- The Fisher kernel makes use of the **Fisher score**, defined as:

$$U_X = \nabla_\theta \log P(X|\theta)$$

- The **Fisher Kernel** defines as:

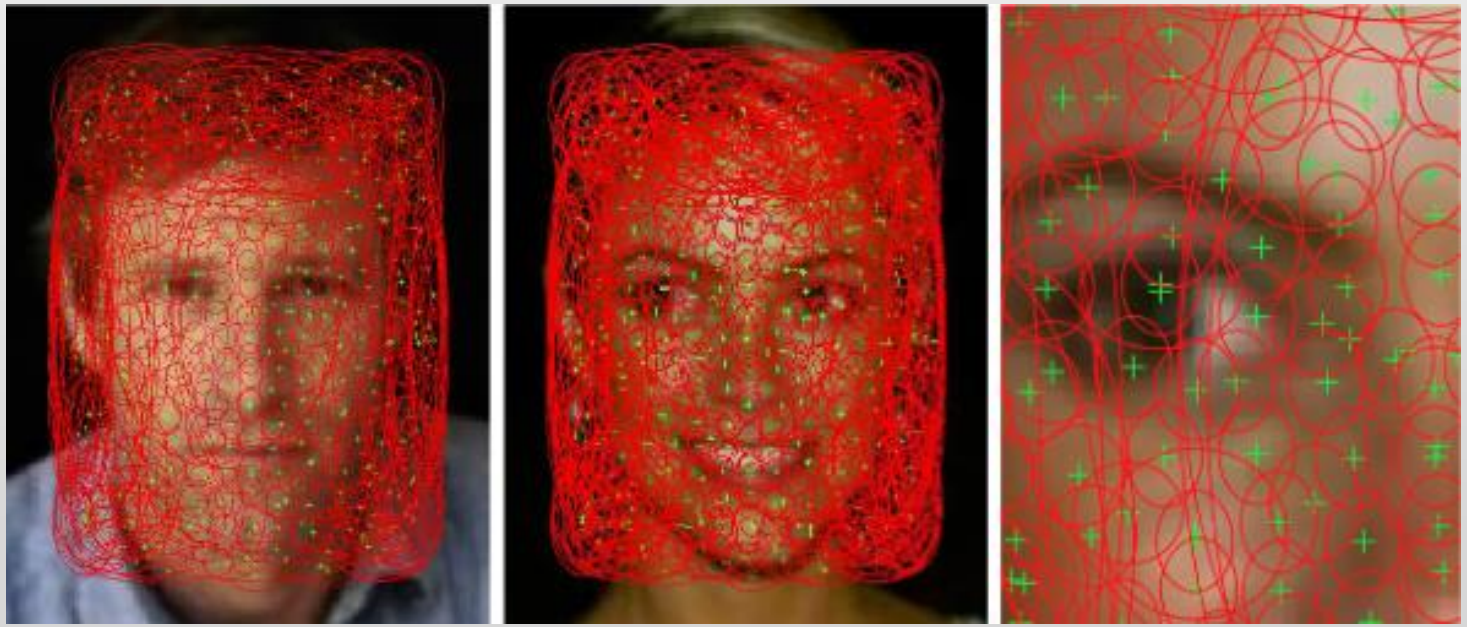
$$K(X_i, X_j) = U_{X_i}^T \mathcal{I}^{-1} U_{X_j}$$

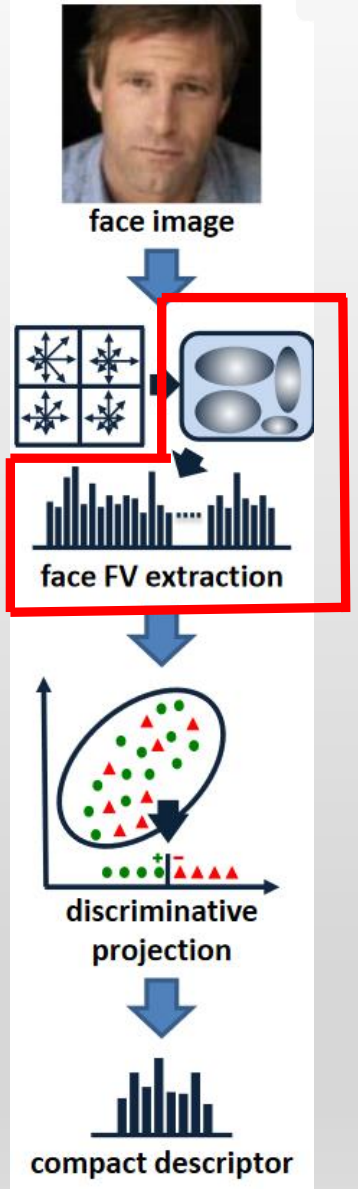
Application

1.3

Face Verification Pipeline with FV

(Perronninet et al., 2010)





Advantages:

- ☑ Efficient to compute.
- ☑ leads to excellent results even with high dimensions.
- ☑ It performs well even with simple linear classifiers.

Disadvantages:

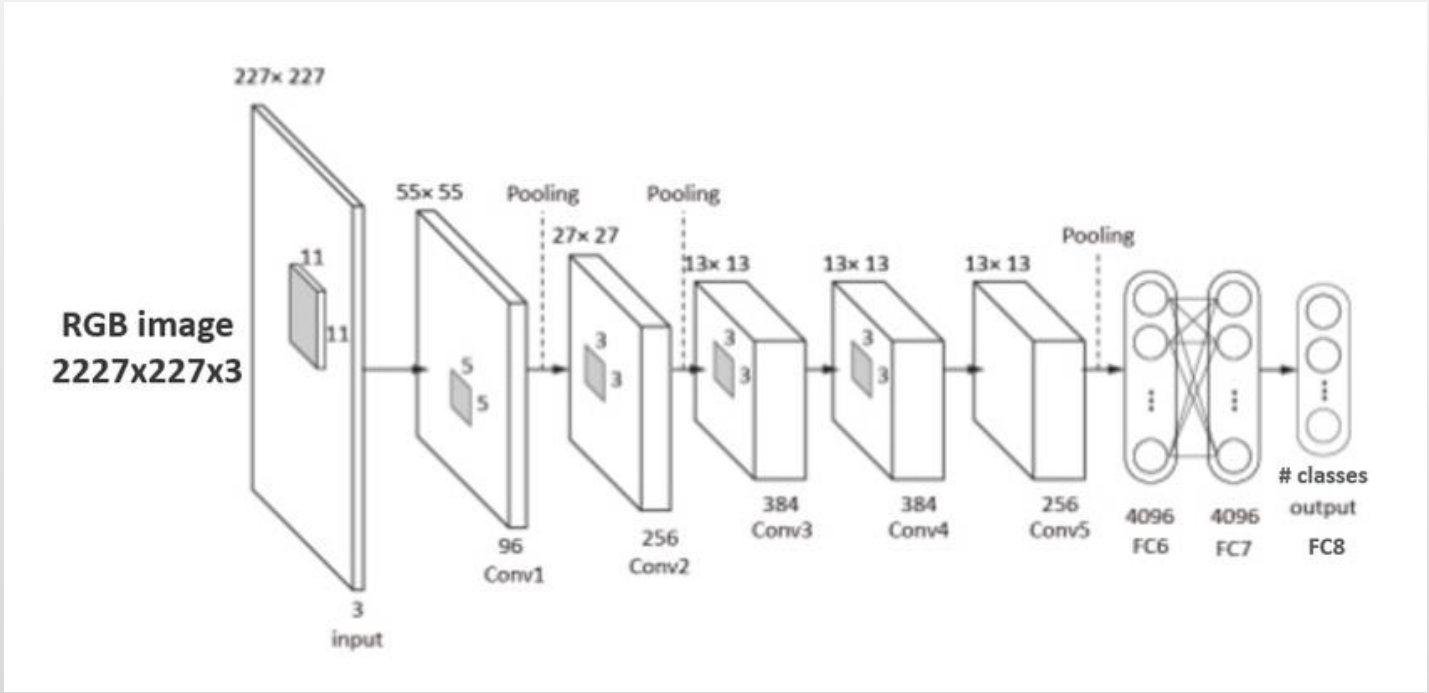
- ☒ The FV is almost dense. This leads to storage as well as input/output issues which make it impractical for large-scale applications as is.

Part1

Typical Image_Classification

1.4 AlexNet

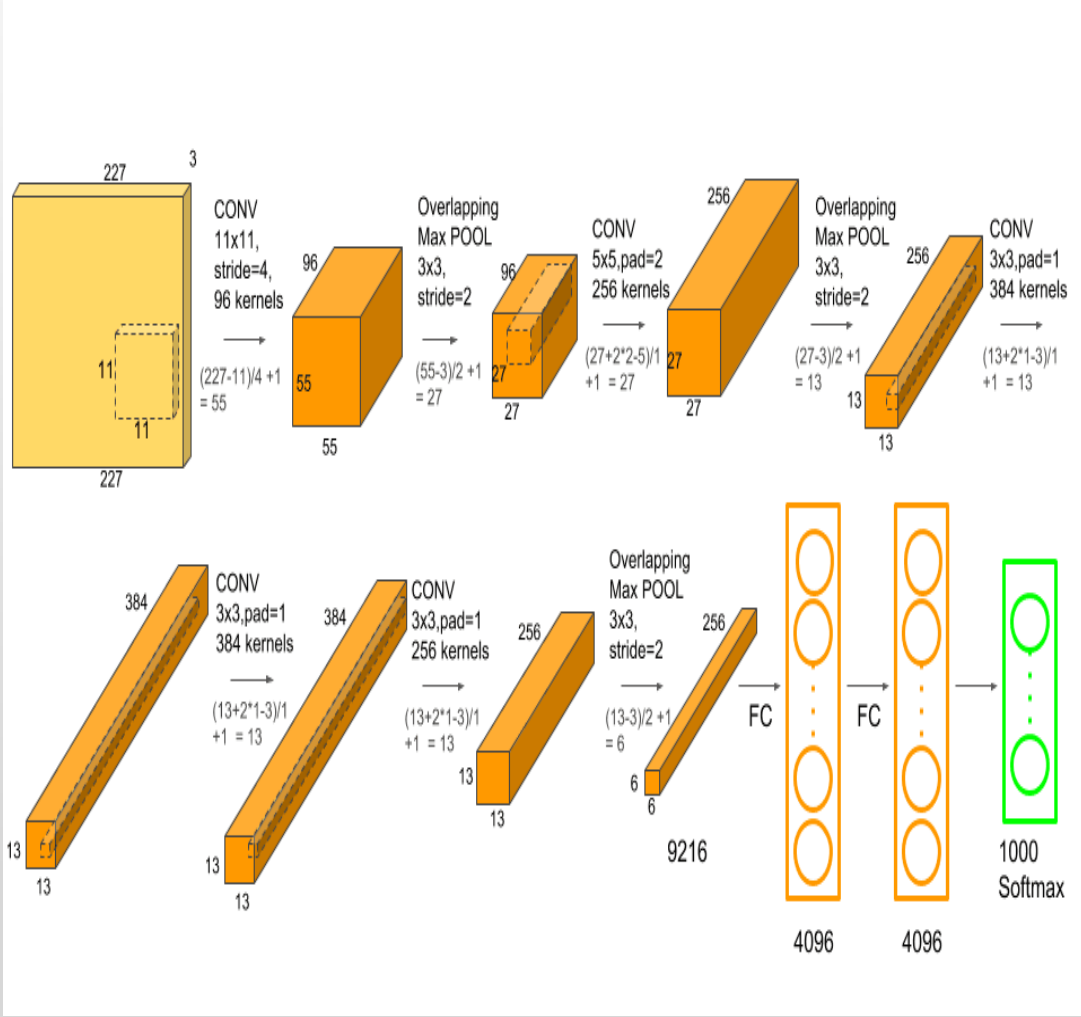
The network achieved a top-5 error of 15.3% on ILSVRC, more than 10.8 percentage points lower than that of the runner up (which was SIFT+FVs) (Krizhevsky et al., 2012).



Description and Features

1.4

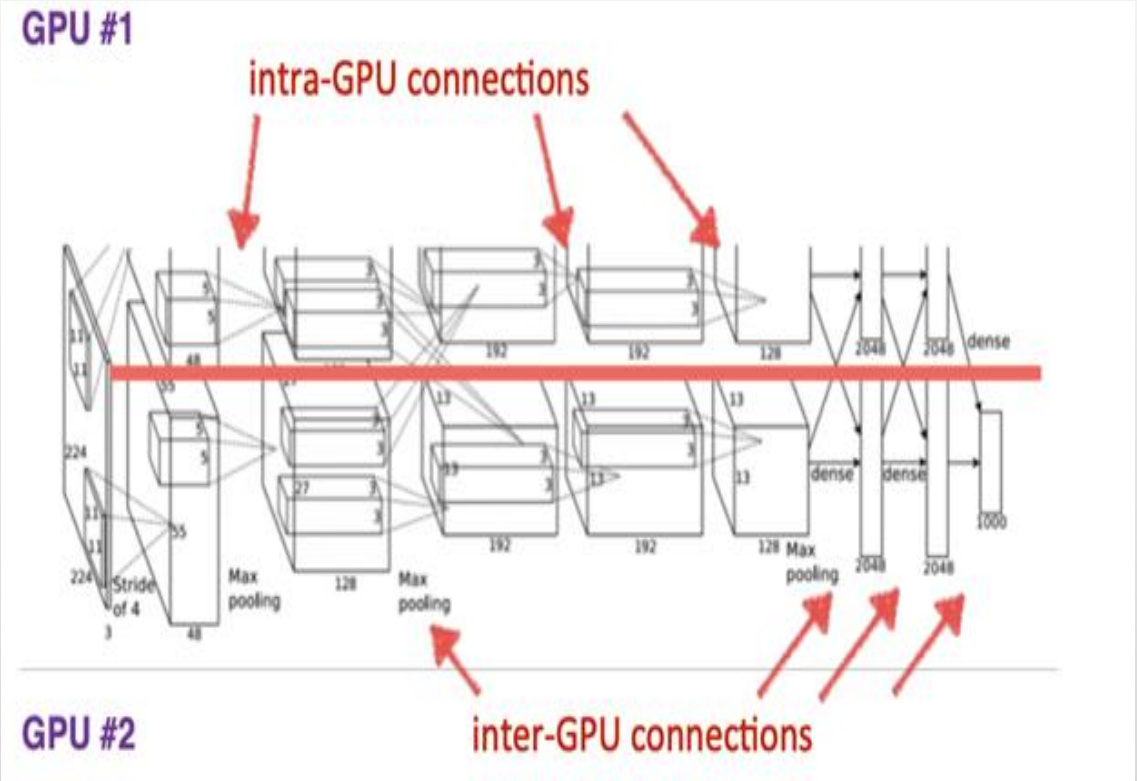
- Depth of the model was essential for its high performance.
- Computationally expensive, but made feasible due to the utilization of GPUs during training (for the first time).
- AlexNet contained eight layers; the first five were convolutional layers, some of them followed by max-pooling layers, and the last three were fully connected layers. It used the ReLU activation function, which showed improved training performance over tanh and sigmoid.



Description and Features

1.4

- Has been cited over 80,000 times up to 2021.
- AlexNet overall has 60 million parameters.
- Still mostly uses in cognitive applications cause of simplest and most similarity to human brain (Schrimpf et al., 2018).
- Due to ReLu, the learned variables can become unnecessarily high, To prevent this, AlexNet introduced Local Response Normalization (LRN).



Local Response Normalization

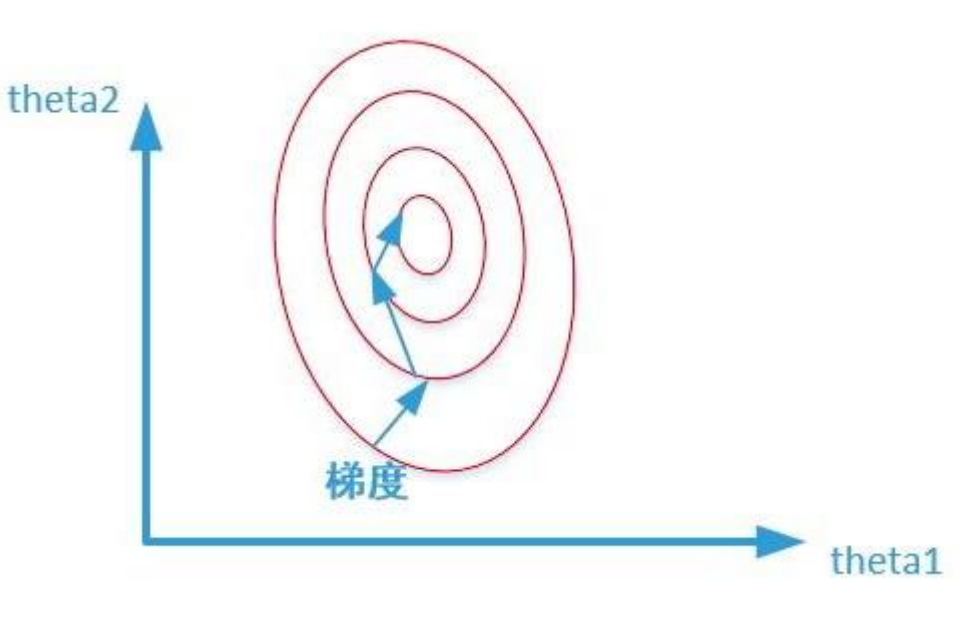
1.4

- When havo no infinity while using unlimited activation functions like ReLU, ELU.
- It is a concept in Neurobiology that refers to the capacity of a neuron to reduce the activity of its neighbors.
- carry out local contrast enhancement so that locally maximum pixel values are used as excitation for the next layers.
- Non-trainable, these layers have recently fallen out of favor because in practice their contribution has been shown to be minimal, if any.
- LRN increases memory consumption and training time with no particular increase in accuracy.

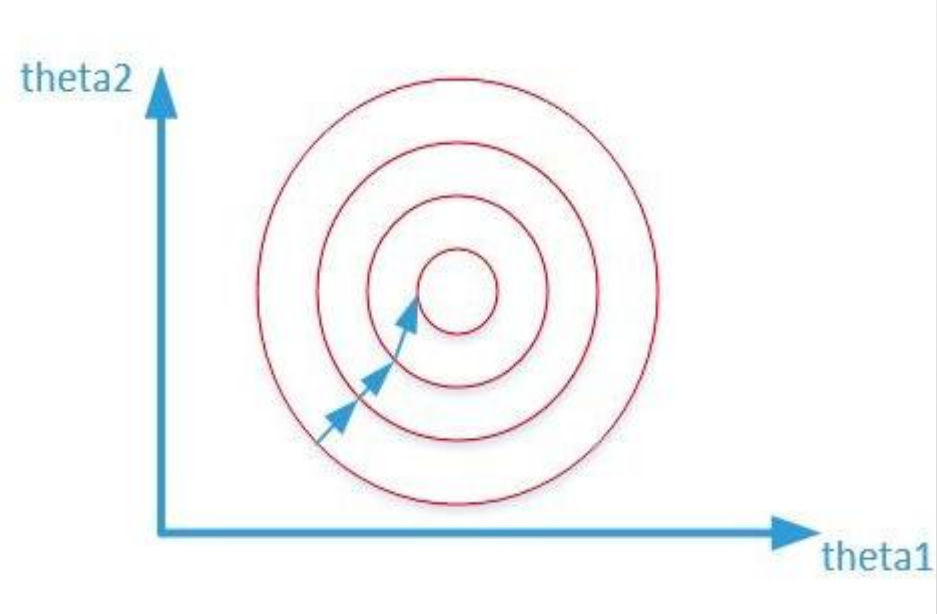
Local Response Normalization

1.4

Before Normalization



After Normalization



Advantages:

- ☑ The ReLu activation function used in this network does not limit the output unlike other activation functions. This means there isn't too much loss of features and there is no VG at training instead of tanh.
- ☑ Rarely overfits with normal LR.

Disadvantages:

- ☒ The depth of this model is very less and hence it struggles to learn features from image sets.
- ☒ It takes more time to achieve higher accuracy results compared to future models.
- ☒ it is often very computationally expensive due to FCs.

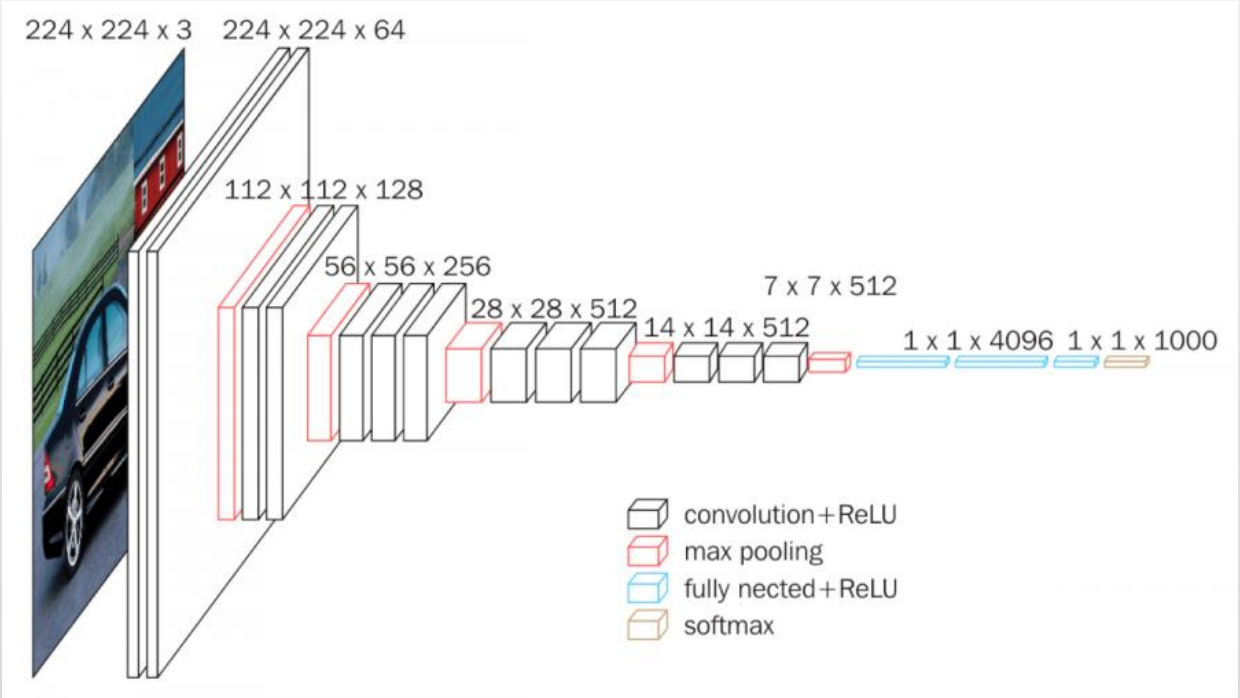
Part1

Typical Image_Classification

1.5 VGGNet

- VGGNet was born out of the need to reduce the # of parameters in the CONV layers and improve on training time.

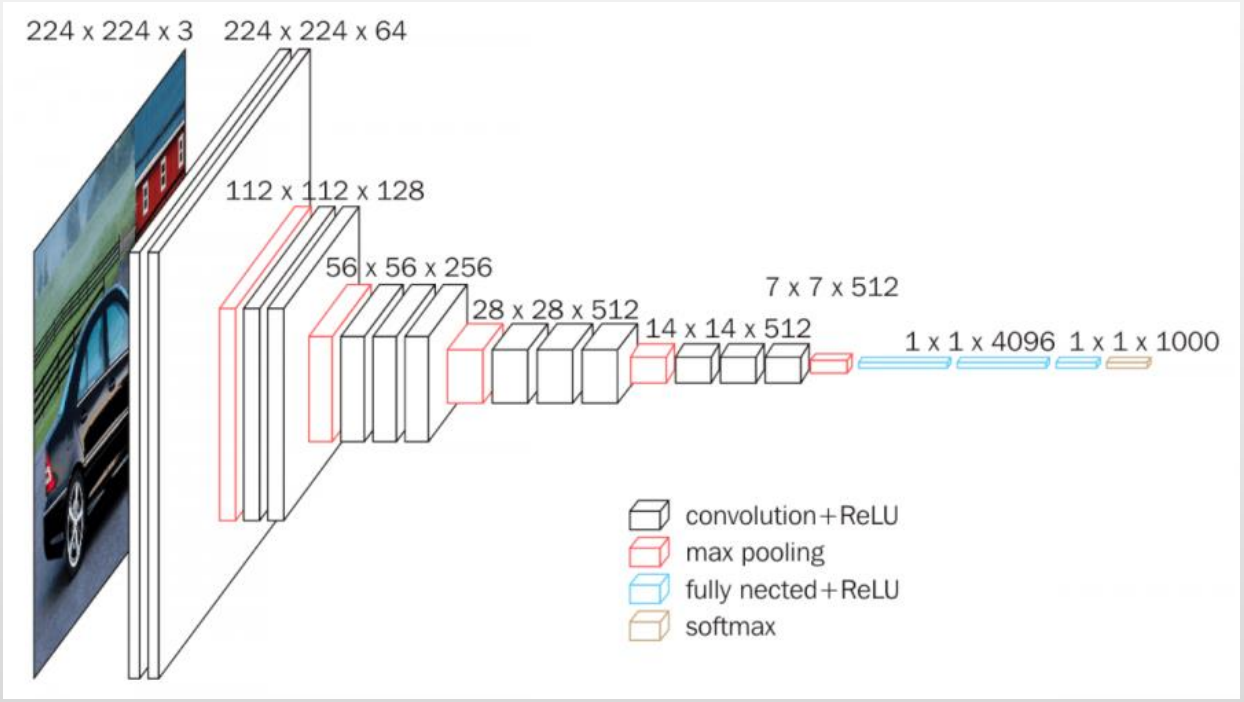
- There are multiple variants of VGGNet which differ only in the total number of layers in the network (Simonyan et al., 2015).



Architecture Explanation

- **Input:** VGG takes in a 224x224 pixel RGB image. For the ImageNet competition, the authors randomly cropped out the 224*224 patch.

- **Convolutional Layers:** The convolutional layers in VGG use a very small receptive field (3x3, the smallest possible size). There are also 1x1 convolution filters which act as a linear transformation of the input, which is followed by a ReLU unit. The convolution stride is fixed to 1 pixel so that the spatial resolution is preserved after convolution.

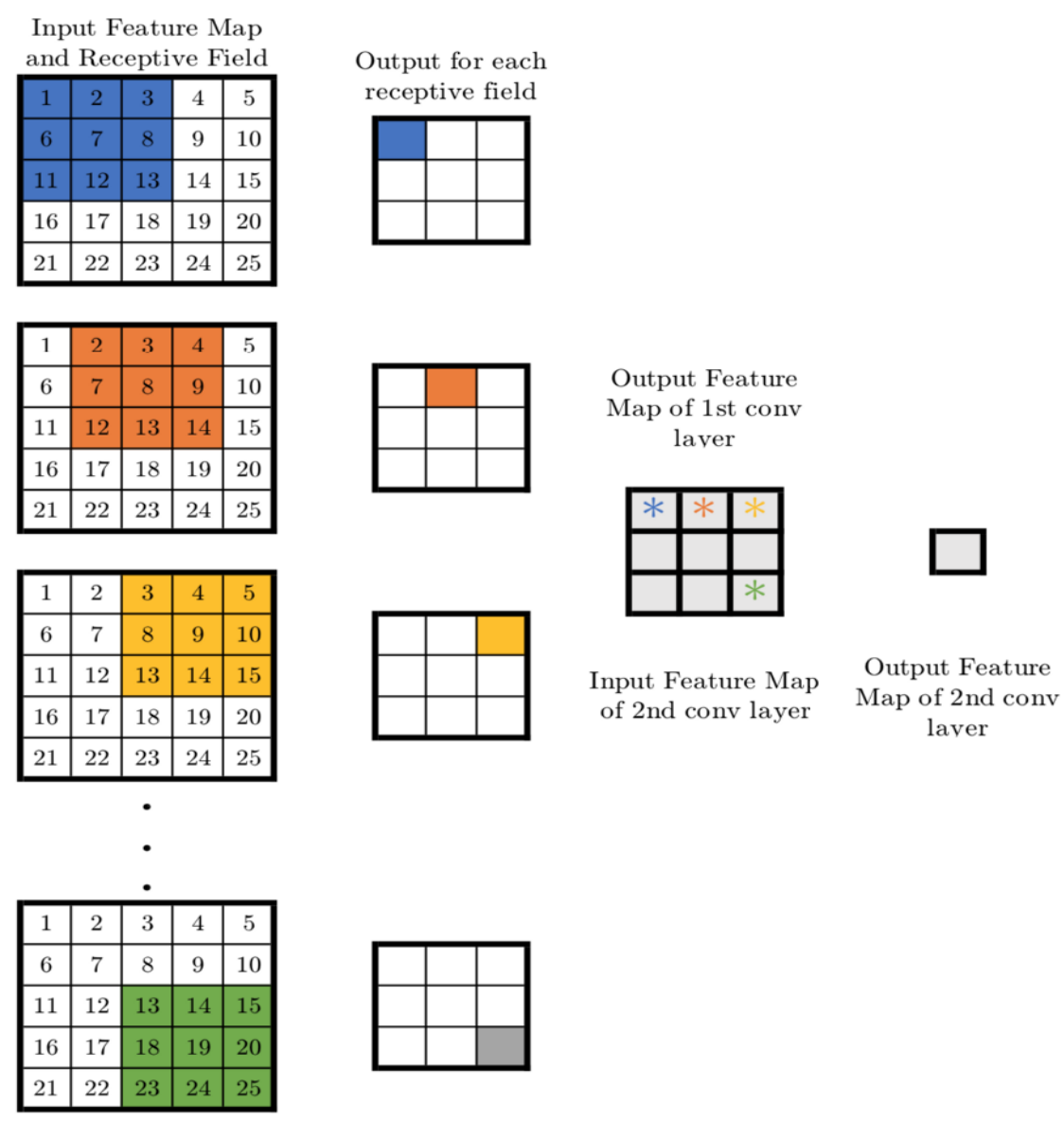


From AlexNet to VGG

- Instead of using large receptive fields like AlexNet (11x11 with a stride of 4), VGG uses very small receptive fields (3x3 with a stride of 1). There are also fewer parameters.

- VGG incorporates 1x1 convolutional layers to make the decision function more non-linear without changing the receptive fields.

- The small-size convolution filters allows VGG to have a large number of weight layers; of course, more layers leads to improved performance.



- For a 5x5 conv layer filter, the number of variables is 25. On the other hand, two conv layers of kernel size 3x3 have a total of 3x3x2=18 variables (a reduction of 28%).
- Similarly, the effect of one 7x7 (11x11) conv layer can be achieved by implementing three (five) 3x3 conv layers with a stride of one. This reduces the number of trainable variables by 44.9% (62.8%).

1.5

Advantages:

- ☑ The increase in the number of layers with smaller kernels saw an increase in non-linearity which is always a positive in deep learning.

Disadvantages:

- ☒ Constantly learning and relearning is a problem with VGG which is why the loss seems to be so unpredictable (explosion of gradients).
- ☒ VGG is slower than the newer ResNet architecture that introduced the concept of residual learning which was another major breakthrough.

Part1

Typical Image_Classification

1.6 Inception Models

- The Inception network was an important milestone in the development of CNN classifiers. Most popular CNNs just stacked convolution layers deeper and deeper, hoping to get better performance.

1.6
Inception Models

The Inception network on the other hand, was complex (heavily engineered). It used a lot of tricks to push performance; both in terms of speed and accuracy. Its constant evolution lead to the creation of several versions of the network.
V1, V2, V3, V4 and Inception ResNet

Problem?

- Because of this huge variation in the location of the information, choosing the **right kernel size** is tough.

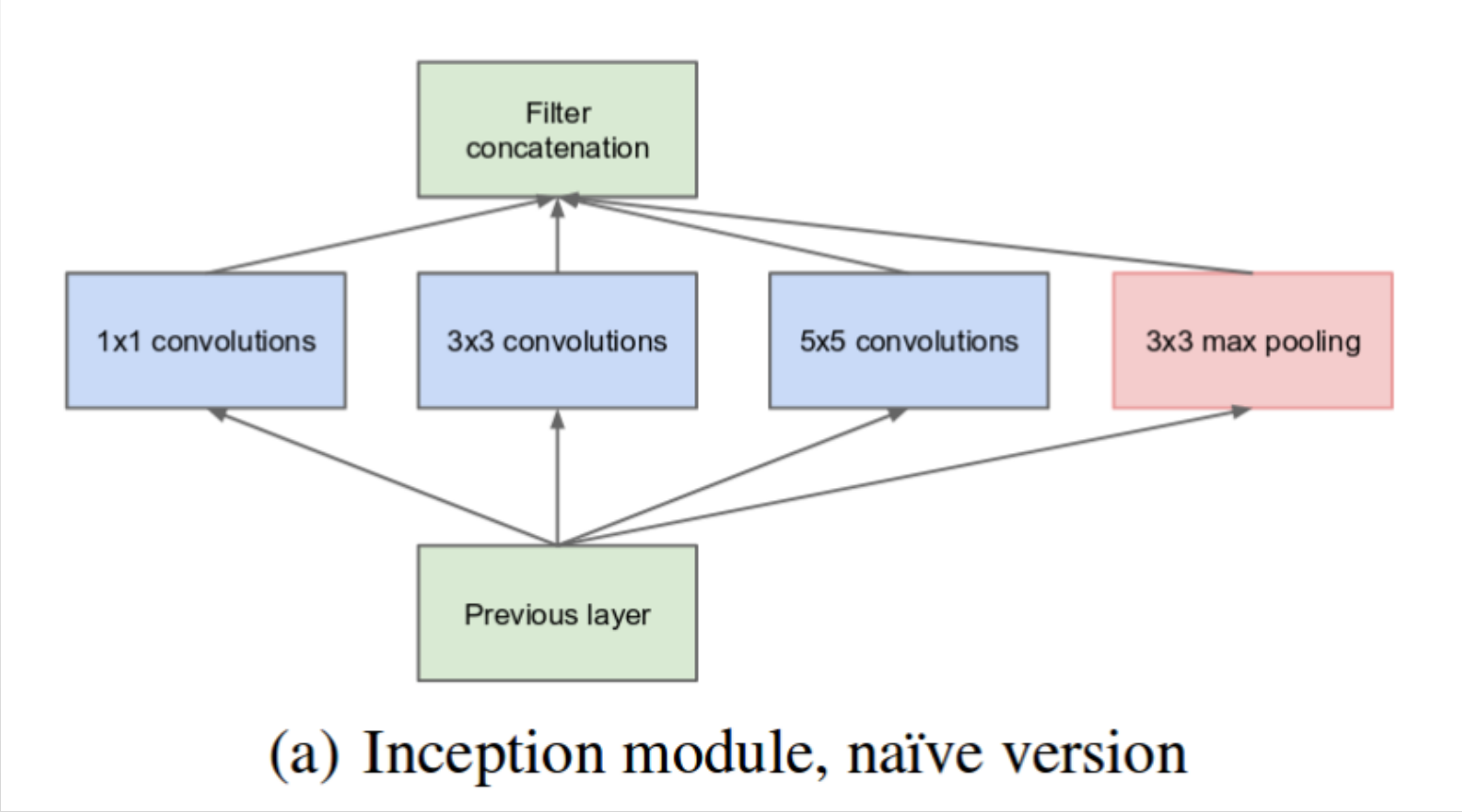
- **Very deep networks** are prone to **overfitting**. It also hard to pass gradient updates through the entire network

- Naively stacking large convolution operations is **computationally expensive**.

1.6

Solution

- Why not have filters with **multiple sizes** operate on the **same level**? The network essentially would get a bit "**wider**" rather than "deeper" and **concatenated** them finally.

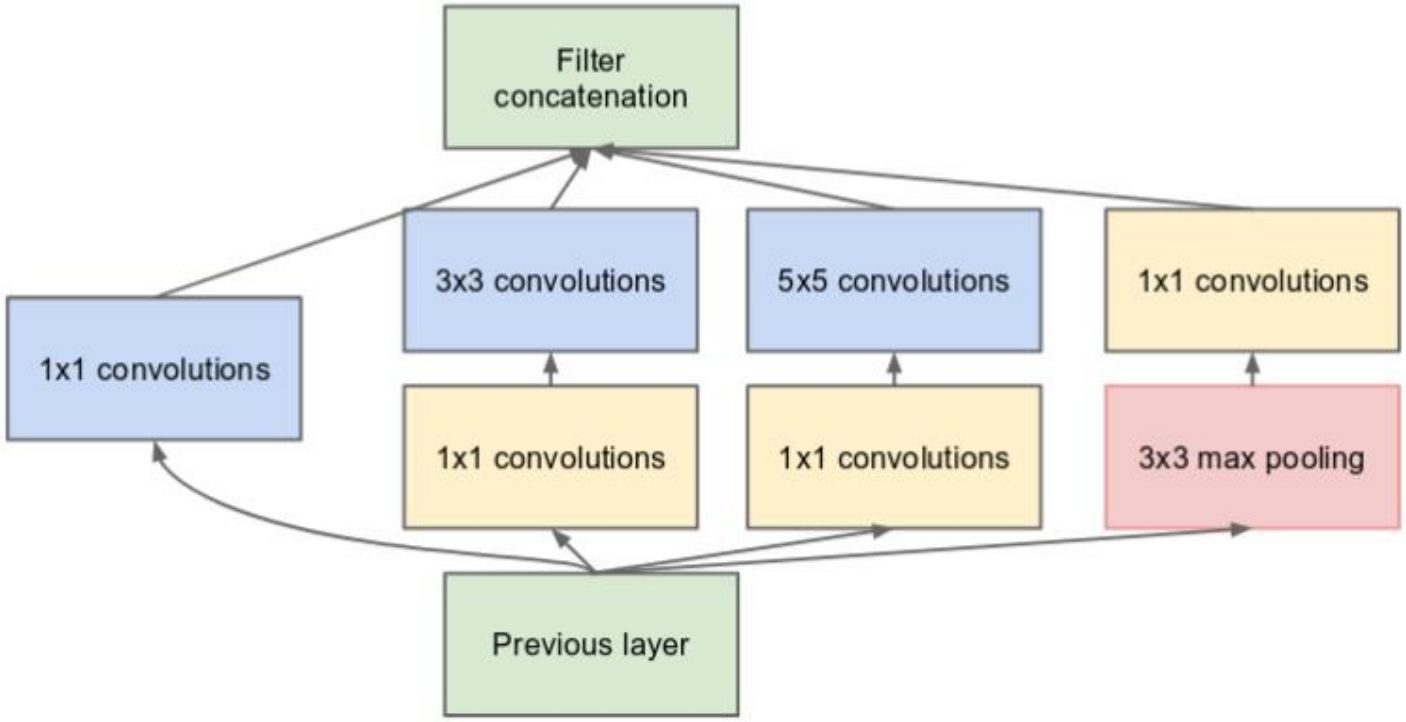


(a) Inception module, naïve version

1.6

Solution

- To make it cheaper, the authors **limit** the number of **input channels** by adding an **extra 1x1 convolution** before the 3x3 and 5x5 convolutions.

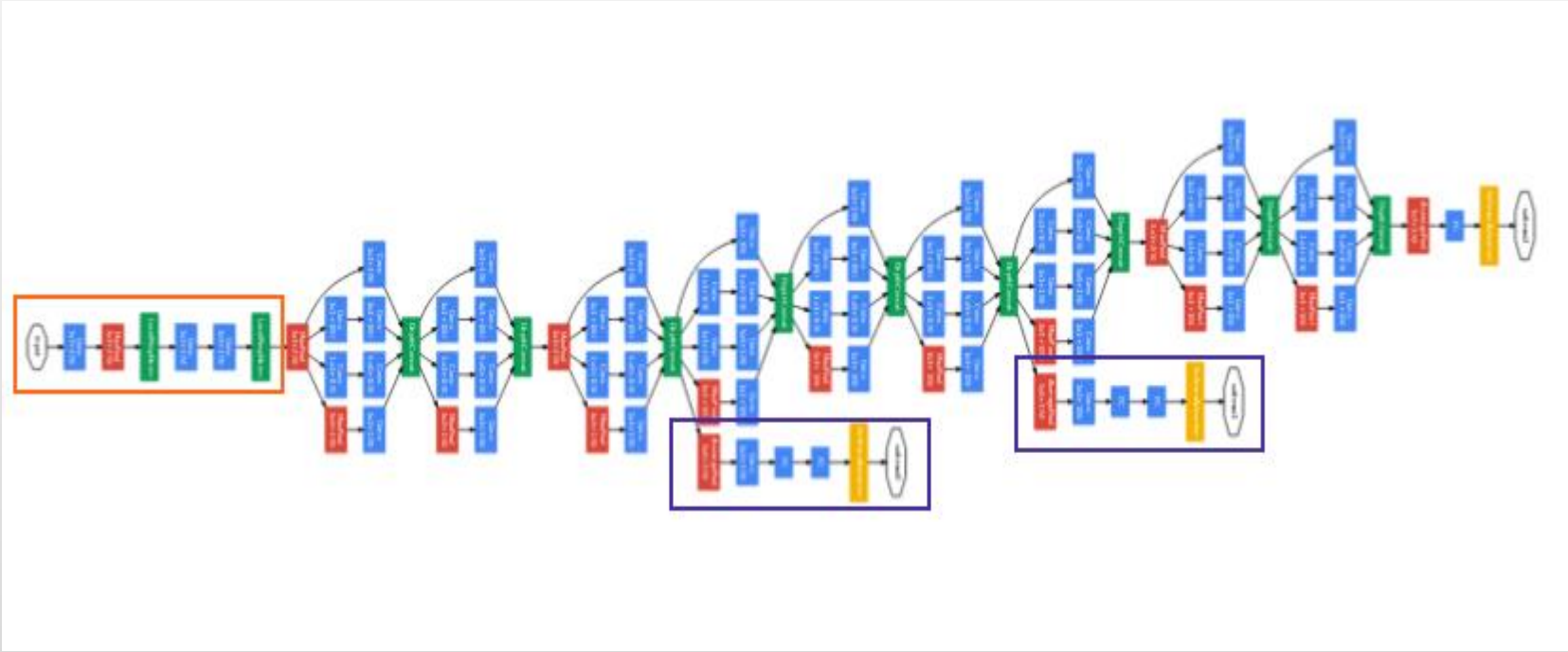


(b) Inception module with dimension reductions

1.6

Architecture

- GoogLeNet has 9 such inception modules stacked linearly. It is 22 layers deep (27, including the pooling layers). It uses global average pooling at the end of the last inception module (Christian et al., 2015).

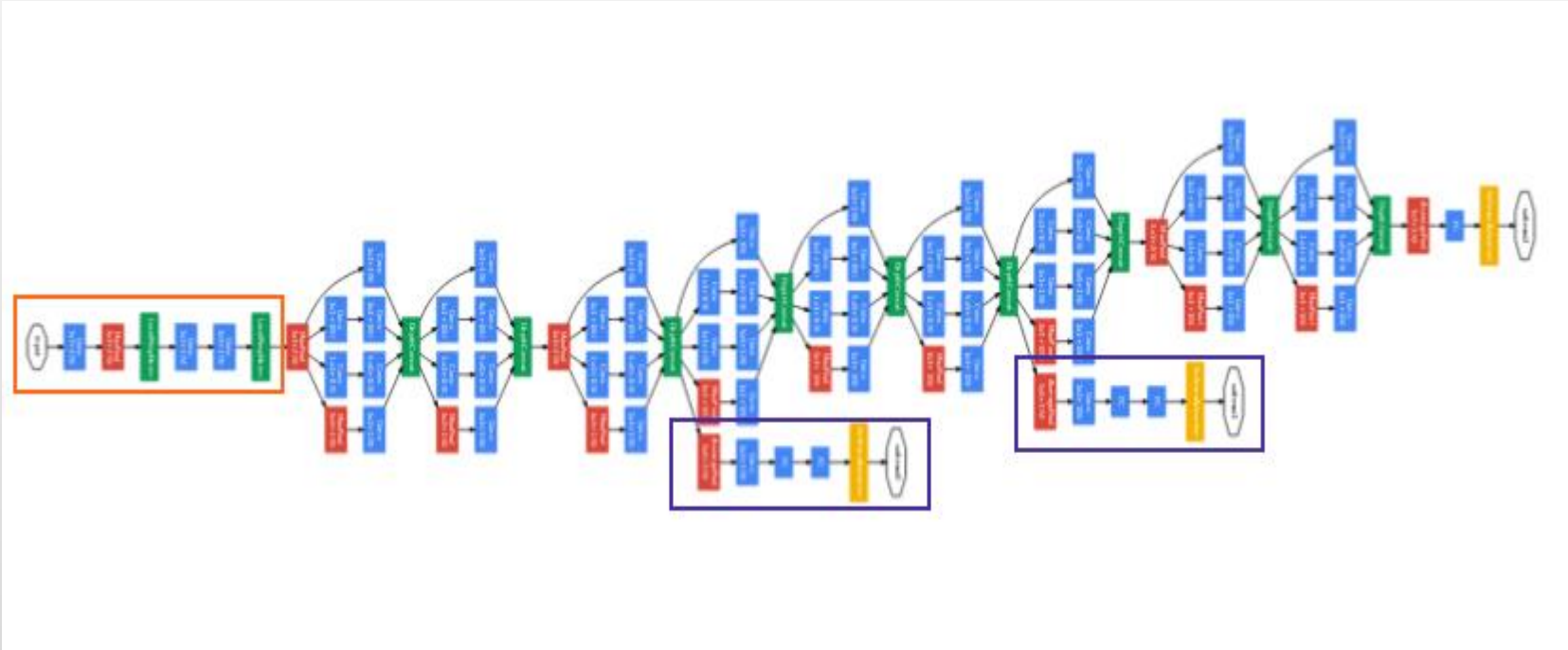


Needless to say, it is a pretty **deep classifier**. As with any very deep network, it is subject to the **vanishing gradient problem**.

1.6

Architecture

- To prevent the **middle part** of the network from "**dying out**", the authors introduced **two auxiliary classifiers** (The purple boxes in the image). They essentially applied softmax to the outputs of two of the inception modules, and computed an **auxiliary loss** over the same labels.



- The **total loss function** is a **weighted sum** of the **auxiliary loss** and the **real loss**. Weight value used in the paper was 0.3 for each auxiliary loss. (just in training)

Inception V2

Premise:

1- Reducing the dimensions too much may cause loss of information, known as a “representational bottleneck”

2- Using smart factorization methods, convolutions can be made more efficient in terms of computational complexity.

1.6

Inception V2

Solutions:

1- **Factorize 5x5** convolution **to two 3x3** convolution operations to improve computational speed.

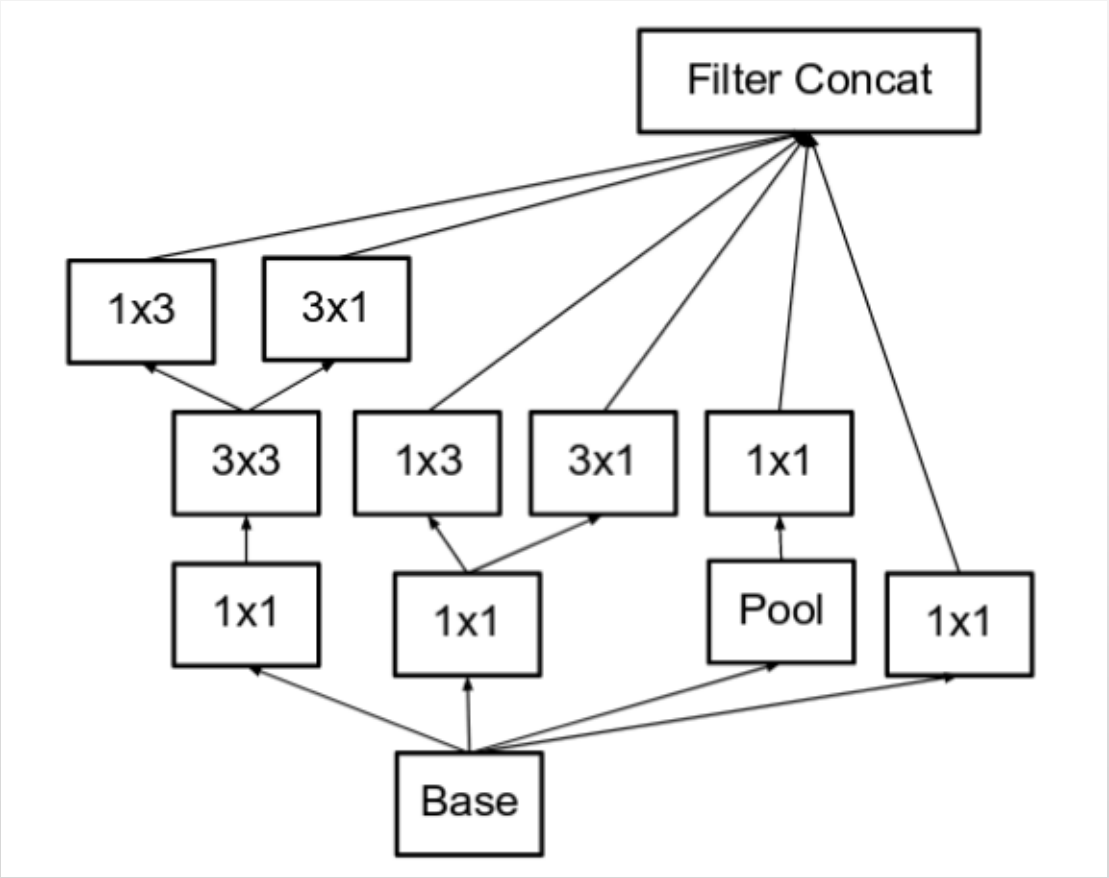
2- Moreover, they **factorize** convolutions of filter size **nxn** to a **combination** of **1xn and nx1** convolutions.

1.6

Inception V2

Solutions:

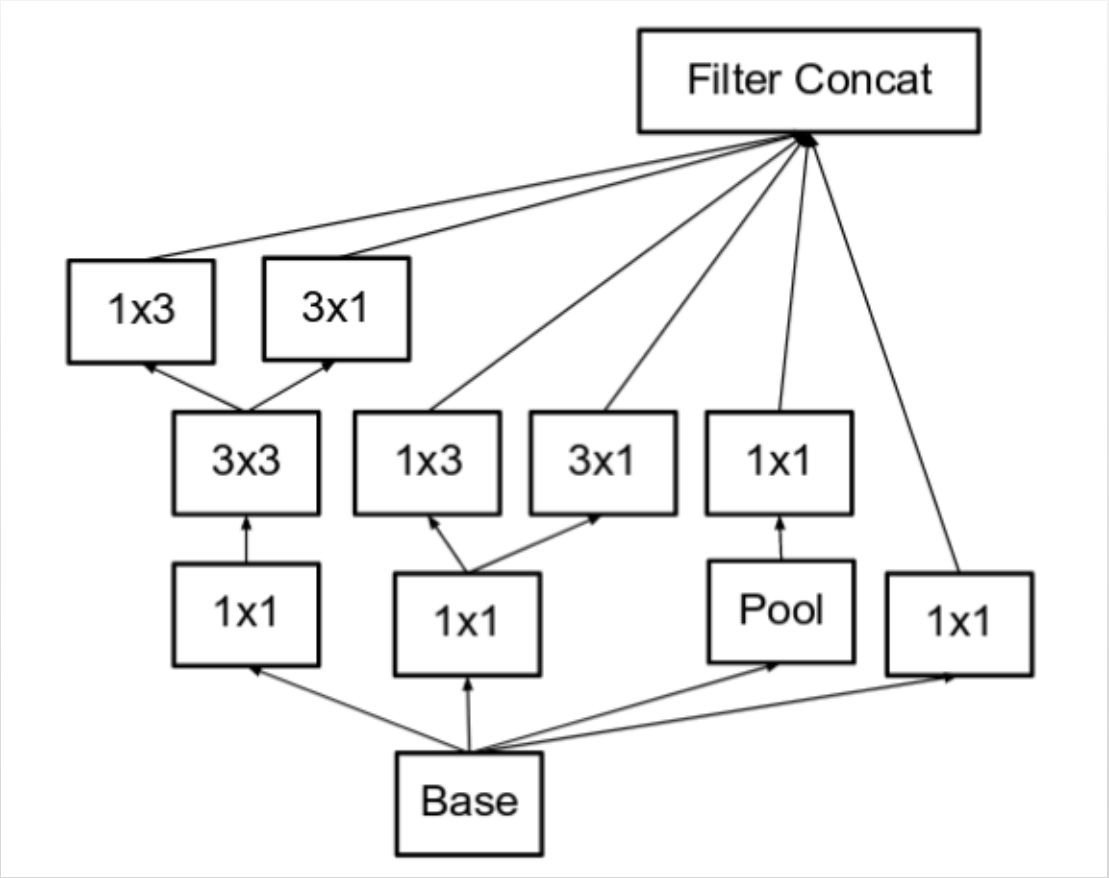
3- The **filter banks** in the module were **expanded** (made wider instead of deeper) to remove the representational bottleneck. If the module was made deeper instead, there would be excessive reduction in dimensions, and hence loss of information.



Inception V2

Solutions:

3- The **filter banks** in the module were **expanded** (made wider instead of deeper) to remove the representational bottleneck. If the module was made deeper instead, there would be excessive reduction in dimensions, and hence loss of information.



Inception V3

Premise:

1- The authors noted that the **auxiliary classifiers** didn’t contribute much until near the end of the training process, when accuracies were nearing saturation.

1.6

Inception V3

Solutions:

1- BatchNorm in the Auxillary Classifiers.

2- Label Smoothing (Christian et al., 2016).

$$\nabla \text{CE} = p - y = \text{softmax}(z) - y$$

1.6

Inception V4

Premise:

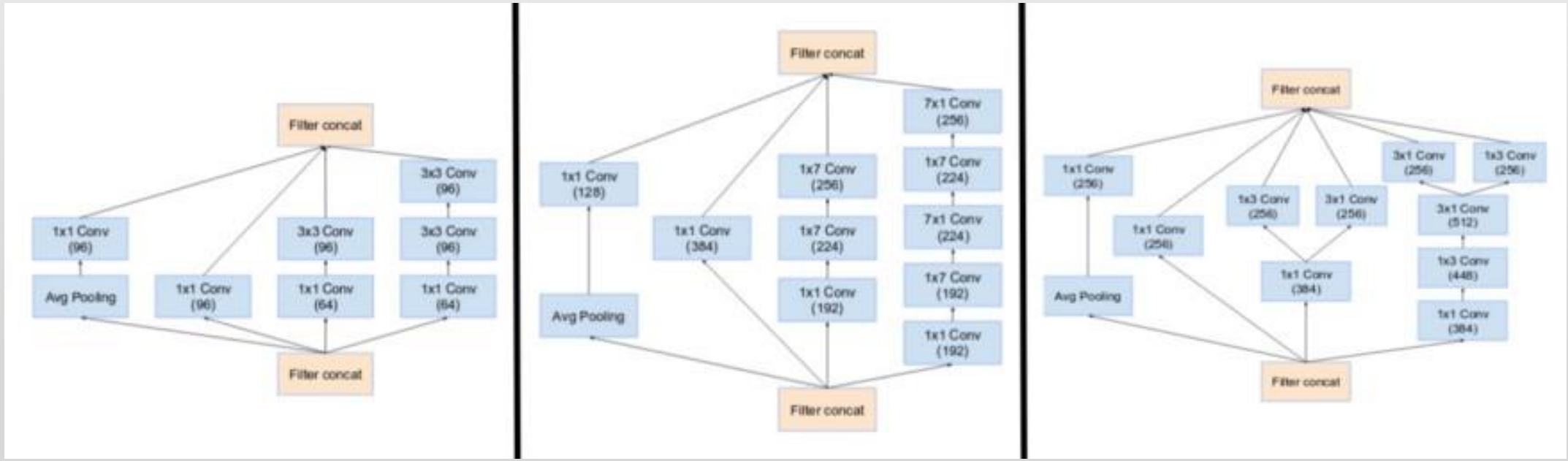
1- Make the modules more **uniform**. The authors also noticed that some of the modules were **more complicated than necessary**.

1.6

Inception V4

Solutions:

1- Introduced specialized "**Reduction Blocks**"



Advantages:

- ☑ Ability to extract features from input data at varying scales through the utilisation of varying convolutional filter sizes.
- ☑ 1x1 conv filters learn cross channels patterns, which contributes to the overall feature extractions capabilities of the network.

Disadvantages:

- ☒ Highly explosion and vanishing of gradients.
- ☒ Hard to Implement.

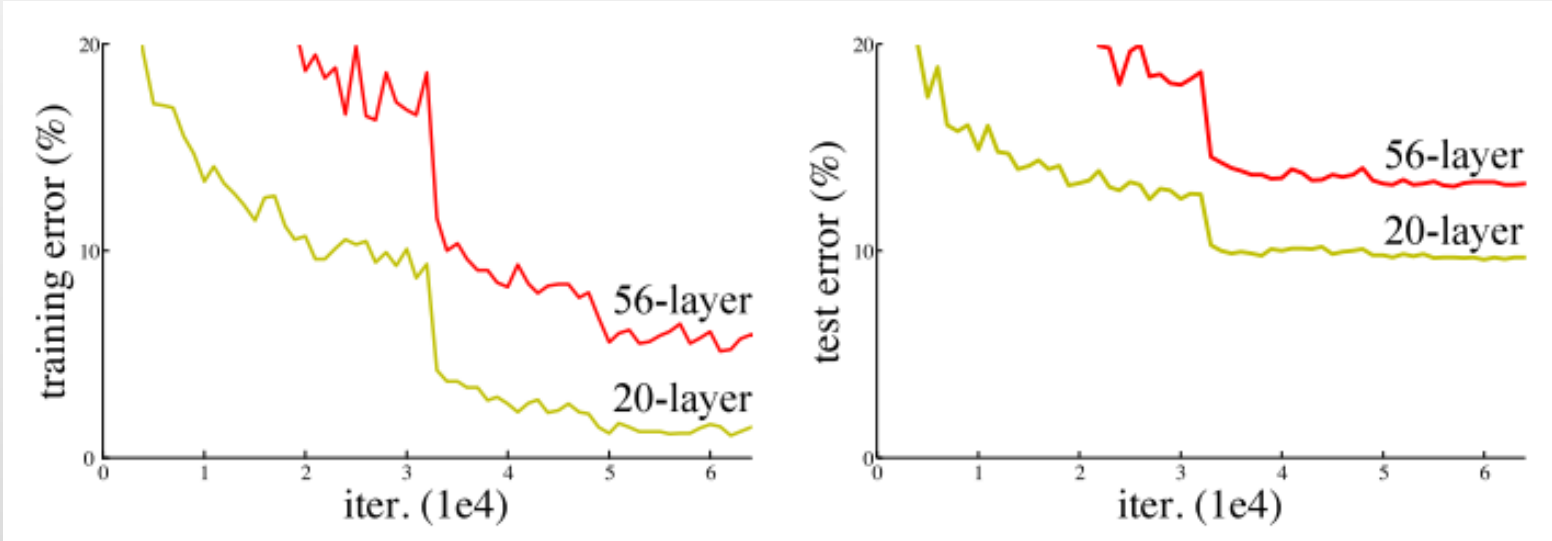
1.6

Part1

Typical Image_Classification

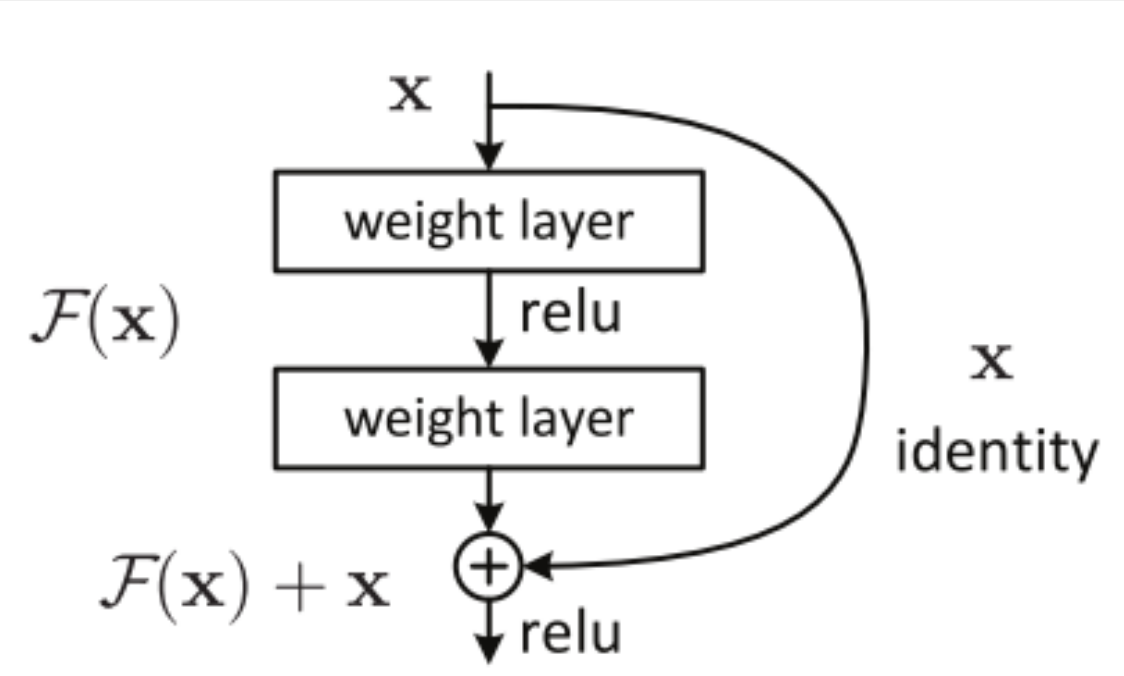
1.7 ResNet

- However, increasing network depth does not work by simply stacking layers together. Deep networks are hard to train because of the notorious vanishing gradient problem.



ResNet – The Idea

- By bypassing the input to the output& prevents from gradient vanishing, cause x itself incorporates in backpropagation procedure.

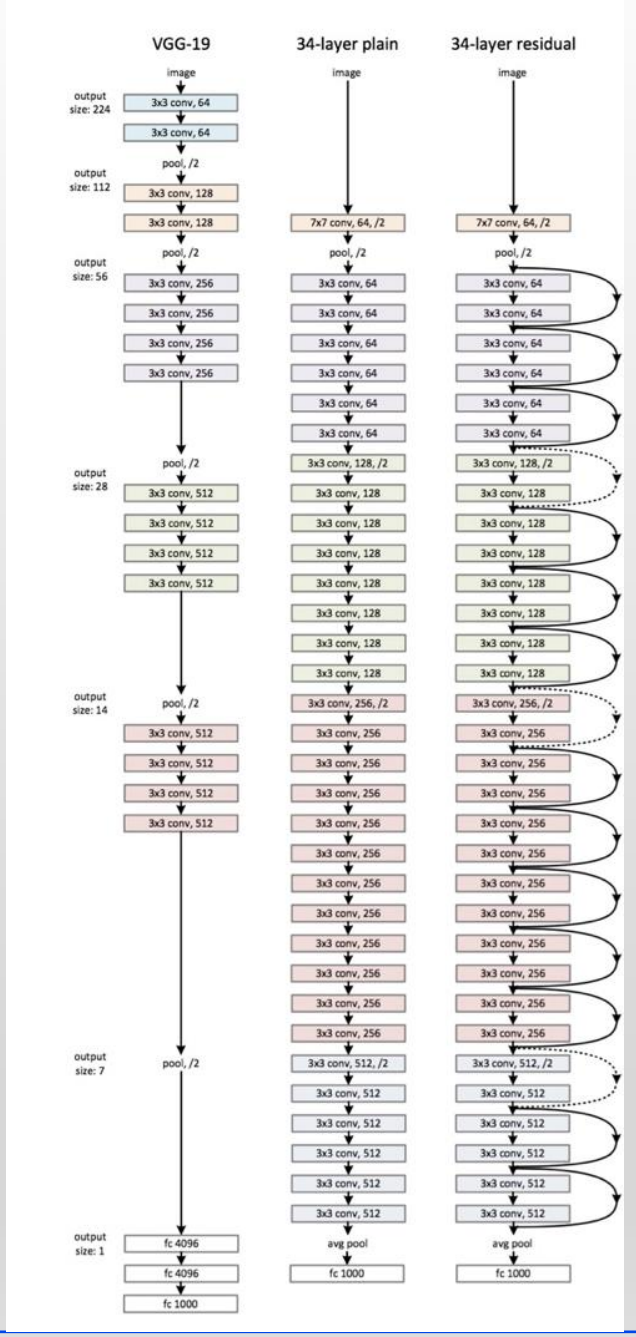


1.7

ResNet – The Idea

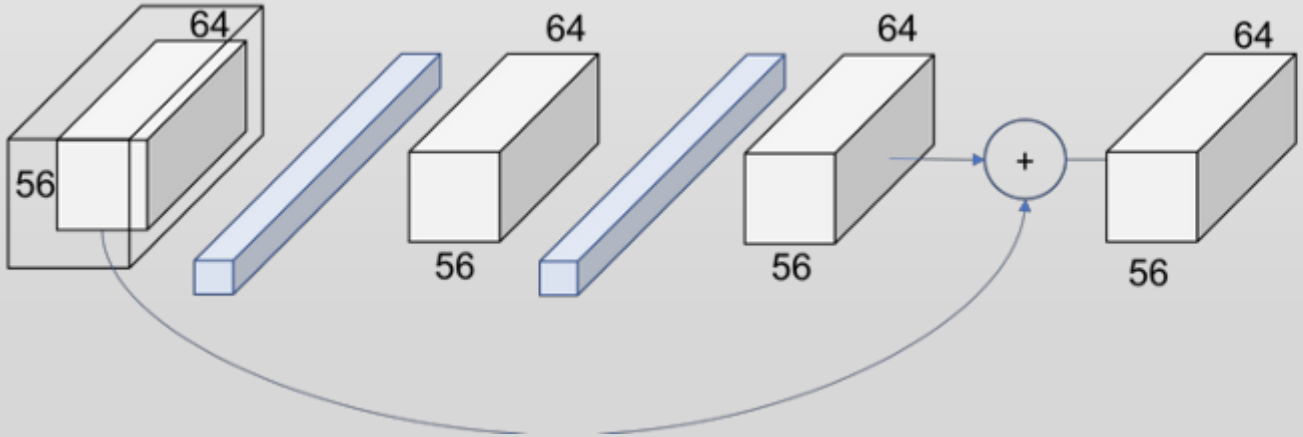
- The dotted line is there, precisely because there has been a change in the dimension of the input volume. Applies self conved for dimension checking.

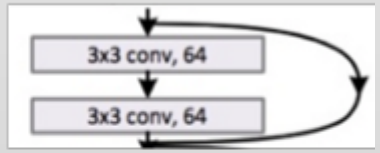
- The dimension doesn't reduce because a padding = 1 is used and a stride of also 1.



ResNet – The Idea

- The dimension doesn't reduce because a padding = 1 is used and a stride of also 1 (Kaiming et al., 2016).





1.7

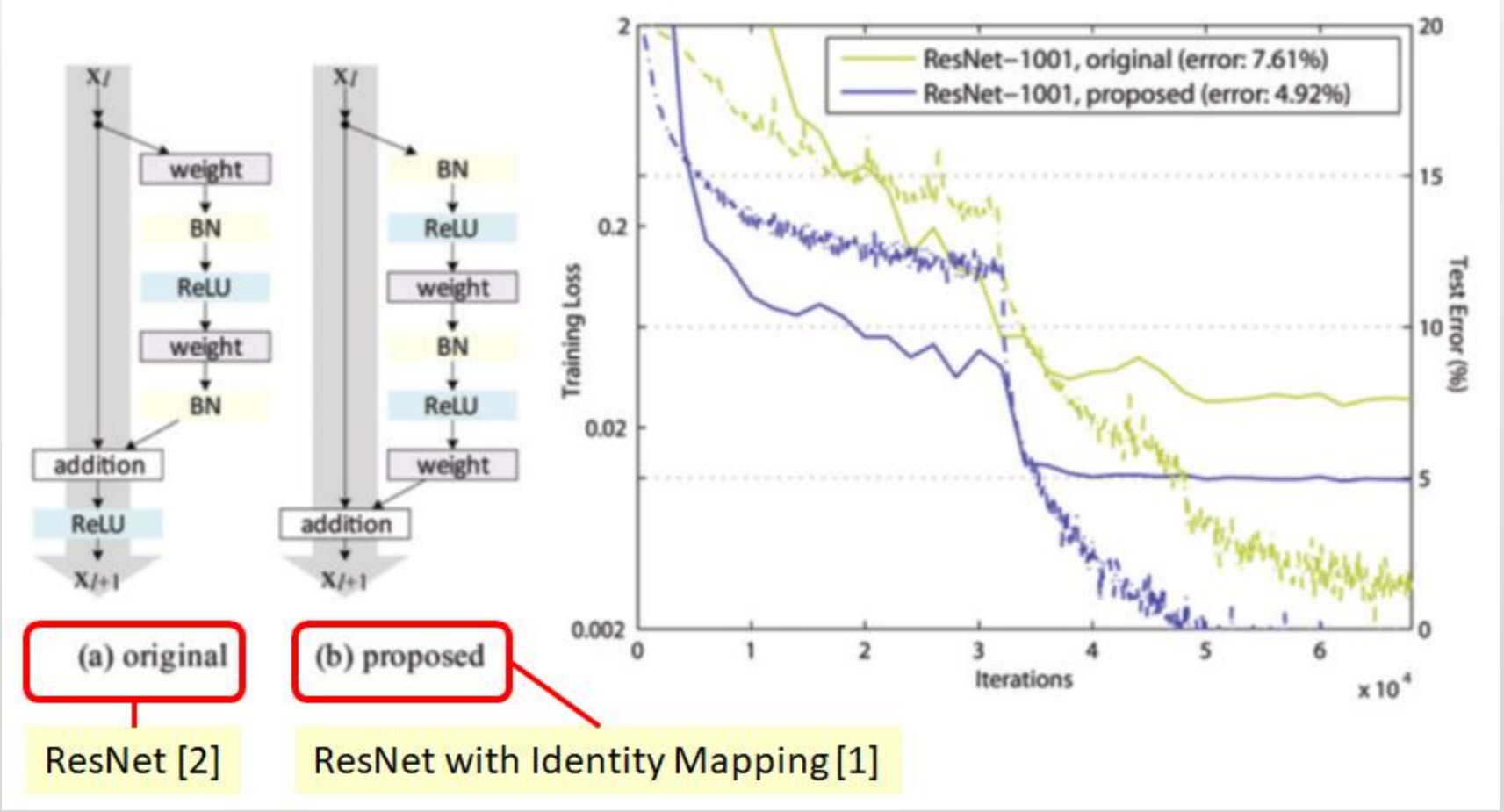
ResNet – V1 vs V2 (Kaiming et al., 2016)

- V1: Convolution then batch normalization then ReLU

- V2: Batch normalization then ReLu then convolution

1.7

ResNet – V1 vs V2



1.7

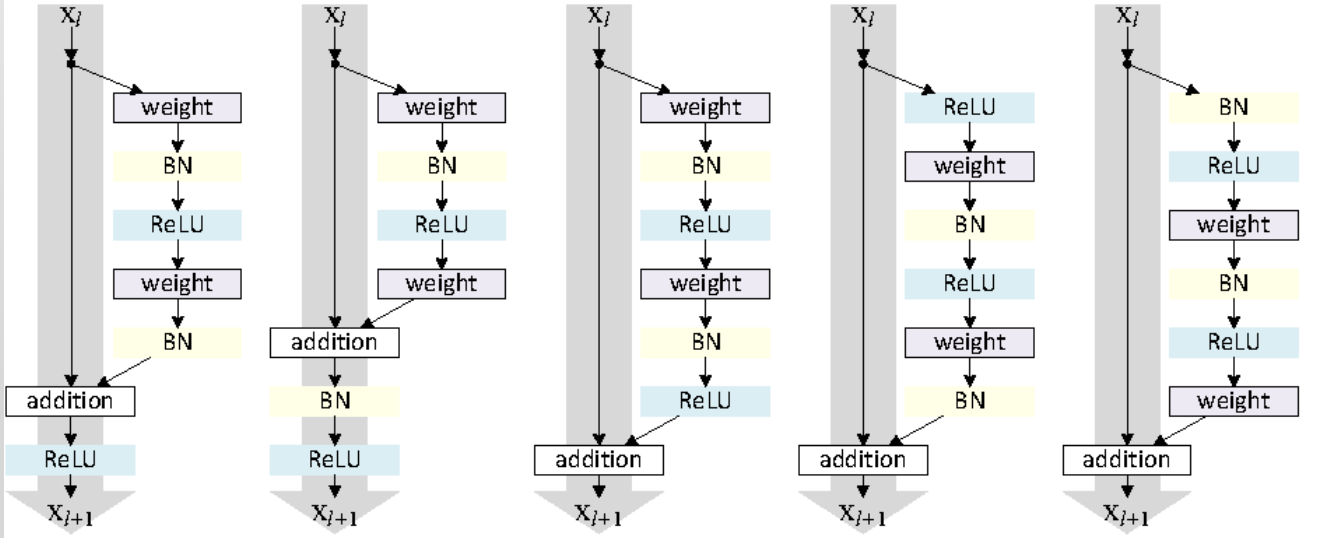
Official Versions of ResNet

Number of Layers	Number of Parameters
ResNet 18	11.174M
ResNet 34	21.282M
ResNet 50	23.521M
ResNet 101	42.513M
ResNet 152	58.157M

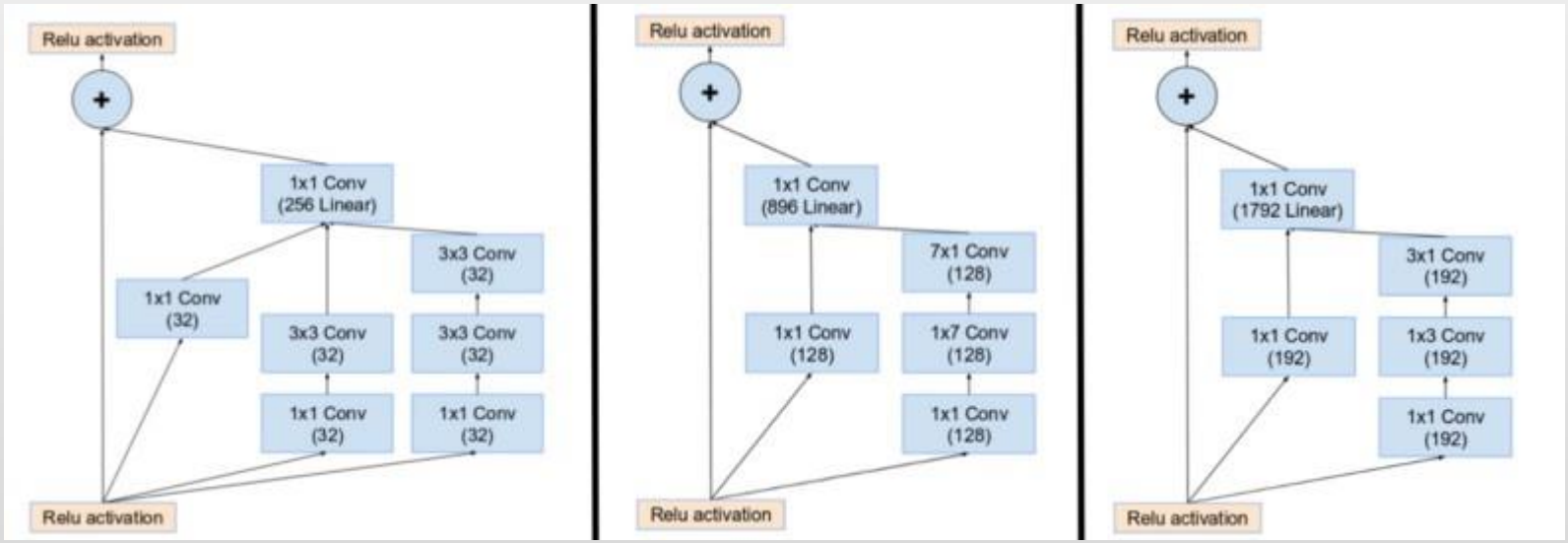
1.7

Other Variations

case	Fig.	ResNet-110	ResNet-164
original Residual Unit [1]	Fig. 4(a)	6.61	5.93
BN after addition	Fig. 4(b)	8.17	6.50
ReLU before addition	Fig. 4(c)	7.84	6.14
ReLU-only pre-activation	Fig. 4(d)	6.71	5.91
full pre-activation	Fig. 4(e)	**6.37**	**5.46**



Inception-ResNet

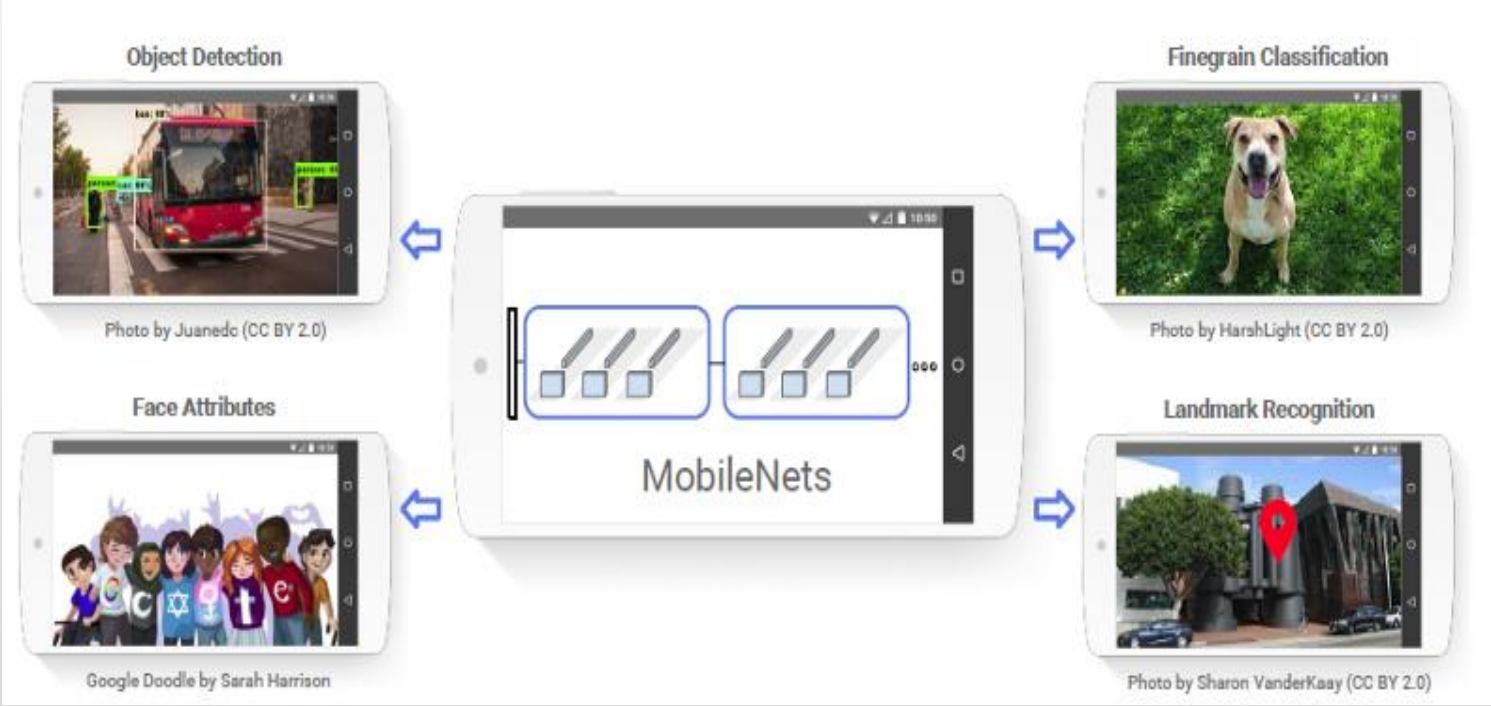


Part1

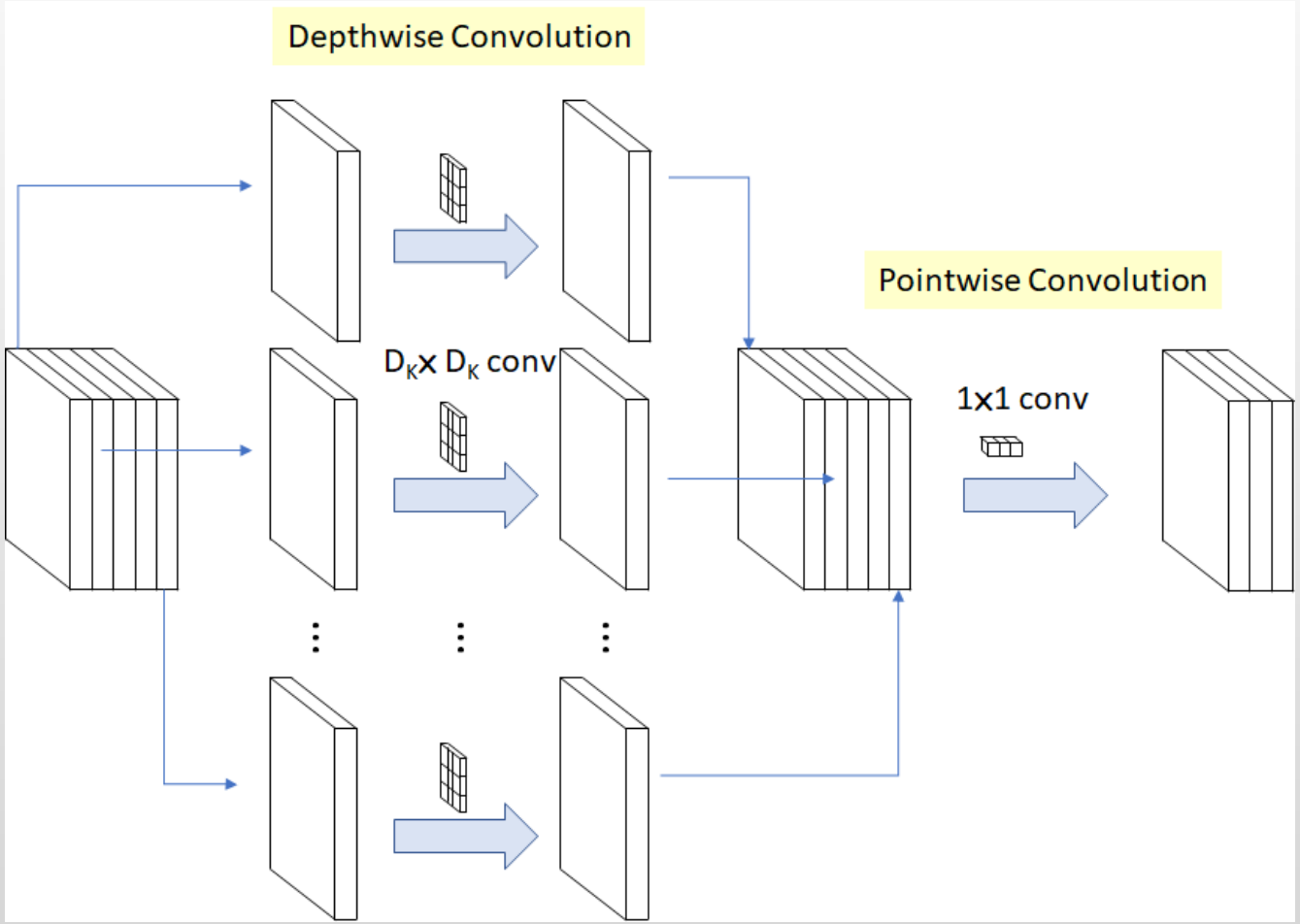
Typical Image_Classification

1.8 MobileNet

- **Depthwise Separable Convolution** is used to reduce the model size and complexity. It is **particularly useful for mobile and embedded vision applications** (Andrew et al., 2017).



Depthwise Separable Convolution



1.8

MobileNet Benchmarks With Width Multiplier

Table 4. Depthwise Separable vs Full Convolution MobileNet

Model	ImageNet Accuracy	Million Mult-Adds	Million Parameters
Conv MobileNet	71.7%	4866	29.3
MobileNet	70.6%	569	4.2

Table 6. MobileNet Width Multiplier

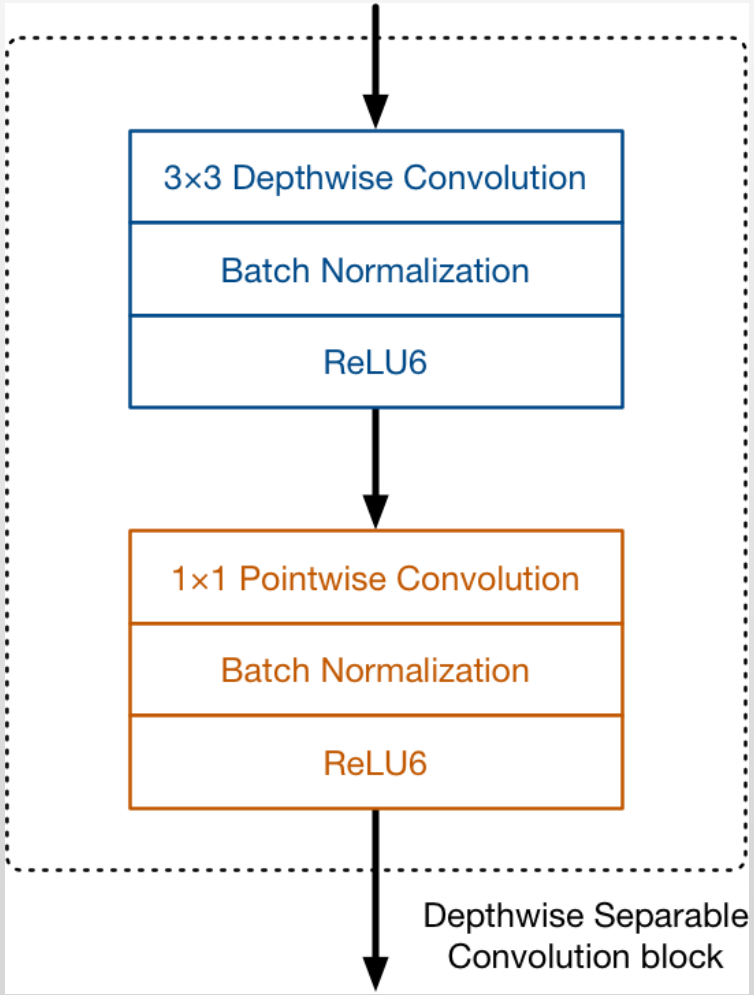
Width Multiplier	ImageNet Accuracy	Million Mult-Adds	Million Parameters
1.0 MobileNet-224	70.6%	569	4.2
0.75 MobileNet-224	68.4%	325	2.6
0.5 MobileNet-224	63.7%	149	1.3
0.25 MobileNet-224	50.6%	41	0.5

Table 8. MobileNet Comparison to Popular Models

Model	ImageNet Accuracy	Million Mult-Adds	Million Parameters
1.0 MobileNet-224	70.6%	569	4.2
GoogleNet	69.8%	1550	6.8
VGG 16	71.5%	15300	138

MobileNet V1 Architecture

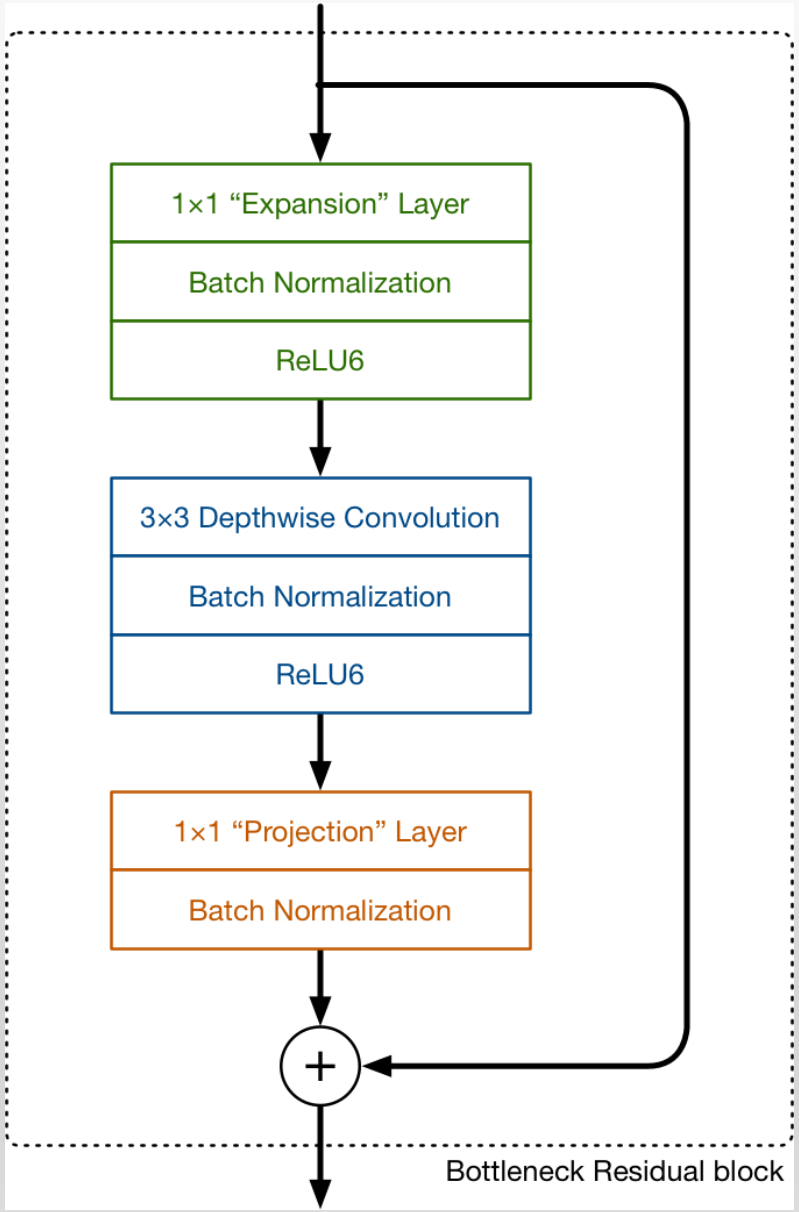
- The full architecture of MobileNet V1 consists of a regular 3×3 convolution as the very first layer, followed by 13 times the above building block.



1.8

MobileNet V2 Architecture

- The first layer is the new kid in the block. This is also a 1×1 convolution. Its purpose is to expand the number of channels in the data before it goes into the depthwise convolution.

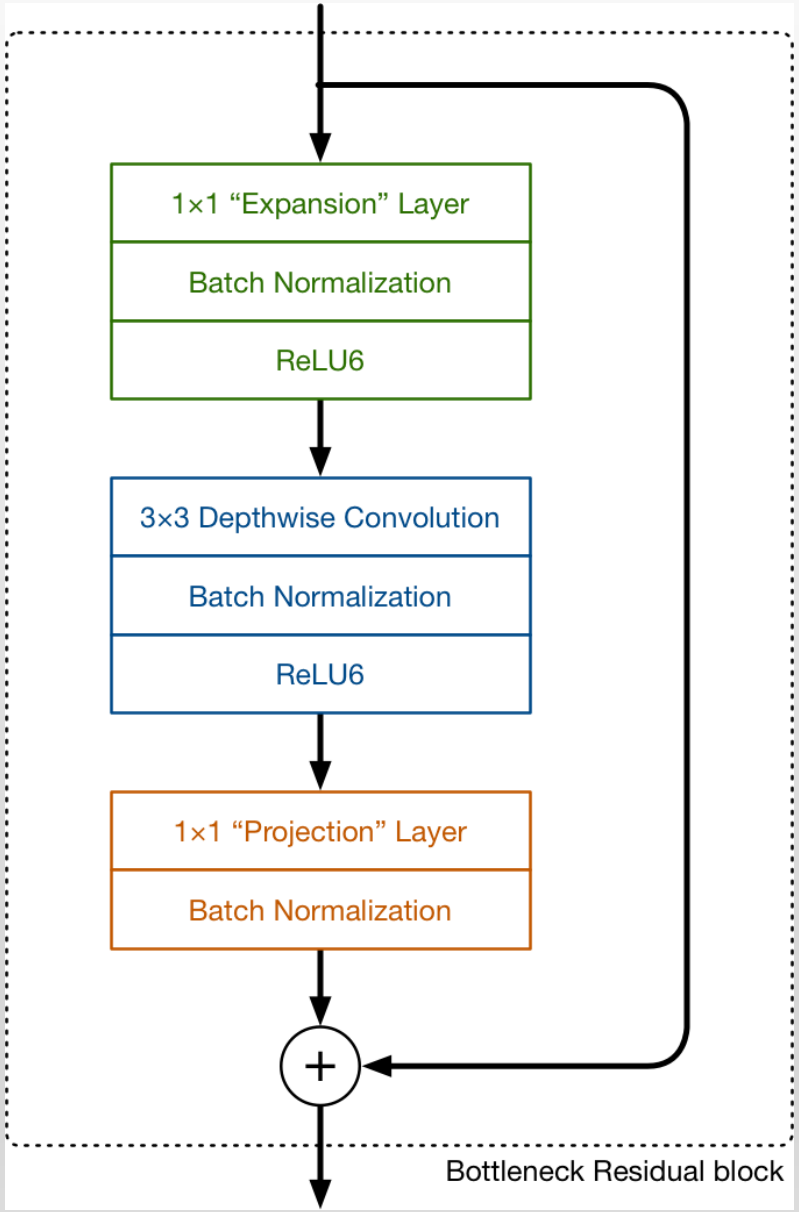


1.8

MobileNet V2 Architecture

- In V2 it does the opposite: it makes the number of channels smaller. This is why this layer is now known as the **projection layer** (**bottleneck layer**)— it projects data with a high number of dimensions (channels) into a tensor with a much lower number of dimensions.

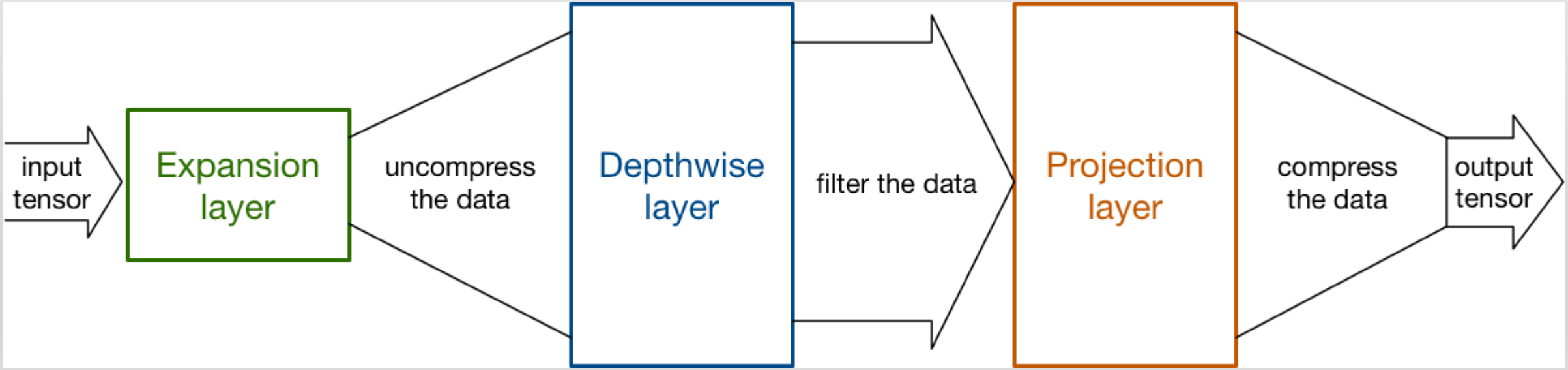
- Residual as ResNet.



Motivation?

- only using low-dimensional tensors doesn't work very well. Applying a convolutional layer to filter a low-dimensional tensor won't be able to extract a whole lot of information (Howard et al., 2018).

Version	Top-1 Accuracy	Top-5 Accuracy
MobileNet V1	70.9	89.9
MobileNet V2	71.8	91.0



1.8

Refrences

[1] D. H. Hubel and T. N. Wiesel, “Receptive fields of single neurones in the cats striate cortex,” The Journal of Physiology, vol. 148, no. 3, pp. 574–591, 1959.

[2] R. A. Weale, “Vision. A Computational Investigation Into the Human Representation and Processing of Visual Information. David Marr,” The Quarterly Review of Biology, vol. 58, no. 2, pp. 299–299, 1983.

[3] DiCarlo, J. J., Zoccolan, D. & Rust, N. C. How does the brain solve visual object recognition? Neuron 73, 415–434 (2012).

[4] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, “Microsoft COCO: Common Objects in Context,” Computer Vision – ECCV 2014 Lecture Notes in Computer Science, pp. 740–755, 2014.

[5] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” 2009 IEEE Conference on Computer Vision and Pattern Recognition, 2009.

R

Refrences

[6] R. Doon, T. K. Rawat, and S. Gautam, “Cifar-10 Classification using Deep Convolutional Neural Network,” 2018 IEEE Punecon, 2018.

[7] A. Kuznetsova, H. Rom, N. Alldrin, J. Uijlings, I. Krasin, J. Pont-Tuset, S. Kamali, S. Popov, M. Malloci, A. Kolesnikov, T. Duerig, and V. Ferrari, “The Open Images Dataset V4,” International Journal of Computer Vision, vol. 128, no. 7, pp. 1956–1981, 2020.

[8] U. Marti and H. Bunke. The IAM-database: An English Sentence Database for Off-line Handwriting Recognition. Int. Journal on Document Analysis and Recognition, Volume 5, pages 39 - 46, 2002.

[9] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, Andrew Y. Ng Reading Digits in Natural Images with Unsupervised Feature Learning NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011.

[10] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” Proceedings of the IEEE, vol. 86, no. 11, pp. 2278–2324, 1998.

Refrences

[11] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman, “VGGFace2: A Dataset for Recognising Faces across Pose and Age,” 2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018), 2018.

[12] S. Yang, P. Luo, C. C. Loy, and X. Tang, “WIDER FACE: A Face Detection Benchmark,” 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[13] Jebara, Tony (2004). Machine Learning: Discriminative and Generative. The Springer International Series in Engineering and Computer Science. Kluwer Academic (Springer).

[14] Tommi Jaakkola and David Haussler (1998), Exploiting Generative Models in Discriminative Classifiers. In *Advances in Neural Information Processing Systems 11*, pages 487–493. MIT Press.

[15] Perronnin, Florent, et al. “Improving the Fisher Kernel for Large-Scale Image Classification.” *Computer Vision – ECCV 2010 Lecture Notes in Computer Science*, 2010, pp. 143–156., doi:10.1007/978-3-642-15561-1_11.

R

Refrences

[16] Krizhevsky, Alex; Sutskever, Ilya; Hinton, Geoffrey E. (2017-05-24). "ImageNet classification with deep convolutional neural networks".

[17] Schrimpf, Martin, et al. “Brain-Score: Which Artificial Neural Network for Object Recognition Is Most Brain-Like?” 2018, doi:10.1101/407007.

[18] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” 3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc., pp. 1–14, 2015.

[19] Szegedy, Christian, et al. “Going Deeper with Convolutions.” *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, doi:10.1109/cvpr.2015.7298594.

[20] Szegedy, Christian, et al. “Rethinking the Inception Architecture for Computer Vision.” *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, doi:10.1109/cvpr.2016.308.

Refrences

[21] He, Kaiming, et al. “Deep Residual Learning for Image Recognition.” *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, doi:10.1109/cvpr.2016.90.

[22] He, Kaiming, et al. “Identity Mappings in Deep Residual Networks.” *Computer Vision – ECCV 2016 Lecture Notes in Computer Science*, 2016, pp. 630–645., doi:10.1007/978-3-319-46493-0_38.

[23] Sandler, Mark, et al. “MobileNetV2: Inverted Residuals and Linear Bottlenecks.” *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, doi:10.1109/cvpr.2018.00474.

[24] Howard, Andrew G., et al. "Mobilenets: Efficient convolutional neural networks for mobile vision applications." *arXiv preprint arXiv:1704.04861* (2017).

R


کالج تخصصی
هوش مصنوعی پارت
مرکز تحقیقات
هوش مصنوعی پارت