

# Bilinear CNN Model for Fine-grained Object Recognition

**Ushasi Chaudhuri**

Center of Studies in Resources Engineering  
IIT Bombay

Credit Seminar Presentation,  
Under the guidance of Prof. Avik Bhattacharya

# Introduction

## Motivation:

- Object recognition is a huge area of research in Image processing.
- With existing models, **Fine-grained classification** is still a challenge.
- One recent work: pairwise feature interactions in a translationally invariant manner.<sup>1</sup>
- Feature extractors are based on CNNs.
- In general, land cover maps in remote sensing images also require fine grained classification properties.
- In future, we would like to study the applicability of this method in remote sensing applications.

---

<sup>1</sup>T. Y. Lin, A. RoyChowdhury, and S. Maji., Bilinear cnn models for fine-grained visual recognition. In 2015 *IEEE International Conference on Computer Vision (ICCV)*, pp 14491457, Dec 2015.

# Convolutional Neural Network

In machine learning, a **convolutional neural network** (CNN, or ConvNet) is a class of deep, feed-forward artificial neural networks that has successfully been applied to analyzing visual imagery.<sup>2</sup>

3

---

<sup>2</sup>[https://en.wikipedia.org/wiki/Convolutional\\_neural\\_network](https://en.wikipedia.org/wiki/Convolutional_neural_network), Accessed on 8.4.2018

<sup>3</sup>[https://raw.githubusercontent.com/vdumoulin/conv\\_arithmetic/master/gif/no\\_padding\\_no\\_strides.gif](https://raw.githubusercontent.com/vdumoulin/conv_arithmetic/master/gif/no_padding_no_strides.gif)

# VGG Net

- VGG network architecture introduced by Zisserman et. al.<sup>4</sup>.
- Very Deep Convolutional Networks for Large Scale Image Recognition.
- Although the concept for 'deep' has changed since then.
- Network characterized by its simplicity.
- Using only  $3 \times 3$  convolutional layers.
- Reducing volume size— max pooling.
- 2 FCN layers ( each 4,096 nodes  $\rightarrow$  softmax classifier).

---

<sup>4</sup>K. Simonyan and A. Zisserman, *Very deep convolutional networks for large-scale image recognition*, CoRR, abs/1409.1556, 2014

# VGG Net

## Pre-training:

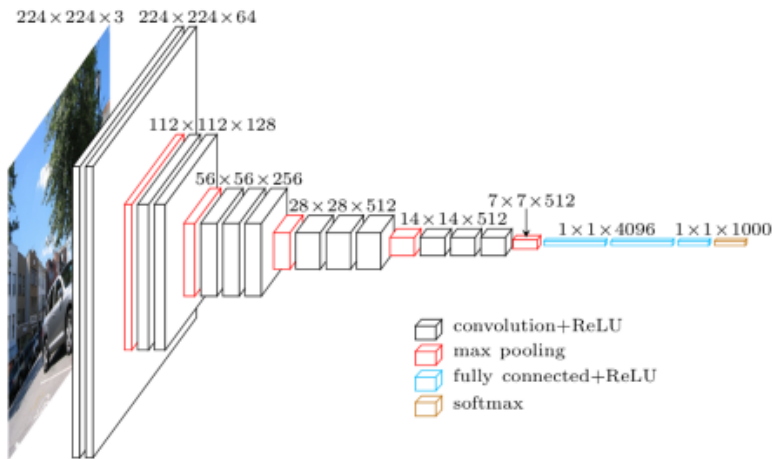
- Used as initialization for the larger, deeper networks.
- Pre-training is time consuming & tedious.
- Requires entire network to be trained before initialization for a deeper network.

## It is built using:

- Convolutions layers (used only  $3 \times 3$  size )
- Max pooling layers (used only  $2 \times 2$  size)
- Fully connected layers at end
- Total 16 layers.

# Architecture

## VGG Net Architecture <sup>5</sup>



<sup>5</sup>

Real-time image-based parking occupancy detection using deep learning - Scientific Figure on ResearchGate.  
Available from: [https://www.researchgate.net/The-architecture-of-a-VGGNet-CNN-after-Wang-et-al-2017\\_fig1323796590](https://www.researchgate.net/The-architecture-of-a-VGGNet-CNN-after-Wang-et-al-2017_fig1323796590)[accessed 10 May, 2018]

# VGG Net

- ① VGG Net is used to find the object label for a given image.
- ② It takes input images of size  $224 \times 224 \times 3$  (RGB images).
- ③ There are two major **drawbacks** with VGG Net:
  - It is painfully slow to train.
  - The network architecture weights themselves are quite large

# VGGNet Configurations

ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input ( $224 \times 224$ RGB image)					
conv3-64	conv3-64 <b>LRN</b>	conv3-64 <b>conv3-64</b>	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 <b>conv3-128</b>	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 <b>conv1-256</b>	conv3-256 conv3-256 <b>conv3-256</b>	conv3-256 conv3-256 conv3-256 <b>conv3-256</b>
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 <b>conv1-512</b>	conv3-512 conv3-512 <b>conv3-512</b>	conv3-512 conv3-512 conv3-512 <b>conv3-512</b>
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 <b>conv1-512</b>	conv3-512 conv3-512 <b>conv3-512</b>	conv3-512 conv3-512 conv3-512 <b>conv3-512</b>
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

6

<sup>6</sup>K. Simonyan and A. Zisserman, *Very deep convolutional networks for large-scale image recognition*, CoRR, abs/1409.1556, 2014



# Bilinear CNN Model

- Proposes a recognition architecture that addresses several drawbacks of both part-based and texture models.
- Consists of 2 feature extractors.
- **Bilinearity** Output derived from their outer product.
- Captures pairwise correlations between the feature channels and part-feature interactions.
- Two stream hypothesis of the visual processing of the human brain.

**The ventral stream is used for object identification and recognition. The dorsal stream helps in processing the objects spatial location relative to the viewer.**

# Bilinear CNN

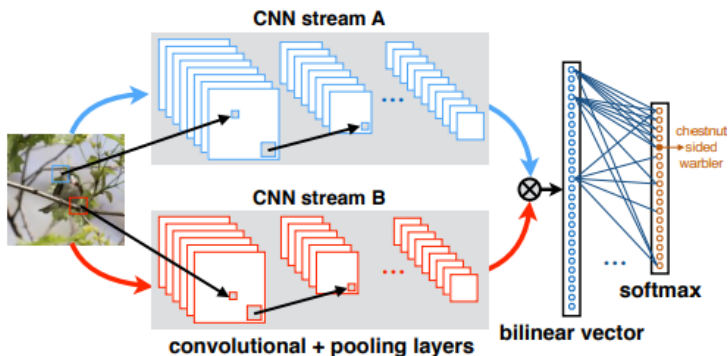


Figure: A bilinear CNN model for image classification. <sup>7</sup>

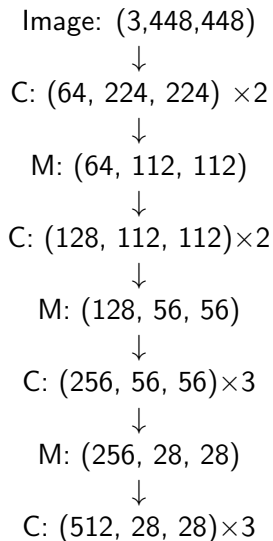
<sup>7</sup>T. Y. Lin, A. RoyChowdhury, and S. Maji., Bilinear cnn models for fine-grained visual recognition. In 2015 *IEEE International Conference on Computer Vision (ICCV)*, pp 1449-1457, Dec 2015.

# Methodology

## Steps for BCNN:

- 1 Convolutional layer, say 'conv64'. If we have a  $H \times W \times 3$  images, we get  $H \times W \times 64$ .
- 2 Maxpooling layer. Size of the matrix:  $H/2 \times W/2 \times 64$ .
- 3 Repeating convolution layer, followed by maxpool layer, and so on.
- 4 Use 16-layer VGG Net, with a D ConvNet configuration (both A and B).
- 5 VGG Net is already trained in the 'ImageNet dataset' (1000 classes). Terminate at first 30 steps of the '*pretrained*' model.

# Methodology



# Methodology

- ⑥ Remove one row and one column, to get: (512, 27, 27).  
Vectorized matrix of the form (512, 27 × 27, 1).
- ⑦ 2 such matrices A & B of dimensions (512, 729, 1).
- ⑧ **bilinearity step**: Find  $A^T B$ .  
(outer product, resultant matrix: (512, 512)).
- ⑨ Signed square root step:

$$y \leftarrow \text{sign}(x) \sqrt{|x|} \quad (1)$$

# Methodology

- 10  $l_2$  normalization

$$z \leftarrow y / \|y\|_2 \quad (2)$$

- 11 For classification, use logistic regression:

$$LR(x) = \frac{\exp(x)}{1 + \exp(x)} \quad (3)$$

- 12 After the logistic regression step. we use a *softmax* layer to classify between the objects.

# Results and Discussions

For our experiments, we have used three datasets.

The experiments have been done on Python, using the PyTorch and TensorFlow library.

The code has also been parallelized by CUDA.

Following datasets have been used:

- CIFAR 10 dataset
- Cats and Dogs dataset
- Stanford Cars dataset

# CIFAR 10 dataset

- 60000 32x32 colour images in 10 classes, with 6000 images per class.
- There are 50000 training images and 10000 test images.<sup>8</sup>

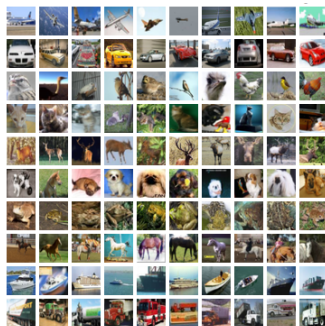



Figure: CIFAR10 Dataset.

<sup>8</sup> Philippe Golle. Machine learning attacks against the asirra captcha, 01 2008. 



# CIFAR 10 dataset

- 10 classes. The labels are as follows:  
0: airplane, 1: automobile, 2: bird, 3: cat, 4: deer, 5: dog, 6: frog, 7: horse, 8: Ship, 9: Truck.
- Sample results on next slide.
- However, the CIFAR10 images are of dimensions  $32 \times 32$  only.
- Also, the classes look quite different from one another.
- Hence, bilinear CNN was not tried on this dataset. The VGG Net on this dataset gave us an accuracy of 92%.

# CIFAR 10 dataset



Target:3, VGG: 3



Target:8, VGG:8



Target:6, VGG:6



Target:1, VGG:1



Target:6, VGG:3



Target: 8, VGG :8



Target:0 , VGG: 8



Target: 6, VGG: 6



Target: 3, VGG: 6



Target:1 , VGG:1

GT: 3 8 8 0 6 6 1 6 3 1 VGG: 3 8 8 8 6 6 1 6 3 1

# Cats and Dogs

- training archive contains 25,000 images of dogs and cats. It has 2 classes, each given a label of 0-1. The labels are as follows:
  - 0: Cat
  - 1: Dog
- The accuracy is found to be 85 % after training for only 5 epochs. It takes around 100 images per minute. So for 250 images, it takes 4 hours per epoch.
- The current state-of-art literature suggests machine classifiers can score above 92.9% accuracy on this task <sup>9</sup>.

---

<sup>9</sup> Philippe Golle. Machine learning attacks against the asirra captcha, 01 2008. 

# Cats and Dogs



Target:0, BCNN: 0   Target: 0, BCNN: 0   Target:0, BCNN:0   Target:0 , BCNN: 0



Target:0, BCNN:1   Target: 1, BCNN: 0   Target:1, BCNN:1   Target: 1, BCNN: 1



Target:1, BCNN:1   Target: 1, BCNN: 1

**Figure:** Cats and Dogs Results. GT: 0 0 0 0 0 1 1 1 1 1 & BCNN: 0 0 0  
0 1 0 1 1 1 1.

# CIFAR 10

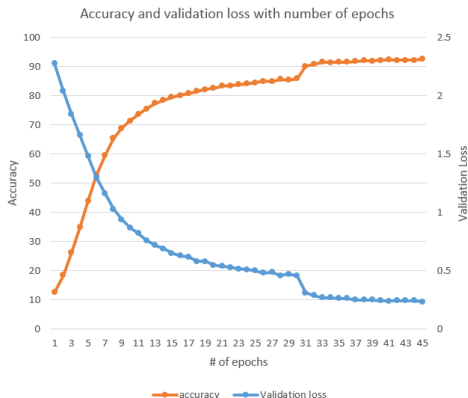


Figure: Number of epoch vs Accuracy and validation loss.

# Stanford Cars Dataset - By the Authors

- Cars dataset contains 16,185 images of 196 classes of cars.
- Data is split into 8,144 training images and 8,041 testing images.
- Each class has been split roughly in a 50-50 split.
- Classes are typically at the level of Make, Model, Year, e.g. 2012 Tesla Model S or 2012 BMW M3 coupe<sup>10</sup>.

<sup>10</sup>

Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In 4th International IEEE Workshop on 3D Representation and Recognition (3dRR-13), Sydney,

# Stanford Cars Dataset - By the Authors

- The paper implemented various combinations of the convolution Net configurations of the VGG Net.
- For a Bilinear CNN using D and D layer, the authors got an accuracy of 82.9%.
- Each class has been split roughly in a 50-50 split.
- Classes are typically at the level of Make, Model, Year, e.g. 2012 Tesla Model S or 2012 BMW M3 coupe <sup>11</sup>.

---

<sup>11</sup> Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In 4th International IEEE Workshop on 3D Representation and Recognition (3dRR-13), Sydney, Australia, 2013.

# Stanford Cars Dataset - By the Authors

method	birds		birds + box		aircrafts		cars		FPS
	w/o ft	w/ ft	w/o ft	w/ ft	w/o ft	w/ ft	w/o ft	w/ ft	
FV-SIFT	18.8	-	22.4	-	61.0	-	59.2	-	10 <sup>†</sup>
FC-CNN [M]	52.7	58.8	58.0	65.7	44.4	57.3	37.3	58.6	124
FC-CNN [D]	61.0	70.4	65.3	76.4	45.0	74.1	36.5	79.8	43
FV-CNN [M]	61.1	64.1	67.2	69.6	64.3	70.1	70.8	77.2	23
FV-CNN [D]	71.3	74.7	74.4	77.5	70.4	77.6	75.2	85.7	8
B-CNN [M,M]	72.0	78.1	74.2	80.4	72.7	77.9	77.8	86.5	87
B-CNN [D,M]	80.1	<b>84.1</b>	81.3	<b>85.1</b>	78.4	83.9	83.9	<b>91.3</b>	8
B-CNN [D,D]	80.1	84.0	80.1	84.8	76.8	<b>84.1</b>	82.9	90.6	10
Previous work	<b>84.1</b> [19], 82.0 [21] 73.9 [38], 75.7 [2]		82.8 [21], 73.5 [24] 73.0 [7], 76.4 [38]		72.5 [4], 80.7 [16]		<b>92.6</b> [21], 82.7 [16] 78.0 [4]		<sup>†</sup> on a cpu

Figure: Results on various dataset, as experimented by the authors <sup>12</sup>.

<sup>12</sup>T. Y. Lin, A. RoyChowdhury, and S. Maji., Bilinear cnn models for fine-grained visual recognition. In 2015 IEEE International Conference on Computer Vision (ICCV), pp 14491457, Dec:2015.



# Conclusions

- Deep convolutional Networks can provide very high accuracy for image classification.
- Fine grained object classification technique.
- Its aspects on various databases ([D D] BCNN).
- In future, study its extend-ability in remote sensing problems, like semantic classification.
- **Two stream hypothesis of the visual processing of the human brain, making the object location and orientation invariant.**

# Thank You