

# Convolutional Neural Networks

---

Nipun Batra

IIT Gandhinagar

August 30, 2025

# Convolutional Neural Networks

# Imagenet

14 million images, 20K categories



# Imagenet

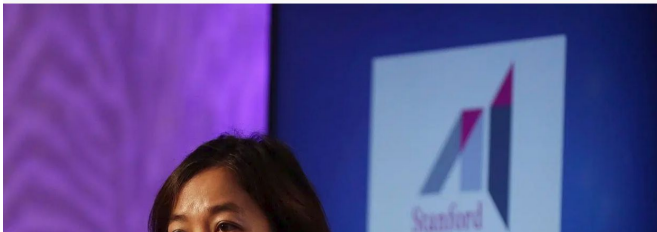
IT'S NOT ABOUT THE ALGORITHM

## The data that transformed AI research—and possibly the world

July 26, 2017



By **Dave Gershgorn**  
Contributor



<https://qz.com/1034972/the-data-that-changed-the-direction-of-ai-research-and-possibly-the-world/>

# Imagenet

- Circa 2006, AI community: “a better algorithm would make better decisions, regardless of the data.”
- Fei Fei Li thought: “the best algorithm wouldn’t work well if the data it learned from didn’t reflect the real world”
- “We decided we wanted to do something that was completely historically unprecedented,” Li said, referring to a small team who would initially work with her. “We’re going to map out the entire world of objects.

# Imagenet

- ImageNet: published in 2009 as a research poster stuck in the corner of a Miami Beach conference center, the dataset quickly evolved into an annual competition to see which algorithms could identify objects in the dataset's images with the lowest error rate.
- "The paradigm shift of the ImageNet thinking is that while a lot of people are paying attention to models, let's pay attention to data," Li said. "Data will redefine how we think about models."

# WordNet



- **S. (n) Eskimo dog, husky** (breed of heavy-coated Arctic sled dog)
  - *Alaskan husky* / *Alaskan husky* / *Alaskan husky*
- **S. (n) working dog** (any of several breeds of usually large powerful dogs bred to work as draft animals and guard and guide dogs)
  - **S. (n) dog, domestic dog, Canis familiaris** (a member of the genus Canis (probably descended from the common wolf) that has been domesticated by man since prehistoric times; occurs in many breeds) "the dog has had all night"
  - **S. (n) canis, canid** (any of various dog-like mammals with nonretractile claws and typically long snout)
    - **S. (n) canis** (a terrestrial or aquatic flesh-eating mammal) "terrestrial carnivores have four or five clawed digits on each limb"
    - **S. (n) placental, placental mammal, eutherian, eutherian mammal** (mammals having a placenta; all mammals except marsupials and monotremes)
      - **S. (n) eutherian, eutherian** (any warm-blooded vertebrate having the skin more or less covered with hair; young are born alive except for the small subclasses of monotremes and associated with milk)
      - **S. (n) vertebrate, vertebrate** (animals having a bony or cartilaginous skeleton with a segmented spinal column and a large brain enclosed in a skull or cranium)
        - **S. (n) chordate** (any animal of the phylum Chordata having a notochord or spinal column)
          - **S. (n) animal, animal; letur, letur, letur, letur, letur, letur** (a living organism characterized by voluntary movement)
          - **S. (n) organism, being** (a living thing that has (or can develop) the ability to act or function independently)
            - **S. (n) form, form, animate, form** (a living (or once living) entity)
              - **S. (n) whole, set** (an assemblage of parts that is regarded as a single entity) "the leg is that part compared to the whole"; "the team is a set"
              - **S. (n) object, physical object** (a tangible and visible entity; an entity that can cast a shadow) "it was full of rocks, balls and other objects"
              - **S. (n) physical entity** (an entity that has physical existence)
                - **S. (n) entity** (that which is perceived or known or inferred to have its own distinct existence (being or soundings))

# WordNet

- In the late 1980s, Princeton psychologist George Miller started a project called WordNet, with the aim of building a hierarchical structure for the English language.
- For example, within WordNet, the word “dog” would be nested under “canine,” which would be nested under “mammal,” and so on. It was a way to organize language that relied on machine-readable logic, and amassed more than 155,000 indexed words.



## Back to Imagenet

- Finding the perfect algorithm seemed distant, Li says. She saw that previous datasets didn't capture how variable the world could be—even just identifying pictures of cats is infinitely complex.
- If you only saw five pictures of cats, you'd only have five camera angles, lighting conditions, and maybe variety of cat. But if you've seen 500 pictures of cats, there are many more examples to draw commonalities from.
- Having read about WordNet's approach, Li met with professor Christiane Fellbaum, a researcher influential in the continued work on WordNet, during a 2006 visit to Princeton. Fellbaum had the idea that WordNet could have an image associated with each of the words, more as a reference rather than a computer vision dataset.

## Back to Imagenet

- Li's first idea was to hire undergraduate students for \$10 an hour to manually find images and add them to the dataset. But back-of-the-napkin math quickly made Li realize that at the undergrads' rate of collecting images it would take 90 years to complete.
- Undergrads were time-consuming, algorithms were flawed, and the team didn't have money—Li said the project failed to win any of the federal grants she applied for, receiving comments on proposals that it was shameful Princeton would research this topic, and that the only strength of proposal was that Li was a woman.
- A solution finally surfaced in a chance hallway conversation with a graduate student who asked Li whether she had heard of Amazon Mechanical Turk, a service where hordes of humans sitting at computers around the world would complete small online tasks for pennies.

# Back to Imagenet

[Main](#) [Instructions](#) [Unsure? Look up in Wikipedia](#) [Google](#) [\[Additional input\]](#) [No good photos? Have expertise? comments? Click here!](#)

**First time workers please click here for instructions.**

Click on the photos that contain the object or depict the concept of: **delta** a low triangular area of alluvial deposits where a river divides before entering a larger body of water; "the Mississippi River delta"; "the Nile delta" *PLEASE READ DEFINITION CAREFULLY!*  
Pick as many as possible. **PHOTOS ONLY, NO PAINTINGS, DRAWINGS**, etc. It's OK to have other objects, multiple instances, occlusion or text in the image.

*Do not use back or forward button of your browser: OCCASIONALLY THERE MIGHT BE ADULT OR DISTURBING CONTENT.*



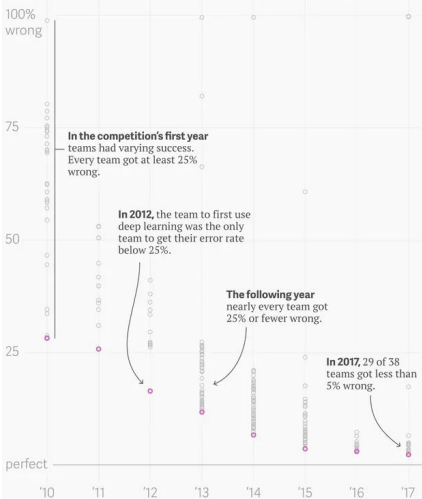
Below are the photos you have selected FROM THIS PAGE ONLY (they will be saved when you navigate to other pages). Click to deselect.

what's this?

## Back to Imagenet

- Even after finding Mechanical Turk, the dataset took two and a half years to complete. It consisted of 3.2 million labelled images, separated into 5,247 categories, sorted into 12 subtrees like “mammal,” “vehicle,” and “furniture.”
- In 2009, Li and her team published the ImageNet paper with the dataset—to little fanfare. Li recalls that CVPR, a leading conference in computer vision research, only allowed a poster, instead of an oral presentation, and the team handed out ImageNet-branded pens to drum up interest. People were skeptical of the basic idea that more data would help them develop better algorithms.
- “There were comments like ‘If you can’t even do one object well, why would you do thousands, or tens of thousands of objects?’”

## ImageNet Large Scale Visual Recognition Challenge results

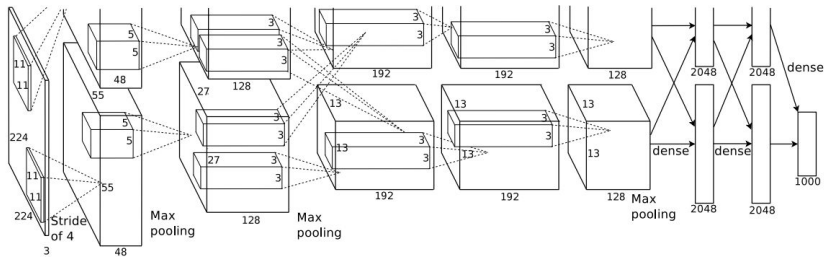


David Yanofsky | Quartz

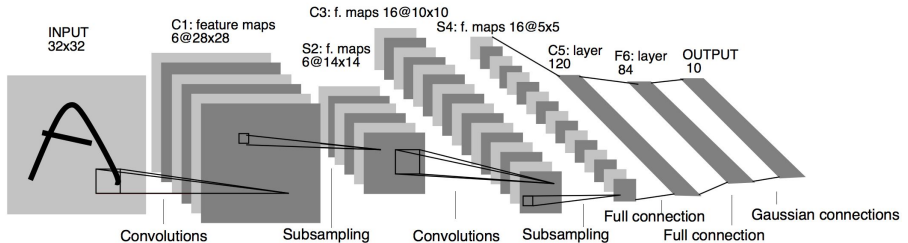
Data: ImageNet

14 million images, 20K categories

# History (AlexNet 2012)



# History (LeCun 1998)



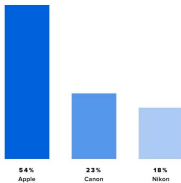
# Modern day cameras



YEAR IN REVIEW

## TOP BRANDS — 2017

The most popular brands used by the Flickr community  
(Percentage of photographers)





## Modern day cameras



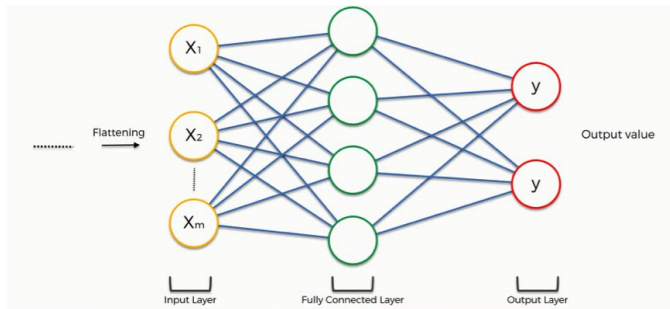
xiaomi

# Mi Note 10 is coming

World's first 108MP penta camera



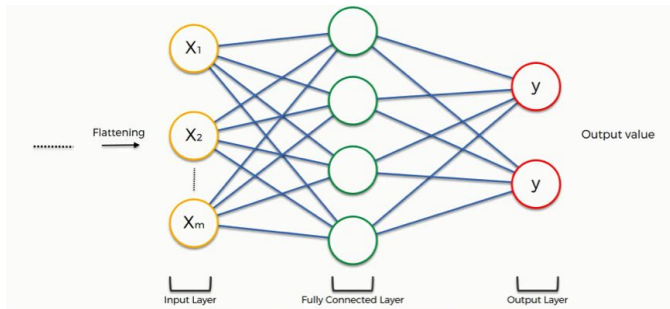
# Modern day cameras suitability for MLPs?



Courtesy:

<https://www.superdatascience.com/convolutional-neural-networks-cnn-step-4-full-connection/>

# Modern day cameras suitability for MLPs?

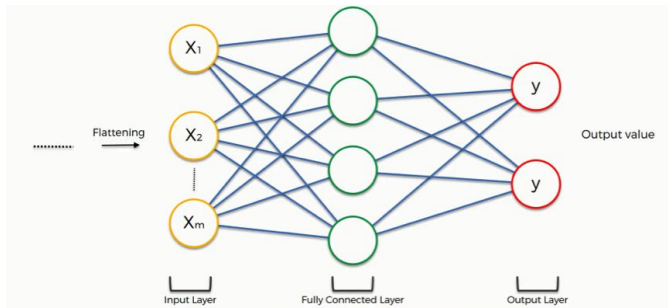


Courtesy:

<https://www.superdatascience.com/convolutional-neural-network-ks-cnn-step-4-full-connection/>

1. If we are classifying cats vs dogs and hidden layer size is 100, what is number of parameters?

# Modern day cameras suitability for MLPs?



Courtesy:

<https://www.superdatascience.com/convolutional-neural-network-ks-cnn-step-4-full-connection/>

1. If we are classifying cats vs dogs and hidden layer size is 100, what is number of parameters?
2.  $N[1] = 100$ ,  $N[0] = 108 \times 1M \times 3$  (for RGB channel)  $\rightarrow$  Billions of params
3. Size of weight matrix assuming each param is 32 bytes is 32 bytes  $\times$  324 billion  $\rightarrow$  several GBs

## Are MLPs well suited for images?



Courtesy:

<https://www.rd.com/advice/pets/common-cat-myths/>

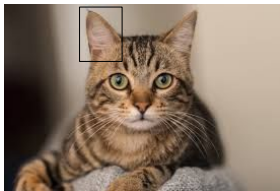


Courtesy:

<https://www.goodhousekeeping.com/life/pets/q21525625/why-cats-are-best-pets/>

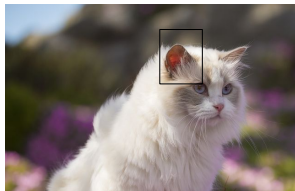
Are both of the above cats?

## Are MLPs well suited for images?



Courtesy:

<https://www.rd.com/advice/pets/common-cat-myths/>

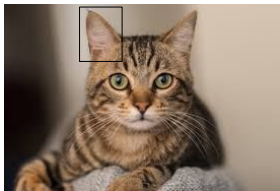


Courtesy:

<https://www.goodhousekeeping.com/life/pets/q21525625/why-cats-are-best-pets/>

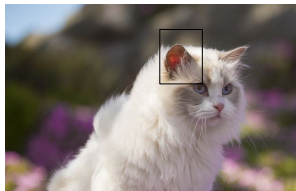
Assume both are 100X100 images and bounded rectangle are 10X10 pixels

## Are MLPs well suited for images?



Courtesy:

<https://www.rd.com/advice/pets/common-cat-myths/>



Courtesy:

<https://www.goodhousekeeping.com/life/pets/q21525625/why-cats-are-best-pets/>

A cat ear is a cat ear, irrespective of the location in the image.

MLP would see these are different input features

Rather, we need “feature detector” that is **translation invariant**.

# Are MLPs well suited for images?



Similar  
pixel  
values

Courtesy:

<https://www.rd.com/advice/pets/common-cat-myths/>

MLPs assume all input features to be independent

But, we have a **spatially local** structure, nearby pixels are similar



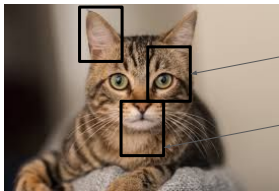
Courtesy:

<https://www.goodhousekeeping.com/life/pets/q21525625/why-cats-are-best-pets/>



# Key Idea

Ear detector



Eye  
detector

Face  
detector

Courtesy:

<https://www.rd.com/advice/pets/common-cat-myths/>



Courtesy:

<https://www.goodhousekeeping.com/life/pets/q21525625/why-cats-are-best-pets/>

Build **local feature** detectors

# Building Block: Filters and Convolution Operation

(A guide to convolution arithmetic for deep learning)

Filter

0	1	2
2	2	0
0	1	2

# Building Block: Filters and Convolution Operation

(A guide to convolution arithmetic for deep learning)

Input

$3_0$	$3_1$	$2_2$	1	0
$0_2$	$0_2$	$1_0$	3	1
$3_0$	$1_1$	$2_2$	2	3
2	0	0	2	2
2	0	0	0	1

Output

12.0	12.0	17.0
10.0	17.0	19.0
9.0	6.0	14.0

# Building Block: Filters and Convolution Operation

(A guide to convolution arithmetic for deep learning)

Input

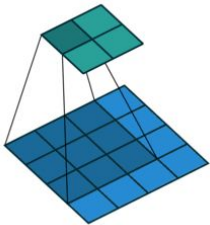
3	$3_0$	$2_1$	$1_2$	0
0	$0_2$	$1_2$	$3_0$	1
3	$1_0$	$2_1$	$2_2$	3
2	0	0	2	2
2	0	0	0	1

Output

12.0	12.0	17.0
10.0	17.0	19.0
9.0	6.0	14.0

# Building Block: Filters and Convolution Operation

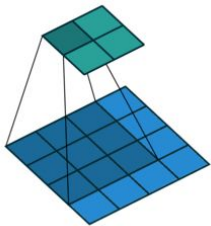
(A guide to convolution arithmetic for deep learning)



Notebook demonstration (edge detection)

# Building Block: Filters and Convolution Operation

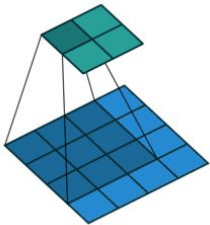
(A guide to convolution arithmetic for deep learning)



Given input image of  $n \times n$  and filter of size:  $f \times f$ ,  
what is the size of the output?

# Building Block: Filters and Convolution Operation

(A guide to convolution arithmetic for deep learning)



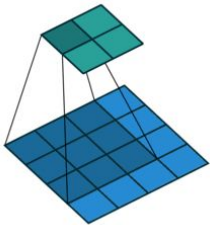
Given input image of  $n \times n$  and filter of size:  $f \times f$ ,  
what is the size of the output?

$n-f+1 \times n-f+1$



# Building Block: Filters and Convolution Operation

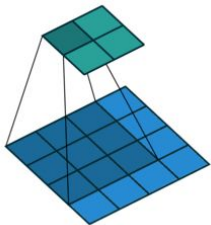
(A guide to convolution arithmetic for deep learning)



Start with a  $32 \times 32$  image and repeated operations of a single  $5 \times 5$  filter, after how many such operations will we have a  $1 \times 1$  output?

# Building Block: Filters and Convolution Operation

(A guide to convolution arithmetic for deep learning)

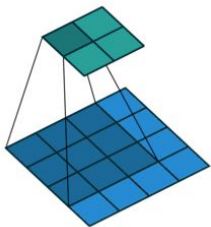


Start with a 32 X 32 image and repeated operations of a single 5 X 5 filter, after how many such operations will we have a 1 X 1 output?

Iteration	$n$	$f$	$n-f+1$
1	32	5	28
2	28	5	24
3	24	5	20
4	20	5	16
...	...	...	...

# Building Block: Filters and Convolution Operation

(A guide to convolution arithmetic for deep learning)



Problem 1: Can not go very deep with repeated convolution as image size reduces quickly

Start with a 32 X 32 image and repeated operations of a single 5 X 5 filter, after how many such operations will we have a 1 X 1 output?

Iteration	$n$	$f$	$n-f+1$
1	32	5	28
2	28	5	24
3	24	5	20
4	20	5	16
...	...	...	...

# Building Block: Filters and Convolution Operation

(A guide to convolution arithmetic for deep learning)

3	3 <sub>0</sub>	2 <sub>1</sub>	1 <sub>2</sub>	0
0	0 <sub>2</sub>	1 <sub>2</sub>	3 <sub>0</sub>	1
3	1 <sub>0</sub>	2 <sub>1</sub>	2 <sub>2</sub>	3
2	0	0	2	2
2	0	0	0	1

12.0	12.0	17.0
10.0	17.0	19.0
9.0	6.0	14.0

How many times is left-most pixel used in a calculation?

# Building Block: Filters and Convolution Operation

(A guide to convolution arithmetic for deep learning)

3	$3_0$	$2_1$	$1_2$	0
0	$0_2$	$1_2$	$3_0$	1
3	$1_0$	$2_1$	$2_2$	3
2	0	0	2	2
2	0	0	0	1

12.0	12.0	17.0
10.0	17.0	19.0
9.0	6.0	14.0

How many times is left-most pixel used in a calculation?

Only once!

# Building Block: Filters and Convolution Operation

(A guide to convolution arithmetic for deep learning)

3	3 <sub>0</sub>	2 <sub>1</sub>	1 <sub>2</sub>	0
0	0 <sub>2</sub>	1 <sub>2</sub>	3 <sub>0</sub>	1
3	1 <sub>0</sub>	2 <sub>1</sub>	2 <sub>2</sub>	3
2	0	0	2	2
2	0	0	0	1

12.0	12.0	17.0
10.0	17.0	19.0
9.0	6.0	14.0

How many times is left-most pixel used in a calculation?

Only once!

How many times is a middle pixel used in a calculation?

Many times. For example, the middle pixel with value 2 used nine times!

# Building Block: Filters and Convolution Operation

(A guide to convolution arithmetic for deep learning)

3	3 <sub>0</sub>	2 <sub>1</sub>	1 <sub>2</sub>	0
0	0 <sub>2</sub>	1 <sub>2</sub>	3 <sub>0</sub>	1
3	1 <sub>0</sub>	2 <sub>1</sub>	2 <sub>2</sub>	3
2	0	0	2	2
2	0	0	0	1

12.0	12.0	17.0
10.0	17.0	19.0
9.0	6.0	14.0

Problem 2: The corner pixels are under-utilised

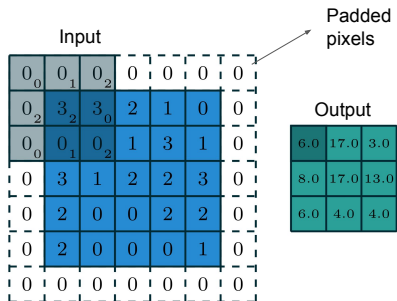
How many times is left-most pixel used in a calculation?

Only once!

How many times is a middle pixel used in a calculation?

Many times. For example, the middle pixel with value 2 used nine times!

# Building Block: Padding





# Building Block: Padding

0	0	0	0	0	0	0
0	3	3	2	1	0	0
0	0	0	1	3	1	0
0	3	1	2	2	3	0
0	2	0	0	2	2	0
0	2	0	0	0	1	0
0	0	0	0	0	0	0

6.0	17.0	3.0
6.0	17.0	13.0
6.0	4.0	4.0

0	0	0	0	0	0	0
0	3	3	2	1	0	0
0	0	0	1	3	1	0
0	3	1	2	2	3	0
0	2	0	0	2	2	0
0	2	0	0	0	1	0
0	0	0	0	0	0	0

6.0	17.0	4.0
6.0	17.0	13.0
6.0	4.0	4.0

0	0	0	0	0	0	0
0	3	3	2	1	0	0
0	0	0	1	3	1	0
0	3	1	2	2	3	0
0	2	0	0	2	2	0
0	2	0	0	0	1	0
0	0	0	0	0	0	0

6.0	17.0	3.0
6.0	17.0	13.0
6.0	4.0	4.0

0	0	0	0	0	0	0
0	3	3	2	1	0	0
0	0	0	1	3	1	0
0	3	1	2	2	3	0
0	2	0	0	2	2	0
0	2	0	0	0	1	0
0	0	0	0	0	0	0

6.0	17.0	3.0
6.0	17.0	13.0
6.0	4.0	4.0

0	0	0	0	0	0	0
0	3	3	2	1	0	0
0	0	0	1	3	1	0
0	3	1	2	2	3	0
0	2	0	0	2	2	0
0	2	0	0	0	1	0
0	0	0	0	0	0	0

6.0	17.0	3.0
6.0	17.0	13.0
6.0	4.0	4.0

0	0	0	0	0	0	0
0	3	3	2	1	0	0
0	0	0	1	3	1	0
0	3	1	2	2	3	0
0	2	0	0	2	2	0
0	2	0	0	0	1	0
0	0	0	0	0	0	0

6.0	17.0	3.0
6.0	17.0	13.0
6.0	4.0	4.0

0	0	0	0	0	0	0
0	3	3	2	1	0	0
0	0	0	1	3	1	0
0	3	1	2	2	3	0
0	2	0	0	2	2	0
0	2	0	0	0	1	0
0	0	0	0	0	0	0

6.0	17.0	3.0
6.0	17.0	13.0
6.0	4.0	4.0

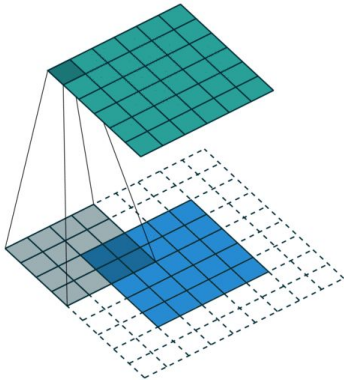
0	0	0	0	0	0	0
0	3	3	2	1	0	0
0	0	0	1	3	1	0
0	3	1	2	2	3	0
0	2	0	0	2	2	0
0	2	0	0	0	1	0
0	0	0	0	0	0	0

6.0	17.0	3.0
6.0	17.0	13.0
6.0	4.0	4.0

0	0	0	0	0	0	0
0	3	3	2	1	0	0
0	0	0	1	3	1	0
0	3	1	2	2	3	0
0	2	0	0	2	2	0
0	2	0	0	0	1	0
0	0	0	0	0	0	0

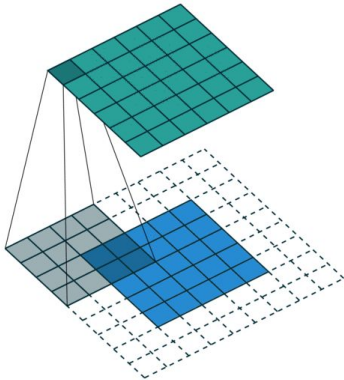
6.0	17.0	3.0
6.0	17.0	13.0
6.0	4.0	4.0

## Building Block: Padding



Ques: Given padding of  $p$  pixel,  $n \times n$  image and filter  $f \times f$ , what is the output size?

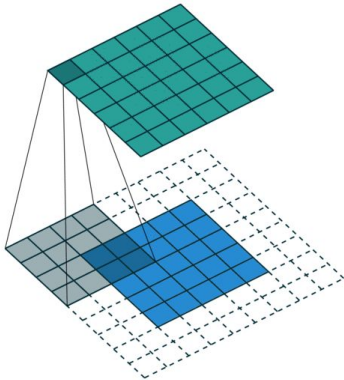
## Building Block: Padding



Ques: Given padding of  $p$  pixel,  $n \times n$  image and filter  $f \times f$ , what is the output size?

$$n+2p-f+1 \times n+2p-f+1$$

## Building Block: Padding

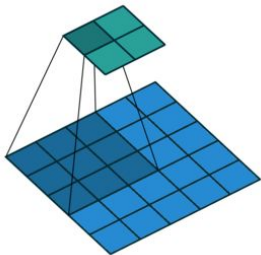


Ques: Given padding of  $p$  pixel,  $n \times n$  image and filter  $f \times f$ , what is the output size?

$$n+2p-f+1 \times n+2p-f+1$$

Same padding: when  $n+2p-f+1 = n$  or,  
 $p = (f-1)/2$

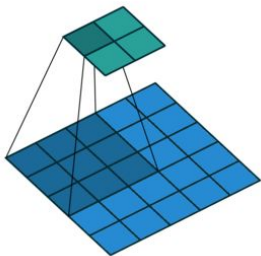
## Building Block: Strides (subsampling)



Skip every  $s$  pixels

Ques: Given  $p$  padding,  $n \times n$  image,  $f \times f$  filter,  $s$  stride, what is output length?

## Building Block: Strides (subsampling)



Skip every  $s$  pixels

Ques: Given  $p$  padding,  $n \times n$  image,  $f \times f$  filter,  $s$  stride, what is output length?

$$\lfloor (n+2p-f)/s \rfloor + 1 \times \lfloor (n+2p-f)/s \rfloor + 1$$

## Building Block: Pooling (subsampling)

3	3	2	1	0
0	0	1	3	1
3	1	2	2	3
2	0	0	2	2
2	0	0	0	1

3.0	3.0	3.0
3.0	3.0	3.0
3.0	2.0	3.0

Max pooling

Similar to filter and convolution operation, but, gives the max value in the  $f \times f$  as the output

## Building Block: Pooling (subsampling)

3	3	2	1	0
0	0	1	3	1
3	1	2	2	3
2	0	0	2	2
2	0	0	0	1

3.0	3.0	3.0
3.0	3.0	3.0
3.0	2.0	3.0

Max pooling

Similar to filter and convolution operation, but, gives the max value in the  $f \times f$  as the output

Works well in practice  
Reduces representation size



## Building Block: Pooling (subsampling)

3	3	2	1	0
0	0	1	3	1
3	1	2	2	3
2	0	0	2	2
2	0	0	0	1

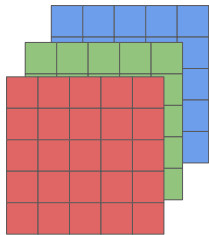
1.7	1.7	1.7
1.0	1.2	1.8
1.1	0.8	1.3

Average pooling

Similar to filter and convolution operation, but, gives the average value in the  $f \times f$  as the output

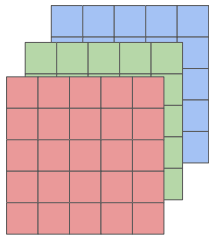
Works well in practice  
Reduces representation size

## Building Block: Multiple channels

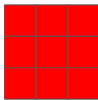


Input:  $n \times n \times c$   
image

## Building Block: Multiple channels

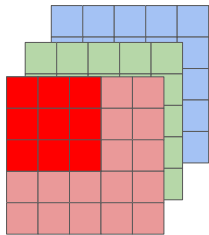


Input:  $n \times n \times c$   
image



Filter for  $r$   
channel:  $f \times f$

## Building Block: Multiple channels



Input:  $n \times n \times c$   
image

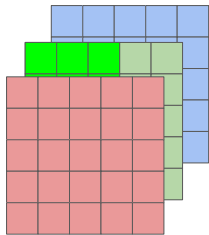


Filter for  $r$   
channel:  $f \times f$



Output for  $r$   
channel:  $n-f+1 \times$   
 $n-f+1$

## Building Block: Multiple channels



Input:  $n \times n \times c$   
image

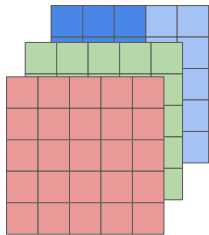


Filter for  $g$   
channel:  $f \times f$



Output for  $g$   
channel:  $n-f+1 \times$   
 $n-f+1$

## Building Block: Multiple channels



Input:  $n \times n \times c$   
image

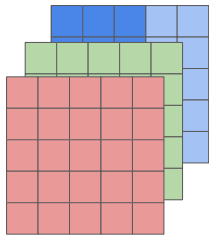


Filter for  $b$   
channel:  $f \times f$

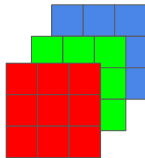


Output for  $b$   
channel:  $n-f+1 \times$   
 $n-f+1$

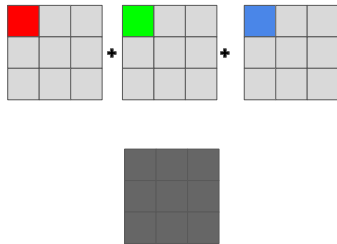
## Building Block: Multiple channels



Input:  $n \times n \times c$   
image

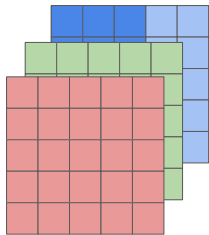


Filter for 3  
channel:  $f \times f \times 3$

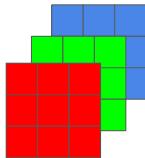


Output for 3  
channel:  $n-f+1 \times$   
 $n-f+1 \times 1$

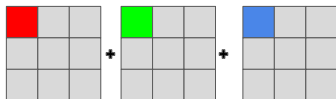
## Building Block: Non-linearity



Input:  $n \times n \times c$   
image



Filter for 3  
channel:  $f \times f \times 3$

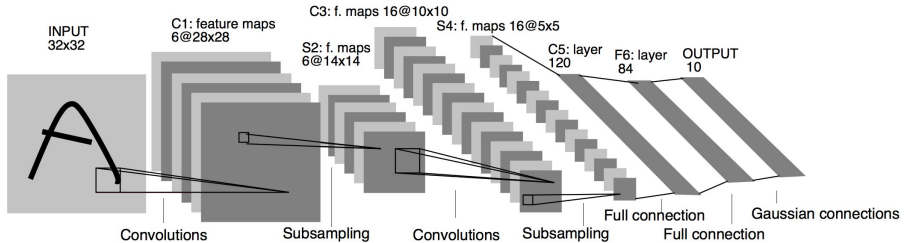


$$g(\text{3x3 grid} + b)$$

Activation Output  
for 3 channel:  
 $n-f+1 \times n-f+1 \times 1$

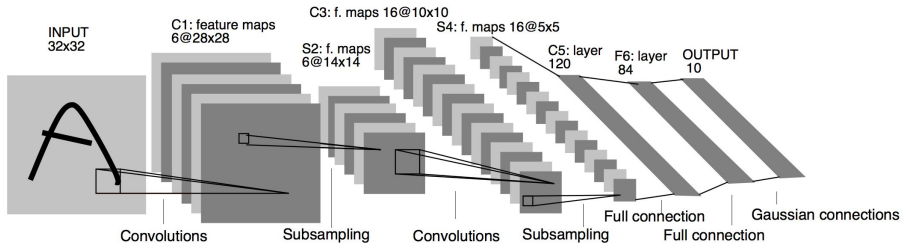


# Exercise LeNet-5



# Exercise LeNet-5

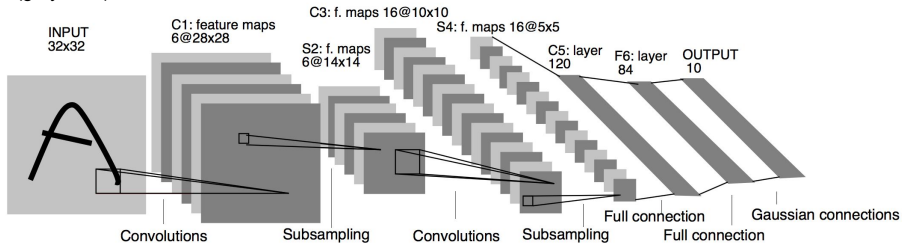
Q1: What is input size?



# Exercise LeNet-5

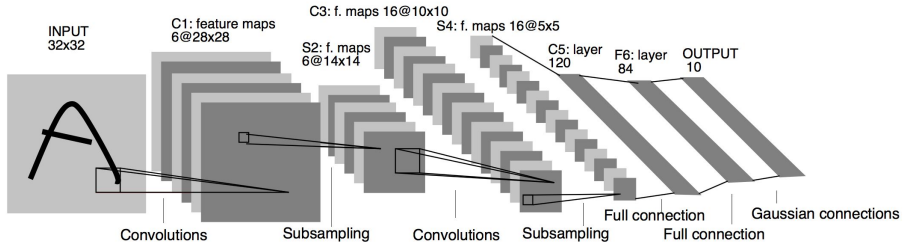
Q1: What is input size?

32X32X1  
(grayscale)



# Exercise LeNet-5

Q2: What is filter size for first layer  
(assume no padding)



# Exercise LeNet-5

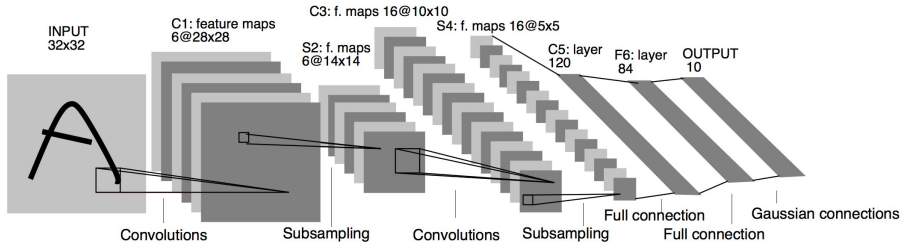
Q2: What is filter size for first layer (assume no padding, 1 stride)

$$5 \times 5: 32 \rightarrow 32 - 5 + 1 = 28$$



# Exercise LeNet-5

Q3: What is number of filters used in first layer?



# Exercise LeNet-5

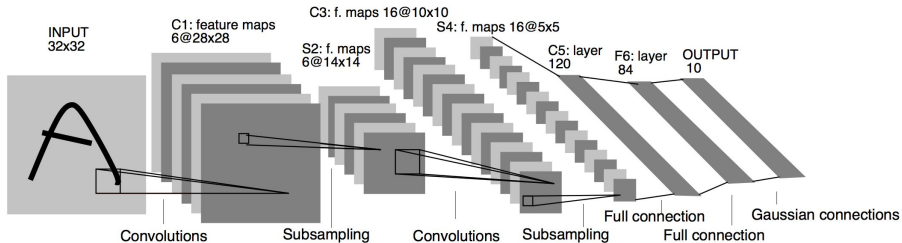
Q3: What is number of filters used in first layer?

6



# Exercise LeNet-5

Q4: What is size of pool filter?

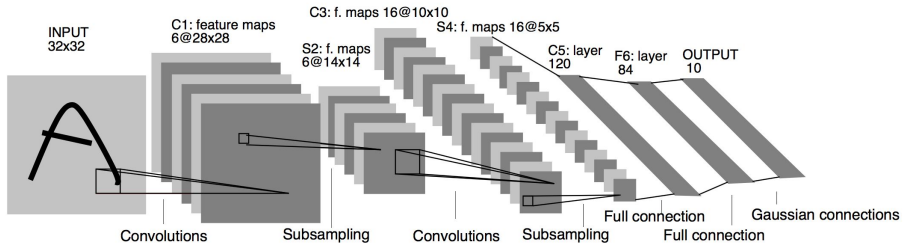




# Exercise LeNet-5

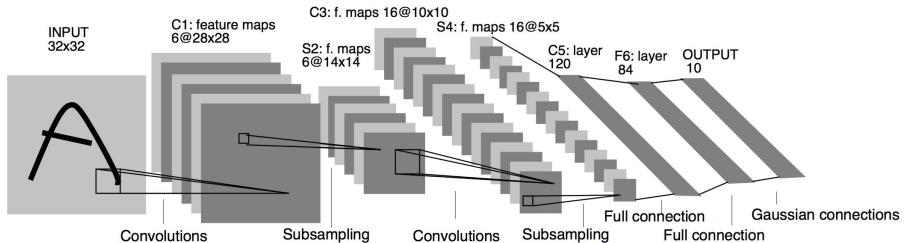
Q4: What is size of pool filter?

$f=2$ ,  $s=2$  (stride 2)



# Exercise LeNet-5

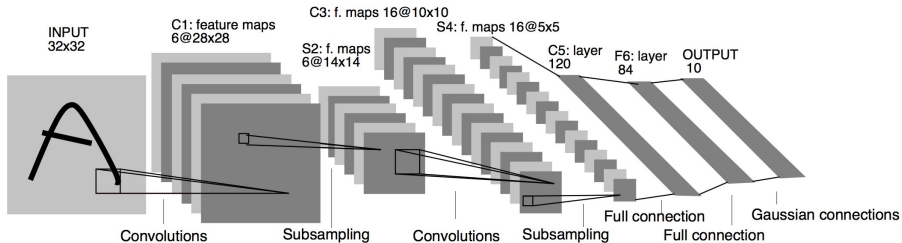
Q5: What is size of filter  
for this layer convolution?



# Exercise LeNet-5

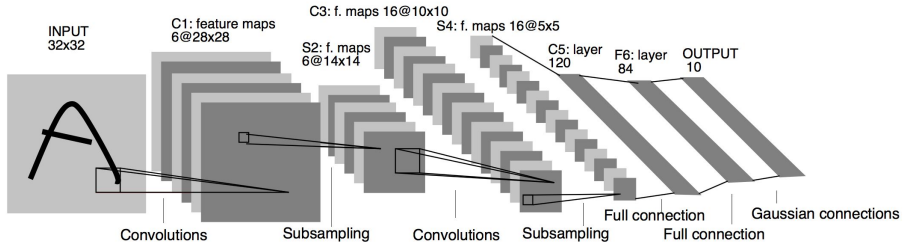
Q5: What is size and number of filter for this layer convolution?

16 filter 5X5 size with stride 1



# Exercise LeNet-5

Q6: What is size of this pool layer?



# Exercise LeNet-5

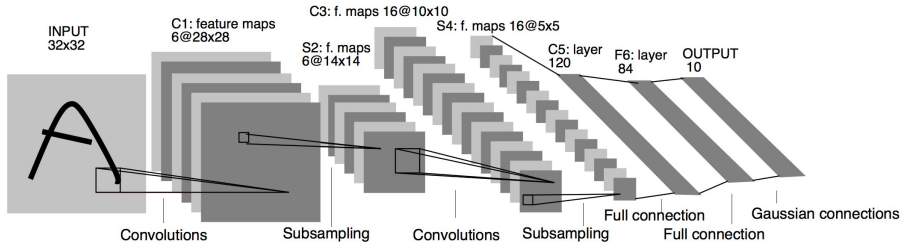
Q6: What is size of this pool layer?

$f=2, s=2$



# Exercise LeNet-5

Q7: This layer is connected to an MLP like layer, how?



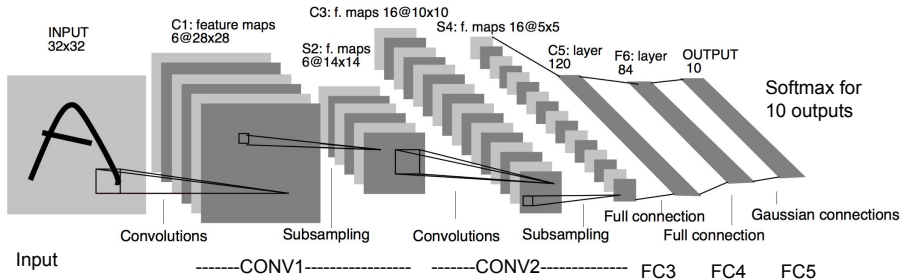
# Exercise LeNet-5

Q7: This layer is connected to an MLP like layer, how?

We flatten 16X5X5 to create a 400X1 matrix



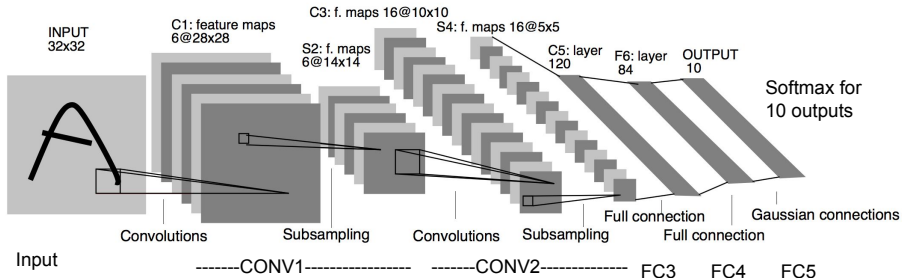
# Exercise LeNet-5





# Exercise LeNet-5

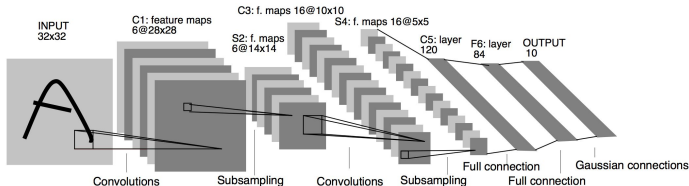
What is the total number of parameters?



# Exercise LeNet-5

What is the total number of parameters?

- CONV1: 6 filters of size 5 X 5 X 1(channel) =  $(6 \cdot 5 \cdot 5) + 6$  biases = 156
- POOL1: No params
- CONV2: 16 filters of size 5 X 5 X 6(six channels) =  $(16 \cdot 5 \cdot 5 \cdot 6) + 16$  biases = 2416
- FC1: Weight matrix of size 120 X 400 + 120 biases = 48120
- FC2: Weight matrix of size 84 X 120 + 84 biases = 10164
- FC3: Weight matrix of size 10 X 84 + 10 biases = 850
- Total = 61,706



# Notebook: LeNet-5, AlexNet, VGG-16

- Notebook

# Training CNNs for own applications

- Train fully from scratch
- Transfer learning -- store activations

# Visualising CNNs

- t-SNE or PCA on last hidden layer ... MNIST
- Same exercise on Imagenet? ..