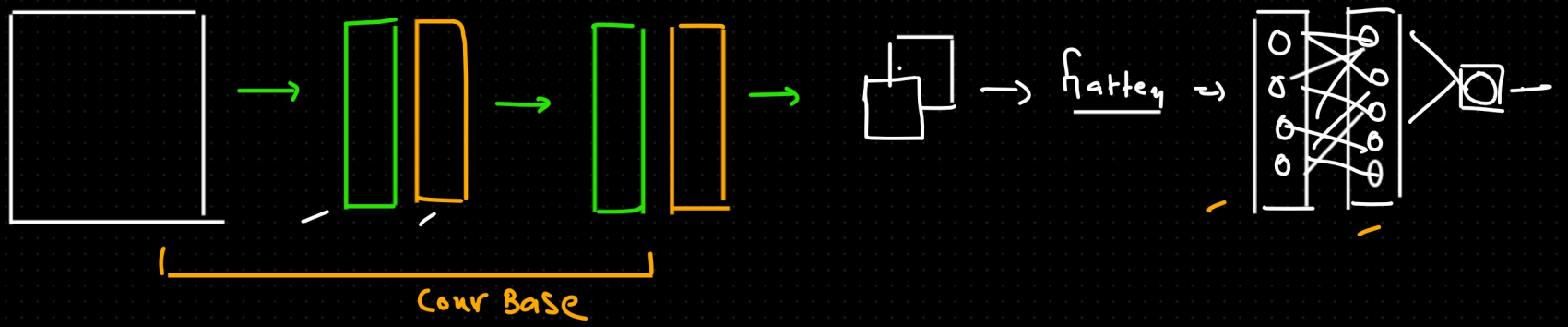
CNN Architecture



{Conv, filters, Stride, Padding, Pooling} Activation, optimizer, Layer, Models, Do, BN

= Experimental | Reasearch | Project

1. LENET
2. Alexnet
3. VGG 16, 19
4. Googlenet
5. Resnet

14 million ⇒ 1.4 cr ⇒ 20,000

IMAGENET ⇒ Deep Learning

(ILSVRC)

Large Scale Visule Recoqnition challenge

⇓

- LENET / LENET-5 = 1989 - Yann Lucan

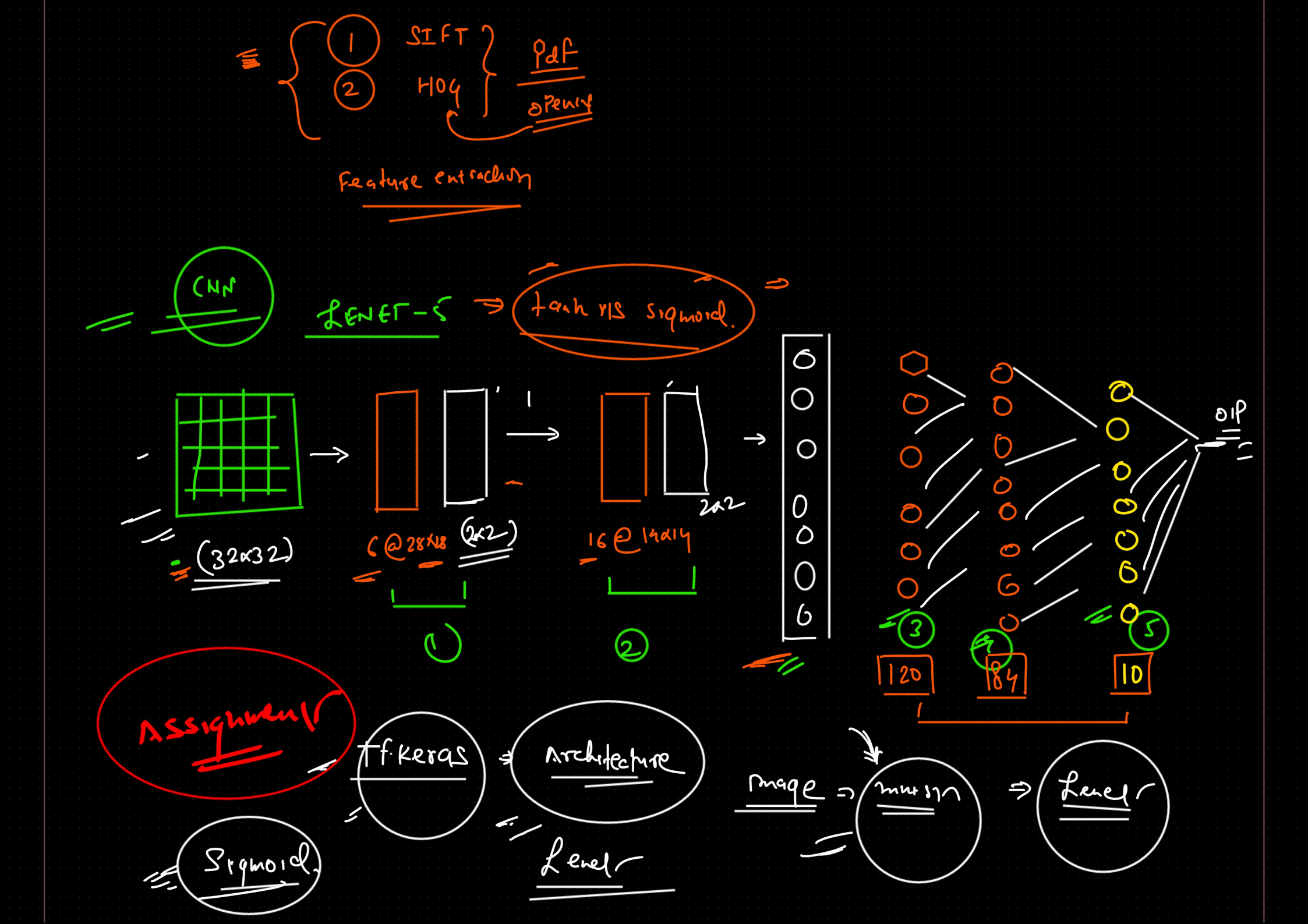
1998 ⇒ Lenet

MNIST

0, 1, 2 ... 9

CNN ⇒ Learning (filter) (BP) f-P ⇔ BP

US-Navy ⇒ Machine ⇓ Number


1
2
SIFT
HOG
pdf
openCV
Feature extraction
CNN
LENET-5
tanh r/s Sigmoid.
(32x32)
6@28x28
(2x2)
2x2
16@14x14
1
2
3
4
5
120
84
10
O/P
Assignment
Tf.keras
Architecture
Sigmoid
Level
Image
Level

Parameter ?

Alexnet ⇒ 2012

Why Pertrained

1. time consuming → training (10000) 2-20 mins (100000) ⇒ Deep network ⇒ Week
2. resource consuming
3. cost expensive.
4. Data ⇒ huge ⇒ well manner

Data hungry

100000, 1000000 ⇒ training

Scrape, Arrange, Label

training ⇒ workaround ⇒ Data (clean, label)
↳ resource

After train ⇒ model

- VGG -
- Resnet -
- google -
- Inception 1,2,3
- Efficient

Pretrain

Which Data ?

Imagenet Data ⇒ load (ferfer Li)

ML, DL

Data

Database

image

Database ⇒ 14 million images ⇒ 20,000 category

1.4 color

Dog
- GR
- CGLRG
- PUG
- RB

Proper manner

Imagenet → 2009

Real Data subset

{10 million, 1000 class}

Flower
- Rose
- L

ILSVRC

2010

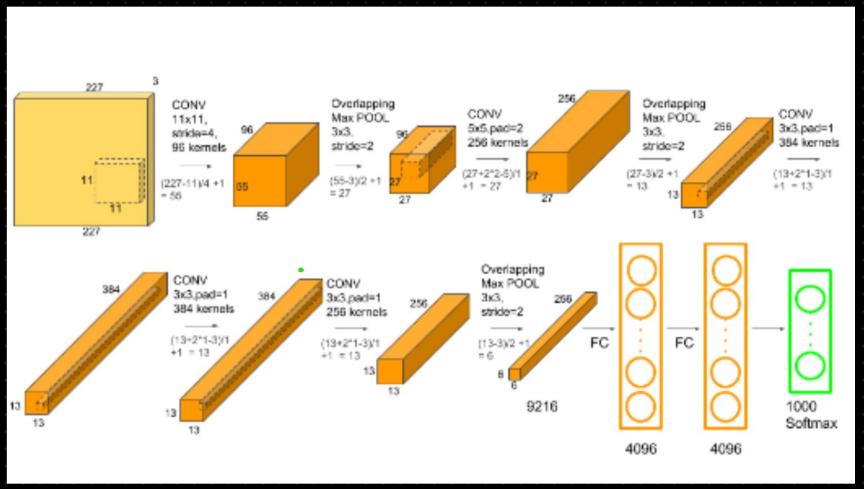
2010 ⇒ ML (SIFT, HOG) ⇒ 28-30 %. Loss

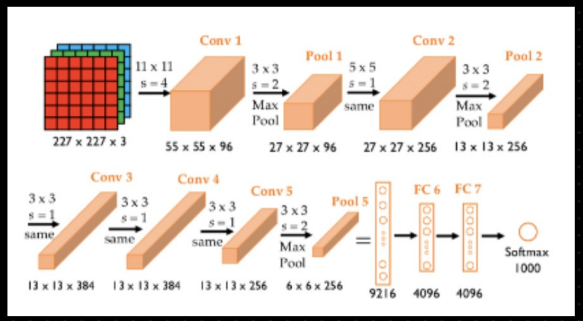
2011 ⇒ 25 %

2012 ⇒ Jeffrey Hinton → Alexnet ⇒ 16 %.

Alexnet
- GPU
- Relu
- Deep Architecture

error rate





Lener, Alexnet

(ILSVRC)

2010/2011 ⇒ ML ⇒ SIFT, HOG ⇒ (25-28 %)

2012 ⇒ Alexnet ⇒ CNN (feature extraction) ⇒ 16 %

2013 ⇒ ZFNET ⇒ CNN ⇒ 11 %

2014 ⇒ VGG ⇒ CNN ⇒ 7 %

2015 ⇒ Google net (Inception) ⇒ Parallel conv ⇒ 6 %

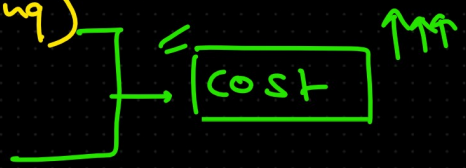
DL

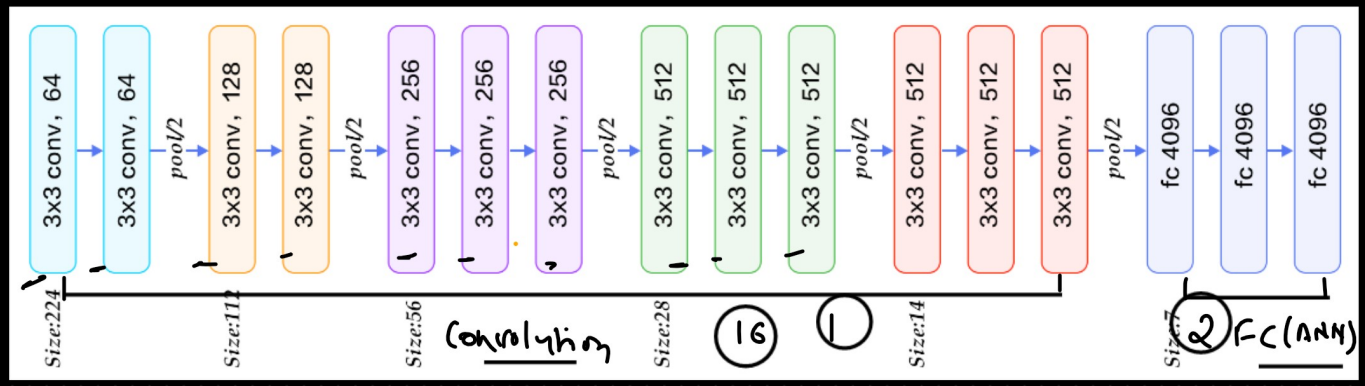
2016 ⇒ Resnet (Skip connection) (Residual block) ⇒ 3 %

Transfer Learning

CNN ⇒ training → Data (clean, label, arrang)

training → high resources



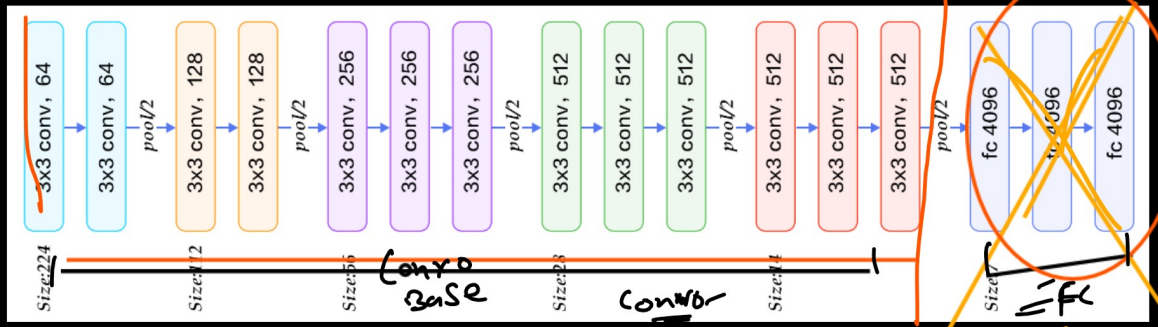


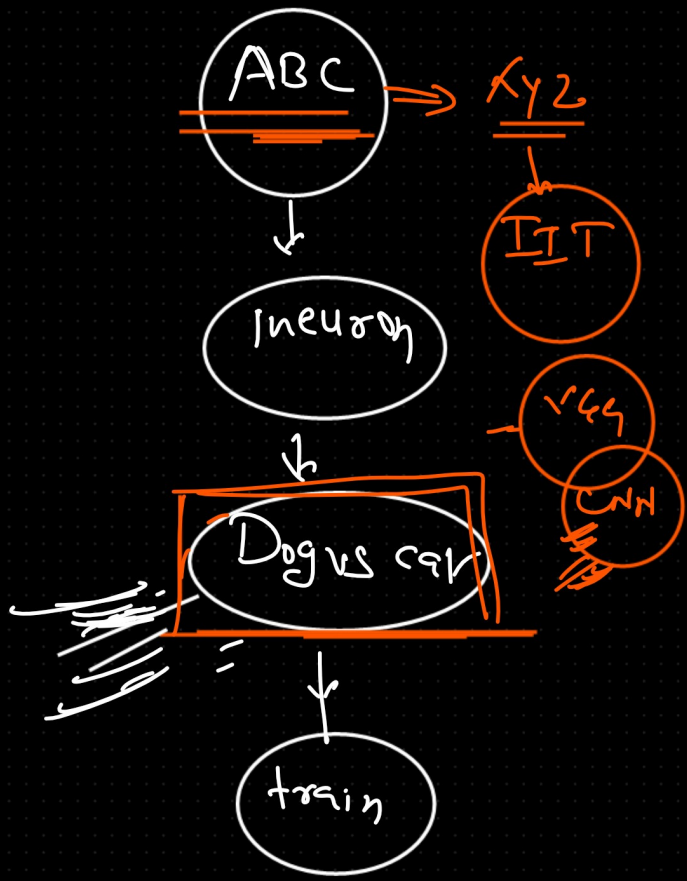
Data → Imagenet ⇒ 1000, 1 million ⇒ CNN Architecture (VGG)

- Dog
- car
- lion
- man
- women
- Ball
- cricket
- chair

Transfer learning ⇒

1. Transfer
2. Learning





ConvoBase + myfc

transfer

H_1	H_2	OP
10	20	1

Transfer Learning

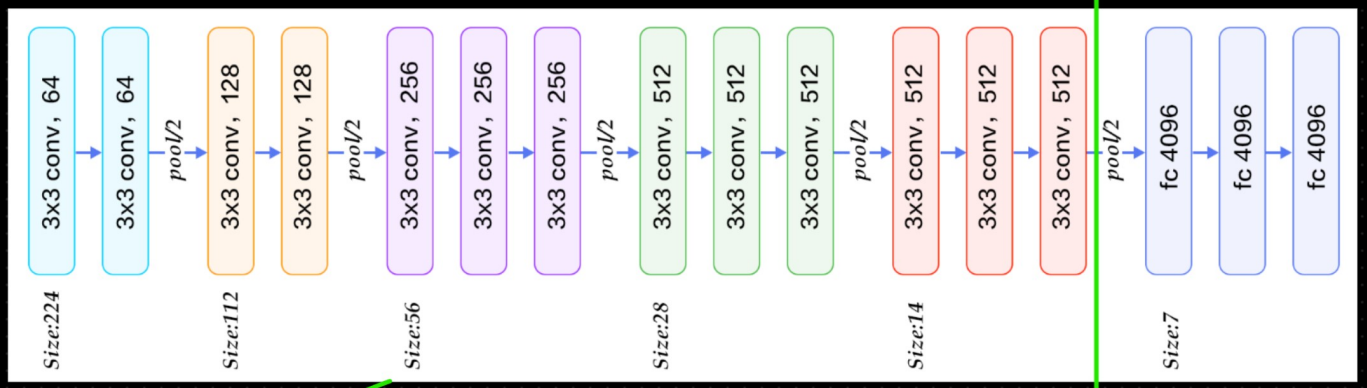
Transfer

1. we extract the base convolution network

2. then we connect our own fully connected layer.

Learning

Fine tuning



Imagenet

13

14

minor miss head

weight => 15

CNN

Iphone 13, 14 | 15

Imagenet

↳ 14

↳ 13

↳ 15 X

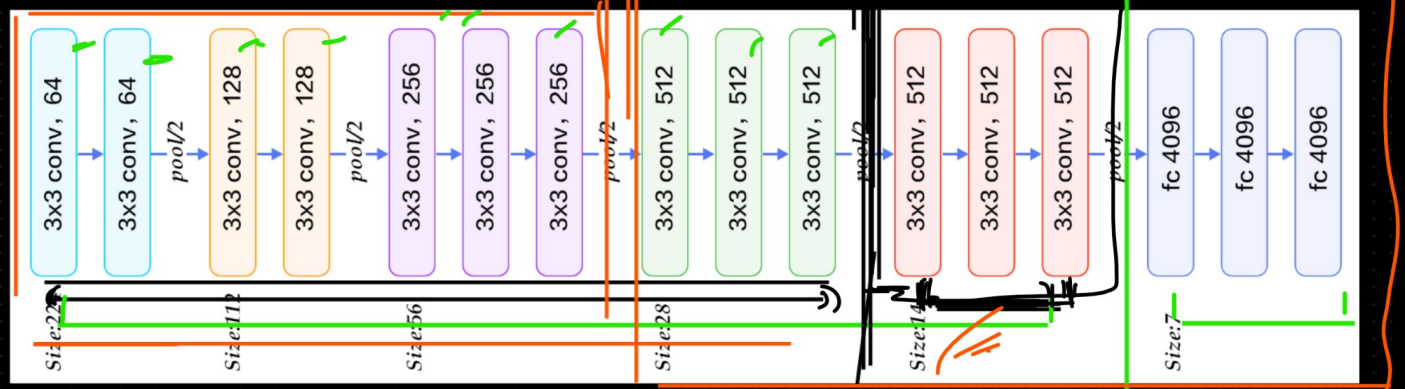
⇒ Convbase (Vgg) ← Image

finetune

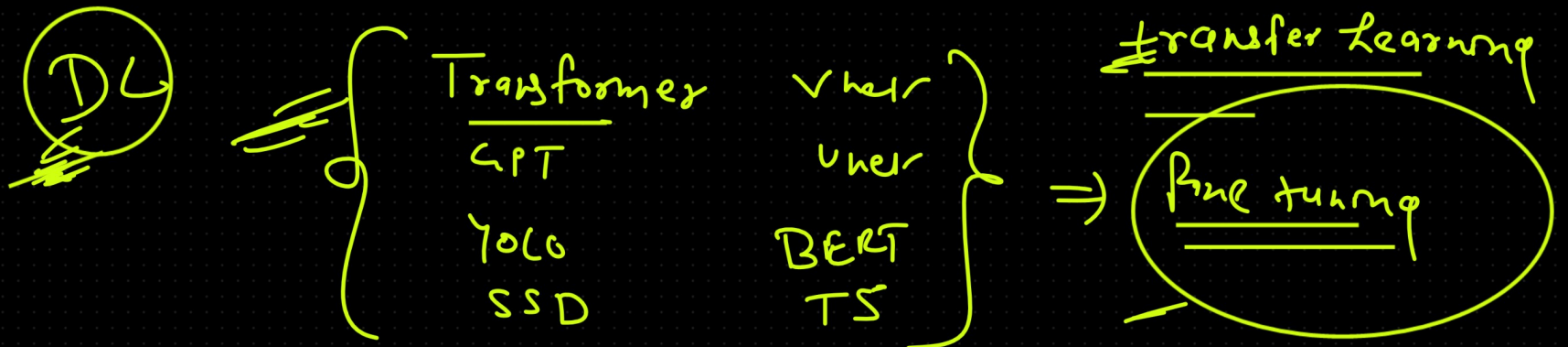
CNN ⇒ I 13, 14, 15

D, C, Alley

Vgg → image → 1000



fine tuning =


DL
Transformers
GPT
YOLO
SSD
Vneir
Uneir
BERT
T5
transfer learning
fine tuning