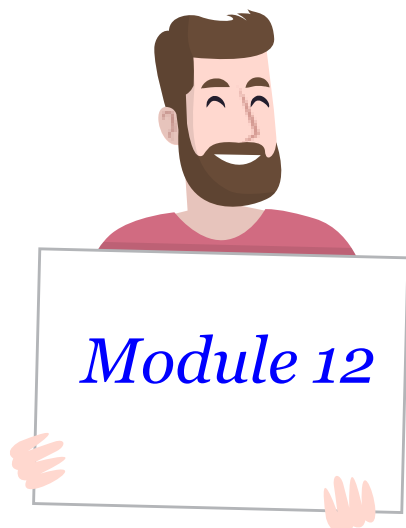




著名的CNN神經網路



designed by  freepik

Estimated time:
45 min.



資訊工業策進會 Institute for Information Industry

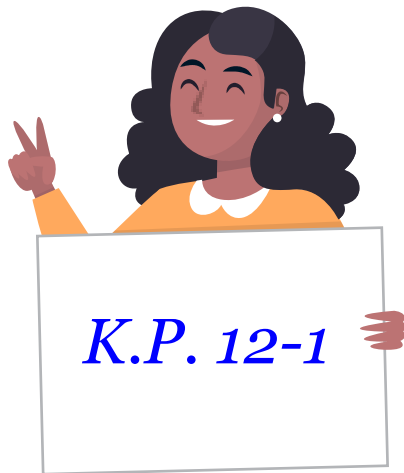
學習目標

- 12-1: ILSVRC比賽
- 12-2: AlexNet與VGG
- 12-3: GoogleNet與ResNet



12-1: ILSVRC比賽

- ILSVRC比賽介紹
- ImageNet介紹
- ILSVRC歷年成績



designed by freepik

ILSVRC比賽介紹

- ILSVRC是一個非常著名的影像分類比賽
 - 給予每位參加者一堆圖片，期望能用影線辨識的方法做圖片分類
 - 在過往，每年都會號召全世界的影像專家來比賽
 - 由於這幾年影像領域有驚人的突破，主辦單位認為該問題已經被解決很好了，所以其2017年以後就停辦了
 - 該比賽使用了ImageNet這個資料集做比賽

IMGENET

ImageNet介紹

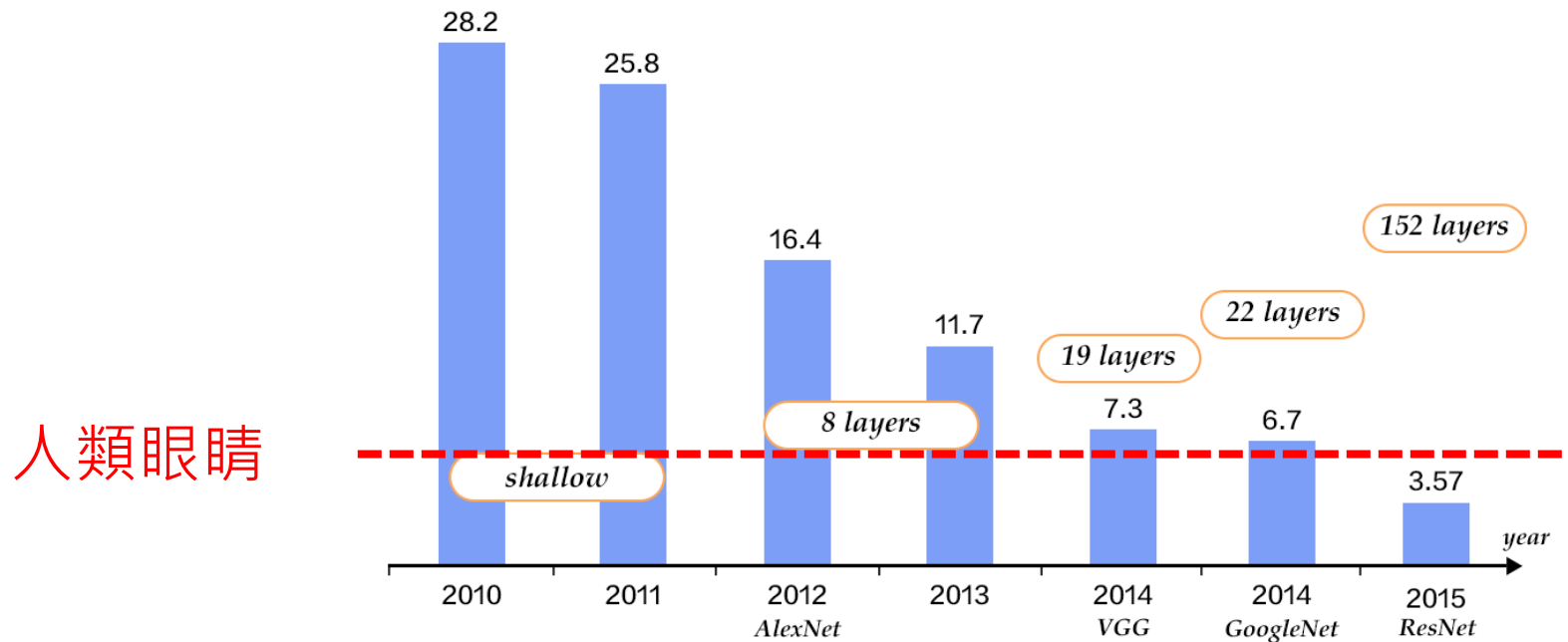
- ILSVRC比賽所使用的資料集叫做ImageNet
 - 其收錄了1500萬張照片以及2.2萬個類別
 - 此資料庫之蒐集是由著名教授李飛飛所主導的



source: <https://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf>

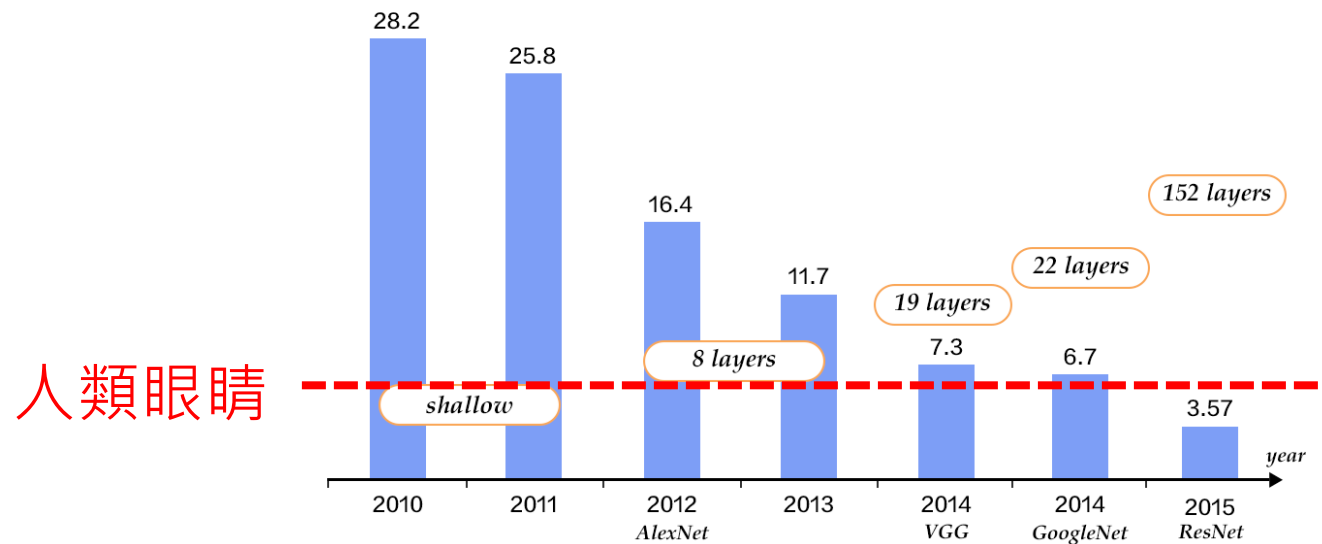
ILSVRC 歷年成績

- ILSVRC 比賽歷年冠軍、亞軍成績如下：
 - 從2012年起，深度學習的方法開始瘋狂拿下第一名，使得大家開始注意到深度學習這個領域
 - 而這幾年來網路也越疊越深，整體來說錯誤率也降低不少



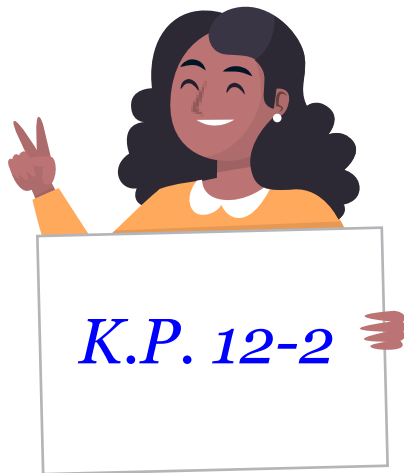
ILSVRC 歷年成績

- 可以注意到2015年ResNet錯誤率已經低於5%(比人眼低了)
 - 5%是人類眼睛在這個影像分類資料集上的錯誤率
- 網路也從8層一直增加到152層
 - 過程中產生不少著名網路，如AlexNet, VGG, GoogleNet, ResNet



12-2: AlexNet與VGG

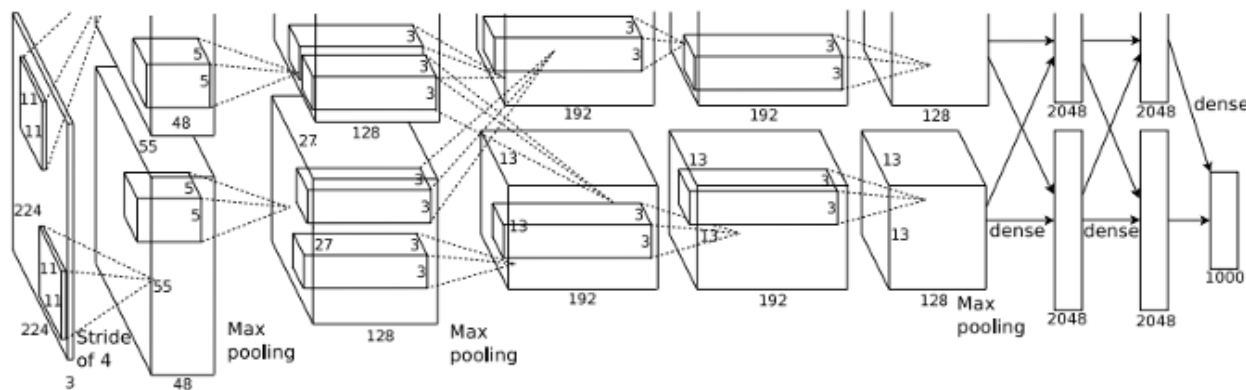
- AlexNet介紹
- VGG介紹



designed by freepik

AlexNet介紹

- AlexNet是2012年ILSVRC比賽的冠軍
 - 是非常著名的CNN網路之一
 - 他將神經網路疊了8層(如右下圖)



params	AlexNet	FLOPs
4M	FC 1000	4M
16M	FC 4096 / ReLU	16M
37M	FC 4096 / ReLU	37M
	Max Pool 3x3s2	
442K	Conv 3x3s1, 256 / ReLU	74M
1.3M	Conv 3x3s1, 384 / ReLU	112M
884K	Conv 3x3s1, 384 / ReLU	149M
	Max Pool 3x3s2	
	Local Response Norm	
307K	Conv 5x5s1, 256 / ReLU	223M
	Max Pool 3x3s2	
	Local Response Norm	
35K	Conv 11x11s4, 96 / ReLU	105M

source: <https://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf>

AlexNet介紹

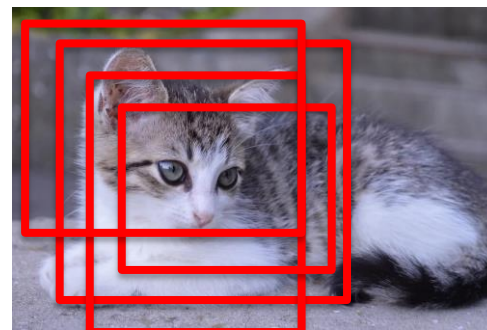
- 除了將神經網路疊了8層，AlexNet也提出不少對訓練網路有用的技巧
 - 包含資料增強、建議使用relu來當激活函數、建議使用maxpooling而非meanpooling、使用dropout技巧

AlexNet 介紹

- 資料增強表示將現有的訓練集資料做簡單的圖像操作來產生更多訓練資料
 - 水平旋轉、隨機裁切等
 - 這個技巧被證實出可以提升網路準確度



flip



random crops/scales

AlexNet 總結

- AlexNet提出了
 - 使用relu激活函數而非sigmoid
 - 使用dropout技巧提升網路準確度
 - 使用maxpooling而非meanpooling
 - 使用資料增強的技巧

VGG

- VGG是2014年ILSVRC比賽的亞軍
 - VGG有分為VGG16以及VGG19，分別代表疊了16層以及19層

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 × 224 RGB image)					
conv3-64	conv3-64 LRN	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 conv3-128	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 conv1-256	conv3-256 conv3-256 conv3-256	conv3-256 conv3-256 conv3-256 conv3-256
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 conv1-512	conv3-512 conv3-512 conv3-512	conv3-512 conv3-512 conv3-512 conv3-512
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 conv1-512	conv3-512 conv3-512 conv3-512	conv3-512 conv3-512 conv3-512 conv3-512
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

VGG16以及VGG19

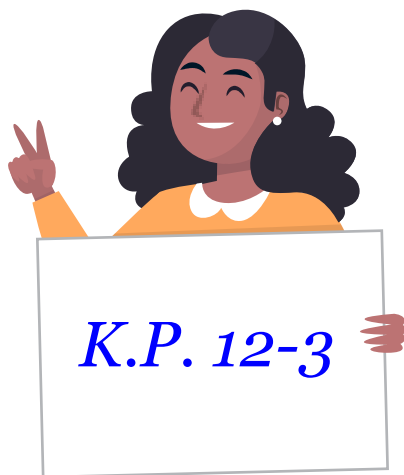
Source: <https://arxiv.org/pdf/1409.1556.pdf>

VGG 總結

- 神經網路越深準度越高
- 1×1 的卷積運算是有用的，但是建議將其設大一點

12-3: GoogleNet與ResNet

- GoogleNet介紹
- ResNet介紹



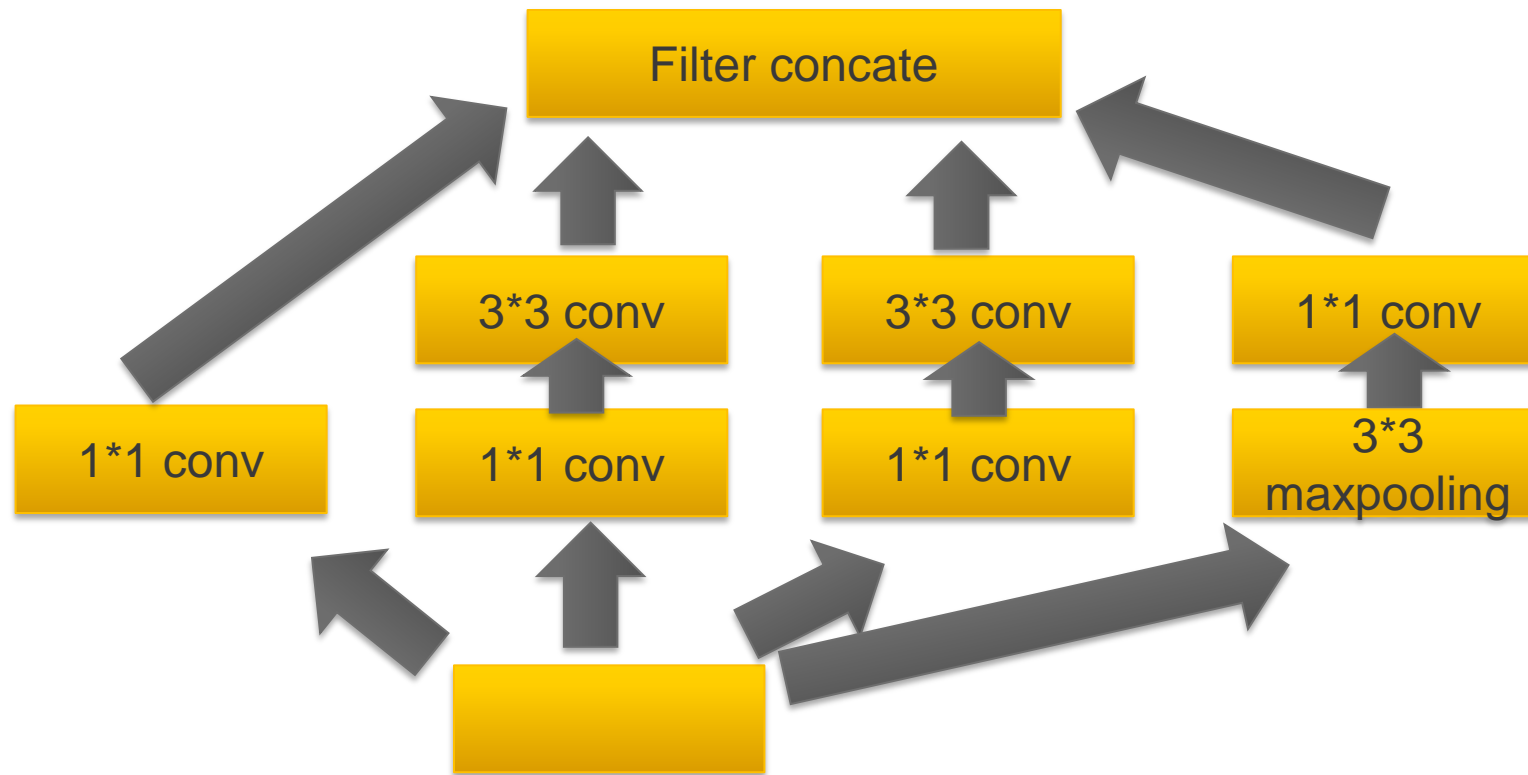
designed by freepik

GoogleNet介紹

- **GoogleNet是2014年ILSVRC比賽的冠軍**
 - 其將神經網路疊了22層
 - 網路內使用了自己定義的inception模組

GoogleNet介紹

- Inception模組架構如下，它會將輸入照片分成四個分支，分別做不同卷積及池化之運算



Inception模組

GoogleNet介紹

- GoogleNet整體網路架構如下：

Conv	Size=3*3 /stride= 2	299*299*3
Conv	Size=3*3 /stride= 1	149*149*32
Conv	Size=3*3 /stride= 1	147*147*32
Pooling	Size=3*3 /stride= 2	147*147*64
Conv	Size=3*3 /stride= 1	73*73*64
Conv	Size=3*3 /stride= 2	71*71*80
Conv	Size=3*3 /stride= 1	35*35*192
Inception module	3*inception module	35*35*288
Inception module	5*inception module	17*17*768
Inception module	3*inception module	8*8*1280
Pooling	8*8	8*8*2048
Linear	Logits	1*1*2048
Softmax	Classification	1*1*1000

GoogleNet 總結

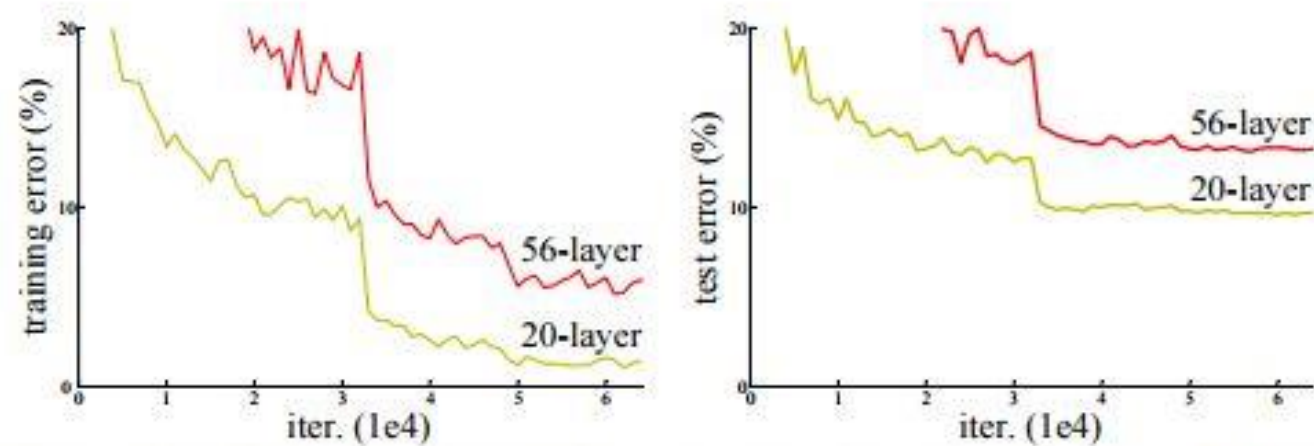
- 使用了自己定義的inception模組
- 比AlexNet所使用參數更少
 - 移除掉了CNN網路最後的全連結層

ResNet介紹

- **ResNet是2015年ILSVRC比賽的冠軍**
 - 總共疊了152層，錯誤率為3.57%
 - 為了將網路疊到超級深，其在網路加了捷徑的技巧

ResNet介紹

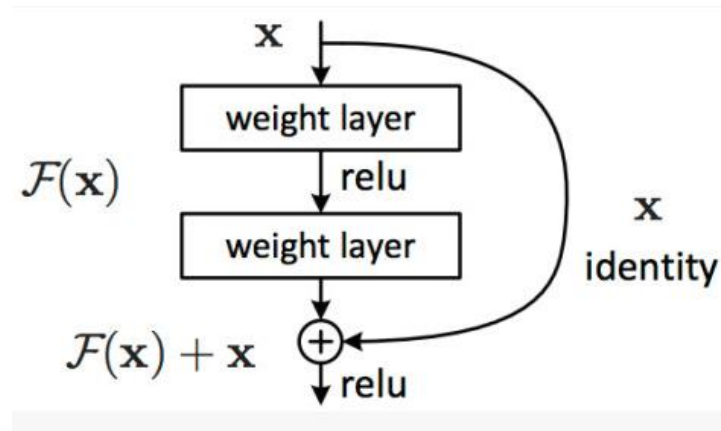
- ResNet一開始在將網路疊非常深的時候，發現如果網路疊超深到一定的程度，可能準確度會開始下降
 - 此問題稱為Degradation Problem



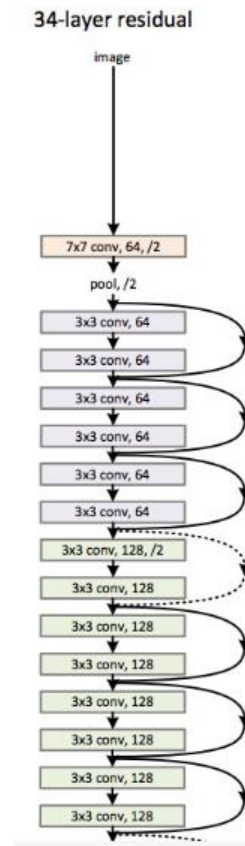
Source: <https://arxiv.org/pdf/1512.03385.pdf>

ResNet介紹

- ResNet為了解決Degradation Problem，將網路模組設計成左下圖，並重複這樣的結構才讓超深網路得以實現



Source: <https://arxiv.org/pdf/1512.03385.pdf>

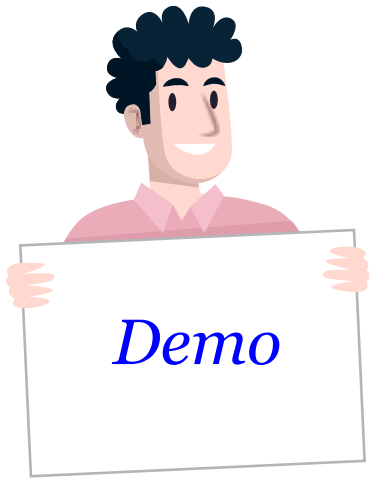


ResNet總結

- 在ILSVRC裡，讓錯誤率下降到比人眼錯誤率還低
- 解決Degradation Problem
 - 讓超深的神經網路得以實現

Demo 12-3

- 資料增強—隨機裁切
- 資料增強—翻面
- 資料增強—調整亮度



designed by freepik

線上Corelab

- 題目1：隨機裁切
 - 請自行準備一張照片，並以此張照片做10次隨機裁切生成多張照片
- 題目2：隨機翻轉
 - 請自行準備一張照片，並以此張照片做10次隨機上下翻轉，觀察相片翻轉的機率
- 題目3：資料增強之整合應用
 - 請自行準備一張照片，做隨機裁切後再做隨機左右翻轉，並將所有輸出照片的亮度調整0.4
 - 總共輸出20張處理過後的照片
 - 隨機裁切大小請設定為原圖的10%

本章重點精華回顧

- ILSVRC比賽
- ILSVRC歷年成績
- AlexNet與VGG介紹
- GoogleNet與ResNet介紹



Lab:Python 簡介

- **Lab01:資料增強—隨機裁切**
- **Lab02:資料增強—翻面**
- **Lab03:資料增強—調整亮度**

Estimated time:

20 minutes

