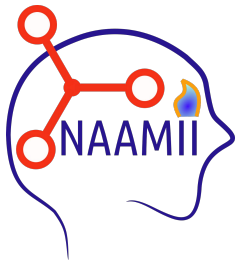
Computer Vision and Deep Learning

Python Ecosystem

Bishesh Khanal, PhD
Research Scientist, NAAMII

Senior AI Scientist, Fusemachines
Visiting Researcher, King College London & Imperial College London

Python Users Group Nepal, Meetup #14
Kathmandu, Nepal
14 September, 2019



https://www.naamii.com.np

NAAMII

https://www.naamii.com.np

Research

Nepal Winter School in AI

Academic Social Outreach

fuse | machines

leapfrog

Startup Incubation & Industry Collaboration

NAAMII

https://www.naamii.com.np

Research

Application deadline coming soon!
https://nepalschool.naamii.com.np

Nepal Winter School in AI

fuse | machines

leapfrog

Academic
Social
Outreach

Startup
Incubation
&
Industry
Collaborati
on

Computer Vision

Deep Learning

Python Ecosystem for Machine Learning

Computer Vision: 3D Modeling

- Automatically compute 3D model from 2D images

Bishesh Khanal and Ajad Chhatkuli. First Nepal Winter School in AI. 2018

Computer Vision: 3D Modeling

- Automatically compute 3D model from images

3D Model of a part of Zurich modeled from images – source VarCity, CVL –ETH Zurich

Computer Vision: Traffic Understanding

Computer Vision: Disaster Area/Damage Identification

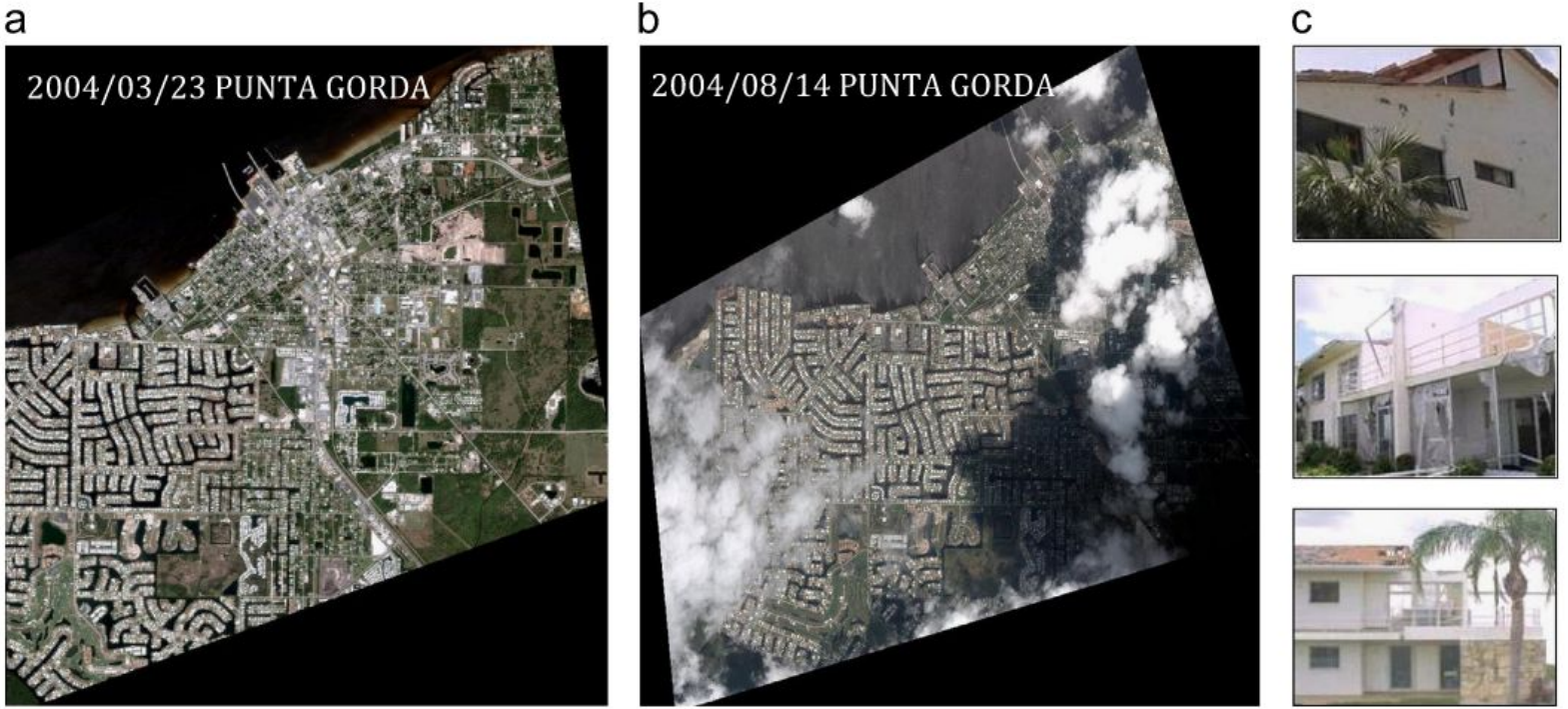


Fig. 1. (a) Pre-storm image (source: DigitalGlobe Co., Ltd.). (b) Post-storm Image (source: DigitalGlobe Co., Ltd.). (c). Field investigation information (source: Womble, 2005)

Radhika et al. Cyclone damage detection ... JWEIA 2015.

Computer Vision: Disaster Area/Damage Identification

Fig. 2. (a) A sample building before cyclone damage. (b) After cyclone damage. (c) Non damaged portion marked in red outline. (d) Ground truth data of the same building (source: Womble, 2005). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Radhika et al. Cyclone damage detection ... JWEIA 2015.

remote sensing

MDPI

Article

Detection of Urban Damage Using Remote Sensing and Machine Learning Algorithms: Revisiting the 2010 Haiti Earthquake

Austin J. Cooner *, Yang Shao and James B. Campbell

Virginia Tech Department of Geography, 115 Major Williams Hall 220 Stanger St., Blacksburg, VA 24060, USA; yshao@vt.edu (Y.S.); jayhawk@vt.edu (J.B.C.)

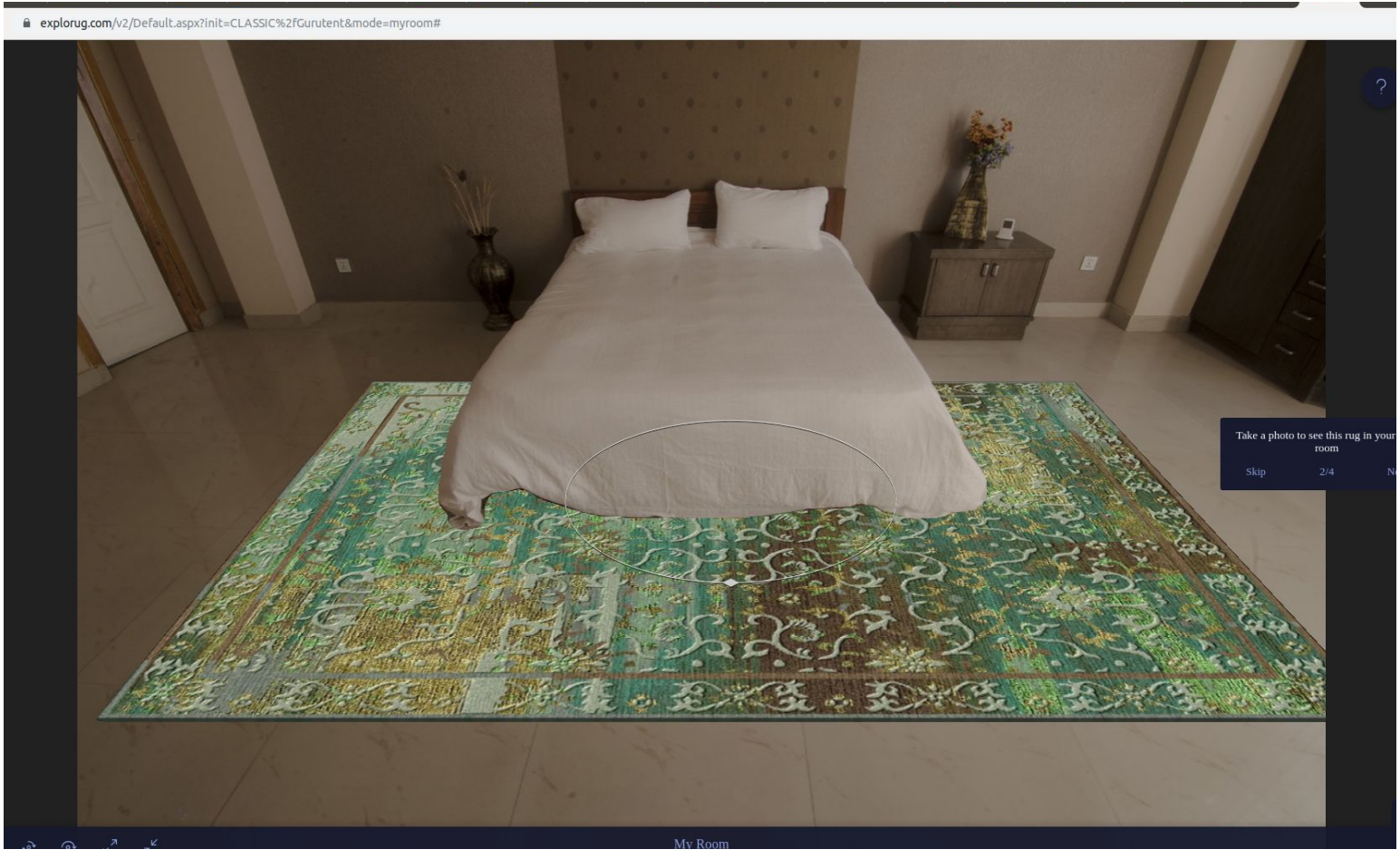
* Correspondence: cooner@vt.edu; Tel.: +1-210-870-7909

Academic Editors: Roberto Tomas, Zhenhong Li and Prasad S. Thenkabail

Received: 5 August 2016; Accepted: 17 October 2016; Published: 20 October 2016

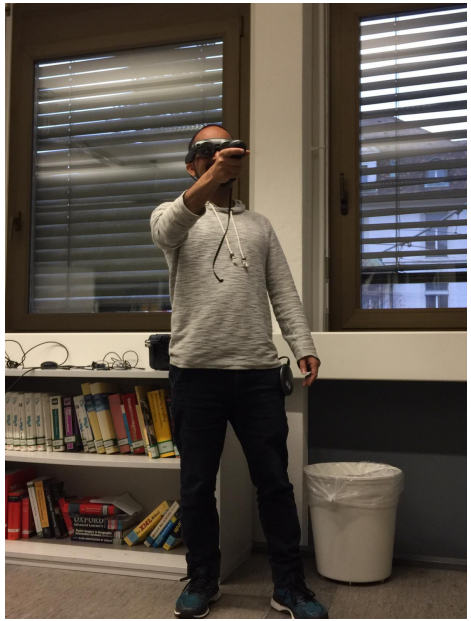
Abstract: Remote sensing continues to be an invaluable tool in earthquake damage assessments and emergency response. This study evaluates the effectiveness of multilayer feedforward neural networks, radial basis neural networks, and Random Forests in detecting earthquake damage caused by the 2010 Port-au-Prince, Haiti 7.0 moment magnitude (M_w) event. Additionally, textural and structural features including entropy, dissimilarity, Laplacian of Gaussian, and rectangular fit

Computer Vision: Augmented Reality

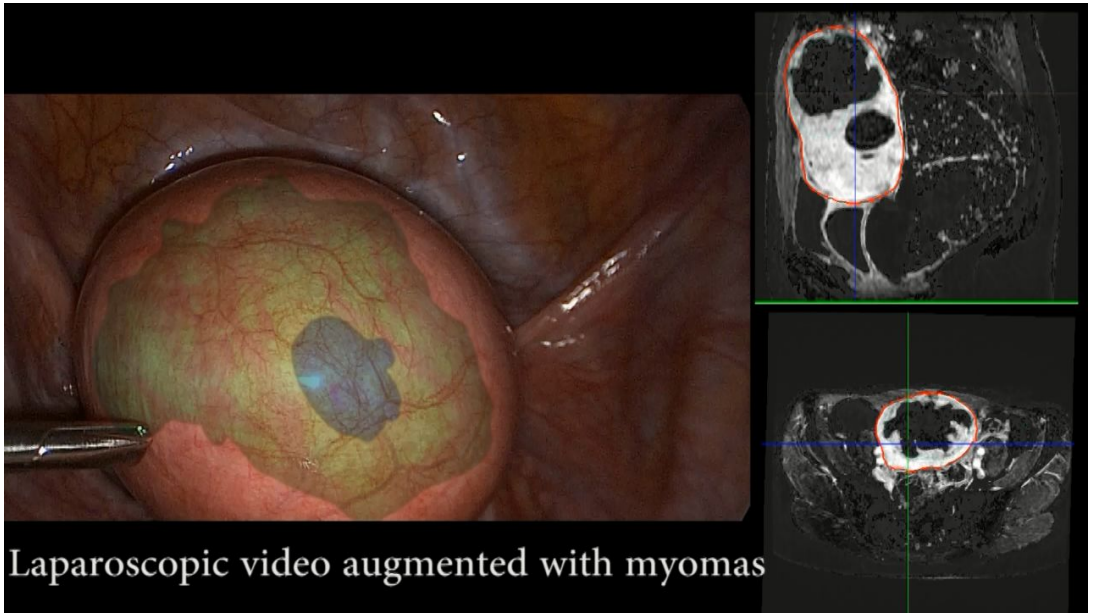


Computer Vision: Virtual/Augmented/Mixed Reality

- Mixed Reality and Virtual Reality

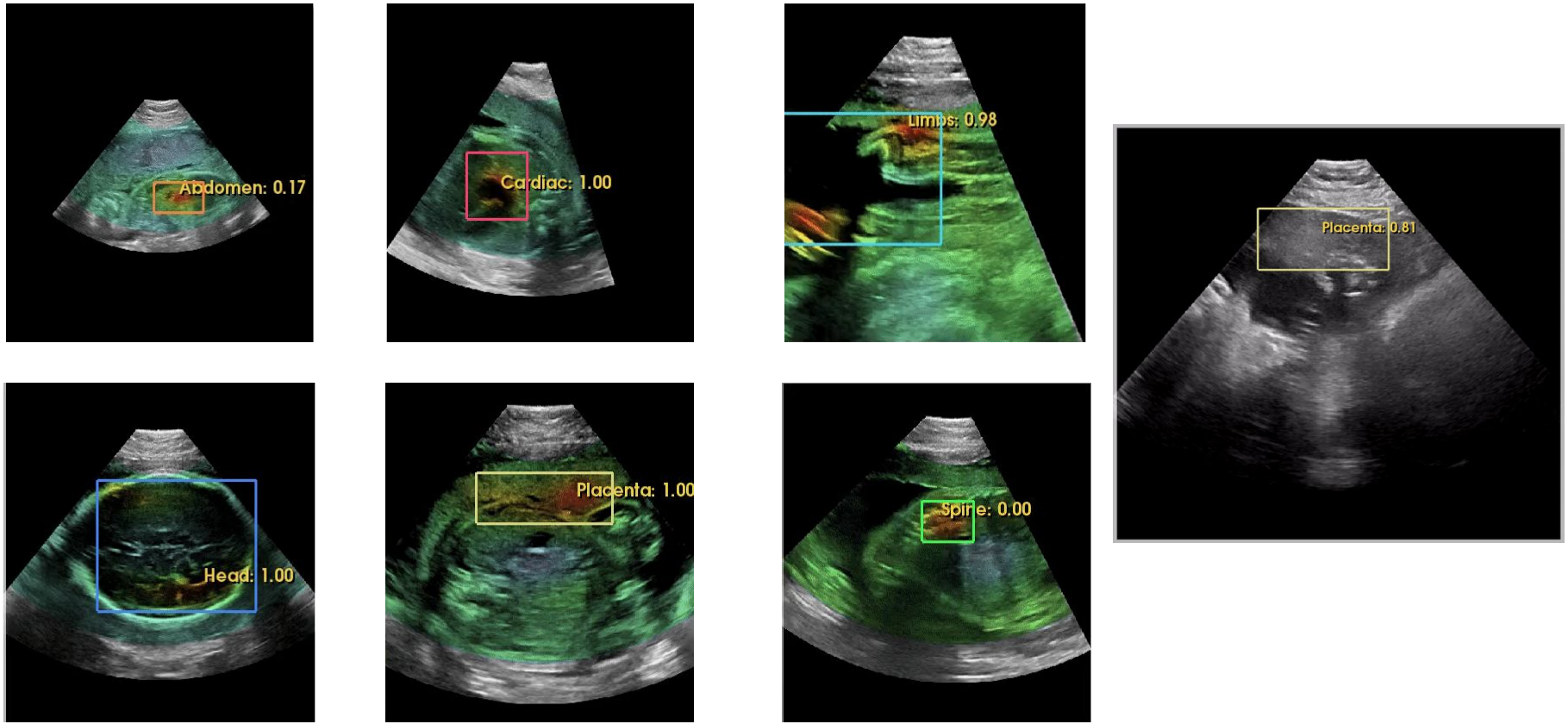


MagicLeap demo



ENCoV - France

Computer Vision: Medical Imaging



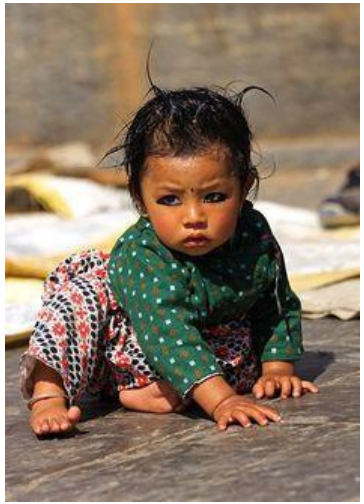
N. Toussaint, **B. Khanal,** *et al*. Weakly Supervised Localisation for Fetal Ultrasound Images. *MICCAI Workshop DLMIA 2018*.

How do we see ?

How do we LEARN to see ?

How do we learn to see ?

We move in order to see and see in order to move
William Gibson

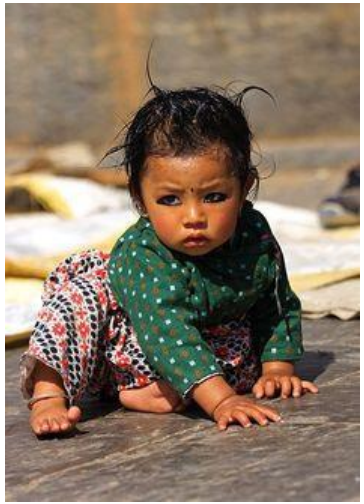


source: Traveltourtrek Nepal

We live in a 3D world

How do we learn to see ?

We move in order to see and see in order to move
William Gibson



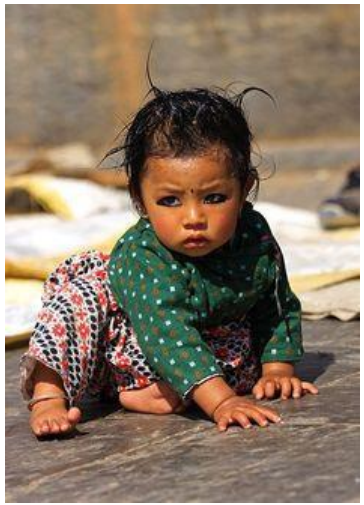
source: Traveltourtrek Nepal

We live in a 3D world

Three pillars of AI - data, learning and knowledge
-Fei Fei Li

How do we learn to see ?

We move in order to see and see in order to move
William Gibson



source: Traveltourtrek Nepal

We live in a 3D world

Three pillars of AI - data, learning and knowledge
-Fei Fei Li

Not just about the eye, a mere sensor.
Human Vision is done in the Brain!

From 3D world to 2D images

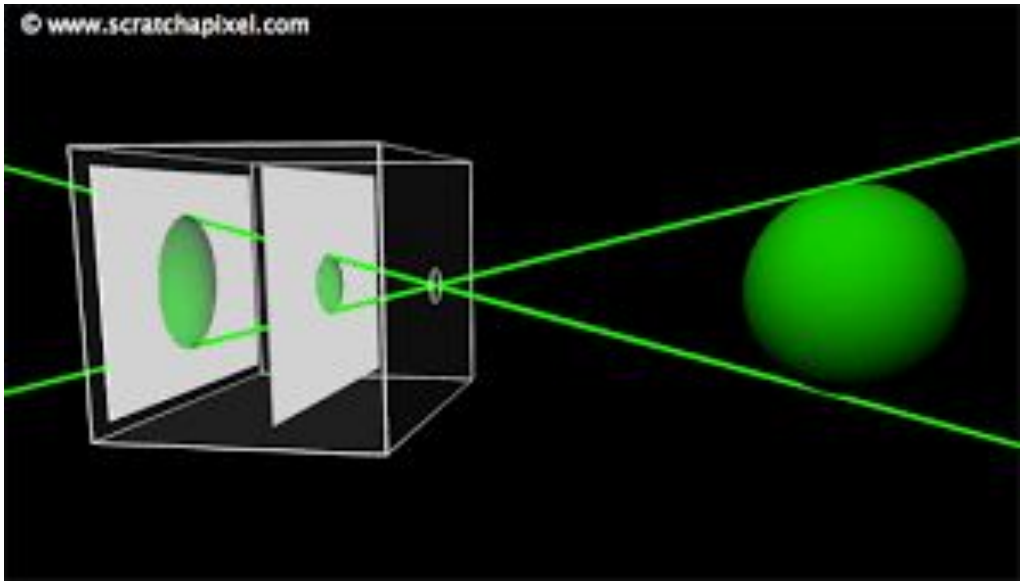
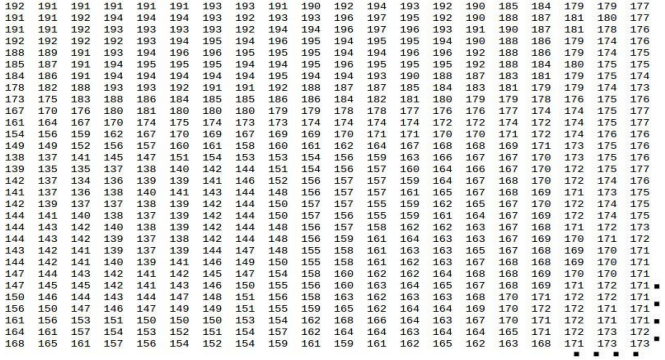
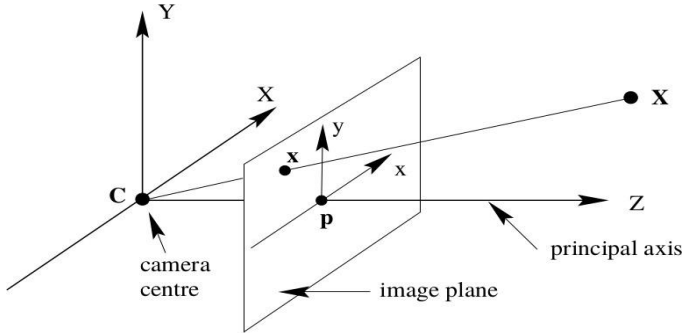


Image Formation

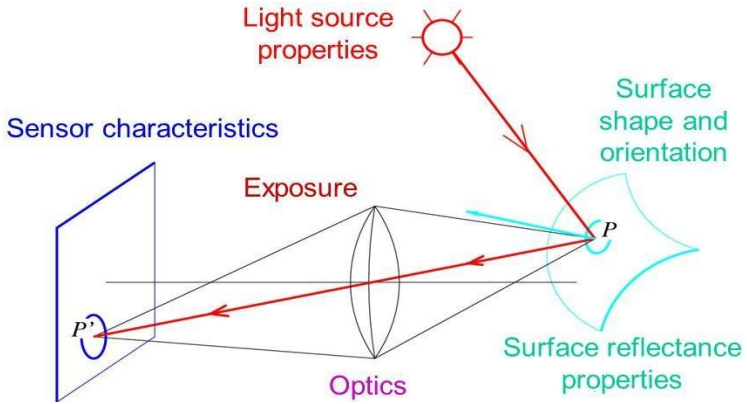
What we see



What machines see



Source: Hartley, Richard, and Andrew Zisserman. Multiple view geometry in computer vision. Cambridge university press, 2003.



Source: L. Fei Fei

What do we see ?

What do we see ?

Brain known to process at different levels.
Marr’s 3 levels:

- Primal sketch : edges, corners, regions : “features”
- 2.5 D sketch : texture, depth concept
- 3D model : the scene as we see it in a 3D world

Hand Engineering to Extract Features

Traditional approach (very popular: 1990's - 2010)

- Extract features designed by engineers/researchers
- Use them to learn higher level semantics

Hand-Engineered Feature Extraction

Hand-Engineered Feature Extraction: Edge Detection

Hand-Engineered Feature Extraction: Threshold

Hand-Engineered Feature Extraction: Harris Corner

Hand-Engineered Feature Extraction: Harris Corner

Many more!

SIFT

SURF

ORB

LBP

Convolutional Filters

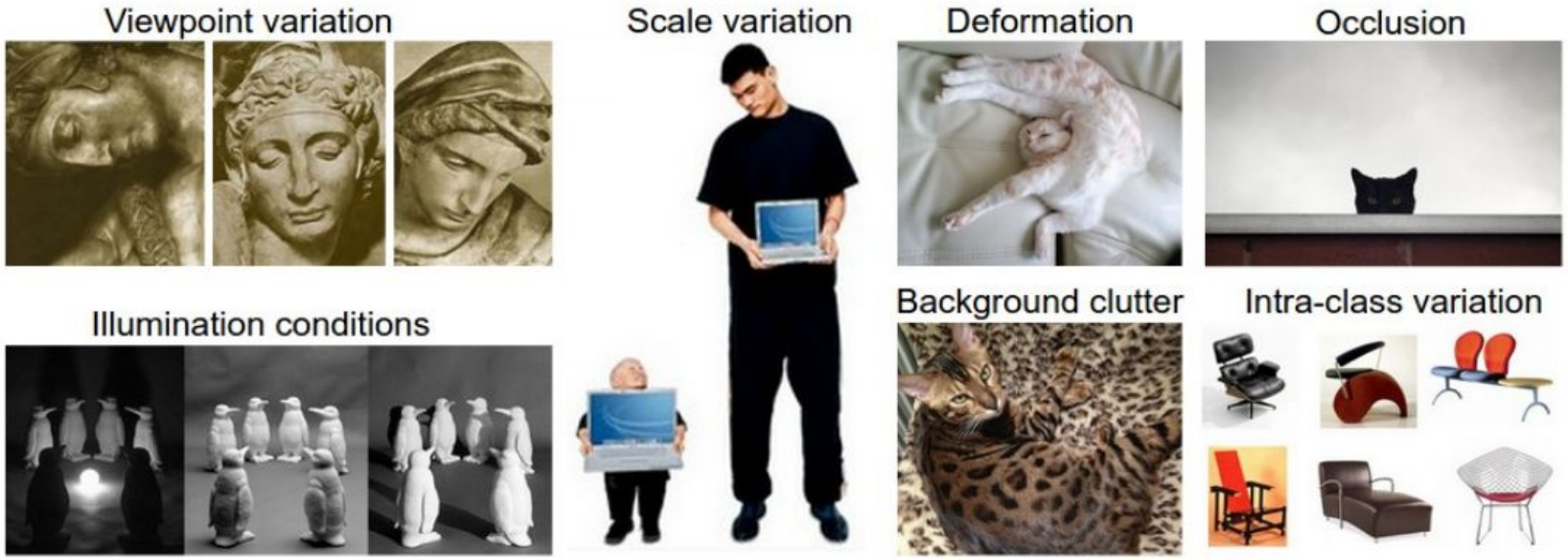
…

Easy to understand

No big data need

Fast if designed well

It is not an easy task!



Source: http://cs231n.github.io/classification

Features depend on imaging conditions and hard to design “invariant” features

Relatively strong domain knowledge required

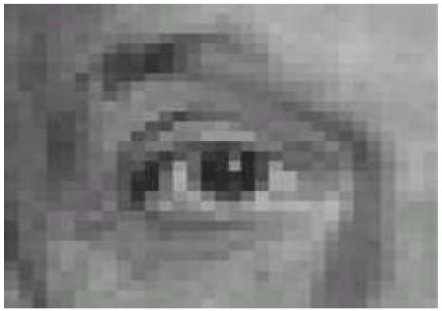
Computer Vision

Deep Learning

Python Ecosystem for Machine Learning

Learn from the data needed features & representation

Building block of feature extraction: convolution/filter/kernel

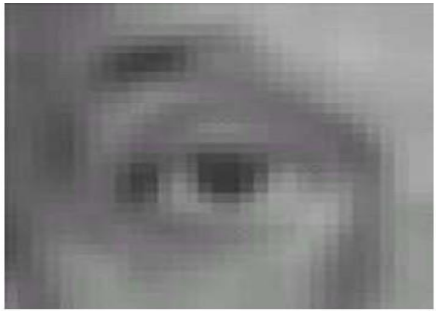


Original

$\frac{1}{9}$

1	1	1
1	1	1
1	1	1

Filter

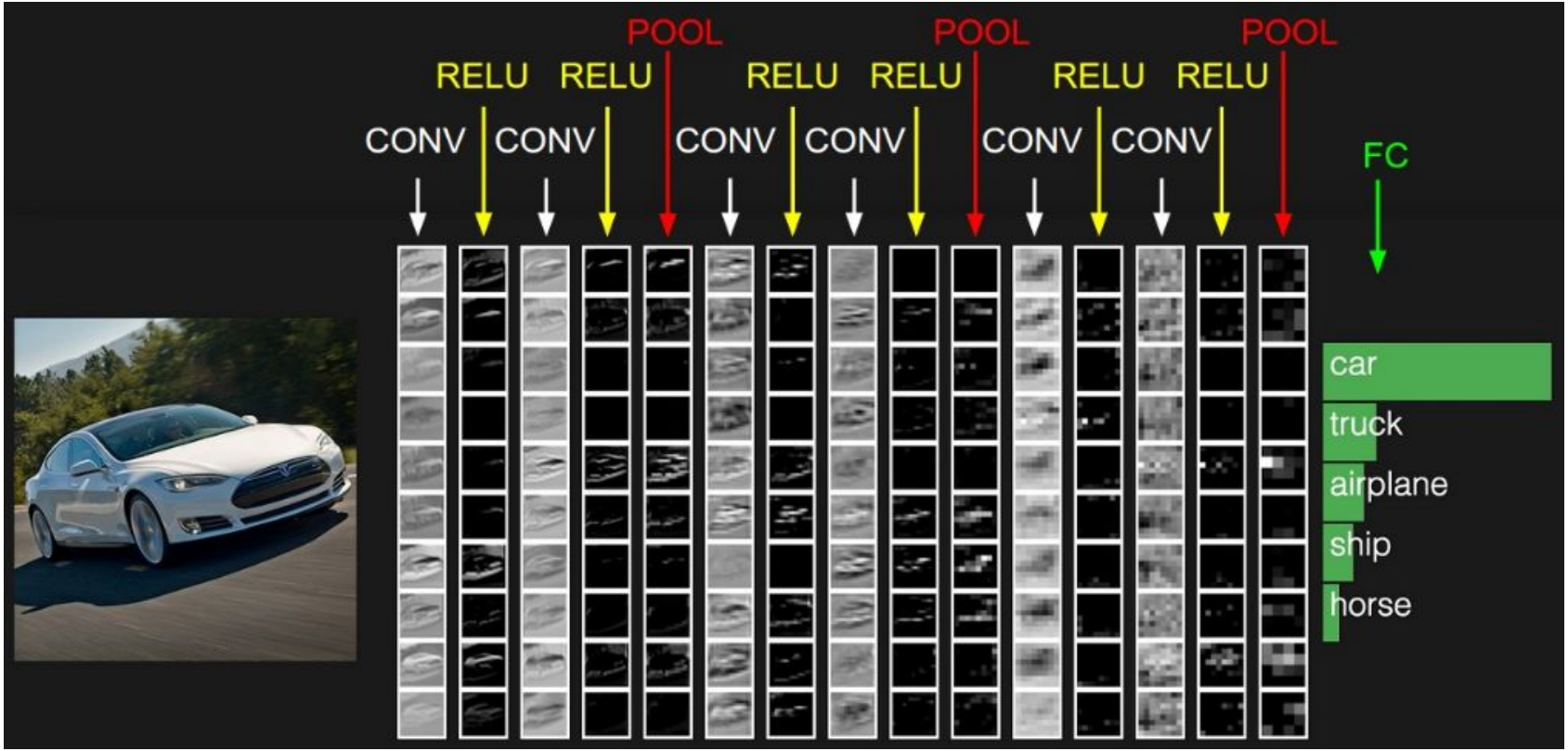


Result

(blur with a box filter)

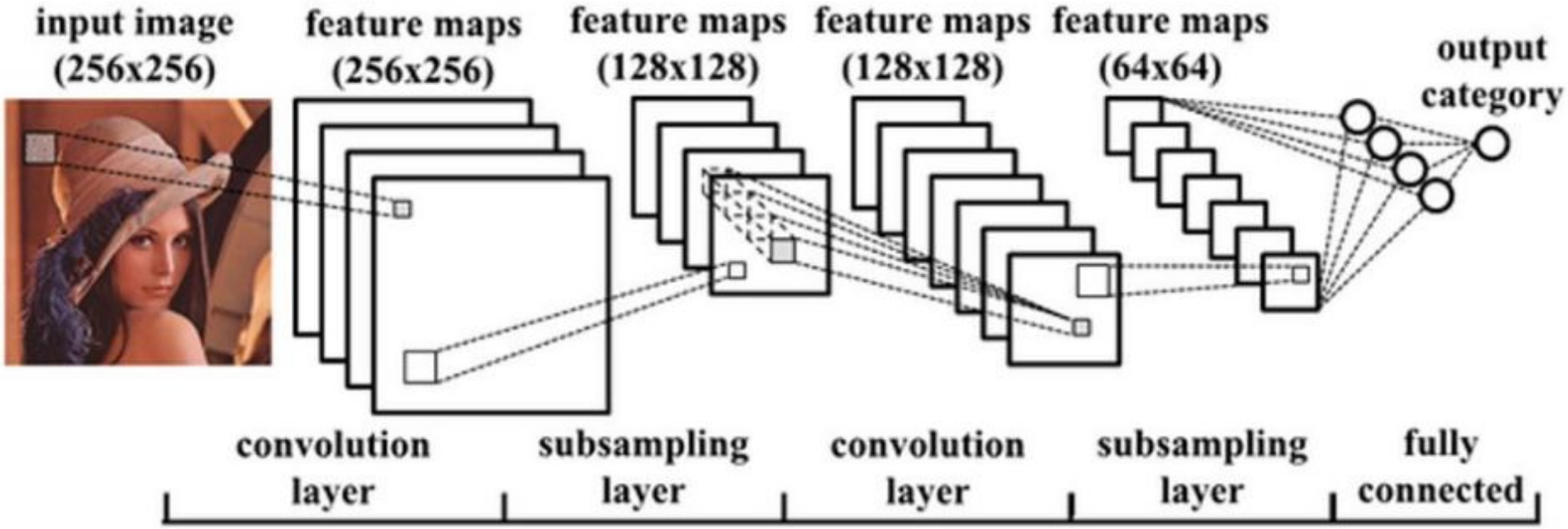
Learn these filters/weights from lot of data on a given task as required

Deep Convolutional Neural Networks most popular



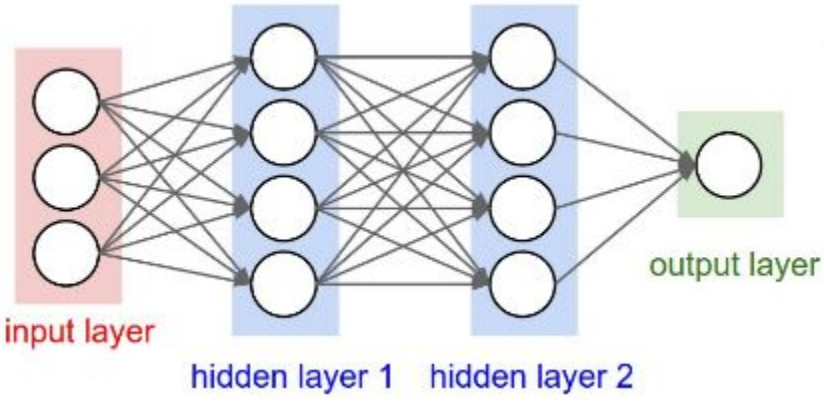
http://cs231n.github.io/convolutional-networks/

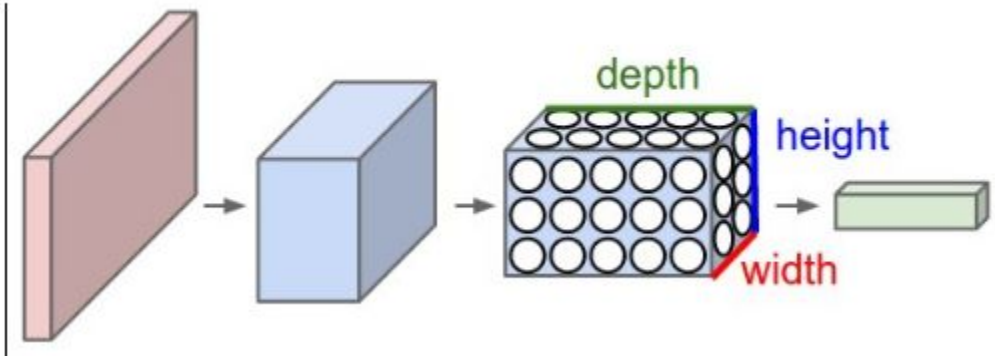
Deep Convolutional Neural Networks most popular



https://community.arm.com/developer/ip-products/processors/b/processors-ip-blog/posts/deploying-convolutional-neural-network-on-cortex-m-with-cmsis-nn.

CNNs





Regular 3 layer network

Convnet arrangement in 3D

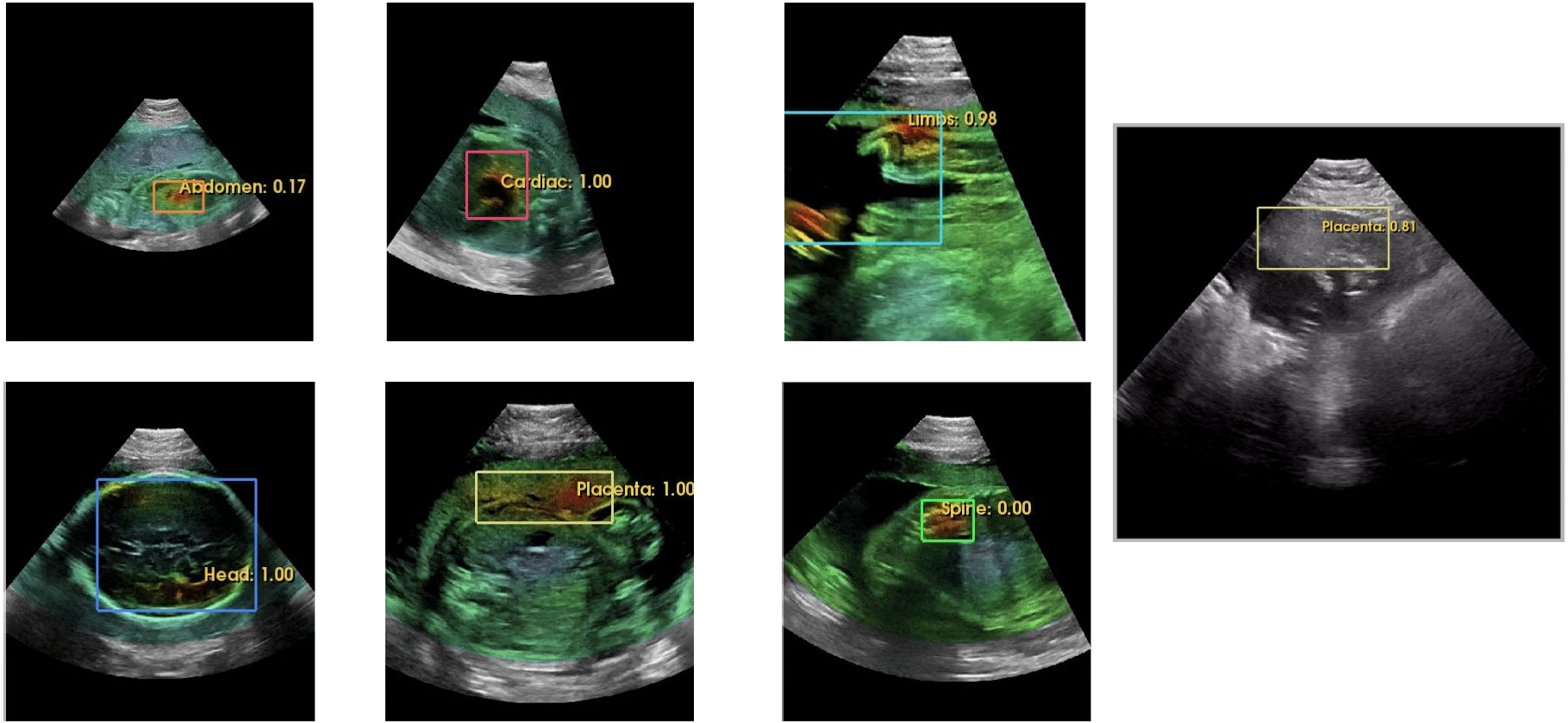
Transforms input 3D volume into output 3D volume

E.g. Input of 3 Channels R,G,B; image of size height X width

http://cs231n.github.io/convolutional-networks/

Example Applications

Example Applications



N. Toussaint, **B. Khanal,** *et al*. Weakly Supervised Localisation for Fetal Ultrasound Images. *MICCAI Workshop DLMIA 2018*.

Are neural networks and CNNs new ?

NNs from 1950s, CNNs from 70s, but optimizing them was a challenge!!

How come they are so popular now ?

Why Deep Learning worked now?

10 years rapidly expanding use of Internet: BIG DATA availability

Massive improvement in Hardware (GPUs)

Advancement in optimization techniques

Great open-source software frameworks to run on those GPUS (mostly python based)

Why Deep Learning worked now?

10 years rapidly expanding use of Internet: BIG DATA availability

Massive improvement in Hardware (GPUs)

Advancement in optimization techniques

Great open-source software frameworks to run on those GPUS (mostly python based)

Not everything solved!!

Less data and learn, Unsupervised learning, best representation

Generalization

Interpretability

Computer Vision

Deep Learning

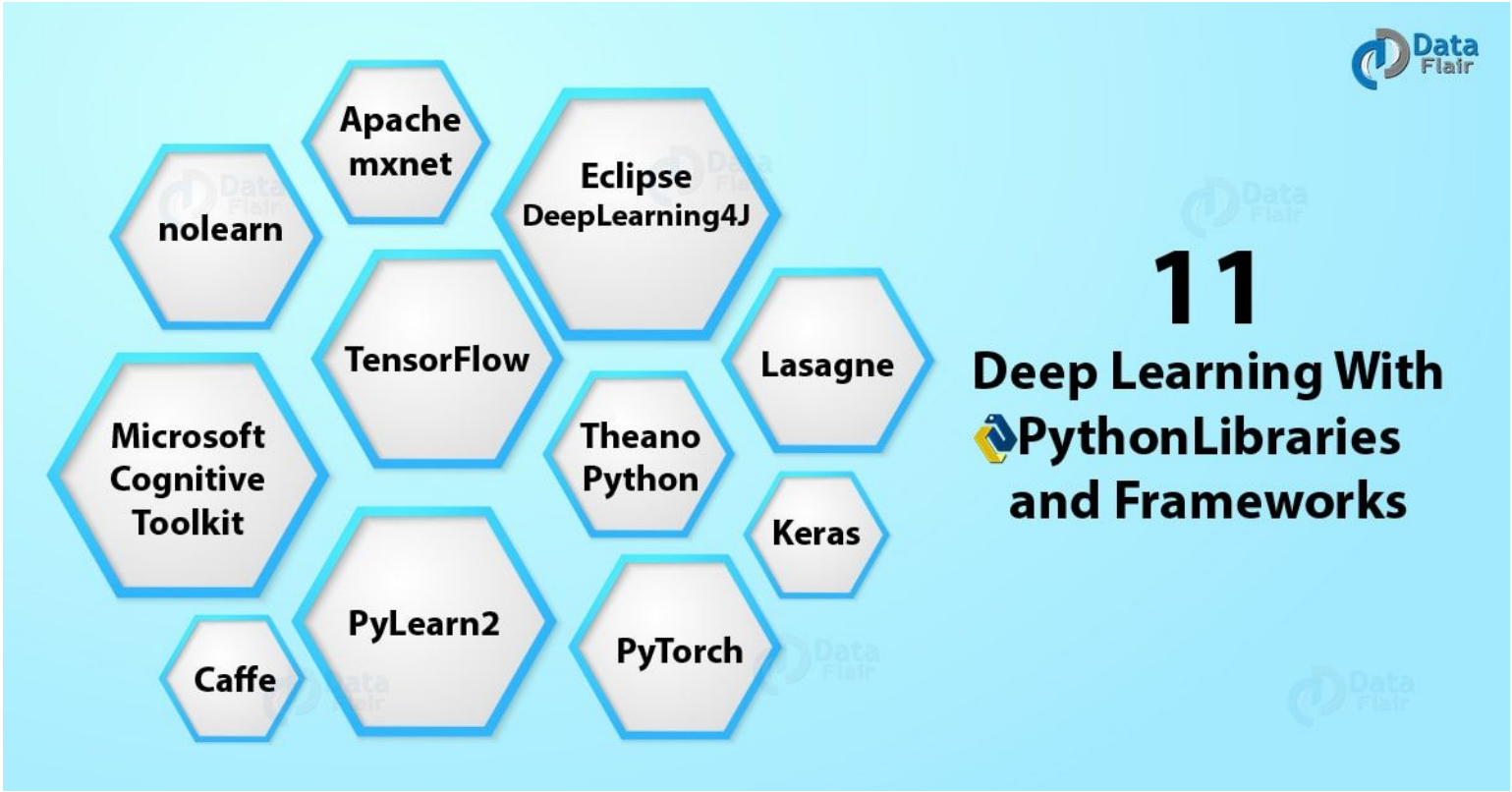
Python Ecosystem for Machine Learning

Python for Scientific Computing & Data Science

```python
import cv2
import numpy as np

img = cv2.imread('image1.jpg')
gray= cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)

gray = np.float32(gray)
dst = cv2.cornerHarris(gray,2,3,0.04)

#result is dilated for marking the corners, not important
dst = cv2.dilate(dst,None)

# Threshold for an optimal value,
# it may vary depending on the image.
#img[dst>0.01*dst.max()]=[0,0,255]

img[dst>0.001*dst.max()]=[0,0,255]

#cv2.imshow('dst',img)
#if cv2.waitKey(0) & 0xff == 27:
#    cv2.destroyAllWindows()

cv2.imwrite('harris-corners-image1.jpg',img)
```

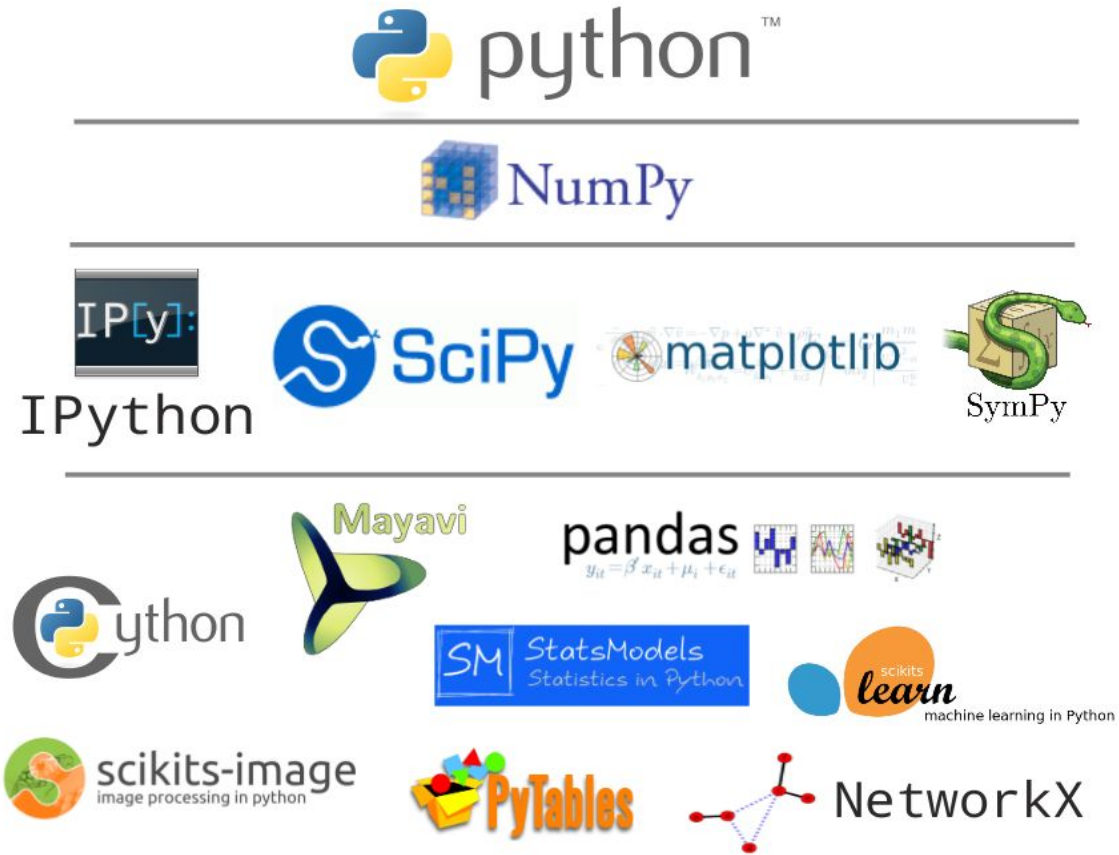
```python
from mmdet.apis import init_detector, inference_detector, show_result
import mmcv
config_file = 'mmdetection/configs/faster_rcnn_r50_fpn_1x.py'
checkpoint_file = '../mmdetection/checkpoints/faster_rcnn_r50_fpn_1x_20181010-3d1b3351.pth'
model = init_detector(config_file, checkpoint_file, device='cuda:0')
img = 'results/Kathmandu-Traffic.jpg'
result = inference_detector(model, img)
show_result(img, result, model.CLASSES, out_file='results/Kathmandu-Traffic-out.jpg')
```

Python for Deep Learning



https://data-flair.training/blogs/deep-learning-with-python-libraries/

Python for Scientific Computing & Data Science



http://www.focusedsupport.com/blog/getting-setup-with-scientific-python/

Internships and vacancies at NAAMII announcing soon

https://www.naamii.com.np/

https://twitter.com/naamii_nepal

https://www.facebook.com/naamiiNepal