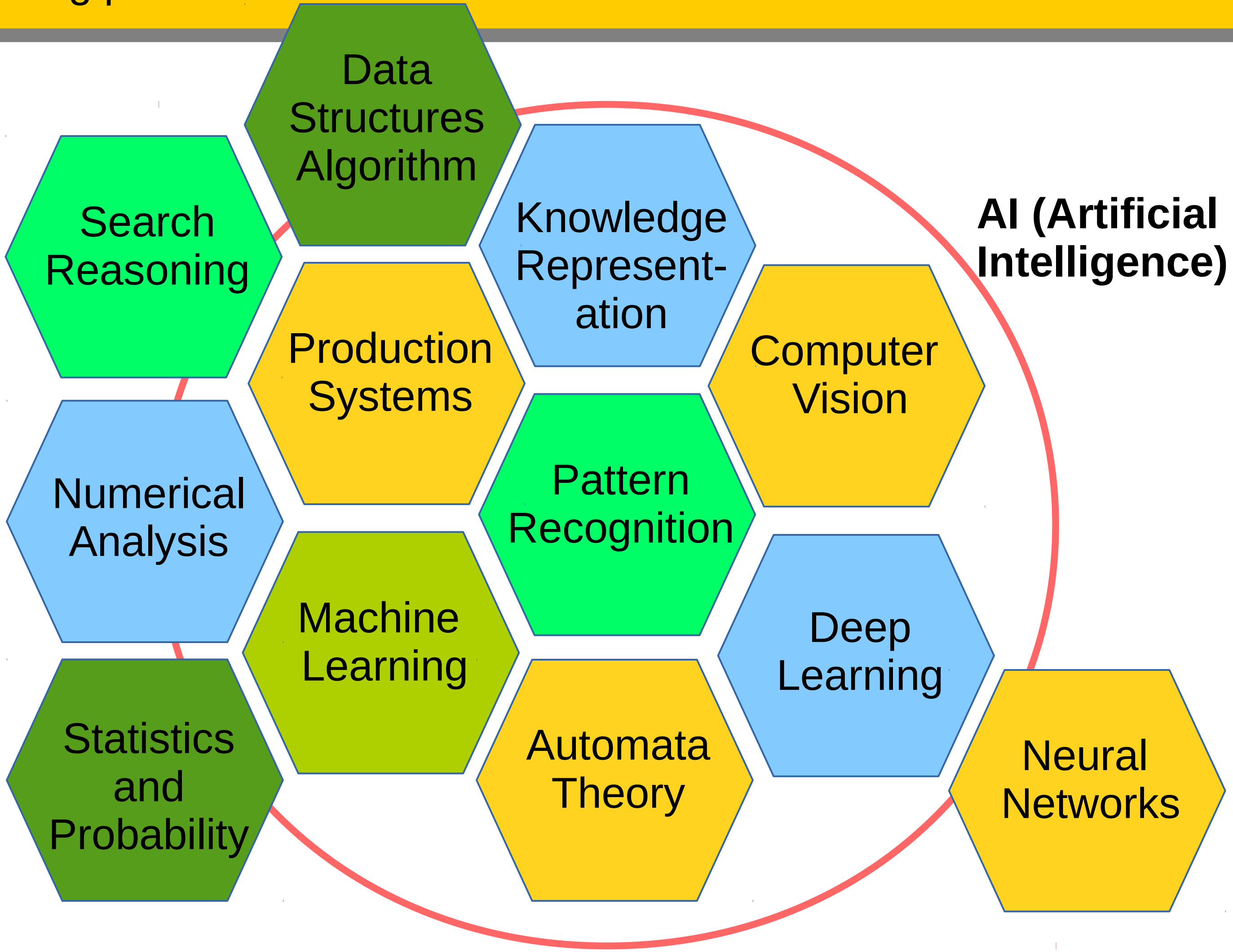
The Scope of AI

What is AI (Artificial Intelligence)? Technology which employs computer to build intelligence capability to mimic human decision making, to assist and release human from decision making process.



Search Reasoning

Numerical Analysis

Statistics and Probability

Data Structures Algorithm

Production Systems

Machine Learning

Knowledge Representation

Pattern Recognition

Automata Theory

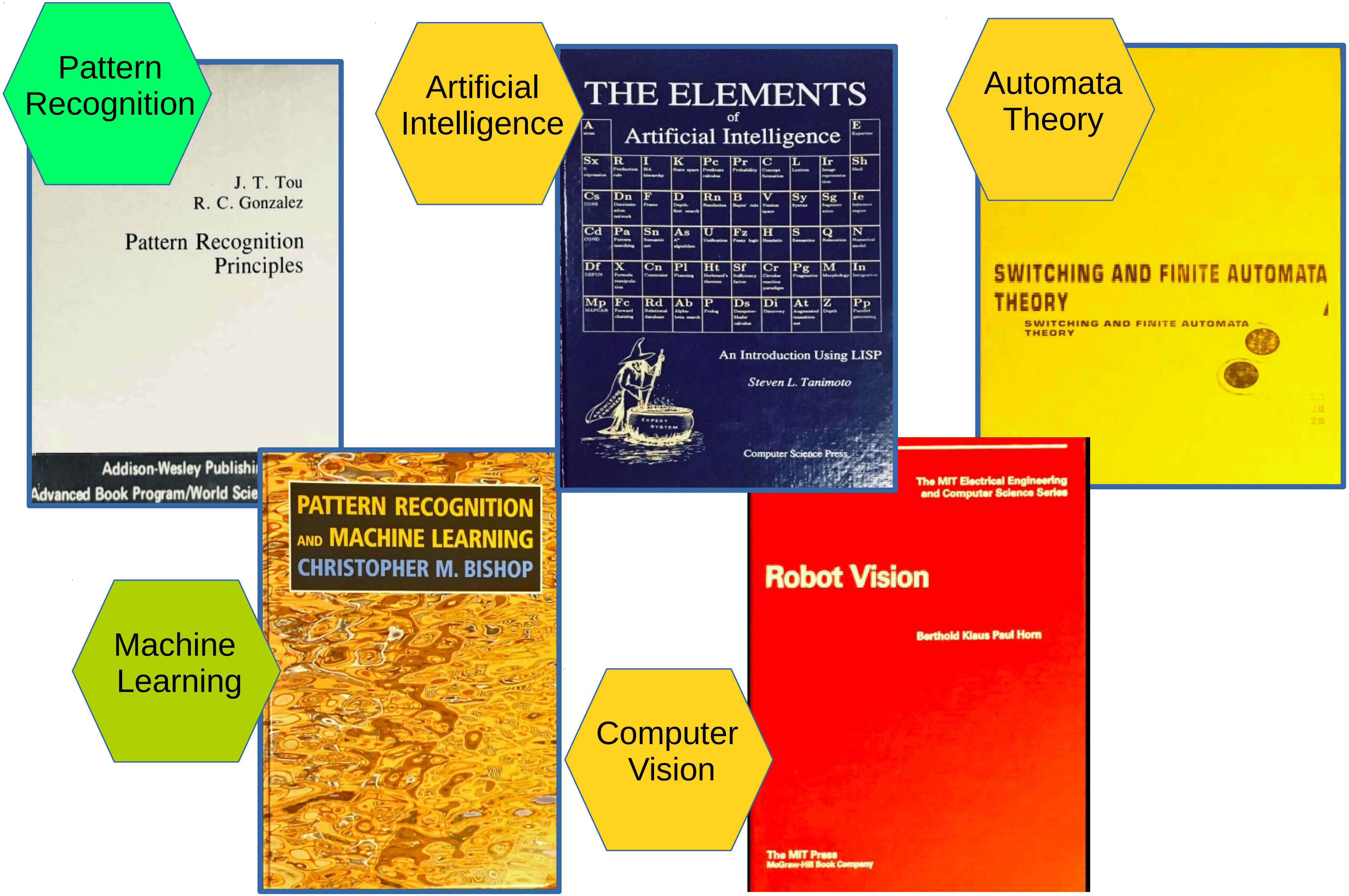
Computer Vision

Deep Learning

Neural Networks

AI (Artificial Intelligence)

Reference on AI and PR


J. T. Tou
R. C. Gonzalez
Pattern Recognition Principles
Addison-Wesley Publishi
Advanced Book Program/World Scie
THE ELEMENTS of Artificial Intelligence
An Introduction Using LISP
Steven L. Tanimoto
Computer Science Press
SWITCHING AND FINITE AUTOMATA THEORY
PATTERN RECOGNITION AND MACHINE LEARNING
CHRISTOPHER M. BISHOP
The MIT Electrical Engineering and Computer Science Series
Robot Vision
Berthold Klaus Paul Horn
The MIT Press
McGraw-Hill Book Company

Pattern Recognition

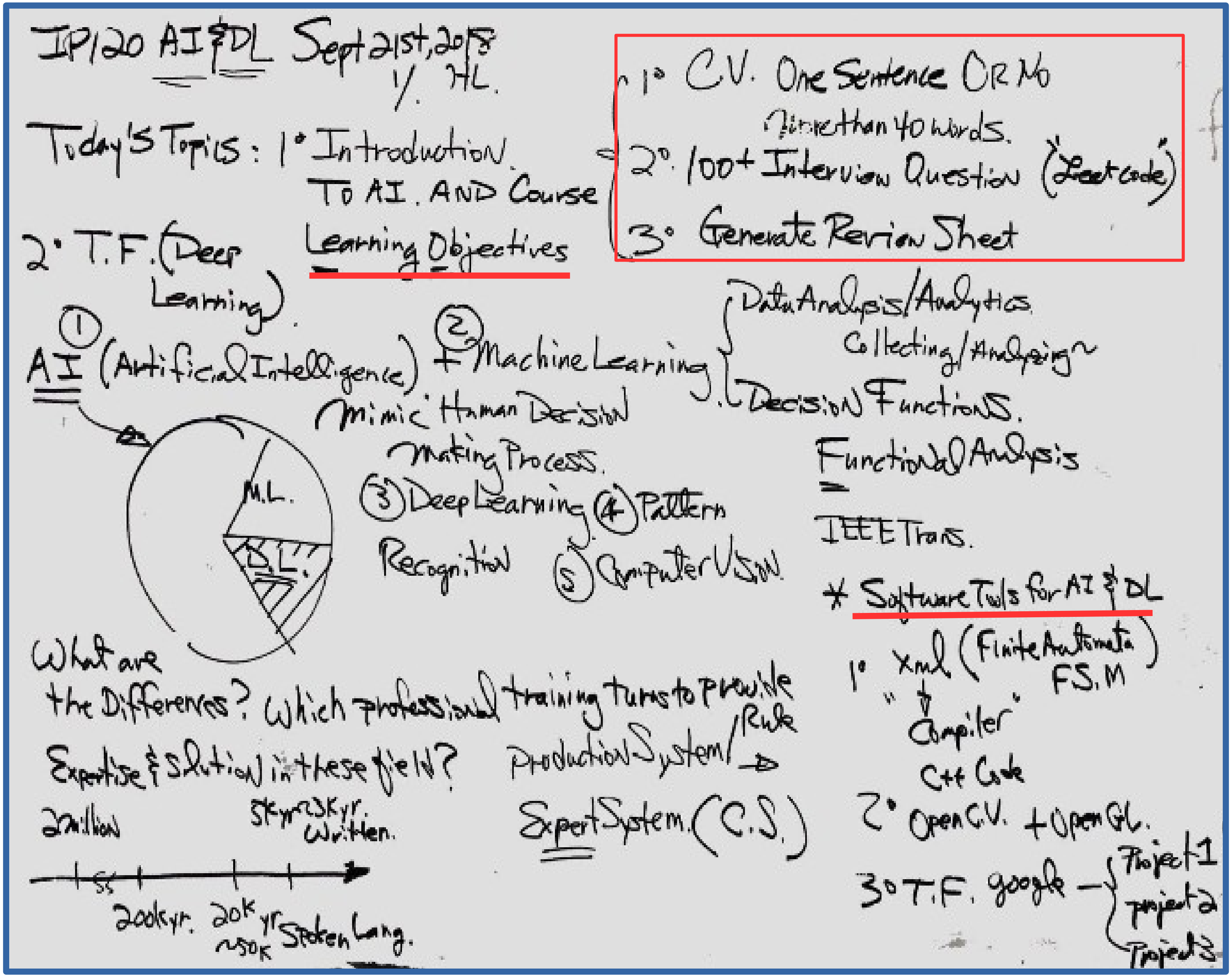
Artificial Intelligence

Automata Theory

Machine Learning

Computer Vision

9-21-2018 The Scope of AI



IP120 AI & DL Sept 21st, 2018
1/. HL.

Today's Topics : 1° Introduction To AI. AND Course Learning Objectives

1° C.V. One Sentence OR No More than 40 words.
2°. 100+ Interview Question ("Leetcode")
3° Generate Review Sheet

2° T.F. (Deep Learning).

① AI (Artificial Intelligence) ② Machine Learning
Mimic Human Decision Making Process.

Data Analysis/Analytics
Collecting/Analyzing
Decision Functions.

Functional Analysis

IEEE Trans.

③ Deep Learning ④ Pattern Recognition ⑤ Computer Vision

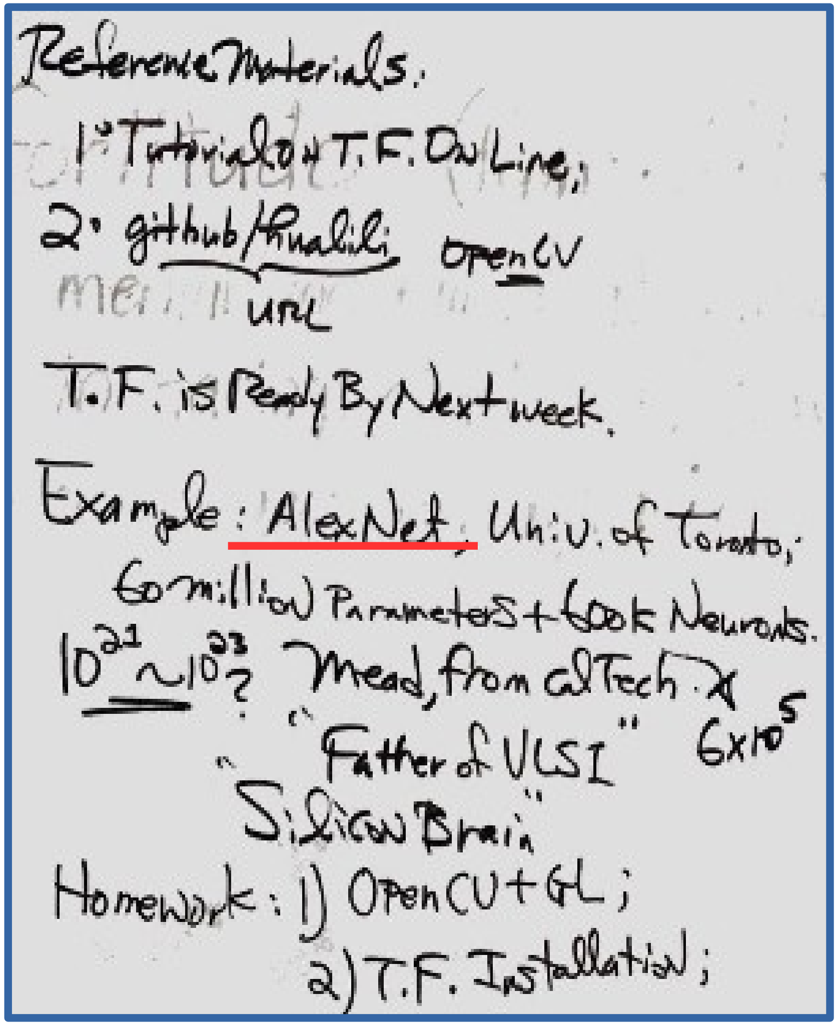
What are the Differences? Which professional training turns to provide Expertise & Solution in these field?

Production System / Rule →
Expert System (C.S.)

2 million
5k yr ~3k yr. Written.
200k yr. 20k yr ~50k Spoken Lang.

* Software Tools for AI & DL
1° Xml (Finite Automata FS.M)
"↓ Compiler"
C++ Code
2° OpenCV. + OpenGL.
3° T.F. google — { Project 1, project 2, Project 3

9-21-2018 Deep Learning and AlexNet Reference



Reference Materials:

1° Tutorial On T.F. On Line;

2° github/hualili OpenCV

url

T.F. is Ready By Next week.

Example: AlexNet, Univ. of Toronto;

60 million Parameters + 600k Neurons.

$10^{21} \sim 10^{23}$? Mead, from calTech. 6×10^5

"Father of VLSI"

"Silicon Brain"

Homework: 1) OpenCV+GL;

2) T.F. Installation;

9-21-2018 CLIPS For Expert Systems

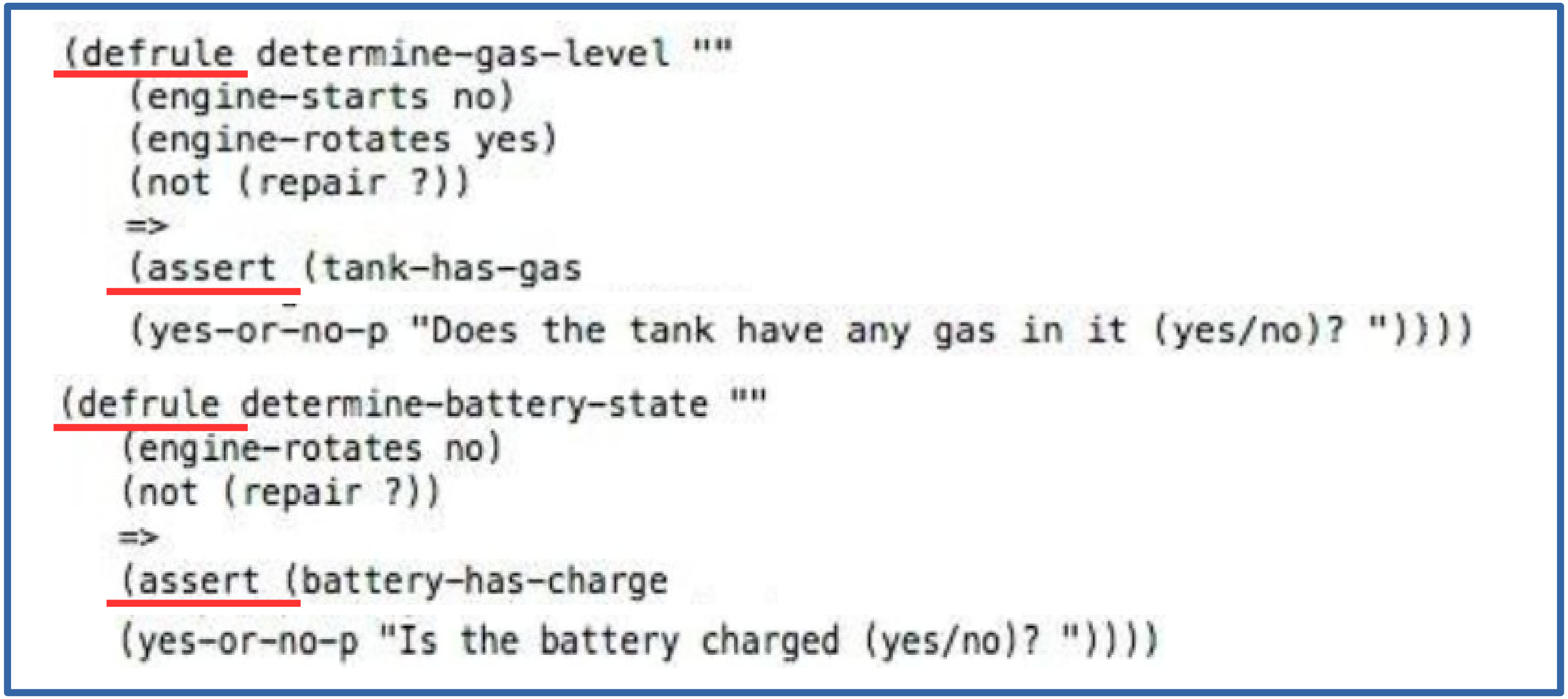
CLIPS

http://clipsrules.sourceforge.net/Version63Beta.html

A Tool for Building Expert Systems

“CLIPS is a forward-chaining rule-based programming language in C with procedural and object-oriented programming facilities”

Sample:



```
(defrule determine-gas-level ""
    (engine-starts no)
    (engine-rotates yes)
    (not (repair ?))
    =>
    (assert (tank-has-gas

    (yes-or-no-p "Does the tank have any gas in it (yes/no)? "))))

(defrule determine-battery-state ""
    (engine-rotates no)
    (not (repair ?))
    =>
    (assert (battery-has-charge

    (yes-or-no-p "Is the battery charged (yes/no)? "))))
```

Mac Version and Windows Version can be down loaded.

https://sourceforge.net/projects/clipsrules/files/CLIPS/6.30/clips_documentation_630.zip/download?use_mirror=superb-sea2&r=https%3A%2F%2Fsourceforge.net%2Fprojects%2Fclipsrules%2Ffiles%2FCLIPS%2F6.30%2F&use_mirror=superb-sea2

9-21-2018 Embed CLIPS To Other C++ Program

CLIPS was designed to be embedded within other programs, the user must provide a main program. Calls to CLIPS are made like any other subroutine. To embed CLIPS, add include statements to the user’s main program file:

#include "clips.h"

Section 4:

Embedding CLIPS

pp. 67

Section 7:

I/O Router System

Appendix C:

I/O Router Examples

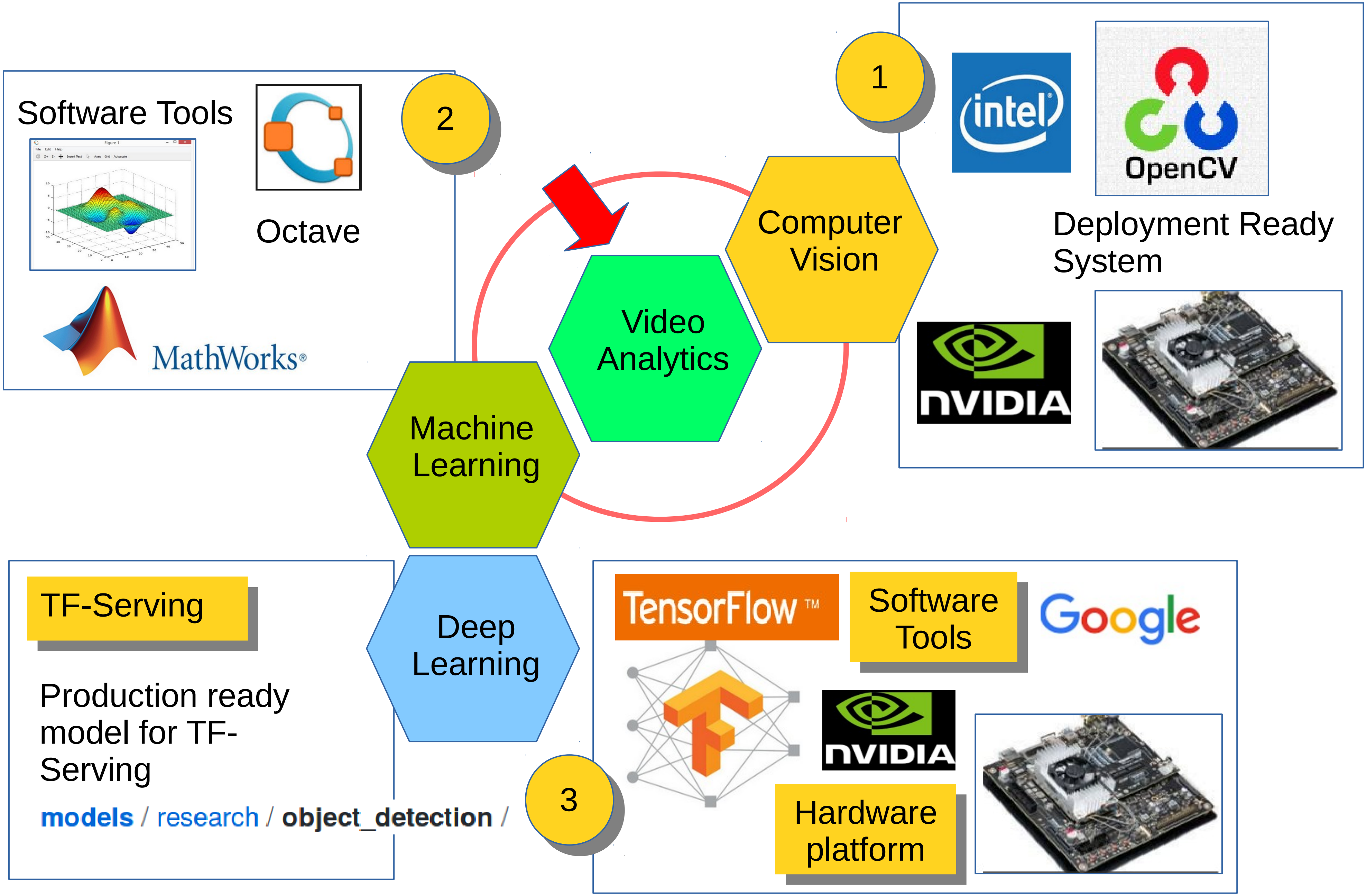
pp. 253

Example: pp. 257

C.3 Batch System

More examples from conference proceedings third_clips_conference_proceedings.pdf

Computer Vision



1

Deployment Ready System

2

Software Tools

Octave

Computer Vision

Video Analytics

Machine Learning

Deep Learning

TF-Serving

Production ready model for TF-Serving

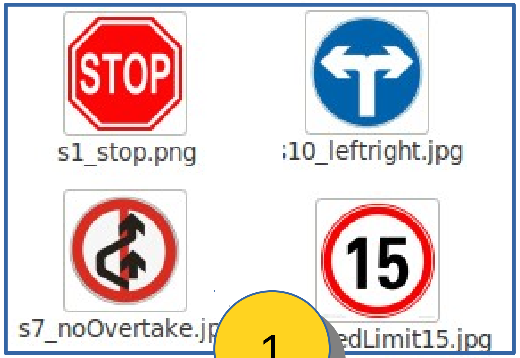
models / research / **object_detection** /

3

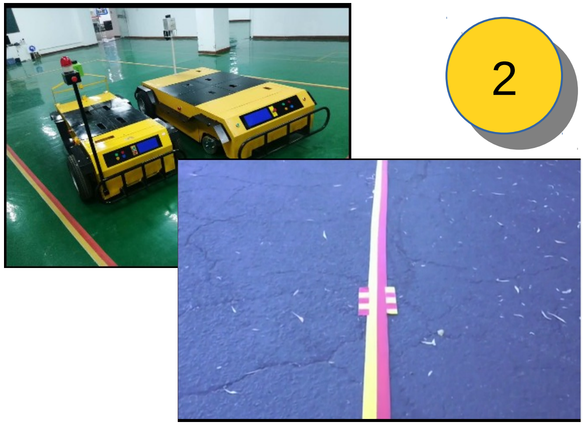
Software Tools

Hardware platform

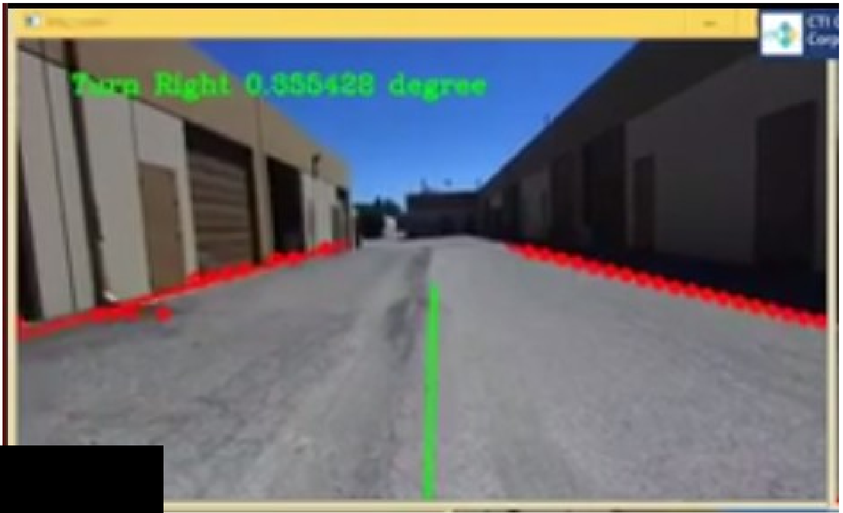
Objective Is To Build Video Search Engine

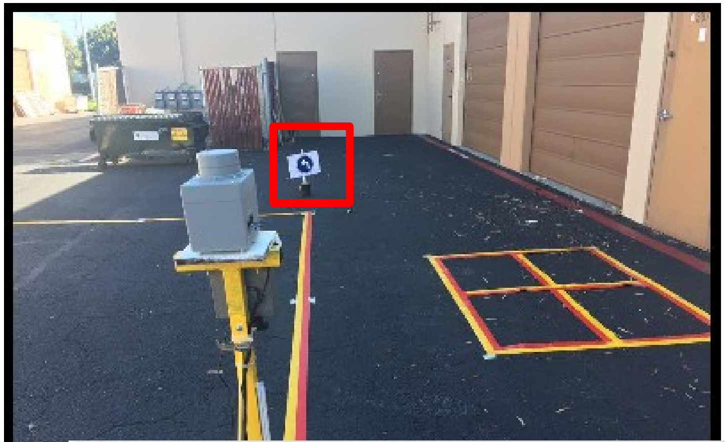


1

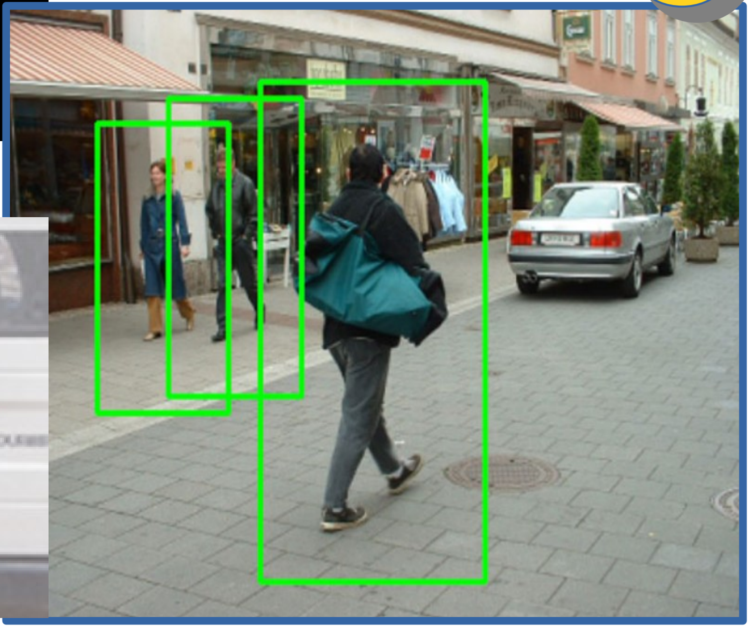


2

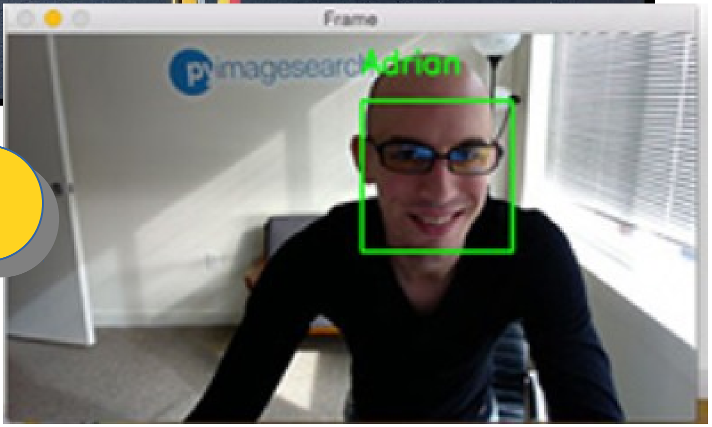




3

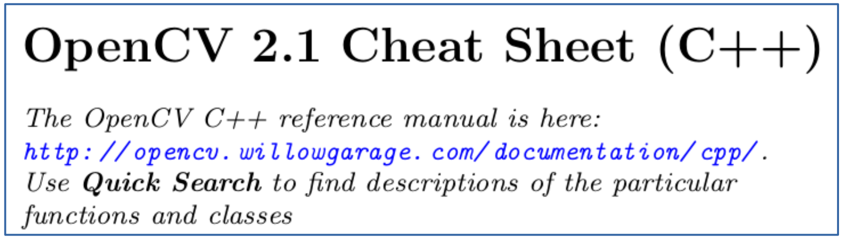


4



5

To Prepare For Interview (1)



OpenCV 2.1 Cheat Sheet (C++)

*The OpenCV C++ reference manual is here: http://opencv.willowgarage.com/documentation/cpp/. Use **Quick Search** to find descriptions of the particular functions and classes*

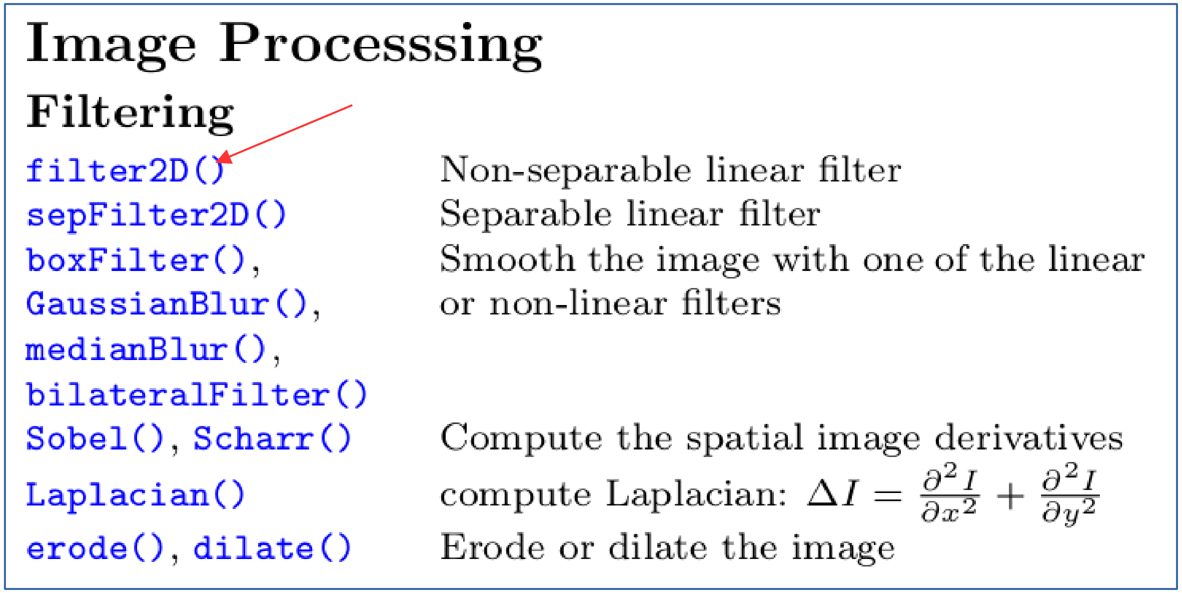
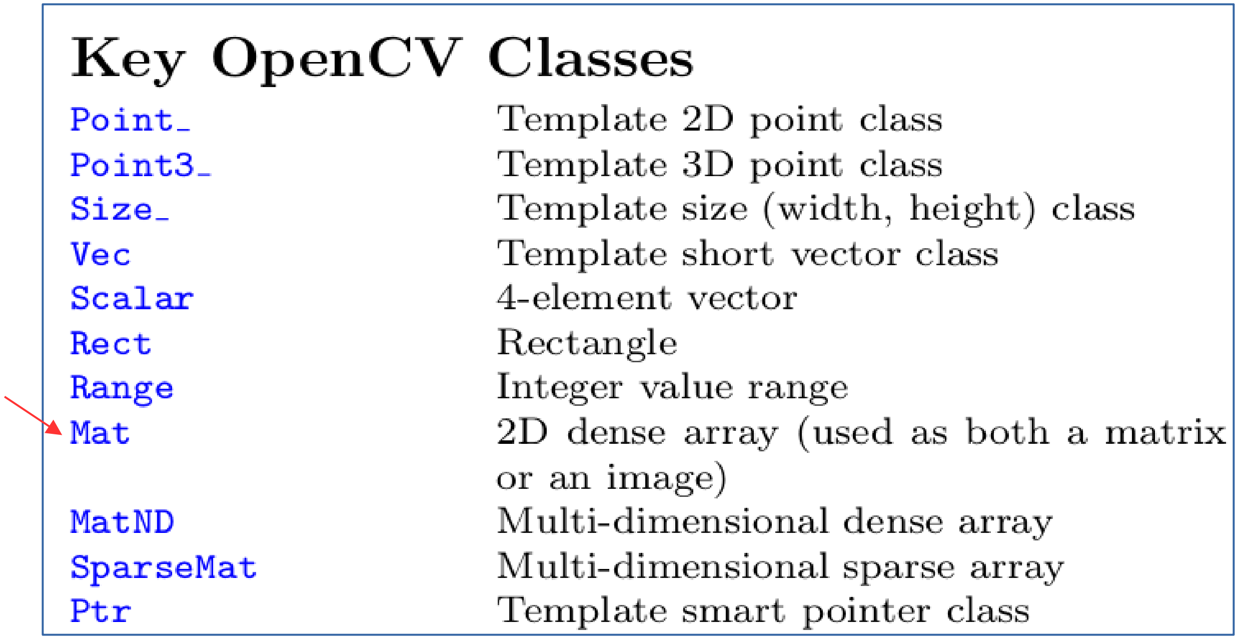


Image Processsing

Filtering

filter2D()	Non-separable linear filter
sepFilter2D()	Separable linear filter
boxFilter(), GaussianBlur(), medianBlur(), bilateralFilter()	Smooth the image with one of the linear or non-linear filters
Sobel(), Scharr()	Compute the spatial image derivatives
Laplacian()	compute Laplacian: $\Delta I = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2}$
erode(), dilate()	Erode or dilate the image



Key OpenCV Classes

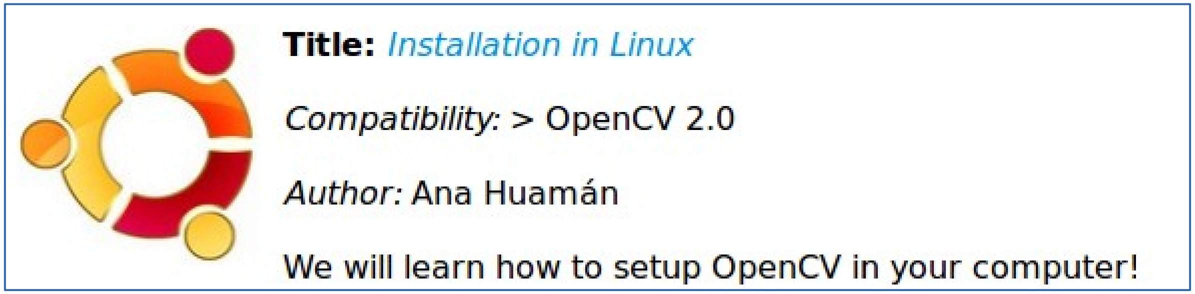
Point_	Template 2D point class
Point3_	Template 3D point class
Size_	Template size (width, height) class
Vec	Template short vector class
Scalar	4-element vector
Rect	Rectangle
Range	Integer value range
Mat	2D dense array (used as both a matrix or an image)
MatND	Multi-dimensional dense array
SparseMat	Multi-dimensional sparse array
Ptr	Template smart pointer class

Set Up OpenCV

http://docs.opencv.org/2.4/doc/tutorials/introduction/table_of_content_introduction/table_of_content_introduction.html

How to set up openCV

http://docs.opencv.org/2.4/doc/tutorials/introduction/linux_install/linux_install.html#linux-installation



Title: *Installation in Linux*

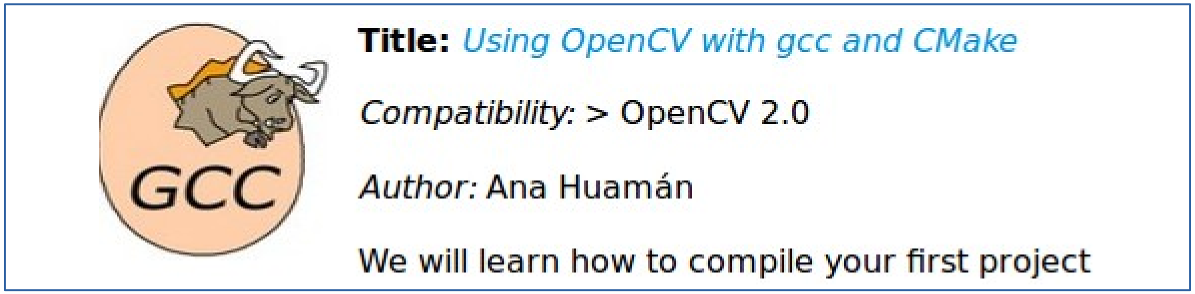
Compatibility: > OpenCV 2.0

Author: Ana Huamán

We will learn how to setup OpenCV in your computer!

How to compile and build

http://docs.opencv.org/2.4/doc/tutorials/introduction/linux_gcc_cmake/linux_gcc_cmake.html#linux-gcc-usage



Title: *Using OpenCV with gcc and CMake*

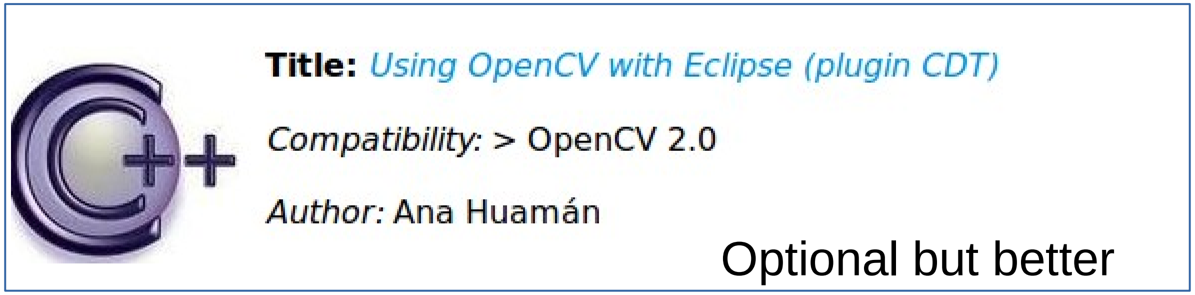
Compatibility: > OpenCV 2.0

Author: Ana Huamán

We will learn how to compile your first project

Using Eclipse

http://docs.opencv.org/2.4/doc/tutorials/introduction/linux_eclipse/linux_eclipse.html#linux-eclipse-usage



Title: *Using OpenCV with Eclipse (plugin CDT)*

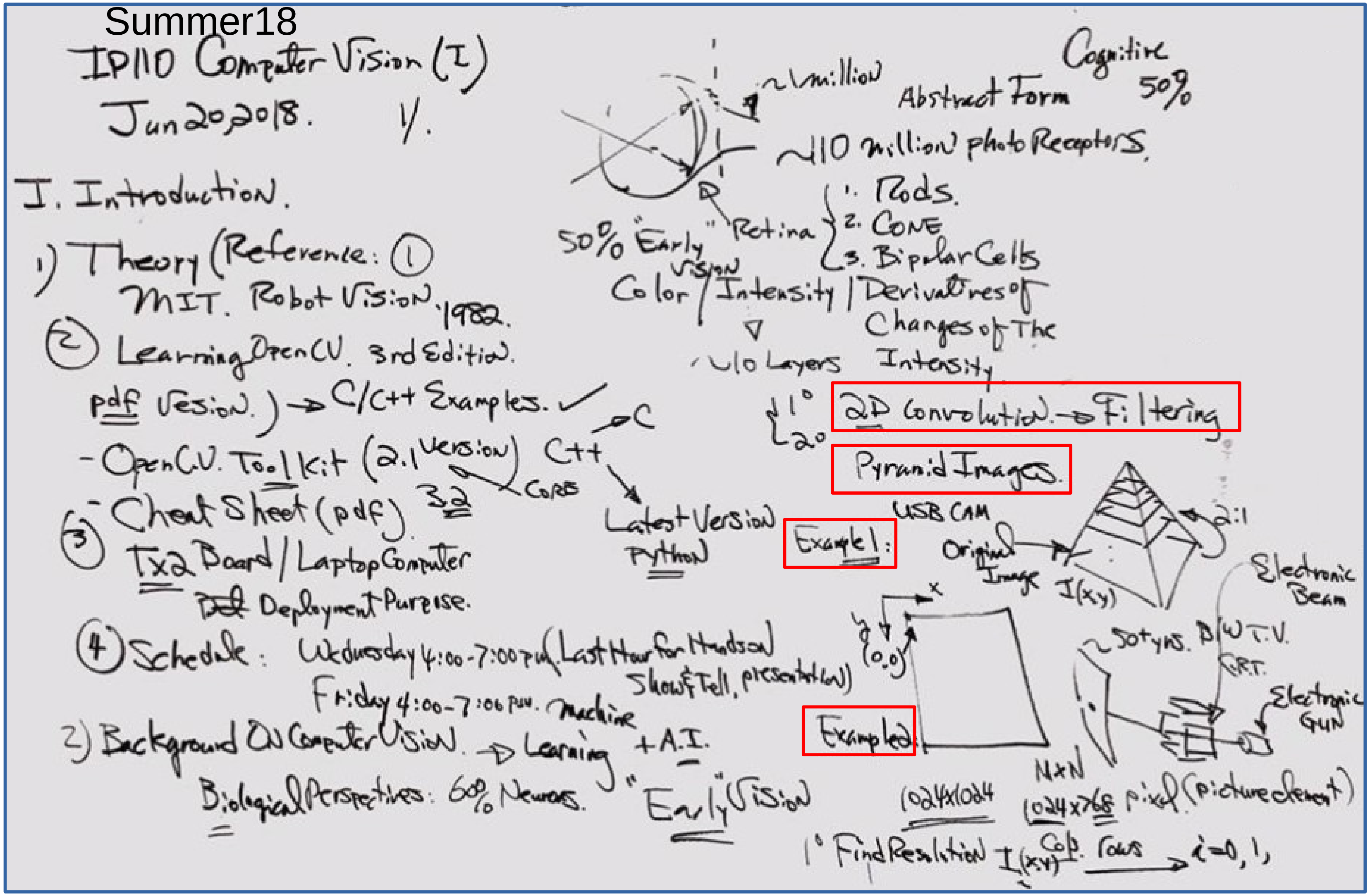
Compatibility: > OpenCV 2.0

Author: Ana Huamán

Optional but better

Introduction 6-18-18

https://github.com/hualili/opencv/tree/master/IP110-Summer18



IP110 Computer Vision (I)

Jun 20, 2018. 1/.

I. Introduction.

1) Theory (Reference: ① MIT. Robot Vision, 1982.

② Learning OpenCV. 3rd Edition. pdf Version.) → C/C++ Examples. ✓

- OpenCV. Toolkit (2.1 Version) C++ → C; Core; → Latest Version Python

- Cheat Sheet (pdf) 3.2

③ Tx2 Board / Laptop Computer ~~Del~~ Deployment Purpose.

④ Schedule: Wednesday 4:00-7:00 pm (Last Hour for Hudson Show & Tell, presentation)

Friday 4:00-7:00 pm.

2) Background On Computer Vision. → machine Learning + A.I.

Biological Perspectives: 60% Neurons. "Early Vision"

~1 million; ~110 million photo Receptors. "Retina" { 1. Rods. 2. Cone 3. Bipolar Cells

50% "Early Vision"; Abstract Form; Cognitive 50%

Color / Intensity / Derivatives of Changes of The Intensity

~10 Layers

{ 1° 2D Convolution. → Filtering

{ 2° Pyramid Images.

Example 1: USB CAM; Original Image I(x,y); 2:1

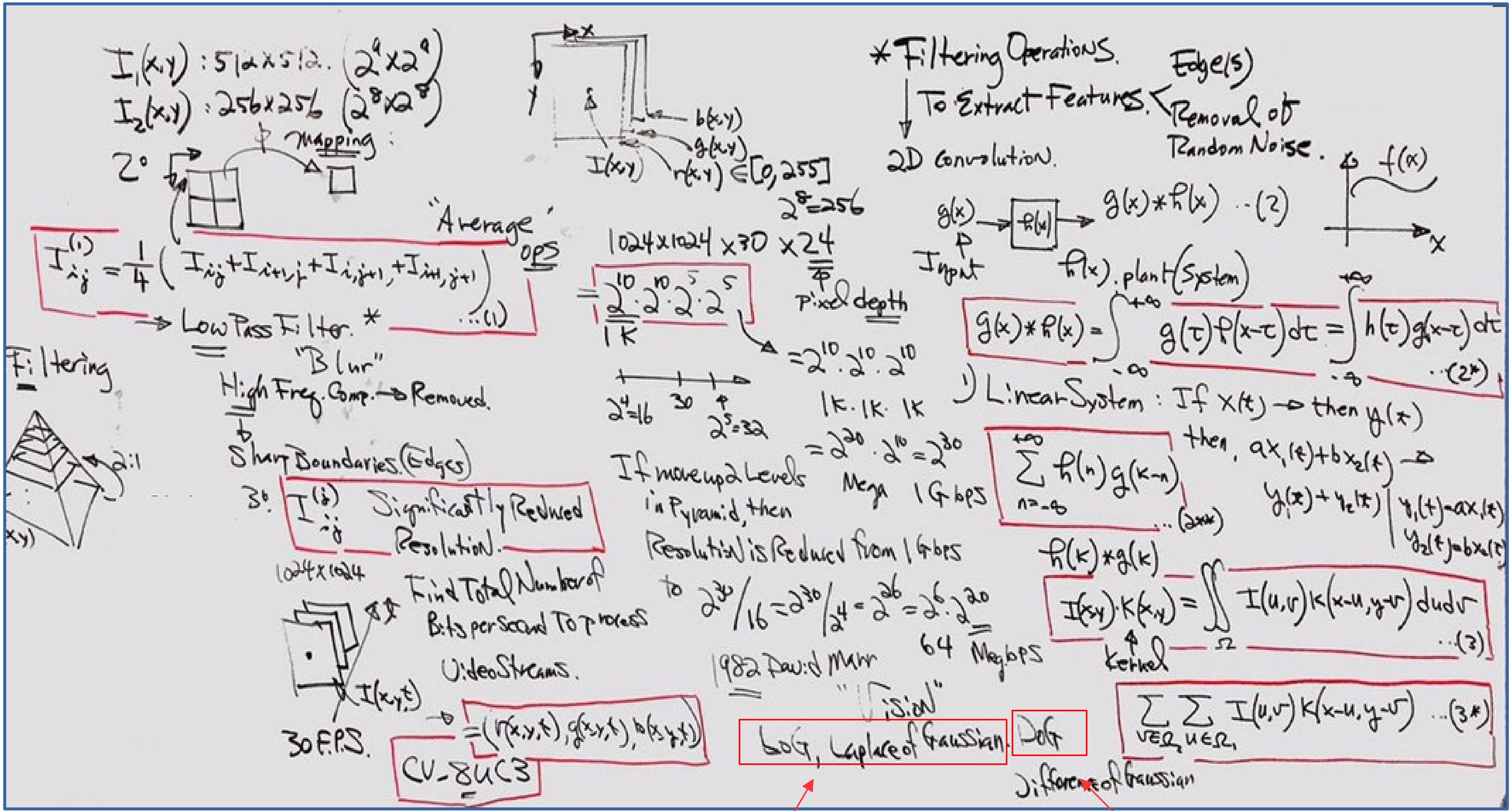
Example 2: (0,0) x y; N×N; 1024×1024; 1024×768 pixel (picture element)

~50 yrs. B/W T.V. CRT. Electronic Beam; Electronic Gun

1° Find Resolution I(x,y) col. rows → i=0, 1,

1. how and why is the image coordinate defined in such as a way? 2. CRT display device? 3. What is pyramid image? Calculation of pyramid resolution?

Pyramid And Convolution 6-18-18



$I_1(x,y)$: 512x512. $(2^9 \times 2^9)$

$I_2(x,y)$: 256x256 $(2^8 \times 2^8)$

Mapping :

2°

$$I^{(1)}_{ij} = \frac{1}{4}\left(I_{ij} + I_{i+1,j} + I_{i,j+1} + I_{i+1,j+1} \right) \quad \ldots (1)$$

"Average" ops

→ Low Pass Filter. *

Filtering

"Blur"

High Freq. Comp. → Removed.

↓

Sharp Boundaries (Edges)

2:1

(x,y)

3°: $I^{(j)}_{ij}$ Significantly Reduced Resolution.

1024x1024

Find Total Number of Bits per Second To process VideoStreams.

I(x,y,t)

30 F.P.S.

→ $= (r(x,y,t), g(x,y,t), b(x,y,t))$

CV_8UC3

x

y

I(x,y)

b(x,y)

g(x,y)

$r(x,y) \in [0, 255]$

$2^8 = 256$

1024x1024 x30 x24

Pixel depth

$= \underset{1K}{2^{10}} \cdot 2^{10} \cdot 2^5 \cdot 2^5$

$= 2^{10} \cdot 2^{10} \cdot 2^{10}$

1K · 1K · 1K

$2^4 = 16$ 30 $2^5 = 32$

$= 2^{20} \cdot 2^{10} = 2^{30}$

Mega 1 Gbps

If move up 2 Levels in Pyramid, then Resolution is Reduced from 1 Gbps to $2^{30}/16 = 2^{30}/2^4 = 2^{26} = 2^6 \cdot 2^{20}$

64 Megbps

1982 David Marr

"Vision"

LoG, Laplace of Gaussian. DoG

Difference of Gaussian

* Filtering Operations.

↓ To Extract Features. < Edge(s) / Removal of Random Noise.

2D Convolution.

$g(x) \rightarrow \boxed{h(x)} \rightarrow g(x) * h(x) \quad \ldots (2)$

Input

h(x). plant (System)

f(x)

x

$$g(x) * h(x) = \int_{-\infty}^{+\infty} g(\tau) h(x-\tau) d\tau = \int_{-\infty}^{+\infty} h(\tau) g(x-\tau) d\tau \quad \ldots (2^*)$$

1) Linear System : If x(t) → then y(t)

then, $a x_1(t) + b x_2(t)$ → $y_1(t) + y_2(t)$ | $y_1(t) = a x_1(t)$, $y_2(t) = b x_2(t)$

$$\sum_{n=-\infty}^{+\infty} h(n) g(k-n) \quad \ldots (2^{**})$$

h(k) * g(k)

$$I(x,y) \cdot K(x,y) = \iint_{\Omega} I(u,v) K(x-u, y-v)\, du\, dv \quad \ldots (3)$$

Kernel

$$\sum_{v \in \Omega_2} \sum_{u \in \Omega_1} I(u,v) K(x-u, y-v) \quad \ldots (3^*)$$

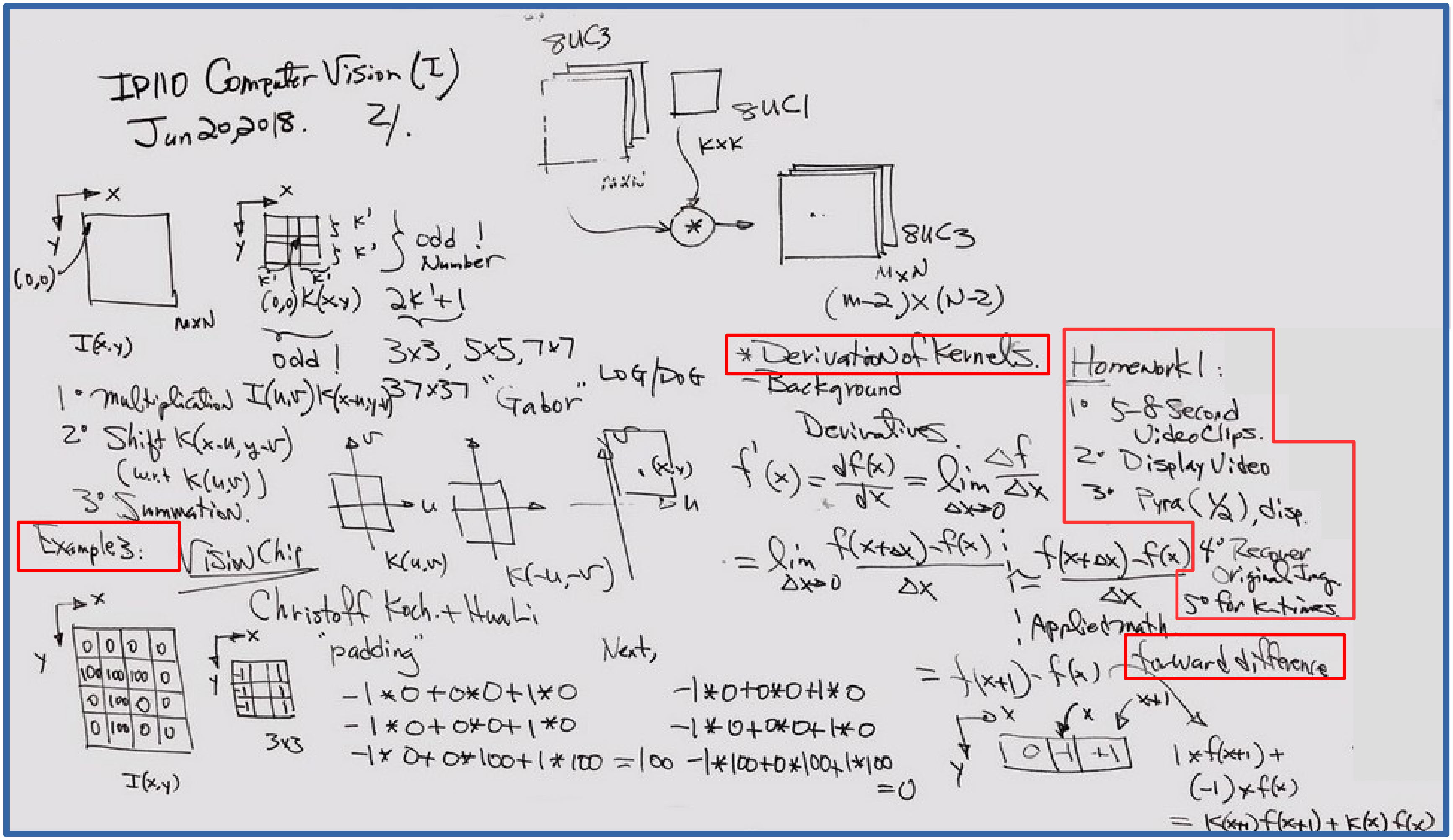
4. How to form pyramid image? And why is it LPF (low pass filter)? 5. Convolution definition?

LoG(x,y; mu, sigma)
Laplace of Gaussian

DoG(x,y; mu1,mu2, sigma1,sigma2)
Difference of Gaussian

Harry Li, Ph.D. , Sum 2018

Convolution Example And Kernel Concept 6-18-18



6. Computation of 2D convolution? 7. what is a kernel? Why is it defined as a symmetric odd numbered square pattern in most cases? 8. OpenCV data type CV_8UC3, and CV_8UC1? 9. OpenCV definition of Mat?