IDC410
A course on Image Processing and Machine Learning
(Lecture 01)

Shashikant Dugad,

IISER Mohali

Outline

- **Course Requirements**

- **Course Schedule**

- **Grading Process**

- **Course Content**

- **Software**

- **Introduction to Images**

Course Requirements

- **For enrollment to this course for *credit;* students should have already passed an IDC409 course on Introduction to Data Science**
  - **Last date for course registration is 9th January or as per the guidelines**
  - **Grades will not be given if IDC409 is not cleared before registering for this course**
  - **Functional Coding knowledge using Python is _MUST_ for this course**

- **For enrollment to this course for *audit*, attendance and continuous evaluation is mandatory**

- **Laboratory activities will be performed by a batch of three students**
  - **Batch will be formed by consecutive roll numbers registered for the course**

- **Viva will be taken separately for each student**
  - **There is a possibility of laboratory quiz/exam/short project**

Course Schedule

End Sem Exams: 2nd to 11th May Theory Exam including entire syllabus

Month	Week	Activity	Tentative flow of Content
January	2	CL_01, CL_02, LAB_01	Orientation, Software Installation, Basic Examples
	3	CL_03, LAB_02	Image Processing: Theory and Practical
	4	CL_04, CL_05, LAB_03	Image Processing: Theory and Practical
February	1	CL_06, CL_07, LAB_04	Image Processing: Theory and Practical
	2	Mid_Sem_1	**Not to be conducted**
	3	CL_08, CL_09, LAB_05	Neural Network: Theory and Practical
	4	CL_10, CL_11, LAB_06	Neural Network: Theory and Practical
March	1	CL_12, CL_13, LAB_07	Neural Network: Theory and Practical
	2	Mid Sem_2	**Theory Exam**
	3	CL_14, CL_15	Neural Network: Theory
	4	CL_16, LAB_08	Application of Neural Network: Theory and Practical
April	1	CL_17, CL_18, LAB_09	Application of Neural Network: Theory and Practical
	2	CL_18, CL_19, LAB_10	Application of Neural Network: Theory and **Practical Exam**
	3	CL_19,	Application of Neural Network: Theory
	4		**Extra Classes/Labs if needed and as permisible**

Schedule and Grading

- **Schedule: Monday and Tuesday at LH4: 4-5 PM, Friday at Computer Lab1 and 2: 2-5 PM**

- **Mid-Sem2 Exam: 20% and Final Exam: 20%**

- **Attendance: 10%**
  - **Minimum attendance required for appearing in mid-sem and final exams: 75%**
  - **Attendance below 75% would require endorsement from dean academics**

- **Continuous Lab Performance (CLP) assessment: 25%**

- **Viva: 25%**
  - **Possibility of Lab Exams or Mini projects (marks to be extracted from CLP and/or Viva)**

* There may be minor modifications/additions as we progress

Syllabus Outline

- **Introduction to OpenCV, GPU and CUDA**

- **Image Handing, Thresholding, Histogram re-shaping algorithms, Optimization of contrast, Image processing, Noise Filters, Gradients, Edge Detection**

- **Neural Network: Motivation, Perceptron, activation function, weights and biases, simple neural networks, Forward and Back propagation, Optimization methods for obtaining weights and biases, fully connected networks**

- **Deep Neural Network, Convolutional Neural Network, Autoencoders, Recurrent Neural Network, Darknet 51, Object detection with You Look Only Once (YOLO), Regression with Graphical Neural Network (GNN), any other important DNN**

OpenCV Software

OpenCV: History, Features and Evolution

- **OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library**

- **The Library has several thousands optimized algorithms for tasks like Image Processing, Object detection, Machine Learning, Regression etc.**
  - **Removing noise from images, Detecting edges, stitching images, analysing videos, flow measurements**
  - **Objects Detection, Posture recognition, Analysing objects in multi dimensional space etc.**

- **The library used extensively by research community, service companies like Google, Microsoft etc.**

OpenCV: History, Features and Evolution

- **Libraries can be accessed through C++, Python, Java and MATLAB interfaces and it supports all major platforms like Windows, Linux, Android and Mac OS and provides interface to multi-thread and GPU based architecture for speedy execution.**

- **NVIDIA based GPU architecture can be accessed through CUDA programming language. GPU provides pne of the most power powerful compact hardware platform for parallel computing which is extremely useful in big data science analytics and real time computing intensive applications**

Software Installations

Programing tools…

- *Python* is one of the most powerful programming languages, mostly used in data science, machine learning, and big data analytics. So, installing Python is essential for all programmers.

- *Jupyter notebook* is the most popular interface to develop your programs

- OS to be used: Windows, Linux-Ubuntu (recommended), macOS

- Softwares to be installed: Python, pip, Jupyter notebook

- Programming Language: Python *(This is pre-requisite for this course)*

- Dependencies: pandas, numpy, Matplotlib, scikit-learn etc.

OpenCV

Installing Python (Windows)

OpenCV

| Python | PSF | Docs | PyPI | Jobs | Community |

python™

Donate   Search   GO   Socialize

About   Downloads   Documentation   Community   Success Stories   News   Events

Download the latest version for Windows

Download Python 3.8.0

Looking for Python with a different OS? Python for Windows,
Linux/UNIX, Mac OS X, Other

Want to help test development versions of Python? Prereleases,
Docker images

Looking for Python 2.7? See below for specific releases

Join the official 2019 Python Developers Survey   Start the survey!

Looking for a specific release?
Python releases by version number:

Shashikant R. Dugad, IISER Mohali

12

Adding Python installation path to Windows' PATH Environment Variable

Ensure Successful installation of Python on Windows

Installing pip (Windows)

- **Download get-pip.py (** https://bootstrap.pypa.io/get-pip.py **) to a folder on your computer**

- **Open a command prompt window by pressing (Windows key + R and then typing cmd) and navigate to the folder containing get-pip.py. Then,**
    > python get-pip.py

- **Verify successful installation of pip by testing:**
    >pip -V or >pip --version

Adding PIP To Windows Environment Variables

- Go to System and Security > System in the Control Panel once it has been opened.

- On the left side, click the Advanced system settings link.

- Then select Environment Variables.

- Double-click the PATH variable under System Variables.

- Click New, and add the directory where pip is installed, e.g. **C:Python33Scripts**, and select OK.

- Verify successful installation of pip by testing:
    >pip –V   OR   **>pip --version**

Installation of Jupyter on Windows

- **Make sure Python and latest version of pip is installed**

- **Open the command prompt by pressing Windows key + R and then typing cmd:**
  > pip install jupyter

- **Launching jupyter notebook: Navigate to your program directory where you want to save your notebooks using the cd command:**
  > cd path\to\your\desired\directory

- **To launch run:**
  > jupyter notebook
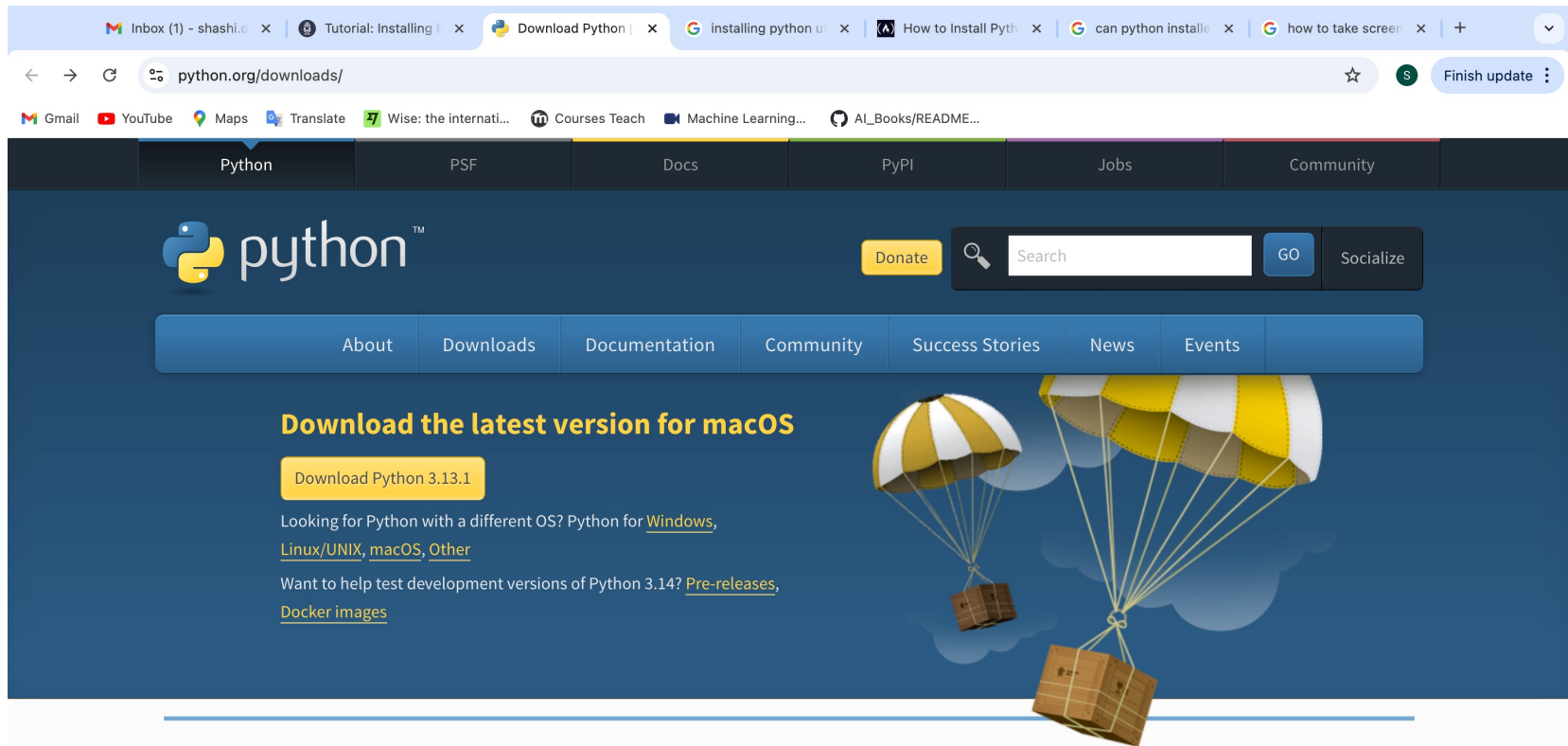
Installing OpenCV (Windows)

- **Open Command Prompt and run the following command**

    > pip install opencv-contrib-python

- **Detailed guide here**

- **To include OpenCV library in your programs use: *import cv2***

Installation of Python on macOS

https://www.python.org/downloads/

Installation of Python on macOS

Once the download is complete, double-click the package to start installing Python. The installer will walk you through a wizard to complete the installation, and in most cases, the default settings work well, so install it like the other applications on macOS. You may also have to enter your Mac password to let it know that you agree with installing Python.

Installing pip on macOS

- **Method1: Installing pip via ensurepip:- Open a new Terminal and Run the following command to start the installation:**

  > python3 -m ensurepip

- **Following output confirms the installation was successful**

```
[marko@Markos-MacBook ~ % python3 -m ensurepip
Defaulting to user installation because normal site-packages is not writeable
Looking in links: /var/folders/5s/mdf1164n5h7c33kzx78656qc0000gn/T/tmp0aae9m0m
Requirement already satisfied: setuptools in /Library/Developer/CommandLineTools
/Library/Frameworks/Python3.framework/Versions/3.9/lib/python3.9/site-packages (
58.0.4)
Processing /private/var/folders/5s/mdf1164n5h7c33kzx78656qc0000gn/T/tmp0aae9m0m/
pip-21.2.4-py3-none-any.whl
Installing collected packages: pip
Successfully installed pip-21.2.4
marko@Markos-MacBook ~ %
```

Installing pip macOS

- **Method2: Installing pip via homebrew:- On Homebrew, pip comes pre-packaged with the Python installation. To install pip, install the Python formula with the following command:**
  > **brew install python**

- **Verify successful installation of pip by testing:**
  >**pip –V   OR   >pip --version**

- **Include it in path:**
  > **export PATH=${PATH}:"/opt/homebrew/opt/python/libexec/bin"**

Installation of Jupyter on macOS

- **Jupyter is preferred framework under which all the python software's may be developed**

- **Open a terminal and give following commands in the terminal**
  > brew install jupyter

- **Restart the macbook**
  > sudo chown *username* /Users/*username*/Library/LaunchAgents/
  Provide sysadmin password
  > brew services start jupyterlab

- **Installation of jupyter is COMPLETE now!**

Installation of Jupyter on macOS

- **To launch jupyter notebook, first go to the directory where your software is located for developing/executing the code and from there:**

  > jupyter notebook

- **On Safari Browser, if you get a message saying: "Pop-up window blocked"**

- **Solution: Open Safari Menu → Settings →Pop-up Windows: On the RHS panel change localhost setting to ALLOW**

Installing Python, pip and OpenCV in Linux (Ubuntu)

- **Install Python using**

  > **sudo apt-get install python**

- **Install pip using**

  > **sudo apt install python3-pip**

- **Install OpenCV**

  > **pip install opencv-contrib-python**

- **Guide for Ubuntu <u>here</u>.**

**Reference for Image Processing: MACHINE VISION. by
Ramesh Jain, Rangachar Kasturi, Brian G. Schunck
Published by McGraw-Hill, Inc., ISBN 0-07-032018-7, 1995**

https://cse.usf.edu/~r1k/MachineVisionBook/MachineVision.files/

What is Image?

What is an Image?

- **Image is a large matrix (N x M) of pixels**

- **Each pixel is represented by it's color**

- **Color is a mixture of 3 fundamental colors namely Red, Green and Blue (RGB) as per intensity of each color**

- **Intensity of each of the color is represented by a single byte, thus having a value between 0-255**

- **For a Grayscale image: Pixel intensity is denoted by a single byte varying between 0 (black) to 255 (white).**

The image read with the OpenCV function imread(), provides the colours in order of BGR (Blue, Green and Red).

```
1  img = cv2.imread("C:\\Users\\Darshita\\Desktop\\OpenCV\\flower.jpg")
```
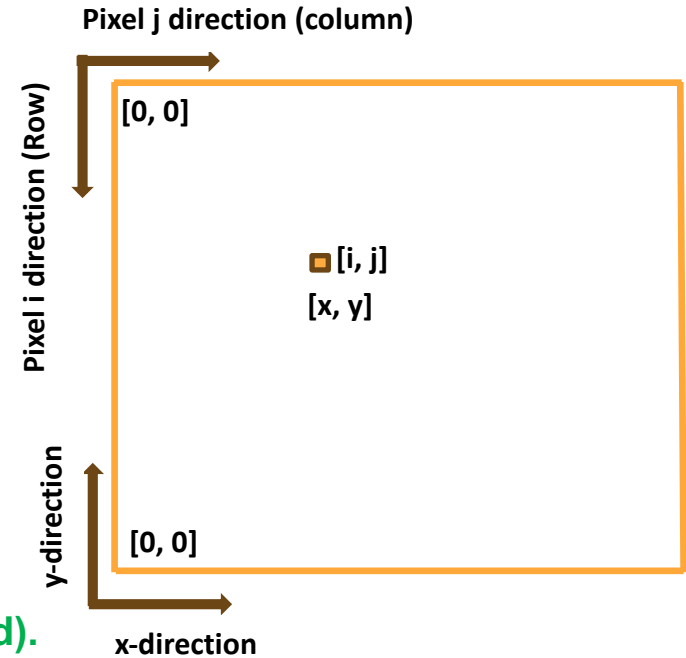
What is an Image?

Image Size: Image denoted by n X m pixels, where n is number of pixels in vertical direction (or number of rows) and m is number of pixels in horizontal direction (or number of columns) correspond to a size of 3nm bytes (before compression) .
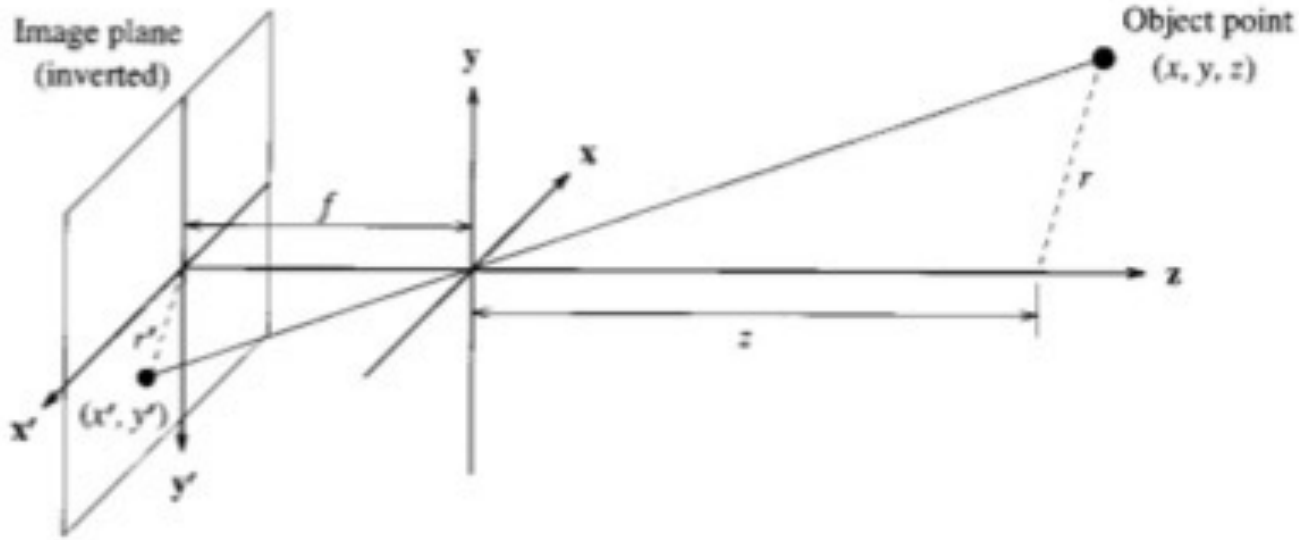
Pixel Location: A location of pixel in an image M is denoted by M(i, j) where, i is a row number (vertical direction) and j is a column number (horizontal direction)

The image read with the OpenCV function imread(), provides the colours in order of BGR (Blue, Green and Red).

Pixel j direction (column)

Pixel i direction (Row)

[0, 0]

[i, j]
[x, y]

y-direction

[0, 0]

x-direction

```
1  img = cv2.imread("C:\\Users\\Darshita\\Desktop\\OpenCV\\flower.jpg")
```
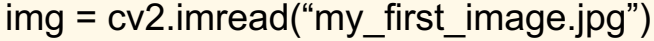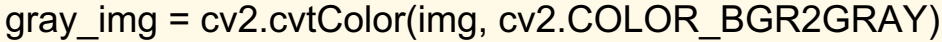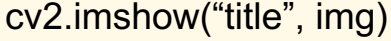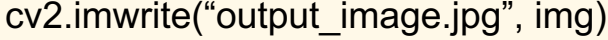
Physical Object Geometry and Image Geometry

$$\frac{r'}{r} = \frac{z}{f}, \quad x' = \frac{f}{z}x \quad \text{and} \quad y' = \frac{f}{z}y \qquad\qquad x' = j - \frac{m-1}{2} \quad \text{and} \quad y' = -\left(i - \frac{n-1}{2}\right)$$

$$r = \sqrt{x^2 + y^2} \text{ and } r' = \sqrt{x'^2 + y'^2}$$

Image size: N X M

Basic OpenCV commands in Python

Operation	Syntax/Algorithm	Result
Read Image	img = cv2.imread("my_first_image.jpg") img = cv2.imread('your_image.jpg', cv2.IMREAD_GRAYSCALE)	
Image Dimension	height, width, channels = img.shape	
	height, width = img.shape //grey scale	
Color to Gray	gray_img = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY) $I_G = (I_R + I_G + I_B)/3$ OR $I_G = 0.299 \times I_R + 0.587 \times I_G + 0.114 \times I_B$	
Display Image	cv2.imshow("title", img)	
Hold Image Window	cv2.waitKey(itime)	itime in msec
Save Image	cv2.imwrite("output_image.jpg", img)	

Additional feature to explore! Drawing regular shapes (lines, circle, rectangles etc) and Text, Image Rotation by a given angle, Image Resizing