COMPUTER ENGINEERING (LM75)

(Lecce - Università degli Studi)

Teaching COMPUTER VISION AND DEEP LEARNING

GenCod A006808

Owner professor Cosimo DISTANTE

Teaching in italian COMPUTER VISION E DEEP LEARNING

Teaching COMPUTER VISION AND DEEP LEARNING

SSD code ING-INF/03

Reference course COMPUTER ENGINEERING

Course type Laurea Magistrale

Credits 9.0

Teaching hours Front activity hours: 81.0

For enrolled in 2022/2023

Taught in 2023/2024

Course year 2

Language ITALIAN

Curriculum Intelligenza artificiale

Location Lecce

Semester Second Semester

Exam type Oral

Assessment Final grade

Course timetable https://easyroom.unisalento.it/Orario

BRIEF COURSE DESCRIPTION

Computer Vision today is everywhere in our society and images have become pervasive, with applications in several sectors; just to mention some in: apps, drones, healthcare and precision medicine, precision agricolture, searching, understanding, control in robotics and self-driving cars. The course introduces the basics of image formation, reconstruction and inferring motion models, as well as camera calibration theory and practice.

Recent developments in neural networks (Deep Learning) have considerably boosted the performance of the visual recognition systems in tasks such as: classification, localisation, detection, segmentation etc. Students will learn the building blocks of a general convolutional neural network, the way how it is trained and optimized, how to prepare a dataset and how to measure the final performance.

REQUIREMENTS

No prior experience with computer vision is assumed, although previous knowledge of visual computing or signal processing will be helpful. The following skills are necessary for this class:

- Math: Linear algebra, vector calculus, and probability. Linear algebra is the most important.
- Data structures: Students will write code that represents images as feature and geometric constructions.
- Programming: A good working knowledge. All lecture code and project starter code will be Python, and Pytorch for Deep Learning, but student familiar with other frameworks such as tensorflow is ok.

COURSE AIMS	Upon completion of this course, students will: 1. Be familiar with both the theoretical and practical aspects of computing with images; 2. Have described the foundation of image formation, measurement, and analysis; 3. Have implemented common methods for robust image matching and alignment; 4. Understand the geometric relationships between 2D images and the 3D world; 5. Have gained exposure to object and scene recognition and categorization from images; 6. Grasp the principles of state-of-the-art deep neural networks; and 7. Developed the practical skills necessary to build computer vision applications.
TEACHING METHODOLOGY	Teaching is based on theoretical and practical lectures. The student will write in python algorithms taught in class
ASSESSMENT TYPE	Oral session. The student will explain the developed project and shall answer two or more questions regarding theoretical aspects of the studied topics
ASSESSMENT SESSIONS	The student must develop a project by choosing a practical simple application with some algorithms done during the course. The choice is at total disposal of the student, as well as the fact of developing it in group os solo. In group setting the students must proof their own activities developed in the common project application. The final examination is based on oral assessment of the topics covered during lectures.
OTHER USEFUL INFORMATION	For the LAB practice, students may use for the deep learning development the Google Colab or Cloud Platform.

FULL SYLLABUS	Introduction to Computer Vision Camera models and colors Image Filtering Fourier - image pyramids and blending Detecting Corners 2D and 3D geometric primitives - Projections Operations with images Image Alignment - warping, homography estimation direct linear transform robust motion estimation with Ransac - perspective n point problem. Registration examples: face recognition, medical imaging Camera Calibration - distortion models and compensations - linear methods for camera parameters. Calibration with a checkerboard LAB - SIFT and camera calibration Multiview geometry - Epipolar geometry, position error estimation, stereo rig, Essential matrix estimation, rectification, Reconstruction, correspondense problem, weak calibration and ransac estimation of fundamental matrix Image Classification - Key nearest neighbor, linear classifiers LAB - Canny edge detection, Hough Transform Image Classification - loss functions, optimization with stochastic gradient descent neural networks LAB - Introduction to Pytorch framework backpropagation, computational graphs and gradient estimation Image Classification - Convolutional Neural Network architecture Normalization; Image Classification - CNN architectures (Alexnet, VGG, GoogleNet, ResNET, DenseNet, SENet, EfficientNet), Siamese Architectures (applications to face verification, people and vehicle re-identification) LAB - CNN Recurrent networks- RNN, LSTM, GRU Language modeling Sequence-to-sequence Image captioning Attention Multimodal attention Self-Attention Transformers Object detection Transfer learning Object detection task R-CNN detector Non-Max Suppression (NMS) Mean Average Precision (mAP) Single-stage vs two-stage detectors YOLO Region Proposal Networks (RPN), Anchor Boxes Two-Stage Detectors: Fast R-CNN, Faster R-CNN Feature Pyramid Networks LAB - Object detection Object segmentation - Single-Stage Detectors: RetinaNet, FCOS Semantic segmentation Instance segmentation Keypoint estimation LAB - Deep Learning application to segmentation Generative Models

Supervised vs Unsupervised learning
Discriminative vs Generative models
Autoregressive models
Variational Autoencoders
Motion estimation, Optical flow
Diffusion models
3D Vision - 3D shape representations
Depth estimation
3D shape prediction
Voxels, Pointclouds, SDFs, Meshes
Implicit functions, NeRF
Videos
Video classification
Early / Late fusion
3D CNNs
Two-stream networks
Transformer-based models
Reinforcement learning

REFERENCE TEXT BOOKS

There is no requirement to buy a book. The goal of the course is to be self contained, but sections from the following textbooks will be suggested for more formalization and information.
The primary course text will be Rick Szeliski's draft Computer Vision: Algorithms and Applications 2nd Edition 2022; we will use an online copy (fill the form) at this link.
We will be using Piazza for all course notes, homework and final project.
A copy and link will be provided in website.
A textbook for Deep Learning with Pytorch script can be accessed at this link
Deep Learning, MIT Press book, Ian Goodfellow and Yoshua Bengio and Aaron Courville