ESOF-5033-WA: Deep Learning Theory and Models (Winter 2020)

Deep and Transfer Learning based on Convolutional Neural Network

This project introduces the students to deep learning using convolutional neural networks. The students will utilize existing pre-trained deep convolutional neural network models to solve a real-world problem. The project consists of two phases

1- Designing and implementing (from scratch) a deep learning convolutional neural network to solve a real-world classification problem using TensorFlow and Keras API. The student will propose a simple architecture to be able to train it using their own dataset from phase 1.

2- The students will use the Google TensorFlow API with Python to experiment with different pre-trained models and use them to solve a classification problem and learn about the transfer learning concept.

Students are required to use Google Collaboratory https://colab.research.google.com/

Project groups: the students are encouraged to form groups of 3-4 members and work together on this project

Data Source:

The students can record a video about the object that they want to identify and then use specific software to extract the different frames in the video and use the frames as the training and testing images. In general, you need hundreds (300 to 400) of images per class to get a good result from the pre-trained models.

Deliverables:

1- Each group will upload “ipython notebook” on the D2L to show their implementation. The work must show at least 3 “THREE” working pre-trained models, link to the data, and a document to explain how to download/configure/train/test the pre-trained models for the given problems. The default implementation will not be accepted. The three models must be used in an **Ensemble model** to enhance the accuracy of object recognition.
2- A video to show the output of each model. The pre-trained model can be used to process video “an example will be provided with resources” (check your D2L).
3- The students are required to produce a similar video and show the class name of the object during the video. Example, if a group is working to identify face expressions, their video must show people “public video” with a bounding rectangle on the faces while they are moving around. The students can compose a video of the testing images if a video cannot be obtained.
4- The output of the Keras model (model that you build from scratch) must be demonstrated in the same manner
5- A summary report must be provided with a comparison between the adapted pre-trained model and Keras model.

Infrastructure: Google Colab

Prerequisite Programming Skills: Sufficient knowledge of Python is required

Deadline for Submission: See next page

Example Projects:

1. Identify a lionfish
2 . Identify environmental sounds from spectrograms (sound images)
3. Find a nucleus in a biomedical pathology image
4. Determine the species of a seedling from an image
5. Invasive Species Monitoring: Identify images of invasive hydrangea
6. Lung cancer detection from CT images
7. Melanoma skin cancer detection (Not available this year)
8. Diabetic Retinopathy Detection
9. Facial expression identification (Not available this year)
10. Bring your own problem (subjected to instructor approval)

Project assessment of the deliverables

Graduate Attribute 3 - Investigation					
Aspect/Level	**1**	**2**	**3**	**4**	**5**
Data analysis and synthesis of information: functional requirements	Majority of functional requirements was missing, incomplete and/or extremely inaccurate.	Some basic functional requirements were modeled, with however several inaccuracies. The product will not be operational.	Most functional requirements were modeled accurately. The content shows basic understanding of key ideas. The product will be operational with restriction.	Accurate identification and modeling of functional requirements.	Uses in-depth analysis of the available algorithms to suggest and introduce an innovative algorithms

The students are encouraged to contact the instructors by email and book appointment to discuss the aspects of the projects and any other concerns.

Detailed Deliverables with expected due dates

Item	Due date (by the end of)	35% total
Project Proposal + Teaming up	2nd Week	No Marks. Penalty 10% will apply if late
1. CNN from scratch “ipython notebook”. With all figs, graph, code comments. 2. Video illustrating the results “you can use the free version of Camtasia studio” 3. Report explaining the problem, solution, and the results.	6th week	15%
1. Three pre-trained CNN models “ipython notebook”. With all figs, graph, code comments. 2. Video illustrating the results per pre-trained model “you can use the free version of Camtasia studio” 3. Report explaining the problem, solution, and the results.	12th week	20%