EU: Multimedia Retrieval & Cloud Computing

Project statement

I. Introduction:

As announced during the course session, this project will be based on the knowledge acquired with the two activities learning tool (Fig.1) to develop and host an indexing and searching application multimedia on Cloud (or Edge) resources.

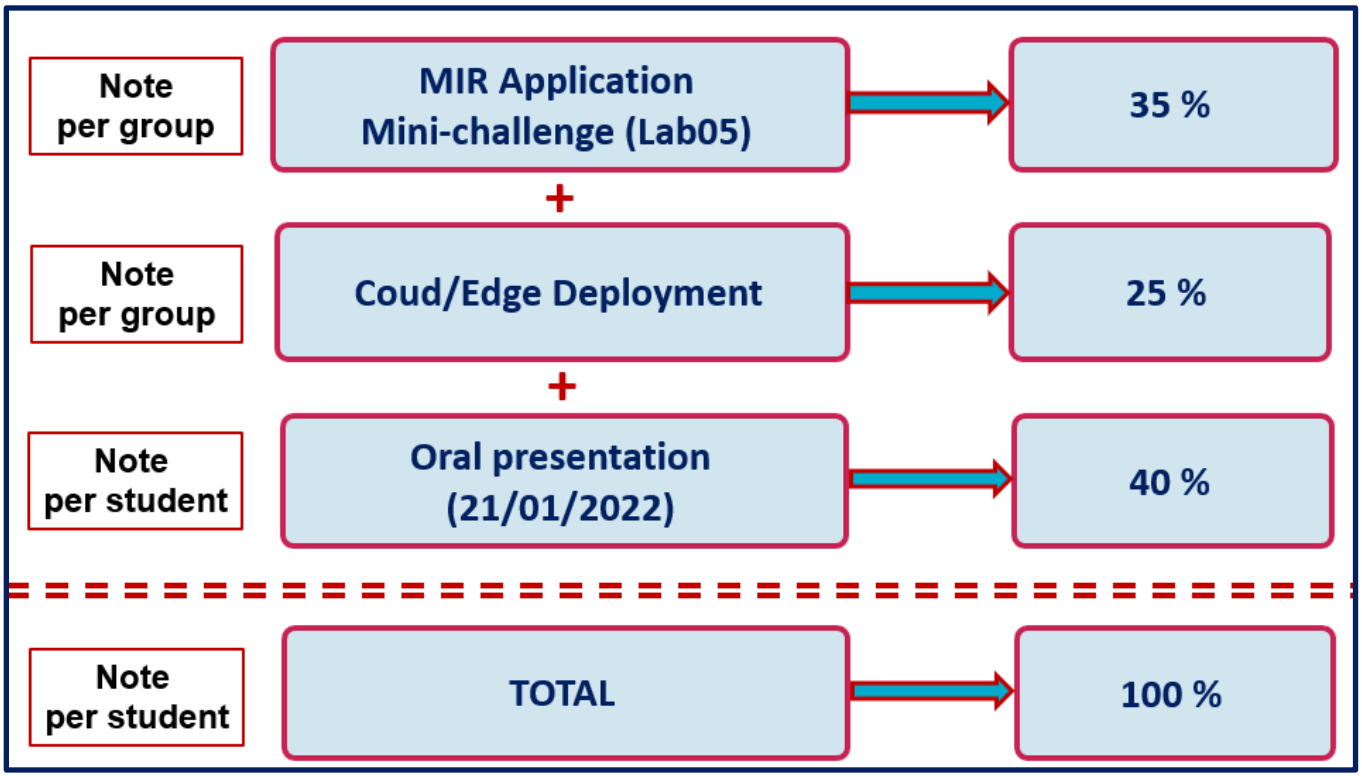

Figure 1: EU Assessment Modalities

The projects will be carried out in groups of two with the following deadlines:

- Date of submission of the project (report of approximately 20 pages + code + manual): the**01/15/2023**through Moodle
- Project presentation mode: face-to-face (Ho. 23)
- Project submission date:**01/20/2023**where each group will have**15 minutes**at most for present your project followed by**5 to 10 minutes**of questions.

Footnote 1:when presenting your projects, you may have questions related to the theory seen in class.

Running order:

- **1:00 p.m. – 1:25 p.m.:**Group 02 (Océane Mélice and Adrien Dubois)
- **1:25 p.m. – 1:50 p.m.:**Group 03 (Selim Descamps and Antoine Minnoye)
- **1:50 p.m. – 2:15 p.m.:**Group 04 (Valentin Corduant and Tanguy Vansnick)
- **2.15 p.m. – 2.40 p.m.:**Group 06 (Syed Tymeur and Maxime Glosener)
- **2:40 p.m. – 3:05 p.m.:**Group 07 (Bilal Hammas and Vivian Ruelle)
- **3:05 p.m. – 3:30 p.m.:**Group 08 (Alan Denys and Romain Monnoyer)
- **3:30 p.m. – 3:55 p.m.:**Group 09 (Romain Poblome and Rebacca Fotso)
- **4:05 p.m. – 4:30 p.m.:**Group 10 (DANTAS MACEDO José António and RIBEIRO COSME Margarida)
- **4:30 p.m. – 4:55 p.m.:**Group 11 (Simon Degouys and Yannick Basesayabo)
- **4:55 p.m. – 5:20 p.m.:**Group 12 (Nicolas Gonze and Antoine Dhanis)
- **5:20 p.m. – 5:45 p.m.:**Group 15 (NOUNDOU NJIKE Thierry and Florian Dubois)
- **5:45 p.m. – 6:10 p.m.:**Group 16 (ESTIEVENART Yorick and VENTUROSO Hugo)
- **6:10 p.m. – 6:35 p.m.:**Group 17 (IAKOVENKO Maksym and Vandekerckhove Adrien)
- **6.35 p.m. – 7 p.m.:**Group 18 (Mbenza Buanga and Patrick TEGUIA TAGNE Harold Wilfrid)

II. Project statement:

The goal of the project is to develop and host an indexing and multimedia search application on Cloud (or Edge) resources. We offer you to use your virtual machines created during the course of "Cloud and Edge Computing" to host your applications. The project consists of two parts:

• II.1. Part 01: Indexing application and multimedia search:

The objective of this part is to develop a search engine exploiting # descriptors, it will be necessary:

1. Index the database with the descriptors of your choice. If several descriptors are chosen, it it will be necessary to give the possibility of combining them;
2. Carry out the search by giving the possibility of choosing the similarity calculation function (Euclidean, Correlation, Chi-square, Bhattcharyya, Brute Force Matcher, Flann, etc.);
3. Display the Top20 and Top50 for query images;
4. Calculate Recall (R), Precision (P), Average Precision (AP), Mean Average Precision (MaP) and R-Precision

You have the choice to work under Python or C++ but this choice will have to be taken into account in the part 2 which consists of hosting your application on a cloud resource.

The groups**2, 3, 4, 10, 11, 15 and 16**will work on the database**Animals**»container**05**classes each of which contains**06** animal breed classes (**30 classes in all**). To test the engine, it will be necessary to make the following requests.

- **Database link:** https://github.com/sidimahmoudi/facenet_tf2/releases/download/AI_MIR_CLOUD/MIR_DATASETS_B.zip

Query index	Class	Pictures
R1, R2, R3	**spiders**	0_5_araignees_tarantula_795, 0_4_araignees_gardenspider_631, 0_1_araignees_wolfspider_259
R4, R5, R6	**Dogs**	1_0_dogs_Siberianhusky_849, 1_3_dogs_Chihuahua_1315, 1_1_dogs_Labradorretriever_1054
R7, R8, R9	**Birds**	2_2_birds_greatgreyowl_2092, 2_4_birds_robin_2359, 2_3_birds_bluejay_2232

The groups**6, 7, 8, 9, 12, 17**and**18**will work on the same database**Animals**»container**05**classes each of which contains**06**animal breed classes (**30 classes in all**). To test the engine, you will have to make these requests:

Query index	Class	Pictures
R1, R2, R3	**Pisces**	3_4_poissons_eagleray_3310, 3_5_poissons_hammerhead_3495, 3_3_poissons_tigershark_3244
R4, R5, R6	**Dogs**	1_2_dogs_boxer_1146, 1_4_dogs_goldenretriever_1423, 1_5_dogs_Rottweiler_1578
R7, R8, R9	**monkeys**	4_3_singes_squirrelmonkey_4082, 4_2_singes_gorilla_4004, 4_1_singes_chimpanzee_3772

The calculation results of the "indexation" descriptors should be presented according to Table 1 below:

Table 1: Indexing and Search Performance Metrics

Your best descriptors	Name(s) descriptor(s)	Indexing time (s)	Indexing time (s)	Search time average per frame(s)
Descriptor No. 01				
Descriptor No. 02				
Descriptor No. 03				

The expected results for each query should look like this:

Table 2: Search Engine Accuracy Metrics

Query index	R		P		PA		MaP	
	Top50	**Top100**	**Top50**	**Top100**	**Top50**	**Top100**	**Top50**	**Top100**
R1								
R2								
...								
R9								

You can also add a TopMax column (Max: number of images per class concerned).

Footnote 2: due to the complexity of its calculation, you will be able to reduce the resolution of the images to calculate SIFT.

II.1. Part 02: hosting the application on Cloud or Edge resource:

The goal of this part is to host your media search application (from part 1) on a Cloud or Edge resource in order to offer a service in the form of Software As A Service "**SAAS**".We you propose to follow these six (06) steps:

1. **Local "characteristic extraction" indexing:**due to the limited performance of your virtual machine (no GPU), we offer you to select your best model(s) and file from image characteristics before copying them to your virtual machine. The indexing phase should therefore not not be hosted on a cloud resource.
2. **Testing and configuring your cloud resource search application:**here it will be necessary to install and configure your virtual machine in order to test your application (part 1) on the Cloud resource.
3. **Generation of the Docker image grouping together the functionalities of your application:**here, you will have to create a Dockerfile containing the instructions needed to run your application. Note that your image will have to handle:
   - **has. Entrance :**a query image;
   - **b. Output :**the indices of the most similar images + the Recall/Precision curve.
4. **Development of a web page to facilitate access to the SAAS service:**here, it will be necessary to develop a page website (withflask Wheredjango evenphp) allowing to :
   - **has.**View project developer information & description/features of your app;
   - **b.**Launch the search application using buttons, labels, etc.
   - **vs.**Display search results: similar images (with similarity rate) + R/P curves
   - **d.**Complete Table 1 and Table 2 (to be included in your report)
5. **Access configuration:**configure access to your service using your @ IP and port number of your choice
6. **Customize your site:**according to your imagination by including a login page
7. **Flexibility of use:**allow the user to choose the method of indexing and searching

Optional :

8. **Combination of services:**use docker-compose to combine services
9. **Dimensionality reduction:**reduce the size of your descriptors without losing engine precision
10. **Cybersecurity:**analyze your site in terms of security before improving it in this sense
11. **Expansion:**host the indexing part using a small database;
12. **Scaling:**program the automatic scaling according to the load of your engine.

- **Footnote 1:** for the**part 2,**you have the option of working with an Edge resource "**Nvidia Jetson xavier** ".In this case, a map (with accessories) will be provided to the interested group and the accommodation will be done only on the Edge resource (and not on Cloud resource);
- **Footnote 2:** for the**part 1**, you have the choice between using your PCs, Google Colab or requesting access to the IG cluster (one access per group);
- **Footnote 3:** for the**part 2**, we can increase the memory capacities (storage and RAM) and calculation according to your needs. This will allow you to install all the necessary tools.

Figure 2 shows an example of hosting the image search application using an image Docker and a web page developed using php and html. You can also view thisvideo for have a simple and clear idea of the expected work.

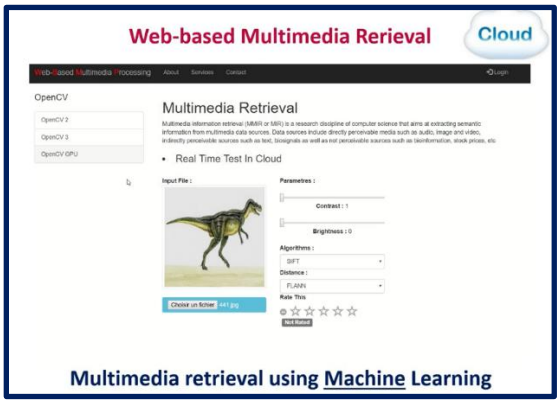

Figure 2: Media Search Application Hosting Example

III. Some interesting links:

- Example of hosting an application**C++**image processing with Docker and php: see thislink .
- Example of hosting an application**python**"Deep Learning" image classification with Docker and php: see thislink .

IV. Project sessions:reserved for the course schedule “ML & DL for Multimedia Retrieval”:

has. 12/12 from 1:30 p.m. to 4:30 p.m.

b. 19/12 from 1:30 p.m. to 4:30 p.m.

V.Contact:Sidi Ahmed Mahmoudi, Aurélie Cools, Amine Roukh and Mohamed Benkedadra