

UE: Multimedia Retrieval & Cloud Computing

Enoncé du projet

I. Introduction :

Comme annoncé en séance de cours, ce projet s'appuiera sur les connaissances acquises avec les deux activités d'apprentissage de l'UE (Fig.1) pour développer et héberger une application d'indexation et recherche multimédia sur ressources Cloud (ou Edge).

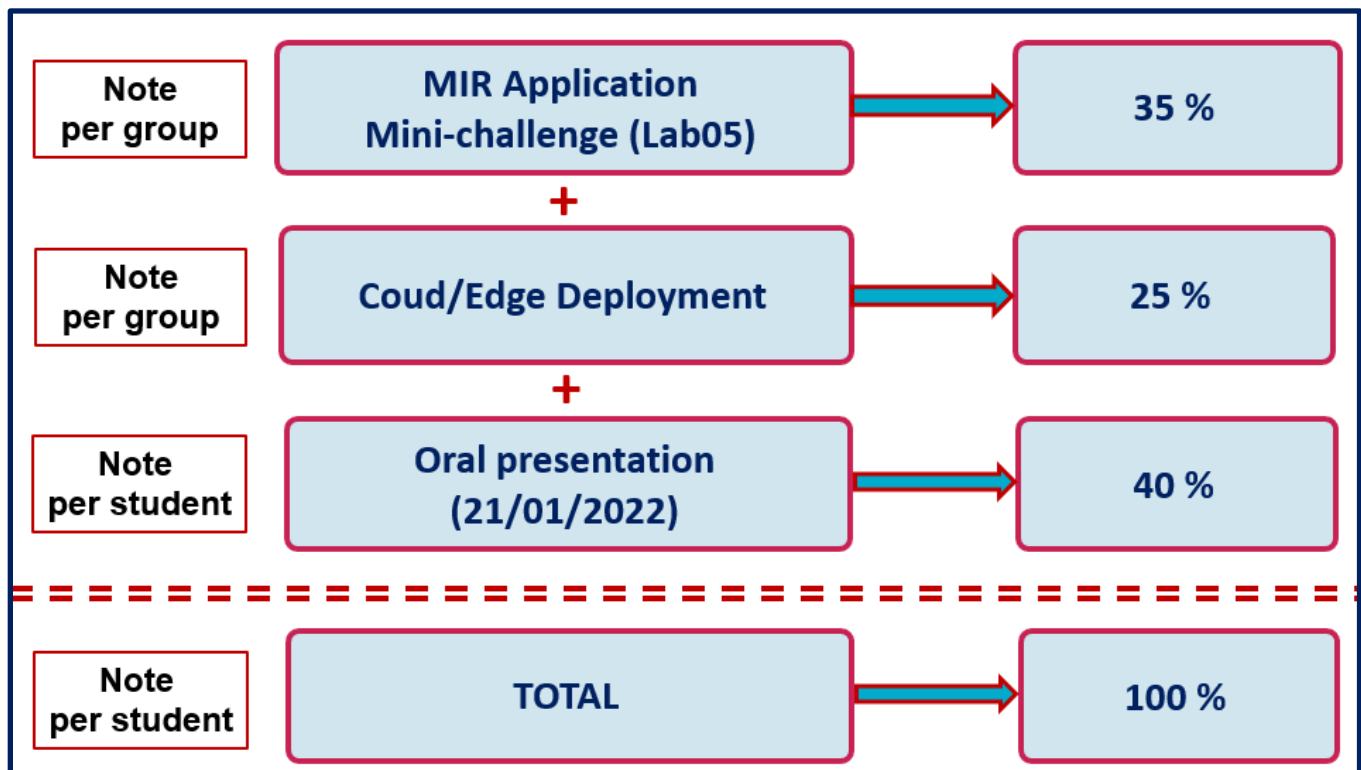


Figure 1: Modalités d'évaluation de l'UE

Les projets seront réalisés par groupe de deux avec les échéances suivantes :

- Date de remise du projet (rapport d'environ 20 pages + code + manuel) : le **15/01/2023** via Moodle
- Mode de présentation du projet : en présentiel (Ho. 23)
- Date de présentation du projet : le **20/01/2023** ou chaque groupe aura **15 minutes** au maximum pour présenter son projet suivi de **5 à 10 minutes** de questions.

Note 1 : lors de la présentation de vos projets, vous pourrez avoir des questions liées à la théorie vue au cours.

L'ordre de passage :

- **13h00 – 13h25** : Groupe 02 (Océane Mélice et Adrien Dubois)
- **13h25 – 13h50** : Groupe 03 (Selim Descamps et Antoine Minnoye)
- **13h50 – 14h15** : Groupe 04 (Valentin Corduant et Tanguy Vansnick)
- **14h15 – 14h40** : Groupe 06 (Syed Tymeur et Maxime Glosener)
- **14h40 – 15h05** : Groupe 07 (Bilal Hammas et Vivian Ruelle)
- **15h05 – 15h30** : Groupe 08 (Alan Denys et Romain Monnoyer)
- **15h30 – 15h55** : Groupe 09 (Romain Poblome et Rebacca Fotso)
- **16h05 – 16h30** : Groupe 10 (DANTAS MACEDO José António et RIBEIRO COSME Margarida)
- **16h30 – 16h55** : Groupe 11 (Simon Degouys et Yannick Basesayabo)
- **16h55 – 17h20** : Groupe 12 (Nicolas Gonze et Antoine Dhanis)
- **17h20 – 17h45** : Groupe 15 (NOUNDOU NJIKE Thierry et Florian Dubois)
- **17h45 – 18h10** : Groupe 16 (ESTIEVENART Yorick et VENTUROSOS Hugo)
- **18h10 – 18h35** : Groupe 17 (IAKOVENKO Maksym et Vandekerckhove Adrien)
- **18h35 – 19h00** : Groupe 18 (Mbenza Buanga et Patrick TEGUIA TAGNE Harold Wilfrid)

II. Enoncé du projet :

Le but du projet est de développer et héberger une application d'indexation et recherche multimédia sur ressources Cloud (ou Edge). Nous vous proposons d'utiliser vos machines virtuelles créées dans le cadre du cours de « Cloud and Edge Computing » pour héberger vos applications. Le projet comprend deux parties :

• II.1. Partie 01 : application d'indexation et recherche multimédia :

L'objectif de cette partie est de développer un moteur de recherche exploitant # descripteurs, il faudra :

1. Indexer la base de données avec les descripteurs de votre choix. Si plusieurs descripteurs sont choisis, il faudra donner la possibilité de les combiner ;
2. Réaliser la recherche en donnant la possibilité de choisir la fonction de calcul de similarité (Euclidéenne, Corrélation, Chi-square, Bhattacharyya, Brute Force Matcher, Flann, etc.) ;
3. Afficher le Top20 et Top50 pour les images requêtes ;
4. Calculer le Rappel (R), Précision (P), Average Precision (AP), Mean Average Precision (MaP) et R-Precision

Vous avez le choix de travailler sous Python ou C++ mais ce choix devra être pris en compte dans la partie 2 qui consiste à héberger votre application sur ressource Cloud.

Les groupes **2, 3, 4, 10, 11, 15 et 16** travailleront sur la base de données « **Animaux** » contenant **05** classes dont chacune contient **06** classes de races d'animaux (**30 classes en tous**). Pour tester le moteur, il faudra faire les requêtes suivantes.

- Lien de la base : https://github.com/sidimahmoudi/facenet_tf2/releases/download/AL_MIR_CLOUD/MIR_DATASETS_B.zip

| Indice requête | Classe | Images |
|----------------|-----------|--------------------------------------------------------------------------------------------|
| R1, R2, R3 | araignées | 0_5_araignees_tarantula_795, 0_4_araignees_gardenspider_631, 0_1_araignees_wolfspider_259 |
| R4, R5, R6 | Chiens | 1_0_chiens_Siberianhusky_849, 1_3_chiens_Chihuahua_1315, 1_1_chiens_Labradorretriever_1054 |
| R7, R8, R9 | Oiseaux | 2_2_oiseaux_greatgreyowl_2092, 2_4_oiseaux_robin_2359, 2_3_oiseaux_bluejay_2232 |

Les groupes **6, 7, 8, 9, 12, 17 et 18** travailleront sur la même base de données « **Animaux** » contenant **05** classes dont chacune contient **06** classes de races d'animaux (**30 classes en tous**). Pour tester le moteur, il faudra faire ces requêtes :

| Indice requête | Classe | Images |
|----------------|----------|----------------------------------------------------------------------------------------|
| R1, R2, R3 | Poissons | 3_4_poissons_eagleray_3310, 3_5_poissons_hammerhead_3495, 3_3_poissons_tigershark_3244 |
| R4, R5, R6 | Chiens | 1_2_chiens_boxer_1146, 1_4_chiens_goldenretriever_1423, 1_5_chiens_Rottweiler_1578 |
| R7, R8, R9 | Singes | 4_3_singes_squirrelmonkey_4082, 4_2_singes_gorilla_4004, 4_1_singes_chimpanzee_3772 |

Les résultats de calcul des descripteurs « indexation » devront se présenter selon le tableau 1 ci-dessous :

Tableau 1: Mesures de performances d'indexation et recherche

| Vos meilleurs descripteurs | Nom de(s) descripteur(s) | Temps d'indexation (s) | Temps d'indexation (s) | Temps de recherche moyen par image (s) |
|----------------------------|--------------------------|------------------------|------------------------|----------------------------------------|
| Descripteur N° 01 | | | | |
| Descripteur N° 02 | | | | |
| Descripteur N° 03 | | | | |

Les résultats attendus pour chaque requête devront se présenter comme suit :

Tableau 2: Mesures de précision du moteur recherche

| Indice requête | R | | P | | AP | | MaP | |
|----------------|-------|--------|-------|--------|-------|--------|-------|--------|
| | Top50 | Top100 | Top50 | Top100 | Top50 | Top100 | Top50 | Top100 |
| R1 | | | | | | | | |
| R2 | | | | | | | | |
| ... | | | | | | | | |
| R9 | | | | | | | | |

Vous pouvez également ajouter une colonne TopMax (Max : nombre d'image par la classe concernée).

Note 2 : en raison de la complexité de son calcul, vous pourrez réduire la résolution des images pour calculer SIFT.

II.1. Partie 02 : hébergement de l'application sur ressource Cloud ou Edge :

L'objectif de cette partie est d'héberger votre application de recherche multimédia (de la partie 1) sur une ressource Cloud ou Edge afin d'offrir un service sous forme de Software As A Service « **SAAS** ». Nous vous proposons de suivre ces six (06) étapes :

1. **Indexation « extraction de caractéristiques » en local** : en raison des performances limitées de votre machine virtuelle (pas de GPU), nous vous proposons de sélectionner votre meilleur modèle (s) et fichier de caractéristiques d'images avant de les copier vers votre machine virtuelle. La phase d'indexation ne doit donc pas être hébergée sur ressource cloud.
2. **Test et configuration de votre application de recherche sur ressource Cloud** : ici, il faudra installer et configurer votre machine virtuelle afin de tester votre application (partie 1) sur la ressource Cloud.
3. **Génération de l'image Docker regroupant les fonctionnalités de votre application** : ici, il faudra créer un Dockerfile regroupant les instructions nécessaires pour faire fonctionner votre application. Notons que votre image devra gérer :
 - a. **En entrée** : une image requête ;
 - b. **En sortie** : les indices des images les plus similaires + la courbe de Rappel/Précision.
4. **Développement d'une page Web pour faciliter l'accès au service SAAS** : ici, il faudra développer une page Web (avec [flask](#) ou [django](#) voire [php](#)) permettant de :
 - a. Afficher les informations des développeurs du projet & description/fonctionnalités de votre application ;
 - b. Lancer l'application de la recherche à l'aide de boutons, labels, etc.
 - c. Afficher les résultats de la recherche : images similaires (avec taux de similarité) + courbes R/P
 - d. Compléter les tableau 1 et tableau 2 (à inclure dans votre rapport)
5. **Configuration d'accès** : configurer l'accès à votre service à l'aide de votre @ IP et numéro de port au choix
6. **Personnaliser votre site** : selon votre imagination en incluant une page de connexion
7. **Flexibilité d'utilisation** : permettre à l'utilisateur de choisir la méthode d'indexation et recherche

Facultatif :

8. **Combinaison de services** : utiliser docker-compose pour combiner les services
9. **Réduction de dimensionalité** : réduire la taille de vos descripteurs sans perdre en précision du moteur
10. **Cybersécurité** : analyser votre site en termes de sécurité avant de l'améliorer dans ce sens
11. **Extension** : héberger la partie indexation en utilisant une petite base de données ;
12. **Mise à l'échelle** : programmer la mise à l'échelle automatique en fonction de la charge de votre moteur.

- **Note 1 :** pour la **partie 2**, vous avez la possibilité de travailler avec une ressource Edge « [Nvidia Jetson Xavier](#) ». Dans ce cas, une carte (avec accessoires) sera fournie au groupe intéressée et l'hébergement se fera uniquement sur la ressource Edge (et pas sur ressource Cloud) ;
- **Note 2 :** pour la **partie 1**, vous avez le choix entre utiliser vos PC, Google Colab ou demander l'accès au cluster IG (un accès par groupe) ;
- **Note 3 :** pour la **partie 2**, on pourra augmenter les capacités de mémoire (stockage et RAM) et de calcul selon vos besoins. Ceci vous permettra d'installer tous les outils nécessaires.

La figure 2 illustre un exemple d'hébergement de l'application de recherche d'images en utilisant une image Docker et une page Web développée à l'aide de php et html. Vous pouvez également visualiser cette [vidéo](#) pour avoir une idée simple et claire du travail attendu.

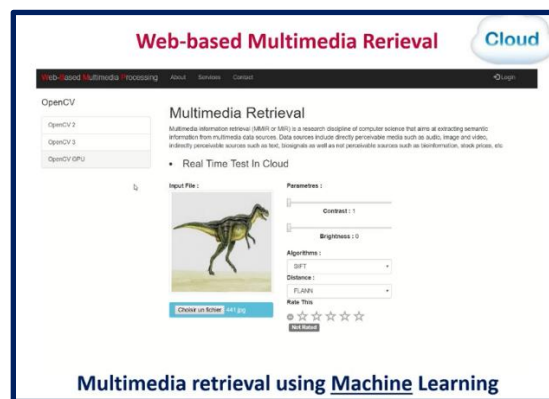


Figure 2: exemple d'hébergement d'application de recherche multimédia

III. Quelques liens intéressants :

- Exemple d'hébergement d'une application **C++** de traitement d'images avec Docker et php : voir ce [lien](#).
- Exemple d'hébergement d'une application **python** de classification d'images « Deep Learning » avec Docker et php : voir ce [lien](#).

IV. Séances Projet : réservées à l'horaire du cours « ML & DL for Multimedia Retrieval » :

- 12/12 de 13h30 à 16h30
- 19/12 de 13h30 à 16h30

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