

INF8770

Technologies multimédias

A2023 - Travail pratique #3

Indexation de contenu pictural

Remise du travail :

- Au plus tard, le 27 novembre 2023, 9h sur Moodle - **aucun retard accepté**

Documents à remettre :

- Votre code source, un rapport **PDF** ainsi que deux fichiers **CSV** (cf q4) dans une archive zippée.
- Ne remettez pas les données.
- Vous devez remettre votre rapport sur www.turnitin.com. Les instructions sont sur Moodle.

Autres directives :

- Il vous est interdit d'utiliser du code écrit par d'autres équipes.
- Il vous est permis d'utiliser du code trouvé sur internet et des librairies externes, mais vous devez nous donner toutes les références.
- Le code lui-même n'est pas évalué, mais il doit être remis au complet.

Pénalités éventuelles :

- Référence non citée (**-0.5 point**);
- Non remise du code (**-0.5 point**);
- Non remise du rapport sur Turnitin (**-0.5 point**);
- Le rapport n'est pas sous le format pdf (**-0.5 point**);
- Trop de chiffres significatifs dans les tableaux/figures (**-0.5 point**);
- Le document remis sur Moodle ne correspond pas exactement à celui remis sur Turnitin (**-0.5 point**)

Préambule : Ce travail pratique inclut une évaluation des qualités 2 et 5 du Bureau canadien d'agrément des programmes de génie (BCAPG).

- **Qualité 2 Analyse de problèmes** : capacité d'utiliser les connaissances et les principes appropriés pour identifier, formuler, analyser et résoudre des problèmes d'ingénierie complexes et en arriver à des conclusions étayées.
- **Qualité 5 Utilisation d'outils d'ingénierie** : capacité de créer et de sélectionner des techniques, des ressources et des outils d'ingénierie modernes et de les appliquer, de les adapter et de les étendre à un éventail d'activités simples ou complexes, tout en comprenant les contraintes connexes.

Travail à réaliser : Dans ce TP, vous devez implémenter un algorithme de recherche de vidéos : à partir d’une image et d’une banque de données, vous devez retrouver la vidéo dans laquelle cette image est extraite. L’algorithme implémenté devra répondre à certains critères de performances.

Objectif de l’algorithme de recherche de vidéos

Étant donnée une image, vous devez retrouver la vidéo dans laquelle cette image est extraite. Si vous trouvez une vidéo candidate, vous devez également retourner le moment où cette trame apparaît (en seconde). Si au contraire, aucune vidéo ne semble être candidate, aucun moment n’est retourné. La figure 1a illustre l’approche générale de l’algorithme.

Description des données

La banque de vidéos contient 100 vidéos en HD (1280×720), sous le format MP4, libres de droit, extraites du site www.pexels.com (ces vidéos sont disponibles sur Moodle). En plus de ces vidéos, vous trouverez 600 images (réparties sur deux dossiers) qui seront les requêtes pour la recherche ainsi qu’un fichier `test_gt.csv` qui indique pour chaque image le nom de la vidéo dans laquelle elle est extraite et son minutage en secondes (dans le cas où l’image provient d’une vidéo absente de la banque de données, ces deux valeurs sont respectivement à “out” et à vide). Les images sous le format PNG ont été extraites sans perte des vidéos et celles sous le format JPEG ont subi une perte.

Organisation des dossiers

Hormis le dossier contenant les vidéos, deux dossiers contenant les images sont disponibles :

- **test** où le fichier `test_gt.csv` contenant les solutions est disponible. Vous devriez établir votre algorithme (méthode et choix des paramètres) à partir des résultats obtenus sur ce dossier, lors des questions 2 et 3 ;
- **challenge** où aucune solution n’est fournie. Ce dossier ne doit être utilisé que pour répondre à la question 4.

La figure 1b illustre l’organisation des dossiers.

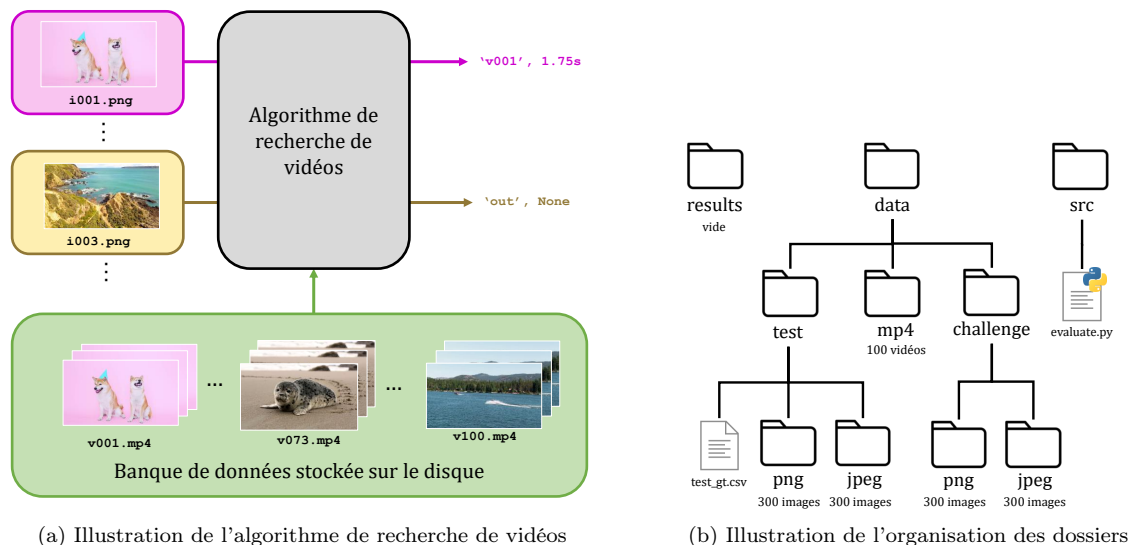


FIGURE 1 : Organisation des fichiers et algorithme de recherche

Format du fichier test_gt.csv

Le fichier contient trois colonnes :

- **image** : le nom de l'image (i001.png et i001.jpeg correspondent à la même trame de la même vidéo) ;
- **video** : le nom de la vidéo si elle est présente dans le dossier des vidéos, sinon "out" ;
- **minutage** : l'instant d'apparition de la trame dans la vidéo (en seconde) si cette vidéo est présente, sinon vide. La première trame de chaque vidéo correspond au minutage 0 seconde.

Mesures de performance

Les deux critères d'évaluation, mesurés sur les PNG et les JPEG, sont les suivantes :

1. le pourcentage de vidéos correctement retrouvées (en %) ;
2. l'écart (en valeur absolue) moyen sur le minutage lorsque la vidéo est correctement retrouvée et présente dans la banque de données (en seconde) ;

Pour mesurer vos performances, enregistrez vos résultats dans un fichier `results/test_png.csv` puis lancez la commande suivante sur les images PNG (idem pour les JPEG) :

```
python evaluate.py --file=../results/test_png.csv --file_gt=../data/test/test_gt.csv
```

Les fichiers de résultats doivent comporter les trois entêtes suivants : **image** qui contient le nom de l'image ('i001'), **video_pred** qui contient le nom de la vidéo candidate ('v001') et **minutage_pred** le minutage correspond (1.75).

En termes de temps d'exécution, nous vous conseillons de modifier votre algorithme si vous constatez qu'il est lent (plus d'une seconde par image en moyenne). Lors du mesure du temps d'exécution, ne comptez pas le temps de chargement des données et/ou modèles. Les requêtes sur les images ne doivent pas être parallélisées.

Question 1 (/4)

Alice, Bob et Carol sont trois étudiants n'ayant pas suivi le cours INF8770. Ils proposent cependant trois algorithmes permettant de retrouver une vidéo à partir d'une image.

Alice propose de stocker les vidéos sous le format MP4 telles quelles sans traitement particulier. Puis, étant donnée une image, elle propose de parcourir toutes les vidéos, trame par trame, et de calculer une distance euclidienne entre les pixels de l'image d'intérêt et la trame de la vidéo. Dès que la distance est nulle, elle retourne la vidéo en question ainsi que le minutage. Si la distance nulle n'est jamais atteinte, elle renvoie un message indiquant que la vidéo n'a pas été trouvée.

Bob propose de stocker les vidéos sous la forme d'images car il pense que cela accélère le traitement compte tenu qu'il s'agit de comparer des images avec d'autres images. Pour cela, il hésite entre les deux formats PNG et JPEG. Pour la suite, il reprend les idées d'Alice : il propose de parcourir toutes les trames extraites et de calculer une distance euclidienne avec l'image d'intérêt. Dès que la distance est nulle, il retourne la vidéo associée à la trame ainsi que le minutage. Si la distance nulle n'est jamais atteinte, il renvoie un message indiquant que la vidéo n'a pas été trouvée.

Carol propose de sélectionner une trame à chaque seconde de chaque vidéo et de la décrire par un histogramme de couleurs (un fichier d'indexation additionnel peut être utilisé). Puis, étant donnée une image, Carol calcule son histogramme de couleurs et retourne la vidéo et le minutage de l'histogramme de couleurs ayant la plus faible distance euclidienne.

D'après vos connaissances en technologies multimédias, discutez de la pertinence de chaque algorithme lors de l'application sur les images en PNG et en JPEG. Quelles recommandations feriez-vous à Bob quant au format de l'image dans son algorithme ?

- **Qualité évaluée : 2.1 Identifier et formuler un problème**

Critère d'évaluation : Analyse des algorithmes présentés par rapport aux deux mesures de performance. Justification du format d'image approprié pour l'algorithme de Bob.

Question 2 (/4)

Implémentez l'un des trois algorithmes proposés et appliquez le sur les images du dossier **test**. Indiquez le temps moyen d'exécution par image ainsi que les performances sur les images en PNG et en JPEG. Est-ce que les résultats en termes de performance sont conformes à vos remarques pour la question 1 ?

NB1 : si vous choisissez l'algorithme de Bob, prenez le format d'images qui vous semble le plus pertinent.

NB2 : si vous choisissez l'algorithme de Carol, vous êtes libres quant aux choix des paramètres de l'histogramme des couleurs.

- **Qualité évaluée : 5.3 Créer ou adapter un outil**

Critère d'évaluation : Capacité à créer un code informatique fonctionnel.

- **Qualité évaluée : 5.4 Intégrer des outils**

Critère d'évaluation : Capacité à bien intégrer les fonctions de bibliothèques de traitement d'images/vidéos dans son propre code.

Question 3 (/12)

Implémentez votre propre algorithme de recherche de vidéos sur les images du dossier **test**. Expliquez votre méthodologie. Discutez de vos choix d'implémentation (sélection d'images clefs, format des données sauvegardées, stratégie de parcours des vidéos, caractérisation d'une image, choix des paramètres sur la base de **test**, etc). L'algorithme devra traiter les images PNG et JPEG de la même manière (en particulier, les paramètres devront être identiques).

Indiquez le temps moyen d'exécution par image ainsi que les performances sur les images en PNG et en JPEG. Comparez vos mesures de performances avec la méthode implémentée à la question 2 vis-à-vis des deux mesures de performances sur les images en PNG et en JPEG du dossier **test**. Synthétisez vos résultats dans un tableau ou un graphique. Discutez des limites de votre algorithme.

- **Qualité évaluée : 2.2 Explorer des approches de résolution et planifier la démarche**

Critère d'évaluation : Justification des choix de votre algorithme au regard des limites rencontrées par ceux décrits à la question 1.

- **Qualité évaluée : 2.4 Produire des résultats**

Critère d'évaluation : Pertinence des résultats et du protocole pour démontrer la qualité de la solution proposée.

- **Qualité évaluée : 2.5 Valider ses résultats et recommander**

Critère d'évaluation : Qualité et exhaustivité de l'analyse des résultats.

- **Qualité évaluée : 2.6 Analyser l'incertitude, la sensibilité et les limites des approches**

Critère d'évaluation : Qualité des explications sur les limites de la méthode.

Question 4 (BONUS d'au plus 2/20, la note maximale étant 20/20 pour le TP)

Appliquez votre algorithme développé à la question 3 sur la base de données **challenge** et retournez deux fichiers CSV (tout autre format sera considéré invalide) nommés **challenge_png.csv** et **challenge_jpeg.csv** portant respectivement sur les images du dossier **challenge/png** et **challenge/jpeg**. Les fichiers de résultats doivent comporter les trois entêtes suivants : **image**, **video_pred** et **minutage_pred**.

Un bonus d'au plus 2 points sera accordé aux quatre équipes ayant les meilleurs résultats sur les 600 images du dossier **challenge**.