

INF6804 Vision par ordinateur

H2020 – Travail Pratique 1

Segmentation vidéo

Objectifs:

- Se familiariser à la manipulation de données et au traitement de séquences d'images à l'aide de bibliothèques logicielles spécialisées pour la vision par ordinateur.
- Se familiariser et expérimenter avec les algorithmes de segmentation basés sur le flot optique.
- Se familiariser et expérimenter avec les algorithmes de détection par classification.
- Expliquer les avantages et les inconvénients des deux approches.

Remise:

- *Tout votre code source* (nous devrions pouvoir exécuter vos tests)
- Un rapport (*.pdf format* de 8 à 15 pages avec une taille de police de 10)
- Remise avant le 7 février 2020, 17h00, sur Moodle –*aucun retard accepté*
- *Vous devez aussi soumettre votre rapport sur TurnItIn!*
 - Enregistrez-vous sur www.turnitin.com en utilisant les informations sur Moodle!

Références:

- Voir les notes de cours sur Moodle (Chapitre 2)

Autres directives:

- Les TP's doivent obligatoirement être faits en équipe de deux, ne remettez qu'une seule version du rapport/code!
-

Présentation

Dans ce TP, vous devez caractériser deux méthodes d'extraction de régions d'intérêt dans une séquence vidéo et déterminer quelle méthode est la meilleure et dans quelles circonstances. Une description de votre travail, de vos expériences, et les réponses aux questions de cet énoncé doivent se retrouver dans un rapport.

Lors de cette séance, vous devrez comparer deux méthodes de segmentation avant-plan/arrière-plan de vidéos, soit une technique de flot optique et une technique de détection par classification. Celles-ci devraient être vues brièvement en classe — vous pouvez utiliser vos notes de cours comme référence pour comprendre leur fonctionnement de base. Pour plus de détails, allez creuser sur internet!

Dans votre rapport, vous devez inclure les éléments suivants (noté sur 20 pts):

1. Présentation des deux méthodes (4 pts):
Dans vos propres mots, donnez la description générale et les principes de vos deux méthodes.
2. Hypothèses de performance dans cas spécifiques (3 pts):
Identifier, selon votre compréhension théorique des deux techniques de segmentation étudiées uniquement, quelle approche devrait être meilleure que l'autre en fonction d'au moins TROIS cas d'utilisation. Par exemple, quelle est la meilleure approche à utiliser si la caméra n'est pas fixe? Pourquoi? Et si le contraste est faible?
3. Description des expériences, bases de données et critères d'évaluation (2 pts):
Décrire en détail les expériences réalisées pour vérifier vos hypothèses du point précédent. Quelles bases de données avez-vous utilisées? Quelles sont les difficultés de ces séquences? Quels sont vos critères d'évaluation?
4. Description des deux implémentations utilisées (2 pts):
Décrire l'implémentation des deux approches étudiées. Si vous n'avez pas tout écrit le code vous-même, d'où provient-il? A-t-il demandé des modifications? Sinon, de quel articles (ou site web) vous êtes-vous inspiré pour l'écrire? Dans tous les cas, quels sont les paramètres principaux utilisés? Comment ont-ils été choisis?
5. Présentation des résultats de tests (3 pts):
Donner les résultats d'évaluation de performance tirés de vos expériences en lien avec les hypothèses du premier point. Utilisez un format approprié — tableaux, figures, ...
6. Discussion des résultats et retour sur hypothèses (3 pts):

Discuter des résultats du quatrième point en fonction des hypothèses du premier point. Quelles hypothèses sont supportées par les résultats? Lesquelles ne le sont pas? Quels tests sont restés sans conclusion? Quels tests pourriez-vous améliorer, et comment?

7. Lisibilité et complétude (3 pts):

À part le contenu, le format doit être soigné et complet.

Vous pouvez utiliser n'importe quelle base de données pour ce travail, mais si vous n'en connaissez pas, nous vous suggérons d'utiliser la base de données CD-NET.

Lors des séances de laboratoire, n'hésitez pas à poser des questions aux chargés — ils peuvent vous aider avec tout problème technique si vous travaillez sur Windows/Linux, ou bien avec votre code si vous travaillez avec du C/C++ ou avec Python/Matlab.

Vous serez pénalisés de 50% de la note totale si vous ne remettez pas votre code. De plus, si votre rapport n'est pas remis sur TurnItIn, celui-ci ne sera pas corrigé. L'ordre de la présentation n'est pas important, tant que tous les éléments ci-haut sont présents. N'oubliez pas de citer vos références!

Ressources

Ensemble de données:

- CDNET dataset (<http://changedetection.net/>)

Librairies de vision par ordinateur:

- OpenCV (https://docs.opencv.org/4.0.0/d9/df8/tutorial_root.html)
- scikit-image (https://scikit-image.org/docs/stable/auto_examples/index.html)

Librairies d'apprentissage profond:

- PyTorch (<https://pytorch.org/tutorials/>)
- Tensorflow (<https://www.tensorflow.org/tutorials>)

Python:

- Guide (<https://wiki.python.org/moin/BeginnersGuide/Programmers>)
- NumPy (<https://docs.scipy.org/doc/numpy/user/quickstart.html>)

Matlab:

- Guide (http://www.mathworks.com/help/pdf_doc/matlab/getstart.pdf)
- Aide-mémoire (<http://web.mit.edu/18.06/www/Spring09/matlab-cheatsheet.pdf>)