

TD VISION - ESIEE 4E SI4

Cyril Meurie

2024

1 Environnement de travail PANDORE

PANDORE est une bibliothèque standardisée d'opérateurs de traitement d'images développée par le laboratoire GREYC à Caen et disponible en version Unix, Linux, MacOS et Windows à l'adresse internet suivante : <https://clouard.users.greyc.fr/Pandore/index-fr.html>. Cette bibliothèque est libre de tout droit d'utilisation mais sans être du domaine public. Elle s'adresse tout particulièrement aux traiteurs d'images puisqu'elle nécessite des connaissances sur les opérations de traitement d'images et la façon de les combiner afin de répondre à un problème donné. La version actuelle (v6.6.10 du 10/01/2021) regroupe des opérateurs traitant d'images 1D, 2D, 3D, en niveaux de gris, en couleurs et multi-spectrales.

Pour travailler dans PANDORE, il faut lancer un Terminal et saisir les commandes à la main. Nous verrons plus loin, qu'il peut être intéressant d'enchaîner plusieurs commandes à l'aide d'un script Shell ou d'utiliser l'interface ARIANE.

2 Installation et configuration de la bibliothèque

Dans la mesure du possible, je vous conseille d'installer la bibliothèque PANDORE sous Linux (il est possible de l'installer dans une machine virtuelle si vous le souhaitez car cela ne requiert pas trop de ressource) mais l'installation sous Windows et Mac est aussi possible. Vous trouverez la procédure d'installation à l'adresse internet suivante : <https://clouard.users.greyc.fr/Pandore/installation-fr.html>. Il existe des versions précompilées, mais il est préférable de recompiler les sources pour s'assurer que la compilation fonctionne correctement quand sera venu le moment de créer ses propres opérateurs. J'ajoute que sous Linux, il est souhaitable d'avoir installé le paquet `qtbase5-dev-tools` au préalable pour bénéficier de la fonctionnalité de visualisation des images au travers de l'opérateur `pvisu`. Enfin, pour plus de facilité dans l'utilisation de la bibliothèque, pensez à enrichir votre `.bashrc` ou `.bashrc.linux` en ajoutant les trois lignes suivantes :

```
- export PANDOREHOME=/opt/pandore
- export PATH=/opt/pandore/bin:$PATH
- export LD_LIBRARY_PATH=/opt/pandore/lib:$LD_LIBRARY_PATH
```

Pour celles et ceux qui ne souhaiteraient pas installer PANDORE sur leur machine, une VM Ubuntu pré-installée est à votre disposition avec comme nom d'utilisateur et mot de passe : `adminrv`. Le lien de téléchargement (protégé par le mot de passe : `tdVision2021.meurie`) est le suivant : <https://cloud.univ-eiffel.fr/s/WPiiWNegRaKedFs>.

3 Premiers pas avec les opérateurs de traitement d'images

PANDORE manipule des images dans un type qui lui est propre (`.pan`). Il peut néanmoins effectuer de multiples conversions de formats d'image (BMP, GIF, TIFF, RAW, EPS, etc). Nous utiliserons principalement des images BMP ou PNG dont la transformation se fera pour le premier format à l'aide de la fonction `pbmp2pan`. PANDORE dispose d'un utilitaire de visualisation des images. Il s'agit de l'opérateur `pvisu`. A titre d'illustration, exécutez les premières commandes suivantes :

- pbmp2pan car.bmp car.pan
- pvisu car.pan

Utiliser les fonctions disponibles dans le menu de pvisu pour visualiser sur l'image car.bmp les points suivants :

- L'(es) histogramme(s) de l'image, le profil le long d'une ligne
- Les niveaux de gris dans un carré (dans du noir, du blanc, du jaune ou à l'intersection)
- Le zoom de l'image
- Le changement de palette d'affichage

4 Exercice de classification de régions d'intérêt

Nous allons maintenant nous familiariser avec la bibliothèque de traitement d'images PANDORE, en répondant à des besoins limités (simplification, transformation couleur/niveaux de gris, changement d'espace couleur, classification pixellaire, segmentation régions/contours) mais utiles pour des problèmes plus complets. En vous aidant de l'index des opérateurs disponible à l'adresse suivante : https://clouard.users.greyc.fr/Pandore/programmes/fr/index_operatorsP0.html, effectuez les opérations suivantes en visualisant et analysant chaque étape :

- Décomposez l'image environnement.png en trois composantes : rouge.pan, vert.pan, bleu.pan
- Tester l'opérateur pfile pour connaître les caractéristiques de votre image

Les opérations par voisinage local sont des opérateurs qui, pour changer la valeur d'un pixel, utilisent les valeurs des pixels voisins. Pour des images 2D en trame carrée, deux types de voisinage sont considérés : le 4-connexité et le 8-connexité. Utiliser les opérateurs disponibles pour effectuer sur l'une des composantes de l'image environnement.bmp, les fonctions suivantes :

- Le filtre moyennneur linéaire
- filtre moyennneur de Nagao
- Le filtre de lissage par médian
- Le filtre de lissage gaussien

Le seuillage cherche à séparer l'histogramme des niveaux de gris en deux ou plusieurs parties homogènes. Considérons que l'image environnement.png possède deux classes d'intérêt à savoir le ciel et le reste. Nous pouvons donc raisonnablement penser qu'une séparation en deux parties peu suffire et permettre de réaliser les opérations suivantes :

- Observez après filtrage, l'histogramme de l'image environnement.png et déterminez vous-même une valeur de seuillage acceptable
- Seuillez l'image entre 0 et votre valeur et visualisez l'image produite.
- Appliquez l'opérateur de multi-seuillage par analyse de la variance interclasse
- Visualisez l'image initiale, l'image produite avec la valeur de seuillage calculée.

Pour aller un peu plus loin, nous pouvons considérer que l'image environnement-vegetation.png possède trois classes d'intérêt : le ciel, la végétation et le reste. Nous pouvons donc cette fois réaliser une séparation en trois classes mais réalisons le de manière automatique sans devoir choisir un seuil qui pourrait être différent pour d'autre images de la même séquence.

- Transformez l'image environnement-vegetation.png au format PANDORE

- Observez son histogramme original
- Appliquer l'opérateur pfisher et visualisez l'image produite
- Créez une image binaire ne conservant que les pixels des classes ciel et végétation

Le changement d'espace de représentation couleur peut permettre de mieux détecter et/ou classer les objets d'intérêt dans une image. Pour vérifier cela, nous allons maintenant appliquer un changement d'espace couleur sur l'image environnement-vegetation.png et réitérez la classification pixellaire :

- Transformer l'image couleur environnement-vegetation.png en niveau de gris à l'aide de l'opérateur prgb2gray
- Changez l'espace couleur de l'image environnement-vegetation.png de Rouge-Vert-Bleu à Teinte-Saturation-Luminance
- Changez l'espace couleur de l'image environnement-vegetation.png vers l'espace de luminance-chrominance $L^*a^*b^*$
- Appliquer de nouveau l'opérateur pfisher et visualisez les changements de classification opérés

5 Premier script Shell avec PANDORE

Pour répondre à un problème plus global, il est nécessaire d'enchaîner plusieurs opérateurs. La solution généralement retenue est la création d'un script shell permettant de (re)lancer l'intégralité d'une procédure en une seule fois et sur un ensemble d'images. Dans l'exemple qui suit, vous créerez donc un script traite.sh permettant d'exécuter les fonctions suivantes :

- Considérer l'image car.bmp
- Pour alléger les traitements, redimensionner car.bmp en 400 pixels * 266 pixels
- Décomposez l'image car.pan en trois composantes : rouge, vert, bleu
- Seuiller l'image de la composante verte pour éliminer les reflets dans la vitre/façade
- Composer une image binaire à partir de cette image et de cette valeur de seuil
- Adapter le script pour qu'il traite également les deux autres composantes couleur
- Convertir les images résultantes en .bmp

6 L'enchaînement d'opérateurs avec ARIANE

ARIANE est un environnement de programmation visuelle par flots de données permettant de programmer graphiquement des applications par simple sélection et enchaînement d'opérateurs (de la bibliothèque PANDORE) représentés dans l'interface par des boîtes connectables. L'utilisateur sélectionne des opérateurs parmi la liste proposée, puis les enchaîne pour former des graphes de flots de données. Les sorties des uns sont alors utilisées comme entrées des autres. Le traitement complet d'une image consiste en l'exécution séquentielle des opérateurs dans l'ordre prescrit par le graphe. La version actuelle d'Ariane (v3.5.0 du 22/08/2018) est téléchargeable à l'adresse suivante : <https://clouard.users.greyc.fr/Ariane/index-fr.html>

Afin de vous familiariser avec cette interface de programmation visuelle qu'est ARIANE, vous allez reproduire le graphe présenté ci-après (Figure ??). Pour ce faire, vous devez respecter les indications énoncées ci-dessous.

- Ouvrir un terminal et lancer l'interface se situant dans `/opt/ariane`

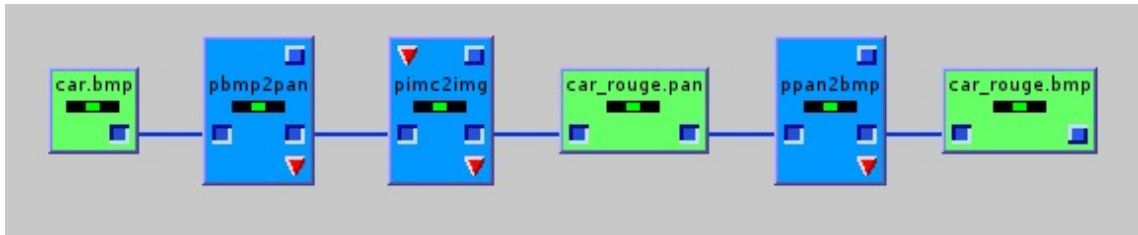


Figure 1: Mon premier graphe de flots de données

- Créer un nouveau graphe Fichier puis Nouveau
- Glisser/Déposer une structure de contrôle de type loader dans la fenêtre de droite
- Double-cliquer sur l'opérateur pour sélectionner un fichier de données car.bmp
- Glisser/Déposer l'opérateur de conversion pbmp2pan dans la fenêtre de droite
- Glisser/Déposer l'opérateur de casting pimc2img dans la fenêtre de droite
- Double-cliquer sur l'opérateur pour régler ses paramètres (0 pour la composante rouge)
- Glisser/Déposer une structure de contrôle de type viewer dans la fenêtre de droite
- Glisser/Déposer l'opérateur de conversion ppan2bmp dans la fenêtre de droite
- Glisser/Déposer une structure de contrôle de type saver dans la fenêtre de droite
- Double-cliquer sur l'opérateur pour renseigner la sauvegarde car_rouge.bmp
- Relier les entrées/sorties par cliquer-glisser d'une sortie vers une entrée
- Exécuter le graphe en ayant sélectionné au préalable le composant saver

7 PANDORE et la création d'opérateurs personnels

Nous allons maintenant répondre à des problèmes concrets plus complexes, en enchaînant divers opérateurs existants de la bibliothèques mais en développant et compilant aussi nos propres opérateurs. Vous avez le choix d'utiliser ou non l'environnement de programmation visuelle par flots de données ARIANE puisqu'il est aussi possible d'appeler ses propres opérateurs. Le tutoriel suivant présente la façon de construire un nouvel opérateur PANDORE. L'exemple utilisé est l'opération d'érosion morphologique sur une image en niveaux de gris. On suppose ici que vous êtes sous Linux que le nouvel opérateur sera construit dans votre dossier personnel et que vous procéderez à la compilation du code (sous Linux). Afin de vérifier que la compilation s'exécute correctement sur votre poste, je vous invite à suivre la procédure ci-dessous :

- Renommer le fichier `Makefile.unix` en `Makefile` : `mv Makefile.unix Makefile`
- Compiler l'opérateur (sous linux) : `make example0`
- Exécuter le fichier sur l'image `tangram` : `./example0 8 tangram.pan sortie.pan`
- Visualiser le résultat obtenu : `pvisu sortie.pan`

8 Première application simple à résoudre

Plaçons nous à présent dans le cadre d'un système de détection d'intrusion. Une caméra de surveillance installée en extérieur et dans une pièce à haute confidentialité fournit en continu des images de la scène. Pour cet exemple, nous utiliserons une image acquise à un instant dit "sans anomalie" bg-garden.bmp ou bg-hall.bmp et une autre acquise lors d'une intrusion fg2-garden.bmp ou fg2-hall.bmp. L'objectif de cet exercice consiste à créer le graphe de flots de données (en utilisant

ARIANE/PANDORE) permettant de mettre en évidence les objets mobiles présents dans la scène. Le cours ne portant que sur une initiation en vision, nous traiterons le problème (que) de façon basique en effectuant une simple soustraction entre les deux images. Pour vous guider dans la stratégie à suivre, voici un certain nombre d'étapes énoncées ci-après que vous devez réaliser. Il vous appartient d'analyser chaque point en faisant par exemple référence à l'intérêt (ou non) de l'étape ajoutée, la composante couleur offrant les meilleurs résultats, l'espace couleur le mieux adapté, le nombre de classe utilisé dans la classification pixellaire, etc., le tout appuyé par des résultats quantitatifs.

- Réaliser un pré-traitement au travers l'utilisation d'un filtre pour éliminer le bruit présent dans l'image.
- Réaliser une soustraction de fond à l'aide de bg-garden.bmp pour extraire la silhouette de l'image fg2-garden.bmp
- Appliquer un (plusieurs) changement(s) d'espace de représentation couleur pour améliorer (ou non) les résultats.
- Appliquer un post-traitement via l'utilisation d'opérations de morphologie mathématique simples, afin d'améliorer la qualité de l'extraction de la silhouette du fond (en éliminant par exemple les petites régions de non-intérêt (feuille, branche d'arbres, zone surexposée, etc)
- Créer votre propre opérateur de colorisation qui : 1/ ré-affecte le fond en noir; 2/ ré-attribue à chaque pixel de la silhouette extraite, la couleur du pixel de fg2-garden.bmp.
- Évaluer la qualité des extractions/segmentations obtenues en utilisant deux méthodes d'évaluation sans/avec référence (par exemple MSE/Vinet en utilisant la référence binaire refb-fg2-garden.bmp ou couleur refc-fg2-garden.bmp).
- Répéter les différentes étapes sur fg3-garden.bmp et fg2-hall.bmp en donnant les ajustements réalisés ou étapes supplémentaires pour obtenir des résultats satisfaisants.