

# Fête de la science

## Initiation au traitement des images

### *Détection automatique de plaques minéralogiques à partir d'un téléphone portable*

**C**et atelier propose de créer un programme informatique pour un téléphone portable programmable permettant la détection automatique de plaques minéralogiques à partir d'images prises avec la caméra du téléphone.

Après ce programme, il est possible d'utiliser un autre programme de lecture automatique de caractères (OCR) pour lire le numéro et lancer une recherche dans une base de données pour connaître le propriétaire du véhicule ou produire automatiquement une contravention !



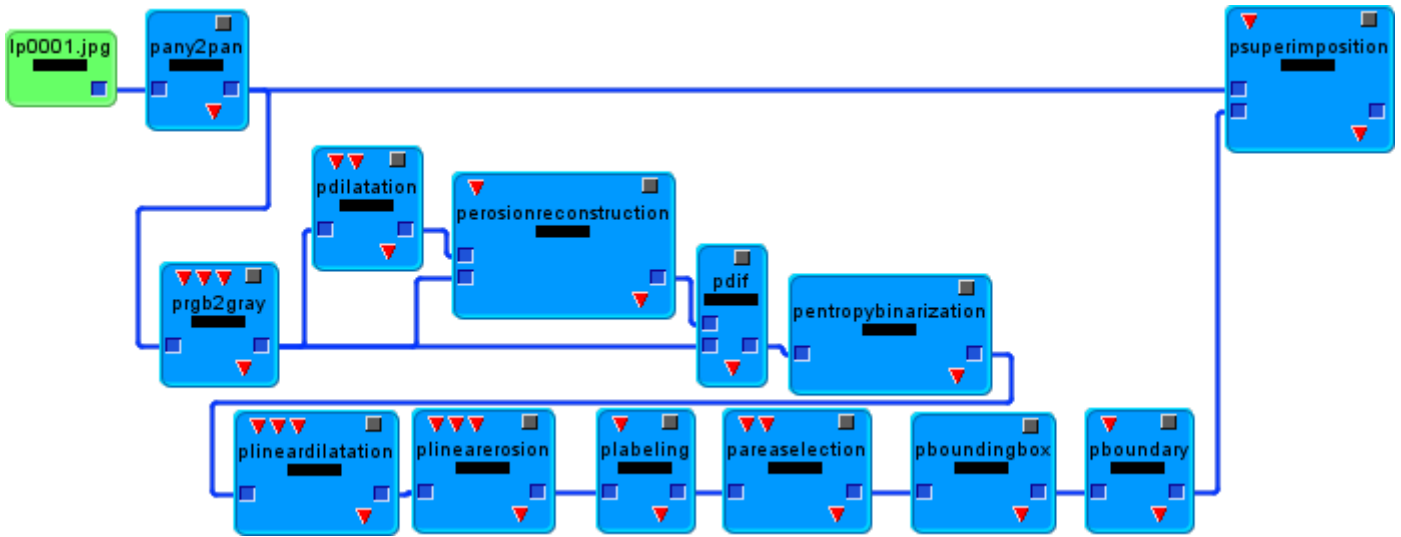
Figure 1. À gauche : l'image initiale obtenue à partir de la caméra d'un téléphone portable. À droite, l'image obtenue après traitement, où la plaque est repérée par un rectangle rouge.

Les images présentent les caractéristiques suivantes :

<b>Type</b>	Image couleur. Chaque pixel est codé sur 3 octets : (1 octet pour le rouge + 1 octet pour le vert + 1 octet pour le rouge)
<b>Taille</b>	384 colonnes × 512 lignes de pixels = $384 \times 512 \times 3 = 589824$ octets = 589 Ko sans compression

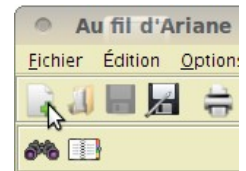
# Exercice

En suivant les étapes ci-après, vous devez réaliser un programme informatique ressemblant au graphe ci-dessous :



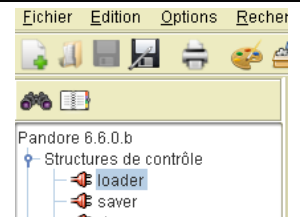
## 1/ Création d'un espace de travail

- ❑ Cliquez sur le menu **Fichier** puis **Nouveau** pour créer un espace de travail.



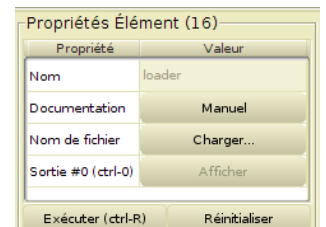
## 2/ Chargement d'une image dans l'espace de travail

- ❑ De l'explorateur situé à gauche de l'interface, déplacez la structure de contrôle **'loader'** ('chargeur' en anglais) dans l'espace de travail par un cliquer-glisser dans l'espace de travail.



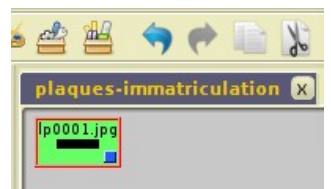
- ❑ Dans la fenêtre des propriétés à droite de l'interface, cliquez sur le bouton **'Charger...'**.

- ❑ Ouvrez le dossier **plaques-immatriculation** sur la clé ENSICAEN.
- ❑ Sélectionnez l'image **lp0001.jpg**.



- ❑ Cliquez à nouveau sur l'élément **'loader'** dans l'espace de travail.

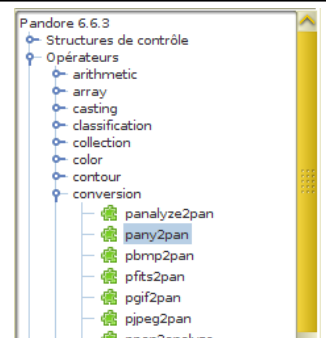
- ❑ Appuyez sur le **bouton vert d'exécution** de la barre d'outils. La signalétique au centre du loader doit passer au vert. *Si elle passe au rouge, il faut recommencer l'opération de sélection du fichier.*



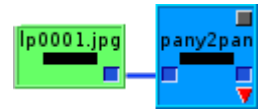
## 3/ Conversion du type de l'image

Le logiciel fonctionne avec le format d'image propriétaire *Pandore*. Il est donc nécessaire de convertir les images du format *JPEG* vers le format *Pandore*.

- ❑ Dans l'explorateur situé à gauche de la fenêtre Ariane, déplacez l'opérateur **conversion::pany2pan** sur l'espace de travail (onglet *Opérateurs* puis onglet *conversion*).
- ❑ Reliez la sortie du **loader** à l'entrée de ce nouvel opérateur en tirant une



ligne fictive entre le plot de sortie de l'opérateur **loader** et le plot d'entrée de **pany2pan** par cliquer-glisser comme indiqué dans la figure de droite.



#### 4/ Transformation de l'image couleur en niveaux de gris

**Observation :** La couleur n'est pas une information discriminante pour repérer les plaques d'immatriculation. On peut simplement considérer qu'une plaque est composée d'un fond plutôt claire et de caractères plutôt sombres. Donc, seules les nuances d'intensité lumineuse sont intéressantes.

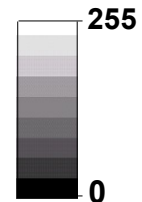
Pour créer une image d'intensité lumineuse à partir d'une image couleur, on se base sur le fait que la lumière blanche contient en même proportion les trois couleurs : rouge, vert et bleu. La valeur d'intensité d'un pixel est donc égale à la moyenne de la valeur de rouge, de vert et de bleu de ce pixel :

$$\text{intensité}(\text{pixel}) = \frac{(1 \times \text{rouge}(\text{pixel}) + 1 \times \text{vert}(\text{pixel}) + 1 \times \text{bleu}(\text{pixel}))}{3}$$

On obtient ainsi une image en niveaux de gris (*gray image* en anglais).

#### Rappel

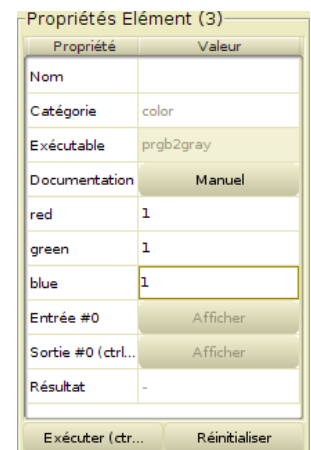
Dans une image en niveaux de gris, les pixels codent une valeur d'intensité lumineuse entre les valeurs 0 (noir) et 255 (blanc).



- ❑ Dans l'explorateur, déplacez l'opérateur **color::prgb2gray** sur l'espace de travail (onglet *opérateur* puis onglet *color*).
- ❑ Reliez **prgb2gray** à **pany2pan** comme dans le schéma à droite.



- ❑ Cliquez sur **prgb2gray**.
- ❑ Dans la fenêtre des propriétés située à droite de l'interface, mettez la valeur 1 pour les trois paramètres *red*, *green* et *blue*.
- ❑ Exécutez l'opérateur à partir du bouton vert dans la barre d'outil.
- ❑ Affichez le résultat en appuyant sur le bouton 'Afficher' de la sortie #0 dans la fenêtre de propriétés. *Vérifiez que l'image obtenue est bien en niveaux de gris.*



#### 5/ Détection de fines structures noires sur fond blanc

**Observation :** Les caractères sont de fines structures noires qui reposent sur un fond plus clair. Pour détecter ces caractères, on va procéder paradoxalement par leur suppression. Il suffira alors de faire la différence entre l'image obtenue après suppression et l'image initiale pour récupérer les caractères (*cf. exemple à droite*).

- i) **DILATATION:** Pour les supprimer, on commence par dilater les régions claires. Ceci a pour effet de faire déborder les régions claires sur les régions noires environnantes et donc de supprimer les caractères

Le principe de la dilatation consiste à remplacer chaque pixel par la valeur maximale de ses 8 plus proches voisins et de lui-même. Dans l'exemple ci-dessous la valeur 10 est remplacée par la valeur 15 :

10	11	15
10	10	14
9	15	14

$$\text{pixel}_i = \max_{j \in \text{Voisins}(i)} \{ \text{pixel}_j \}$$

10	11	15
10	15	14
9	15	14

On procède ainsi pour tous les pixels de l'image.

#### Détection des caractères



Une image de plaque.



i) après la dilatation.



ii) après l'érosion.



iii) après la différence.

### Exercice à faire sur une feuille séparée

- ❑ Calculez l'image résultant de la dilatation de la petite image ci-dessous qui représente une croix sombre sur un fond clair :

210	220	11	112	110
211	225	12	212	110
9	10	11	12	11
212	115	11	112	112

- ii) **EROSION**: La dilatation a pour effet d'augmenter la surface des régions claires. L'opération inverse (une érosion) permet aux régions claires de retrouver leur taille d'origine.

Le principe de l'érosion consiste à remplacer chaque pixel par la valeur minimale de ses 8 plus proches voisins. Dans l'exemple ci-dessous la valeur 10 est remplacée par la valeur 9 :

10	11	15
10	10	14
9	15	14

$$pixel_i = \min_{j \in \text{Voisins}(i)} \{ pixel_j \}$$

10	11	15
10	9	14
9	15	14

### Exercice à faire sur une feuille séparée (suite)

- ❑ Calculez l'image érodée de l'image que vous avez calculée au-dessus. Comparez avec l'image initiale. Conclusion sur la croix noire.

Intégrons maintenant la dilatation de l'image dans le graphe.

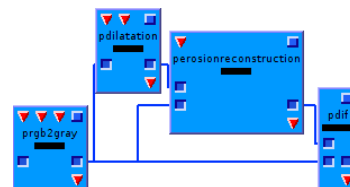
- ❑ Déplacez l'opérateur **morphology::pdilatation** sur l'espace de travail.
- ❑ Reliez son entrée à la sortie de **prgb2gray**.
- ❑ Mettez les valeurs *num\_se* = 1 et *halFSIZE* = 3 (3 signifie de prendre en compte les 48 plus proches voisins de chaque pixel;  $48 = 8^1 + 8^2 + 8^3$ ).
- ❑ Exécutez l'opérateur et visualisez l'image de sortie #0. Vérifiez que les numéros d'immatriculation ont bien disparu.

Procédons ensuite à l'érosion.

- ❑ Ajoutez l'opérateur **morphology::perosionreconstruction**.
- ❑ Le reliez à **pdilatation** et à **prgb2gray** comme sur le schéma de droite.
- ❑ Mettez la valeur du paramètre *connexity* = 8.
- ❑ Exécutez l'opérateur et visualisez l'image de sortie #0. Vérifiez que la plaque a repris sa taille normale.

Enfin, faisons la différence avec l'image de niveaux de gris.

- ❑ Déplacez l'opérateur **arithmetic::pdif** sur l'espace de travail.
- ❑ Le reliez à la fois à **perosionreconstruction** et **prgb2gray**.
- ❑ Exécutez l'opérateur et visualisez l'image de sortie #0. Vérifiez que les caractères ont bien été détectés.



Le graphe à construire.

## 6/ Seuillage

**Observation** : Dans l'image obtenue précédemment, les caractères apparaissent comme les pixels les plus clairs. L'opération de **seuillage** va permettre de sélectionner les pixels qui nous intéressent.

Le seuillage consiste simplement à mettre à blanc les pixels de valeur supérieure à une certaine valeur et à noir les autres. On obtient donc une image en noir et blanc où les pixels blancs sont les pixels qui nous intéressent.

$$pixel_i = \begin{cases} 0; & \text{si } pixel_i < \text{seuil} \\ 255; & \text{si } pixel_i \geq \text{seuil} \end{cases}$$

Ici, la valeur de seuil est obtenue automatiquement après le calcul d'une grandeur mathématique : **l'entropie** (voir l'explication à droite).

- Dans l'explorateur situé à gauche de la fenêtre Ariane, déplacez l'opérateur **thresholding::pentropybinarization** sur l'espace de travail.
- Reliez cet opérateur à **pdif**.
- Exécutez l'opérateur *et visualiser l'effet du seuillage*.

**Entropie** : c'est une mesure qui décrit le désordre dans les valeurs de niveaux de gris des pixels de l'image.

$$entropie = \sum_{i=1}^{|image|} p_i \log(p_i)$$

où  $p_i$  est la fréquence d'apparition de la valeur de  $i$  dans l'image (i.e., le nombre de fois où la valeur  $i$  apparaît dans l'image divisé par le nombre total de pixels dans l'image).

Plus la valeur d'entropie est élevée plus il existe de valeurs différentes et donc plus il y a de désordre.

La valeur d'entropie utilisée pour le seuil est celle qui permet de séparer les pixels de l'image en deux classes (pixels sombres et pixels clairs) de telle manière que le désordre à l'intérieur de chacune de ces deux classes soit minimum.

## 7/ Détection des zones denses en caractères

**Observation** : La plaque d'immatriculation est une zone horizontalement dense en caractères. Pour détecter cette zone, l'idée consiste à dilater uniquement à l'horizontal les caractères d'une certaine épaisseur pour qu'ils finissent par se toucher. On va prendre ici une dilatation de 10 pixels. La dilatation horizontale consiste à remplacer un pixel par la valeur maximale parmi ses 10 voisins de gauche et ses 10 voisins de droite.

On procède ensuite à la transformation inverse, une érosion, pour que les régions reprennent leur taille initiale, sauf pour les traits regroupés qui restent regroupés.

- Dans l'explorateur situé à gauche de la fenêtre Ariane, déplacez l'opérateur **morphology::plinedilatation** et l'opérateur **morphology::plinearerosion** sur l'espace de travail.
- Reliez l'opérateur **plinedilatation** à **pentropybinarization** et **plinearerosion** à **plinedilatation** comme sur le schéma page 2.
- Mettez la valeur de *orientation1=0*, *orientation2=0* et *halfsize=10* pour les deux opérateurs.
- Exécutez et visualisez le résultat. Vérifiez que les caractères ont bien été regroupés dans une seule région.

**Dilatation pour regrouper deux traits proches :**



i) 2 contours proches



ii) Dilatation des contours de taille 5



iii) Dilatation des contours de taille 10.



iv) Érosion de taille 10 pour retrouver la largeur initiale.

## 8/ Élimination des petites régions

**Observation** : À ce stade, nous avons sélectionné la plaque mais aussi des choses autres. Pour ne sélectionner que la plaque, nous allons utiliser le fait qu'une plaque correspond en général à une région de grande surface. Toutefois, c'est vous qui devez déterminer la taille d'une plaque d'immatriculation en

nombre de pixels.

Pour vous aider, le logiciel de visualisation des images permet de dessiner un rectangle avec la souris et d'afficher en bas à gauche le nombre de pixels correspondant à ce rectangle (voir image ci-contre).



Le coin bas gauche de la fenêtre de visualisation des images affiche la taille du rectangle dessiné.

**pareaselection** : sélection des régions sur leur valeur de surface en nombre de pixels.

- Ajoutez l'opérateur **segmentation::plabeling** sur l'espace de travail et le reliez à **plinearerosion**.
- Mettez la valeur de paramètre *connexity* à 8.
- Puis ajoutez l'opérateur **region::pareaselection** et le reliez à **plabeling**.
- Mettez la valeur du paramètre *relation* à 1 et la valeur que vous venez de déterminer pour le paramètre *threshold*.
- Exécutez et vérifiez le résultat; il ne doit rester que la région de la plaque. Au besoin, changez la valeur du paramètre *threshold* et recommencez.

## 9/ Visualisation des résultats

Nous avons maintenant réussi à détecter la plaque d'immatriculation. Pour visualiser le résultat, nous allons superposer la frontière de la plaque sur l'image initiale. La plaque sera repérée par la boîte englobante (le rectangle exinscrit).

- Déplacez l'opérateur **region::pboundingbox** sur l'espace de travail et le reliez à **pareaselection**.
- Reliez **pareaselection** à **pboundingbox**.
- Ajoutez l'opérateur **region::pboundary** sur l'espace de travail, et mettez la valeur 8 pour *connexity*.
- Reliez **pboundingbox** à **pboundary**.
- Ajoutez l'opérateur **visualization::psuperimposition** sur l'espace de travail. Mettez la valeur du paramètre *band* à 1.
- Reliez la sortie de **pany2pan** à l'entrée n°1 de **psuperimposition** et la sortie de **boundingbox** à l'entrée n°2 de **psuperimposition** (voir le schéma page 1).
- Exécutez et vérifiez la cohérence des résultats.

## 10/ Autres images

S'il est bien construit, le même plan, sans modification, doit pouvoir détecter les plaques sur toutes les autres images du dossier.

- Changez l'image en double-cliquant sur **loader** du début de graphe.
- Sélectionnez une nouvelle image.
- Réexécutez l'opérateur **psuperimposition** en fin de chaîne de traitement.
- Analysez le résultat et ajuster la valeur de surface que vous avez mis dans **pareaselection**.