

L3 Images

TD/TP 2

Sylvain Lobry

09/02/2020

Slides: www.sylvainlobry.com/L3Images

Sondages!

<https://www.wooclap.com/L3IMAGES2>

Sommaire

1 - Projet

2 - TD

Projet

Règle #1: regarder les images

- Toujours regarder les (ou des) images en entrées
- Plus tard: regarder visuellement les sorties de vos algorithmes

Projet

Règle #1: regarder les images



Projet

Règle #1: regarder les images



Projet

Règle #1: regarder les images



Projet

Règle #1: regarder les images



Projet

Règle #1: regarder les images



Projet

Etiquetage

- Historiquement, les gens utilisent <http://labelme.csail.mit.edu/Release3.0/>

The screenshot shows the LabelMe website. On the left, a red-bordered box highlights the "Log In" form. It contains fields for "Username" and "Password", a "Forgot your password?" link, and a "Login" button. Below the form is a note about account creation. On the right, a circular inset shows a street scene with various objects like trees, buildings, and a car, each outlined in different colors (yellow, green, blue) to demonstrate the labeling process. Below the inset, the text "Label objects in the images." is visible, along with three red dots at the bottom.

Welcome to LabelMe, the open annotation tool.

The goal of LabelMe is to provide an online annotation tool to build image databases for computer vision research. You can contribute to the database by visiting the annotation tool.

Log In

Username
Password [Forgot your password?](#)

Login

Unfortunately, we cannot create any new LabelMe accounts. If you have an existing LabelMe account, you will still be able to log in. We recommend that you [install the LabelMe tool](#) to label your own images.

(c) MIT, Computer Science and Artificial Intelligence Laboratory. [Accessibility](#)

Projet

Etiquetage

- Historiquement, les gens utilisent <http://labelme.csail.mit.edu/Release3.0/>
- Inscriptions closes...
- Solution possible de remplacement <https://github.com/wkentaro/labelme>
- Executables disponibles, installation via pip possible.
- Autre solution possible: <https://github.com/tzutalin/labelImg/>

Projet

Etiquetage

- A vous!
- Lien vers les images: <https://moodle.u-paris.fr/mod/resource/view.php?id=257999>

Sommaire

1 - Projet

2 - TD

TD

Manipuler des images en Java

- Java AWT (Abstract Window Toolkit) -> API pour la création d'interface graphique en Java
- Classe `java.awt.image` pour représenter une image dans une GUI:
 - Contient informations telles que: type, taille, ... et les données (raster)
 - `Java.awt.image` -> données non accessibles
- Classe `java.awt.BufferedImage` -> les données sont tamponnées (et donc accessibles)
- Constructeur à utiliser: `BufferedImage(int width, int height, int imageType)`
- `imageType` peut être:
 - `TYPE_3BYTE_BGR`
 - `TYPE_4BYTE_ABGR`
 - `TYPE_BYTE_BINARY`
 - `TYPE_BYTE_GRAY`
 - ...

TD

Opérations de base

- Taille: `img.getWidth()` et `img.getHeight()`
- Accéder aux données: `img.getRaster()` (objet de classe `WritableRaster`)
- Lire un pixel de l'image: `img.getRaster().getPixel(int x, int y)`
- Écrire un pixel de l'image `img.getRaster().setPixel(int x, int y, int[] couleur)`
 - Couleur dépend du type de données. Pour ARGB: `int[] bleu = {0,0,255,255};`
- Dans le cas de RGB, accès direct:
 - `Img.getRGB(int x, int y)`
 - `Img.setRGB(int x, int y, int couleur)` -> Attention, couleur codée par octet sur un entier (voir exemple TD1)

TD

Entrées/Sorties

- Classe javax.imageio.ImageIO -> swing aussi pour la creation d'interfaces utilisateur
- ImageIO.read(obj) -> obj peut-être:
 - File (pour un fichier en local)
 - URL (pour une image depuis internet)
- Possibilité de sauvegarder l'image avec ImageIO.write

TD

Cette semaine

- Calculs d'histogrammes, seuil
- Exercice 1: seuiller cette image. Qu'en pensez-vous ?

