

Télécom Physique Strasbourg - Université de Strasbourg - TIC Santé 2A

TP 2 : Images et Vision

Introduction

Objectifs du TP :

- Utilisation de la librairie OpenCV
- Seuillage, binarisation, normalisation et segmentation d'images.

Pour la compilation des programmes, on utilisera cmake et le compilateur g++ sous le système Linux. On pourra trouver la documentation d'OpenCV sur le site <http://docs.opencv.org> et un bref résumé de certaines fonctions dans le fichier http://docs.opencv.org/2.4/opencv_cheatsheet.pdf. Il est conseillé de créer un répertoire par exercice.

Application

Nous disposons d'une image "coins.jpg" de pièces de monnaie. Nous souhaitons écrire un programme pour compter automatiquement le nombre de pièces dans l'image et identifier la région couverte par chaque pièce.

Nombre de pièces de monnaie

1. Visualiser l'image "coins.jpg".
2. Convertir l'image lu en niveau de gris et visualiser le résultat.
3. Convertir l'image en niveau de gris en une image binaire (en noir=0 et blanc=255). On utilisera pour cela la fonction `cv::threshold`. Les pixels d'intensité supérieure à 40 seront blancs. Visualiser l'image.
4. La fonction `cv::distanceTransform` calcule la distance de tout pixel au pixel noir le plus proche. Utiliser cette fonction pour créer la carte des distances à partir de votre image binaire. On utilisera la distance `CV_DIST_L2` avec un masque 3×3 . Visualiser le résultat.
5. Normaliser la carte des distances afin de ramener les distances entre les valeurs 0 et 1. On utilisera la norme `NORM_MINMAX` comme type de norme. Visualiser.
6. Binariser la carte des distances normalisée de sorte que les distances supérieures à 0.5 prennent la valeur 1 (0 pour les autres). Visualiser.
7. Utiliser `cv::findContours` pour trouver les contours présents dans la carte des distances (normalisée et seuillée). Que représente le nombre de contours ?

Localisation les pièces de monnaie

1. Créer une image (un seul canal de 32 bits), de la même taille que coins.jpg, initialisée entièrement avec des 0.
2. Dessiner un petit cercle blanc de rayon 3 pixels centré au coordonnées (5,5).
3. Dessiner les contours des pièces de monnaie dans cette nouvelle image. Donner une valeur différente (1,2,3...) à la région délimitée par le contour associé à chaque pièce de monnaie. Pour cela, on utilisera `cv::drawContour`. L'image ainsi obtenue, dite image de marqueurs, servira de guide pour la segmentation de l'image d'origine
4. Utiliser `cv::watershed` pour segmenter l'image d'origine coins.jpg avec l'aide de l'image de marqueurs.
5. Colorier chaque région correspondant à une pièce par une couleur différente. Visualiser le résultat.