Balsan Matthieu

Connesson Thomas

Date du TP2(15/11/2021)

# Rapport de TP2 – **Lecture automatique de chiffres par analyse d’image**

## Introduction

Au cours de ce TP, nous allons programmer la lecture automatique d’une image à partir d’un niveau de gris représenté par des pixels généralement codés sur 8 bits. Les pixels sont placés sur un tableau d’une longueur H et d’une largeur W. Nous souhaitons que l’image soit traitée de façon automatique. Nous allons dans notre cas utiliser la reconnaissance par corrélation avec des modèles. A la suite d’étapes détailler au cours du TP, nous obtiendrons une image basée sur l’analyse à l’image modèle de corrélation maximale.

## Travail préparatoire

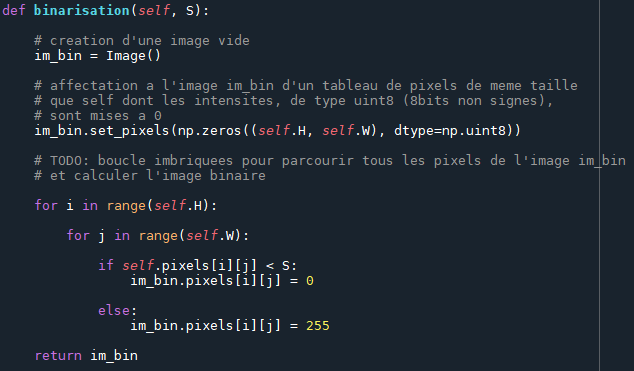
### Question (1)

D’après notre analyse, nous observons que l’attribut pixel permet de définir une image et que les attributs H et W sont des attributs de taille correspondant respectivement à la hauteur et la largeur de l’image.

### Question (2)

Pour passer d’un pixel codé sur 256 valeurs à un pixel prenant deux choix de valeurs (0 ou 255), nous allons utiliser un seuil de binarisation qui nous permettra de dire si le pixel prend la valeur 0 ou 255 et allons créer deux boucles sachant qu’une boucle sera incrémenté dans l’autre. De ce fait, une première boucle pour la hauteur est créée. Au sein de cette boucle, nous allons créer une nouvelle boucle pour définir chaque pixel d’une ligne dans sa largeur W. Une fois la boucle écrite, nous instaurons un seuil de binarisation en disant : Si la valeur du pixel est inférieure au seuil, alors elle prendra la valeur 0 sinon elle prendra la valeur 255.

Voici le codage correspondant à cette demande :



### Question (3)

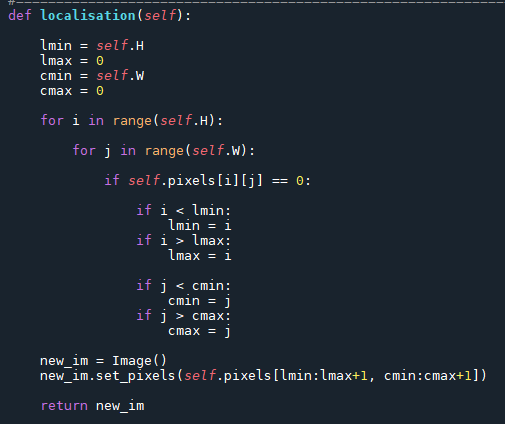
La méthode de localisation nous demande de créer une sous image à partir d’un rectangle ne prenant en compte que les pixels qui sont dans ce même rectangle. Une fois le rectangle défini, nous pouvons retourner l’image sélectionnée par le rectangle.

Nous commençons par dire que le rectangle est définir par 4 attributs : lmin,lmax et cmin, cmax.

Disons que : lmin= self.H, lmax=0 et cmin=self.W cmax= 0

En faisant cette hypothèse, nous pouvons dire que si la valeur d’un pixel ayant comme coordonnées [i][j] est la même que 0, alors on compare notre valeur de ligne avec lmax et lmin et les colonnes avec cmax et cmin. Si le numéro de ligne ou de colonne respecte la condition, alors elle remplace la valeur précédente.

Voici la commande en python :



## Reconnaissance automatique de chiffre

### Question 1

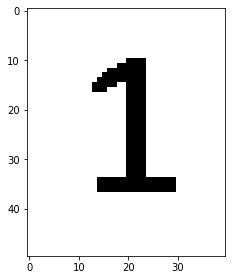
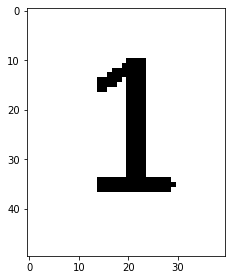
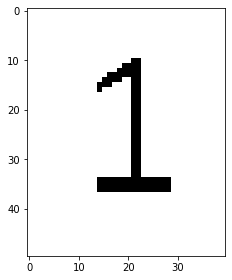
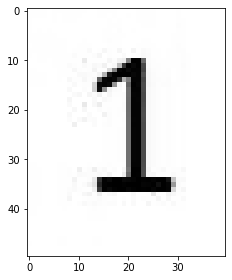


Image non binarisé Image binarisé avec S = 70 Image binarisé avec S = 165 Image binarisé avec S = 220

On peut voir qu’avant la binarisation, le 1 est un peu flou et qu’après, ses contours sont nets. On peut également voir que plus le seuil est élevé, plus le nombre de pixels en noirs qui compose le 1 sont nombreux.

### Question 2

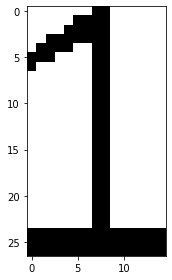
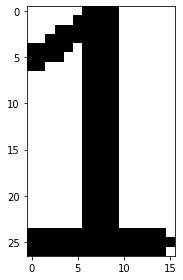
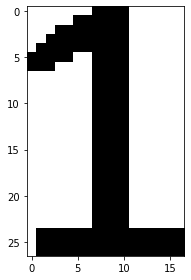
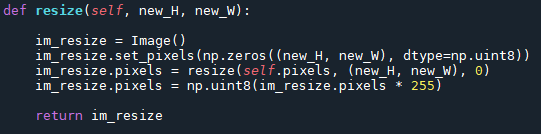
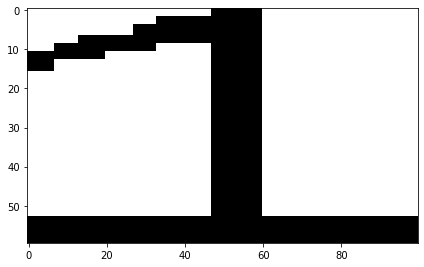


Image localisée avec S = 70 Image localisée avec S = 165 Image localisée avec S = 220

### Question 3

Pour la méthode resize, nous allons l’établir sur le programme de localisation. Nous prenons notre image précédente avec un nombre de pixels différents nous donnant une nouvelle hauteur (new\_H) et une nouvelle largeur (new\_W)

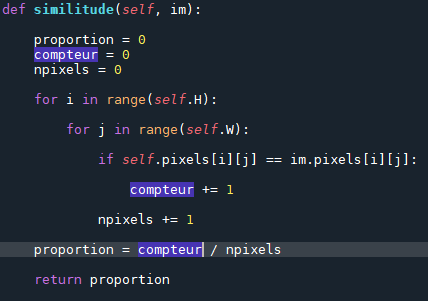
Voici le programme en python :



### Question 4

Pour la méthode similitude, nous allons reprendre la méthode précédente. Nous cherchons à obtenir assez de similitude entre les deux images pour obtenir un chiffre entre 0 et 1.

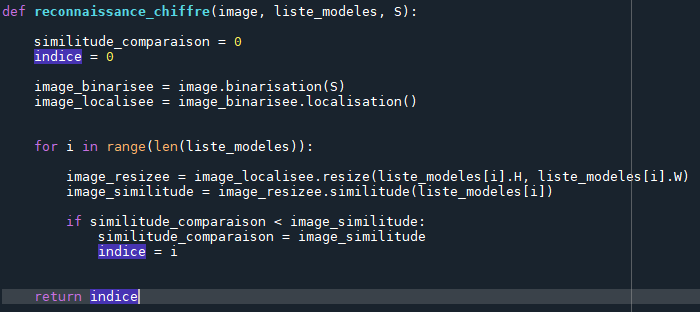
Voici le programme python de la méthode suivante :



### Question 5

Pour la méthode reconnaissance, nous allons reprendre la méthode précédente. Nous cherchons à déterminer le chiffre qui se trouve sur l’image. On redimensionne l’image en fonction du modèle, on cherche une similitude entre l’image et le modèle et enfin on réalise cela pour une liste de modèle. On garde modèle avec l’indice de similitude le plus élevé.

Voici le programme python de la méthode suivante :



## Conclusion

En conclusion, nous avons pu établir un programme permettant la lecture automatique de chiffre par analyse d’image. Nous avons décomposé le programme en plusieurs étapes. Une première qui permet de pixeliser notre image avec des nombres binaires, une seconde en recadrant l’image avec les pixels nécessaires uniquement. Une troisième étape permettant de donner une taille à notre image afin de la redimensionner. Une quatrième étape nous donnant la ressemblance entre notre image redimensionnée ainsi l’image localisée et binarisée. Une dernière étape permettant de la reconnaissance et l’identification du numéro sur l’image en question. Toutes ces étapes nous ont permis de créer un programme permettant d’effectuer de façon plus simpliste, une reconnaissance d’objet avec un modèle.