

Dernière mise à jour	Informatique	Denis DEFAUCHY
14/06/2023	7 - Matrices de pixels et images	TD 7-17 – Recadrage projectif

Informatique

7

Matrices de pixels et images

TD 7-17

Recadrage projectif

Dernière mise à jour	Informatique	Denis DEFAUCHY
14/06/2023	7 - Matrices de pixels et images	TD 7-17 – Recadrage projectif

Contexte

On souhaite créer un algorithme qui recadre un document pris en photo en définissant manuellement les 4 coins de l'image pour en déduire l'image recadrée. Nous allons mettre en place l'algorithme réalisant ce travail par géométrie projective, ce qui permettra d'obtenir l'image comme si elle avait été prise verticalement en tous ses points. Voici un exemple d'application pour visualiser les pistes du plus bel aéroport de France (Fontenay-Trésigny 😊) :



Merci à Pascal Peter qui a publié la méthode claire ici ([lien](#)).

Affichage de l'image

Afin d'assurer un fonctionnement rapide sur tous les ordinateurs, je vous mets à disposition un dossier à télécharger COMPLETEMENT, soit le dossier contenant tous les fichiers, et non les fichiers pris séparément

Sans ouvrir le dossier, faite juste « Télécharger – Téléchargement direct » puis mettez ce dossier dans votre répertoire personnel.

[LIEN](#)

Dossier_Partagé

Copier dans Dropbox

⬇ Télécharger

Nom

7-17 - Recadrage projectif

Si le téléchargement est sous forme de Rar, Zip... Pensez à dézipper l'archive afin d'avoir le dossier voulu !

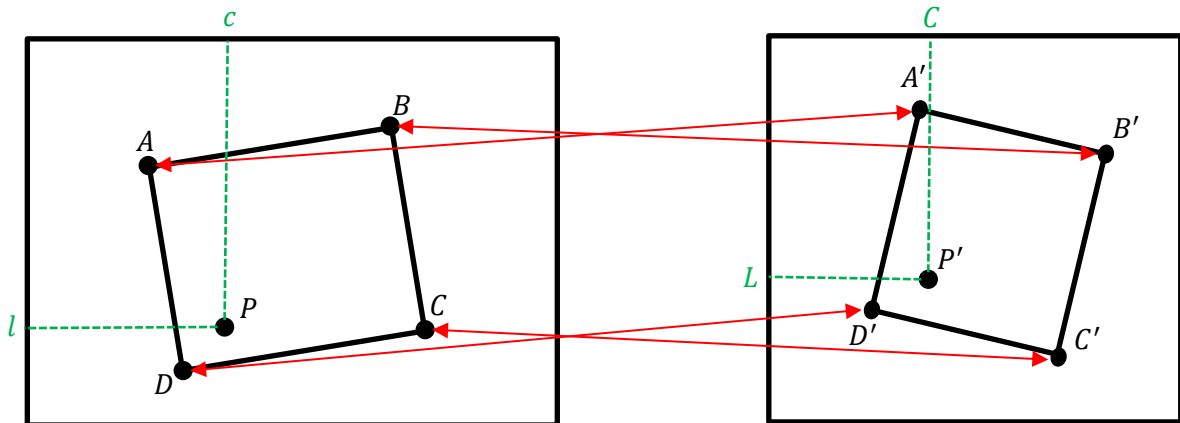
Vous y trouverez un code élève et l'image « Image.bmp » ainsi que sa version numpy « Image.npy ».

Question 1: Télécharger et exécuter le code fourni qui créera et affichera l'image « Image » sous Python

Dernière mise à jour	Informatique	Denis DEFAUCHY
14/06/2023	7 - Matrices de pixels et images	TD 7-17 – Recadrage projectif

Géométrie projective

Soit une image contenant une zone plane délimitée par 4 points $ABCD$ (ordre respectant le contour). On souhaite transformer cette zone en une autre zone plane délimitée par les points $A'B'C'D'$. La zone $A'B'C'D'$ correspondra à la transformation du plan $ABCD$ en un plan $A'B'C'D'$, quel que soit le point de vue.



Le recadrage classique d'images sera tel que $(A'B')$ et $(C'D')$ soient des droites horizontales, et que $(A'D')$ et $(B'C')$ des droites verticales, mais la méthode que nous abordons ici est valable de manière plus générale.

On pose :

$$A = \begin{pmatrix} lA \\ cA \end{pmatrix} ; \quad B = \begin{pmatrix} lB \\ cB \end{pmatrix} ; \quad C = \begin{pmatrix} lC \\ cC \end{pmatrix} ; \quad D = \begin{pmatrix} lD \\ cD \end{pmatrix}$$

$$A' = \begin{pmatrix} lA' \\ cA' \end{pmatrix} ; \quad B' = \begin{pmatrix} lB' \\ cB' \end{pmatrix} ; \quad C' = \begin{pmatrix} lC' \\ cC' \end{pmatrix} ; \quad D' = \begin{pmatrix} lD' \\ cD' \end{pmatrix}$$

Soit un point $P = \begin{pmatrix} l \\ c \end{pmatrix}$ et son image recherchée par la transformation projective $P' = \begin{pmatrix} L \\ C \end{pmatrix}$.

Dans toute la suite, une matrice sera une liste de listes, et un vecteur sera une liste. On utilisera « solve » du sous module « linalg » de « numpy » pour résoudre les systèmes linéaires et « dot » de numpy pour le produit matrice/vecteur.

On crée la matrice $K = \begin{bmatrix} cA & cB & cC \\ lA & lB & lC \\ 1 & 1 & 1 \end{bmatrix}$ et le vecteur $V = \begin{bmatrix} cD \\ lD \\ 1 \end{bmatrix}$. On en déduit la solution (k_1, k_2, k_3) du système :

$$K \begin{bmatrix} k_1 \\ k_2 \\ k_3 \end{bmatrix} = V$$

On crée la matrice $K' = \begin{bmatrix} cA' & cB' & cC' \\ lA' & lB' & lC' \\ 1 & 1 & 1 \end{bmatrix}$ et le vecteur $V' = \begin{bmatrix} cD' \\ lD' \\ 1 \end{bmatrix}$. On en déduit la solution (k'_1, k'_2, k'_3) du système :

$$K' \begin{bmatrix} k'_1 \\ k'_2 \\ k'_3 \end{bmatrix} = V'$$

Dernière mise à jour	Informatique	Denis DEFAUCHY
14/06/2023	7 - Matrices de pixels et images	TD 7-17 – Recadrage projectif

On crée les rapports :

$$R_1 = \frac{k'_1}{k_1} \quad ; \quad R_2 = \frac{k'_2}{k_2} \quad ; \quad R_3 = \frac{k'_3}{k_3}$$

On crée la matrice $M = \begin{bmatrix} cA & lA & 1 \\ cB & lB & 1 \\ cC & lC & 1 \end{bmatrix}$ et les vecteurs $V_1 = \begin{bmatrix} R_1 cA' \\ R_2 cB' \\ R_3 cC' \end{bmatrix}$, $V_2 = \begin{bmatrix} R_1 lA' \\ R_2 lB' \\ R_3 lC' \end{bmatrix}$ et $V_3 = \begin{bmatrix} R_1 \\ R_2 \\ R_3 \end{bmatrix}$

On propose une matrice $M_a = \begin{bmatrix} a_{11} & a_{12} & a_{31} \\ a_{21} & a_{22} & a_{32} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$ telle que chaque ligne soit solution d'un système :

- (a_{11}, a_{12}, a_{13}) solution de $M_a \begin{bmatrix} a_{11} \\ a_{12} \\ a_{13} \end{bmatrix} = V_1$
- (a_{21}, a_{22}, a_{23}) solution de $M_a \begin{bmatrix} a_{21} \\ a_{22} \\ a_{23} \end{bmatrix} = V_2$
- (a_{31}, a_{32}, a_{33}) solution de $M_a \begin{bmatrix} a_{31} \\ a_{32} \\ a_{33} \end{bmatrix} = V_3$

On introduit le vecteur $U = \begin{bmatrix} c \\ l \\ 1 \end{bmatrix}$. On détermine le vecteur par produit matrice/vecteur

$$U' = M_a U = \begin{bmatrix} a \\ b \\ c \end{bmatrix}$$

On obtient alors les lignes L et C ainsi : $\begin{cases} L = \frac{b}{c} \\ C = \frac{a}{c} \end{cases}$

On veillera à arrondir ces valeurs à l'entier le plus proche et à transformer le résultat au format entier.

On propose de créer deux listes LP des points [A,B,C,D] et LPp des points [Ap,Bp,Cp,Dp].

Question 2: Proposer la fonction LC_Proj(l,c,LP,LPp) permettant de renvoyer l'image L,C du couple l,c par transformation projective des points de LP vers le points de LPp

On propose les listes de points suivantes :

```
A,B,C,D = [10,15],[15,70],[60,65],[70,10]
LP = [A,B,C,D]
Ap,Bp,Cp,Dp = [0,0],[0,100],[1000,100],[1000,0],
LPp = [Ap,Bp,Cp,Dp]
```

Question 3: Vérifier votre transformation projective sur les 4 points A, B, C et D

Dernière mise à jour	Informatique	Denis DEFAUCHY
14/06/2023	7 - Matrices de pixels et images	TD 7-17 – Recadrage projectif

Recadrage

On souhaite maintenant pouvoir générer une nouvelle image de dimensions N_I, N_C dans laquelle seront appliquées à chaque pixel (l, c) la couleur du pixel (L, C) de l'image à recadrer.

On pose :

- $LA=[A1, A2, A3, A4]$ la liste des points de l'image à recadrer
- $LB=[B1, B2, B3, B4]$ la liste des points des 4 coins de l'image recadrée

On récupérera les dimensions de l'image originale à recadrer (im) pour n'appliquer la couleur aux pixels de la nouvelle image que si (L, C) appartient bien à im .

Question 4: Proposer une fonction `recadrage(im, LA, LB, NI, NC)` réalisant le travail attendu

On souhaite dans un premier temps créer une image recadrée carrée de 500 lignes et colonnes.

Question 5: Définir les listes LA et LB pour cette application

Question 6: Réaliser le recadrage attendu



Dernière mise à jour	Informatique	Denis DEFAUCHY
14/06/2023	7 - Matrices de pixels et images	TD 7-17 – Recadrage projectif

Proportions sur zone rectangulaire

Dans la partie précédente, nous avons recadré une zone rectangulaire sous forme carrée. Il nous faudrait connaître les vraies proportions de cette zone pour obtenir un résultat représentant la réalité, mais il est difficile de connaître ces proportions.

On suppose que les listes de points LA et LB sont toujours organisées de la même manière :

- A1 & B1 : Point en haut à gauche
- A2 & B2 : Points en haut à droite
- A3 & B3 : Point en bas à gauche
- A4 & B4 : Points en bas à gauche

Ainsi, les segments B1B2 et B3B4 sont théoriquement horizontaux et B2B3 et B4B1 verticaux.

On se propose de respecter la procédure suivante :

- Détermination des longueurs (en pixels) des 4 segments
- Calcul de la valeur moyenne des longueurs horizontales h
- Calcul de la valeur moyenne des longueurs des segments verticaux v
- Calcul du rapport $k = \frac{h}{v}$
- Définir les dimensions de l'image recadrée avec NI=1500 et Nc=k*NI, arrondi à l'entier le plus proche et au format entier

Question 7: Proposer une fonction Longueur(Pi,Pj) prenant en arguments deux points (listes [li,ci]) et renvoyant la longueur en pixels du segment PiPj

Question 8: Proposer une fonction Prop(LP) renvoyant le rapport k sur la liste LP organisée comme LA et LB

Question 9: Réaliser l'application permettant d'afficher les pistes de l'aéroclub recadrées dans des proportions correctes

Nouveau point de vue

Question 10: En jouant sur les points de LB et les dimensions NI,Nc, générer une image semblant être prise d'un autre point de vue, comme ci-dessous

