

(ROB313 - TD1)

# Estimation d'homographies et reconstruction de panoramas

BRAMBILLA Davide Luigi - GOMES DA SILVA Rafael

December 7, 2019

## 1 Question 1 - Rectification d'images

Dans cette section, l'objectif est de développer un algorithme afin d'implémenter une homographie qui puisse rectifier une image selon une direction désirée.

### 1.1 La fonction *homography2d* implémentée

Ci-dessous nous avons reporté le code qui nous avons implémenté:

```
% TODO : Create matrix A
A = zeros(3*length(x1),9); %initialization matrix: in this case(3*numofpoint x 9)

for i = 1 : length(x1) %size of x1: number of points
    xi = x1(:,i);
    x = x2(1,i);
    y = x2(2,i);
    w = x2(3,i);

    A(3*i-2,:) = [[0 0 0] -w*xi' y*xi'];
    A(3*i-1,:) = [ w*xi' [0 0 0] -x*xi'];
    A(3*i ,:) = [-y*xi' x*xi' [0 0 0] ];
end

% TODO : perform SVD
[U,S,V] = svd(A);

% TODO : Extract homography from SVD result
h = V(:,9); %vector of 9 elements
H = reshape(h,3,3)'; %transformation of the vector into a matrix
```

Le code commence avec la construction de la matrice  $A$  comme définie dans les diapositives du cours: pour chaque point qui devra se correspondre, on calculera la partie correspondante de la matrice  $A$ . La matrice  $A$  est une matrice de taille  $2*numpoints \times 9$ .

En suite, nous calculerons la *décomposition en valeurs singulières* au travers du commande Matlab  $[U,S,V] = svd(X)$  qui donne en sortie une matrice diagonale  $S$  avec des éléments non négatifs et deux matrices  $U$  et  $V$  de façon qui  $X = U*S*V^T$ . Nous allons nous intéresser à la matrice  $V$  étant donnée que sa dernière colonne sera  $h$ . Enfin, nous allons reconstituer la matrice carré  $H$  à partir du vecteur  $h$  qui sera la matrice représentant l'opération d'homographie.

### 1.2 La vérification de l'algorithme

Afin de vérifier l'algorithme, nous avons pris l'image reporté ci-dessous, qui représente une mosaïque mais qui est prise par un point de vue latéral.

Notre but sera de restituer la même image mais vue de dessus de façon que il n'y aura pas des déformations sur le mosaïque.

Pour faire cela nous avons commencé en prenant des points sur l'image d'origine (nous avons choisi les coins du mosaïque):



Figure 1.1: Choix des points pour rectifier l'image du mosaïque

Finalement, nous avons calculé la transformation homographique  $H$  au travers de la fonction *homography2d* implémentée et nous avons transformé l'image.

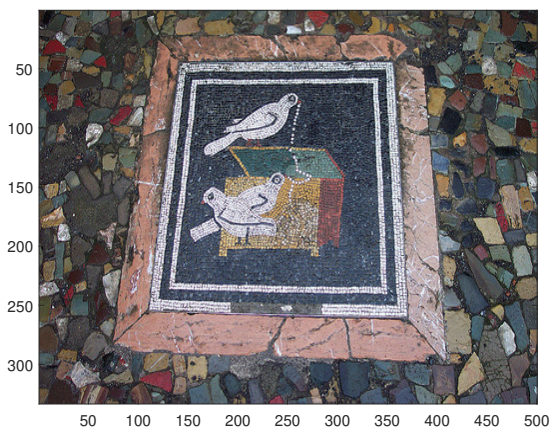
```
% Choix des points - the corners of the image
p1 = [142 45];
p2 = [356 45];
p3 = [111 255];
p4 = [371 261];

% Matrice Pt0
Pt0 = [ p1(1) p2(1) p3(1) p4(1); ... % Choice of the points to transform
        p1(2) p2(2) p3(2) p4(2); ...
        1     1     1     1]

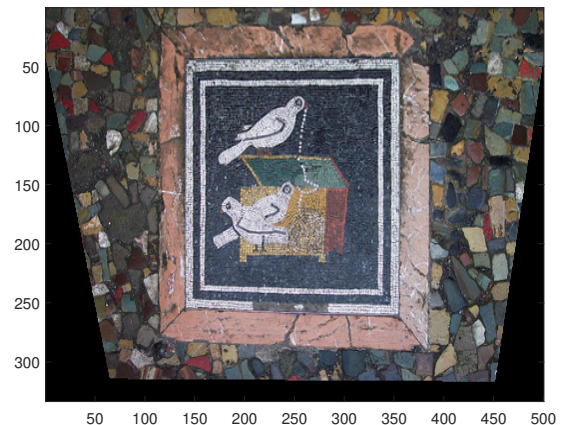
% Matrice PtD
PtD = [ p1(1) p2(1) p1(1) p2(1); ... % Remapping of the points
        p1(2) p2(2) p3(2) p4(2); ...
        1     1     1     1]

H = homography2d(Pt0,PtD);
imagesc(vgg_warp_H(imrgb,H));
```

Le résultat obtenu est reporté ci-dessous où on se peut rendre compte que le mosaïque est maintenant bien centré et il n'a pas subi des rotations.



(a) Image Pompei avant la rectification



(b) Image Pompei après la rectification

Figure 1.2: Exemple de rectification d'image avec une photo d'un mosaïque de Pompei

## 2 Question 2 - Assemblage d'images

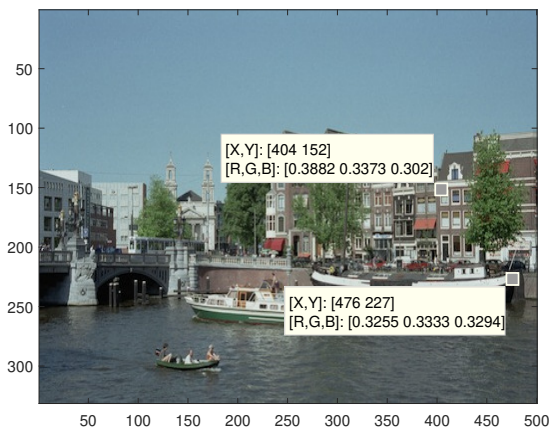
Dans cette section il nous est demandé d'utiliser l'homographie définie ci-dessus afin de pouvoir corriger l'orientation des images proposées et les unifier en une image panoramique.

### 2.1 Assemblage de deux images

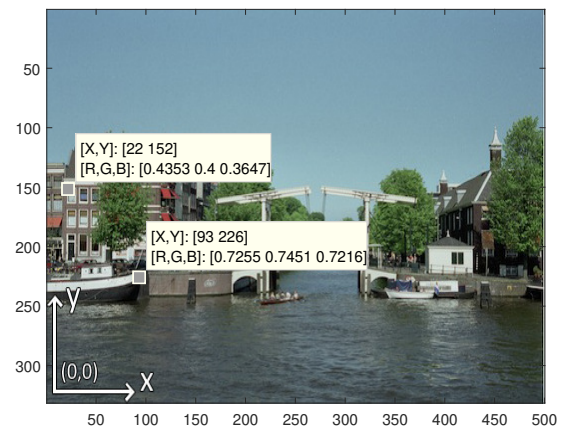
Au début, nous nous sommes concentrés en unifiant deux images de Amsterdam.

Nous avons noté que l'image  $a$  sur la droite était une continuation de l'image  $b$  sur la gauche. Donc nous avons créé une homographie qui permettait de faire correspondre aux points de la partie droite de l'image  $A$  les points de la partie de gauche de l'image  $b$  comme il est possible de voir dans l'image 2.1.

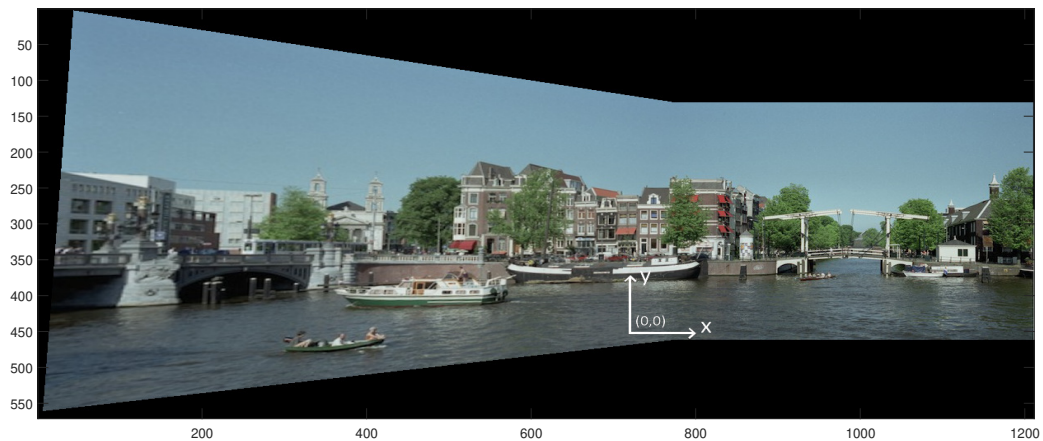
Par rapport à la Q1, nous avons choisi d'augmenter le nombre de points sur lesquels nous allons calculer une homographie jusqu'à 7 afin d'avoir une meilleure précision.



(a) Première image du panorama



(b) Deuxième image du panorama



(c) Combinaison des images 2.1a et 2.1b pour composer un panorama

Figure 2.1: Exemple assemblage d'images à l'aide de l'homographie  $H$  pour composer une image panoramique

En particulier nous avons définie deux différentes homographies.

La première à appliquer à l'image  $a$  afin de la pouvoir positionner dans une position compatible avec le positionnement de l'image  $b$ .

La deuxième qui est une homographie *identité* à appliquer à l'image  $b$  afin de la pouvoir positionner dans le même espace de l'image  $a$ .

En fait, afin de pouvoir unifier les deux images nous devons les positionner dans la position correcte et, pour faire cela, nous avons utilisé la fonction  $im\_warpeda = vgg\_warp\_H(ima, Ha, 'linear', bbox)$  en spécifiant la taille du *binding box* qui définit la taille de l'image panoramique.

## 2.2 Assemblage de trois images

Pour faire l'assemblage des trois images nous nous sommes basé sur le même principe expliqué dans la section précédente.

En particulier, nous avons choisi de appliquer l'homographie identité à l'image du milieu et de appliquer deux homographies différentes aux images de gauche et de droite après avoir choisi les points de correspondances entre les deux couples d'images.

Nous avons unifié les images en deux étapes: en unifiant l'image centrale avec une des images latérales et finalement en unifiant l'image résultante avec l'image restante.

Nous avons reportés ci-dessus les résultats obtenus:

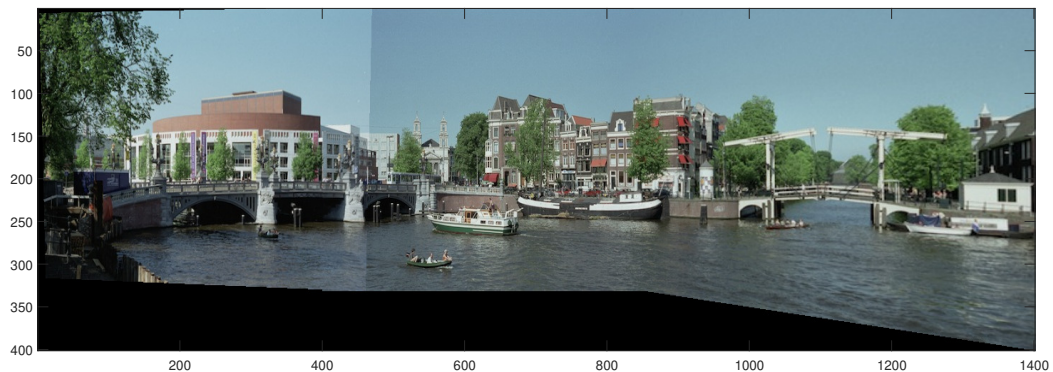
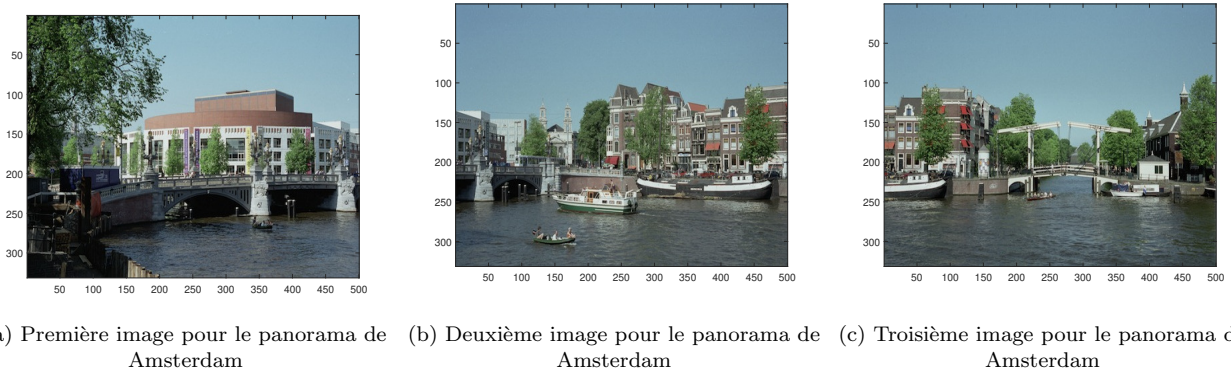
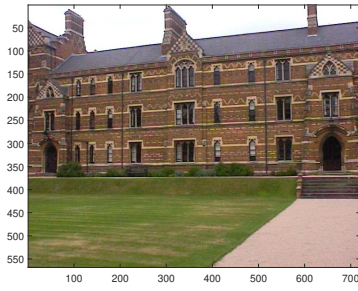
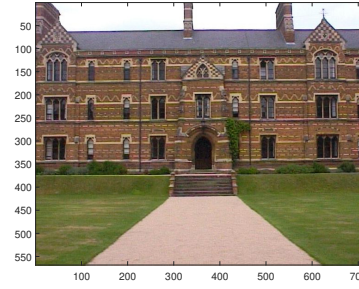


Figure 2.2: Exemple de panorama avec une photo de Amsterdam

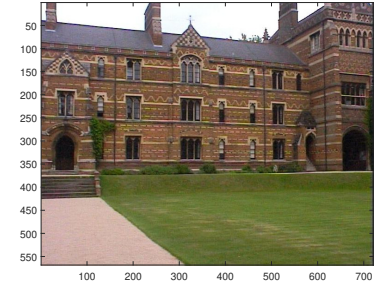




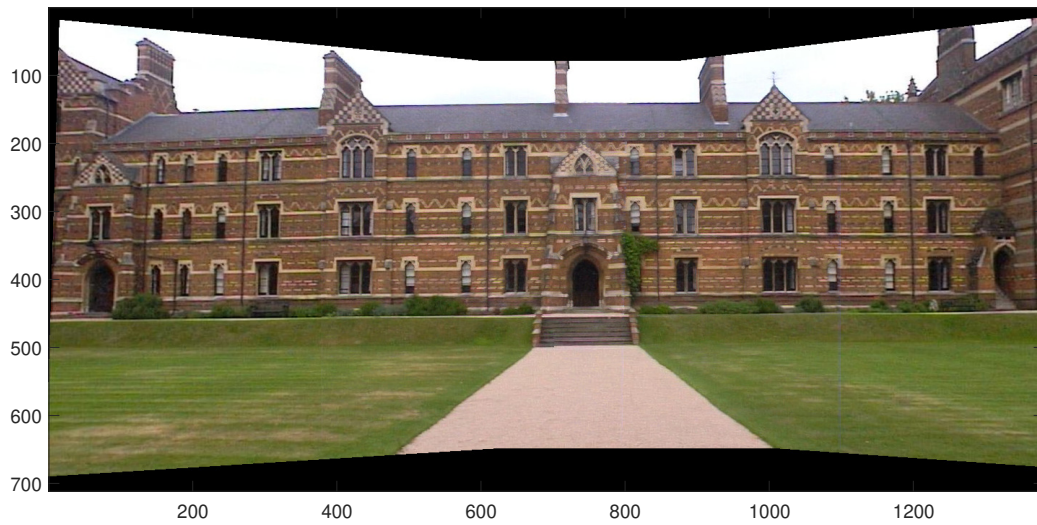
(a) Première image pour le panorama de Keble college



(b) Deuxième image pour le panorama de Keble college



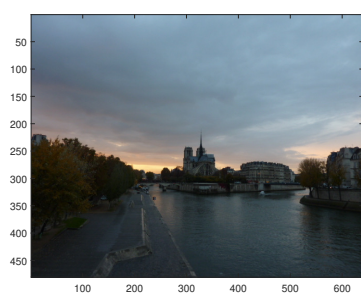
(c) Troisième image pour le panorama de Keble college



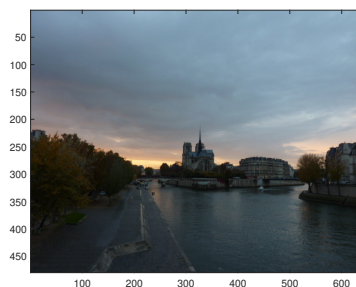
(d) Combinaison des images 2.3a, 2.3b et 2.3c pour composer le panorama de Keble college

Figure 2.3: Exemple de panorama avec une photo de Keble college

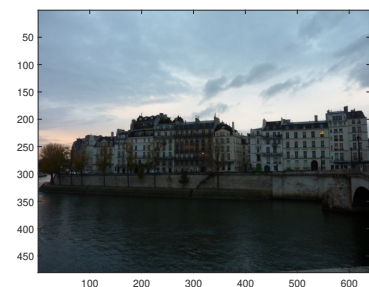
Dans cette photo nous avons choisir les points minutieusement autour des fenêtres car autrement le résultat de la fusion des images n'était pas trop précis.



(a) Première image pour le panorama de Paris



(b) Deuxième image pour le panorama de Paris



(c) Troisième image pour le panorama de Paris



(d) Combinaison des images 2.4a, 2.4b et 2.4c pour composer le panorama de Paris

Figure 2.4: Exemple de panorama avec une photo de Paris