



Projet de réalité augmentée


La réalité augmentée

- Qu'est-ce que la réalité augmentée ?
- Où trouve-t-on la réalité augmentée ?



Présentation



- Projet AGI E14 2014
- Sujet proposé et encadré par M. FASQUEL D'après les travaux de J.E. Solem
- Projet réalisé par : Masmoudi Nassim, laz Zakaria, Cassard Benjamin, Cateau Sébastien



Notre projet dans tout ça

L'objectif de notre projet est de faire interagir une plateforme réelle et un objet virtuel de notre choix.

- Reconnaissance d'un objet réel
- Localisation spatiale de notre objet
- Projection d'un objet virtuel



Outils et méthodes

- "Computer vision" de J.E. Solem
- Langage: Python
 - Orienté objet
 - Gestion des exceptions
 - Multi-plateformes
 - Projet Dynamique Fort
 - Développement web
 - Bibliothèque 3D





Notions importantes du projet



Projet de réalité augmentée


La réalité augmentée

- Qu'est-ce que la réalité augmentée ?
- Où trouve-t-on la réalité augmentée ?



Présentation



- Projet AGI E14 2014
- Sujet proposé et encadré par M. FASQUEL D'après les travaux de J.E. Solem
- Projet réalisé par : Masmoudi Nassim, laz Zakaria, Cassard Benjamin, Cateau Sébastien



Notre projet dans tout ça

L'objectif de notre projet est de faire interagir une plateforme réelle et un objet virtuel de notre choix.

- Reconnaissance d'un objet réel
- Localisation spatiale de notre objet
- Projection d'un objet virtuel



Résultats : les tests finaux



Outils et méthodes

- "Computer vision" de J.E Solem
- Langage: Python
 - Orienté objet
 - Gestion des exceptions
 - Multi-plateformes
 - Projet Dynamique Fort
 - Développement web
 - Multi-plateformes
 - Gestion des exceptions
 - Multi-plateformes 3D



Gestion de projet



Notions importantes du projet

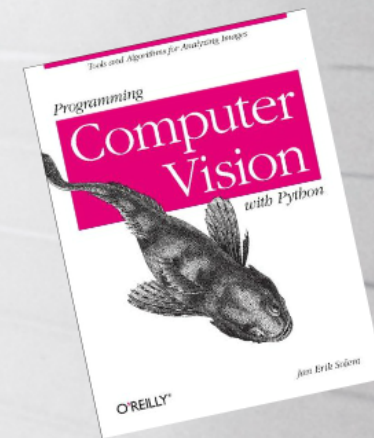


Conclusion



Présentation

- Projet AGI EI4 2014
- Sujet proposé et encadré par M. FASQUEL
D'après les travaux de J.E. Solem
- Projet réalisé par :
Masmoudi Nassim
Iaz Zakaria
Cassard Benjamin
Cateau Sébastien



La réalité augmentée

• **Qu'est-ce que la réalité augmentée ?**

• **Où trouve-t-on la réalité augmentée ?**



Localisation via son Smartphone

- Utilisation pour le tourisme
- Nouvelle approche de navigation GPS



Essayage virtuel

- Essayer des lunettes en ligne
- Essayages de vêtements ou accessoires pour les sites e-commerce



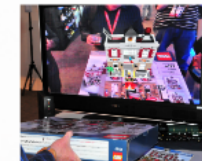
Catalogue virtuel

- Modélisation de l'intérieur de la maison
- Incorporation 3D de meubles en vente sur le catalogue BUT



LEGO

- Visualisation de la maquette 3D sur la boîte lego
- Déplacement de la structure LEGO



Localisation via son Smartphone

- Utilisation pour le tourisme
- Nouvelle approche de navigation GPS



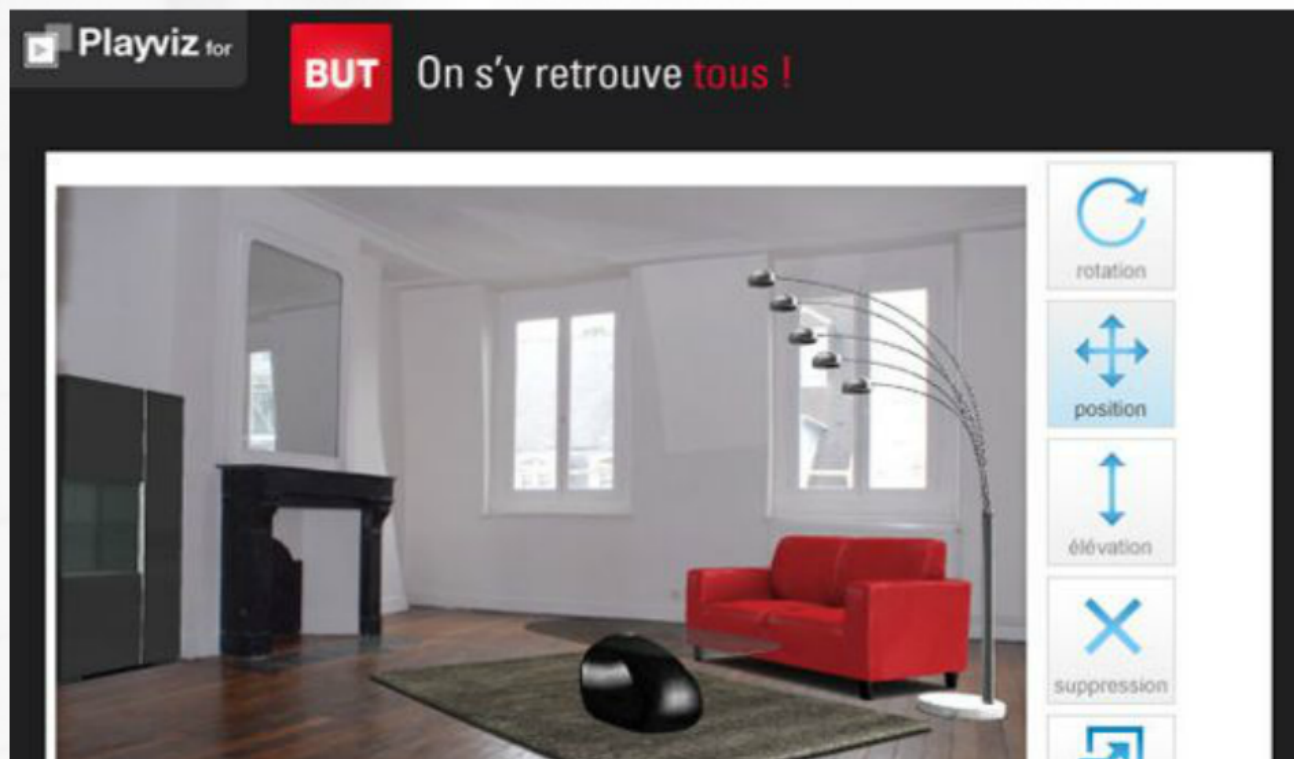
Essayage virtuel

- Essayer des lunettes en ligne
- Essayages de vêtements ou accessoires pour les sites e-commerce



Catalogue virtuel

- Modélisation de l'intérieur de la maison
- Incorporation 3D de meubles en vente sur le catalogue BUT



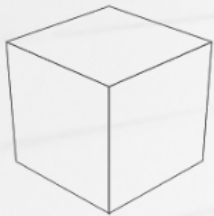
LEGO

- Visualisation de la maquette 3D sur la boîte lego
- Déplacement de la structure LEGO

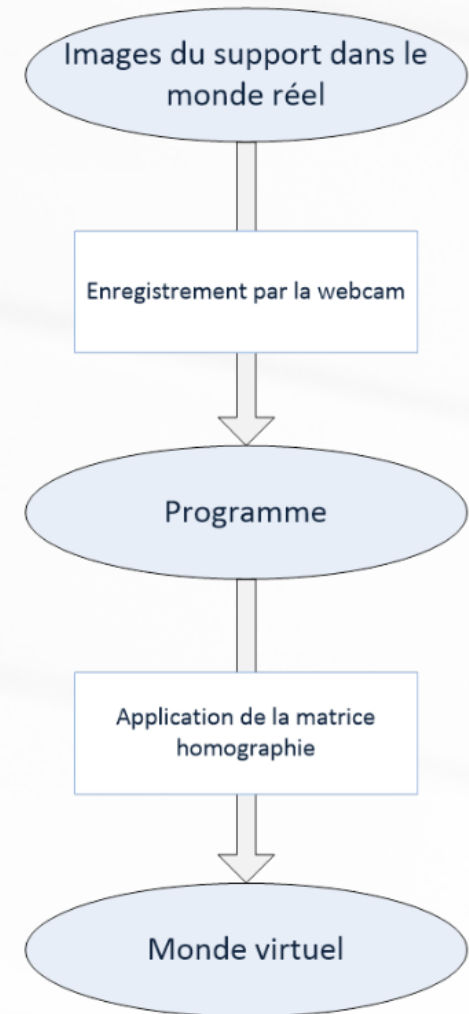


Notre projet dans tout ça

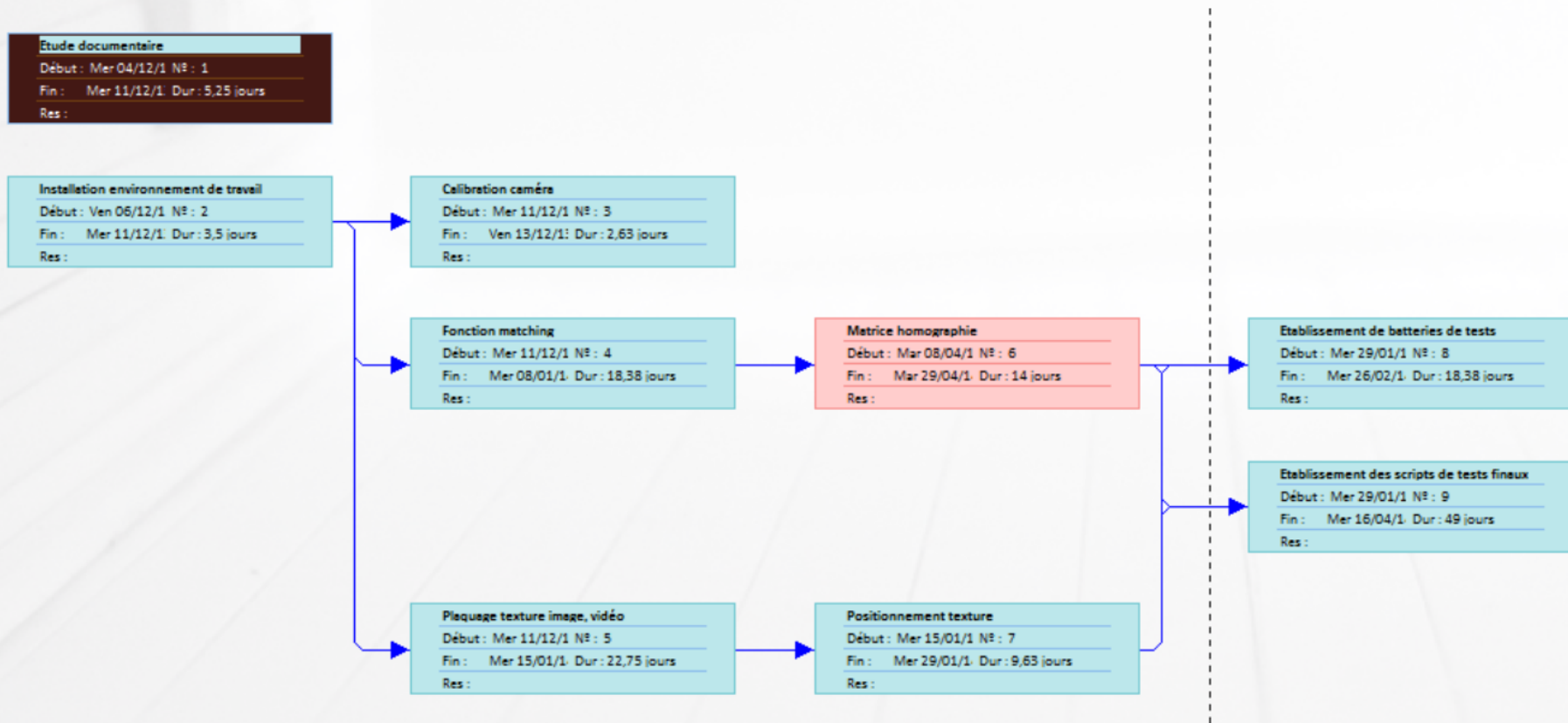
L'objectif de notre projet est de faire interagir une plateforme réelle et un objet virtuel de notre choix.

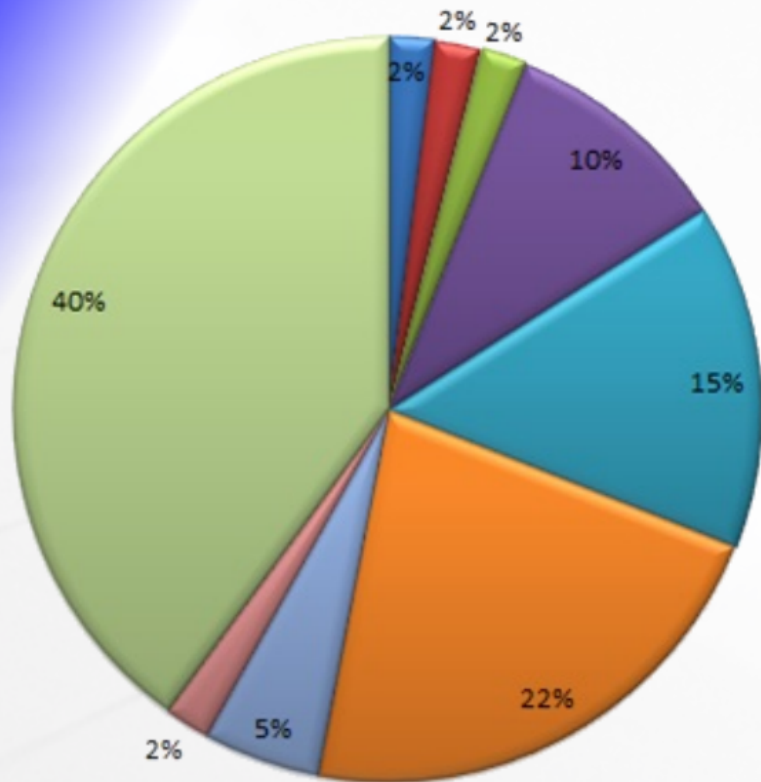


- Reconnaissance des points-clés
- Estimation matrice homographie
- Projection et recalage objet virtuel



Gestion de projet



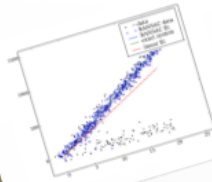


- Etude documentaire
- Installation environnement de travail
- Calibration caméra
- Fonction matching
- Plaquage texture image, vidéo
- Matrice homographie
- Positionnement texture
- Etablissement de répertoires de tests
- Etablissement des scripts tests

Notions importantes du projet

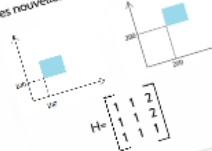
Traitement d'image

- Librairie OpenCV
- SIFT (Scale Invariant Feature Transform)
- Algorithme Ransac



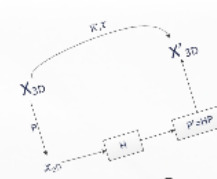
Homographie

- Transformation linéaire 2D
- Matrice de rotation + matrice de translation
- Calcul des nouvelles coordonnées

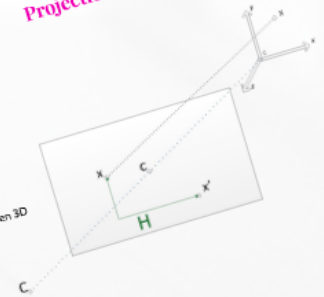


$$H = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}$$

Projection et recalage

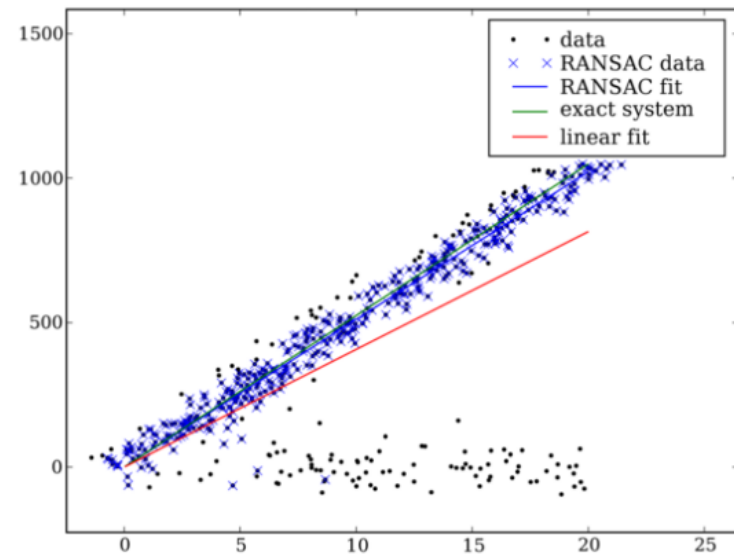


1. Environnement réel 3D
2. Enregistrement camera 2D
3. Projection dans un monde virtuel en 3D
4. Affichage à l'écran en 2D



Traitement d'image

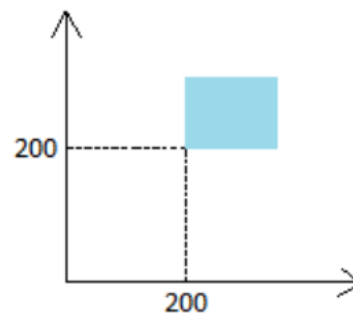
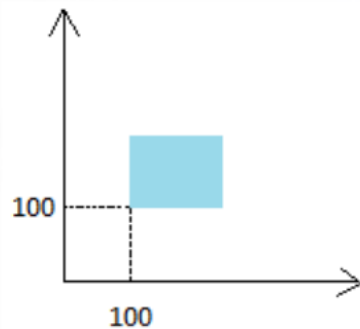
- Librairie Opencv
- SIFT (Scale Invariant Feature Transform)
- Algorithme Ransac



Homographie

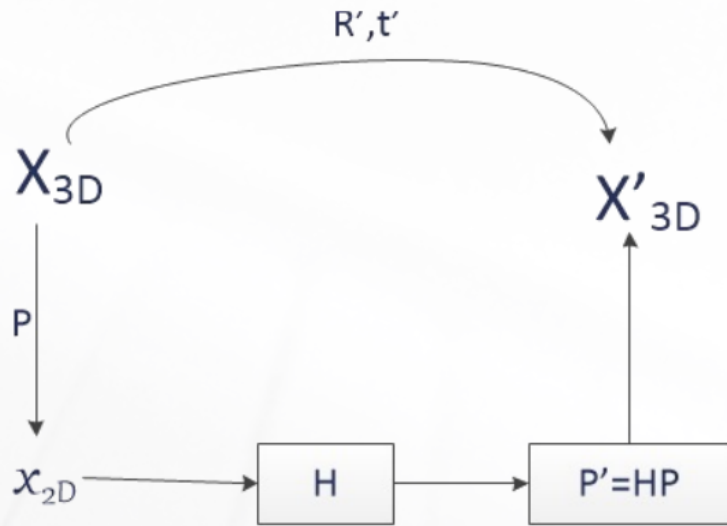
- Transformation linéaire 2D
- Matrice de rotation + matrice de translation
- Calcul des nouvelles coordonnées

$$H = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}$$

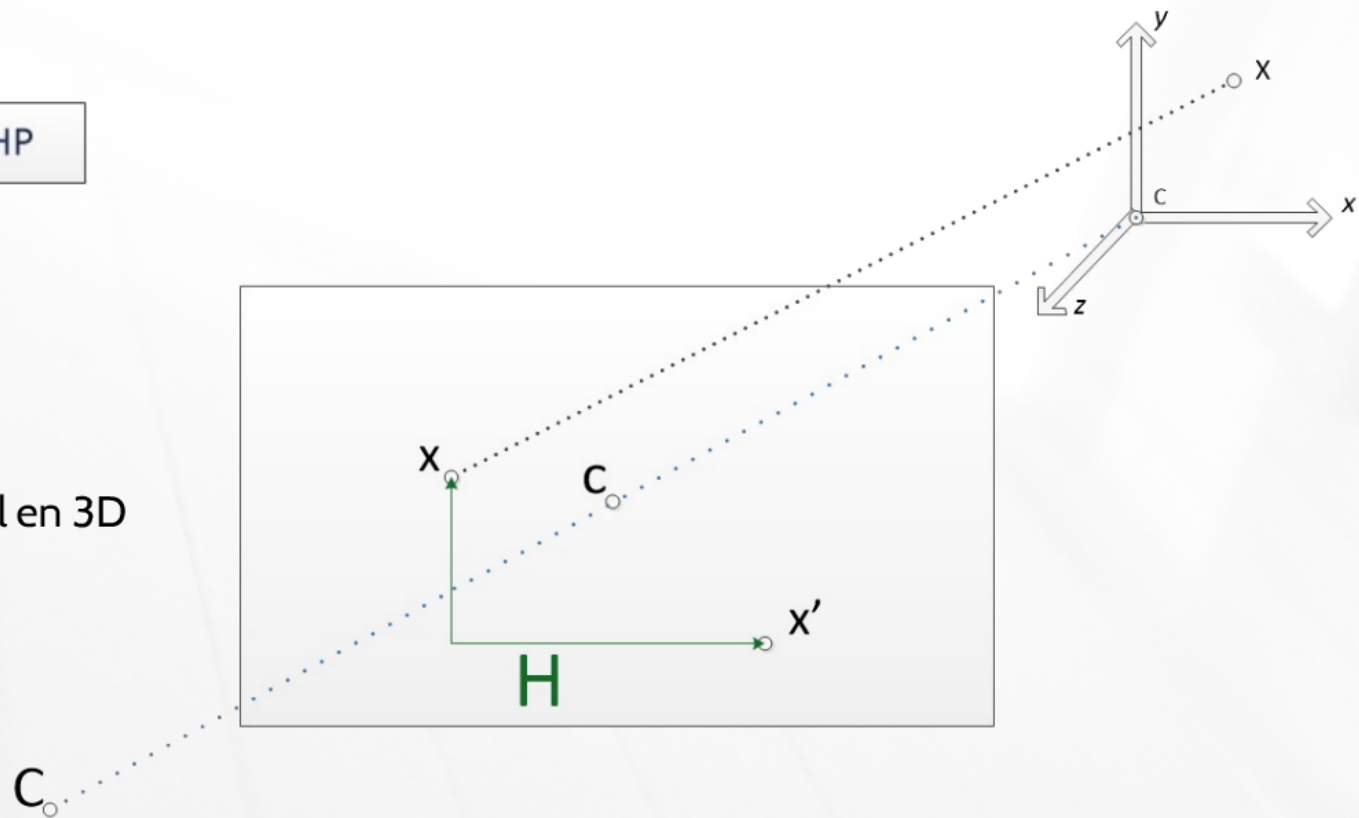


$$H = \begin{bmatrix} 1 & 1 & 2 \\ 1 & 1 & 2 \\ 1 & 1 & 1 \end{bmatrix}$$

Projection et recalage

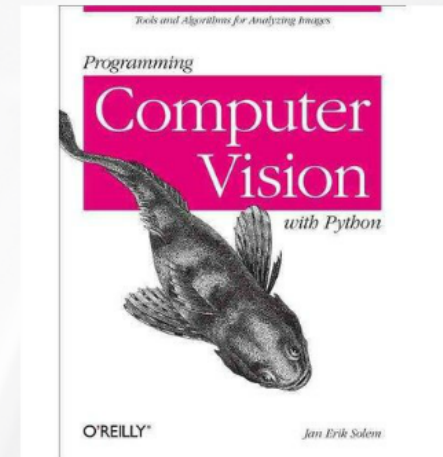


1. Environnement réel 3D
2. Enregistrement caméra 2D
3. Projection dans un monde virtuel en 3D
4. Affichage à l'écran en 2D



Outils et méthodes

- "Computer vision" de J.E Solem
- Langage Python
 - Multi-plateformes
 - Typage dynamique fort
 - Développement web
 - Orienté objet
 - Gestion des exceptions
 - Modélisation 3D



Résultats : les tests finaux

Test1.py

pygame OpenGL

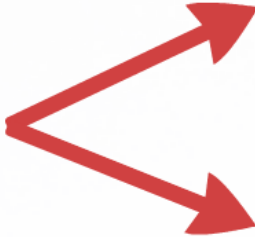
GL_MODELVIEW
GL_PROJECTION

$$\begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix}$$


Test2.py

image = cam.getImage() draw_background(image)

draw_cube(0.1xy)



Test5.py

box_cam1, box_cam2, H = algorithm_sift(K)

Test3.py

- detectAndTrack()()
- match_template(dst, src)
- RotateModel()
- H_from_sisac()()()()()()
- find_img()()()()()

Résultats obtenus :



Test4.py

- my_camera()()()()
- cube_poses()
- Camera()
- project()

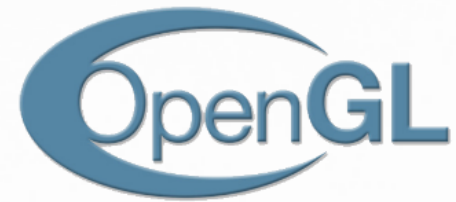
Résultats obtenus :

Test6.py

- Video()
- CreateAdequate()
- numpyimg2opencvimg()
- operation()
- VideoFrame()
- numpyimg2opencvimg()

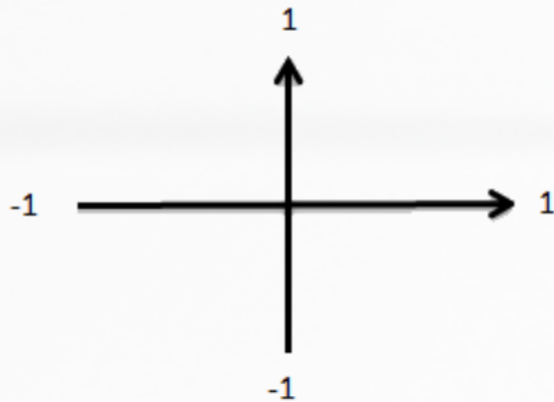
Résultats obtenus : Vidéo comprenant l'objet 3D qui suit le mouvement de l'image

Test1.py

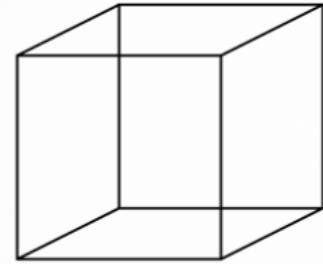


GL_MODELVIEW
GL_PROJECTION

$$\begin{bmatrix} R & \mathbf{t} \\ 0 & 1 \end{bmatrix}$$

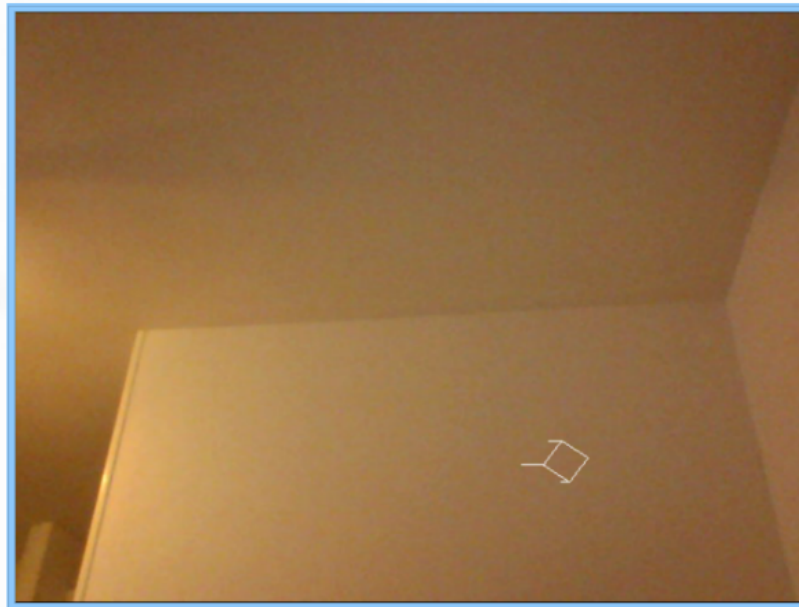


Test2.py



```
image = cam.getImage()
```

```
draw_background(image)
```



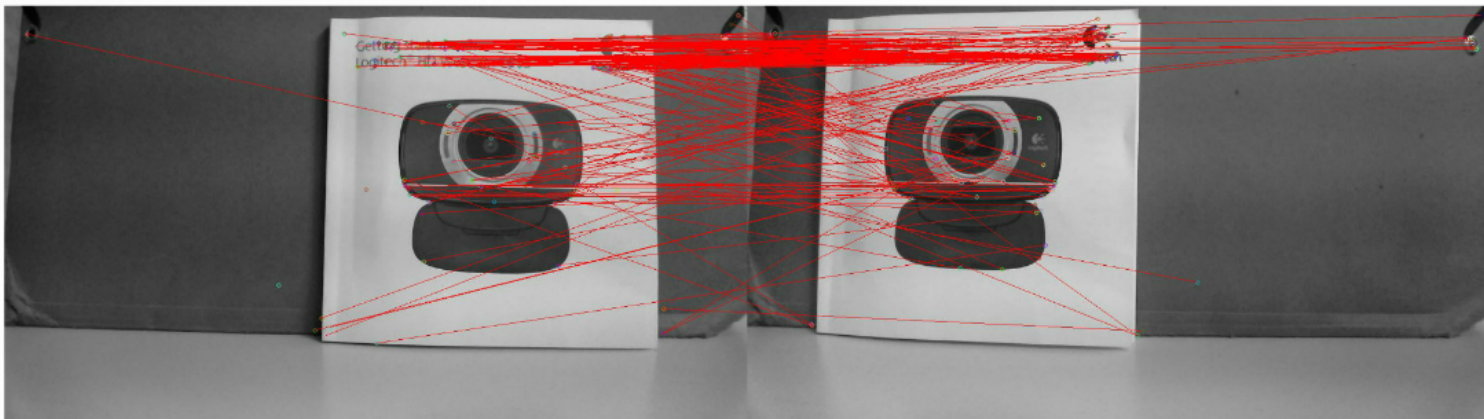
```
draw_cube(0.1,x,y)
```

Test3.py

- `detectAndCompute(im)`
- `match_twosided(des1,des2)`
- `RansacModel()`
- `H_from_ransac(fp,tp,model)`
- `line(im,pt1,pt2,color)`



Résultats obtenus :



Test4.py

- my_calibration(h,w)
- cube_points()
- Camera()
- project()



Résultats obtenus :



Test5.py



`box_cam1, box_cam2, H = algorithm_sift(K)`



Test6.py

- Video()
- CreateVideoWriter()
- numpyrgb2opencvrgb()
- operation()
- WriteFrame()
- numpyrgb2opencvrgb



scikit-image
image processing in python

Résultats attendus : Vidéo comprenant l'objet 3D qui suit le mouvement de l'image

Conclusion

Axes d'évolution

Difficultés rencontrées

