

Initiation à la Vision par Ordinateur

Bertrand VANDEPORTAELE

Laboratoire d'Analyse et d'Architecture des Systèmes
Groupe Robotique Action et Perception

bertrand.vandeportaele@iut-tlse3.fr
Licence Pro Robotique

March 18, 2021

Sommaire

- 1 Introduction
- 2 Capteurs, plateformes et calculateur
 - Capteurs
 - Plateformes
 - Calculateurs
- 3 Traitement d'image
- 4 Géométrie pour la vision
 - Modélisation Géométrique
 - Géométrie projective
 - Étalonnage Géométrique
 - Stéréovision
 - Applications
- 5 Considérations

Section 1

Introduction

Objectifs

- La perception pour la robotique est un vaste domaine.
- La vision par ordinateur (au sens large) (Computer Vision) recouvre:
 - Traitement du signal
 - 2D: image
 - 3D: séquence d'images
 - Géométrie
 - Modélisation de l'incertitude
 - I.A. et apprentissage
 - Sémantique
 - Couplage Action et Perception
 - ...
- Ce cours est une introduction

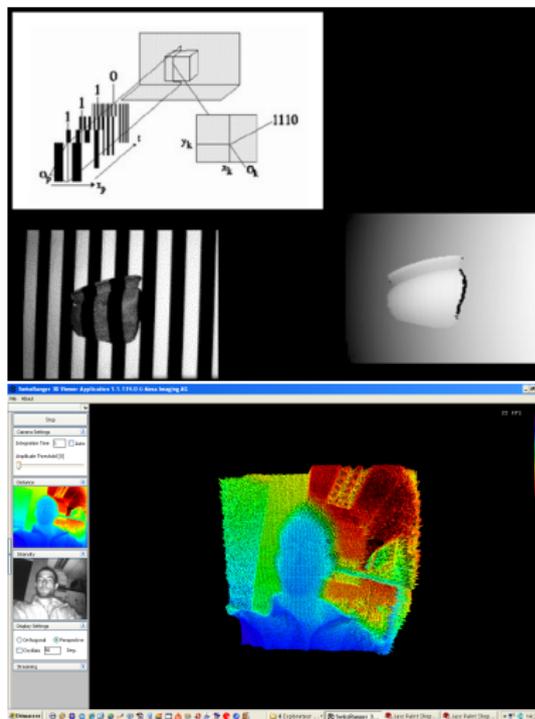
Section 2

Capteurs, plateformes et calculateur



Perception pour la robotique: Capteurs Actifs

- Emet sa propre source de lumière et mesure celle réfléchiée par la scène
 - Vision avec projection de lumière structurée ou non
 - Temps de vol (TOF)
 - Ultrasons
 - Télémètres Lidar à balayage
 - Capteurs matriciels RGB-D (Kinect 2), plutôt en intérieur
- Shape From Shading



Perception pour la robotique

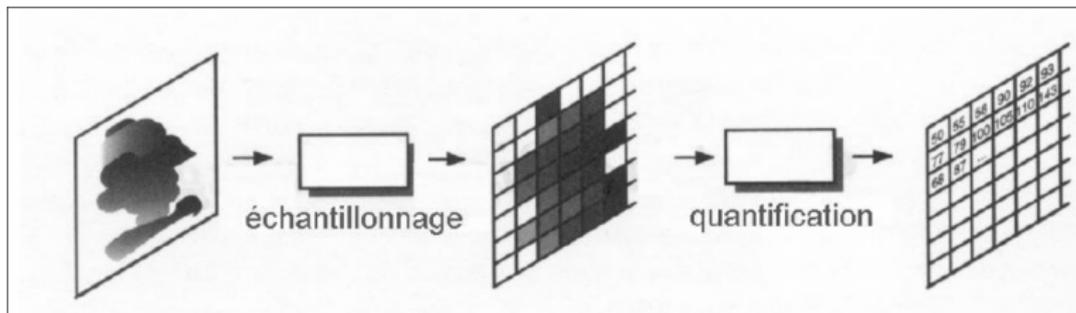
- Déplacement et orientation (Pan-Tilt)
- Zoom motorisé
- Différentes contraintes
 - Prix, poids, compacité
 - Grande dynamique
 - Cadence, latence
 - Robustesse



Section 3

Traitement d'image

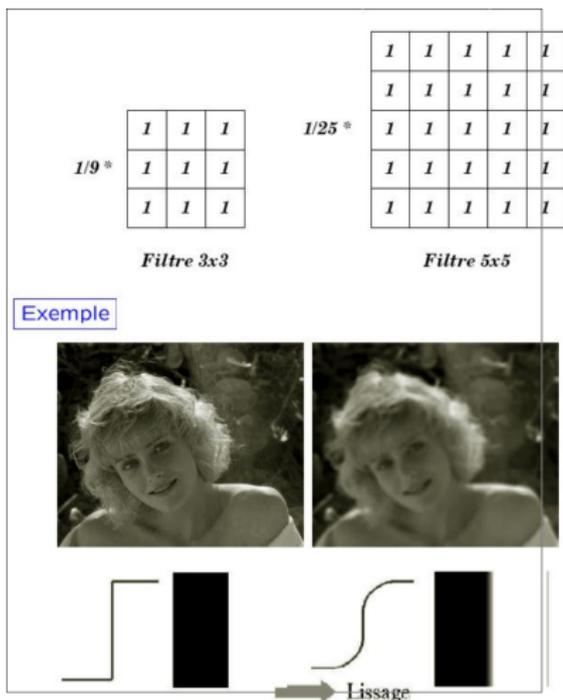
Qu'est ce qu'une image numérique



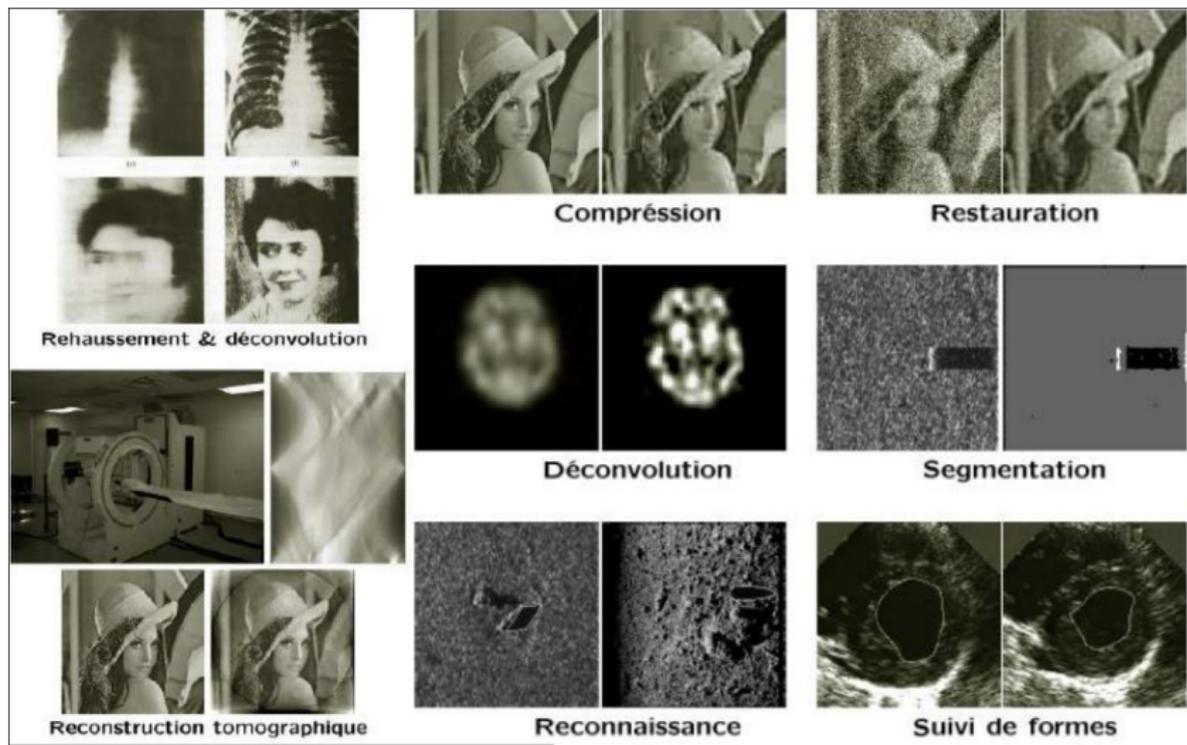
1

¹http://www.isima.fr/~vbarra/IMG/pdf/diapos_cours.pdf

Exemple de traitement: filtrage

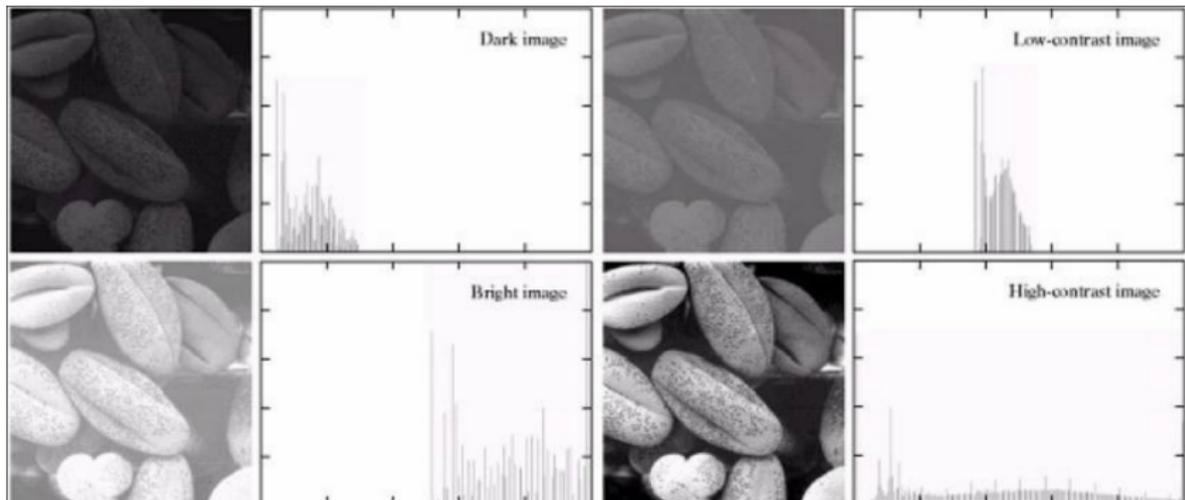


Autres exemples de traitements



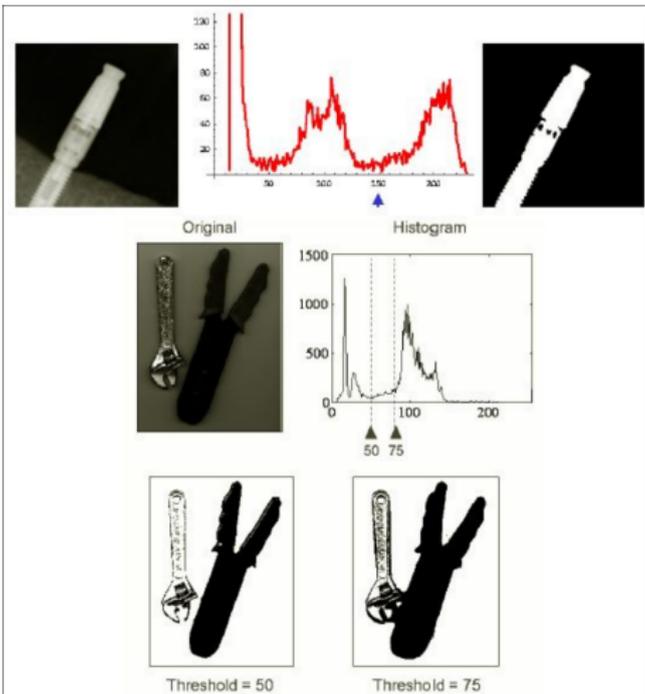
³http://www.isima.fr/~vbarra/IMG/pdf/diapos_cours.pdf

Histogramme



- Correction par calcul et application d'une table de correspondance (LUT)
- N'améliore l'image que pour la visualisation

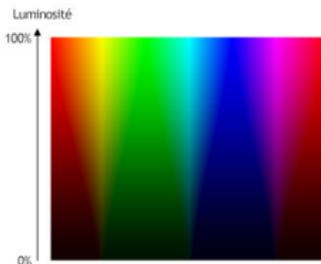
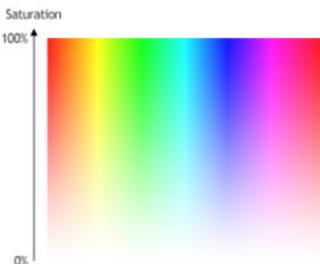
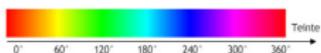
Seuillage



5

Couleur, niveau de gris, image binaire

- GRAY
- RGB, BGR...
- RGBA
- Mais aussi d'autres espaces colorimétriques
 - Teinte Saturation Luminosité
 - YUV
 - CMJN
 - ...



Opérateur de morphologie mathématique

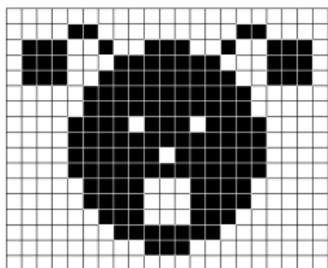
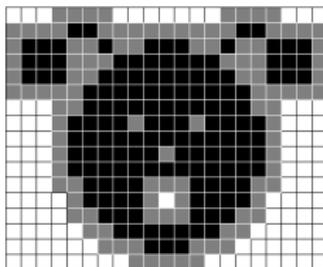
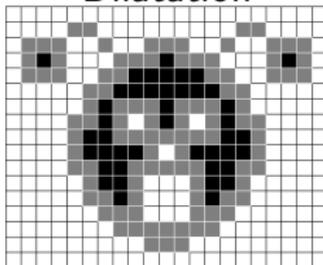


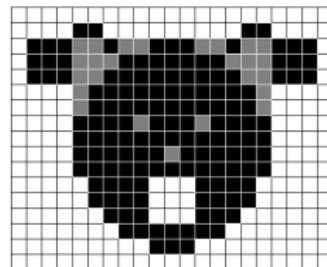
image originale



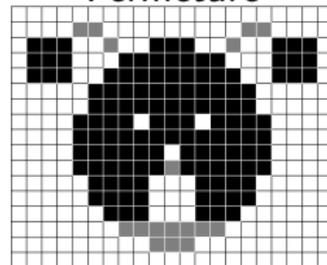
Dilatation



Erosion



Fermeture



Ouverture

Segmentation et étiquetage d'image

Attribution d'un numéro à chaque pixel = label



Image d'origine



Image segmentée

7

⁷<http://www.umiacs.umd.edu/~mishraka/activeSeg.html>

Extraction et appariement de points

- Détecteur de points invariants à des changements:
 - de point de vue (répétabilité), d'échelle, de rotation, de transformation affine, d'illumination
- Ces points d'intérêt sont généralement des coins: Harris, KLT, SIFT, FAST, ORB, BRIEF...
- On associe parfois un descripteur au point pour pouvoir l'apparier avec une observation dans une autre image



8

⁸http://www.roborealm.com/help/Harris_dest.jpg

Extraction de contours et segments



Extraction de structures géométriques



Exemple de détection de coniques

Extraction de plans



- Analyse des points de fuite associés aux directions détectées
- Ce n'est plus du traitement d'image, mais de la Vision, traitement qui nécessite une analyse plus globale, à " plus haut niveau"

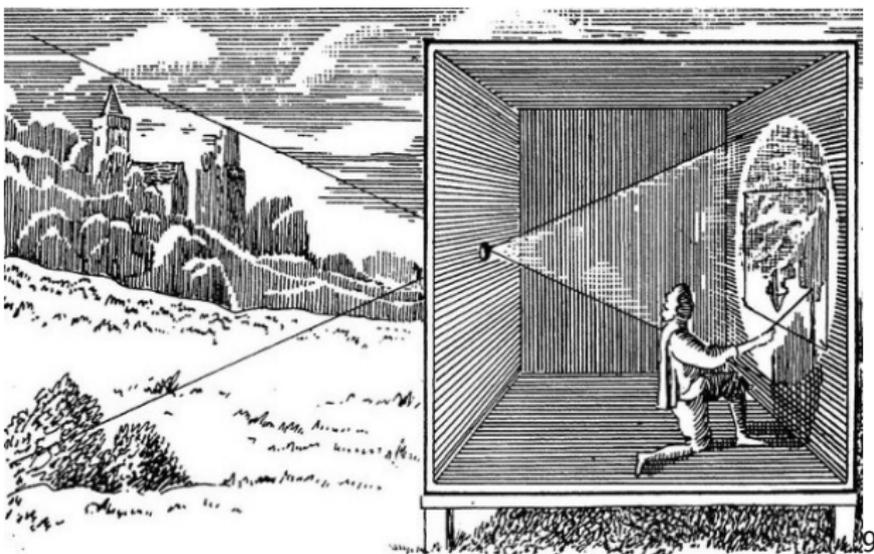
Section 4

Géométrie pour la vision

Modèle paramétrique: Exemple de la balance 1/2

- On souhaite mesurer la masse m d'un objet
- La force générée est le poids P , g est l'accélération à un endroit donné ($g \approx 9,8m/s^2$): $P = g.m$
- Longueur L d'une éprouve se déformant sous l'effet du poids:
 $L = L_0 + K_1.P$
- Montage dynamomètre piézoélectrique fournit tension V dont la valeur varie linéairement avec la déformation:
 $V = V_i + K_2.L$
- Quantification à l'aide d'un Convertisseur Analogique Numérique (version simplifiée): $N = (V_{out} - V_0).K_3.V$
- Au final N est observé comme fonction paramétrique de m et des différents paramètres
 - Modèle direct: $N = f(m)$
 - Modèle inverse (le modèle direct est inversible, il n'y a pas de "perte" d'information): $m = g(N)$

Les caméras



- Taille du trou d'épingle (pinhole): Image floue ou sombre
- Projection perspective
- Centre de projection=centre optique

⁹<http://www.albertdemun.fr/college/art2mun/>

Les caméras



- Optiques fisheye jusqu'à 240 x360 degrés de champ de vision
- Très courte distance focale
- Attention aux abhérations chromatiques
- Projection sur un plan?

Les caméras



- Systèmes catadioptriques (lentilles + miroir(s))
 - Augmentation du champ de vision

Les caméras

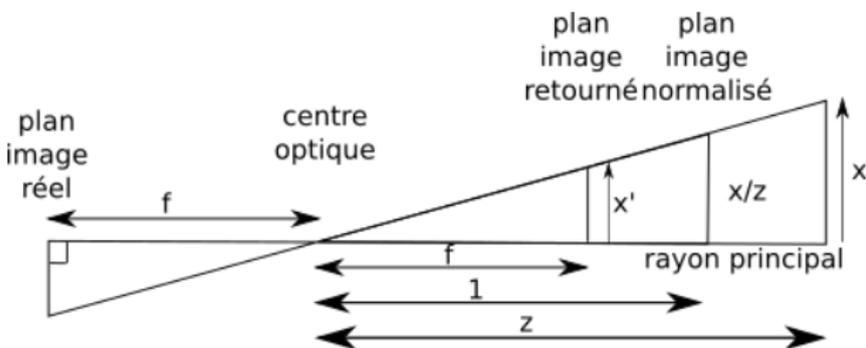


- Systèmes catadioptriques (lentilles + miroir(s))
 - Stéréovision avec une seule caméra observant différents miroirs
 - Pas de problème de synchronisation

Les caméras numériques

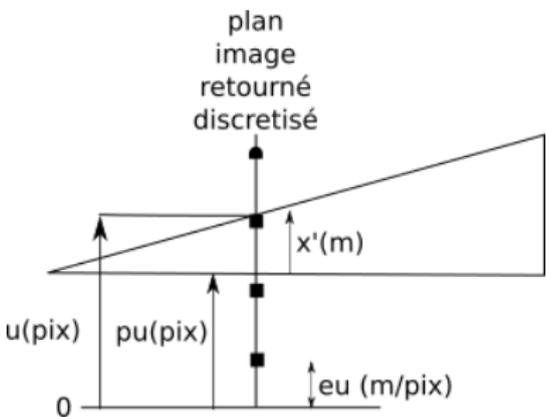
- Discrétisation du plan image en pixel
- Résolution en pixel
- Fréquence d'acquisition (FPS)
- Bus de communication
- Type d'obturateur:
 - Global (global shutter)
 - Tournant (rolling shutter), webcam et obturateur mécanique à double rideaux
- Dynamique: quantification de la quantité de lumière reçue pendant la durée d'intégration sur un nombre de bits donné
 - réponse type linéaire
 - réponse type log
 - High Dynamic Range:
 - capteur dédié
 - caméras DVS (120dB)
 - intégration de plusieurs images avec des expositions différentes (enfuse)

Modèle paramétrique: La caméra



- Le rayon principal (axe optique) est l'orthogonal au plan image passant par le centre optique
- D'après Thalès: $x' = f \cdot x / z$ (tout en unité métrique)
- La caméra est un appareil de mesure d'angle

Modèle paramétrique: La caméra

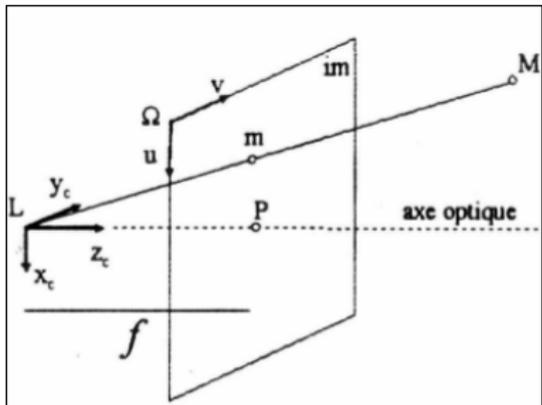


- Discretisation du plan image:

$$u = (x'/e_u) + p_u = f \cdot e_u^{-1} \cdot x/z + p_u = \alpha_u \cdot x/z + p_u$$

- u position en pixel de la projection dans l'image
- $\alpha_u = f \cdot e_u^{-1}$ focale pixellique
- p_u point principal

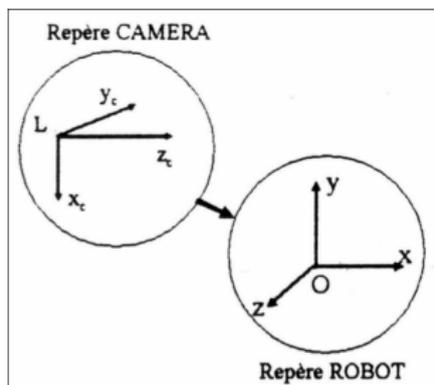
Modèle paramétrique: La caméra



L'image $m = (w \ u \ w)^T$ d'un point $M = (x_c \ y_c \ z_c \ 1)^T$ exprimé dans le repère caméra est obtenue:

$$\begin{pmatrix} w \cdot u \\ w \cdot v \\ w \end{pmatrix} = \begin{pmatrix} \alpha_u & 0 & p_u & 0 \\ 0 & \alpha_v & p_v & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} x_c \\ y_c \\ z_c \\ 1 \end{pmatrix}$$

Modèle paramétrique: La caméra



Changement de repère entre repère de travail (ROBOT) et repère

$$\begin{pmatrix} x_c \\ y_c \\ z_c \\ 1 \end{pmatrix} = \begin{pmatrix} R & t \\ 0_3^T & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \text{ avec } t = -R \cdot \begin{pmatrix} l_x \\ l_y \\ l_z \end{pmatrix}$$

Position du centre optique dans le repère ROBOT: $(l_x \ l_y \ l_z)^T$

Modèle paramétrique: La caméra

Combinaison matrice de projection et changement de repère:

L'image $m = (wu \ wv \ w)^T$ d'un point $M = (x \ y \ z \ 1)^T$ exprimé dans le repère de travail est obtenue à partir de la matrice caméra $\tilde{C} = (C \ \tilde{c}_4)$.

$$\tilde{C} = \begin{bmatrix} \alpha_u r_1^T + p_u r_3^T & \alpha_u t_x + p_u t_z \\ \alpha_v r_2^T + p_v r_3^T & \alpha_v t_y + p_v t_z \\ r_3^T & t_z \end{bmatrix} ; m = \tilde{C} M$$

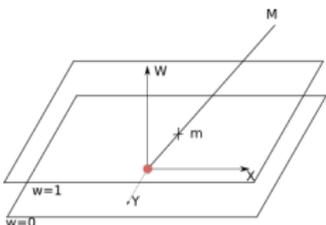
- Paramètres intrinsèques: $\alpha_u, \alpha_v, p_u, p_v$ (4ddl)
- Paramètres extrinsèques: R (3 ddl), t (3ddl)

Paramétrisation des rotations

- Quaternion
- Rodrigues
- Euler
- Matrice
- En général non commutatif sauf pour les rotations infinitésimales

Espaces projectifs

Définition: $\mathbb{P}^n = \mathbb{R}^{n+1} - \{0\}$



- L'espace \mathbb{P}^2 contient les points $M = (x \ y \ w)^T$:
 - les points finis ($w \neq 0$) dont les coordonnées sont $m = \left(\frac{x}{w} \ \frac{y}{w} \right)^T$ dans \mathbb{R}^2
 - les points à l'infini dont la direction est $m = (x \ y)^T$ dans \mathbb{R}^2 .
- Permet d'appliquer des rotations et translations (et échelle et projection...) par produit matriciel
- Une Homographie est une application projective qui
 - conserve les droites
 - conserve les bi-rapports

Espaces projectifs

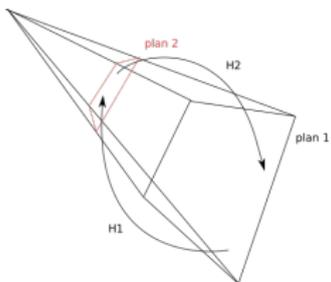
La matrice \tilde{C} est une homographie de \mathbb{P}^3 vers \mathbb{P}^2

$$\tilde{C} = \begin{bmatrix} \alpha_u r_1^T + p_u r_3^T & \alpha_u t_x + p_u t_z \\ \alpha_v r_2^T + p_v r_3^T & \alpha_v t_y + p_v t_z \\ r_3^T & t_z \end{bmatrix} ; m = \tilde{C} M$$

- Elle permet de projeter des points de \mathbb{R}^3
- Elle permet de projeter des directions de \mathbb{R}^3 , dont l'image s'appelle un point de fuite
- L'image d'un point (fini ou à l'infini) est un point (fini ou à l'infini)

Homographie

On considère souvent les homographies de \mathbb{P}^2 vers \mathbb{P}^2



- Ces homographies sont des matrices 3×3
- Si H_1 est inversible alors $H_2 = H_1^{-1}$
- Les homographies peuvent être composées par produit matriciel
- Cas particulier de matrice caméra qui s'applique à un plan unique
- Choix arbitraire du plan $z = 0$ possible en appliquant une rotation au repère de travail (éliminer la 3^o colonne de \tilde{C})

Distorsions

La caméra réelle n'est pas un trou d'épingle!

- Modélisation des distorsions radiales (Coussinet ou Barillet)
 - Fonction polynomiale (Taylor)
 - Appliquées dans le plan image normalisé
- Modélisation des distorsions tangentielles
- Choisir le nombre de degrés de liberté dans le modèle
- Rectification = obtenir une image qui aurait été acquise sans distorsion

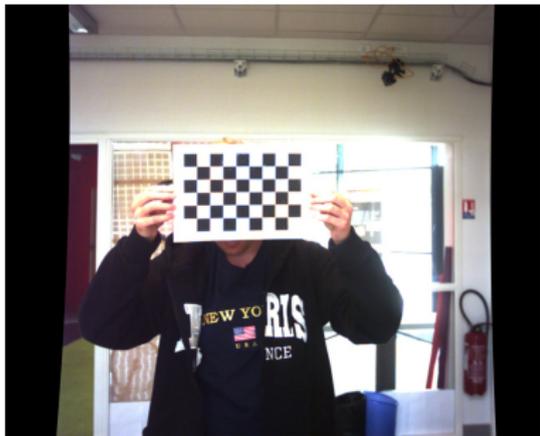


Estimation des paramètres du modèle

- Observation d'une mire (généralement 2D) connue à différentes poses
- Détection subpixellique des coins
- Minimisation de l'erreur de reprojection entre les points observés et les points prédits par le modèle paramétrique
- Grand nombre d'inconnues

Mires de différentes tailles

Permet d'échantillonner pour différentes distances: le volume de travail



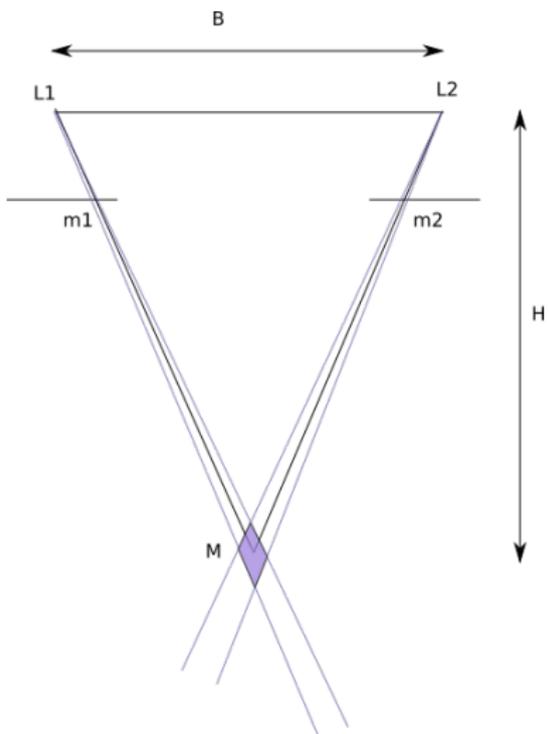
Petite mire pour acquérir des points proches



Grande mire pour acquérir des points lointains

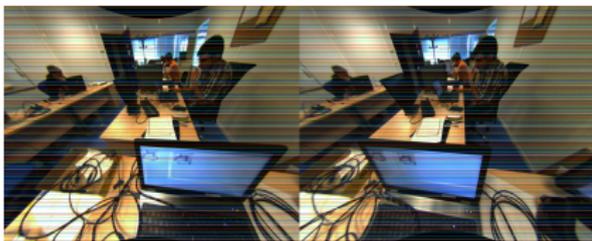
Principe de la Stéréovision

- Lever l'indétermination de distance grâce à plusieurs observations (par triangulation)
- Écart intraoculaire (Baseline)
- Rapport B/h
- Plus de 2 caméras pour:
 - Augmenter le champ de vision stéréo
 - Augmenter la précision
 - Augmenter la robustesse

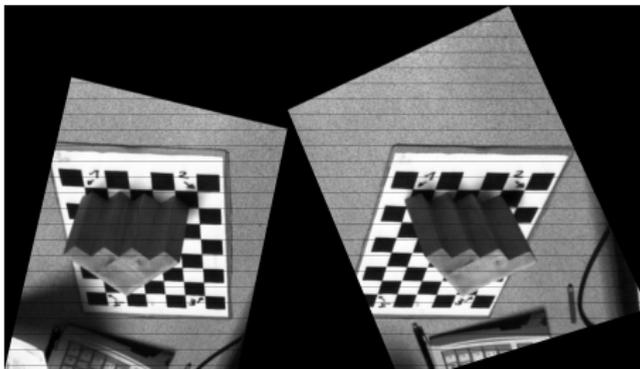
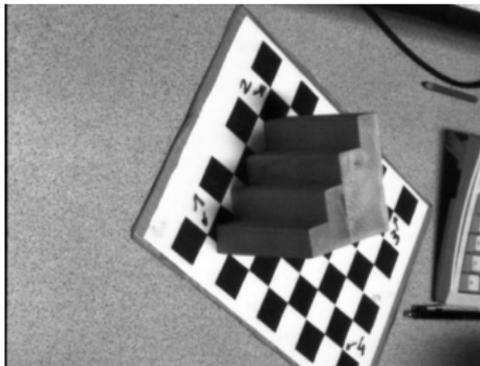
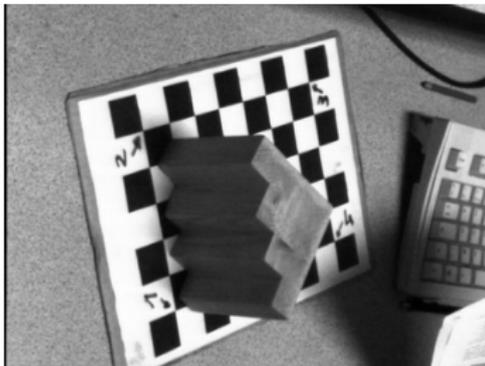


Rectification épipolaire

- Reprojection de la paire d'images sur un plan commun (Homographie)
- Epipoles à l'infini
- Rectification = obtenir une paire d'images sur laquelle les pixels homologues sont sur les mêmes lignes
- Appariement automatique (épars ou dense)



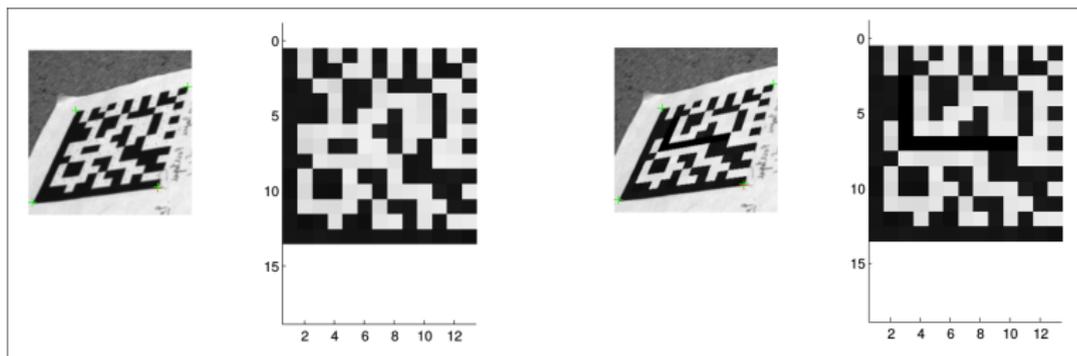
Rectification épipolaire



Pour jouer...

- Sujet de TP en matlab pour l'étalonnage monoculaire et stéréo <http://bvdp.free.fr/enseignements/>

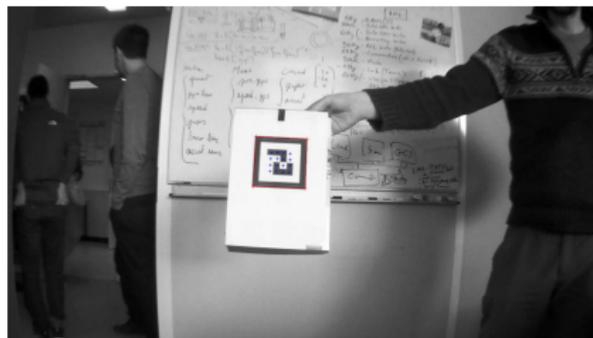
Homographies pour la rectification d'image



- Reprojection de l'image sur le plan du tag
- modification de l'image du tag
- Reprojection du tag sur le plan de l'image

Homographies pour la localisation de TAG

Algorithme Perspective-N-Points



- Détection automatique des coins du tag
- Vérification du code par reprojection
- Calcul des 6ddl de la pose à partir de l'observation des 4 coins

Mosaiquage d'images par Homographies

- Intégration des images sur un plan lors du déplacement/rotation de la caméra.
- Valide uniquement pour un plan, modèle 2D de l'environnement.



Section 5

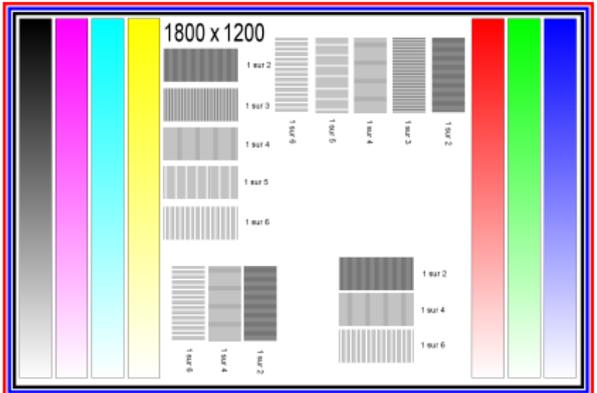
Considérations

Quelques faits généraux sur les traitements

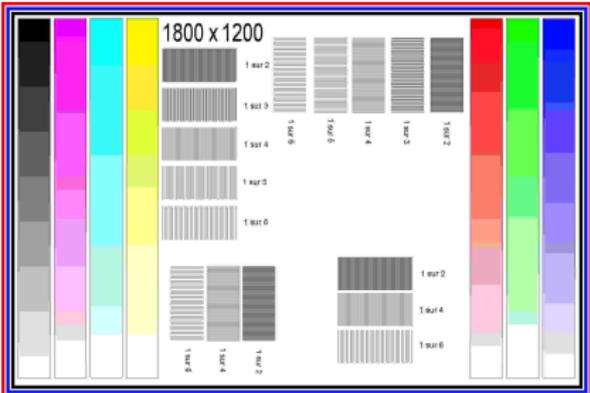
- Traitement d'image "lourd" en nombre d'opérations
- Attention aux fausses impressions:
 - Ce n'est pas parce que votre système visuel sait faire qu'un système automatique saura et inversement...
- Attention aux seuils à régler "à la main"

Importance de traiter des données brutes

Pas de compression avec perte (JPEG)



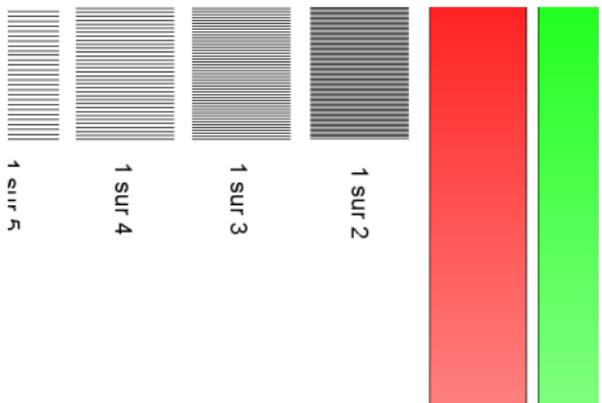
Originale



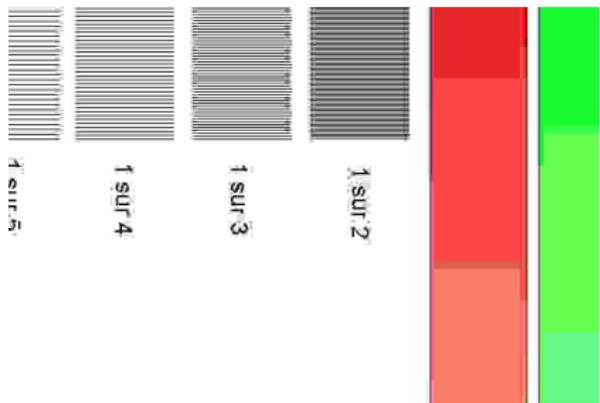
Comprimée JPEG

Importance de traiter des données brutes

Pas de compression avec perte (JPEG)



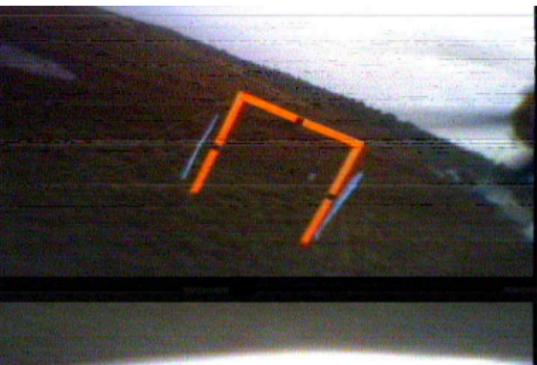
Originale



Comprimée JPEG

Choix de la caméra

Pas de flux video analogique



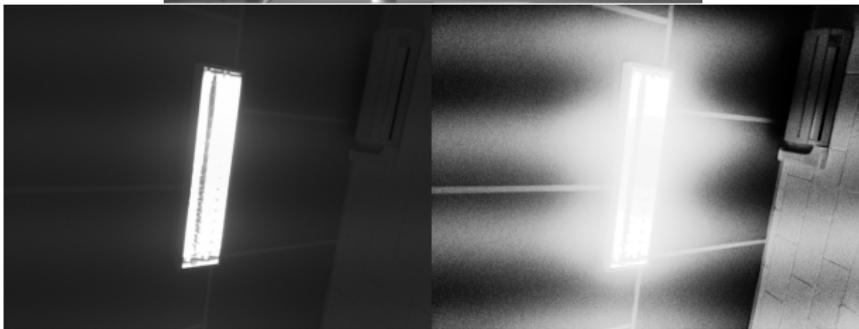
Choix de la caméra

Pas d'autofocus

Change les paramètres intrinsèques en permanence

Choix de la caméra

Pas de Rolling shutter



Bibliothèque

- Ne pas réinventer la roue... sauf si...
- OpenCV
 - Binding pour protoypage en Python
 - Attention aux versions (actuelle = 4.1.2)
- Eigen

Architecture logicielle

- La vision est une fonctionnalité d'un robot parmi plusieurs : motricité, communication, interaction, tâche principale, gestion de l'énergie, décision...
- Architecture modulaire
- MiddleWare ROS (Robot Operating System)
 - Modules (Noeuds) et Serveur
 - Publisher/Suscriber et Services

Démos

Et questions....