

Robotique Mobile

08 - Navigation utilisant une carte

David Filliat

Alexandre Chapoutot

Goran Frehse

prenom.nom@ensta-paris.fr

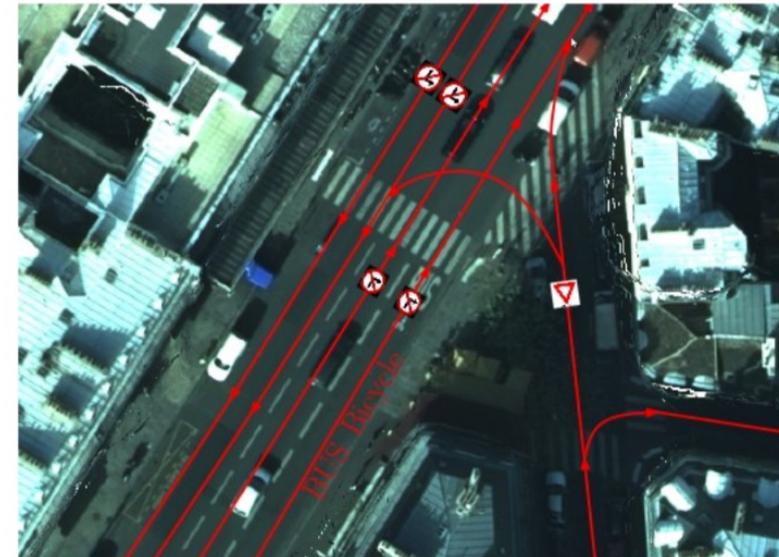
Navigation réactive

- Champ d'application restreint
- Très rapide
- Robuste (potentiellement ...)



Navigation utilisant une carte

- Construction d'un modèle de l'env.
- Planification de déplacements à long terme



Carte :

Modèle interne de l'environnement

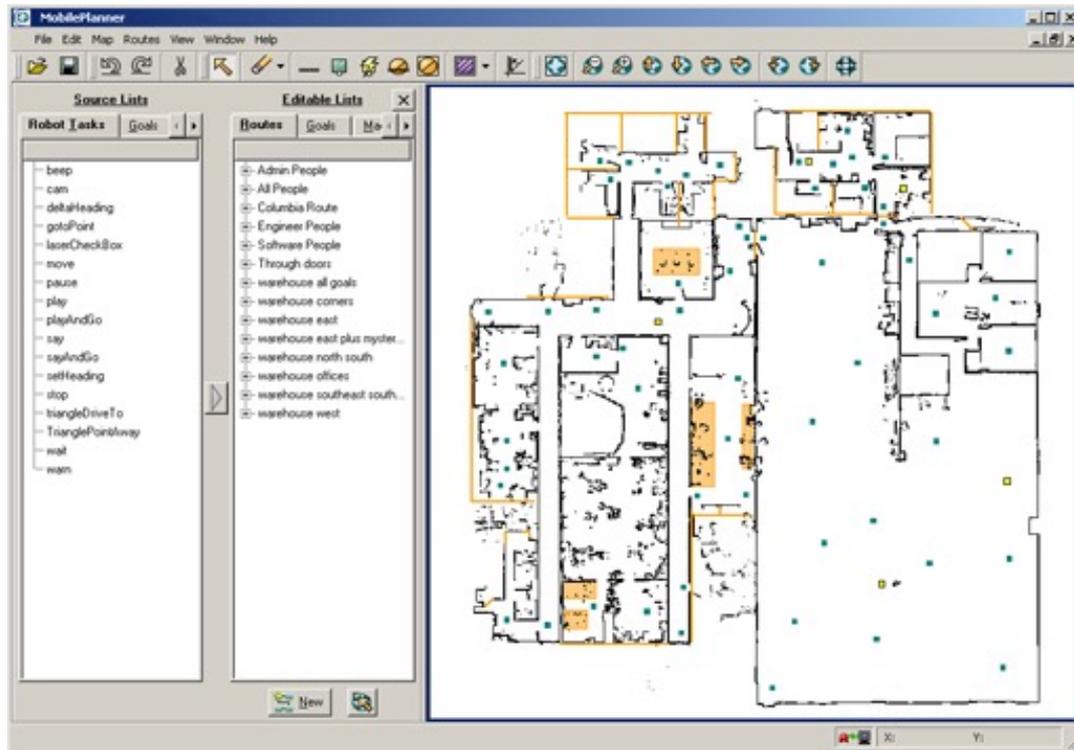
Mémorise *sous diverses formes* les informations proprioceptives et les perceptions

Permet de relier des perceptions à une position

- pour compenser la dérive de la proprioception
- pour résoudre le perceptual aliasing

Connaissance de l'environnement

- Pour repérer le robot, planifier des chemins
- Pour donner une vision globale à l'opérateur
- Pour la carte elle-même (relevé de plan d'architecte)



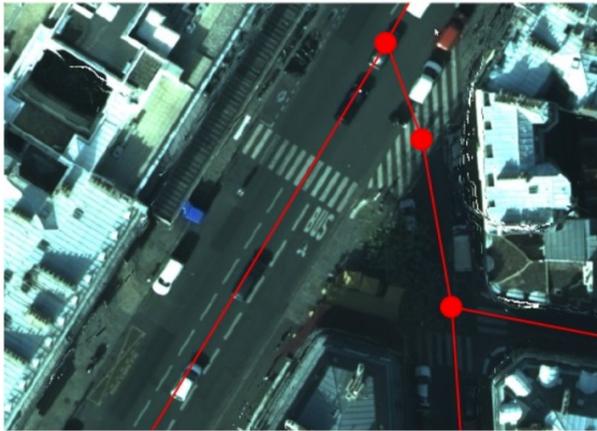
Connaissance de l'environnement

- Ex : cartographie 3D, Mobile Mapping Systems

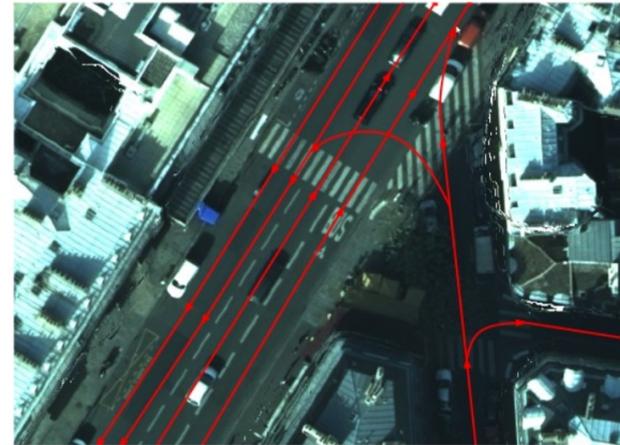


Connaissance de l'environnement

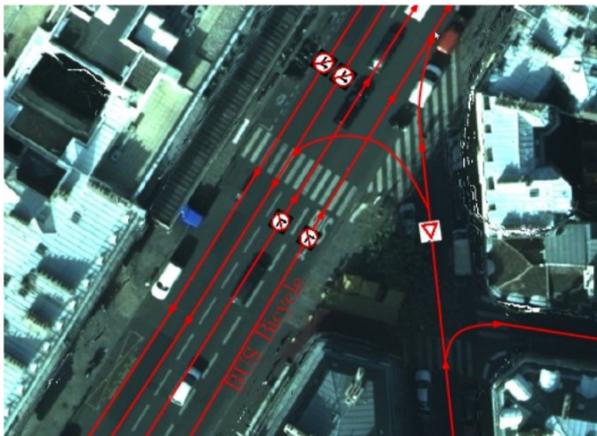
- Ex: cartes routières, pouvant être adaptées au véhicule autonome



1) Linear graph (macro-scale topology / geometry)



2) Linear graph (micro-scale topology / geometry)



3) Niveau 2 + traffic rules



4) Graphe des surfaces (échelle micro + règles de trafic)

Cartographie

- Construction de la carte

Localisation

- Estimer la position du robot dans une carte connue

Planification

- Calculer un chemin de la position courante jusqu'au but

Cartographie \longleftrightarrow Localisation



Robot autonome : Cartographie et localisation simultanée (**SLAM**)

Environnement plan (2D)

- plan de coupe d'un télémètre laser
- mobilité simplifiée

Environnement statique (mais bruité)

- pas de modélisation de portes ouvertes/fermés
- mais présence d'obstacles dynamiques possible (personnes, mobilier...) -> bruit

Estimation séparée de la position et de la direction

- notamment pour cartes topologiques (vision panoramique)
- pour des plates-formes holonomes
 - > boussole, gyroscopes, recalage

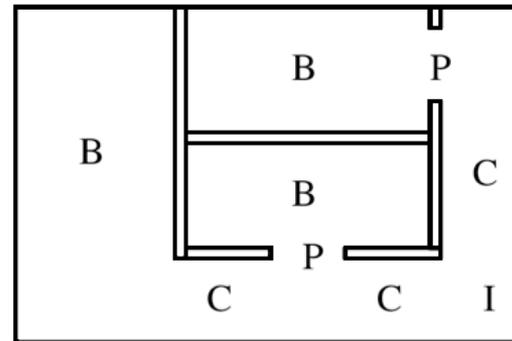
Représentations de l'environnement

Cartes topologiques

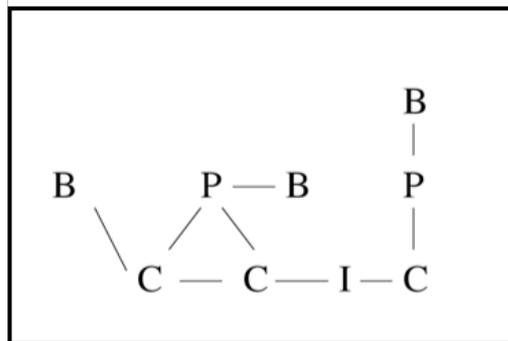
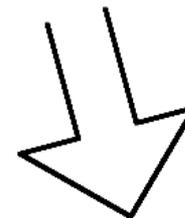
- Graphe de lieux et de transitions entre lieux
- perception sans modèle métrique

Cartes métriques

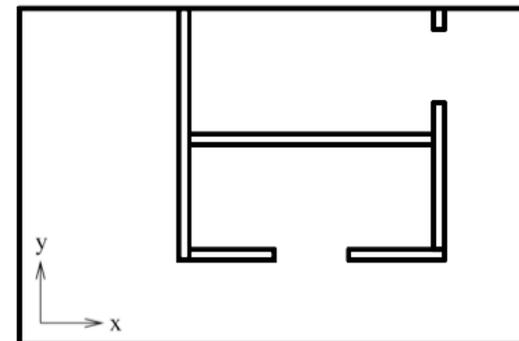
- Ensemble d'objets dans un espace commun
- perception avec modèle métrique



Environnement réel



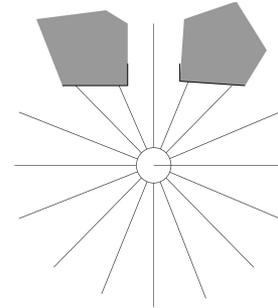
Carte topologique



Carte métrique

Nœuds définis par des perceptions

- Image panoramique
- Scan laser
- Configuration des murs



Nœuds placés selon différents critères .

- Correspondant à des perceptions données (porte, intersection...)
- Espacement régulier
- Classification non supervisée (ajout de nœud quand les perceptions ont assez changé)

Liens définissant un moyen de transition entre lieux

- Procédure à suivre (comportement, suivi de couloir...)
- Position relative
- Implicite par définition de la position des nœuds

Avantages :

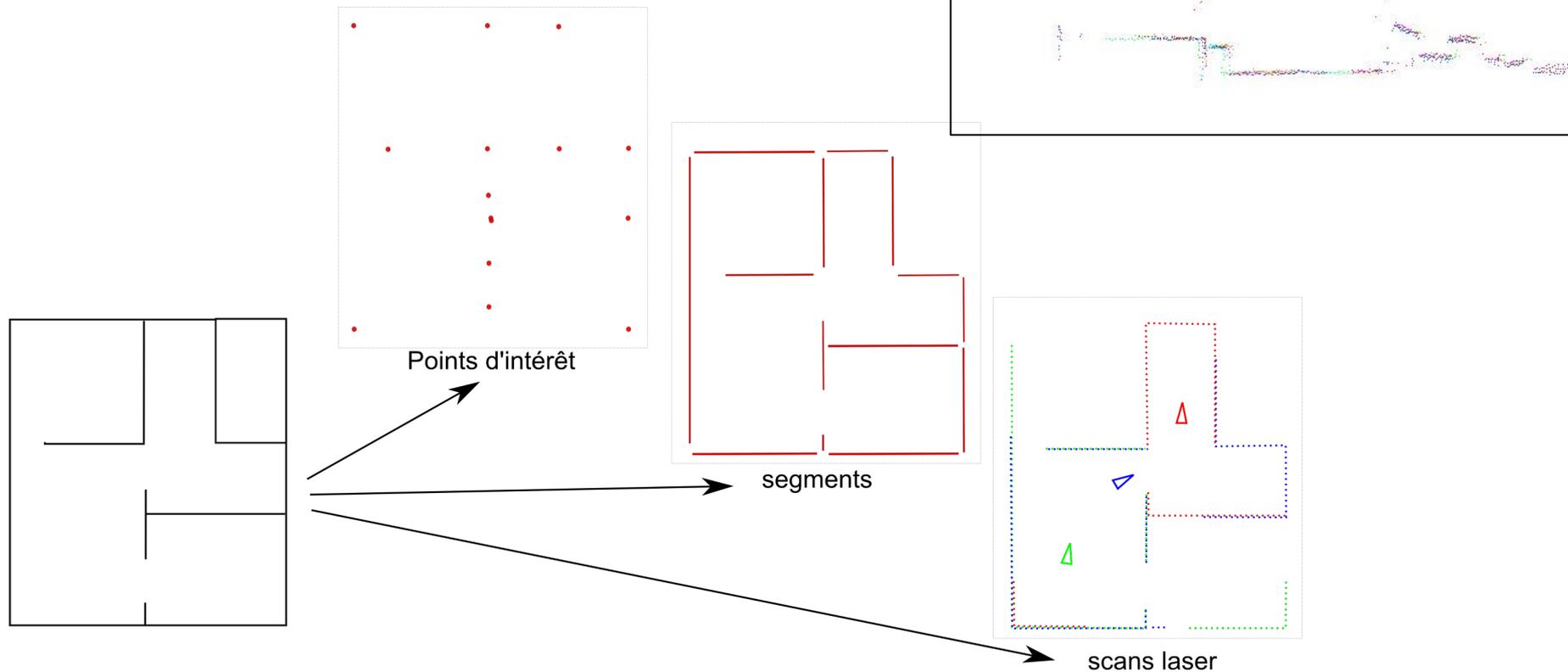
- Discrétisation pour la planification
- Proche des données capteurs
- Utilisation de capteurs sans modèles métriques
- Séparation des info proprioceptives et des perceptions (cartographie)
- Localisation grossière, mais rapide

Inconvénients :

- Séparation des info proprioceptives et des perceptions (localisation)
- Pas d'information sur les lieux non visités
- Pb de « perceptual aliasing » -> construction difficile
- Représentation très liée à un robot particulier

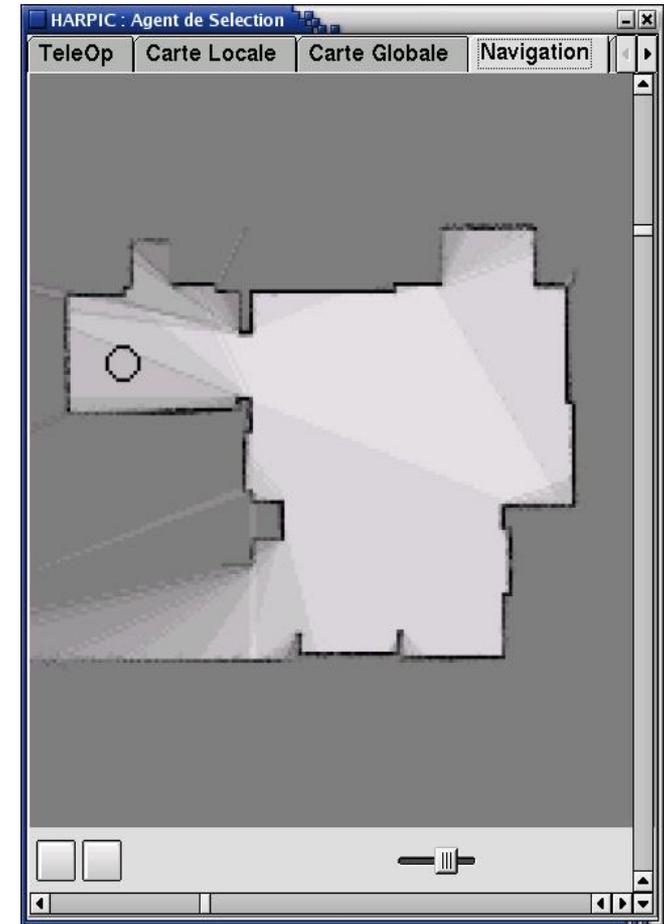
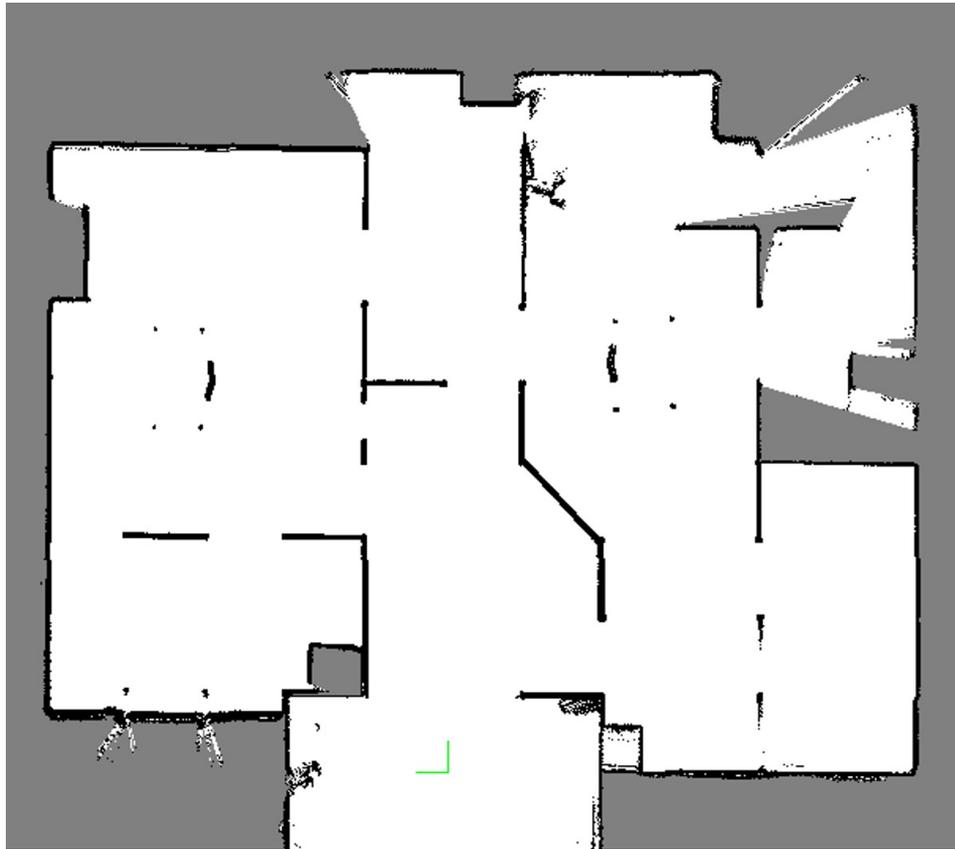
Ensemble d'objets dans un repère commun

- Points, segments, polygones
- Scans laser



Ensemble d'objets dans un repère commun

- Espace libre (grille d'occupation)



Avantages

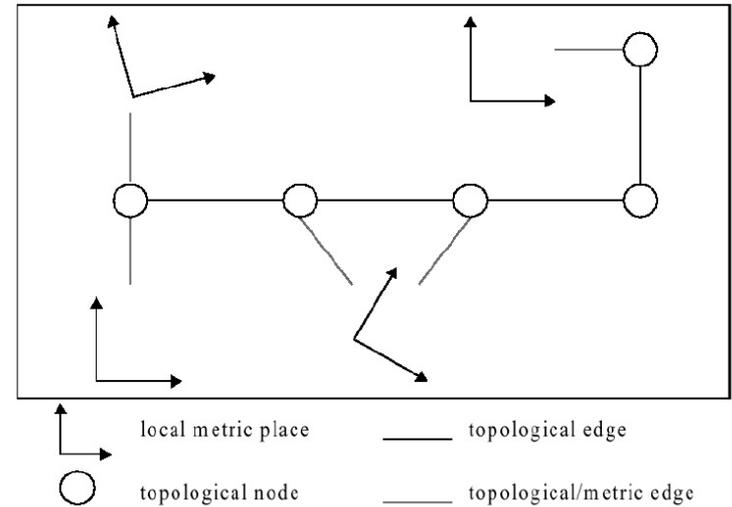
- Représentation de lieux non visités (mais « vus » de loin)
- Utilisation de la géométrie
- Localisation précise
- Représentation indépendante du robot (utilisation d'un modèle métrique)

Inconvénients

- Planification moins directe que pour les cartes topologiques
- Nécessite un modèle métrique
- Fusion au sein d'un même espace -> difficulté de modifications futures

Représentations mixtes (topo-métrique)

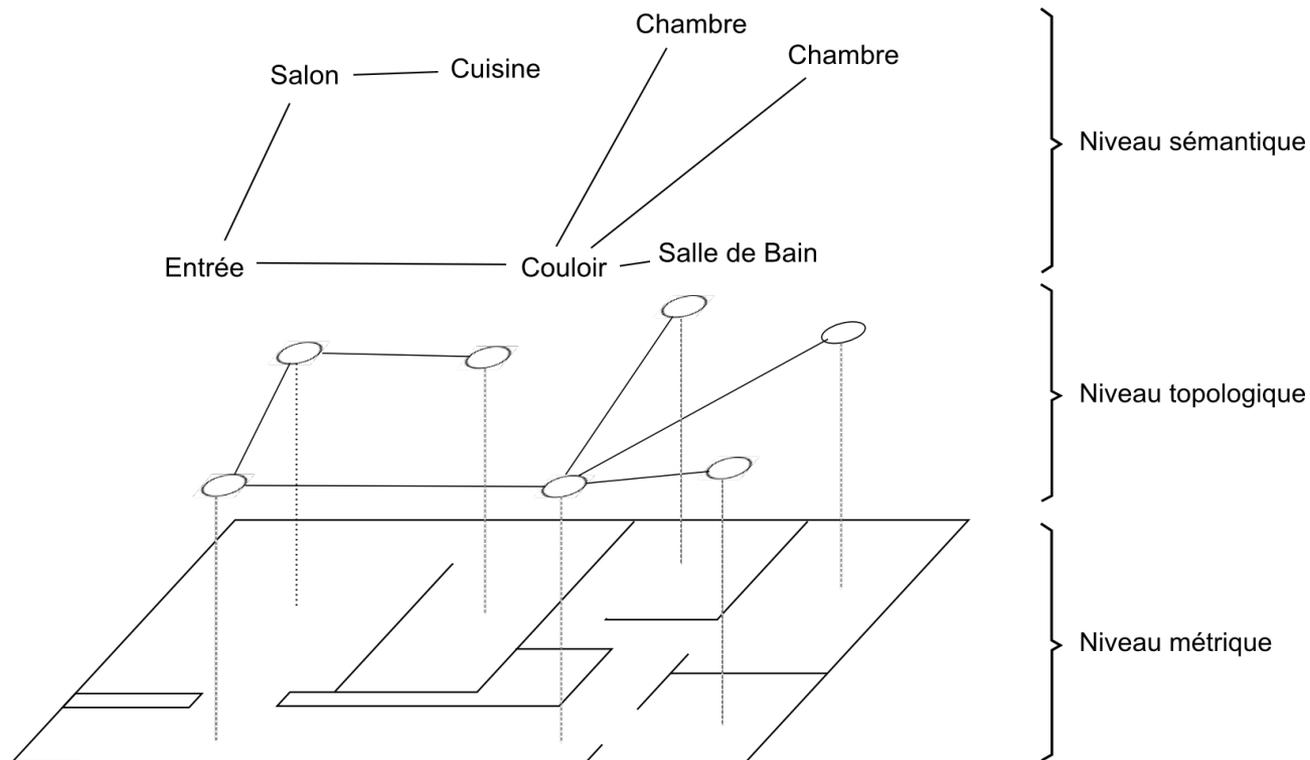
- Nœuds (images) avec positions relatives métriques



- Nœuds = cartes métriques locales

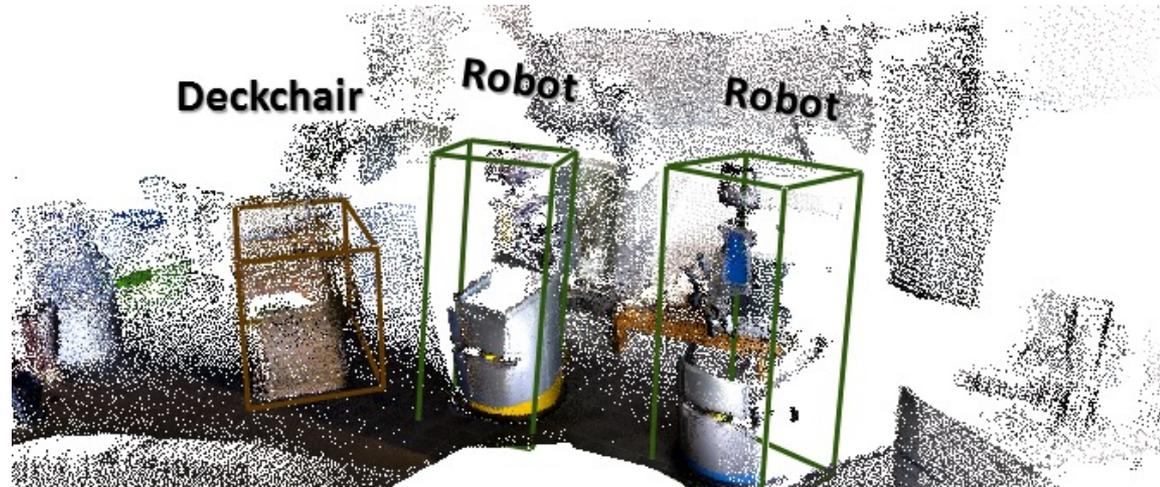
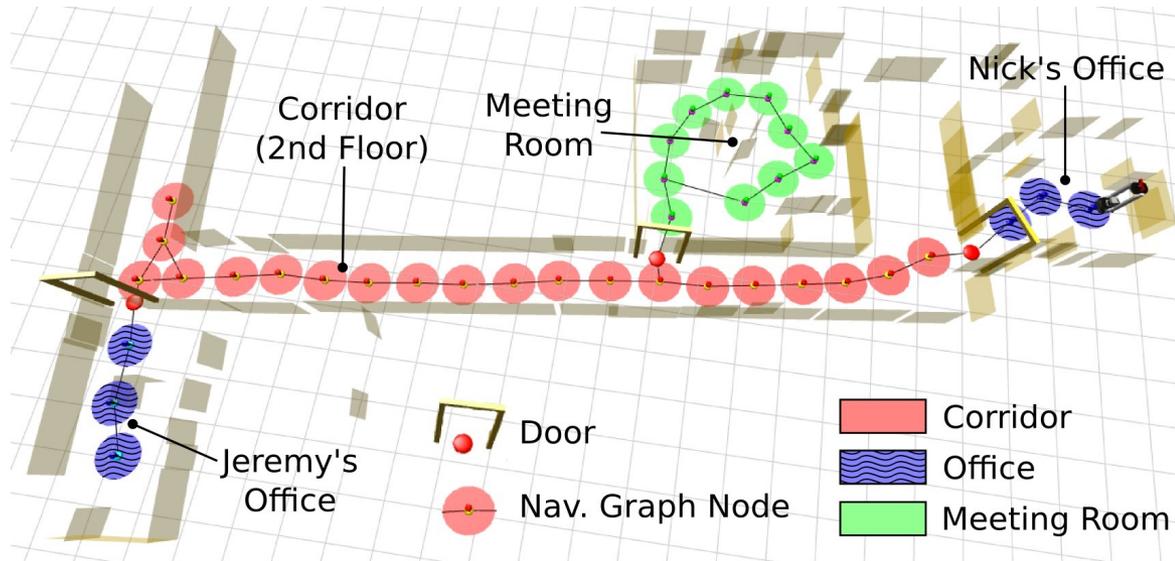
Hiérarchies de représentations

- Métrique
- Topologique
- Sémantique (type de lieux, objets ...)



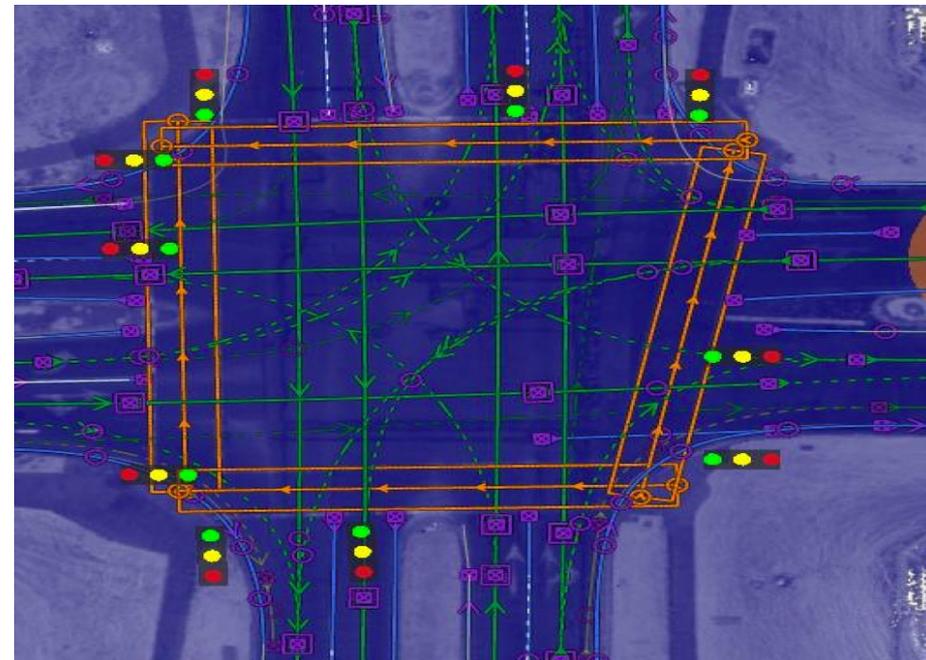
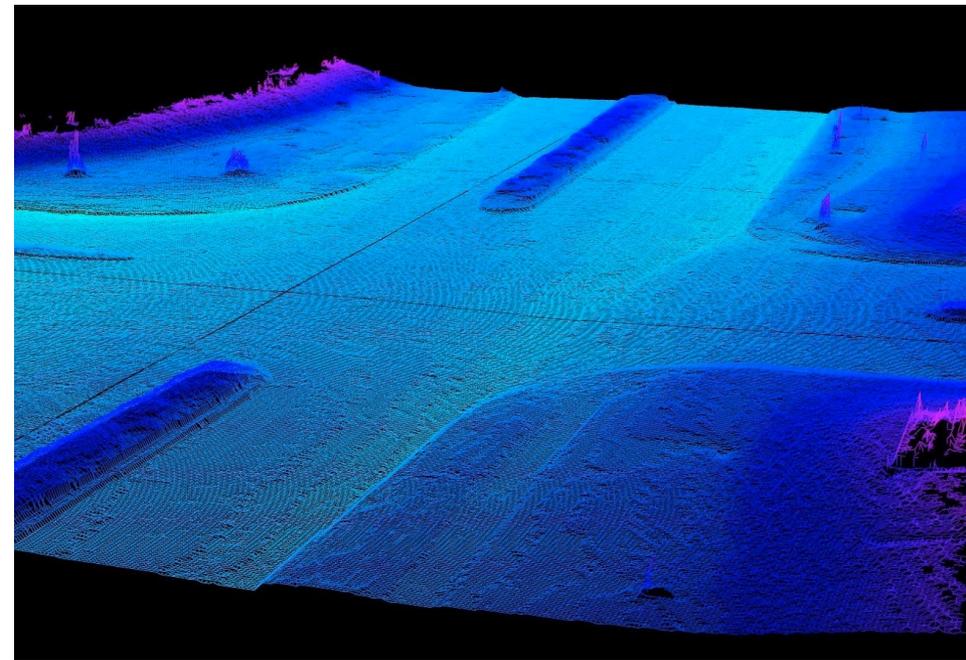
Cartes Sémantiques

A. Pronobis



Waymo HD map

- Nuage de points, doit permettre de se localiser à 10 centimètres
- Infos topologiques sur les routes, les intersections ...
- Info sémantiques des feux, panneaux, priorités ...
- Construit semi-automatiquement à partir de véhicules non autonomes



Navigation utilisant une carte - En résumé

- Les cartes sont des modèles internes de l'environnement qui relient perceptions et proprioceptions, utilisées pour la navigation mais aussi pour d'autres applications
- Trois problèmes : **Cartographie, Localisation et Planification**
- Cartographie et Localisation sont inter-dépendantes, on parle de **Simultaneous Localization and Mapping (SLAM)**
- Deux types de cartes principaux : **topologique et métrique** avec des nombreux intermédiaires et éventuellement des informations sémantiques

Robotique Mobile

09 – Localisation 01

David Filliat

Alexandre Chapoutot

Goran Frehse

prenom.nom@ensta-paris.fr

Données disponibles pour la localisation:

- **Perceptions :**

Problèmes de « perceptual aliasing » et de variabilité perceptuelle

- **Proprioception :**

Problème de dérive au cours du temps

→ **fusion** des informations

Localisation « locale »

- Estimation de la position par mise à jour d'une estimation initiale

Localisation globale

- Estimation de la position sans estimation initiale « lost robot »
- Estimation de la position avec une fausse estimation initiale « stolen robot »

3 catégories de méthodes correspondant à :

- 3 manières de traiter le « perceptual aliasing »
 - amélioration des perception
 - utilisation de la proprioception
- 3 manières d'utiliser les perceptions et la proprioception
 - corrections locales
 - estimations globales
- 3 capacités de localisation différentes

1. Inférence directe de position

- Utilisation des perceptions seules
- Améliorer le traitement des perceptions
- Localisation globale en environnement sans « *perceptual aliasing* »

2. Suivi de position

- Perceptions (hypothèses de position) + proprioception pour sélectionner une hypothèse
- Résoudre les ambiguïtés par sélection d'une position
- Localisation « locale » à *partir d'une position initiale connue*

3. Suivi d'hypothèses multiples

- Perceptions (hypothèses de position) + proprioception pour évaluer les probabilités des différentes hypothèses
- Résoudre les ambiguïtés en évaluant toutes les hypothèses
- Localisation globale

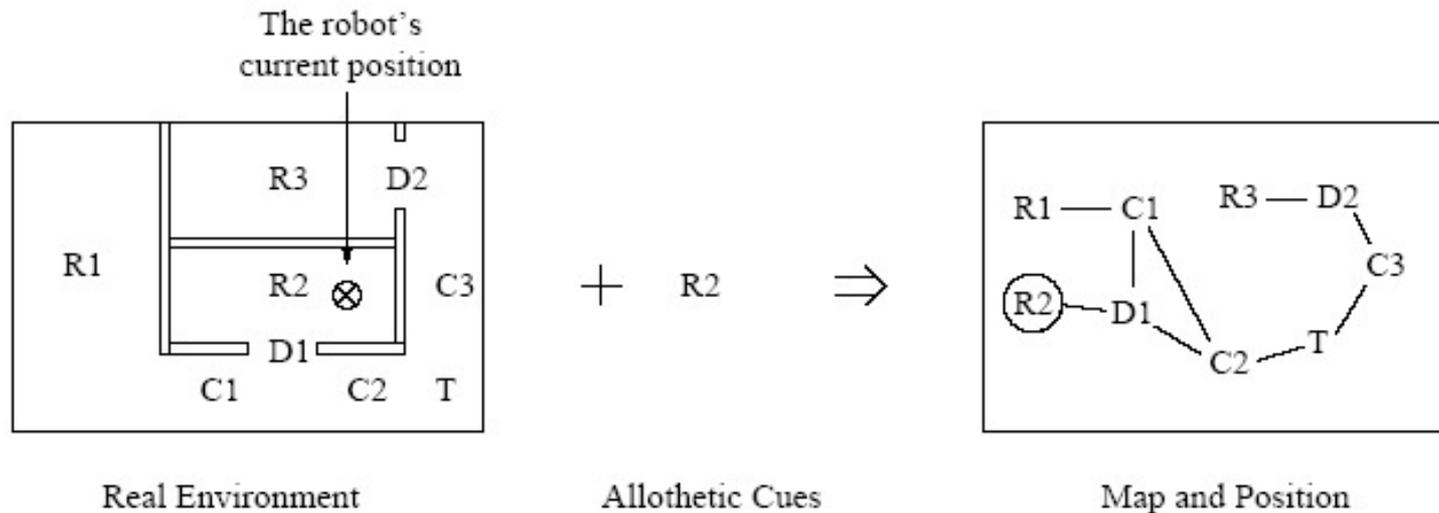
Inférence de position

Pré requis : carte sans « perceptual aliasing »

(tous les nœuds ont une définition différente)

Pour se localiser :

- Comparer les perceptions courantes avec chacun des nœuds
- Le nœud le plus similaire correspond à la position courante



D'un point de vue probabiliste

- Utilise un modèle inverse
- Maximum de vraisemblance → Limité à des environnements simples

Chaque nœud est défini par des attributs extraits d'une image panoramique

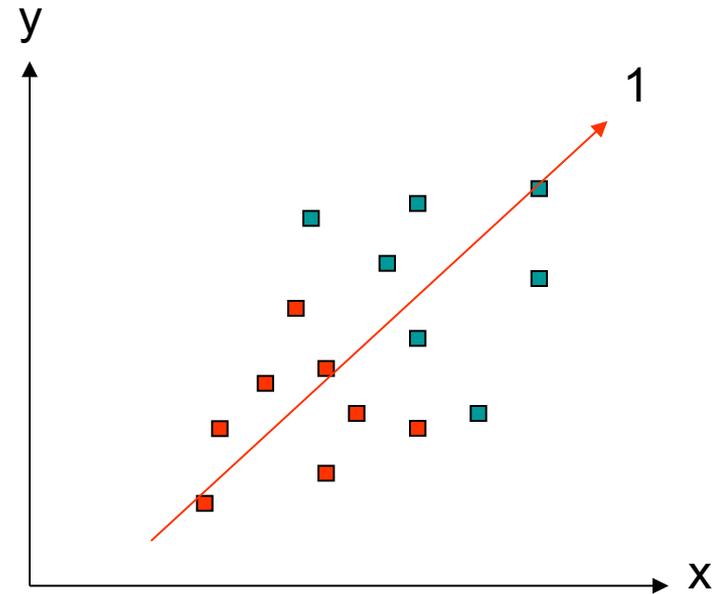
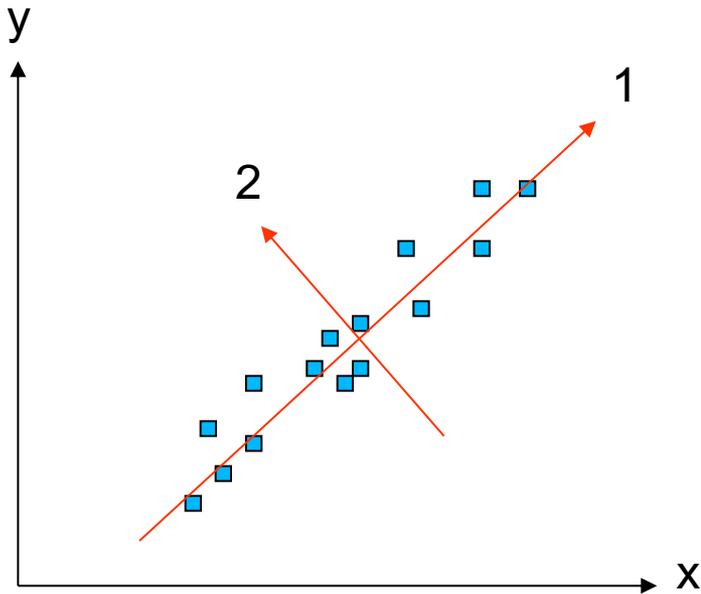
D. Skocaj, H. Bischof, and A. Leonardis. *A robust PCA algorithm for building representations from panoramic images* In Proc. ECCV02. Springer, 2002

Construction de la carte :

- Base de donnée d'images couvrant l'environnement
- Analyse en composantes principales (projection sur une base de faible dimension)
- ACP robuste incluant détection d'« outliers »
- Mémorisation pour chaque nœud des premières composantes pour permettre une bonne discrimination

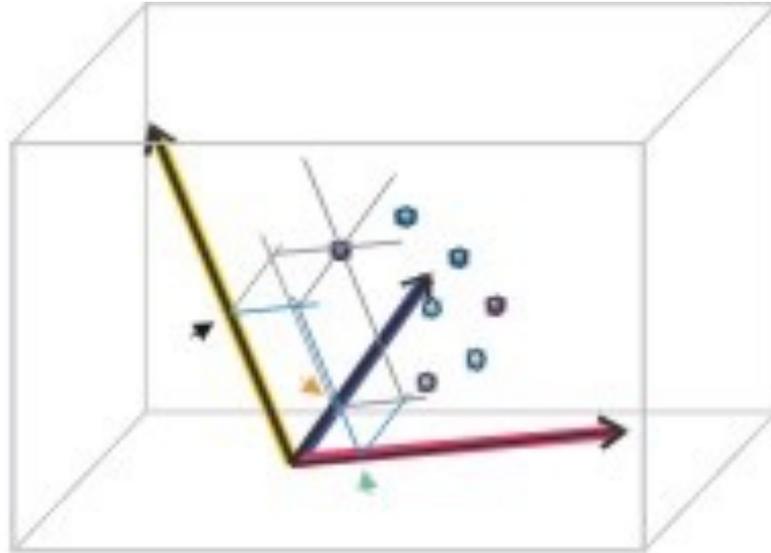
Analyse en composante principales

- Recherche d'une base orthonormée maximisant la variance de la projection



Analyse en composante principale - images

- Points = images = carte
- Vecteurs de base = images

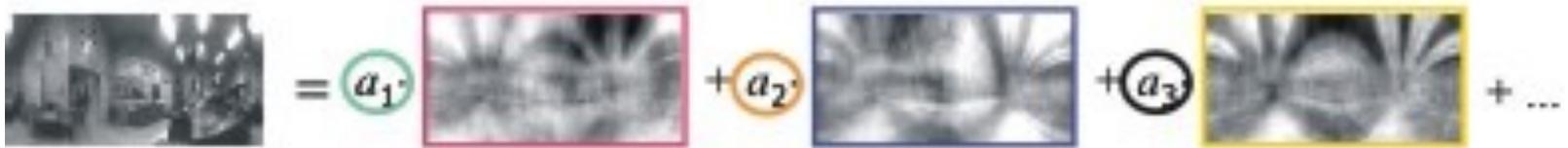


$$\text{Image} = a_1 \cdot \text{PC}_1 + a_2 \cdot \text{PC}_2 + a_3 \cdot \text{PC}_3 + \dots$$

- Mémorisation des images par leurs n premières coordonnées

Localisation

- Prise d'une image
- Décomposition sur la base utilisée



- Recherche de l'image (en donc du nœud) la plus proche en distance euclidienne dans l'espace des composantes principales

$$\text{pos} = \underset{i}{\text{argmin}} d([a_1 \dots a_n], [a_{1,j} \dots a_{n,j}])$$

Caractéristiques

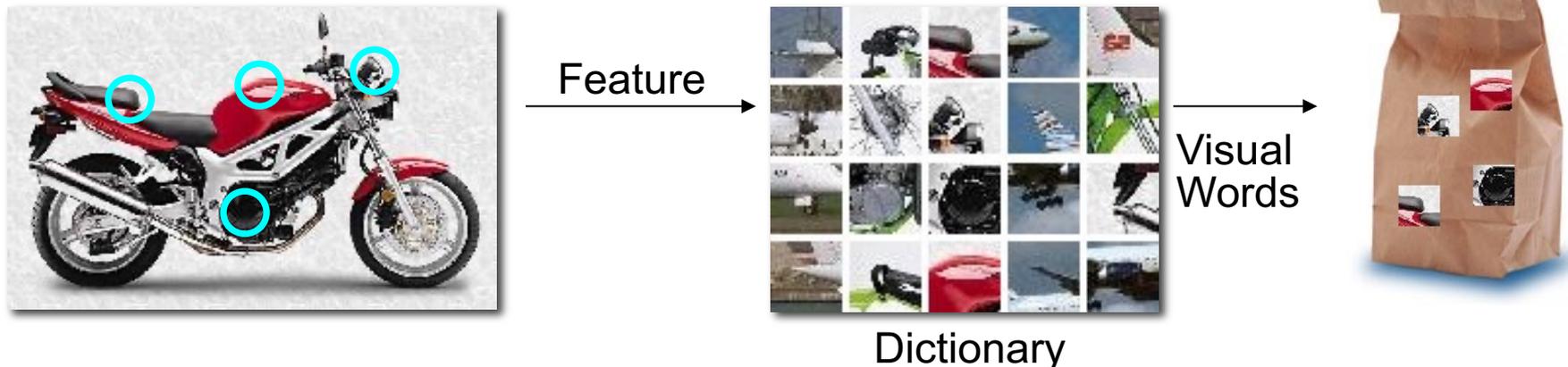
- Matching rapide
- Résistance au bruit
- Sensibilité aux occultations ?

Reconnaissance de la pièce courante

- Utilisation d'une caméra perspective
- Caractérisation par des points d'intérêts visuels locaux
- Perception active pour améliorer la reconnaissance

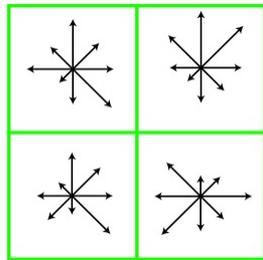
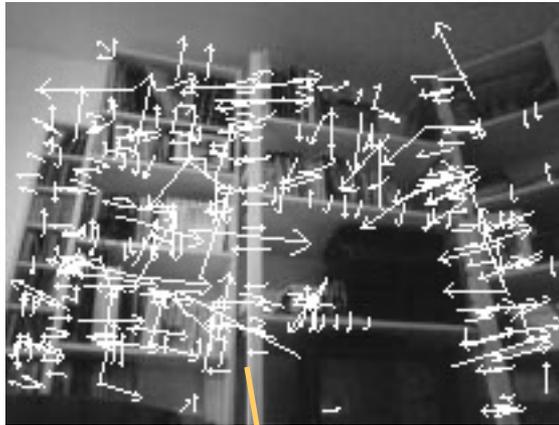
D. Filliat, A visual bag of words method for qualitative localization and mapping. ICRA 2007

Bag of visual words



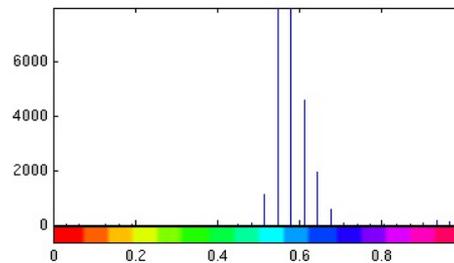
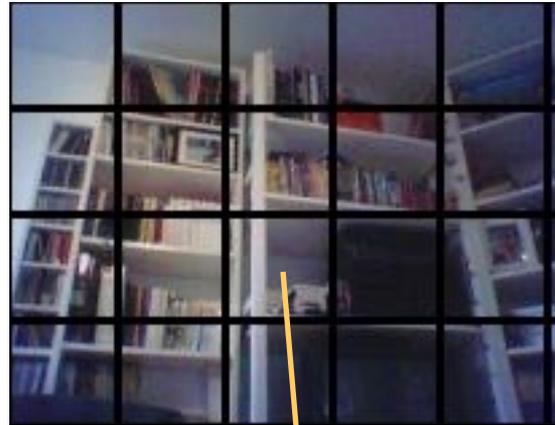
Features

SIFT



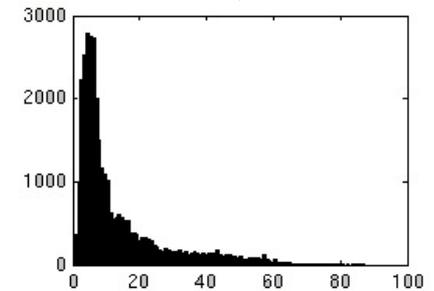
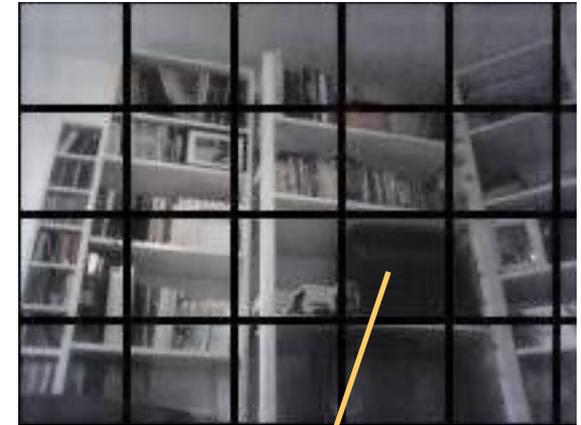
Dim 128

H histograms



Dim 16

V histograms



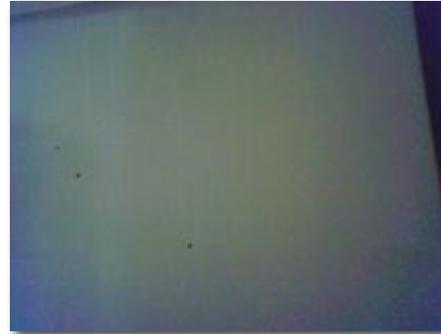
Dim 16

χ^2 distance

$$\|H_1 - H_2\|^2 = \sum_i \frac{(H_{1,i} - H_{2,i})^2}{H_{1,i} + H_{2,i}}$$

Localisation pour la robotique personnelle

- Robustesse aux manipulations du robot et aux mauvaises images



- Localisation qualitative
 - Reconnaître la pièce
 - Base pour méthodes plus précises
 - Comportements liés à la pièce

Structure du problème :

Des images correspondent
à plusieurs pièces



Toutes les images prises depuis une
position correspondent à la même pièce



Localisation
active

*Prendre des images
informatives*

*Prendre de nouvelles
images jusqu'à la certitude*

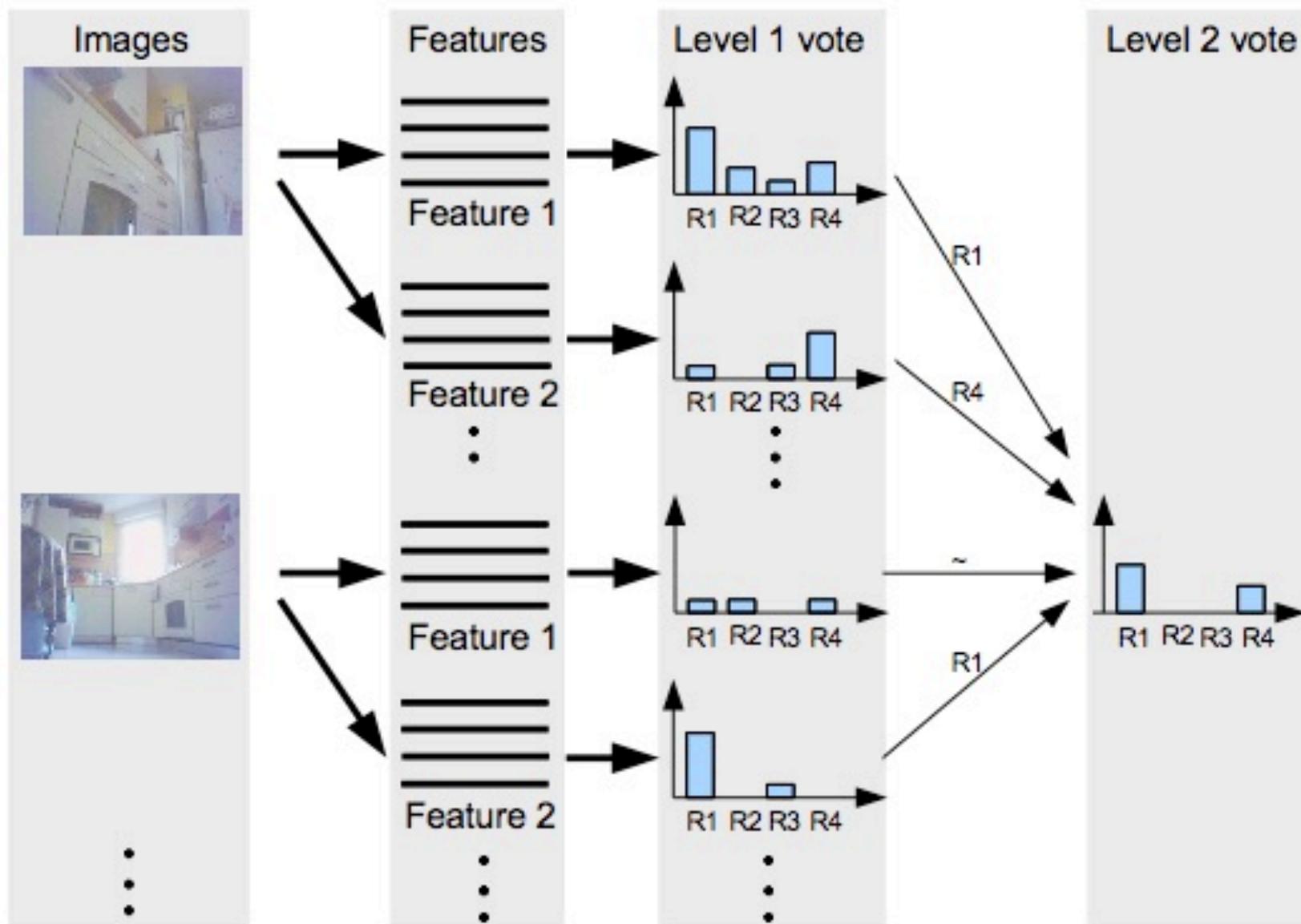
Carte :

- Dictionnaire pour chaque type de feature
- Pour chaque mot : nombre de vue dans chaque pièce

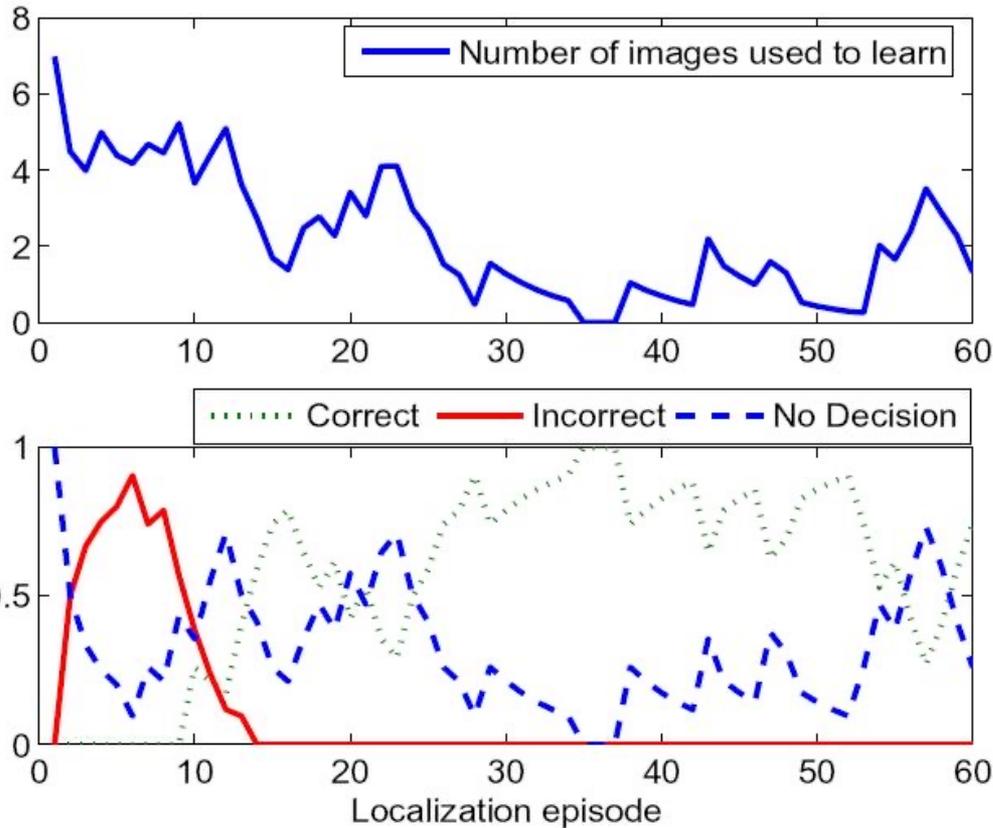
Localisation active :

- Méthode de vote à 2 niveaux
- Premier niveau : sélection des images pertinentes
- Deuxième niveau : besoin de nouvelles informations

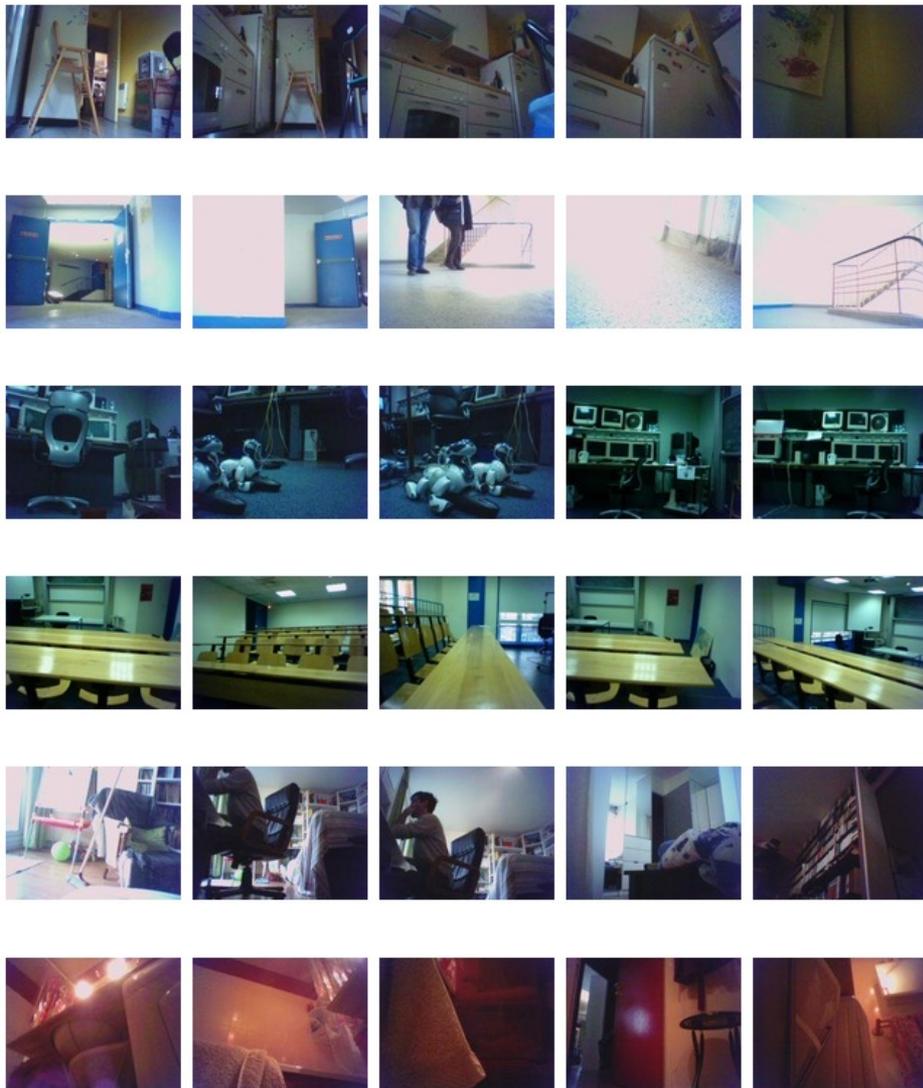
$$quality = \frac{n_{Winner} - n_{Second}}{\sum_i n_i}$$



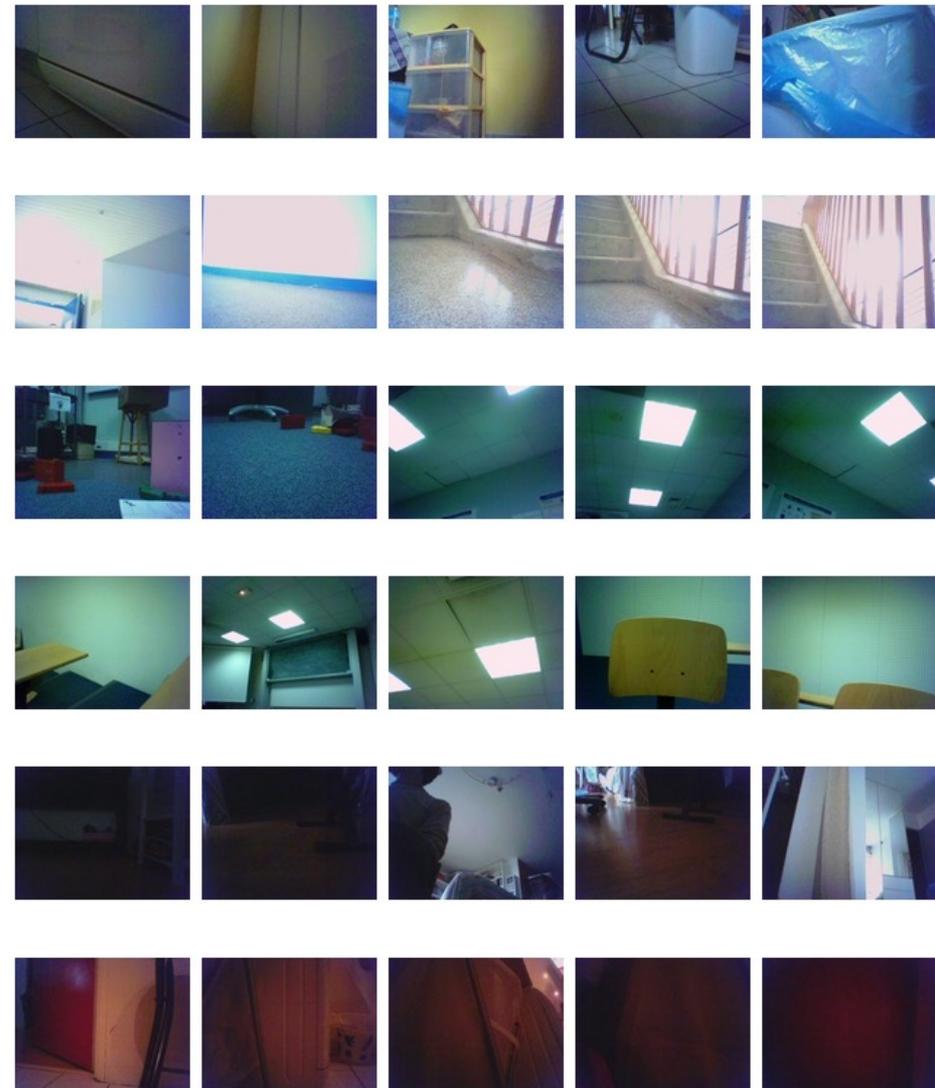
Apprentissage en ligne



SIFT High quality



SIFT Low quality



KTH-INDECS database [Filliat08]



Barbara's office



Corridor



Elin's office



Kitchen



Surroundings of the printer



Cloudy

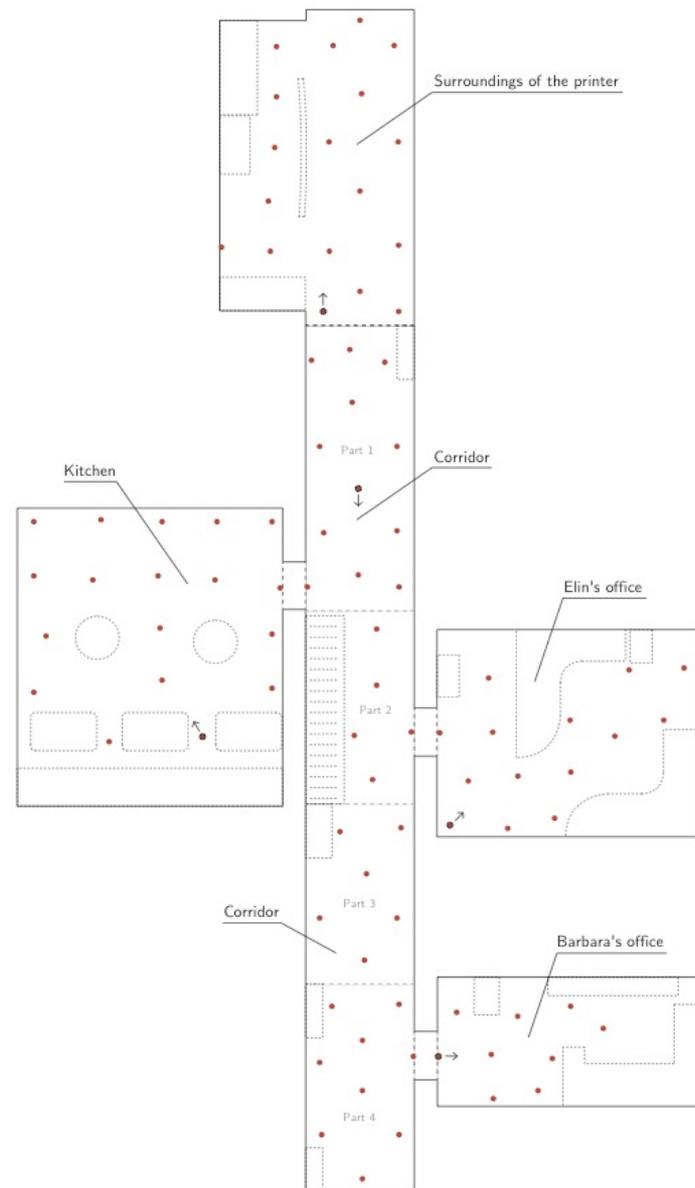


Night

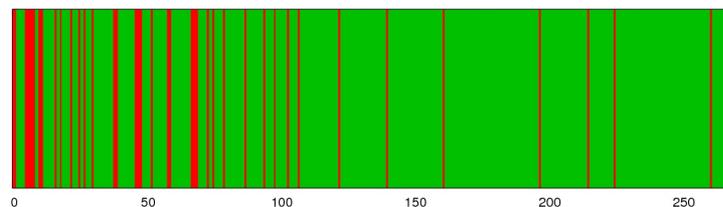
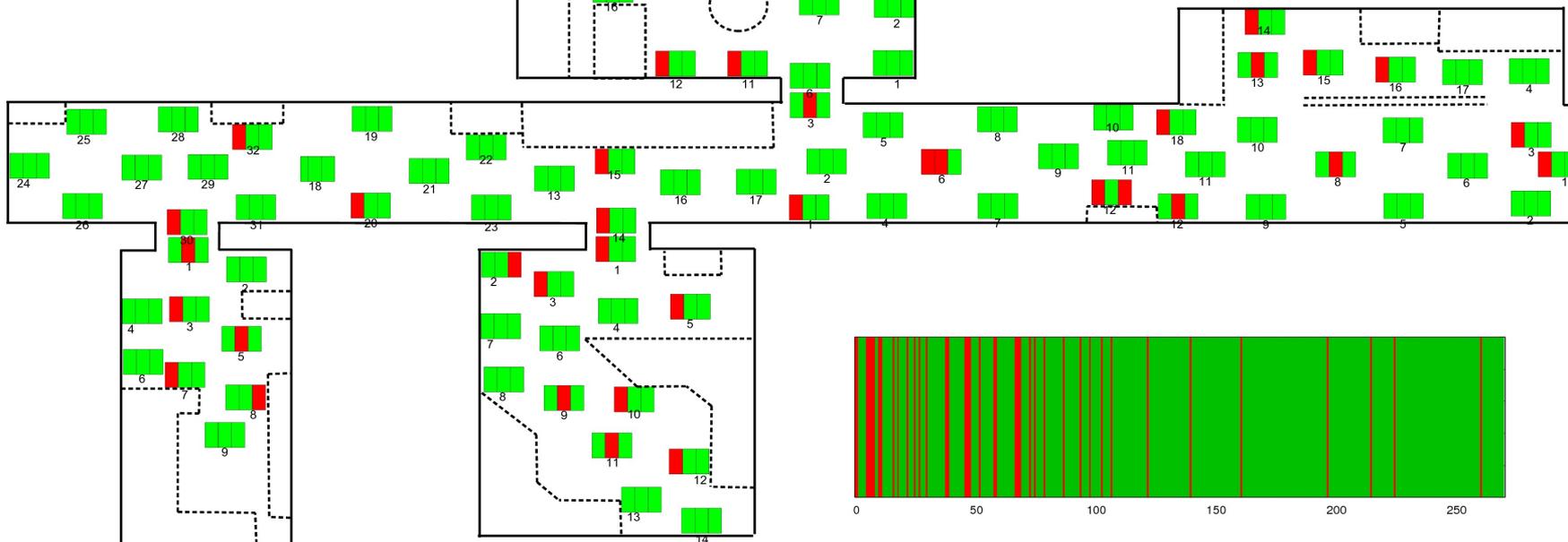
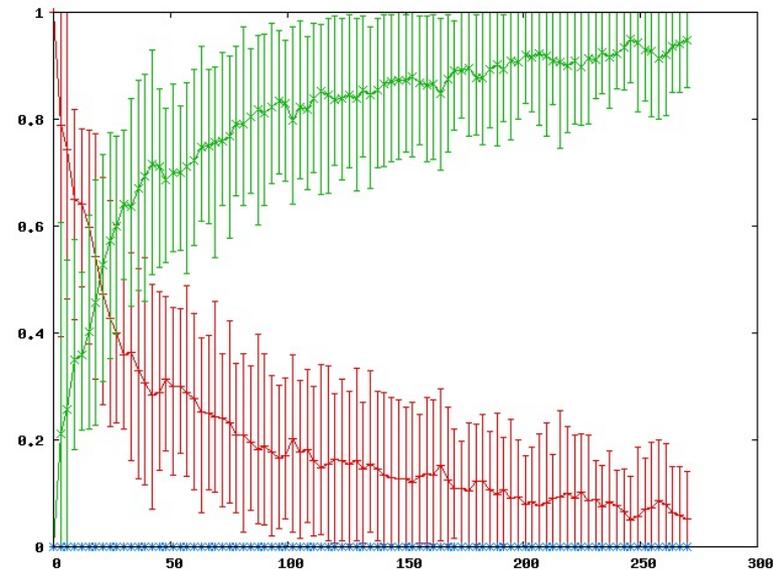
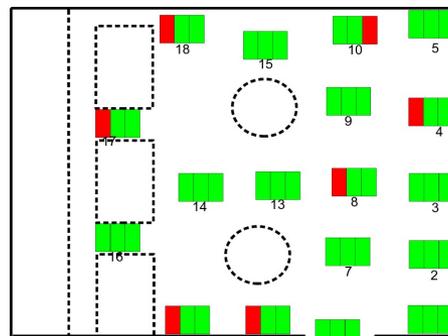


Sunny

Barbara's office



KTH-INDECS database [Filliat08]

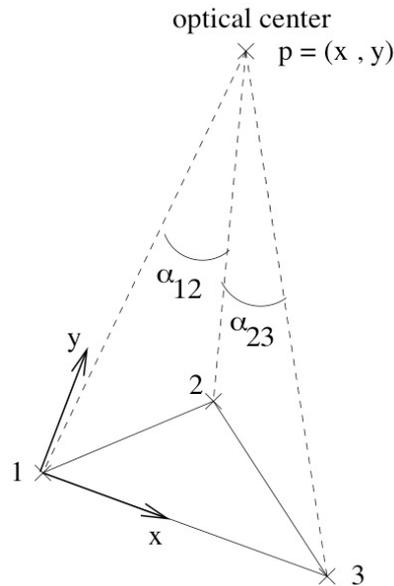


Pré requis : carte sans « perceptual aliasing »

(amers uniques ou environnement complexe sans ambiguïtés)

Pour se localiser :

- Calcul direct de la position à partir des perceptions (utilisant un modèle métrique)
- Par exemple : triangulation



$$\angle_{P12} = -\arctan \left(\frac{l_{23} \sin(\alpha_{12}) (\sin(\alpha_{13}) \cos(\angle_{123}) - \cos(\alpha_{13}) \sin(\angle_{123}))}{l_{12} \sin(\alpha_{23}) + l_{23} \sin(\alpha_{12}) (\cos(\alpha_{13}) \cos(\angle_{123}) + \sin(\alpha_{13}) \sin(\angle_{123}))} \right) \quad (\text{A.6})$$

$$\angle_{23P} = -\angle_{P12} - \alpha_{13} + \angle_{123} \quad (\text{A.7})$$

$$x = l_{13} \frac{\tan(\angle_{23P} + \angle_{132})}{\tan(\angle_{P12} + \angle_{213}) + \tan(\angle_{23P} + \angle_{132})}$$

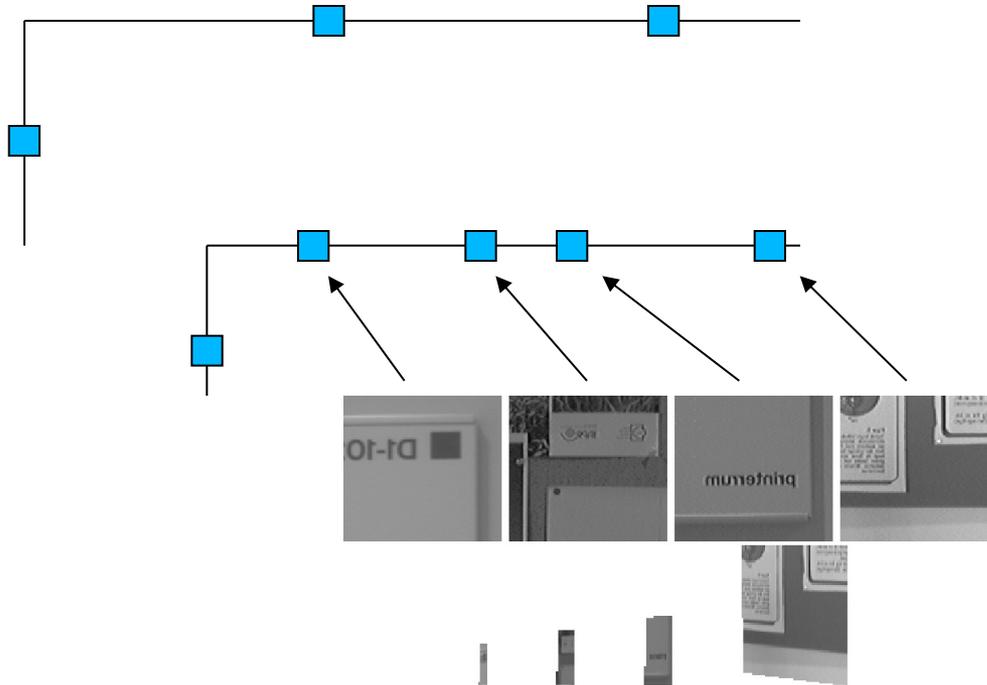
$$y = l_{13} \frac{\tan(\angle_{23P} + \angle_{132}) \tan(\angle_{P12} + \angle_{213})}{\tan(\angle_{P12} + \angle_{213}) + \tan(\angle_{23P} + \angle_{132})}$$

Attention à la configuration des amers

Exemple

Optimal Landmark Selection for Triangulation of Robot Position.

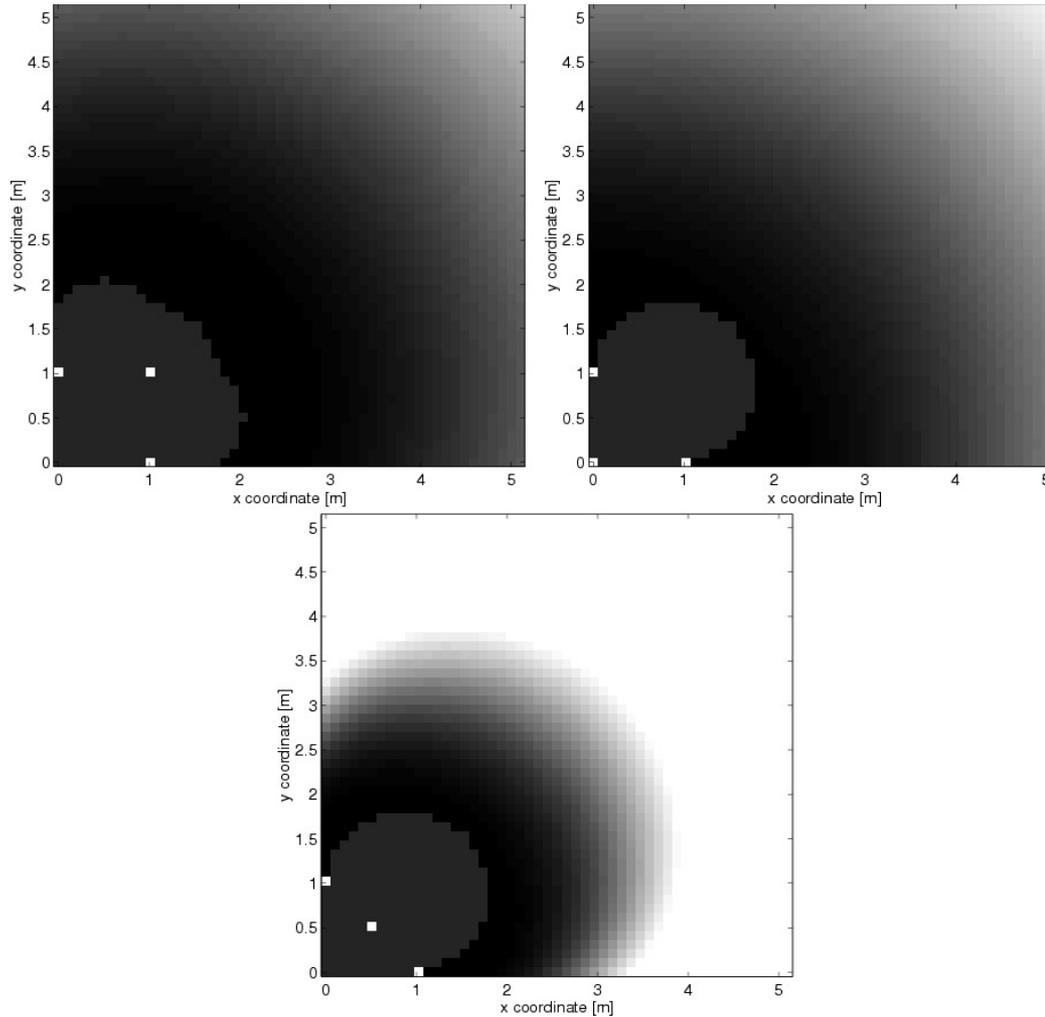
Claus B Madsen and Claus D. Andersen



Carte contenant des amers
ponctuels tous identifiables

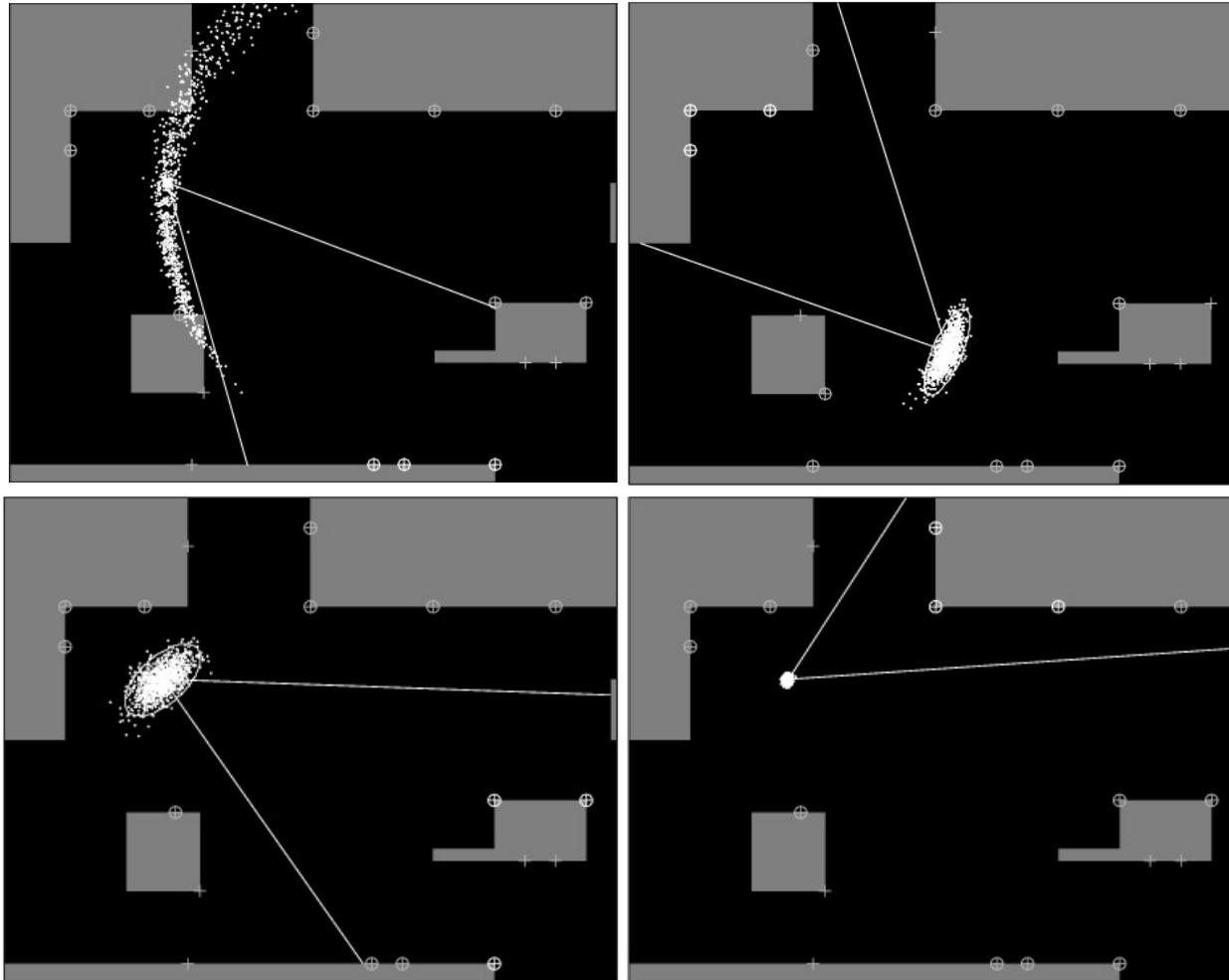


Erreur dépend des configurations d'amers



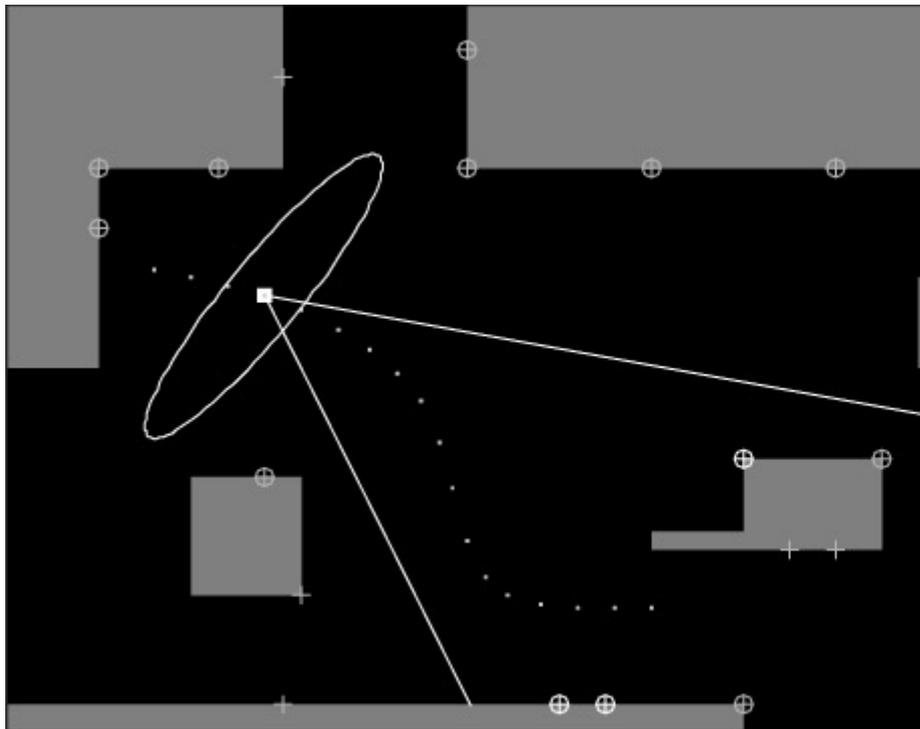
Exemple de localisations

- En fonction des erreurs de perception

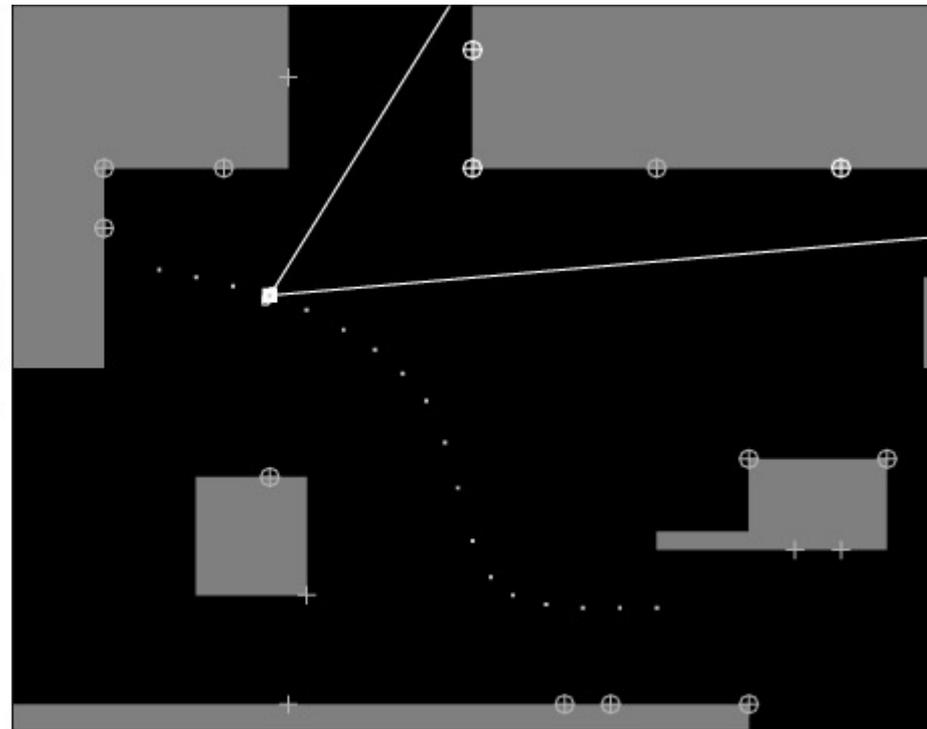


Sélection automatique de la direction

- En fonction d'une estimation de la position



Camera fixe



Direction optimale

Méthodes d'*inférence directe* en pratique

- Mise en œuvre simple (notamment topologique)
 - Utiles pour la localisation globale (lost robot)
 - Problèmes de minima locaux
 - Pas d'utilisation de la proprioception
 - Limité lorsque la taille et la complexité de l'environnement augmentent
- Utilisation d'autres méthodes en complément
- Utilisation d'une estimation initiale de la position
suivi de position/d'hypothèses multiples

Localisation 01 - En résumé

- Il existe trois catégories de méthodes de localisation:
 - Inférence directe de position
 - Suivi de position
 - Suivi d'hypothèses multiples
- Les méthodes d'inférence directes n'utilisent que les perceptions, elles sont souvent assez simples et peuvent être efficaces en utilisant la vision dans des environnements limités, mais trouvent vite leurs limites quand les environnements grandissent.

Robotique Mobile

10 - Localisation 02 – Suivi - ICP

David Filliat

Alexandre Chapoutot

Goran Frehse

prenom.nom@ensta-paris.fr

1. Inférence directe de position

- Utilisation des perceptions seules
- Améliorer le traitement des perceptions
- Localisation globale en environnement sans « *perceptual aliasing* »

2. Suivi de position

- Perceptions (hypothèses de position) + proprioception pour sélectionner une hypothèse
- Résoudre les ambiguïtés par sélection d'une position
- Localisation « locale » à *partir d'une position initiale connue*

3. Suivi d'hypothèses multiples

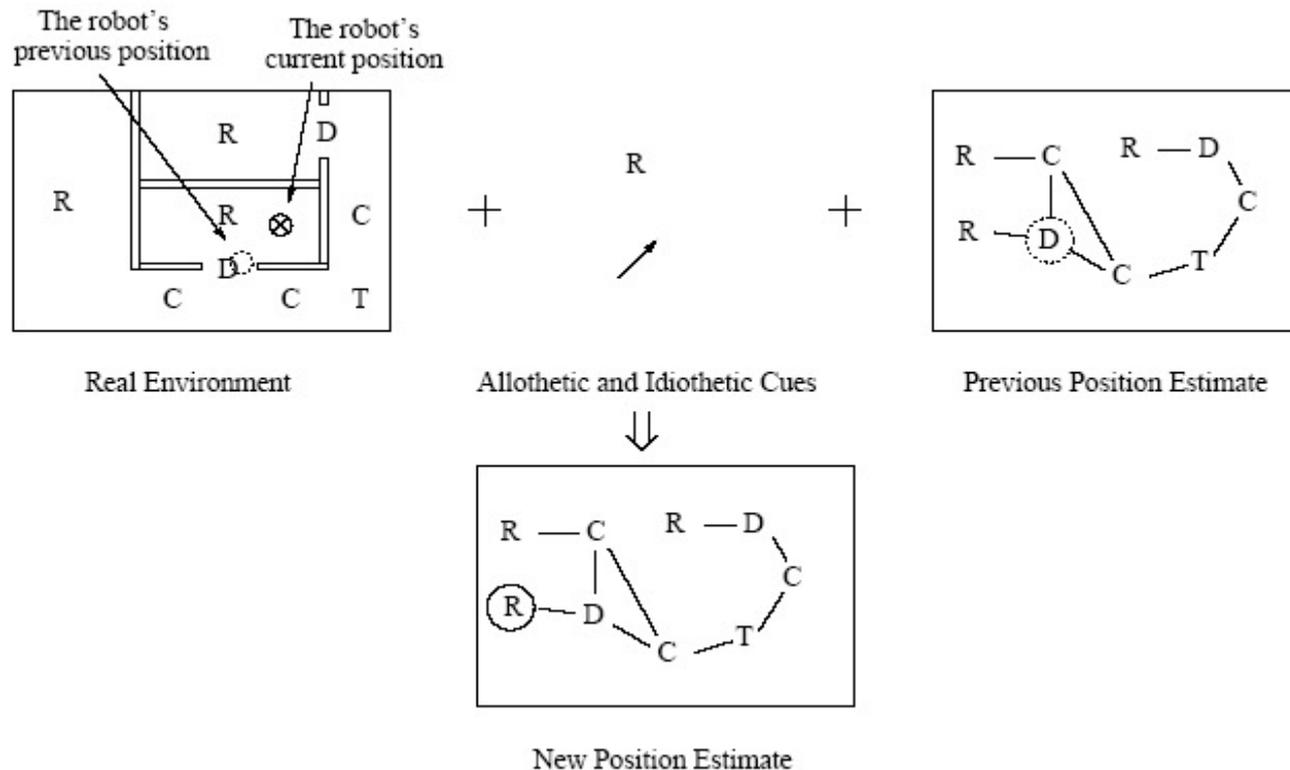
- Perceptions (hypothèses de position) + proprioception pour évaluer les probabilités des différentes hypothèses
- Résoudre les ambiguïtés en évaluant toutes les hypothèses
- Localisation globale

Suivi de position

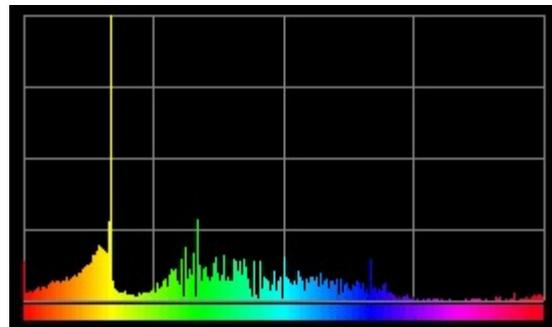
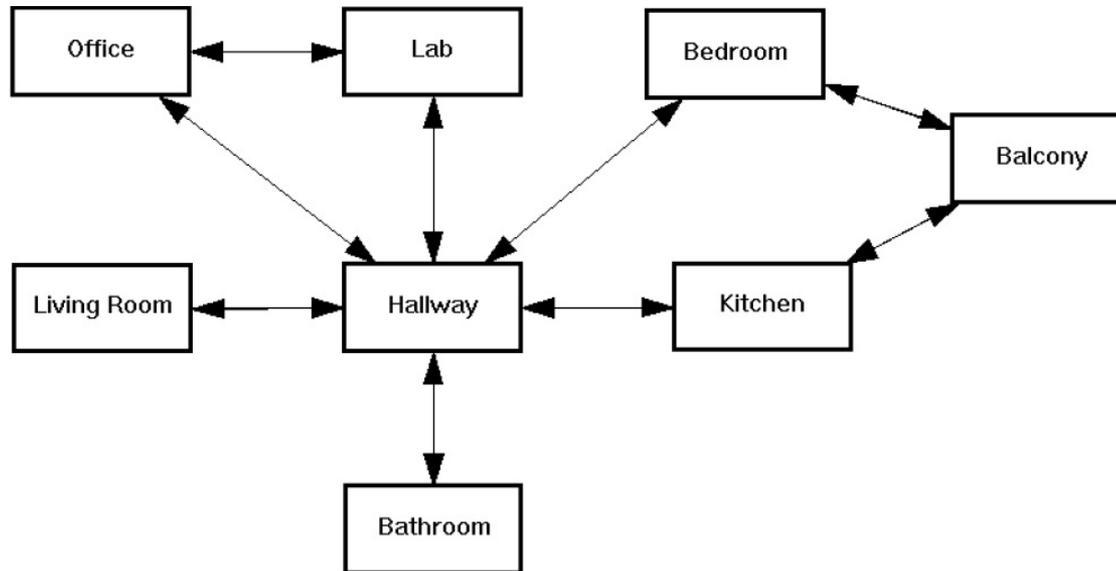
Localisation dans une carte

avec ambiguïtés perceptuelles

- Recherche des nœuds correspondant aux perceptions courantes
- Sélection du nœud cohérent avec la position précédente et les données proprioceptives



Appearance-Based Place Recognition for Topological Localization

Ulrich, I., and Nourbakhsh, I.

Images représentées
Par 6 histogrammes
en H,S,V,R,G & B

Cartographie

Acquerir des images dans tout l'environnement

Calculer les histogrammes de chaque image

Mémoriser les histogrammes et la pièce associée

Localisation

1. Construire les histogrammes de l'image acquise.
2. Pour chaque histogramme, déterminer l'image la plus proche ***pour la pièce courante et les pièces voisines***
3. Voter pour la pièce correspondant aux images retrouvées.
4. Donner la position en cas de vote unanime

Corrélation de cartes

- Carte locale :
 - Construction d'une carte locale avec les perceptions/déplacements sur une petite fenêtre temporelle (sonars/IR)
 - Utilisation d'une seule perception (scan laser)
- Prédiction de la position globale par l'odométrie
- Recherche de la position qui permet la meilleure superposition carte locale/carte globale

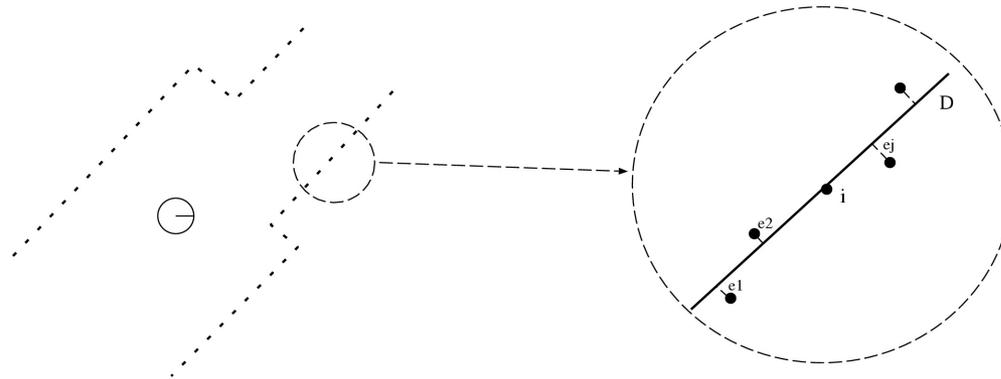
Très nombreuses variantes, par exemple :

LU and MILIOS (1997) *Robot Pose Estimation in Unknown Environments by Matching 2D Range Scans* Journal of Intelligent and Robotic Systems 18: 249–275

Rofer, T. (2002). *Using histogram correlation to create consistent laser scan maps*. IEEE International Conference on Intelligent Robotics Systems.

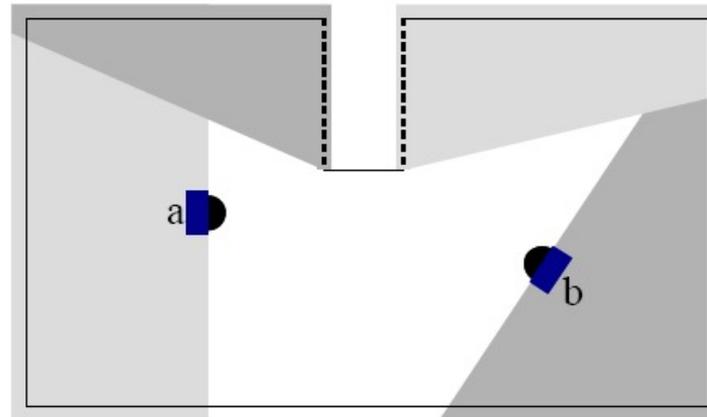
Utilisation des tangentes aux scans (Rofer, 2002)

- recherche par moindres carrés

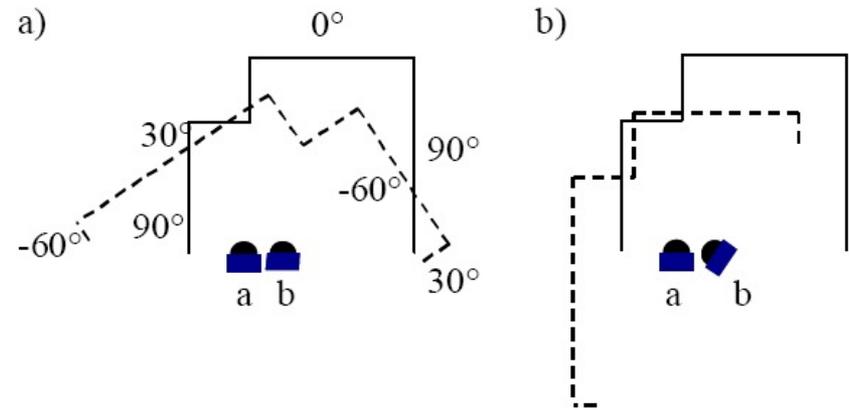
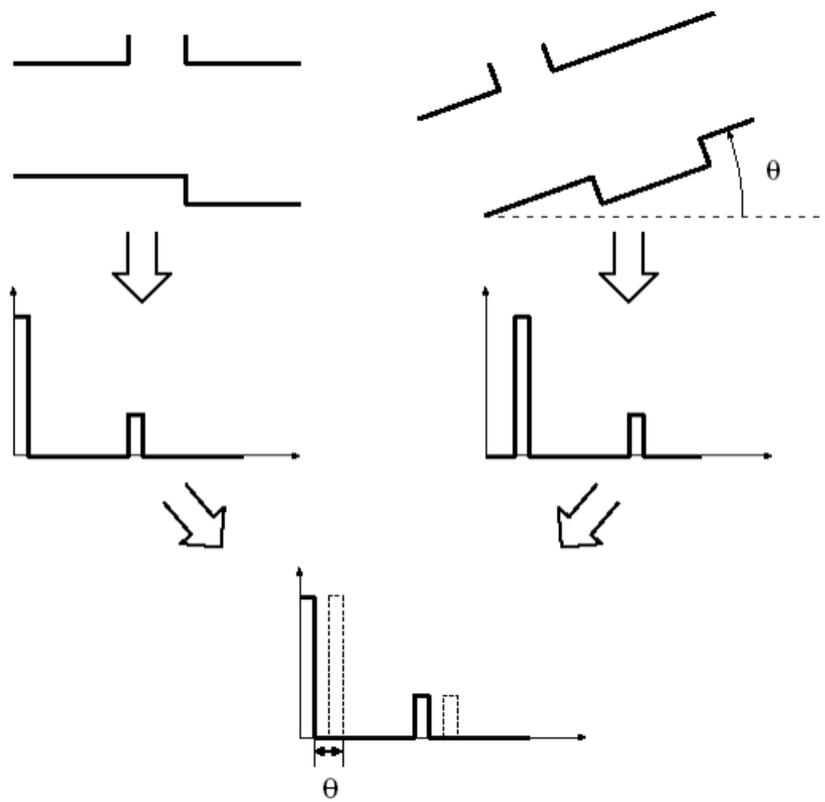


- Suppression des zones mutuellement invisibles

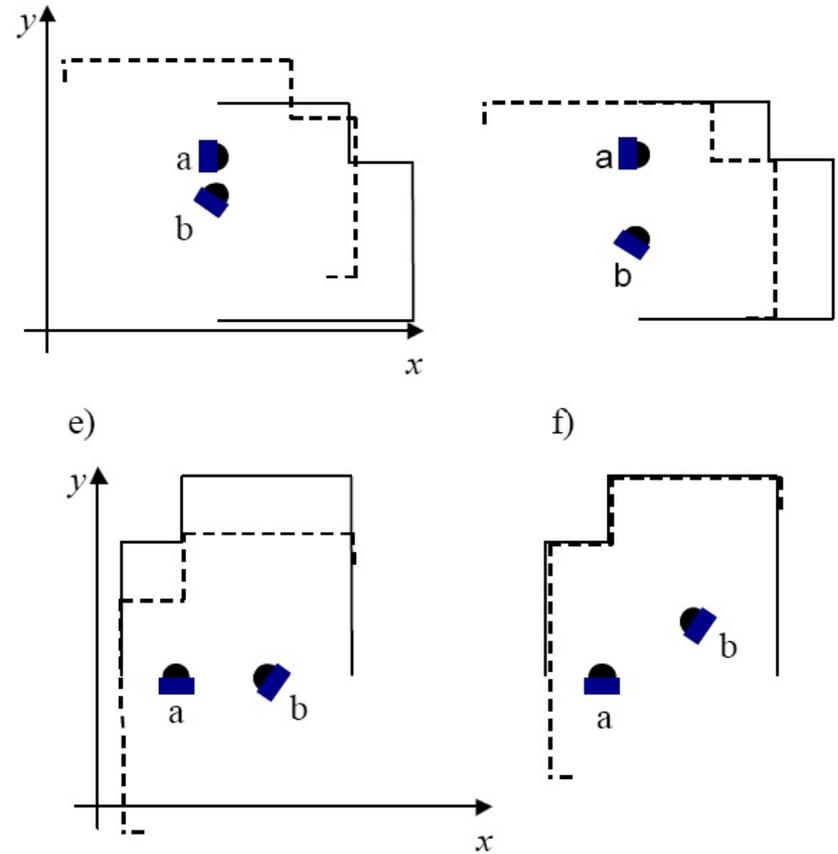
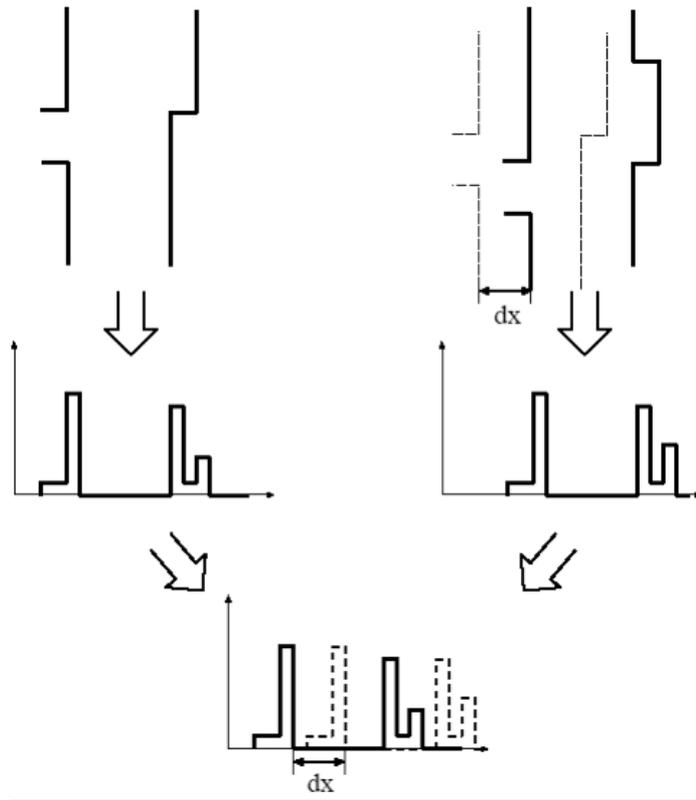
Utilise l'estimation
de la position



Corrélation des histogrammes des directions des tangentes pour corriger l'orientation

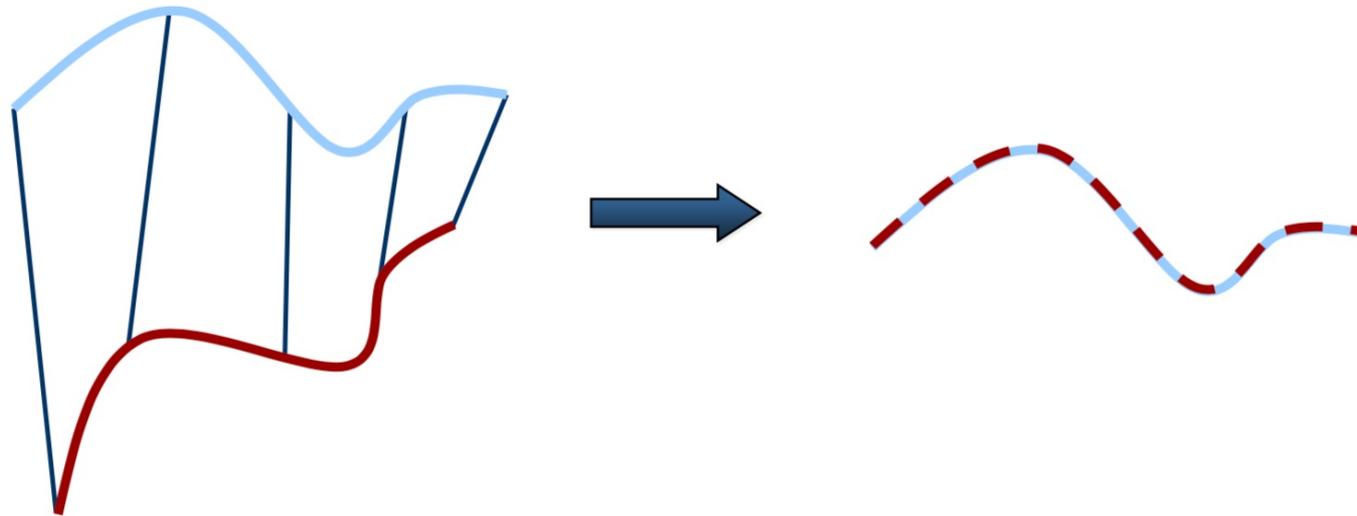


Corrélation des histogrammes du nombre de points pour corriger les translations



Approche Iterated Closest Point

- Calculer la transformation superposant des scans est facile si on connaît les correspondances des points



- Mais elles sont inconnues : estimer avec une correspondance approximative, puis itérer
- Nombreuses variantes : règles de mises en correspondances, filtrage des points, ...

Estimation de la transformation

- Partant de 2 listes de points en correspondance :

$$X = \{x_1, \dots, x_{N_x}\}$$

$$P = \{p_1, \dots, p_{N_p}\}$$

- Calculer les barycentres :

$$\mu_x = \frac{1}{N_x} \sum_{i=1}^{N_x} x_i \quad \text{and} \quad \mu_p = \frac{1}{N_p} \sum_{i=1}^{N_p} p_i$$

- Centrer les points :

$$X' = \{x_i - \mu_x\} = \{x'_i\}$$

$$P' = \{p_i - \mu_p\} = \{p'_i\}$$

Estimation de la transformation

- Calculer la matrice de covariance (2x2 en 2D; 3x3 en 3D)

$$W = \sum_{i=1}^{N_p} x_i' p_i'^T$$

- Calculer la SVD (ici pour des points 3D, idem en 2D)

$$W = U \begin{bmatrix} \sigma_1 & 0 & 0 \\ 0 & \sigma_2 & 0 \\ 0 & 0 & \sigma_3 \end{bmatrix} V^T$$

- La SVD donne la transformation optimale minimisant l'erreur :

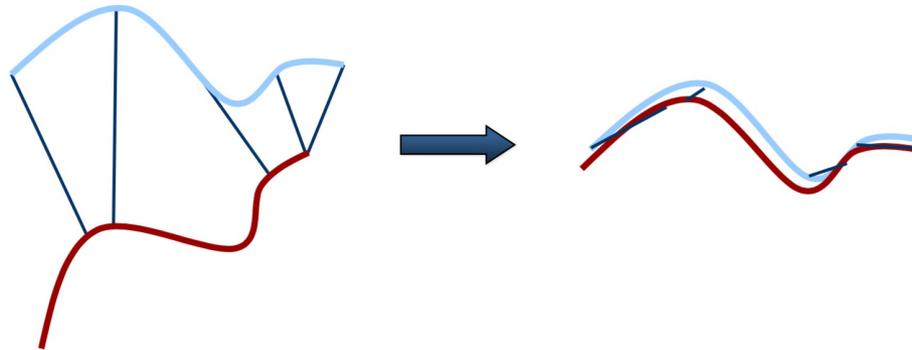
$$R = UV^T$$

$$t = \mu_x - R\mu_p$$

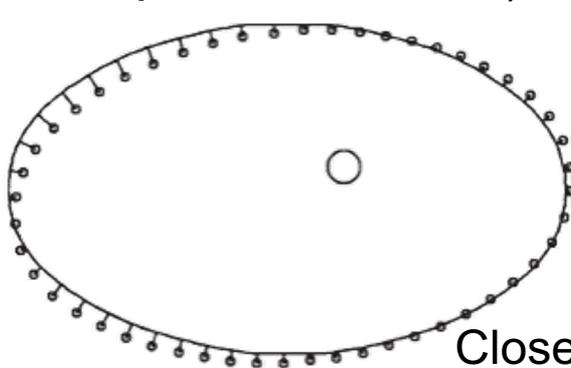
$$E(R, t) = \frac{1}{N_p} \sum_{i=1}^{N_p} \|x_i - Rp_i - t\|^2$$

Trouver les correspondances

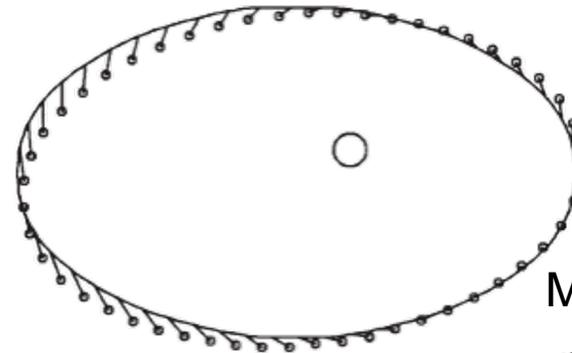
- La transformation précédente est optimale si les correspondances sont parfaites, mais elles sont inconnues
- Approche de base : associer au point le plus proche



- Variante : associer les points ayant la même distance (convergence plus rapide en rotation)



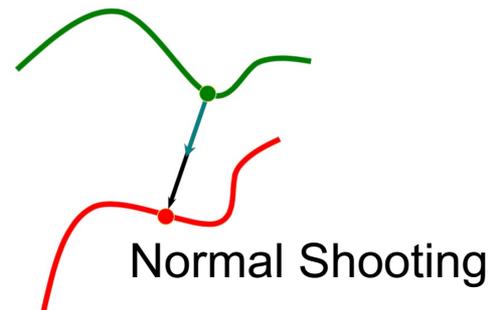
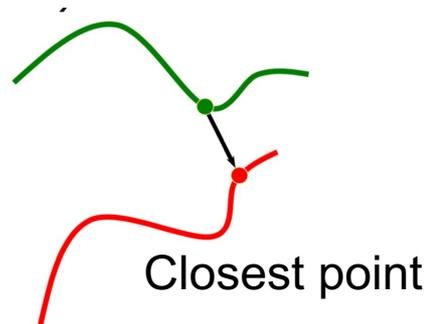
Closest point



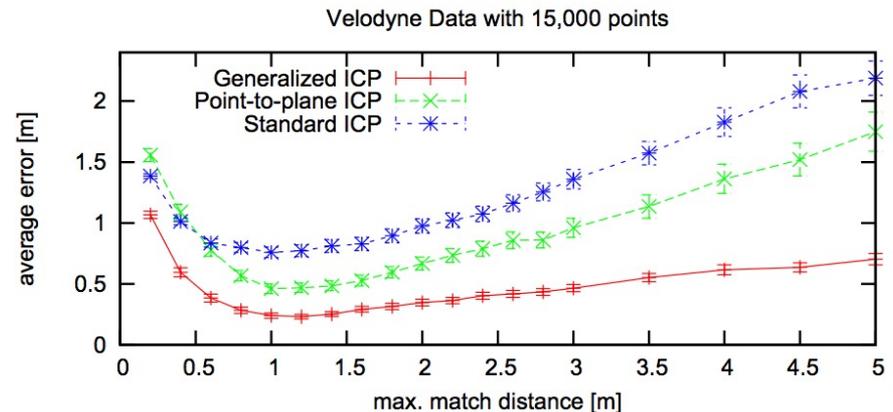
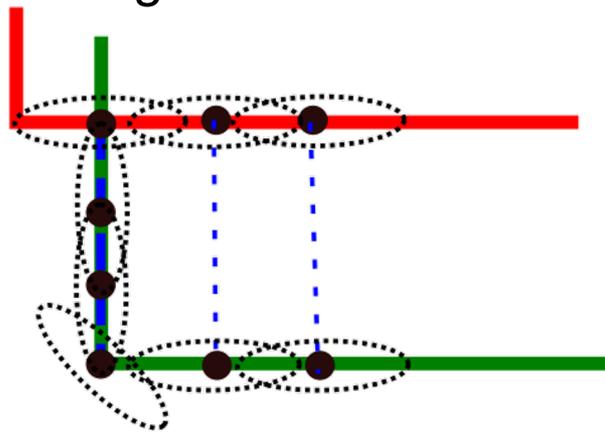
Matching range
point

Trouver les correspondances

- Associer dans la direction de la normale : meilleur pour les surfaces régulières, mais moins bon sur le bruit

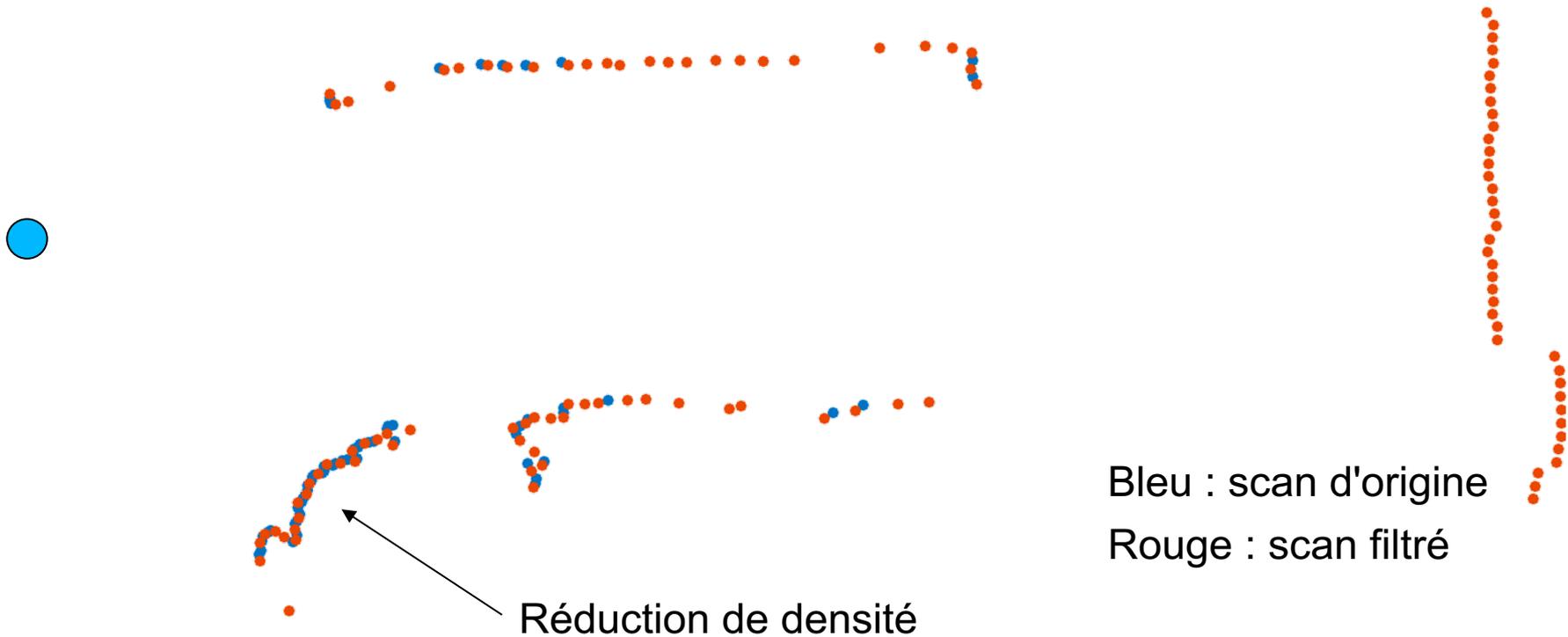


- Associer point à plan : améliore pour les zones planes
- ICP généralisée : association selon les covariances locales



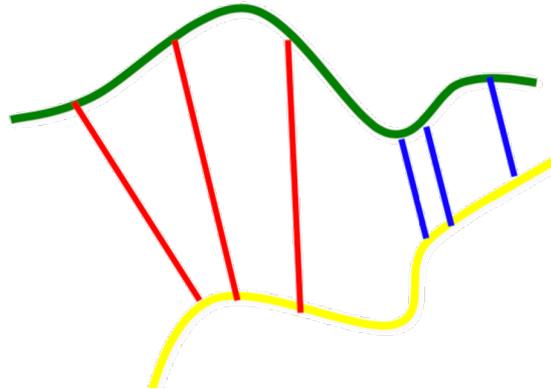
Filtrer les points

- Échantillonnage aléatoire : pour les scans trop denses (vitesse)
- Enlever les points trop loin (moins précis / denses)
- Equilibrer la densité des scans : enlever les points trop proches dans un même scan (lié à la distance du télémètre)

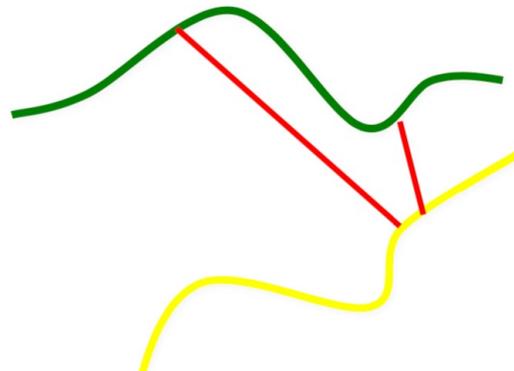


Filtrer les correspondances

- Ne garder que les correspondances les plus proches : par exemple 50% les meilleures



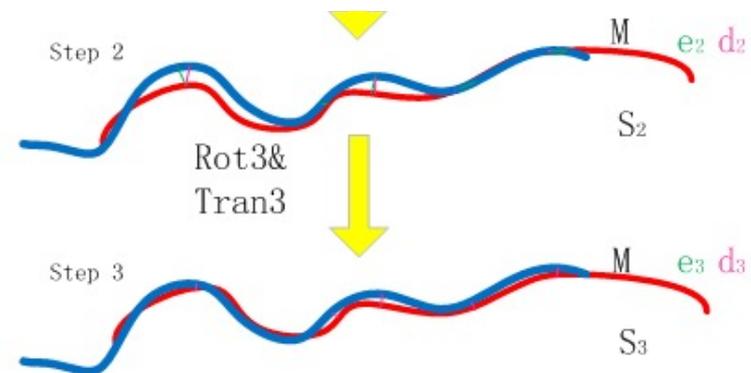
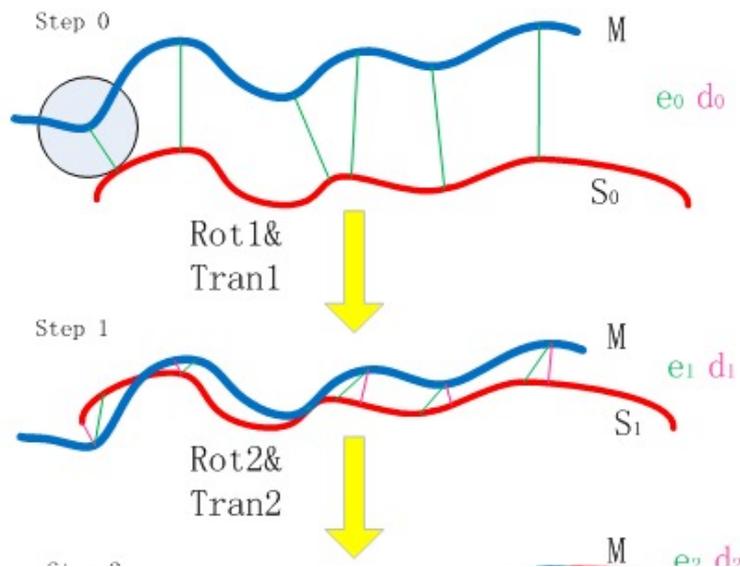
- Rejeter les correspondances dont les paires de points et leurs images ont des distances très différentes



Iterated Closest Point : Algorithmme

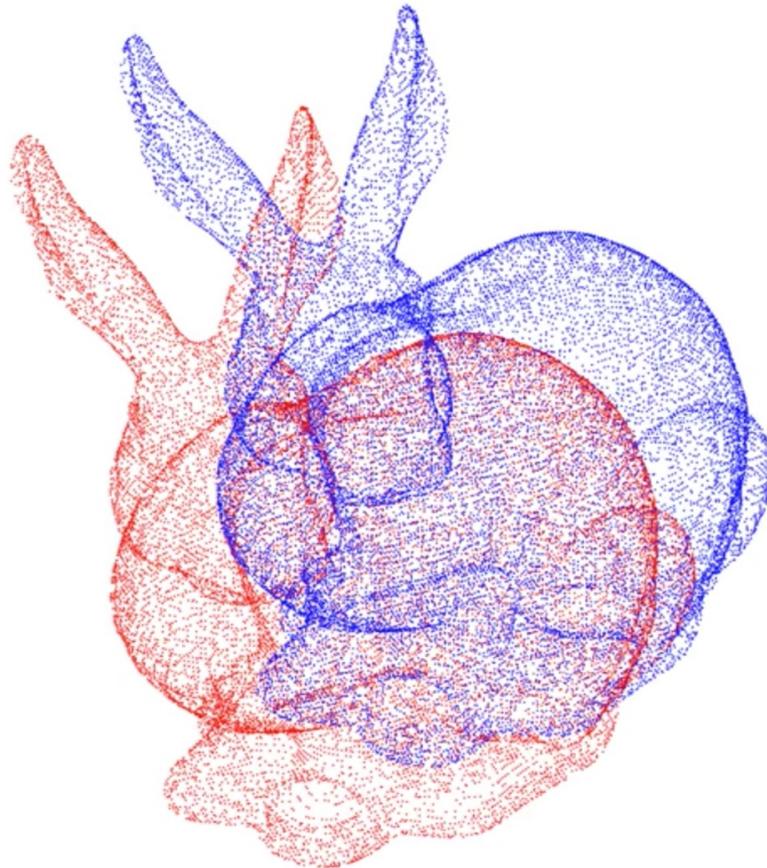
- Initialiser les points p_i d'une position estimée (odométrie par ex.)
- Filtrer les points
- Répéter
 - Associer chaque point du scan à la référence
 - Filtrer les correspondances
 - Calculer R, t en minimisant
 - Appliquer R, t au scan

$$E(R, t) = \frac{1}{N_p} \sum_{i=1}^{N_p} \|x_i - Rp_i - t\|^2$$



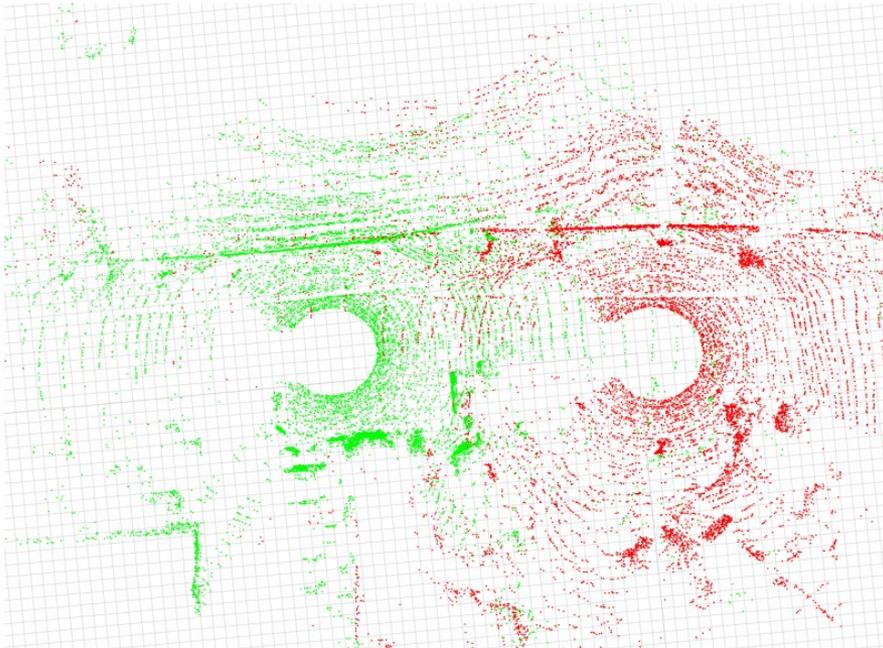
ICP en 3D

Iteration 0

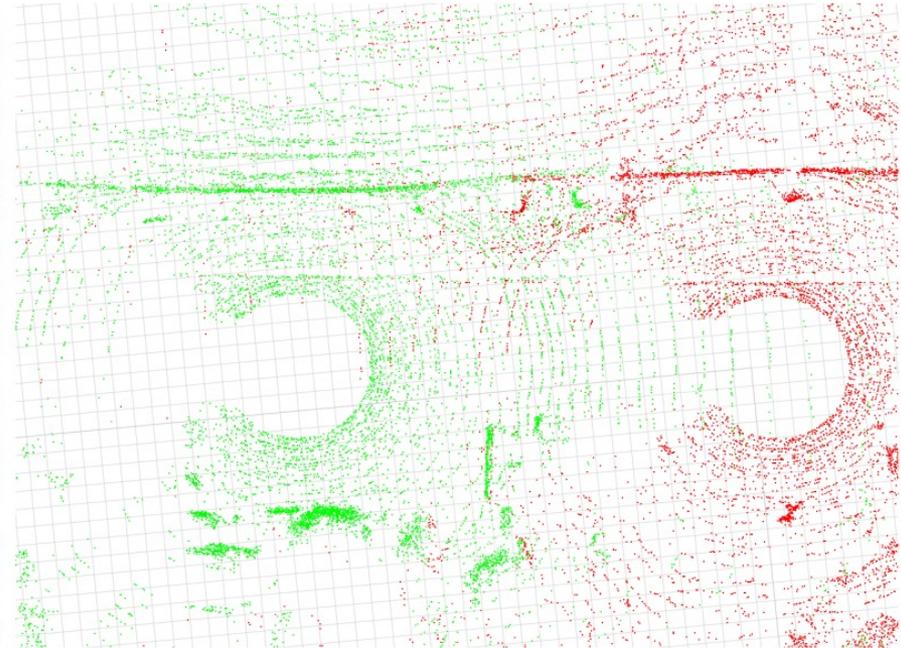


En pratique

- Méthode relativement simple a mettre en œuvre et efficace
- Besoin de capteurs avec fort recouvrement
 - télémètre à balayage mono ou multi nappes
 - Champ de 360 degrés donne de bien meilleures performances



(a) Initial alignment



(c) Generalized-ICP

Localisation 02 - En résumé

- Les méthodes de suivi de position résolvent les ambiguïtés en contraignant la position à partir de la position précédente
- Avec les télémètres laser, la méthode Iterated Closest Point (ICP) et ses nombreuses variantes sont couramment utilisées
- Le principe général est d'associer chaque point d'un scan avec un point du scan de référence selon une heuristique (point le plus proche, distance au plan tangent...), calculer la position qui minimise cette distance, puis ré-itérer