

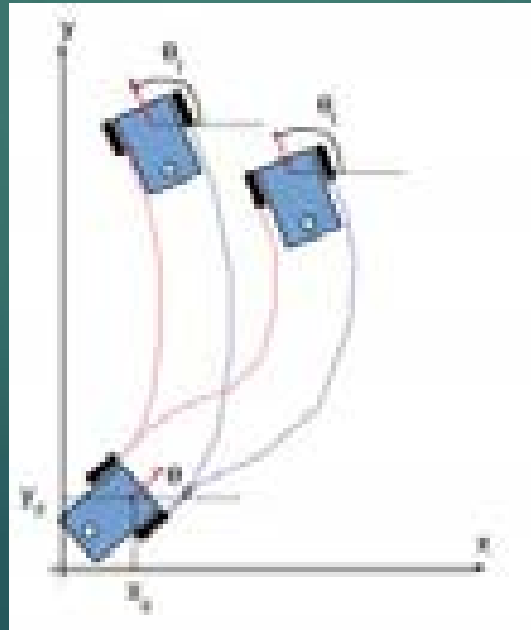
La réalité augmentée

4. ODOMÉTRIE VISUELLE

- 4.1 Recalage temporel
- 4.2 Simultaneous localization and mapping (SLAM)
- 4.3 Méthodes denses

Odométrie visuelle

- ▶ Odométrie : du grec *hodos* (voyage) et *metron* (mesure) = mesure des mouvements (positionnement relatif)
- ▶ Historiquement pour des robots mobiles, grâce à des capteurs embarqués permettant de mesurer le déplacement du robot (de ses roues)



Source : Wikipedia

Odométrie visuelle

- ▶ Odométrie visuelle : utilisation d'un flux d'images acquis en temps réel pour calculer le mouvement
- ▶ Deux types d'approches suivant qu'un modèle 3D de la scène est disponible ou non :
 - ▶ Modèle disponible : recalage temporel (*camera tracking*)
 - ▶ Modèle non disponible : stéréo dense (Kinect, ...) ou *Simultaneous Localization And Mapping* (SLAM)

4. Odométrie visuelle

4.1 RECALAGE TEMPOREL

Exemple : projet des ponts de Paris Inria / Crai

5

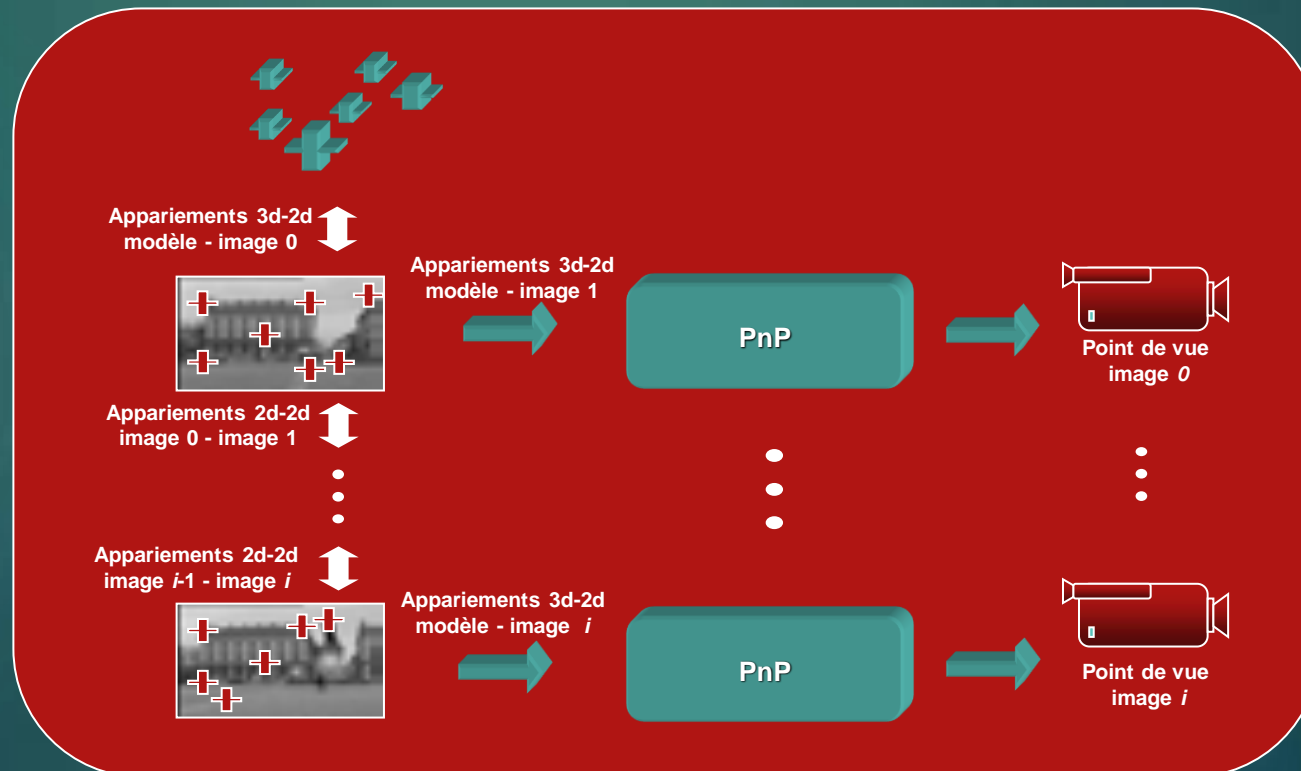


Exemple : projet des ponts de Paris Inria / Crai



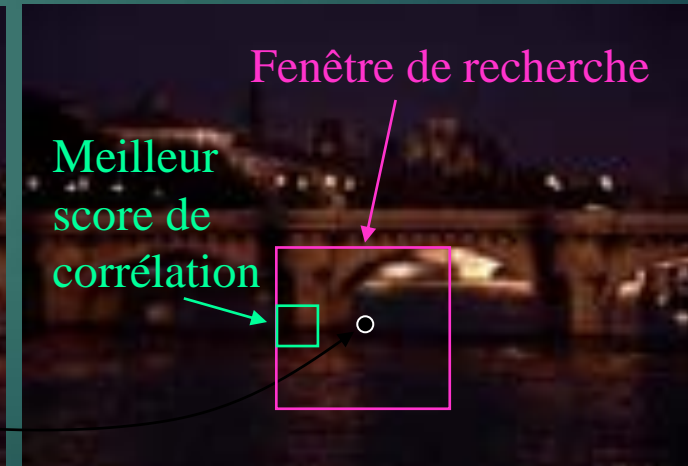
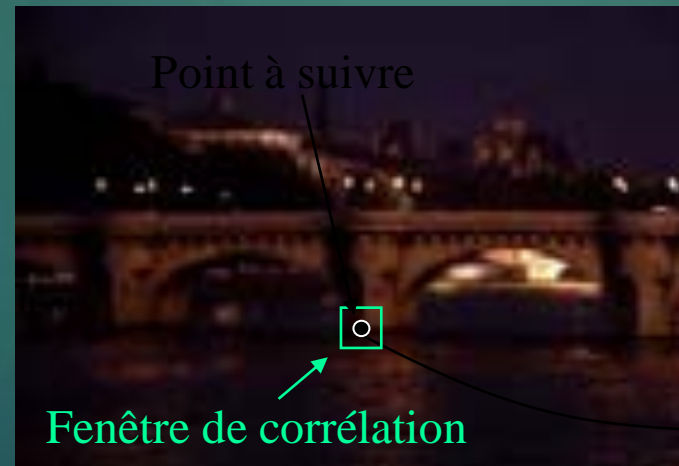
Recalage temporel

- ▶ Le problème revient à apparier des primitives entre images consécutives de la séquence



Méthode 1 : désignation manuelle et suivi supervisé

- ▶ Utilisation d'une fenêtre de corrélation
- ▶ Recherche du motif le plus proche dans une fenêtre de recherche
- ▶ Exemple de score de corrélation : $\sum_{\text{fenêtre}} (I'-I)^2$



Méthode 2 : détection et suivi automatiques



Quels sont les bons points à suivre et comment les suivre ?

Shi and C. Tomasi. Good features to track. CVPR'94

L'algorithme *Good features to track*

10



On cherche des points faciles à retrouver dans le cas d'une translation 2D de l'image
Pour quels points x la translation d peut-elle être détectée de manière fiable ?

$$\text{dissimilarity measure} = \sum_i \sum_j \left(I_2 \left(\mathbf{x} + \mathbf{d} + \begin{pmatrix} i \\ j \end{pmatrix} \right) - I_1 \left(\mathbf{x} + \begin{pmatrix} i \\ j \end{pmatrix} \right) \right)^2$$

L'algorithme *Good features to track*

11

$$\text{dissimilarity measure} = \text{diss}(\mathbf{d}) = \sum_i \sum_j \left(\mathbf{I}_2 \left(\mathbf{d} + \begin{pmatrix} i \\ j \end{pmatrix} \right) - \mathbf{I}_1 \left(\begin{pmatrix} i \\ j \end{pmatrix} \right) \right)^2$$

$$\text{Taylor expansion: } \mathbf{I}_2 \left(\mathbf{d} + \begin{pmatrix} i \\ j \end{pmatrix} \right) \approx \mathbf{I}_2 \left(\begin{pmatrix} i \\ j \end{pmatrix} \right) + g_x(i, j) \mathbf{d}_x + g_y(i, j) \mathbf{d}_y$$

Le Jacobien de $\text{diss}(\mathbf{d})$ vaut 0 quand $\text{diss}(\mathbf{d})$ est minimal

$$\frac{\partial \text{diss}(\mathbf{d})}{\partial \mathbf{d}_x} = 2 \sum_i \sum_j g_x(i, j) \cdot \left(\mathbf{I}_2 \left(\begin{pmatrix} i \\ j \end{pmatrix} \right) + g_x(i, j) \mathbf{d}_x + g_y(i, j) \mathbf{d}_y - \mathbf{I}_1 \left(\begin{pmatrix} i \\ j \end{pmatrix} \right) \right) = 0$$

$$\frac{\partial \text{diss}(\mathbf{d})}{\partial \mathbf{d}_y} = 2 \sum_i \sum_j g_y(i, j) \cdot \left(\mathbf{I}_2 \left(\begin{pmatrix} i \\ j \end{pmatrix} \right) + g_x(i, j) \mathbf{d}_x + g_y(i, j) \mathbf{d}_y - \mathbf{I}_1 \left(\begin{pmatrix} i \\ j \end{pmatrix} \right) \right) = 0$$

L'algorithme *Good features to track*

12

$$\frac{\partial \text{diss}(\mathbf{d})}{\partial \mathbf{d}_i} = 2 \sum_j g_{ij} \cdot (l_1(i,j) + g_{ij} \mathbf{d}_i + g_{ij} \mathbf{d}_j - l_1(i,j)) = 0$$

$$\frac{\partial \text{diss}(\mathbf{d})}{\partial \mathbf{d}_j} = 2 \sum_i g_{ij} \cdot (l_1(i,j) + g_{ij} \mathbf{d}_i + g_{ij} \mathbf{d}_j - l_1(i,j)) = 0$$

Conduit au système linéaire :

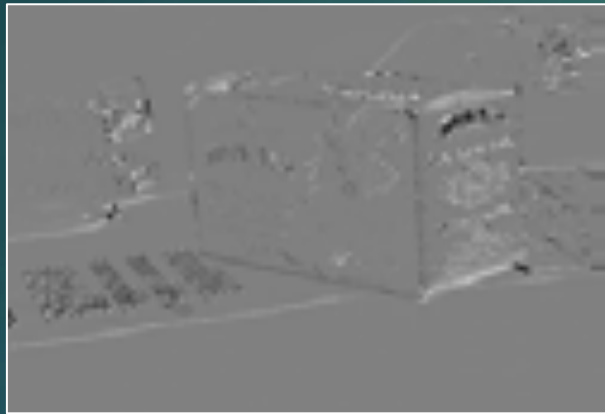
$$\underbrace{\begin{bmatrix} \sum_j (g_{ij}^2) & \sum_j (g_{ij} g_{kj}) \\ \sum_i (g_{ij} g_{kj}) & \sum_i (g_{ij}^2) \end{bmatrix}}_{\mathbf{Z}} \begin{pmatrix} \mathbf{d}_i \\ \mathbf{d}_j \end{pmatrix} = \sum_i \sum_j (l_1(i,j) - l_2(i,j)) \begin{pmatrix} g_{ij} \\ g_{ij} \end{pmatrix}$$

Peut être résolu de manière fiable quand les deux valeurs propres de \mathbf{Z} sont larges (le système est alors bien conditionné)

"Comeness" measure: $\min(\lambda_1, \lambda_2)$

L'algorithm *Good features to track*

13



$$g_x g_y$$



$$(g_x)^2$$



$$(g_y)^2$$



$$\text{Gauss}(\cdot) * g_x g_y$$



$$\text{Gauss}(\cdot) * (g_x)^2$$



$$\text{Gauss}(\cdot) * (g_y)^2$$

L'algorithme *Good features to track*

14

- ▶ Points de Harris : maxima locaux de $\min(\lambda_1, \lambda_2)$



Méthode 3 : détection et appariement automatiques

1. Détecter automatiquement des points d'intérêt (Harris, FAST, ...) dans deux images consécutives ($i-1$ et i) du flux vidéo
2. Pour chaque point de l'image $i-1$, calculer un score de corrélation avec les points de l'image i situés dans la zone de recherche
3. Choisir la paire de points obtenant le score le plus élevé si ce score est supérieur à un certain seuil
4. Variante : répéter la procédure en inversant le rôle des deux images et ne garder que les couples de points qui se choisissent mutuellement

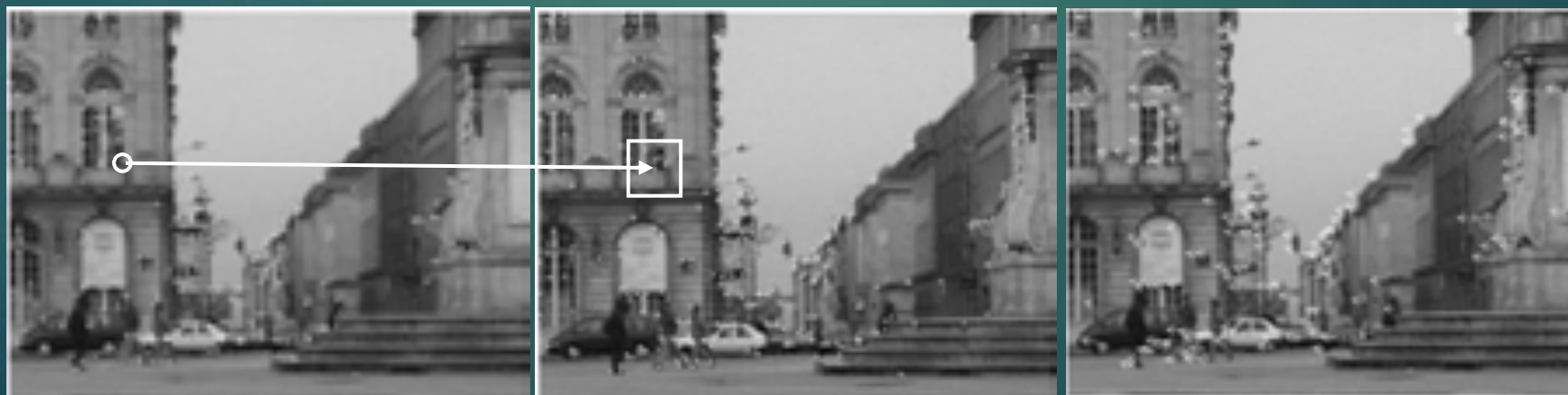


Image de gauche

Image de droite

Appariements

Tracking de plans

- ▶ Les méthodes de suivi de point ne permettent pas de savoir si les points sont correctement appariés
- ▶ La méthode « Markerless tracking using planar structures in the scene » permet de palier ce problème



Homographies

- ▶ Rappel de la projection d'un point 3D :

$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \sim \begin{pmatrix} k_u f & 0 & u_0 \\ 0 & k_v f & v_0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \mathbf{R}_{11} & \mathbf{R}_{12} & \mathbf{R}_{13} & \mathbf{T}_1 \\ \mathbf{R}_{21} & \mathbf{R}_{22} & \mathbf{R}_{23} & \mathbf{T}_2 \\ \mathbf{R}_{31} & \mathbf{R}_{32} & \mathbf{R}_{33} & \mathbf{T}_3 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

- ▶ Si le point 3D est sur le plan $Z=0$:

$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \sim \begin{pmatrix} k_u f & 0 & u_0 \\ 0 & k_v f & v_0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \mathbf{R}_{11} & \mathbf{R}_{12} & \mathbf{R}_{13} & \mathbf{T}_1 \\ \mathbf{R}_{21} & \mathbf{R}_{22} & \mathbf{R}_{23} & \mathbf{T}_2 \\ \mathbf{R}_{31} & \mathbf{R}_{32} & \mathbf{R}_{33} & \mathbf{T}_3 \end{pmatrix} \begin{pmatrix} X \\ Y \\ 0 \\ 1 \end{pmatrix}$$

$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \sim \begin{pmatrix} k_u f & 0 & u_0 \\ 0 & k_v f & v_0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \mathbf{R}_{11} & \mathbf{R}_{12} & \mathbf{T}_1 \\ \mathbf{R}_{21} & \mathbf{R}_{22} & \mathbf{T}_2 \\ \mathbf{R}_{31} & \mathbf{R}_{32} & \mathbf{T}_3 \end{pmatrix} \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} = \mathbf{H} \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix}$$

$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \sim \mathbf{H} \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix}$$

Où \mathbf{H} est une matrice 3x3 inversible appelée homographie

Homographies

18

- ▶ Projections du point 3D dans les images i et $i+1$

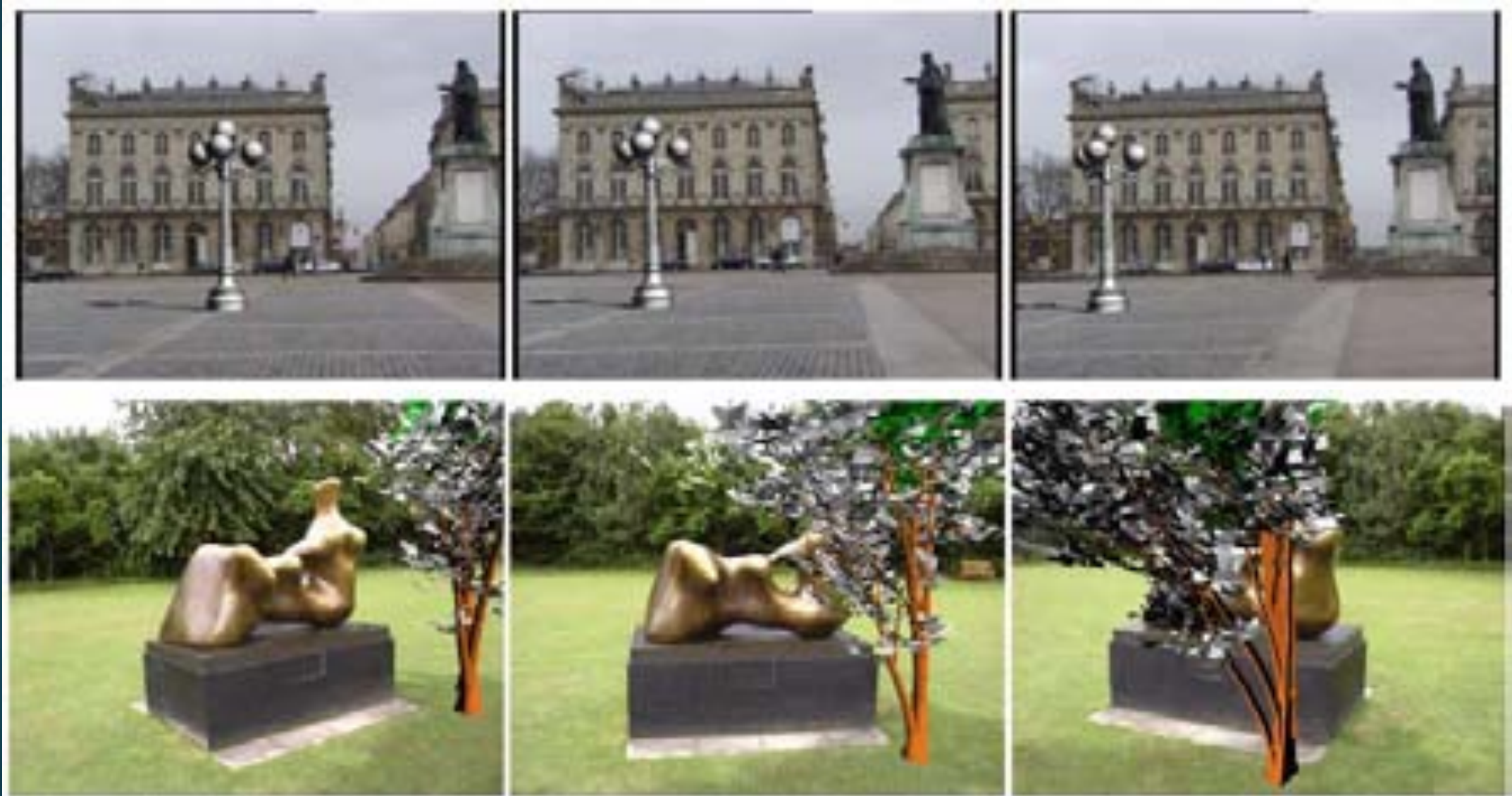
$$\left. \begin{array}{l} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \sim \mathbf{H}_i \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} \\ \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} \sim \mathbf{H}_{i+1} \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} \end{array} \right\} \Rightarrow \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} \sim \mathbf{H}_{i+1} \mathbf{H}_i^{-1} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \Rightarrow \boxed{\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} \sim \mathbf{H} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}}$$

La matrice 3×3 \mathbf{H} est l'homographie qui associe **n'importe quel** point du plan $Z=0$ de l'image i en son homologue dans l'image $i+1$

- ▶ Approche RANSAC (RANdom Sample Consensus) pour identifier les appariements corrects (inliers)
 - ▶ Appariement par corrélation de n couples de points (m_j, m'_j)
 - ▶ Tirage aléatoire de 4 paires de points
 - ▶ Calcul de \mathbf{H}
 - ▶ Pour chacun des $n-4$ points non tirés, calcul de la distance euclidienne entre $\mathbf{H}m_j$ et m'_j et comptage du nombre de couples pour lesquels cette distance est inférieure à un certain seuil (inliers)
 - ▶ Cette procédure est répétée m fois et on garde l'homographie \mathbf{H} ayant donné lieu au plus grand nombre d'inliers (consensus)

Exemples de résultats

19



Généralisation au multiplans

20



[Simon & Berger, IEEE CGA 2002]

Généralisation au multiplans

21



[Simon & Berger, IEEE CGA 2002]

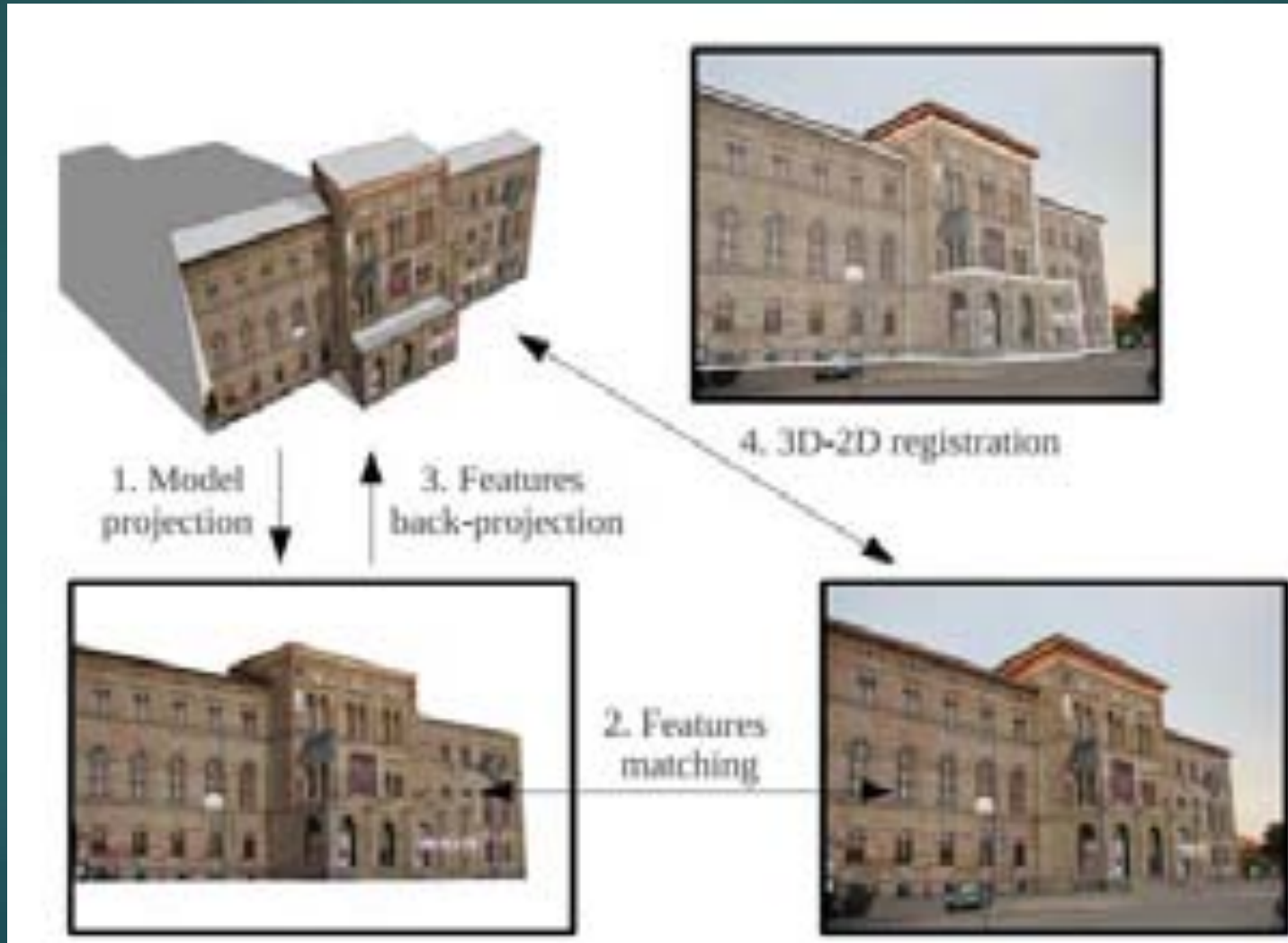
Suivi de contours (couplage avec un système outside-in)

22

Localisation et modélisation en réalité augmentée
11/01/2024

Klein Drummond ISMAR'04
Tracking Video

Tracking par synthèse



Tracking par synthèse utilisant la corrélation 2D (coins de Harris)

24



[Lepetit et al.],
EPFL Lausanne

Couplage avec les points de fuite

25



Localisation et modélisation en réalité augmentée
11/01/2024

Couplage avec les points de fuite

26



Localisation et modélisation en réalité augmentée
11/01/2024

Tracking par synthèse utilisant la corrélation 1D (points de contours)

27

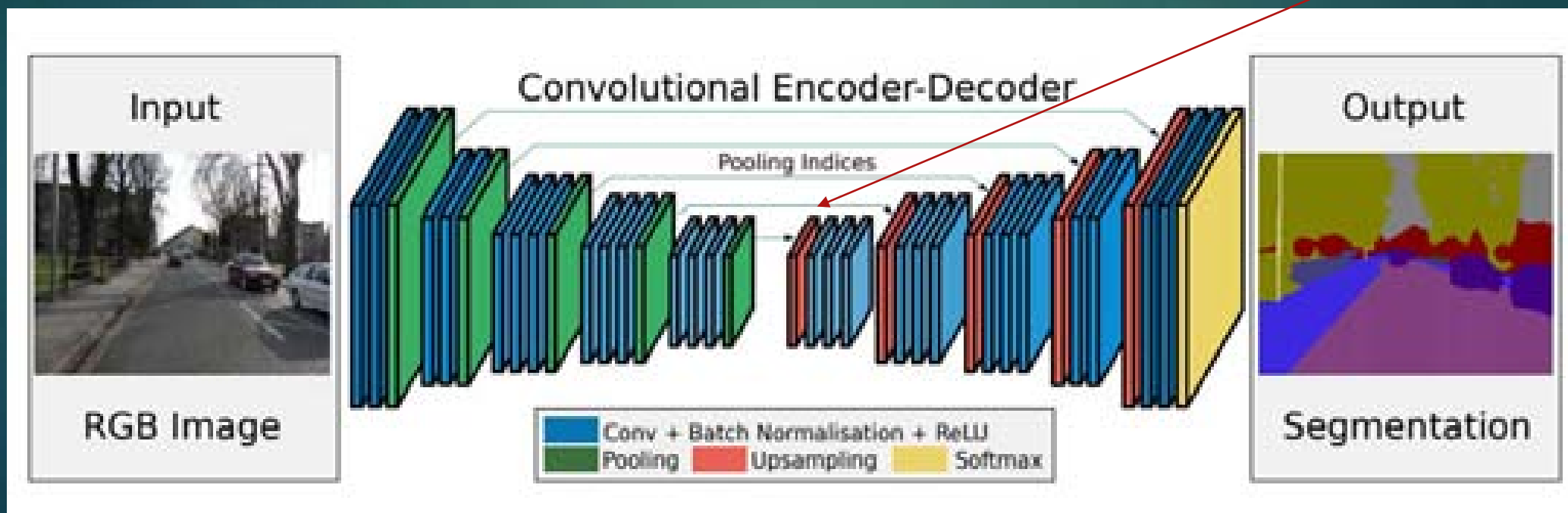
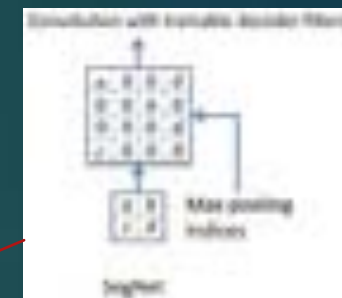


[Reitmayr & Drummond, ISMAR 2006] : Overview of the tracking process. A view is rendered from the prior camera pose (a), then edgels are extracted from the grayscale image (b). The prior pose overlaid over the video image (c) is updated with measurements (d) to yield the posterior pose (e).

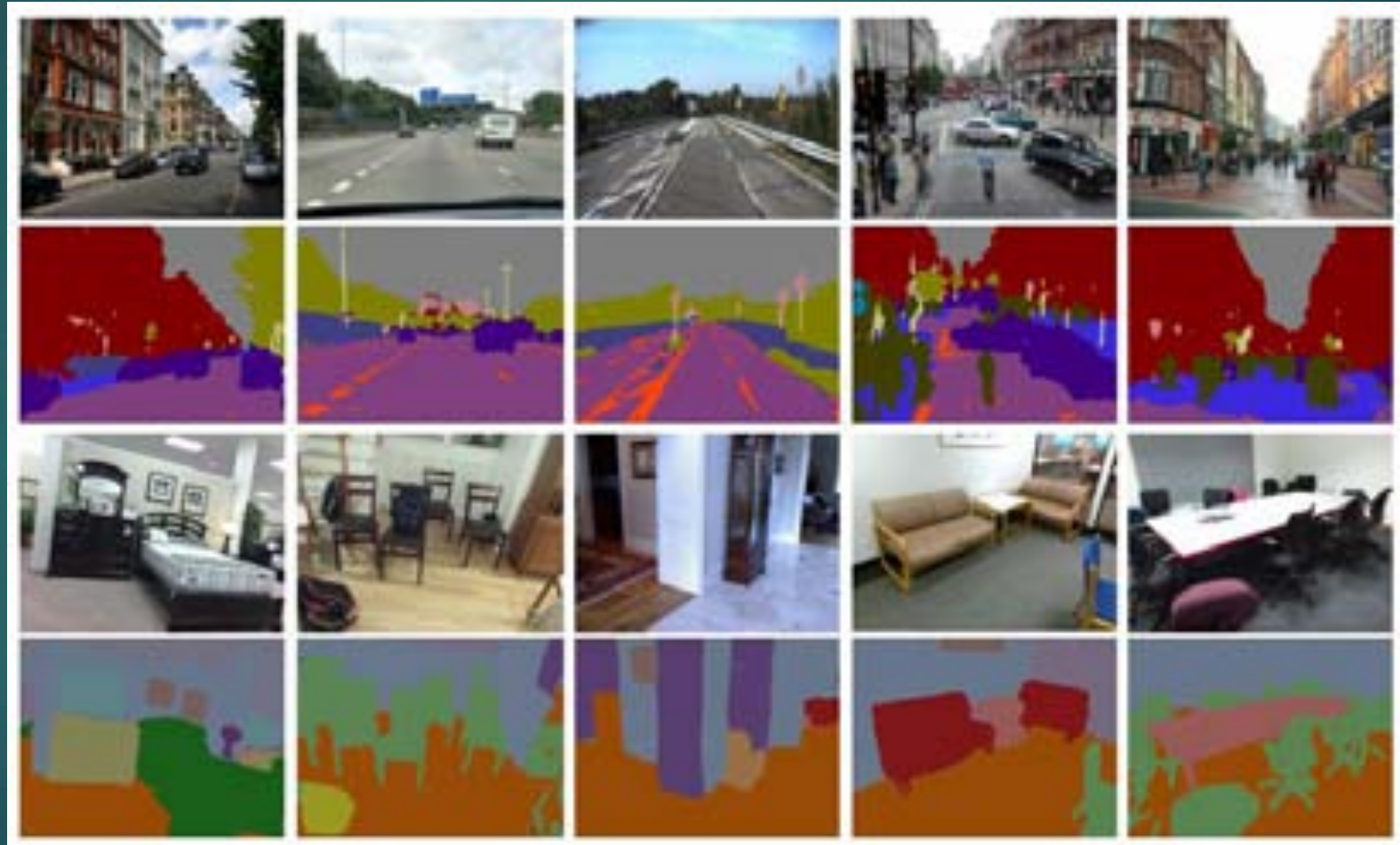
Recalage sémantique

28

- ▶ SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation, Badrinarayanan et al., CoRR 2015

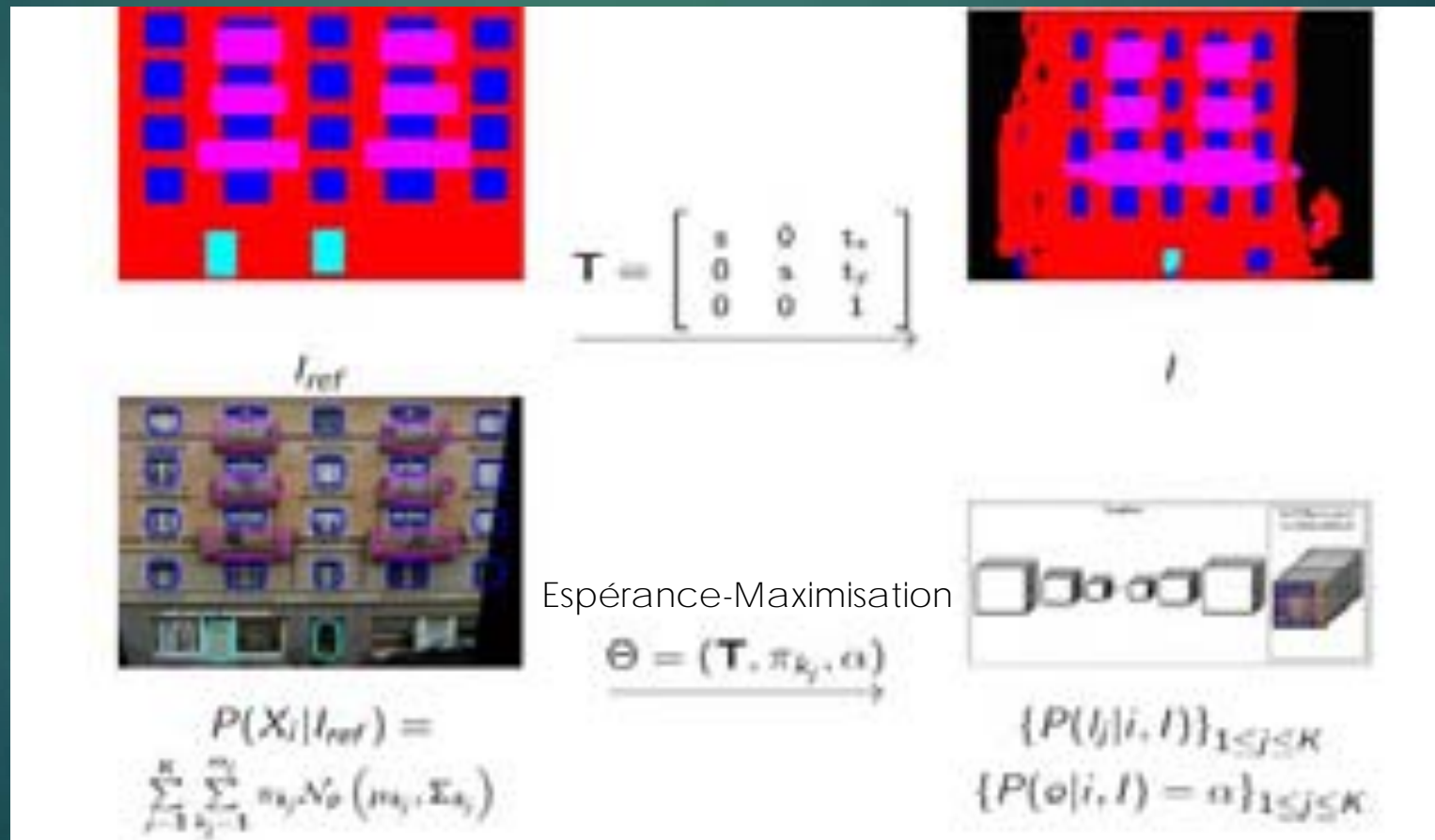


Recalage sémantique



Recalage sémantique

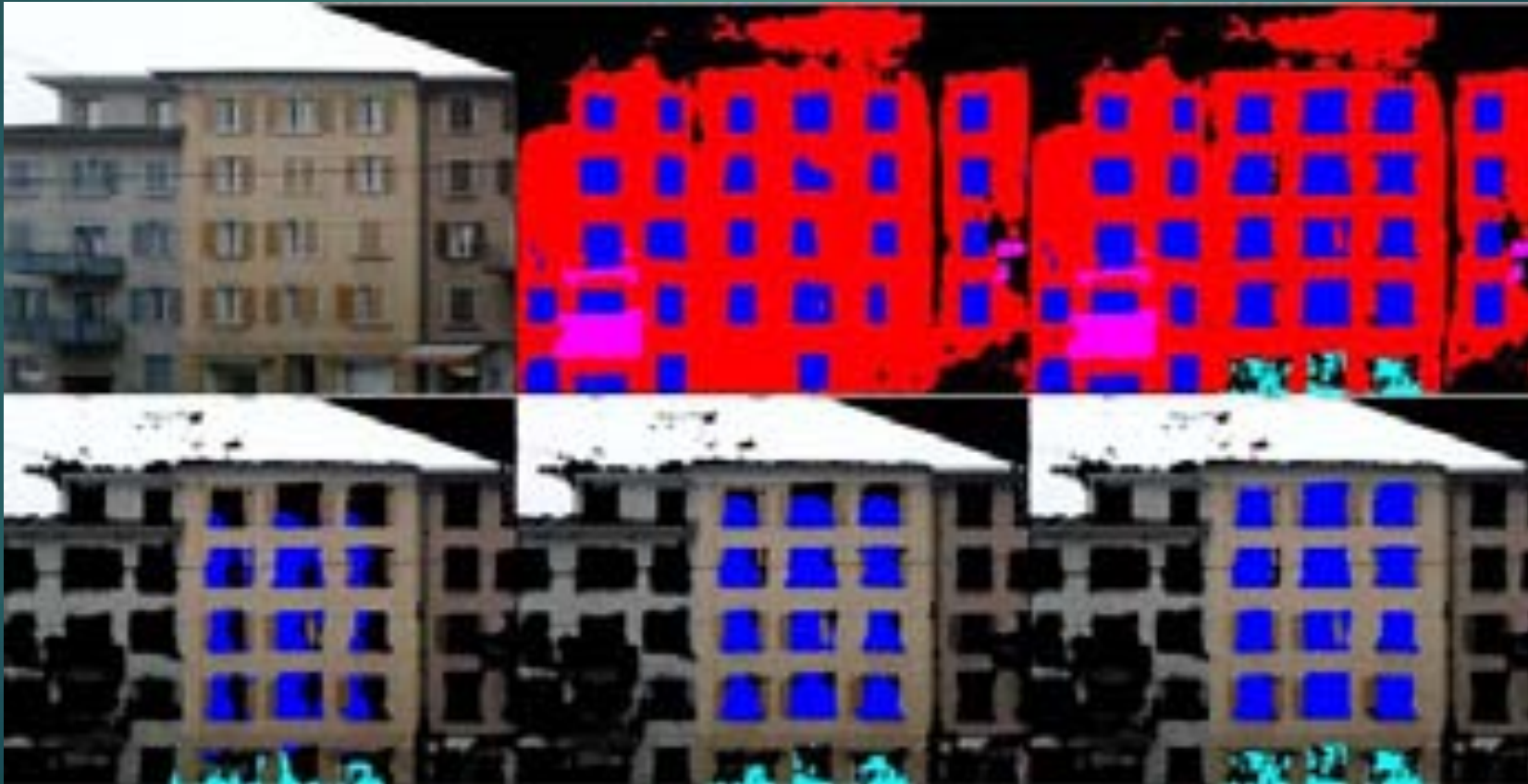
- ▶ Model-image registration of a building's facade based on dense semantic segmentation. [Fond et al. 2021]



Recalage sémantique

31

- ▶ Model-image registration of a building's facade based on dense semantic segmentation. [Fond et al. 2021]



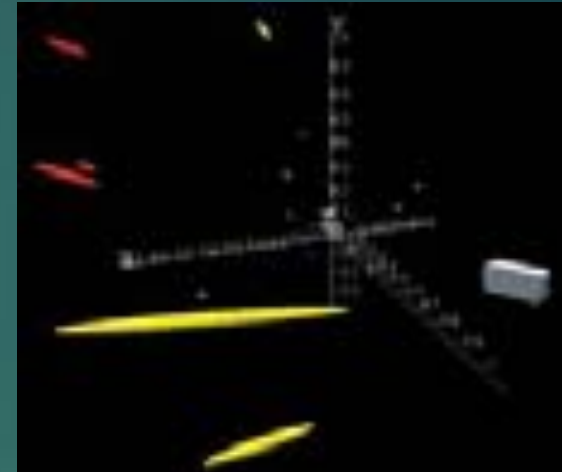
4. Odométrie visuelle

4.2 SIMULTANEOUS LOCALIZATION AND MAPPING (SLAM)

La méthode MONOSLAM

(monocular simultaneous localisation and mapping)

- ▶ [Davison et al., PAMI 2007] : estimation simultanée de la position de la caméra (*localisation*) et de la carte 3D des points détectés (*mapping*)
- ▶ Processus amorcé à l'aide d'une cible connue (poster, ...)
- ▶ Filtrage temporel d'un vecteur d'état contenant la vue courante et les points 3D de la carte, associé à une matrice de covariance (⇒ ellipsoïdes d'incertitudes) :



$$\hat{\mathbf{x}} = \begin{pmatrix} \hat{\mathbf{x}}_c \\ \hat{\mathbf{y}}_1 \\ \hat{\mathbf{y}}_2 \\ \vdots \end{pmatrix}, \quad \mathbf{P} = \begin{bmatrix} \mathbf{P}_{xx} & \mathbf{P}_{xy1} & \mathbf{P}_{xy2} & \dots \\ \mathbf{P}_{yx1} & \mathbf{P}_{y1y1} & \mathbf{P}_{y1y2} & \dots \\ \mathbf{P}_{yx2} & \mathbf{P}_{y2y1} & \mathbf{P}_{y2y2} & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

$$\mathbf{x}_c = \begin{pmatrix} \mathbf{r}^{(W)} \\ \mathbf{q}^{(W)} \\ \mathbf{v}^{(W)} \\ \omega^A \end{pmatrix}$$

La méthode MONOSLAM

(monocular simultaneous localisation and mapping)

34

- ▶ Application à la RA

Real-Time
Camera Tracking
in Unknown Scenes

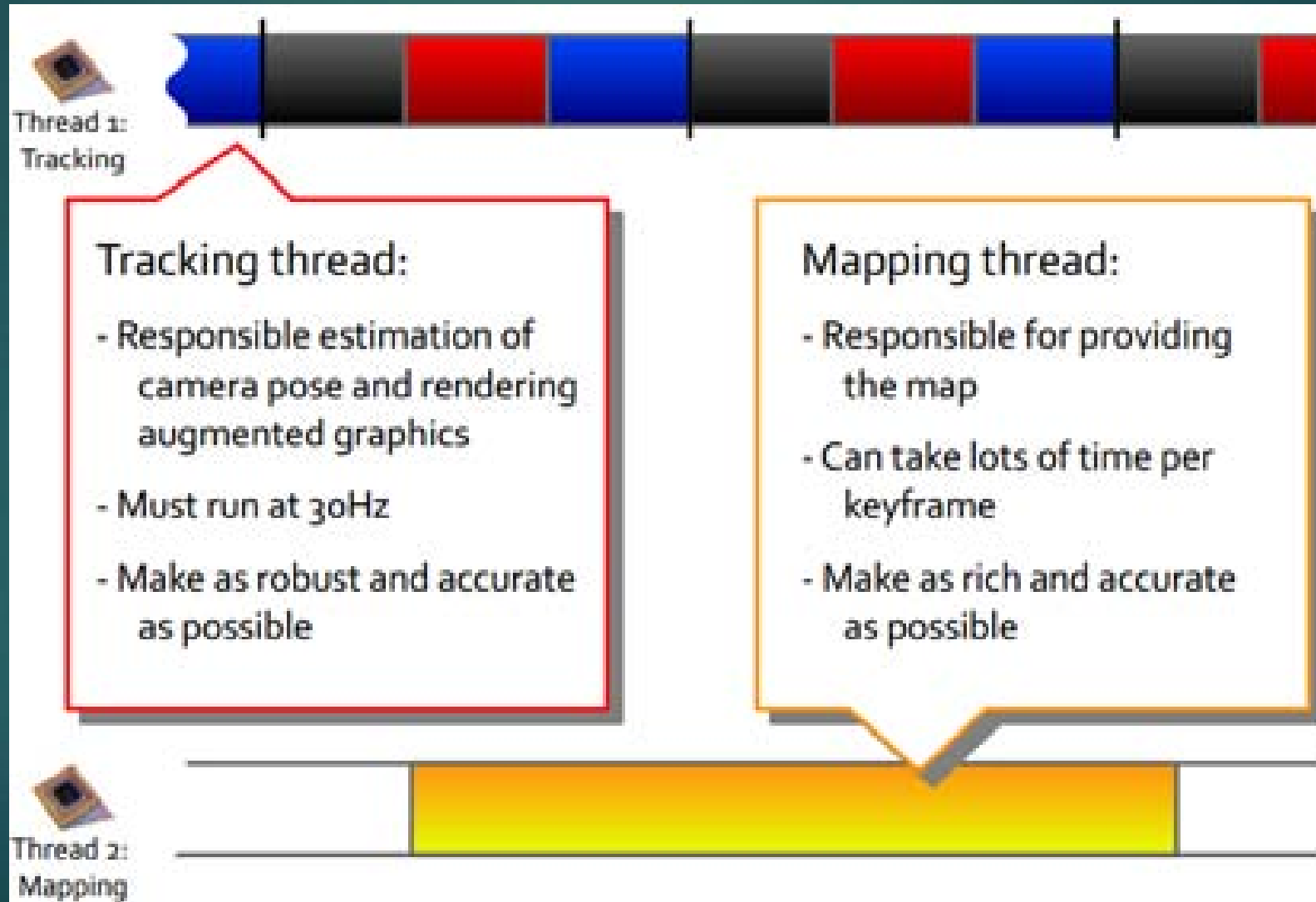
La méthode PTAM

(parallel tracking and mapping)

- ▶ Problèmes avec MonoSLAM :
 - ▶ Mettre à jour toute la carte 3D à chaque instant est très coûteux
 - ▶ L'algorithme est sensible aux corruptions de la carte 3D
- ▶ ⇒ introduction de l'algorithme PTAM [Klein & Murray ISMAR'07]
- ▶ Idée principale : le mapping est dissocié du tracking et effectué en parallèle sur un thread différent
- ▶ La tracking a lieu en temps réel
- ▶ Le mapping n'a pas besoin d'être revu à chaque instant, un bundle adjustment basé sur des images clés est lancé régulièrement (SFM classique, évite les problèmes de filtrage)

La méthode PTAM

(parallel tracking and mapping)



La méthode PTAM

(parallel tracking and mapping)

37

Parallel Tracking and Mapping
for Small AR Workspaces

ISMAR 2007 video results

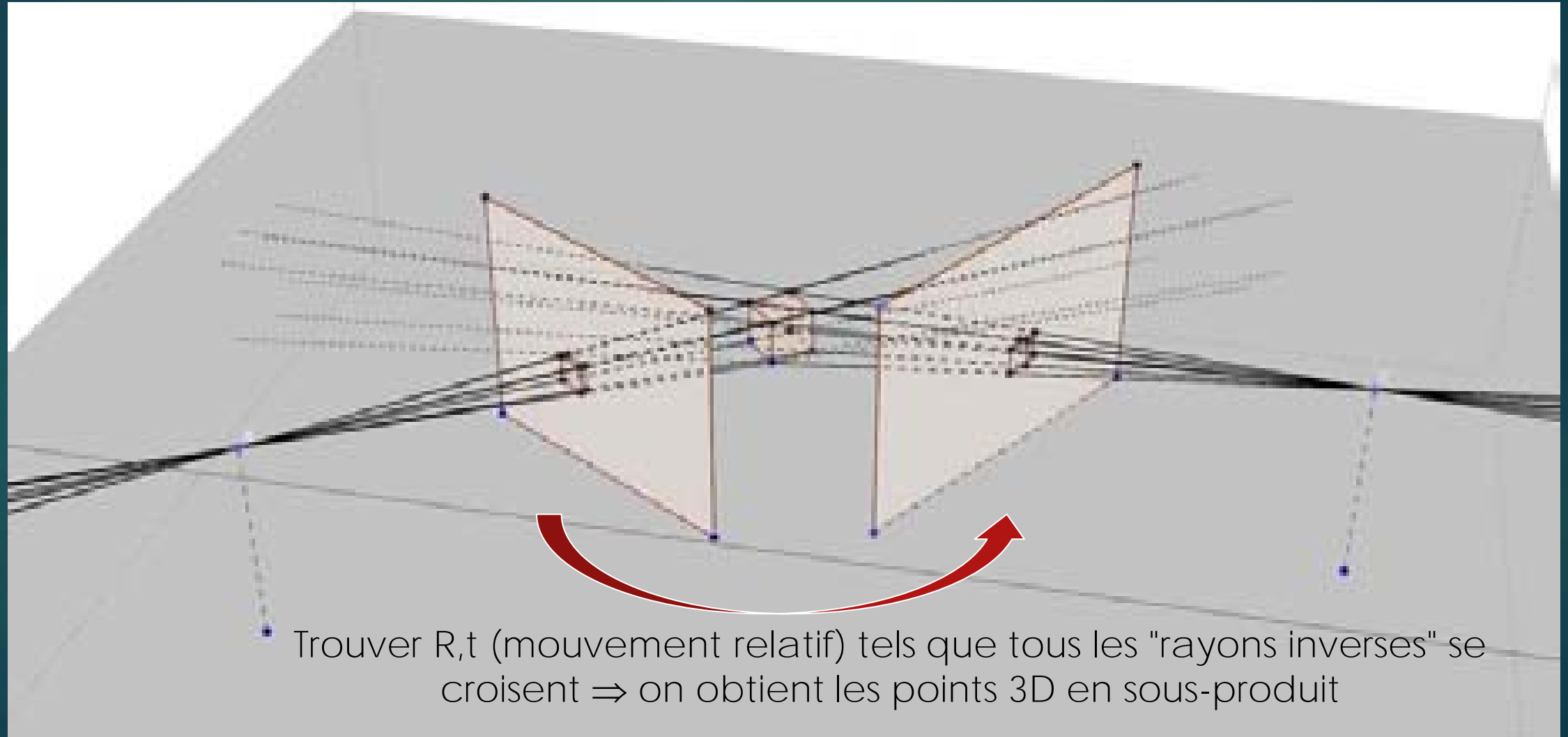
Georg Klein and David Murray
Active Vision Laboratory
University of Oxford

4. Odométrie visuelle

4.3 MÉTHODES DENSES

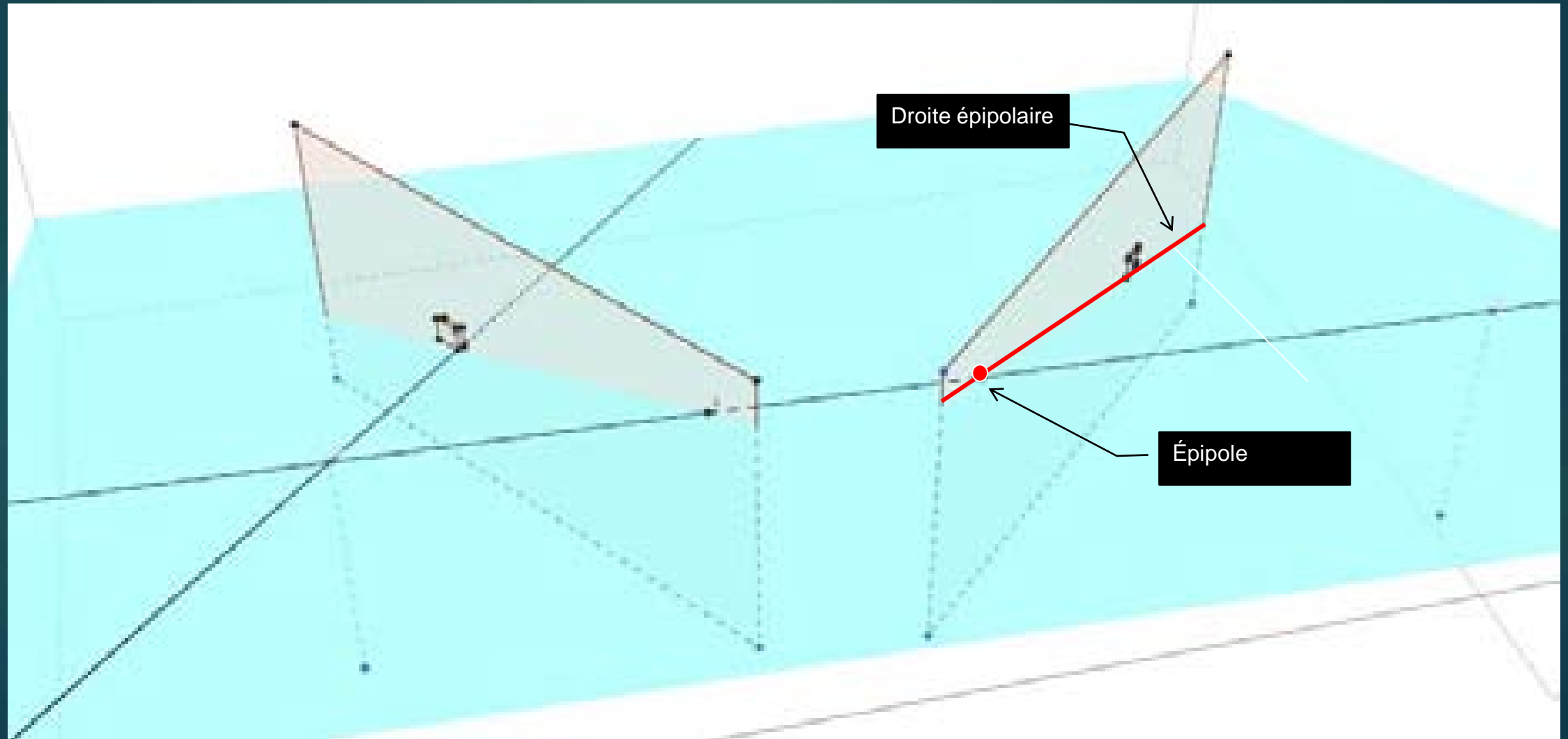
Stéréovision

39



La contrainte épipolaire

40



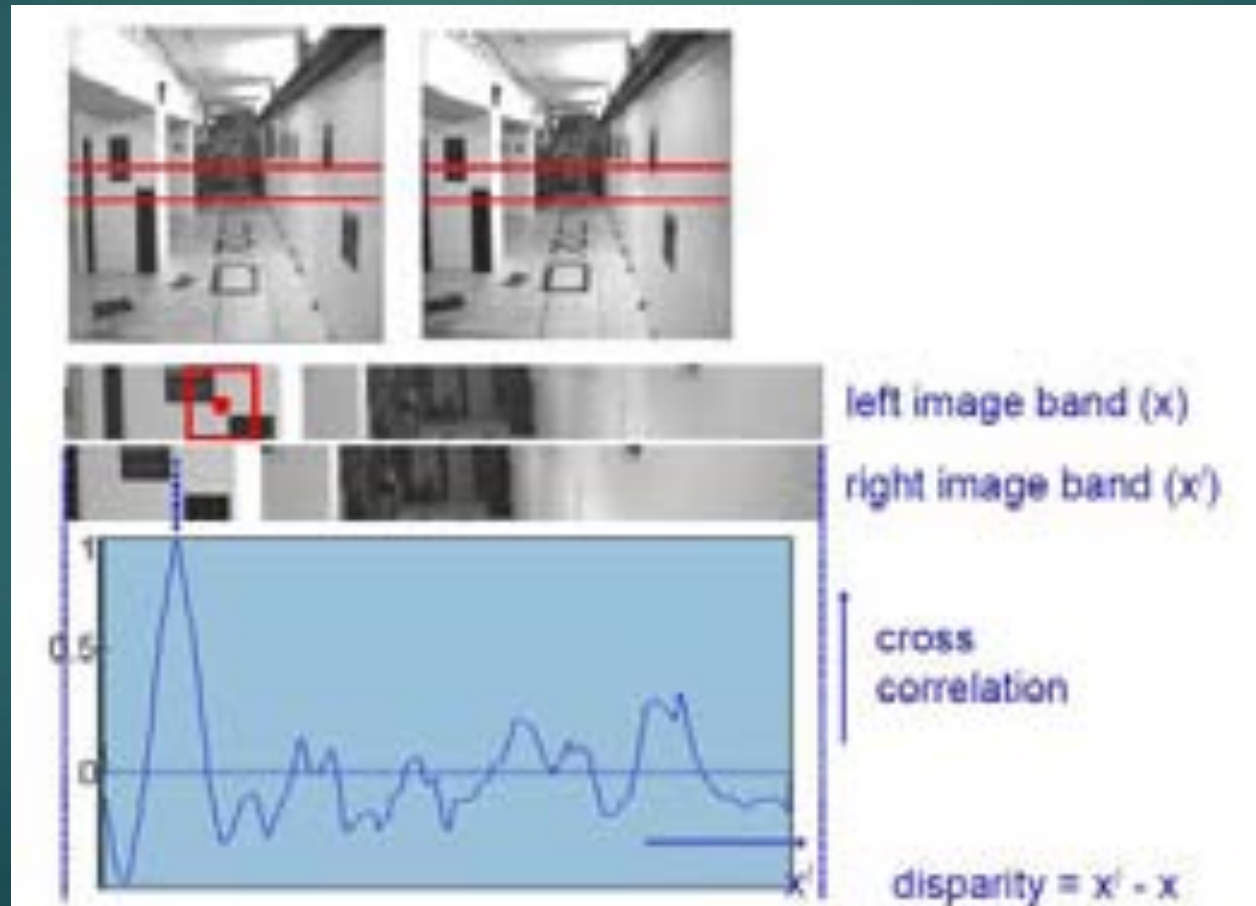
La contrainte épipolaire



Courtesy Oxford, Visual Geometry Group

La contrainte épipolaire

- Utile pour l'appariement de points (exemple sur deux images rectifiées)



Exemple de reconstruction dense

43

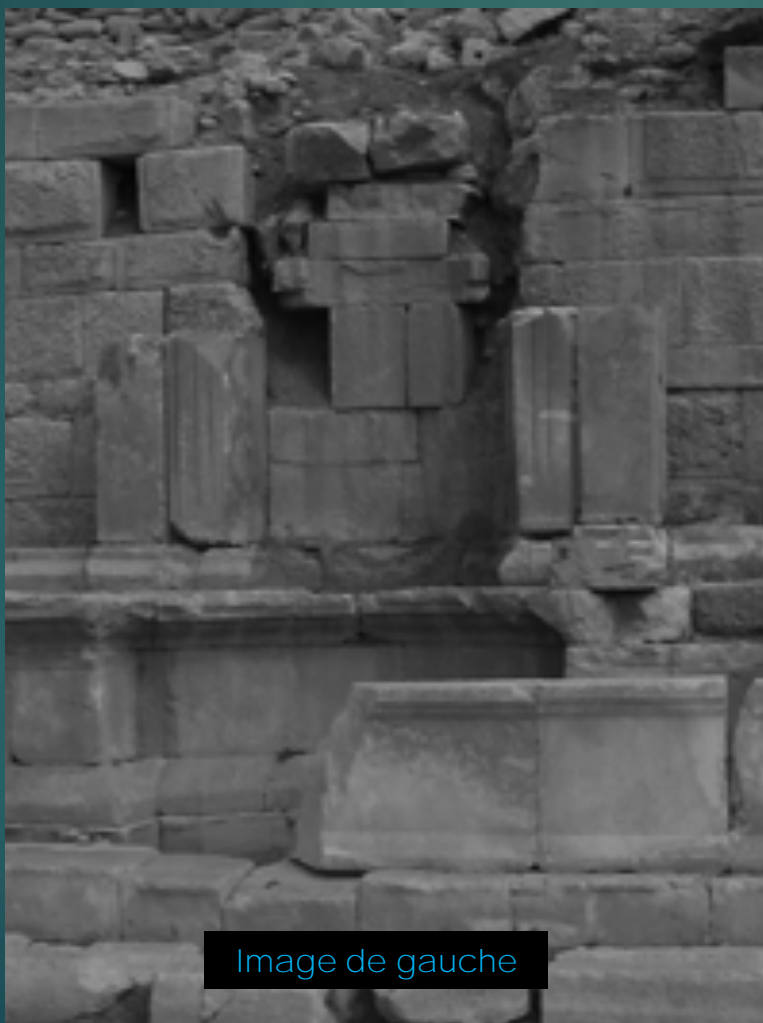


Image de gauche



Image de droite

Exemple de reconstruction dense

44



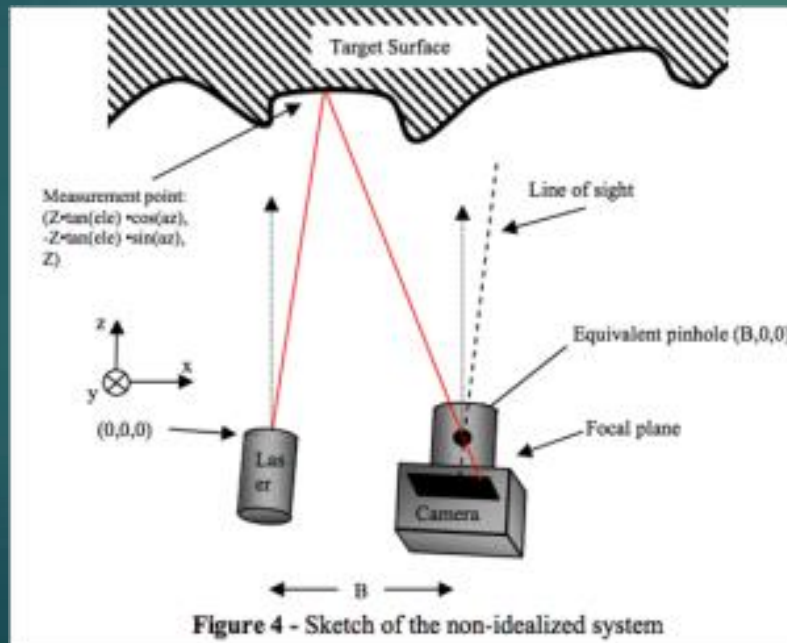
Carte de
profondeur



Vue du nuage de
points reconstruits
triangulé et texturé

Courtesy Oxford, Visual Geometry Group

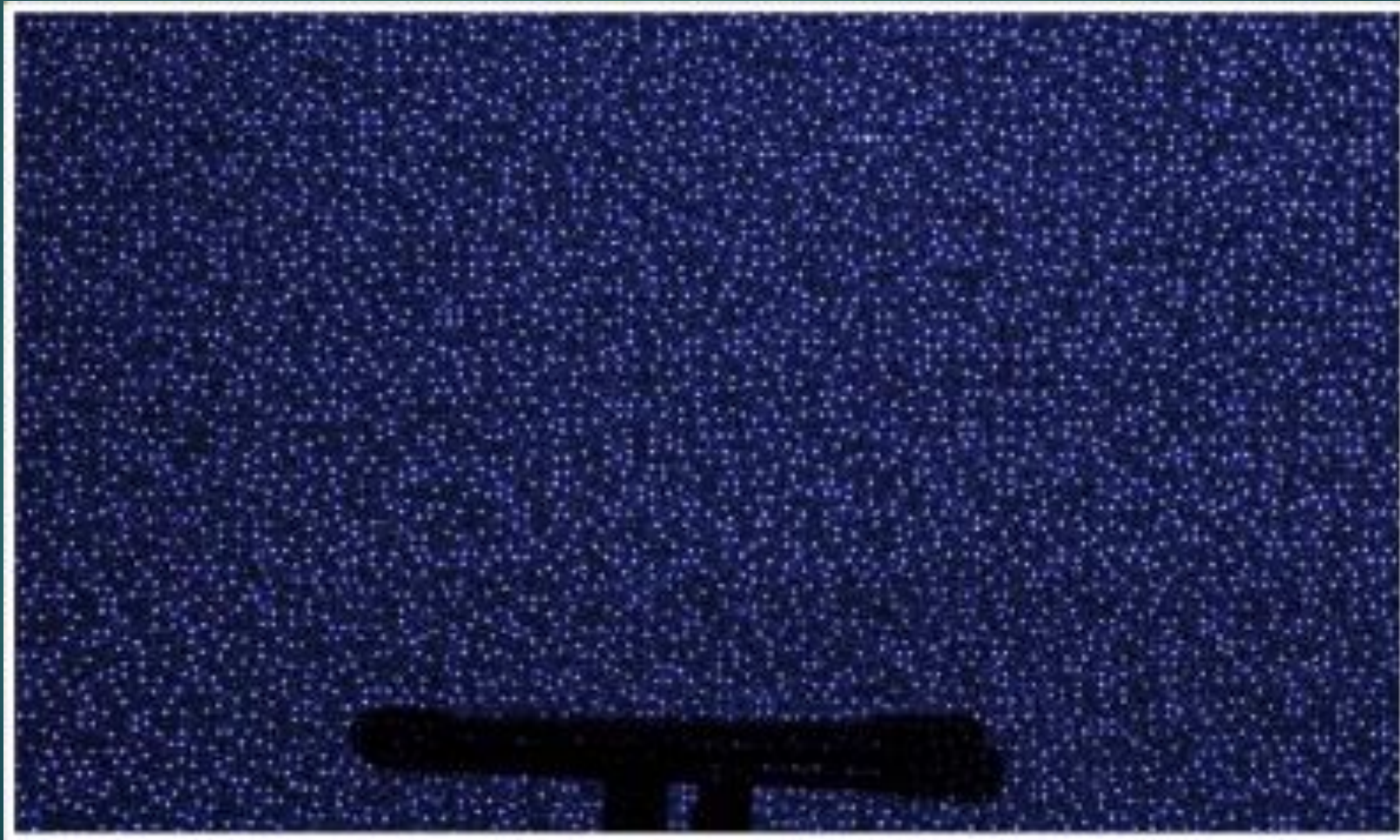
Avec la Kinect...



Avec la Kinect...

46

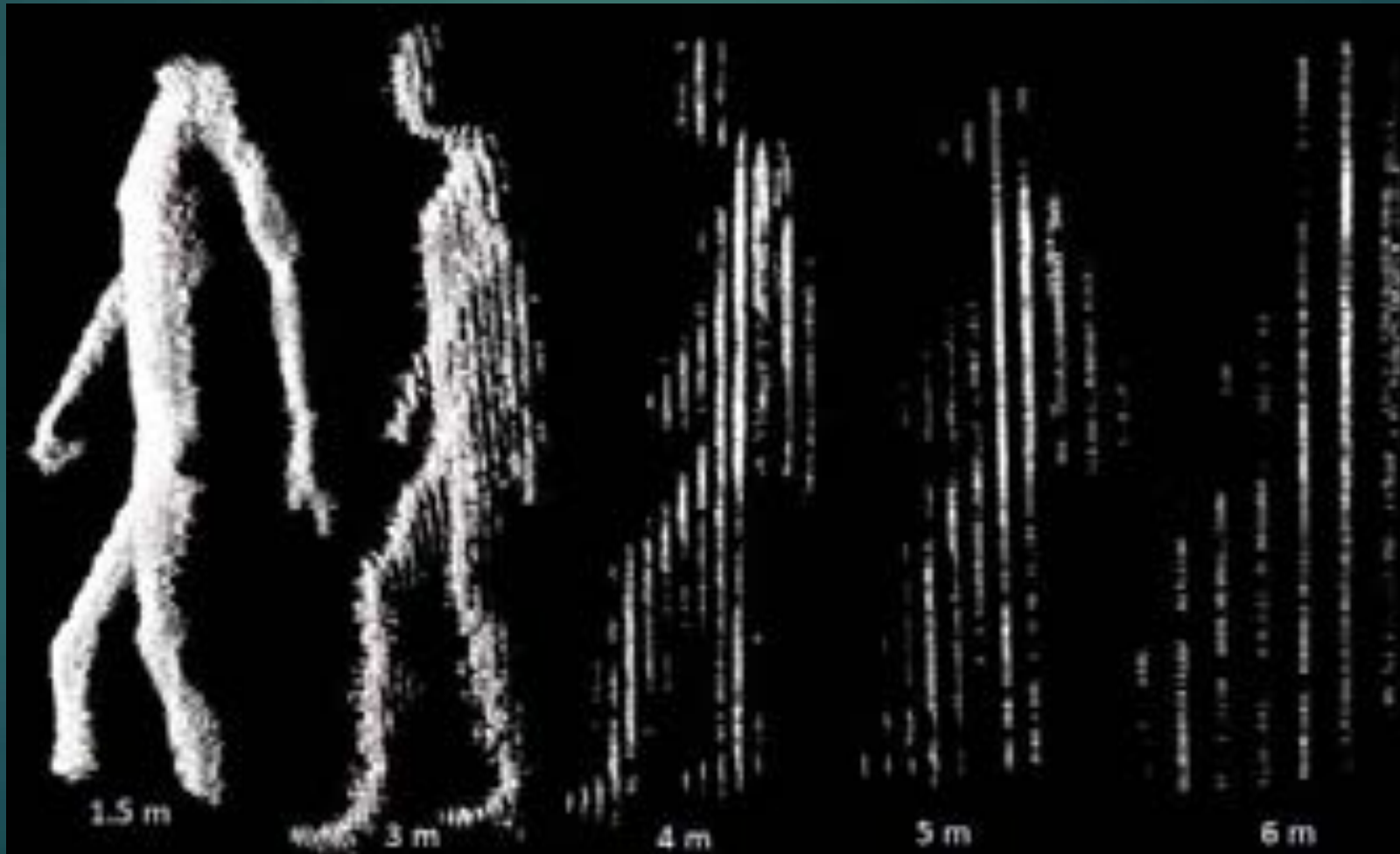
- ▶ Motifs IR



Avec la Kinect...

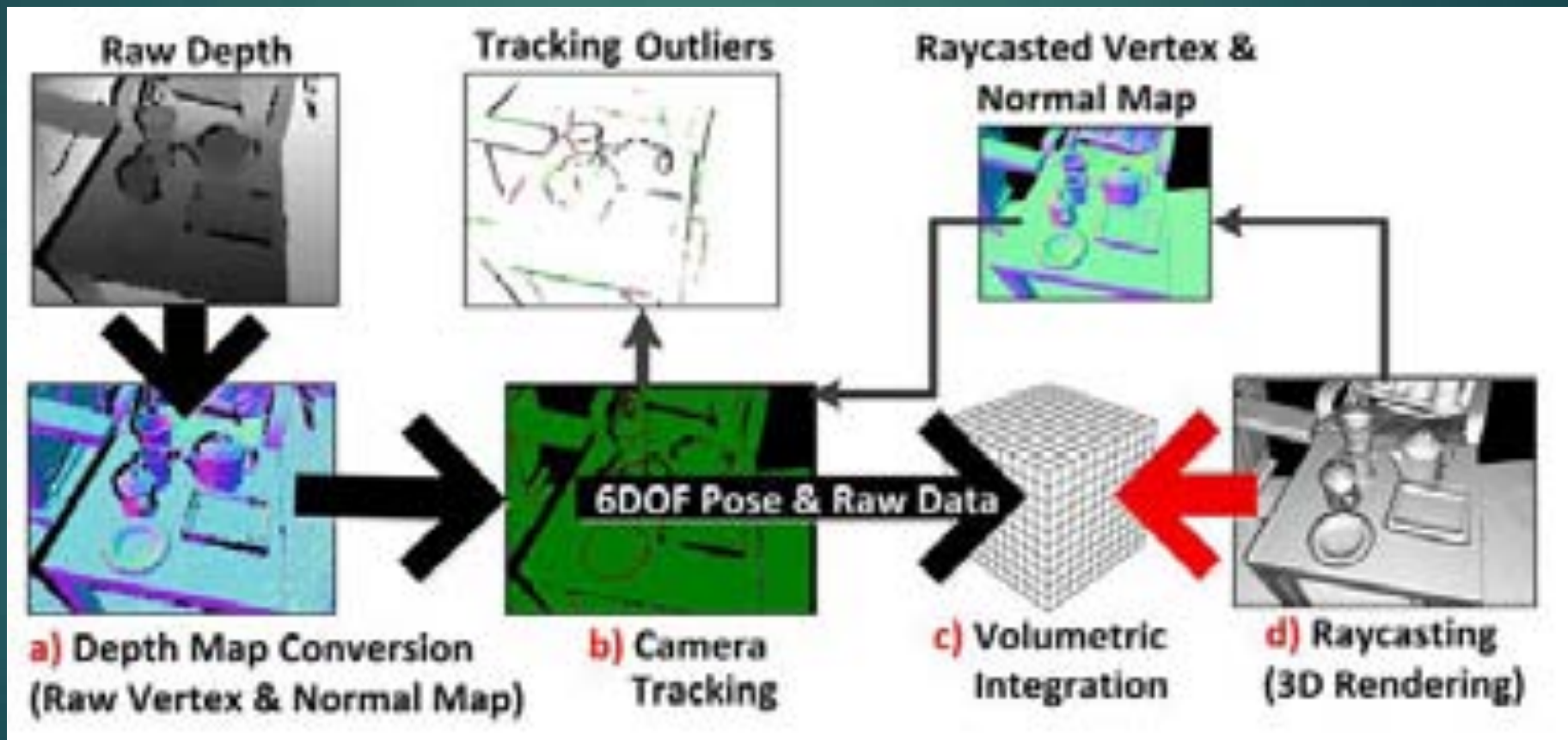
47

- ▶ Précision vs profondeur (source : K. Litomisky, UC)



L'agorithme KinectFusion

[Newcombe et al. ISMAR 2011]



L'algorithme KinectFusion

[Newcombe et al. ISMAR 2011]

SIGGRAPH Talks 2011

KinectFusion:

**Real-Time Dynamic 3D Surface
Reconstruction and Interaction**

**Shahram Izadi 1, Richard Newcombe 2, David Kim 1,3, Otmar Hilliges 1,
David Molyneaux 1,4, Pushmeet Kohli 1, Jamie Shotton 1,
Steve Hodges 1, Dustin Freeman 5, Andrew Davison 2, Andrew Fitzgibbon 1**

1 Microsoft Research Cambridge 2 Imperial College London

3 Newcastle University

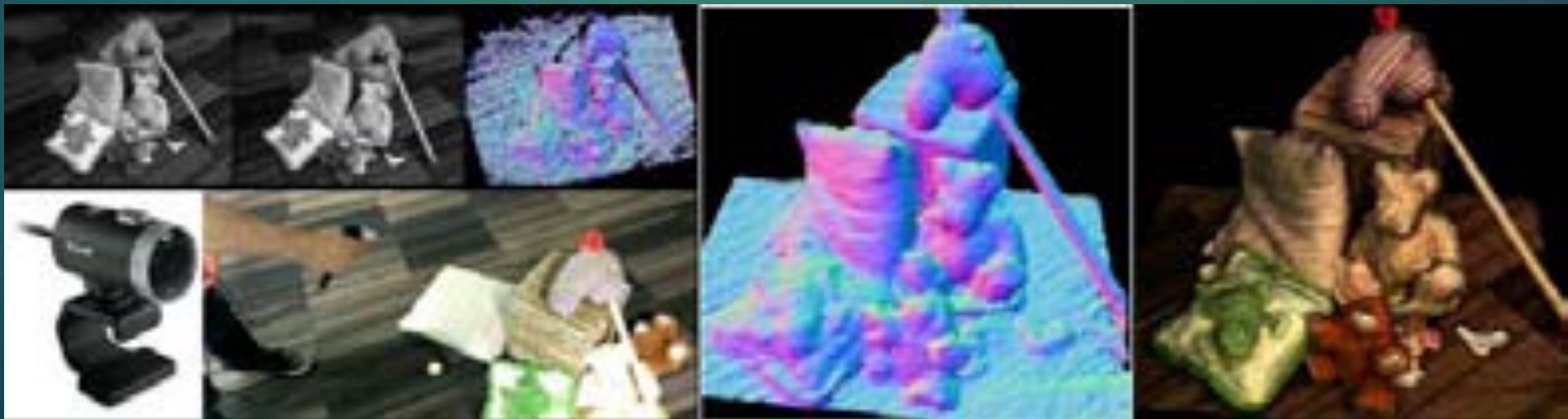
4 Lancaster University

5 University of Toronto

L'algorithmes MonoFusion

50

- ▶ Reconstruction dense de petites scènes avec une simple webcam [Pradeep et al. ISMAR 2013]
- ▶ PTAM pour le tracking de caméra
- ▶ Reconstruction dense entre une vue clé et la vue courante (P et P' sont donnés par PTAM)
- ▶ Extension et correction de la reconstruction dense en utilisant l'algorithmes KinectFusion



Gaussian Splatting SLAM

51

arXiv 12.2023

Hidehobu Matsuki^{1*}

Riku Murai^{2*}

Paul H. J. Kelly²

Andrew J. Davison¹

¹Dyson Robotics Laboratory, Imperial College London

²Software Performance Optimisation Group, Imperial College London

Website: <https://rmurai.co.uk/projects/GaussianSplattingSLAM/>

Video: https://youtu.be/x604ghp9R_Q/



Figure 1. From a single monocular camera, we reconstruct a high fidelity 3D scene live at 30fps. For every incoming RGB frame, 3D Gaussians are incrementally formed and optimised together with the camera poses. We show both the rasterised Gaussians (left) and Gaussians shaded to highlight the geometry (right). Notice the details and the complex material properties (e.g. transparency) captured. Thin structures such as wires are accurately represented by numerous small, elongated Gaussians, and transparent objects are effectively represented by placing the Gaussians along the rim. Our system significantly advances the fidelity a live monocular SLAM system can capture.

Gaussian Splatting SLAM

52

https://www.youtube.com/watch?v=x604ghp9R_Q

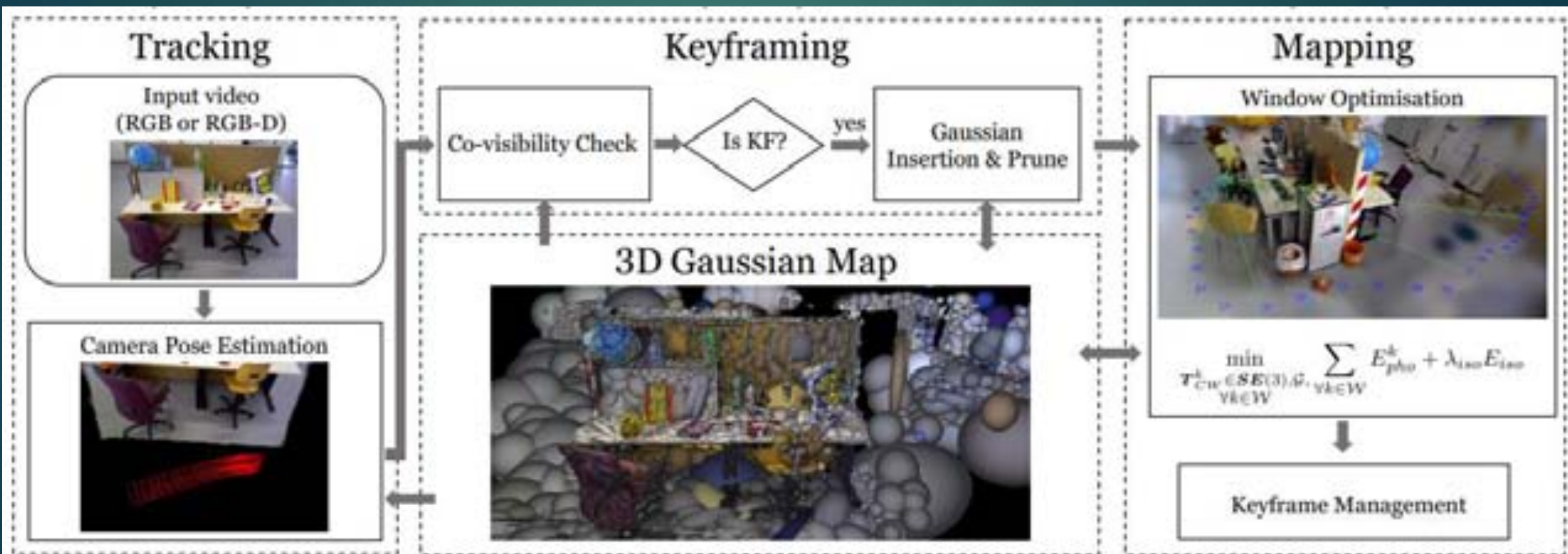


Figure 2. SLAM System Overview: Our SLAM system uses 3D Gaussians as the only representation, unifying all components of SLAM, including tracking, mapping, keyframe management, and novel view synthesis.

Principaux inconvénients de l'odométrie visuelle

- ▶ Problème de l'initialisation
 - ▶ Suivi de plans : quatre points indiqués dans la première image
 - ▶ Tracking par synthèse [Lepetit] : modèle aligné manuellement dans la première image
 - ▶ Tracking par synthèse [Reitmayr] : GPS + capteurs d'orientation pour initialiser
 - ▶ SLAM : initialisation manuelle ou utilisant un poster
 - ▶ Dans le cas du SLAM, il faut placer la scène virtuelle au sein du nuage de points reconstruit à la volée : impossible de préparer les intégrations 3D en amont
- ▶ Ces méthodes étant récursives (le résultat obtenu dans l'image i dépend de celui obtenu dans l'image $i-1$), des erreurs de précision peuvent s'accumuler au cours du temps
- ▶ Ces méthodes « plantent » dès que le suivi se passe mal entre deux images