L3 SD + EEEA - Traitement d'images avancé

Stéréovision

3ème partie

Caroline Petitjean

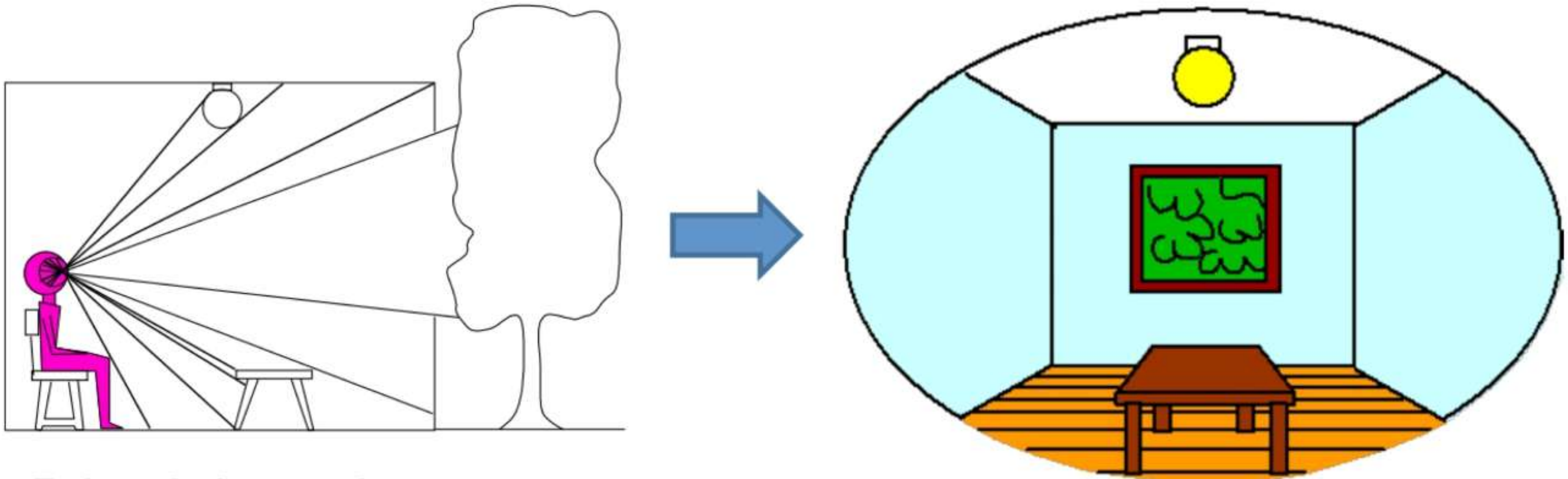
UNIVERSITÉ DE ROUEN NORMANDIE

Plan

- Intro, mouvement et profondeur
- Longueur focale d'une caméra
- Estimation de la profondeur avec 2 caméras parallèles
- Stéréo matching
- Stéréovision dense vs éparse
- Dataset KITTI

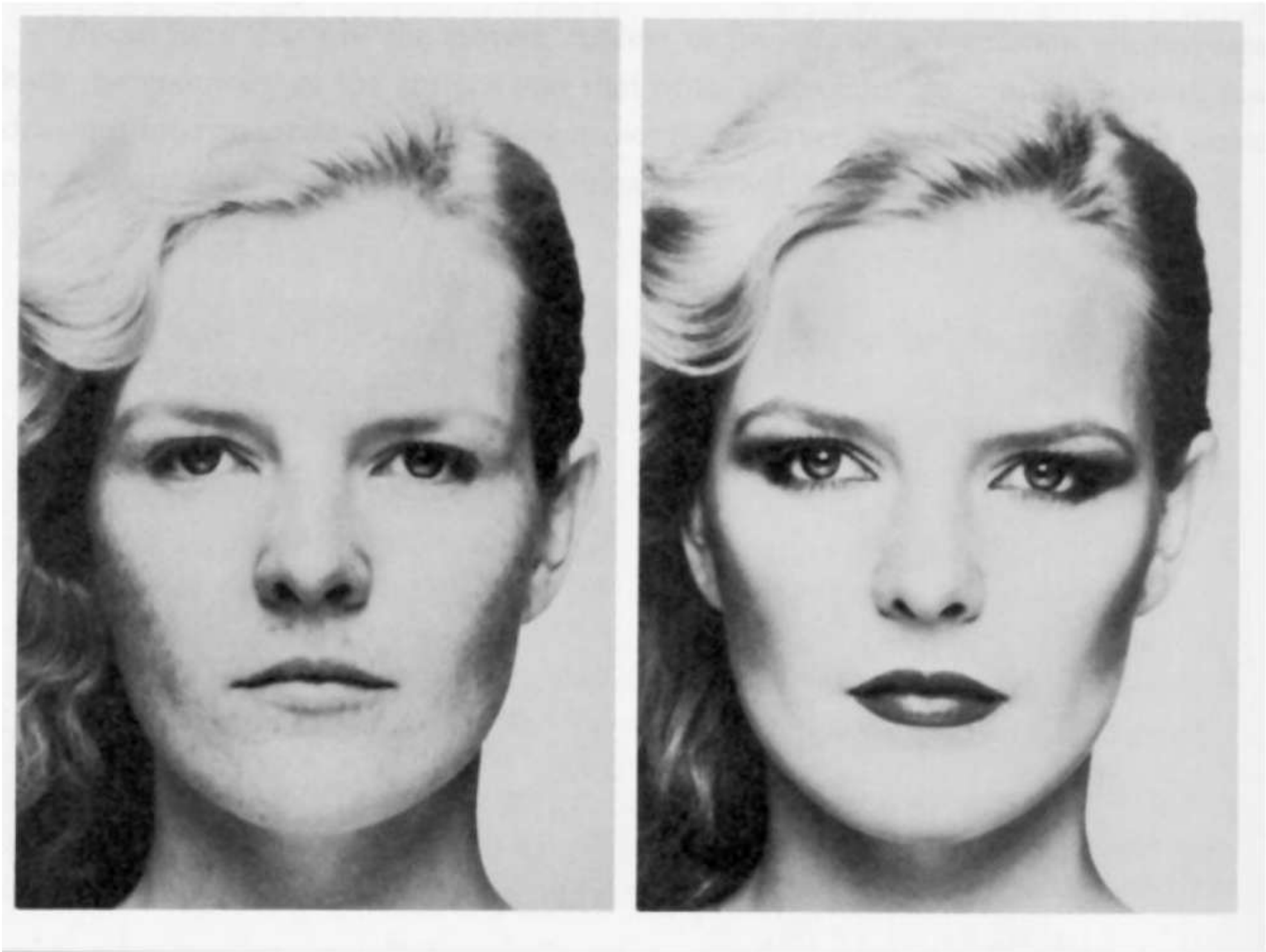
Réduction de la dimension

3D world *2D image*



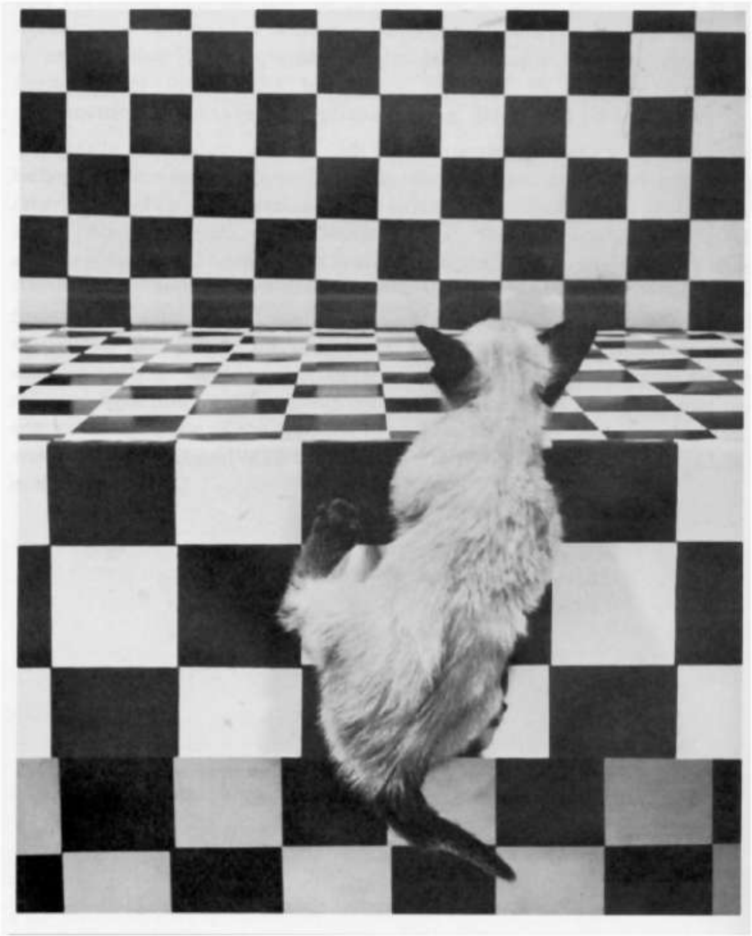
Point of observation

- Comment récupérer la 3D à partir des images en 2D ?



Merle Norman Cosmetics, Los Angeles

Slide credit: J. Hayes

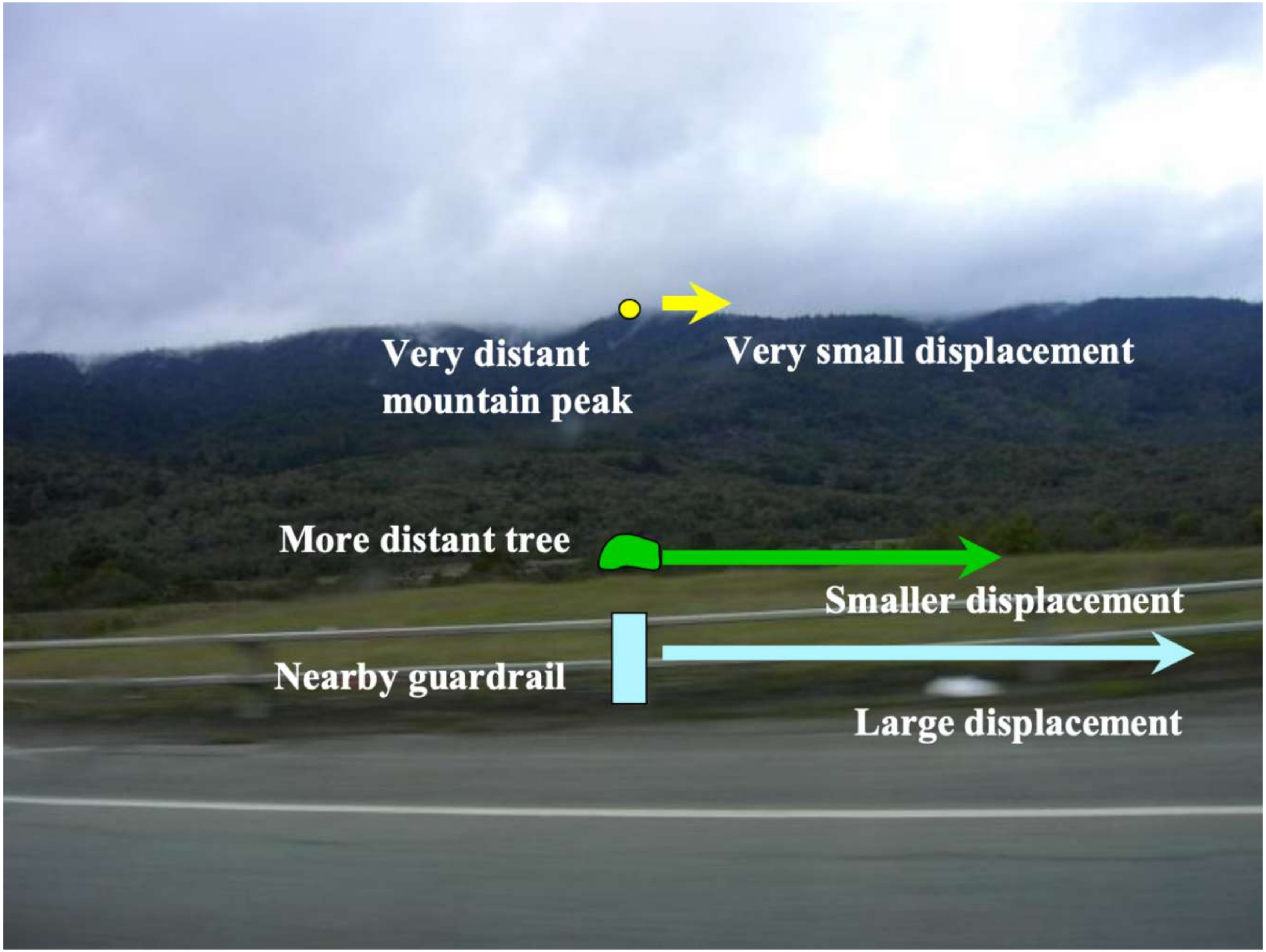


The Visual Cliff, by William Vandivert, 1960

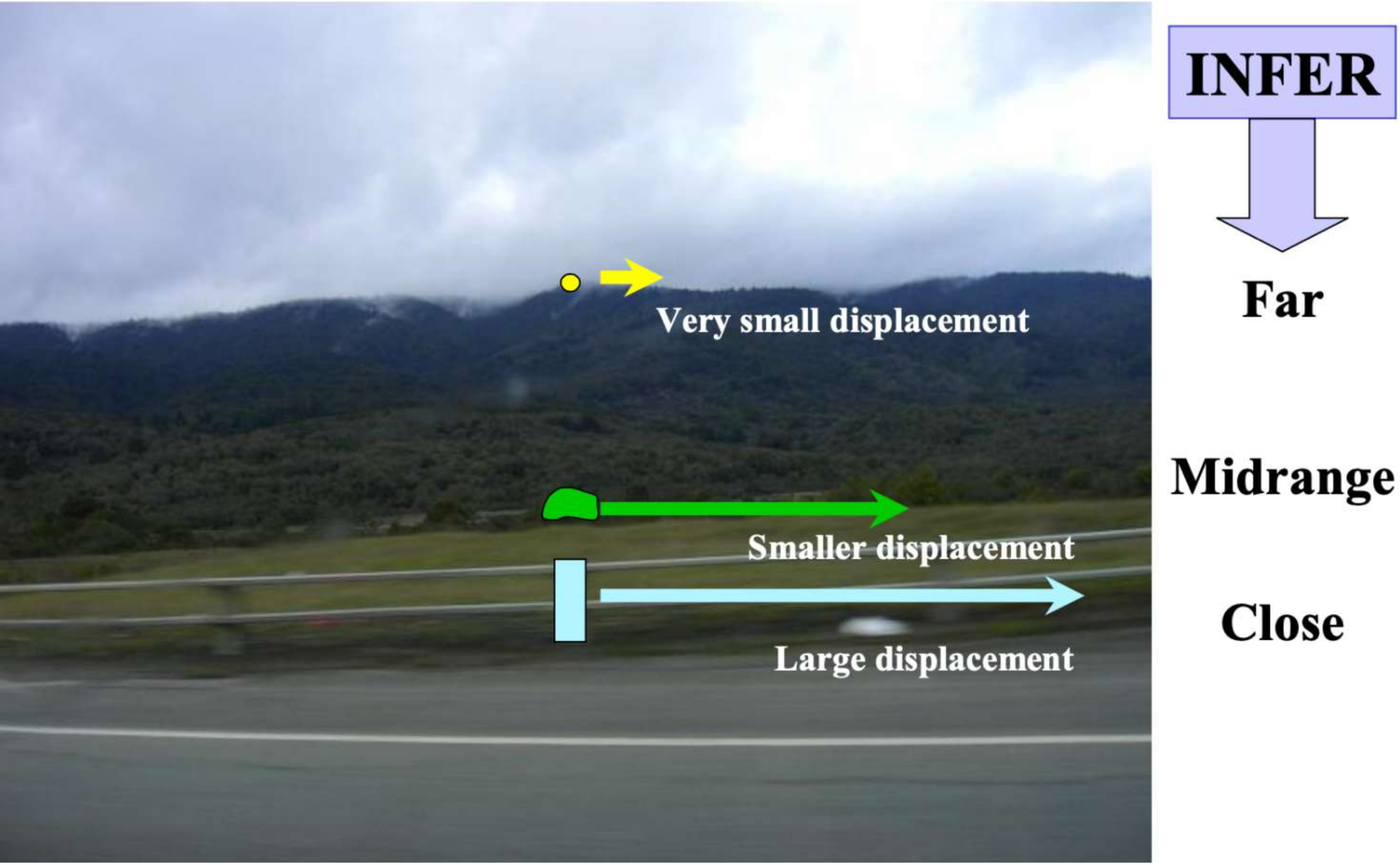
Focus

From *The Art of Photography*, Canon

Mouvement



Very distant
mountain peak
Very small displacement
More distant tree
Smaller displacement
Nearby guardrail
Large displacement


INFER
Far
Very small displacement
Midrange
Smaller displacement
Close
Large displacement

Dans ce cours

- On s'intéresse au fait d'estimer la profondeur d'une scène à partir du mouvement
- On dispose de 2 images acquises par 2 caméras

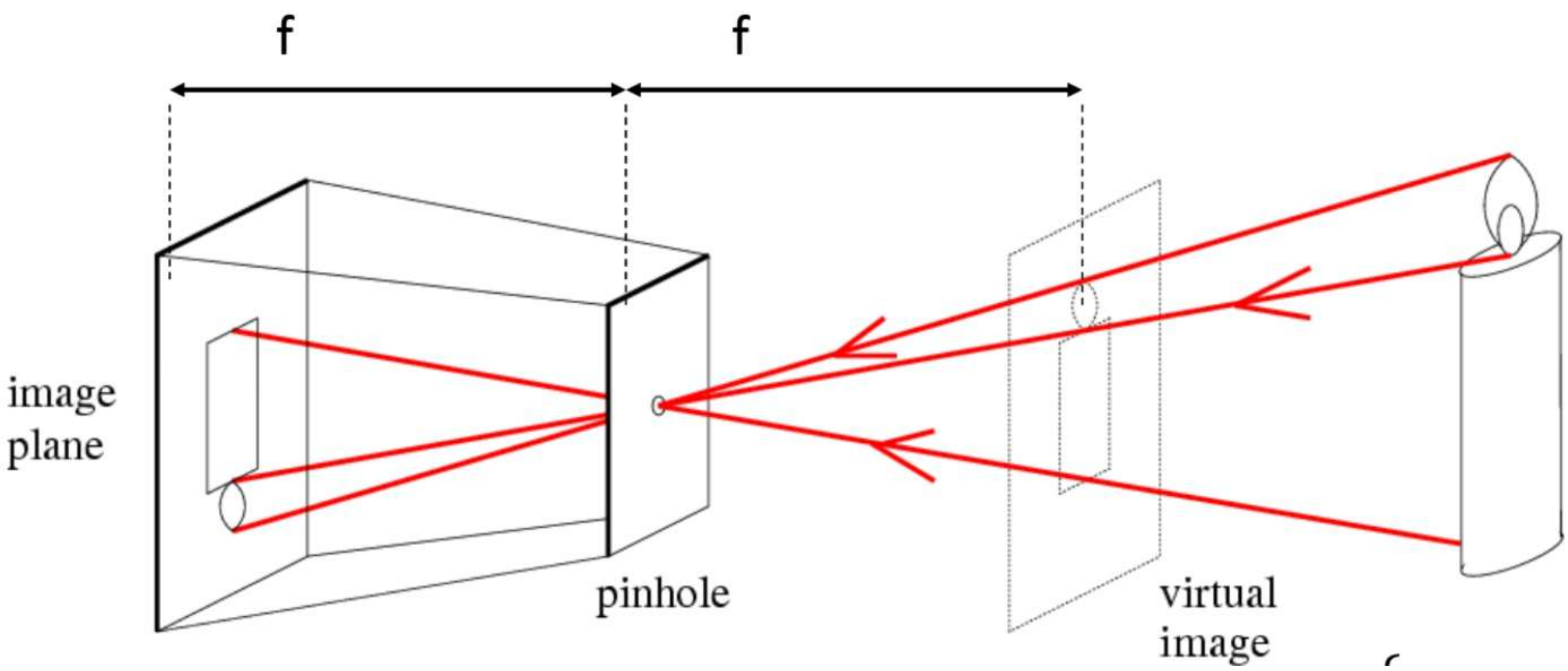
Acquisition binoculaire

Elphel

ZED / Stereolabs

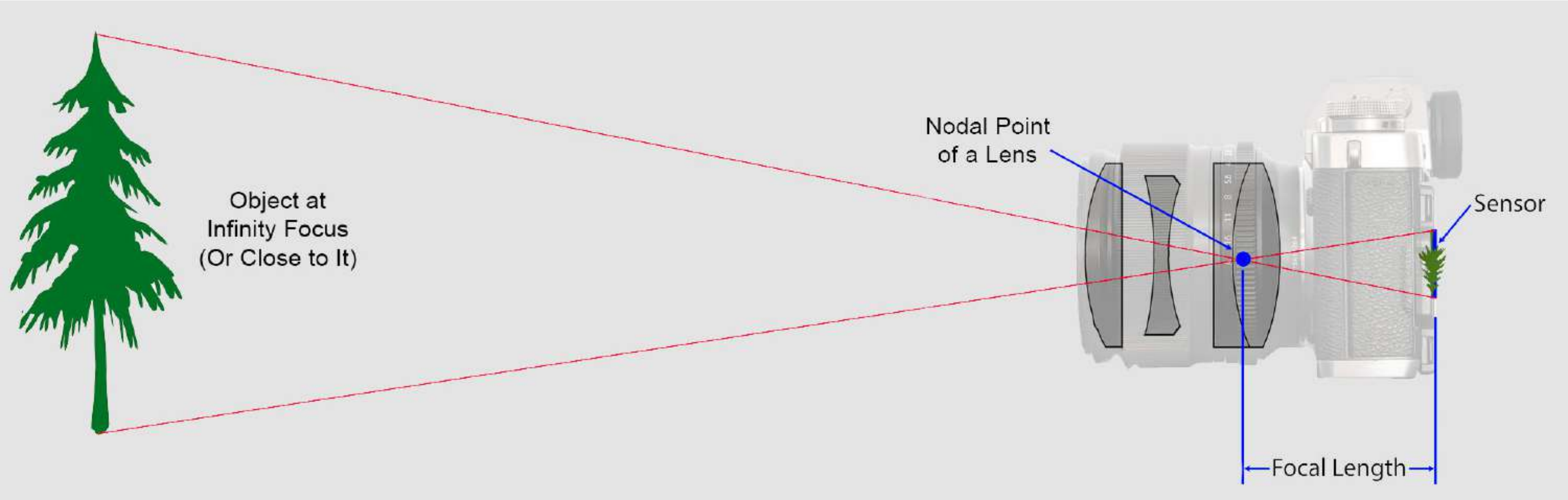
Acquisition binoculaire :
- Deux caméras, acquisition simultanée.
- Ou : une caméra qui se déplace (si scène statique !).
- Intérêt : acquisition de la profondeur.
- Applications : industrie, robotique, automobile (ADAS), réalité augmentée, cinéma, métrologie, cartographie, etc.

Pinhole caméra (sténopé)



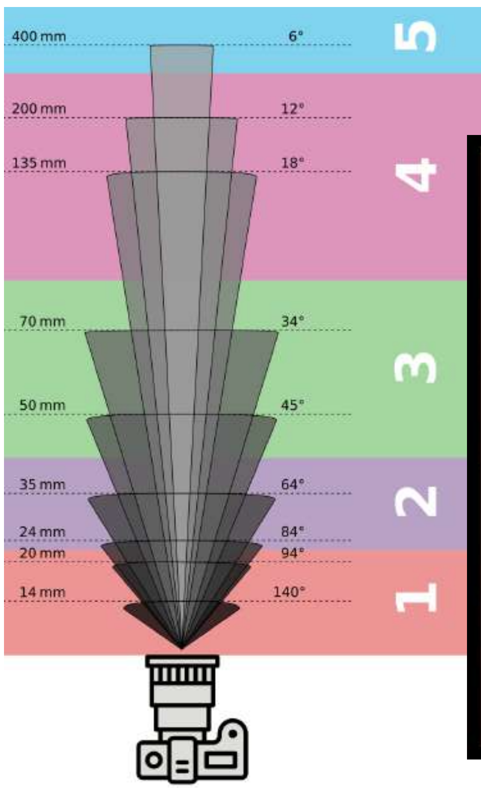
Habituellement, le plan de l'image est dessiné devant le point focal

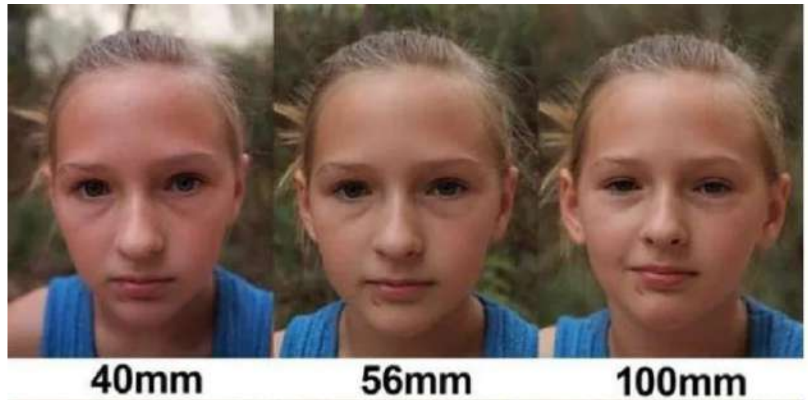
Longueur focale



https://photographylife.com/what-is-focal-length-in-photography

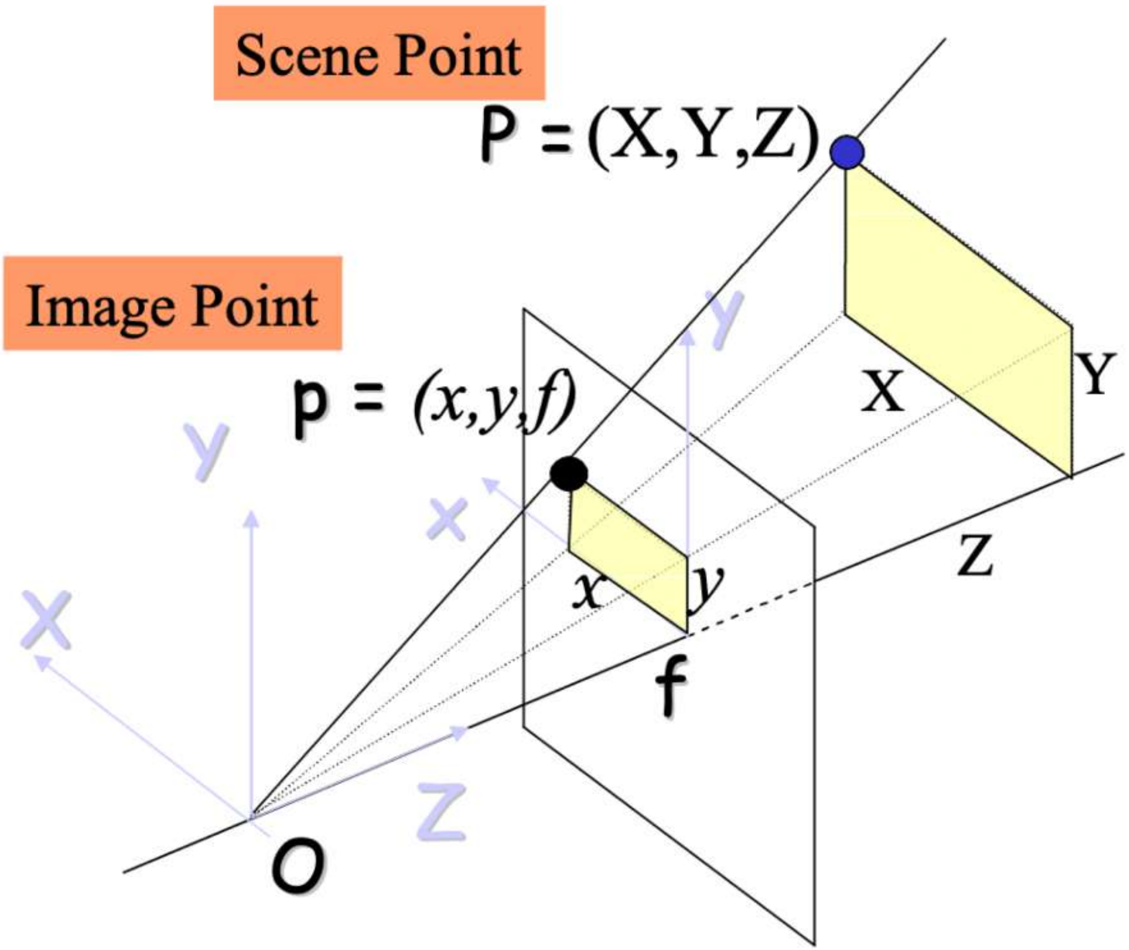
Variation de la longeur focale





focal-length-lesson-explains-

logs/phone-cameras-selfies-
distort-face-making-face-nose-look-wider-longer/

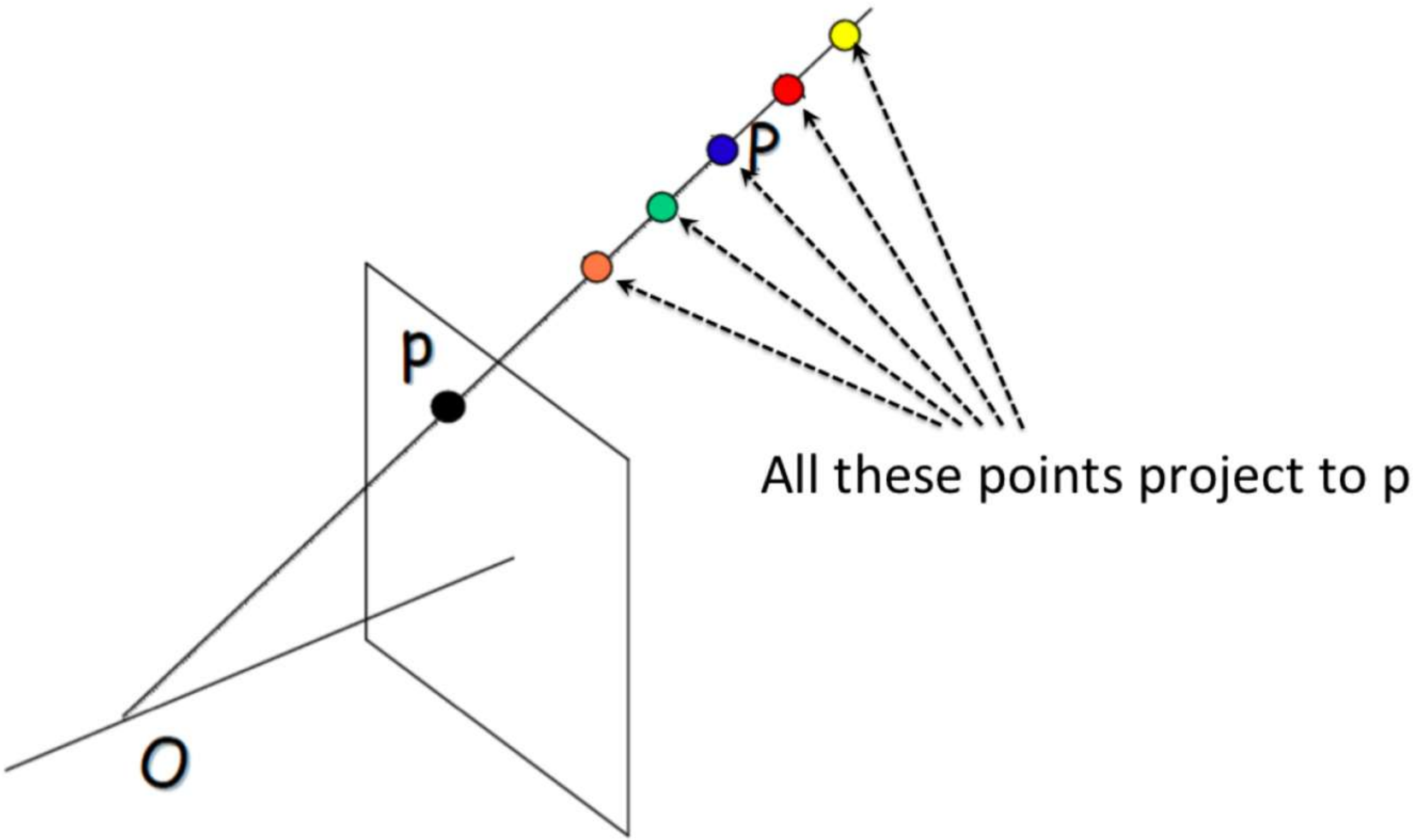


Perspective Projection Eqns

x en fonction de X, Z, f ?

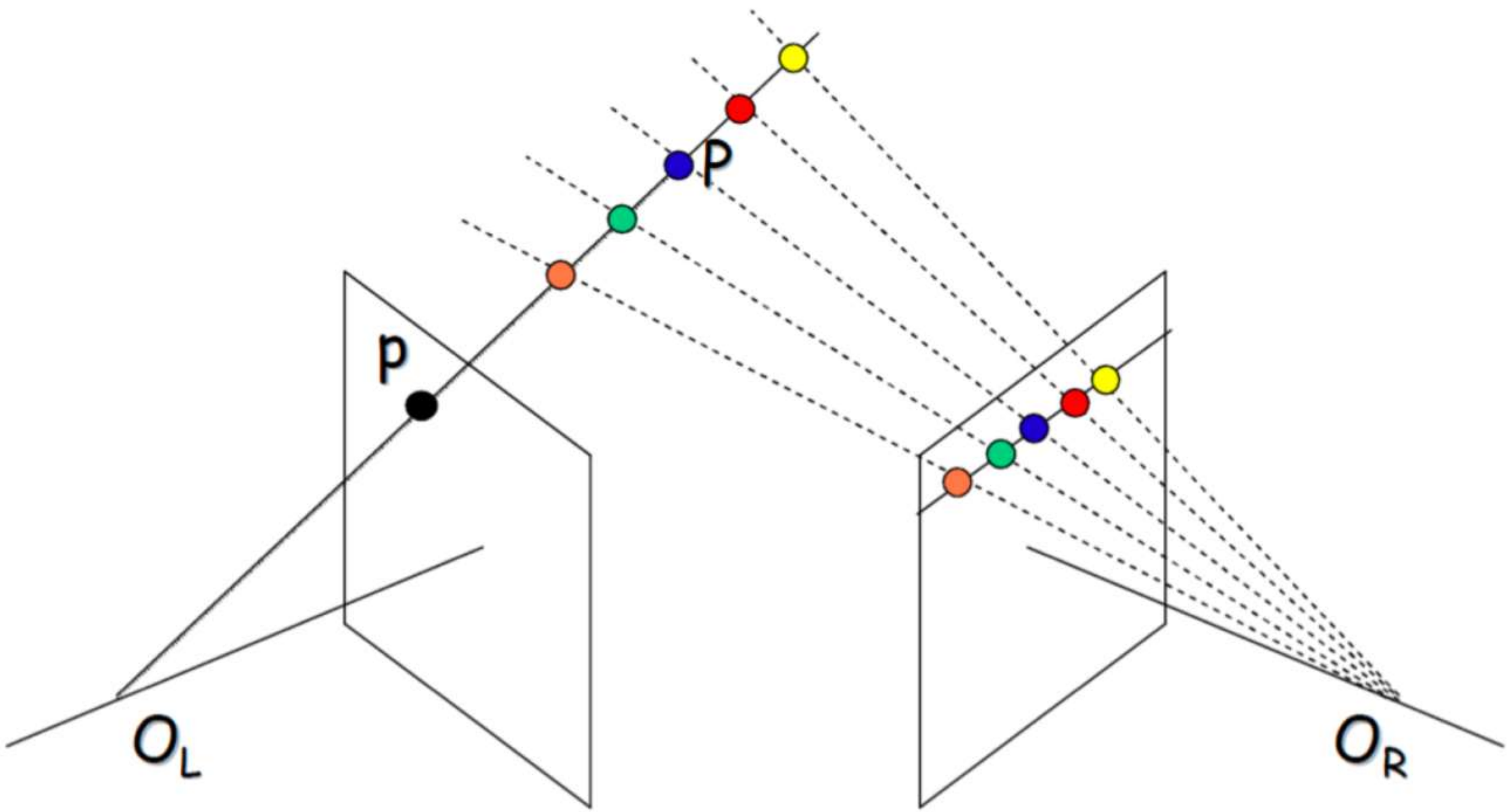
y en fonction de Y, Z, f ?

Une seule caméra

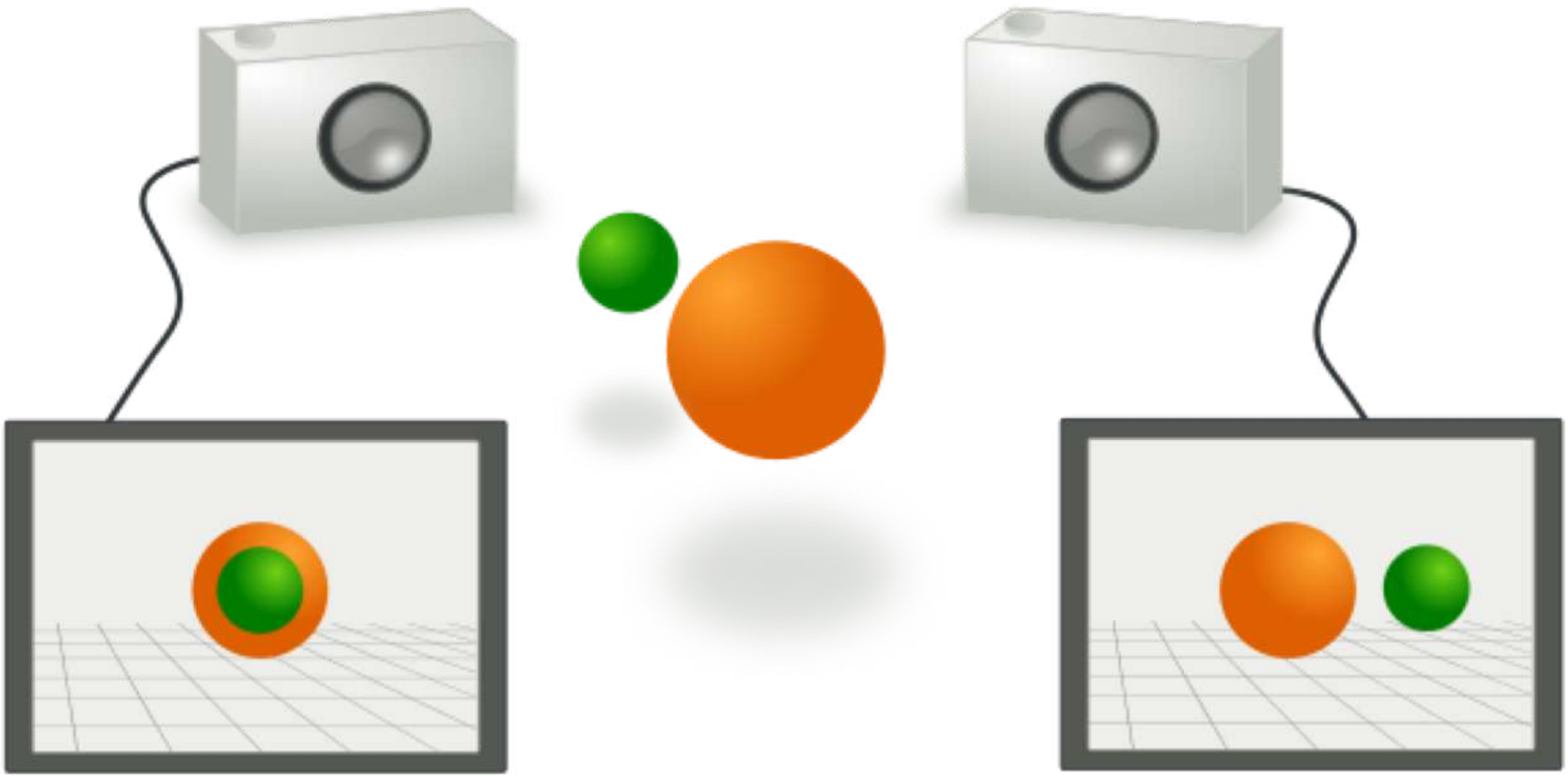


Ajout d'une 2ème caméra

- All points on projective line to P in left camera forment une ligne dans l'image de droite



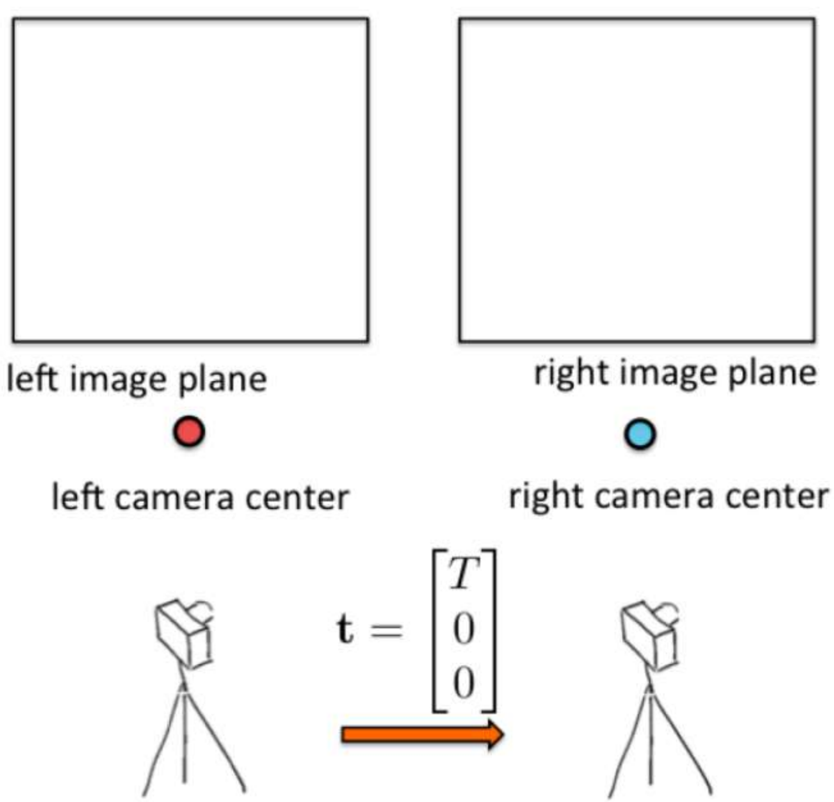
Géométrie épipolaire



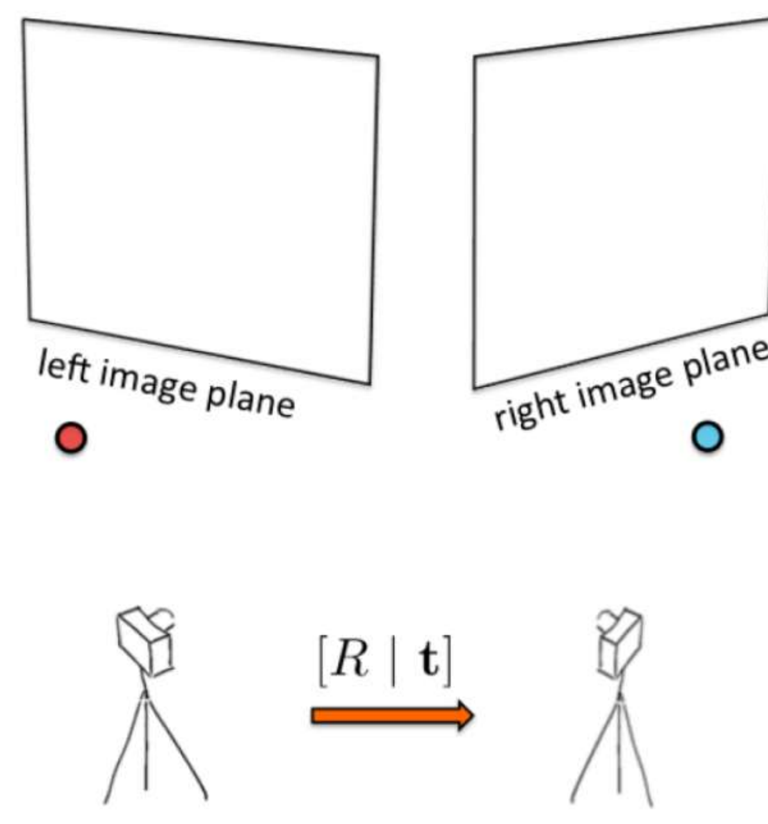
Par Arne Nordmann (norro) — Own illustration, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=3550774

Géométrie épipolaire

Parallel stereo cameras:

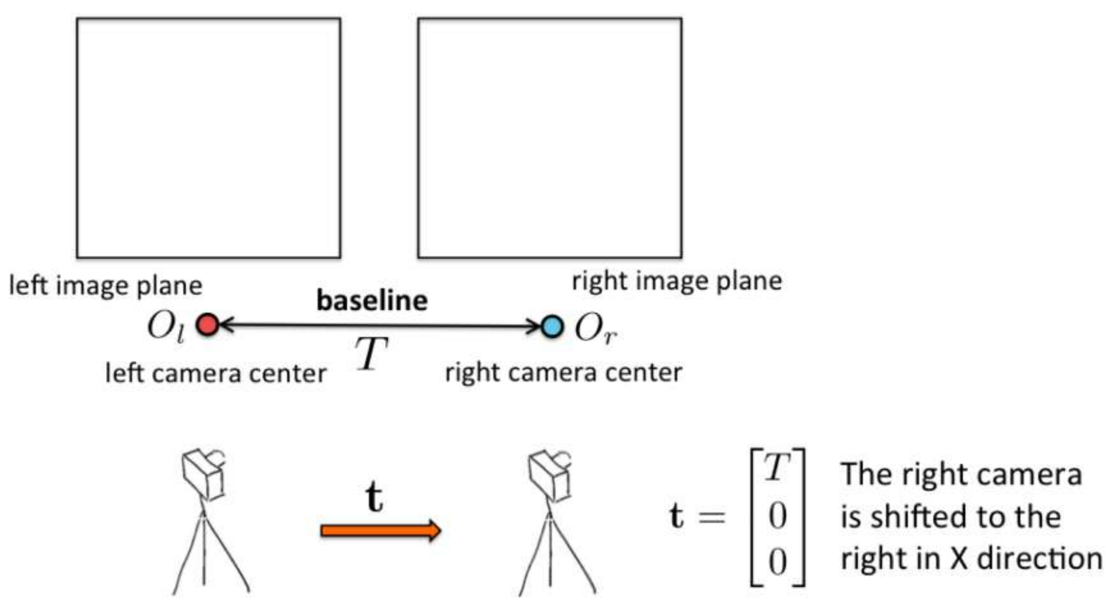


General stereo cameras:

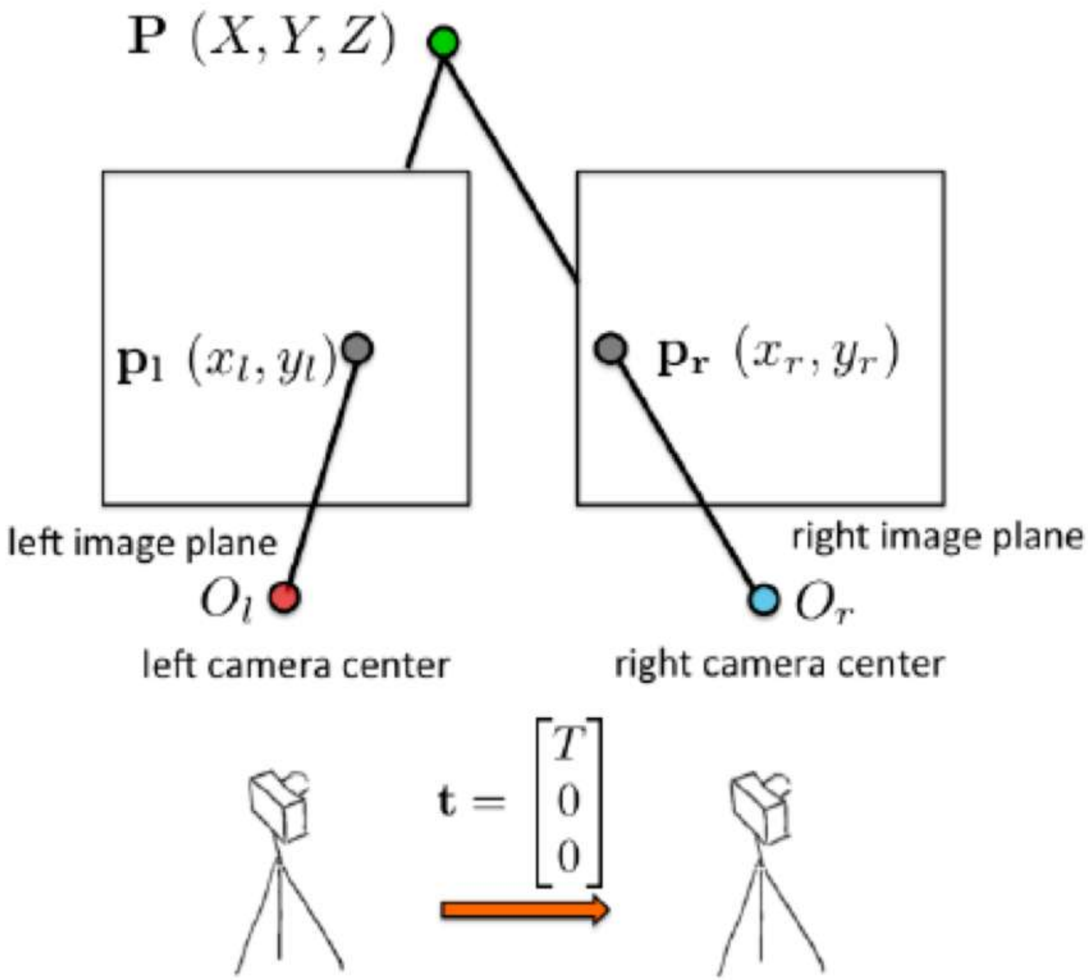


Cas simples :
2 caméras parallèles

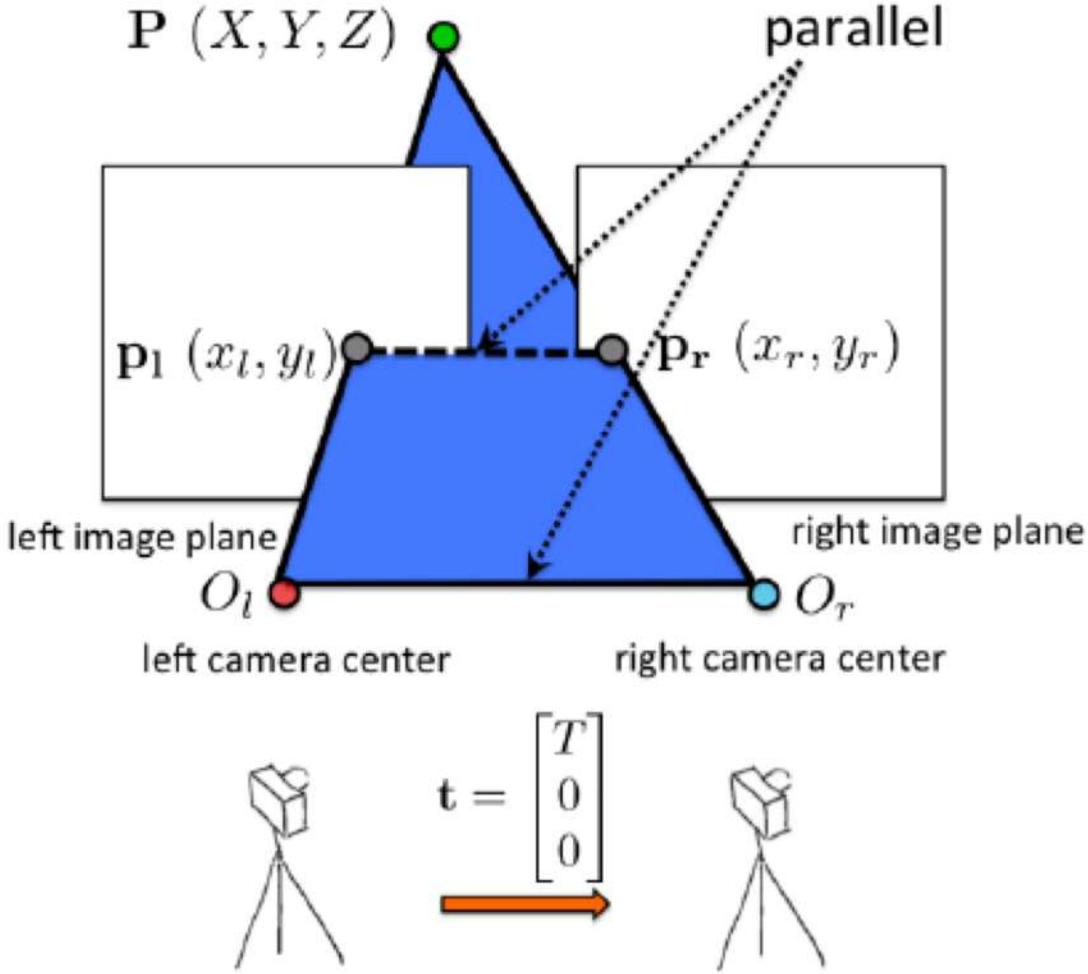
- Image planes of cameras are parallel to each other and to the baseline
- Camera centers are at same height
- Focal lengths are the same



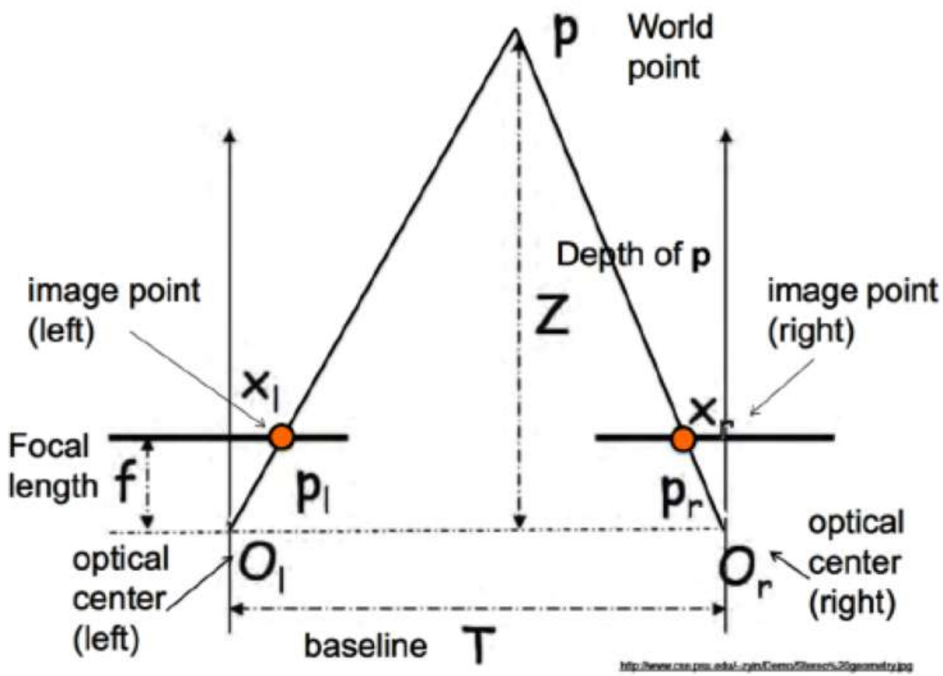
Estimation de la profondeur



Estimation de la profondeur



Estimation de la profondeur



$$Z = \frac{f \cdot T}{\boxed{x_l - x_r}}$$

baseline

focal length disparity

- Pour chaque point pl=(xl,yl), comment je peux trouver pr=(xr,yr) ?

left image

right image

left image

right image

the match will be on this line (same y)

(CAREFUL: this is only true for parallel cameras. Generally, line not horizontal)

On cherche ce point

x_l left image

right image x_l

the match will be **on the left** of x_l

how do I find it?

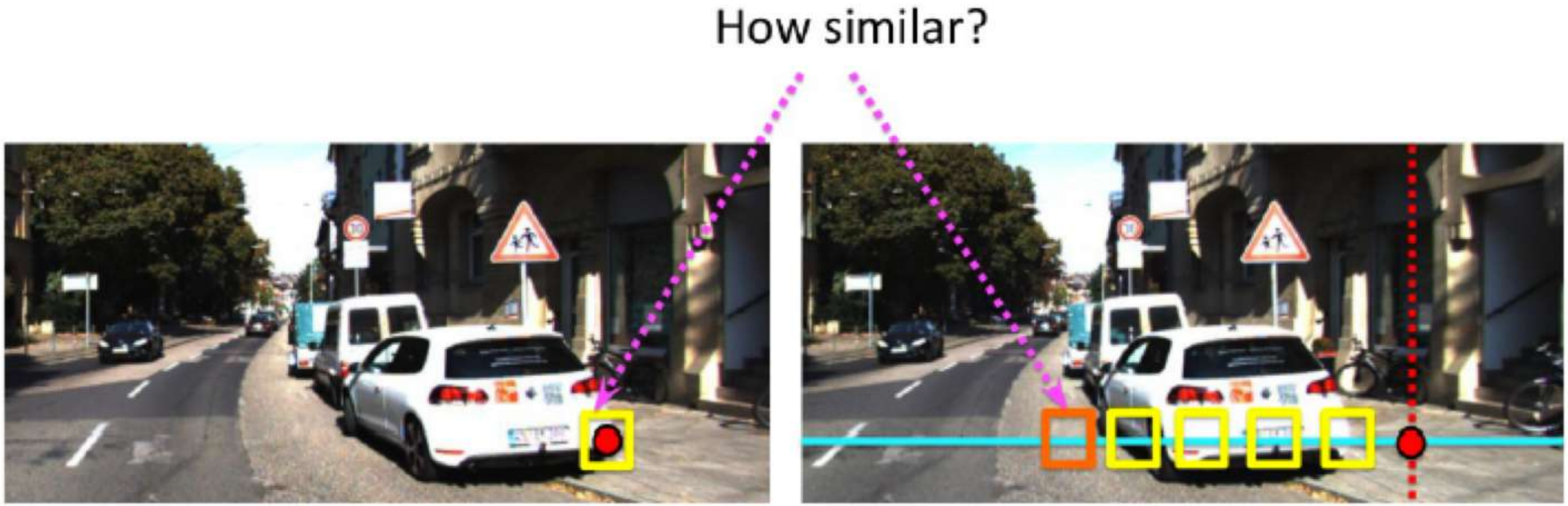
We call this line a **scanline**

left image

right image

we **scan** the line and **compare** patches to the one in the left image

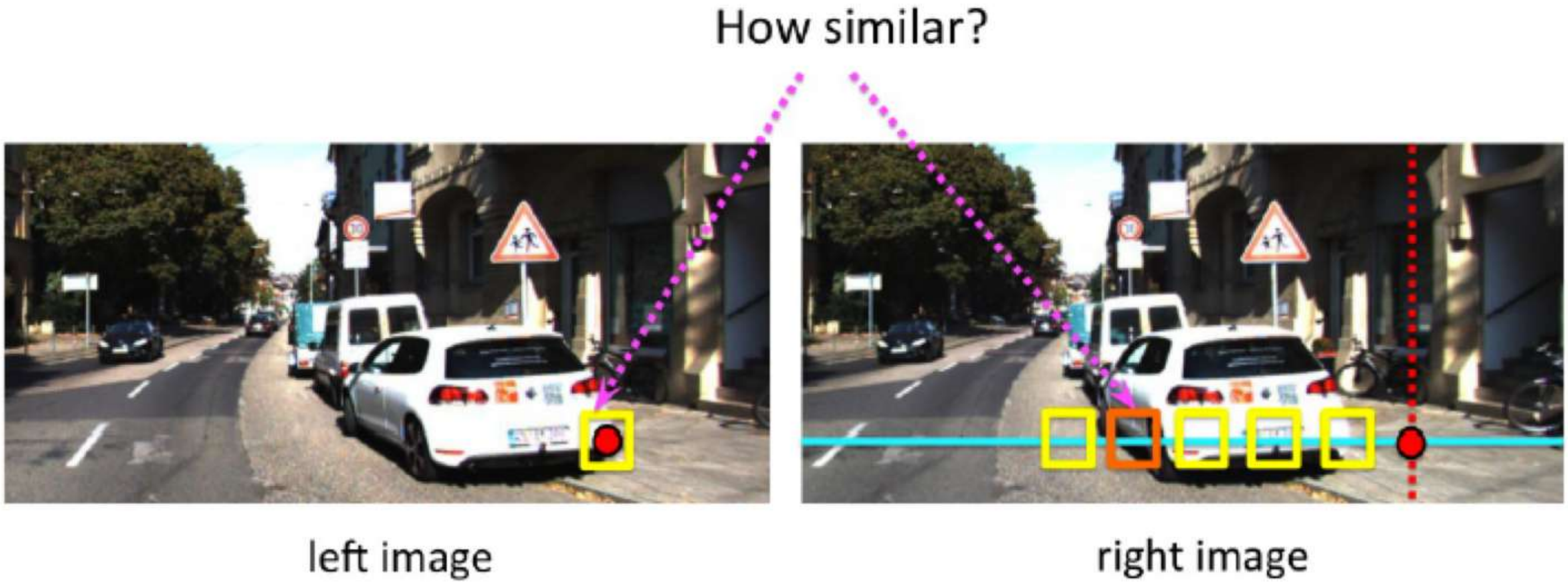
We are looking for a patch on scanline most similar to patch on the left



left image right image

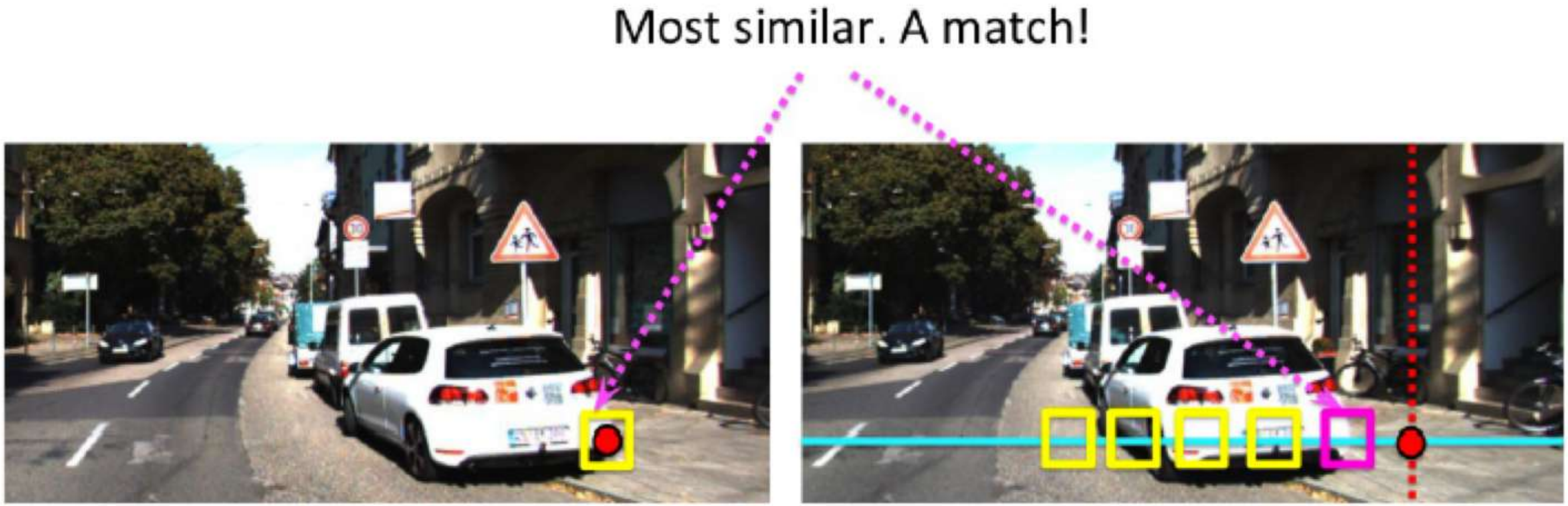
we **scan** the line and **compare** patches to the one in the left image

We are looking for a patch on scanline most similar to patch on the left



we **scan** the line and **compare** patches to the one in the left image

We are looking for a patch on scanline most similar to patch on the left



left image

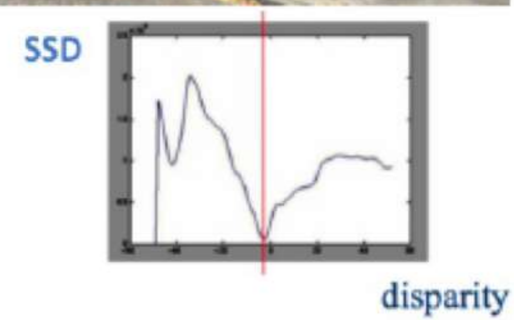
right image

we **scan** the line and **compare** patches to the one in the left image

We are looking for a patch on scanline most similar to patch on the left

$$SSD(\text{patch}_l, \text{patch}_r) = \sum_x \sum_y (I_{\text{patch}_l}(x,y) - I_{\text{patch}_r}(x,y))^2$$

left image



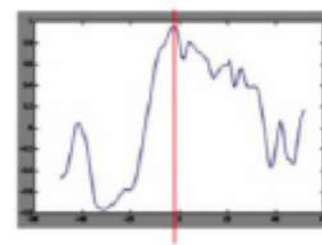
Compute a matching cost

Matching cost**: SSD (look for minima)**

$$NC(\text{patch}_l, \text{patch}_r) = \frac{\sum_x \sum_y (I_{\text{patch}_l}(x,y) \cdot I_{\text{patch}_r}(x,y))}{||I_{\text{patch}_l}|| \cdot ||I_{\text{patch}_r}||}$$

left image

Norm.
Corr.



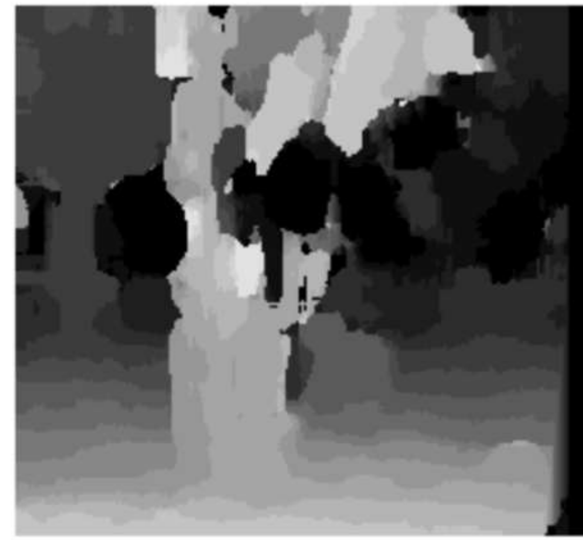
disparity

Compute a matching cost

Matching cost: **Normalized Corr. (look for maxima)**

Effet de la taille de la fenêtre

W = 3

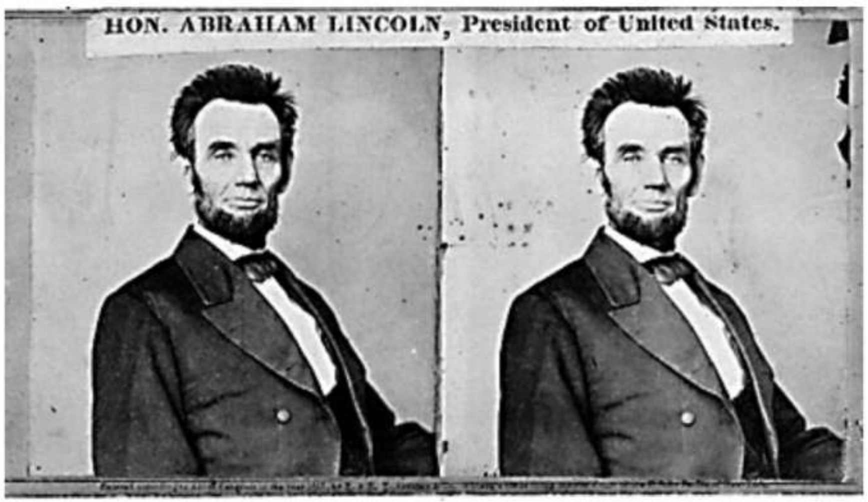


W = 20

– Smaller window
+ More detail
• More noise

– Larger window
+ Smoother disparity maps
• Less detail

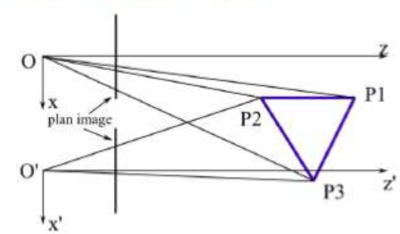
Contraintes pour trouver les similarités



Textureless surfaces

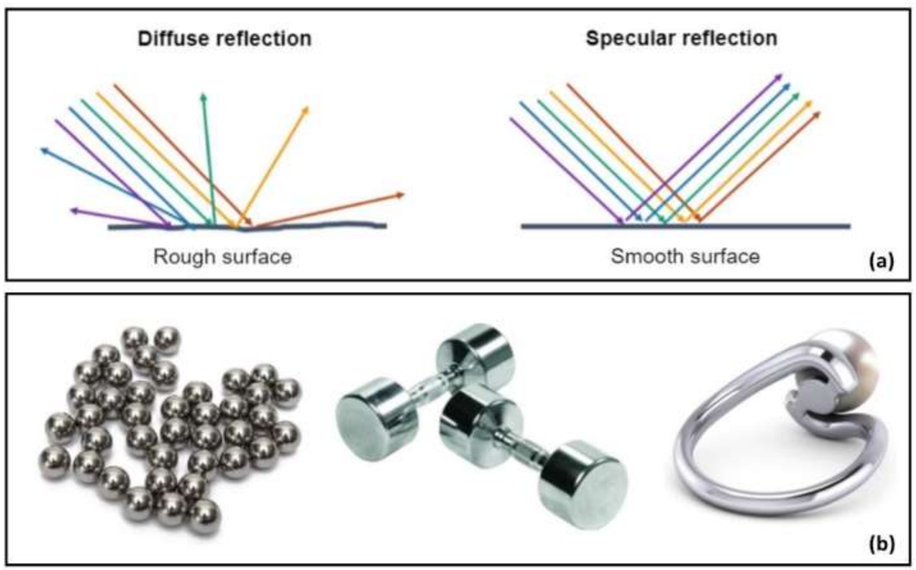
Occlusions, repetition

- Occultation : si (P1,P2,P3) est un objet opaque, le point P1 n'aura pas de projection dans la caméra droite.

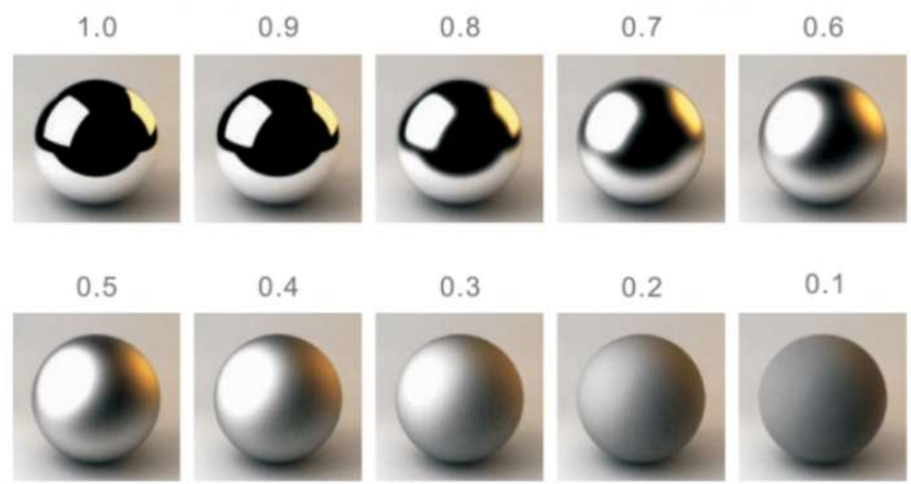


Contraintes de similarité

- surface spéculaire
- « Qui réfléchit la lumière comme un miroir »

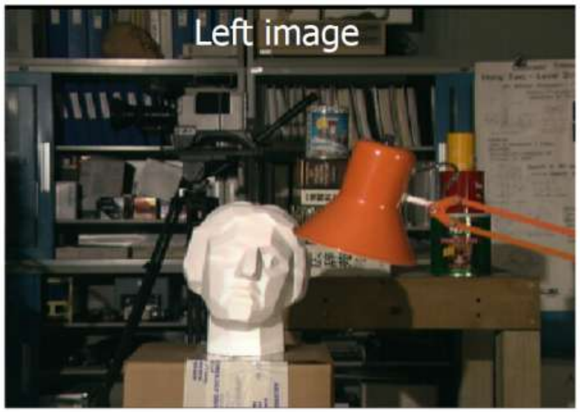


https://www.mdpi.com/2075-1702/11/1/91

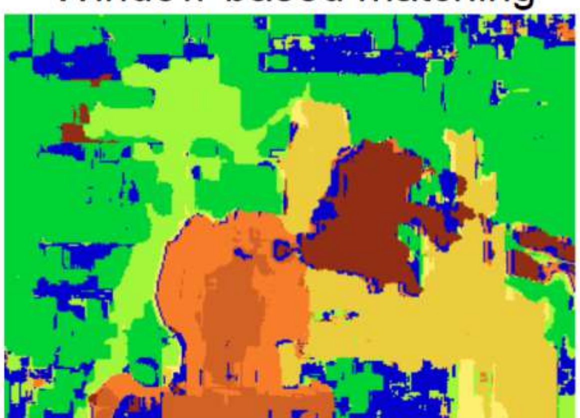


https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8960991

Résultats



Window-based matching



Ground truth

Résultats meilleurs avec autre méthode…

Graph cuts

Ground truth

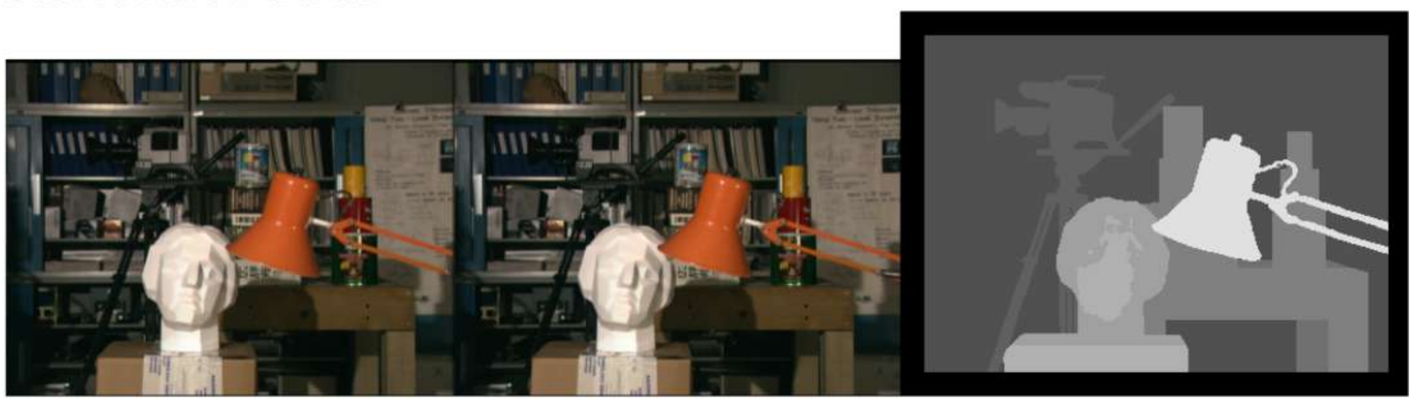
Y. Boykov, O. Veksler, and R. Zabih,
Fast Approximate Energy Minimization via Graph Cuts, PAMI 2001

Stéréovision dense vs éparse

Stéréovision dense

- \- Production d'une carte de profondeur, donnant Z pour tout point de l'image.
- \- Coût calculatoire élevé dû au volume de données à traiter, lié à la résolution.
- \- Nécessité d'un post-traitement pour l'analyse de scène : segmentation et/ou localisation (techniques classiques de vision par ordinateur).

► Stéréovision dense



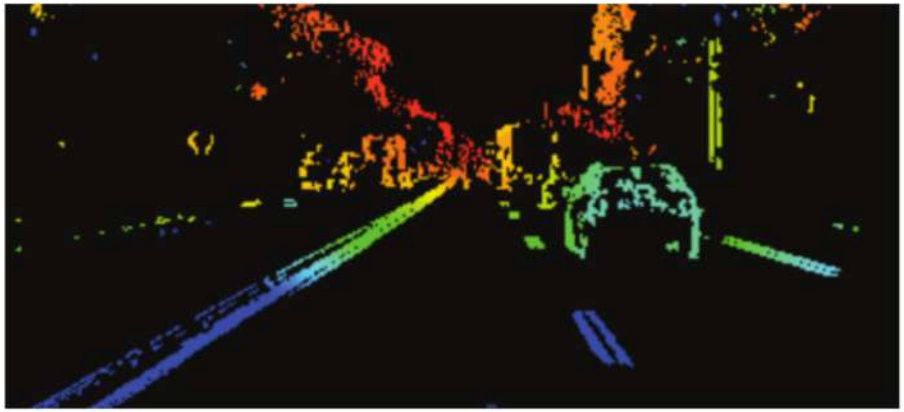
Stéréovision dense vs éparse

Stéréovision éparse

- \- La profondeur est calculée uniquement sur des points caractéristiques (angles, coins, lignes, obstacles, etc.).
- \- Volume de données à traiter réduit.
- \- Mais les performances sont liées à celles du détecteur.

→ Point clé dans les deux cas éparse ou dense : l'appariement !

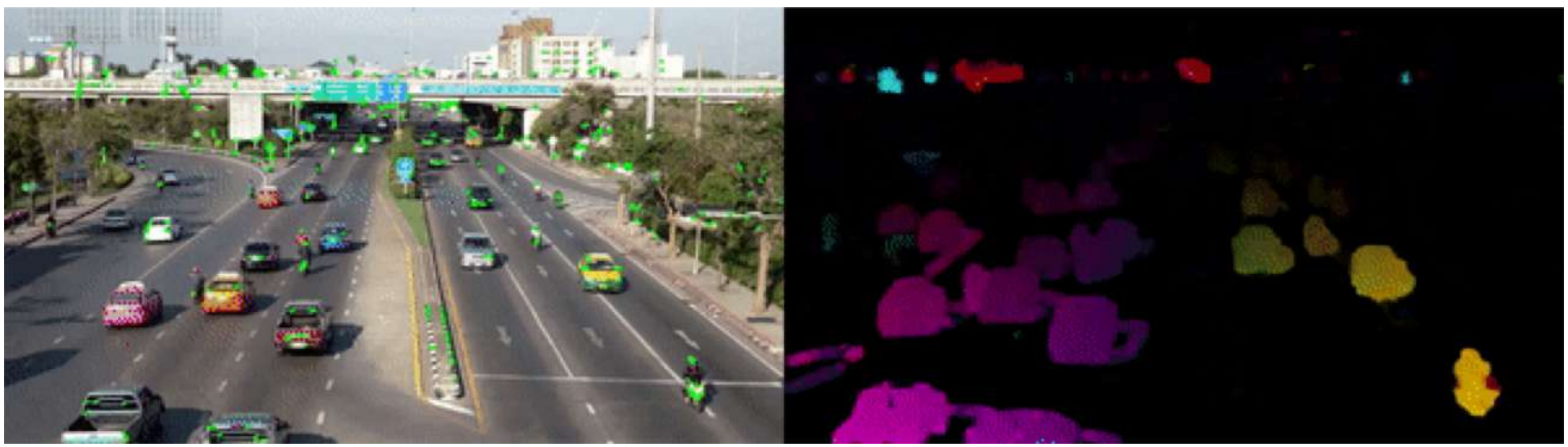
▸ Stéréovision éparse



source : JP Tarel, LIVIC

Rappel : flot optique

2 visualisations : flèches ou couleur HSV



https://www.sefidian.com/2019/12/16/a-tutorial-on-motion-estimation-with-optical-flow-with-python-implementation/

KITTI dataset

Welcome to the KITTI Vision Benchmark Suite!

We take advantage of our autonomous driving platform Annieway to develop novel challenging real-world computer vision benchmarks. Our tasks of interest are: stereo, optical flow, visual odometry, 3D object detection and 3D tracking. For this purpose, we equipped a standard station wagon with two high-resolution color and grayscale video cameras. Accurate ground truth is provided by a Velodyne laser scanner and a GPS localization system. Our datsets are captured by driving around the mid-size city of Karlsruhe, in rural areas and on highways. Up to 15 cars and 30 pedestrians are visible per image. Besides providing all data in raw format, we extract benchmarks for each task. For each of our benchmarks, we also provide an evaluation metric and this evaluation website. Preliminary experiments show that methods ranking high on established benchmarks such as Middlebury perform below average when being moved outside the laboratory to the real world. Our goal is to reduce this bias and complement existing benchmarks by providing real-world benchmarks with novel difficulties to the community.

Share

https://www.cvlibs.net/datasets/kitti/

Estimation de la profondeur monoculaire

- https://ipolcore.ipol.im/demo/clientApp/demo.html?id=45901&key=BBD9A73D64B80F908982FF902F9CEA16

Sources

- Cours Fei-Fei Li, Stanford vision lab, lecture9-10 stereo cs131
- CSE486, Penn State USA Robert Collins, Lecture 08: Introduction to Stereo
- Sanja Fidler CSC420: Intro to Image Understanding, Univ of Toronto Canada
- Stéréovision : Généralités & géométrie épipolaire Systèmes de vision, INSA - ASI4 Sebastien Kramm