

# Hands-on AI based 3D Vision

## Summer Semester 25

Lecture 4\_0 – From Classical to Modern Stereo Vision and Depth Estimation

Prof. Dr.-Ing Gerard Pons-Moll  
University of Tübingen / MPI-Informatics

EBERHARD KARLS  
UNIVERSITÄT  
TÜBINGEN



# Overview

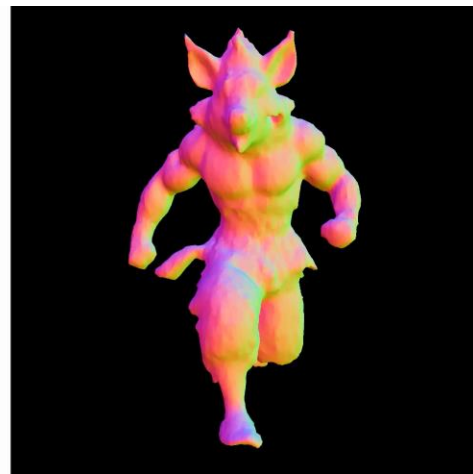
- Intro
- Recap of Epipolar Geometry
- Stereo Matching and Depth Estimation
- Multiview Stereo Matching
- Monocular Depth Estimation

# Recovering 3D from an Image

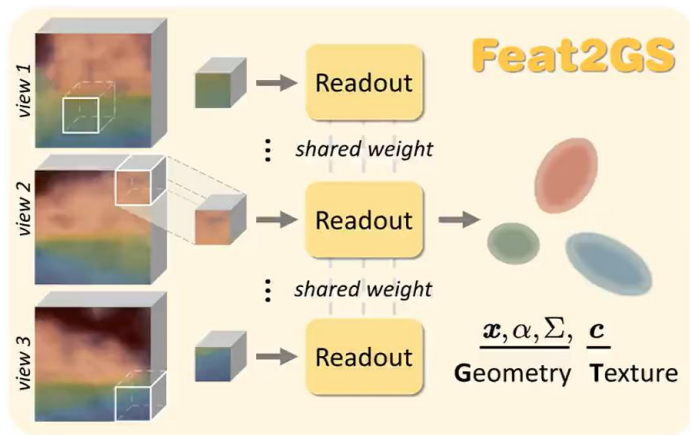
One of the main goals of 3D Vision is the reconstruction of 3D from images



Help artists  
to **speed-up**  
their  
workflow




# Democratize 3D asset creation



Splatting



*"Probing Schemes"*

Probing	-Geometry	-Texture	-All
Feature- Readout	$\underline{x}, \alpha, \Sigma$	$\underline{c}$	$\underline{x}, \alpha, \Sigma, \underline{c}$
Free-Optimize 	$\underline{c}$	$\underline{x}, \alpha, \Sigma$	/

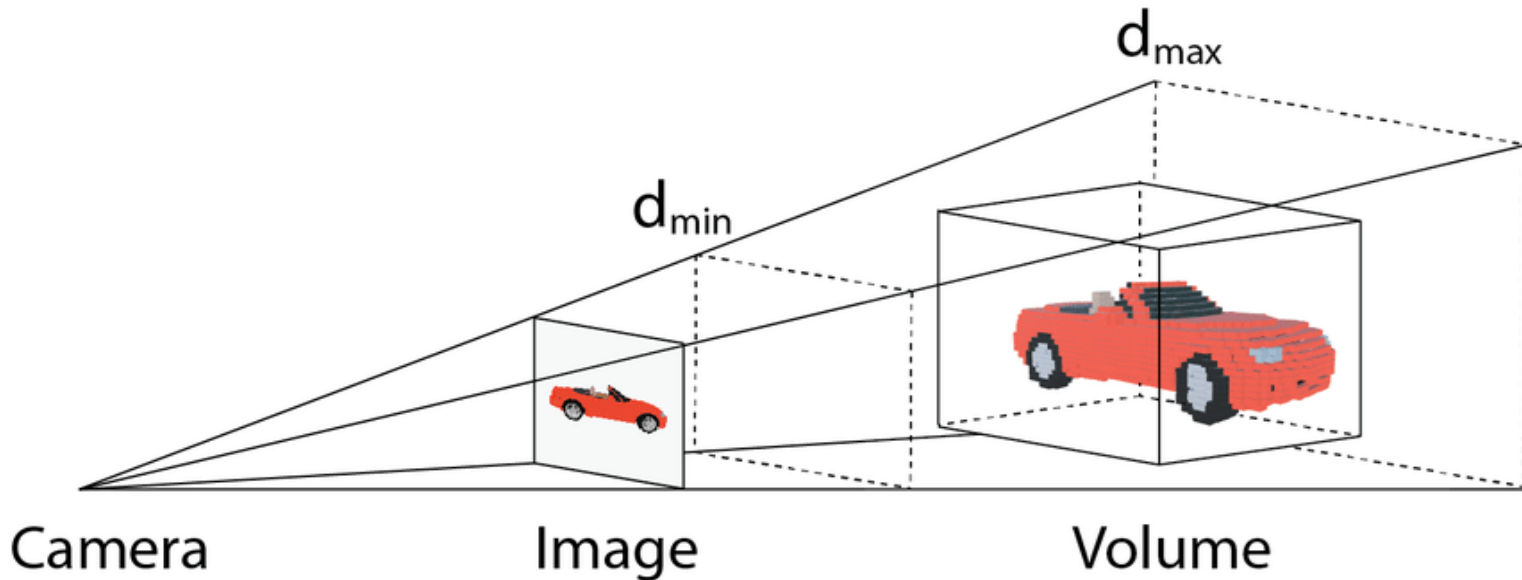
Novel View Synthesis

Feat2GS, Chen et al. CVPR 2025

# Recovering 3D from an Image

However 3D from an image is an **ill-posed** problem

During image formation, we **lose depth** information



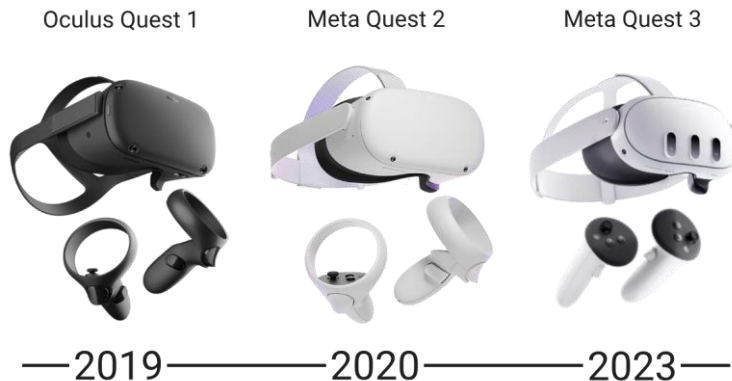
# Recovering 3D from an Image

Luckily, we have more than one point of view on the world that surrounds us.

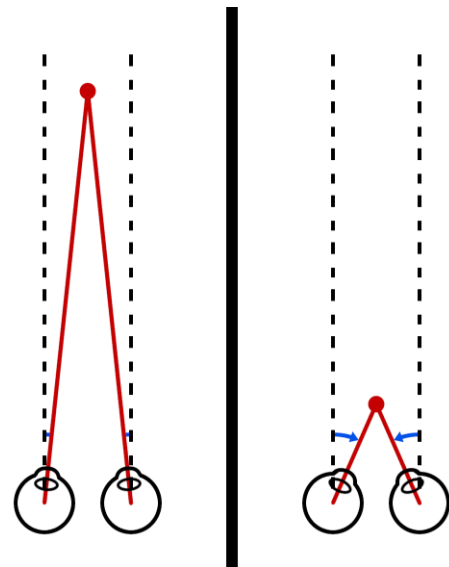
We can infer depth from our **two-eye system**



Wooden stereo viewing device, late 19th century.  
Swiss National Museum



Oculus/Meta Quest visors.



# Recovering 3D from an Image

We would like to replicate the same behaviour, for computers

Left  
image



Right  
image



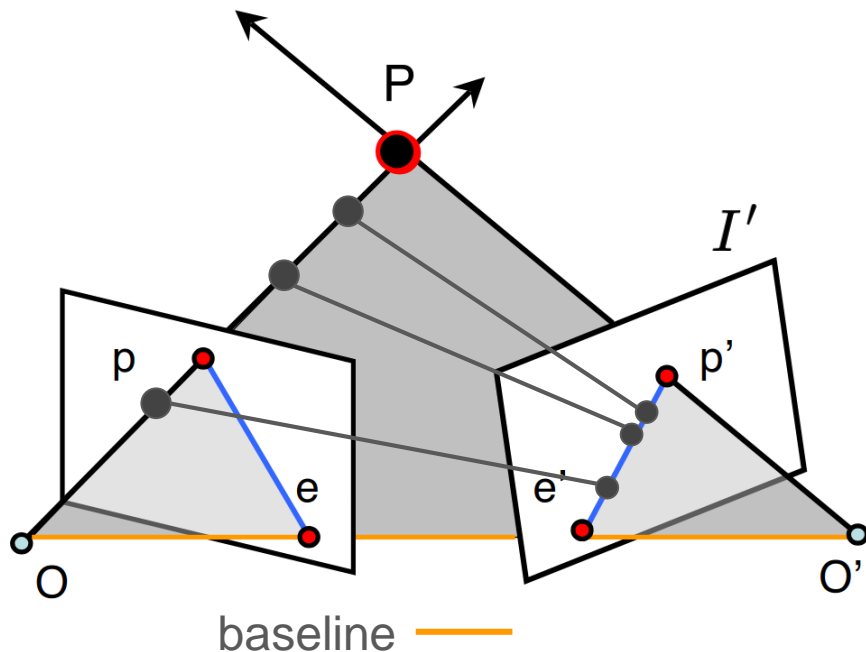
Depth image



# Epipolar Geometry

Recalls of epipolar geometry:

- Cameras with **center**  $O$  and  $O'$   
The line that connects them: **baseline**
- The intersections between the image planes and the baseline are called **epipoles** ( $e$ ,  $e'$ )
- If a point  $p$  is observed in image  $I$ , the corresponding point  $p'$  must lie along the corresponding **epipolar line** in image  $I'$



# Epipolar Geometry

Mapping points of image  $I$  to lines of image  $I'$  for canonical cameras:

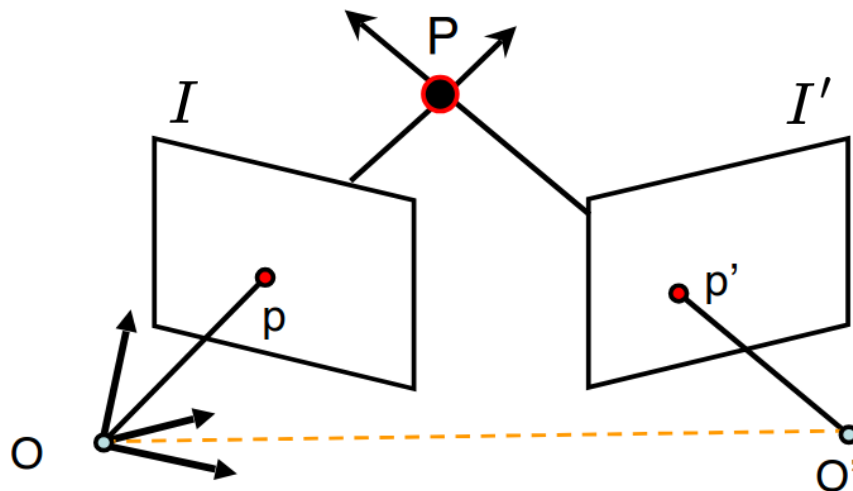
- Essential Matrix  $E$

$$p^T E p' = 0$$

$$E = [T_{\times}] R$$

$[T_{\times}]$  is the matrix representation of the cross product with translation (skew)

$R$  is the Rotation Matrix



# Epipolar Geometry

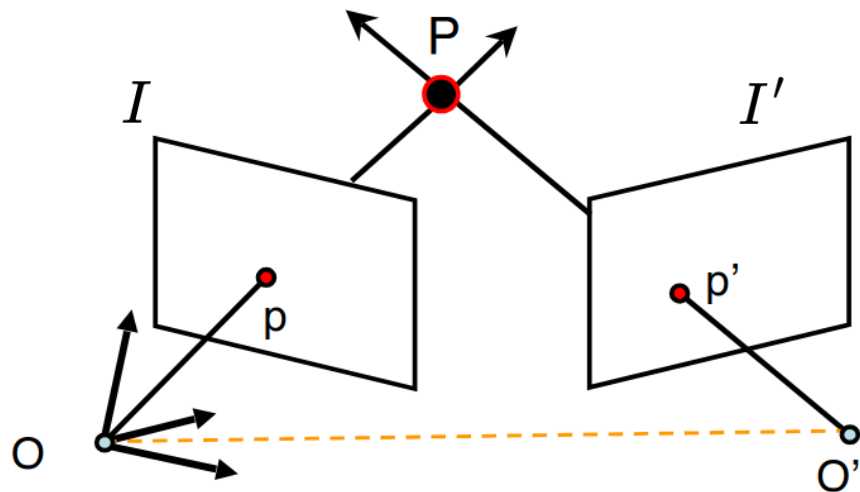
Mapping points of image  $I$  to lines of image  $I'$ :

- Fundamental Matrix  $F$

$$p^T F p' = 0$$

$$F = K^{-T} \cdot E K'^{-1}$$

with intrinsic camera parameters  $K$   
and  $K'$  respectively

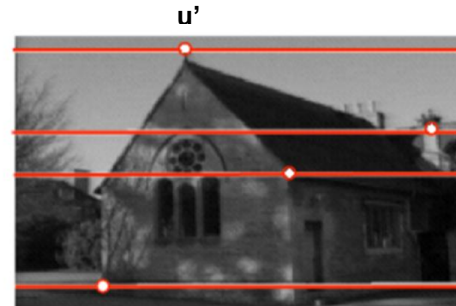
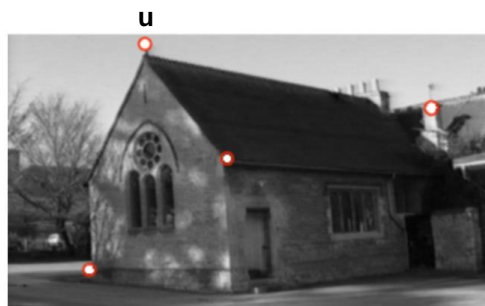
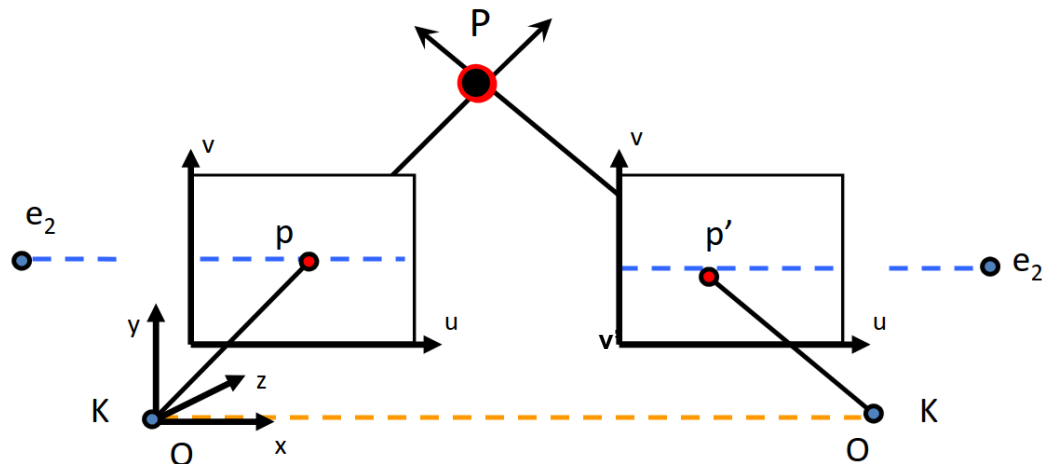


# Epipolar Geometry

If we have **parallel image planes**:

- **Epipolar Lines** are **horizontal** and **Epipoles** go to **infinity**
- **y-coordinates** are **equal**

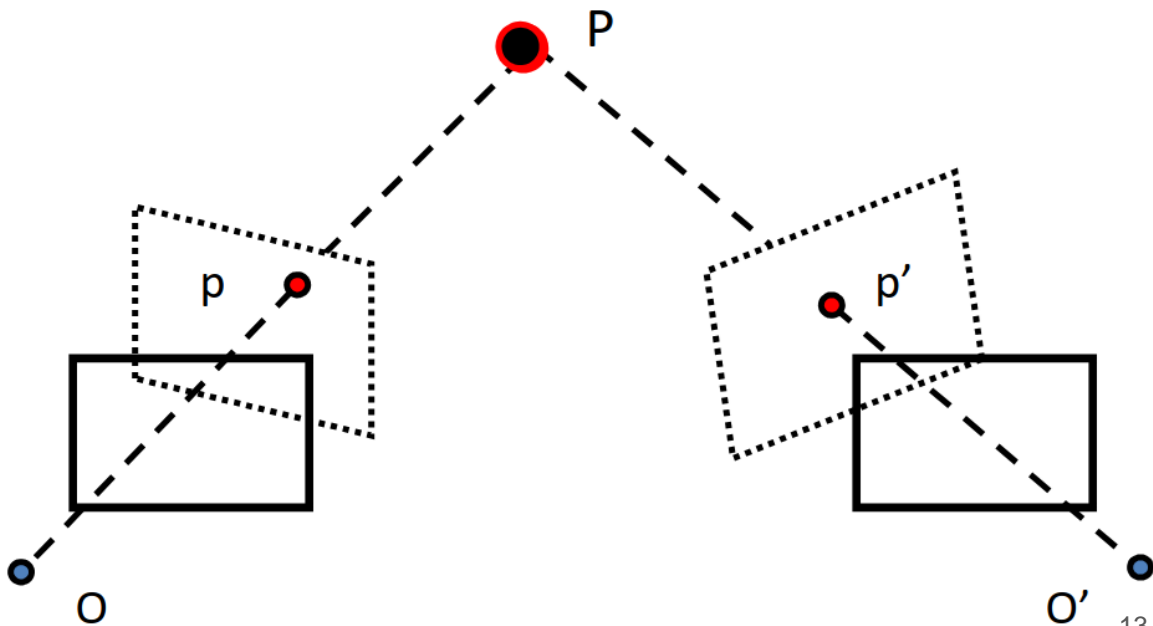
$$p = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \quad p' = \begin{bmatrix} u' \\ v \\ 1 \end{bmatrix}$$



# Image Rectification

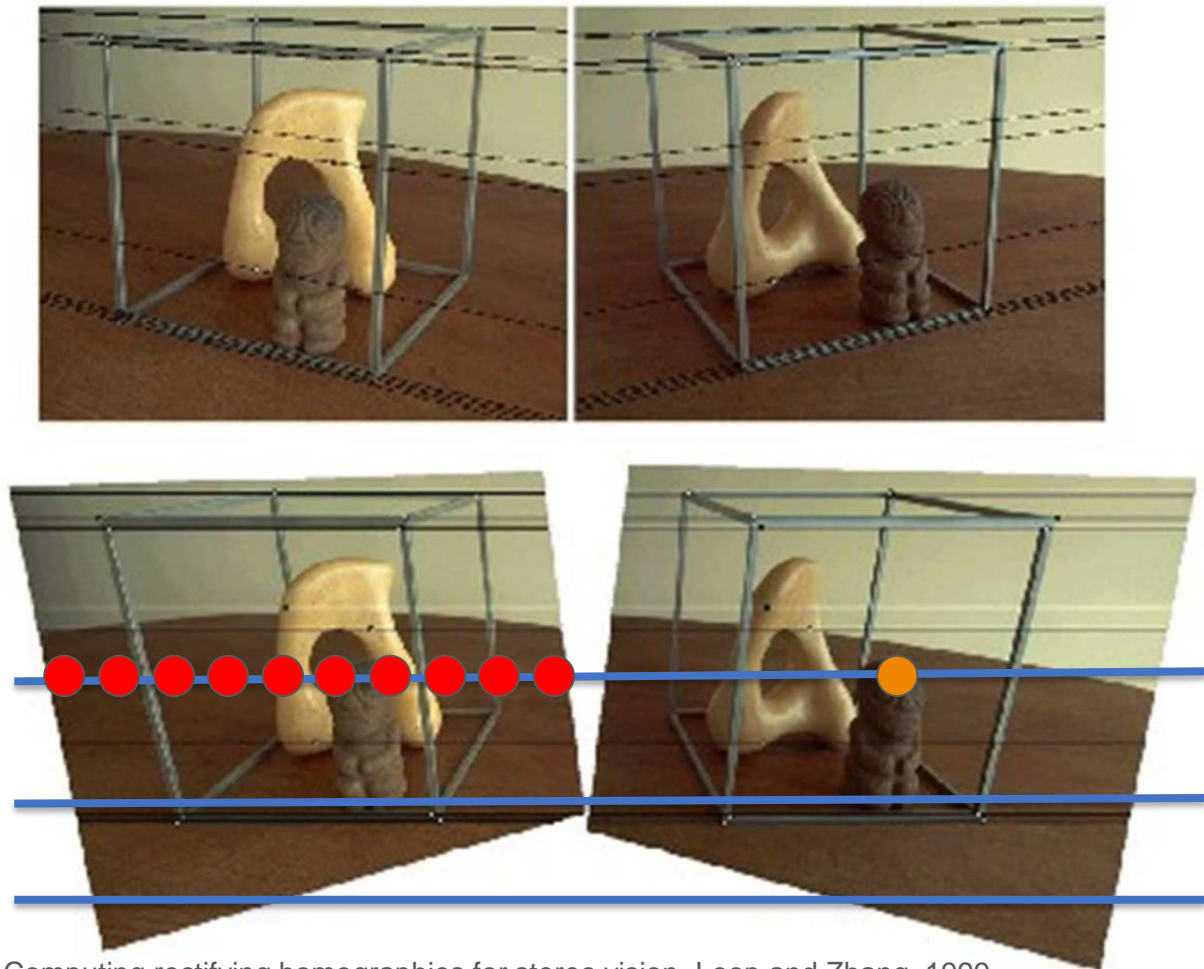
The problem of warping camera planes to make them parallel is called **rectification**

Existing algorithms solve rectification by reprojecting image planes onto a common plane parallel to the line between optical centers



# Image Rectification

A point in the right image  
corresponds to a set of  
candidate points along the  
same row in the left image



# Image Rectification

After rectification, finding the essential matrix  $E$  becomes trivial.

$$p^T E p' = 0 \qquad E = [T_{\times}] R$$

Image planes only differ by a translation:  $R = I \qquad t = (T, 0, 0)$

$$E = [t_{\times}] R = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -T \\ 0 & T & 0 \end{bmatrix}$$

# Stereo Matching

# Stereo Matching

Given a pair of rectified stereo images, the goal of **Stereo Matching** is to compute the **disparity** for each pixel in the reference image, where disparity is defined as the **horizontal displacement** between a pair of corresponding pixels in the left and right images.

# Disparity

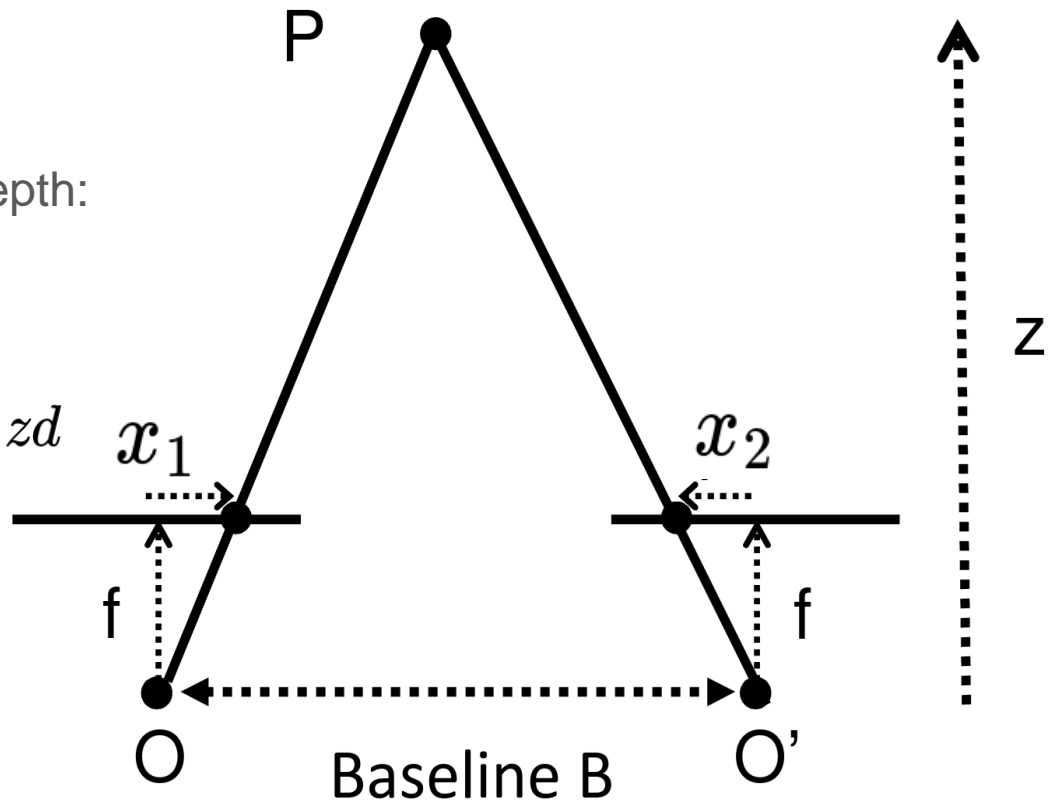
Disparity can be used to calculate depth:

disparity:  $d = (x_1 - x_2)$

Triangle Proportionality Theorem:

$$\frac{z - f}{b - d} = \frac{z}{b} \Rightarrow zb - fb = zb - zd$$
$$\Rightarrow z = \frac{fb}{d}$$

This reduces depth estimation to finding **matches** between points of two rectified images to calculate the **disparity**



# Finding Matches between Images



Left Image



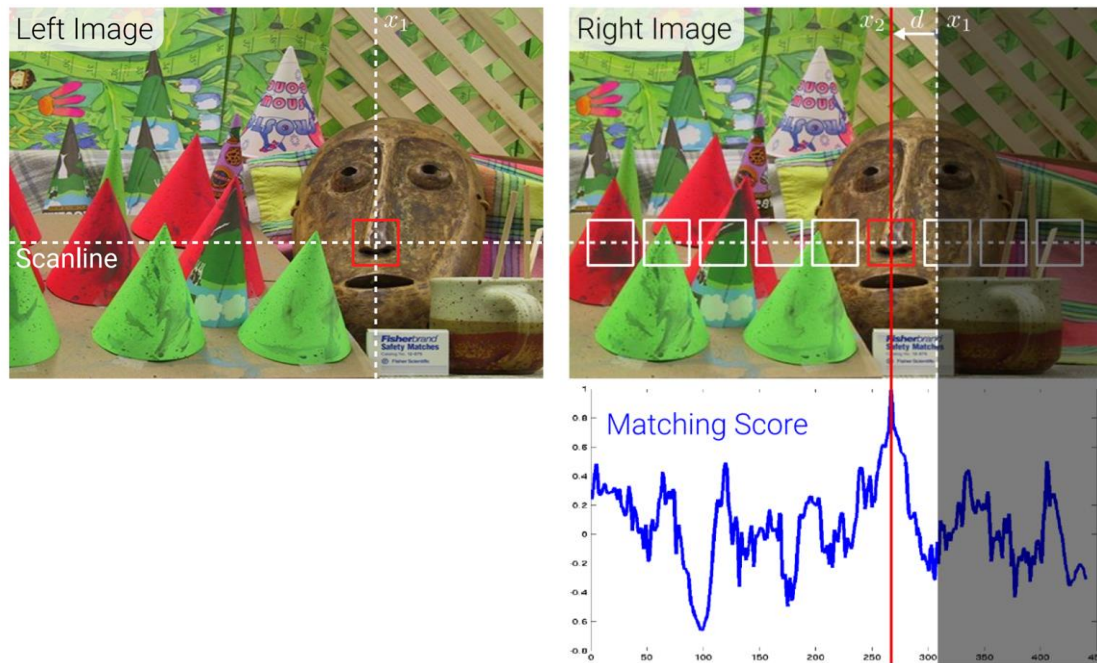
Right Image

High-accuracy stereo depth maps using structured light,. Scharstein et al. 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition

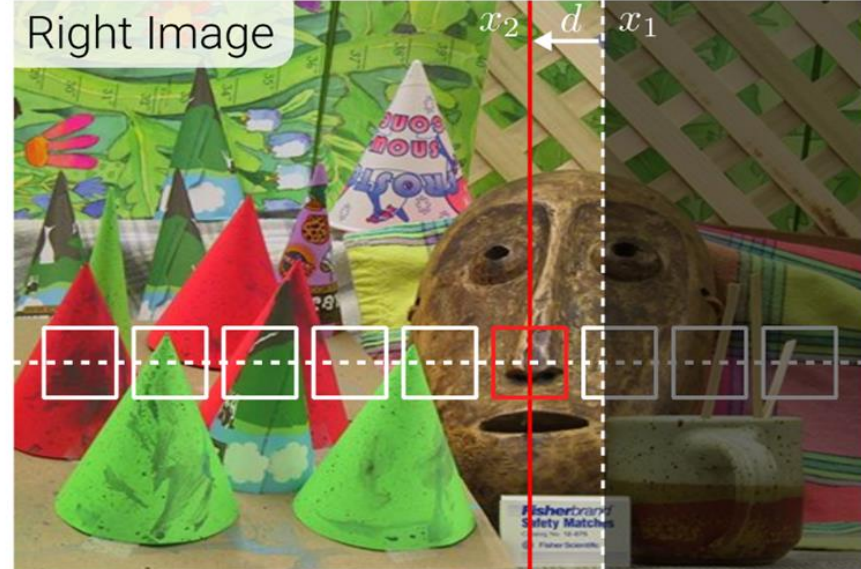
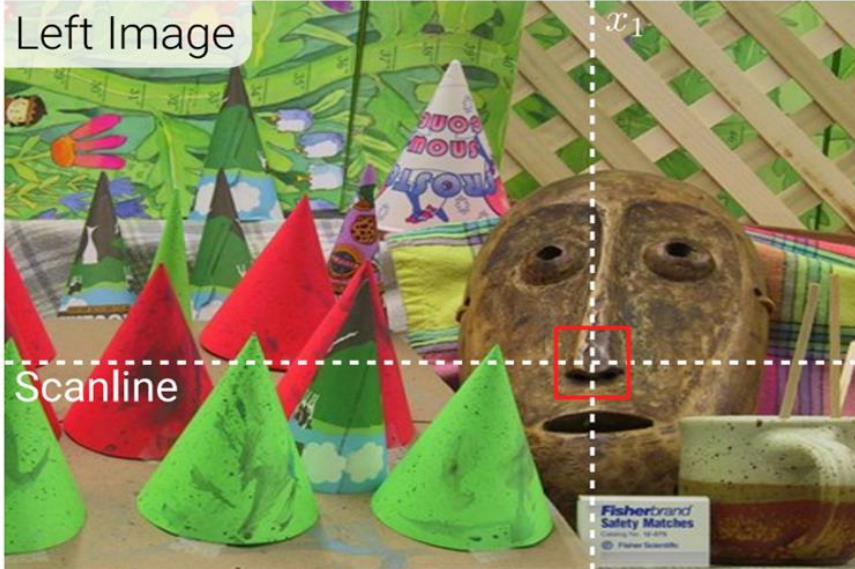
# Stereo Matching and Similarity Metrics

## Similarity Metric Approaches

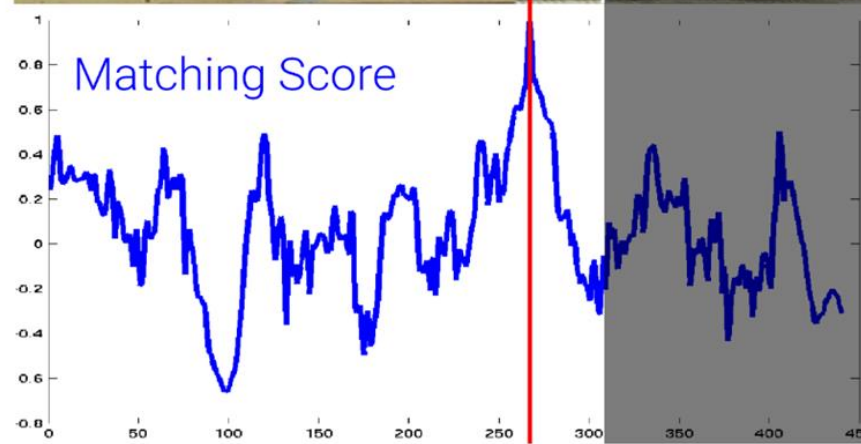
- Consist of **sliding** a **window** along the epipolar lines and compute a **cost** through a **Matching/Score Function**



Evaluation of Stereo Matching Costs on Images with Radiometric Differences.  
Hirschmuller and Scharstein, TPAMI, 2009.



- 1) Choose a **patch** on the left image (red square on the left image)
- 2) Look for a **corresponding patch** on the right image, along the **epipolar line** (or scanline)
  1. Use a **sliding window** (squares on the right image) on the right image, and evaluating the score function.
- 3) Then, select the patch with the **highest score** as a match (red square on the right image)



# Stereo Matching

Many similarity metrics were proposed:

- Sum of Absolute Differences (SAD)

$$\text{SAD} = \sum |I_L(x, y) - I_R(x - d, y)|$$

- Sum of Squared Differences (SSD)

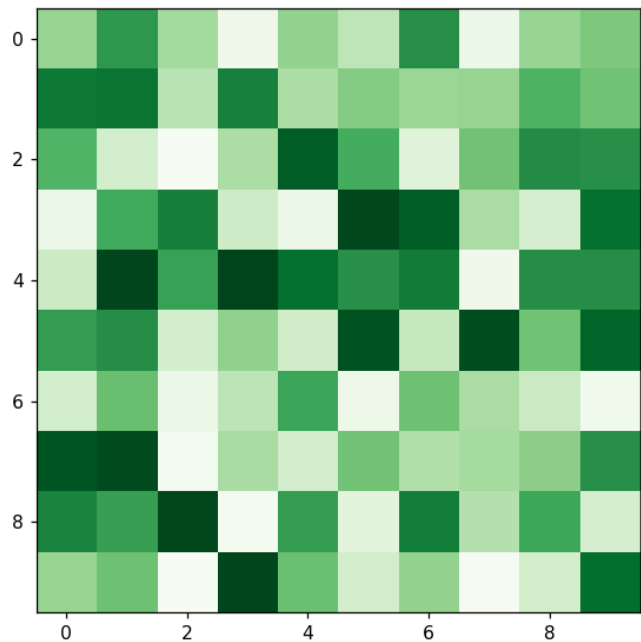
$$\text{SSD} = \sum (I_L(x, y) - I_R(x - d, y))^2$$

- Normalized Cross-Correlation (NCC)

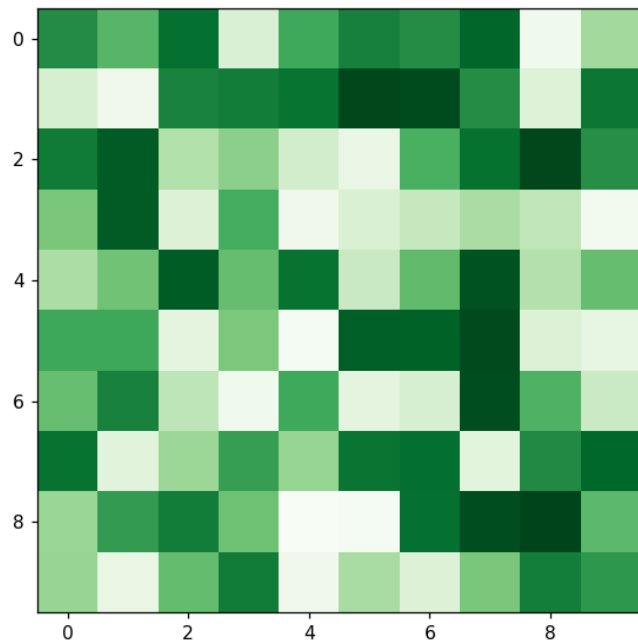
$$\text{NCC} = \frac{\sum (I_L - \bar{I}_L)(I_R - \bar{I}_R)}{\sqrt{\sum (I_L - \bar{I}_L)^2 \sum (I_R - \bar{I}_R)^2}}$$

# Cost Volume

Let's visualize the process:



Left Image

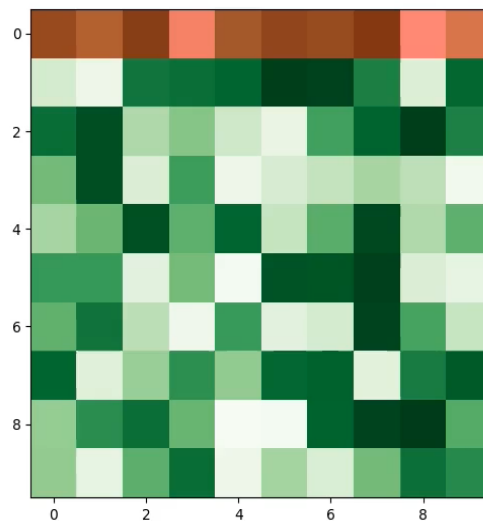
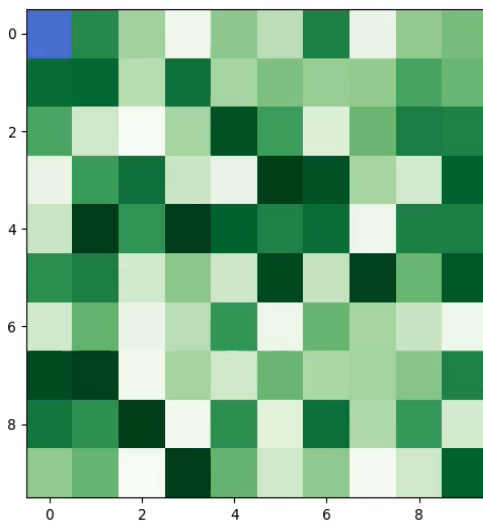
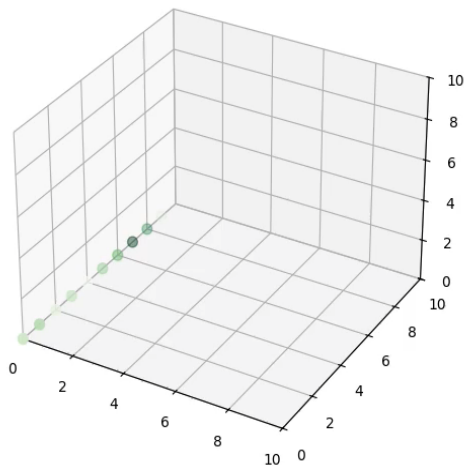


Right Image

# Cost Volume

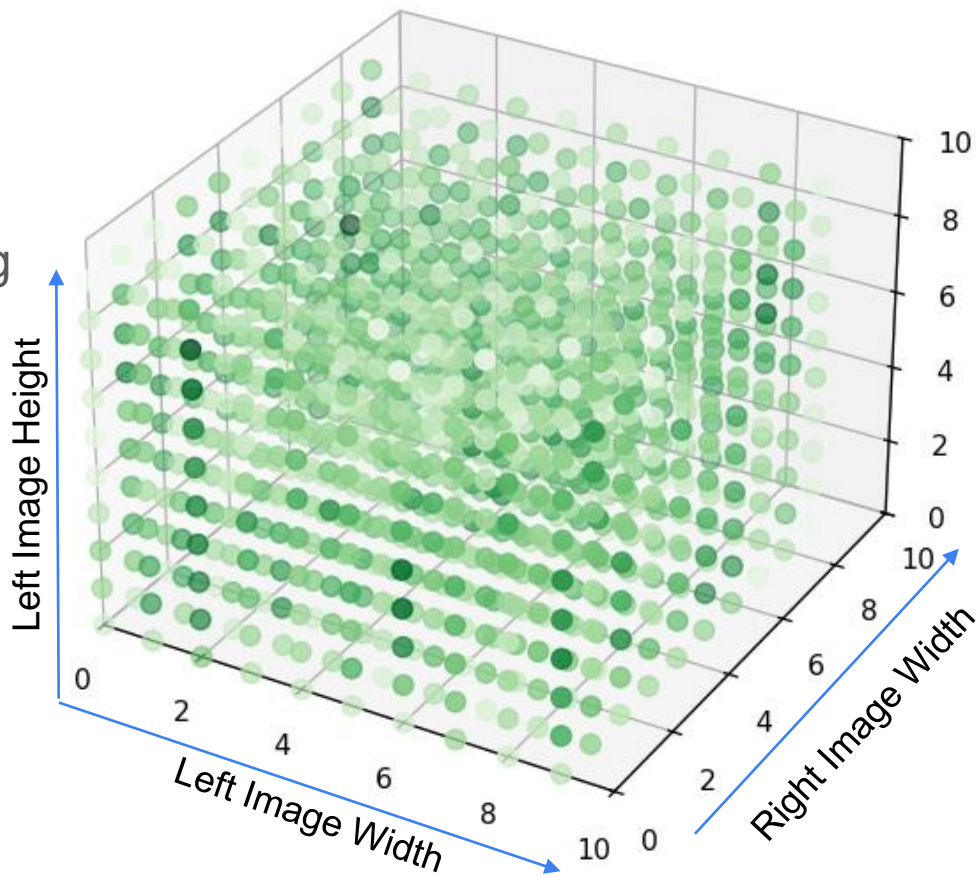
Let's visualize the process:

Choose a metric, and use the sliding window to compute the cost.



# Cost Volume

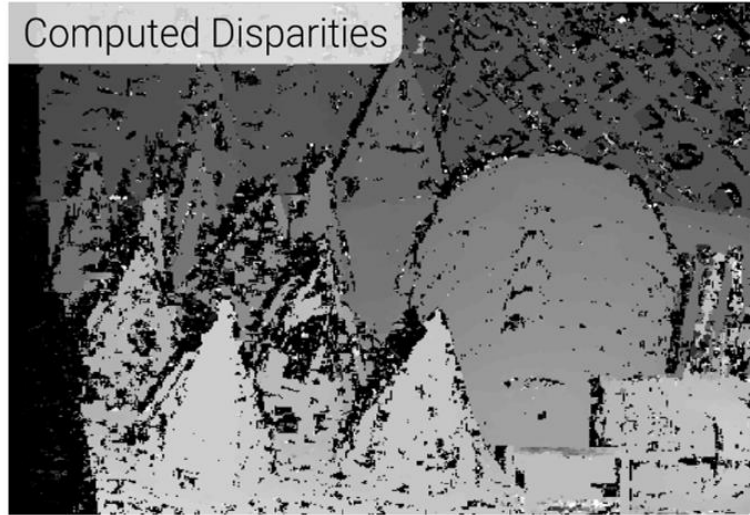
Let's visualize the process:  
Choose a metric, and use the sliding window to compute the cost.  
The resulting 3D volume is called **cost volume**, and contains the result of running the cost function between each patch on the left image, and each patch on the corresponding epipolar line of the right image



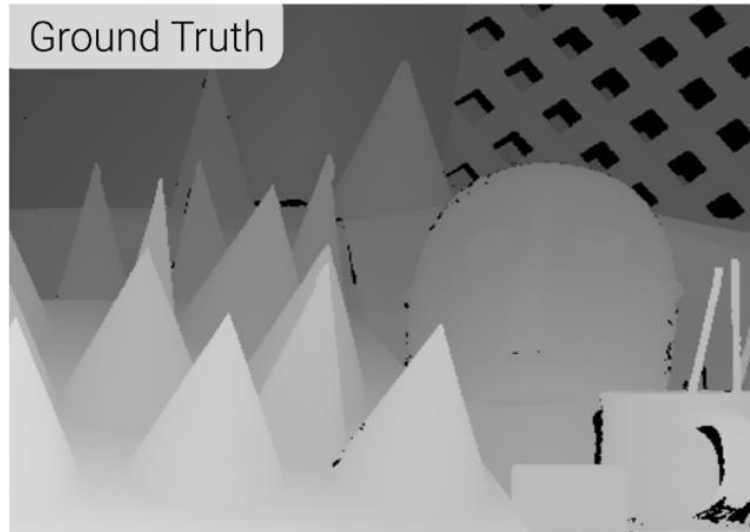
Left Image



Computed Disparities



Ground Truth



Evaluation of Stereo Matching Costs on Images with Radiometric Differences. Hirschmuller and Scharstein, TPAMI, 2009.

# Limitations

## Limitations – Repeated Patterns



## Limitations – Window Size



$W = 3$



$W = 20$

## Limitations – Textureless Surfaces

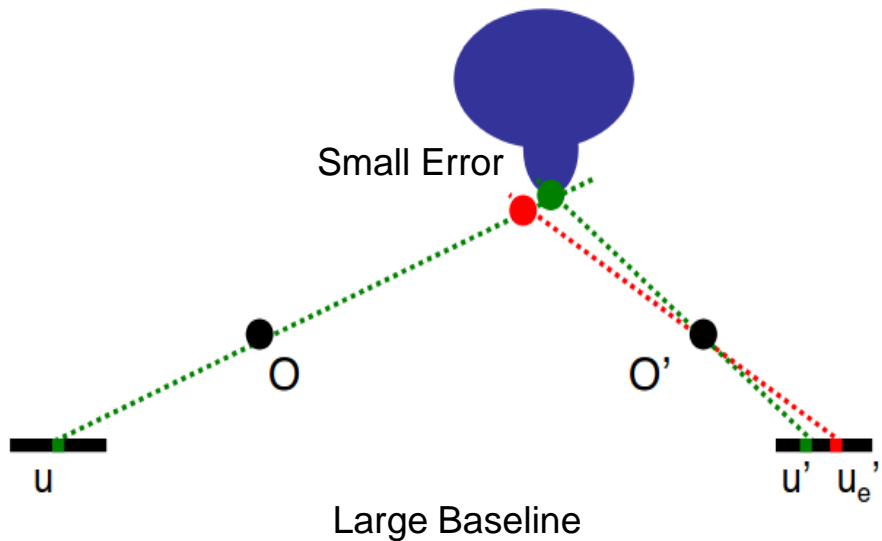


**Mismatch**

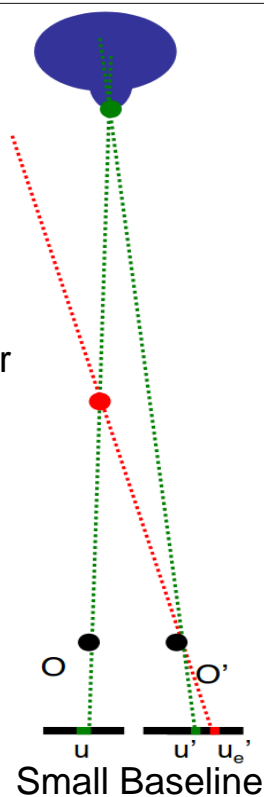
# Limitations – Illumination Change



# Limitations – Foreshortening

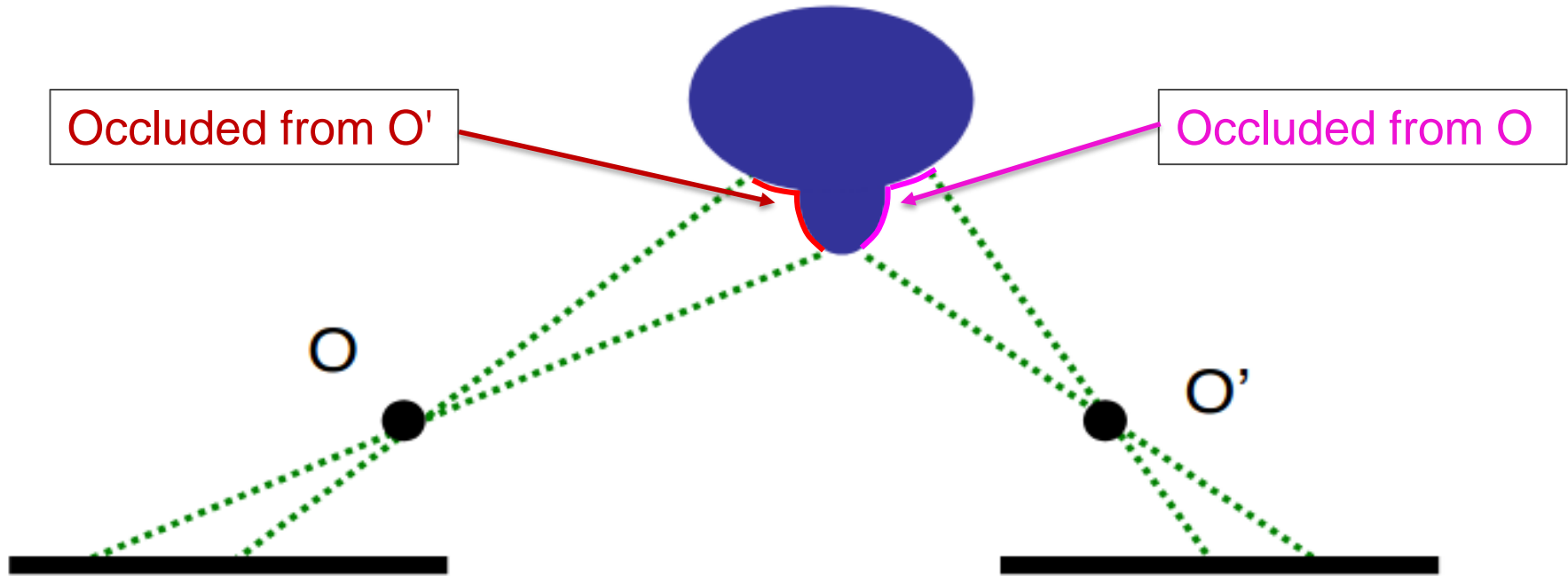


Large Error



With a short baseline, small errors in estimating disparities result in larger depth estimation errors

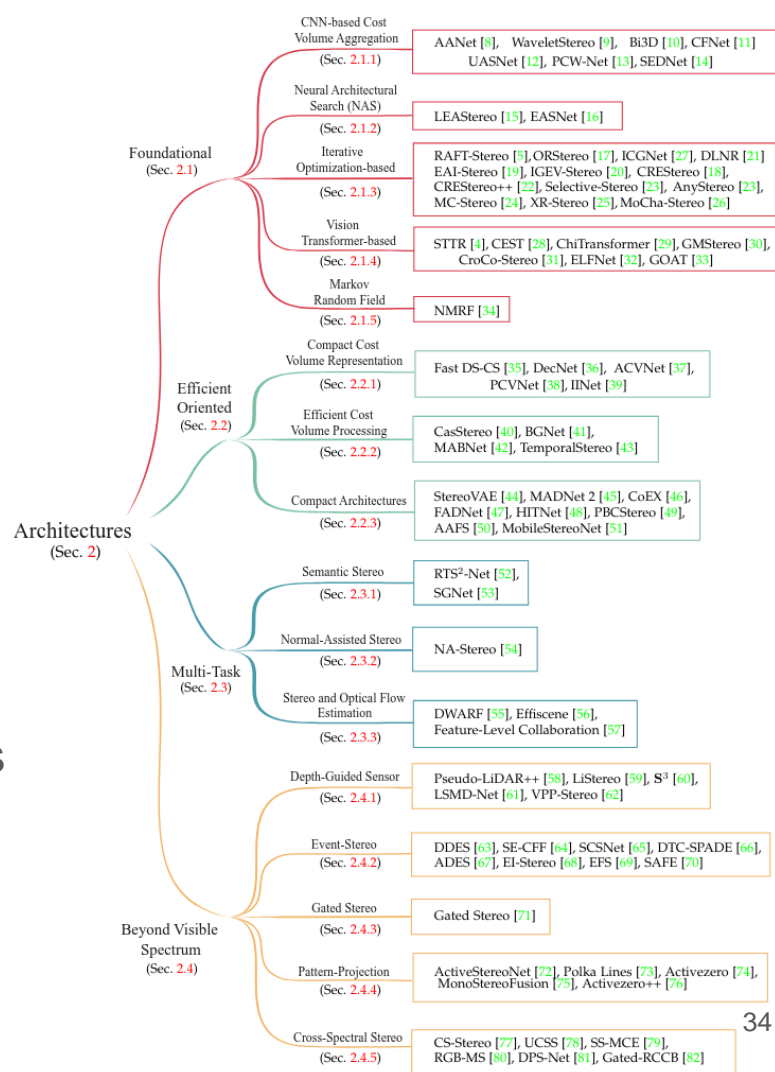
# Limitations – Occlusions



# Modern Stereo Matching

Nowadays, directly evaluating cost functions on hand-crafted features is not considered a good approach.

**Modern approaches** use advanced **deep learning** techniques to solve Stereo Matching, by **learning** to extract features from images.

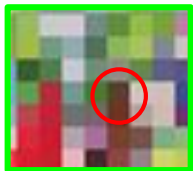


# Modern Stereo Matching

## Learning to match windows



Reference Example (taken from left image)



Positive Example (taken from right image).  
The central pixels is the same as the central 3D point of the reference image.



Negative Window (taken from right image).  
The central pixels is different from the central 3D point of the reference image.



Left Image



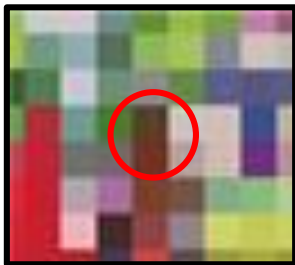
Right Image

Stereo Matching by Training a Convolutional Neural Network to Compare Image Patches. Zbontar and LeCun, JMLR, 2016.

# Modern Stereo Matching

## CNN + Siamese networks

- Extract features** with a network, and predict the matching cost  $[0, 1]$ .
- For each patch, find the best match
- Training: triplets of reference, positive and negative patches. **BCELoss**.



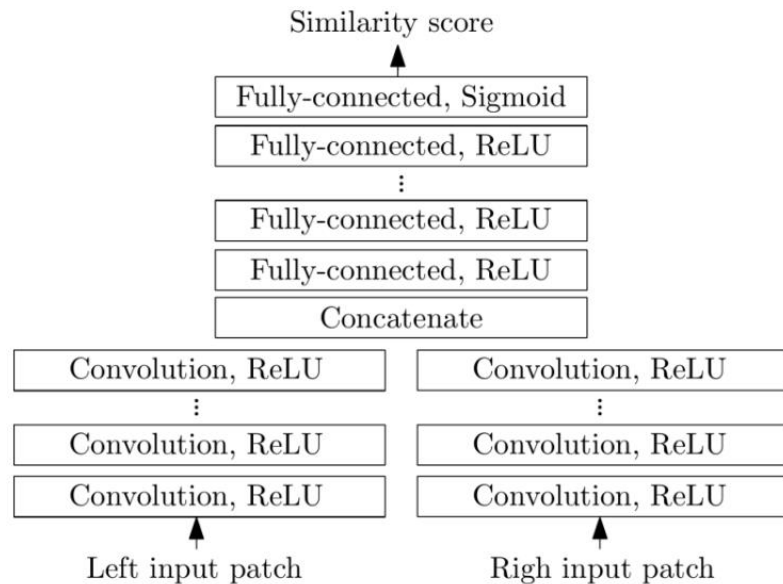
Reference



Positive

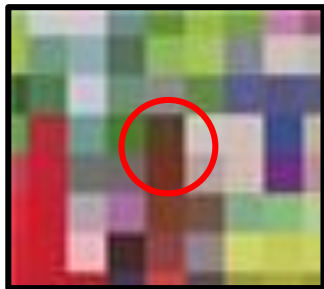


Negative



Stereo Matching by Training a Convolutional Neural Network to Compare Image Patches. Zbontar and LeCun, JMLR, 2016.

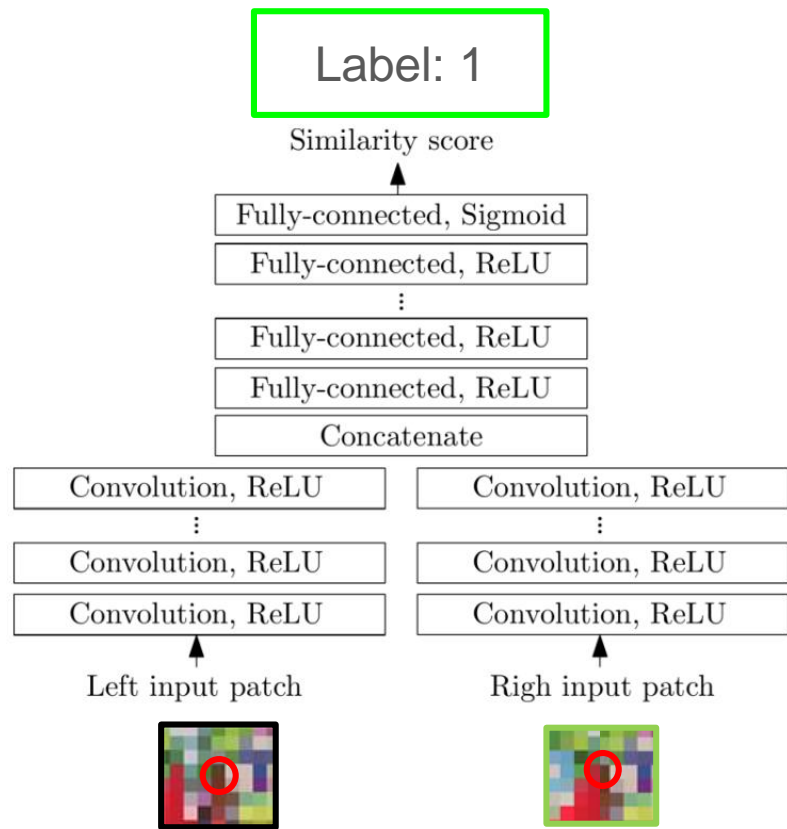
# Modern Stereo Matching



Reference

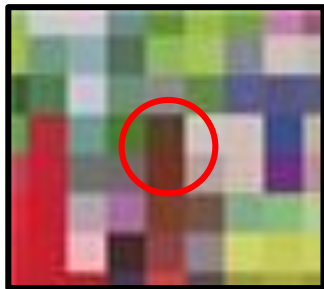


Positive



Stereo Matching by Training a Convolutional Neural Network to Compare Image Patches. Zbontar and LeCun, JMLR, 2016.

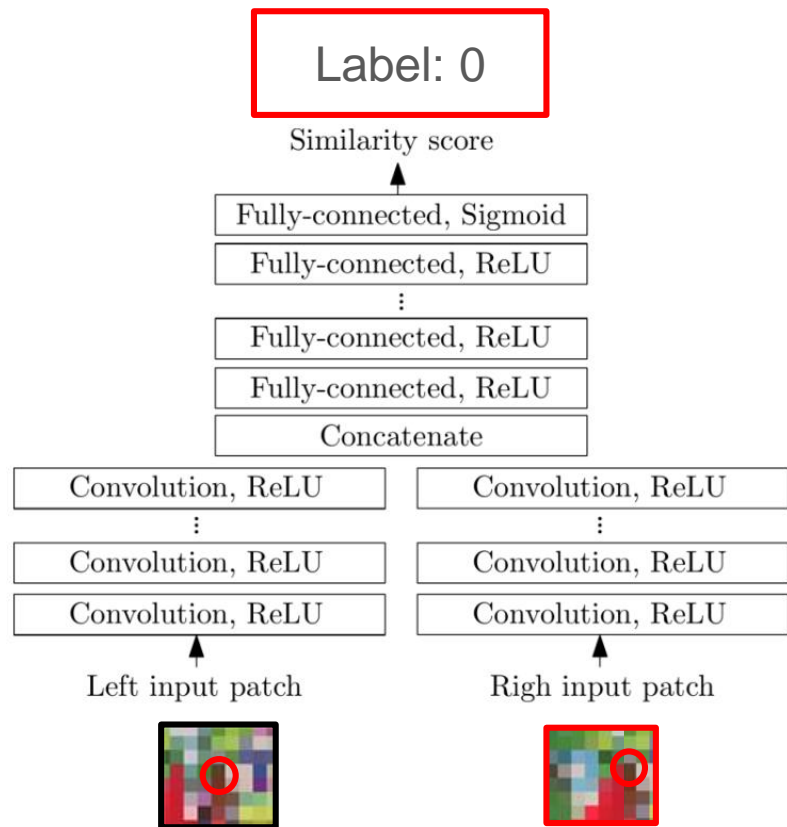
# Modern Stereo Matching



Reference



Negative

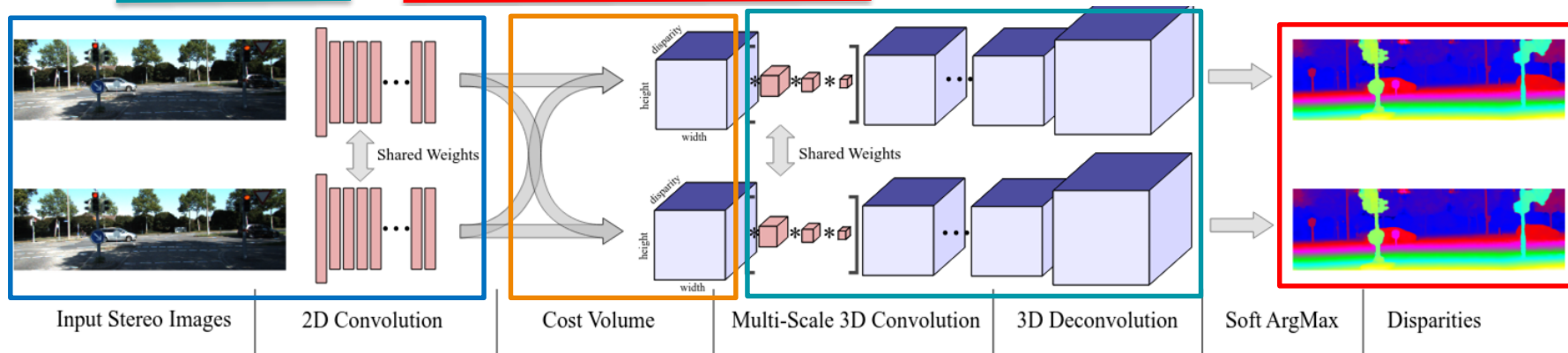


Stereo Matching by Training a Convolutional Neural Network to Compare Image Patches. Zbontar and LeCun, JMLR, 2016.

# Modern Stereo Matching

GC-Net:

- **End-to-End** architecture, trained with **L1** loss (GT disparity vs predicted disparity)
- Key idea: calculate **disparity cost volume** and apply 3D convolutions on it
- 1) extracts features with 2D convolutions 2) Cost volume construction 3) 3D convolutions 4) Disparity map prediction

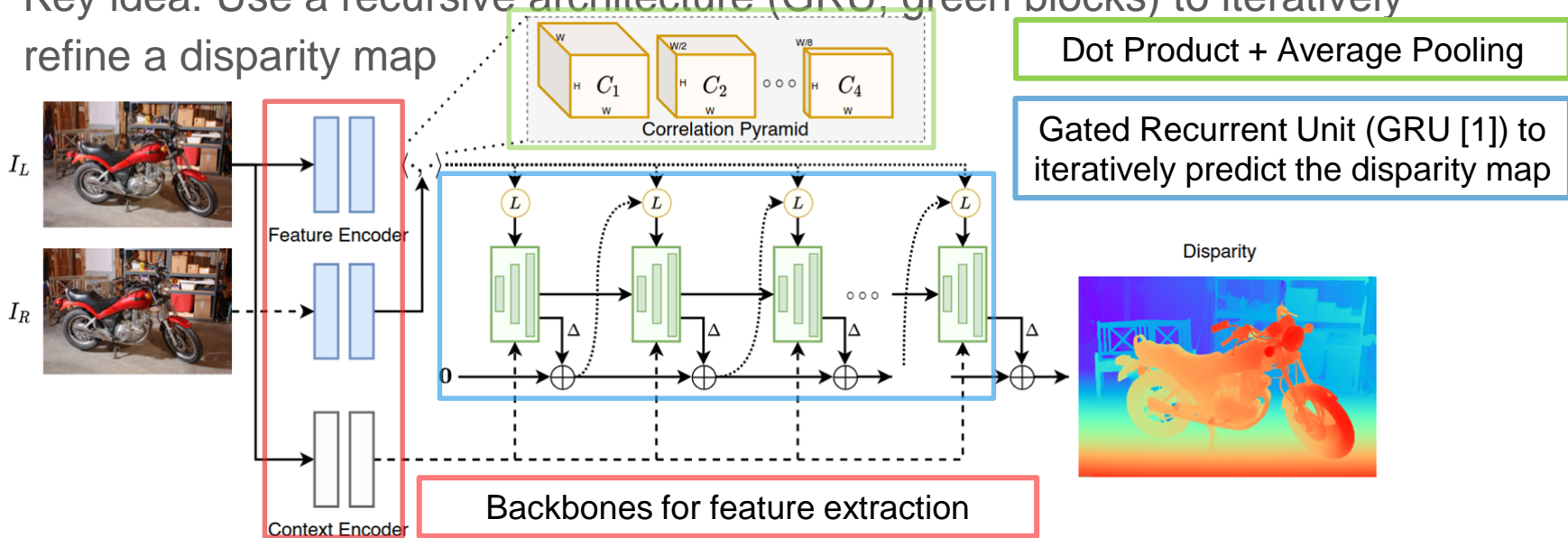


End-to-End Learning of Geometry and Context for Deep Stereo Regression. Kendall et al. ICCV, 2017.

# Modern Stereo Matching

## RAFT-Stereo:

- Key idea: Use a recursive architecture (GRU, green blocks) to iteratively refine a disparity map



# RAFT-Stereo Results



RAFT-Stereo: Multilevel Recurrent Field Transforms for Stereo Matching. Lipson et al. 3DV, 2021.

# Downstream Task

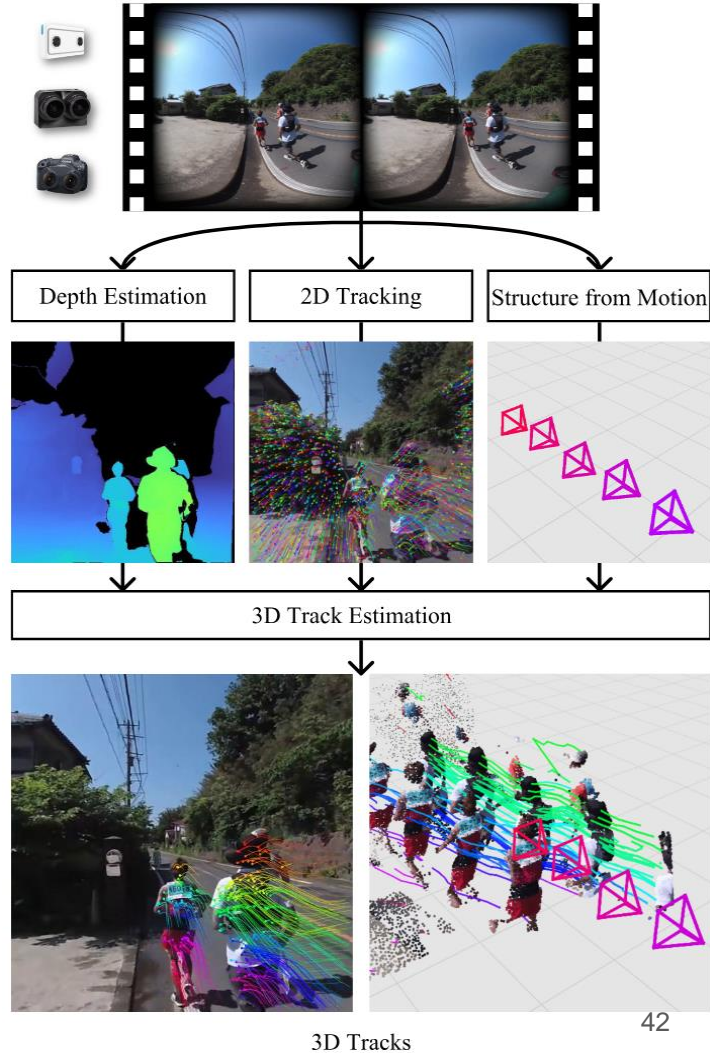
## *Stereo4D: Learning How Things*

*Move in 3D from Internet Stereo Videos. Jin et al. CVPR 25*

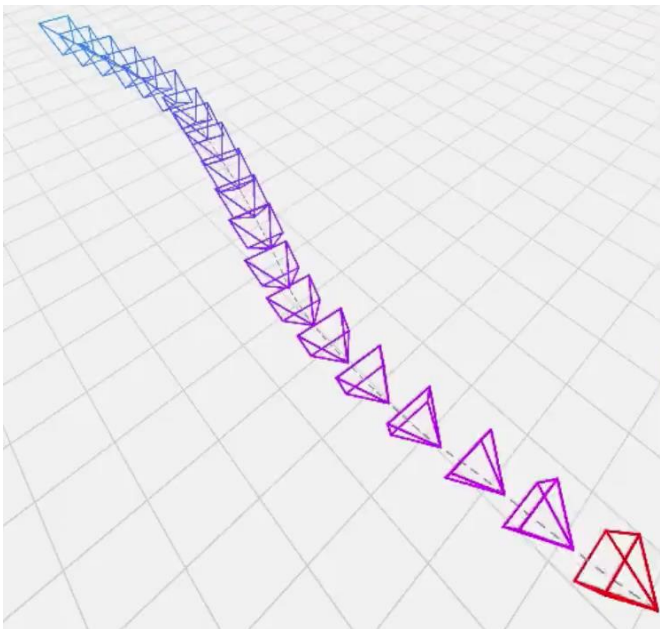
- Input:
  - Stereo Video
- Output:
  - Movement of points in 3D (tracks)
- Key idea:
  - Extract:
    - Depth with Stereo Matching (RAFT[1])
    - Camera Trajectory (SfM / COLMAP)
    - 2D point tracking (BootsTAP [2])

[1] RAFT: Recurrent all-pairs field transforms for optical flow. Teed et al. ECCV, 2020

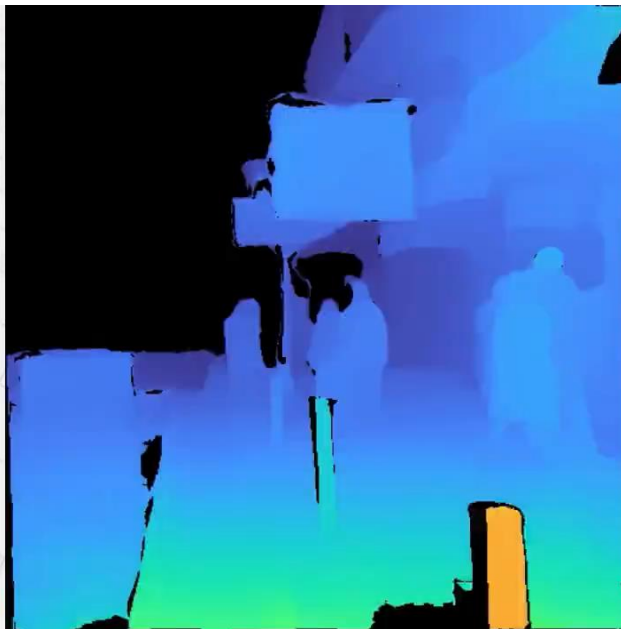
[2] BootsTAP: Bootstrapped Training for Tracking-Any-Point. Doersch et al. ICCV, 2024



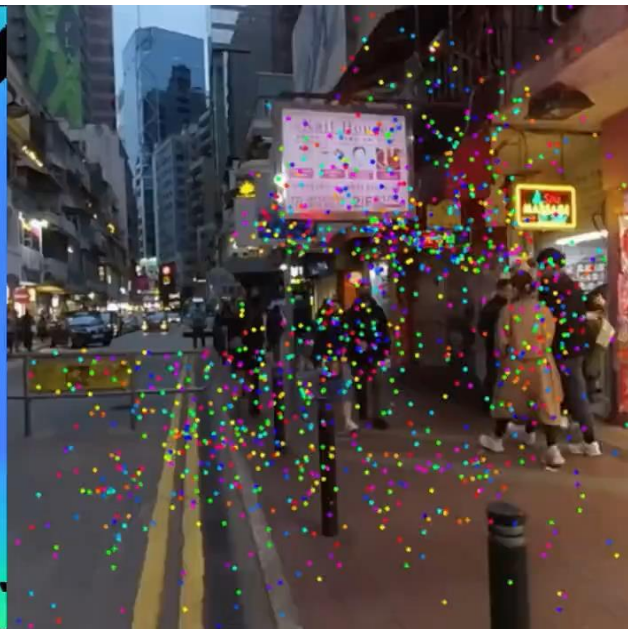
# Downstream Task



Camera Position



Disparity Map (from RAFT)



Point Tracking

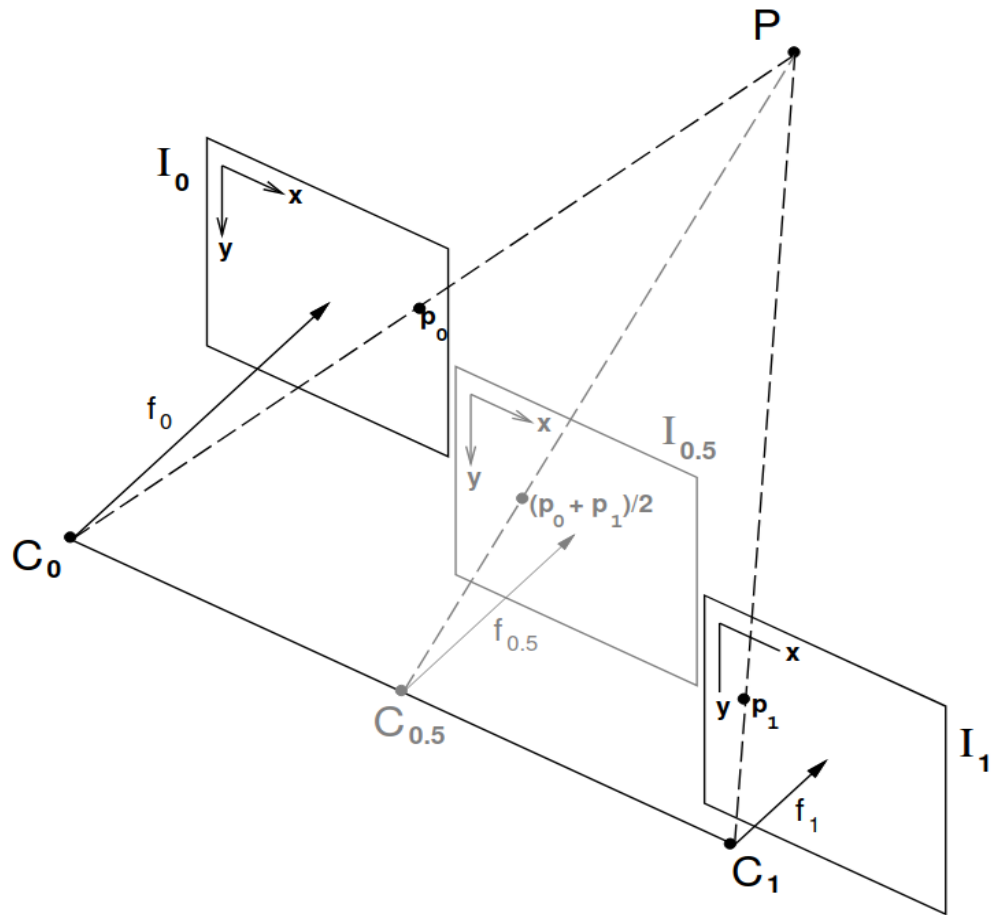
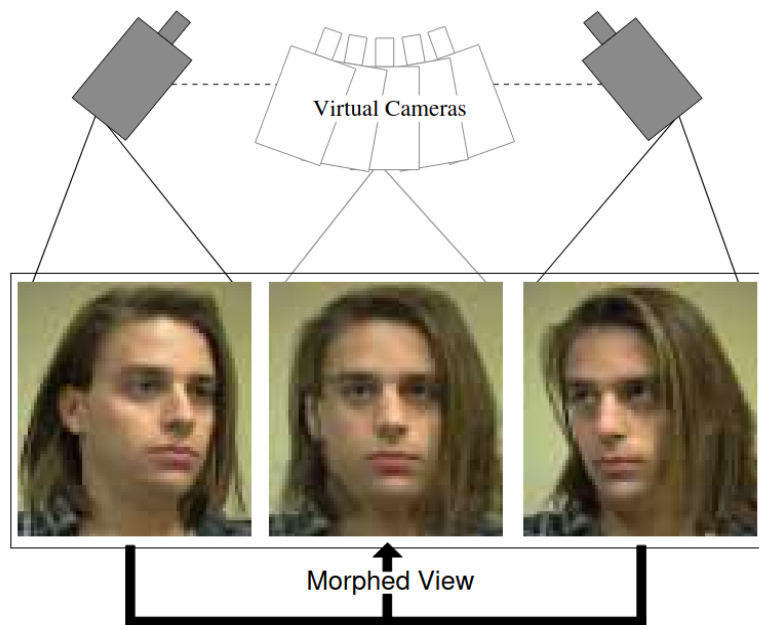
Stereo4D: Learning How Things Move in 3D from Internet Stereo Videos



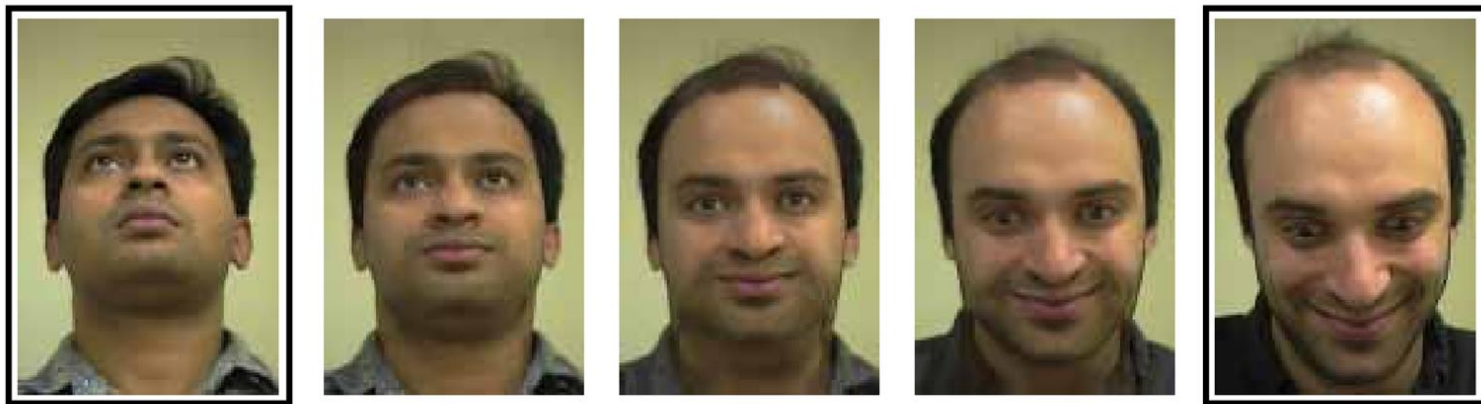
# Other Applications of Rectified Images

# View Morphing

Rectified images can be used for view morphing



# View Morphing



$\mathcal{I}_0$

$\mathcal{I}_{0.25}$

$\mathcal{I}_{0.5}$

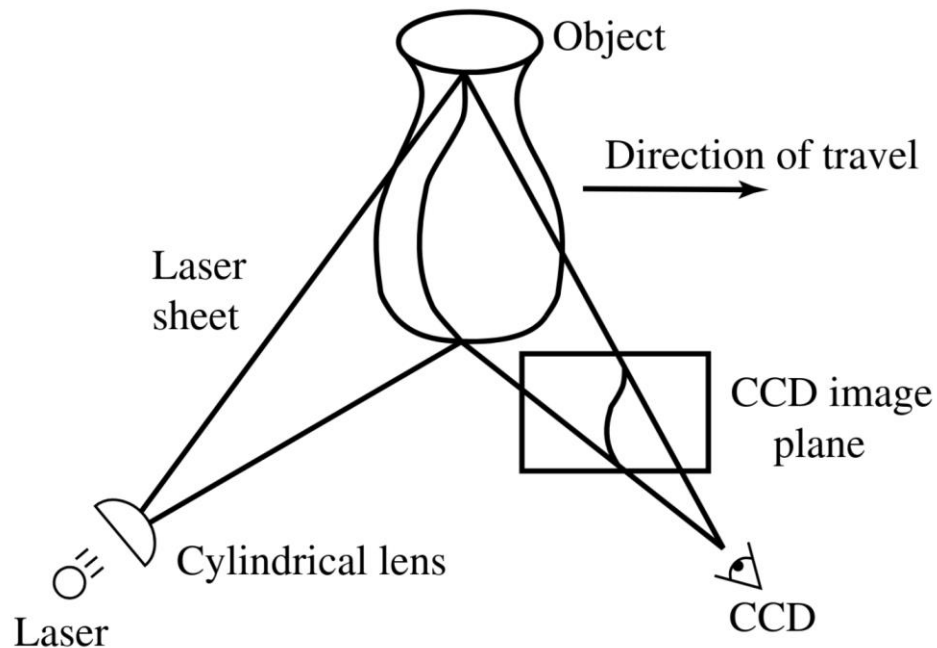
$\mathcal{I}_{0.75}$

$\mathcal{I}_1$

View morphing, Seitz and Dyer, 1996

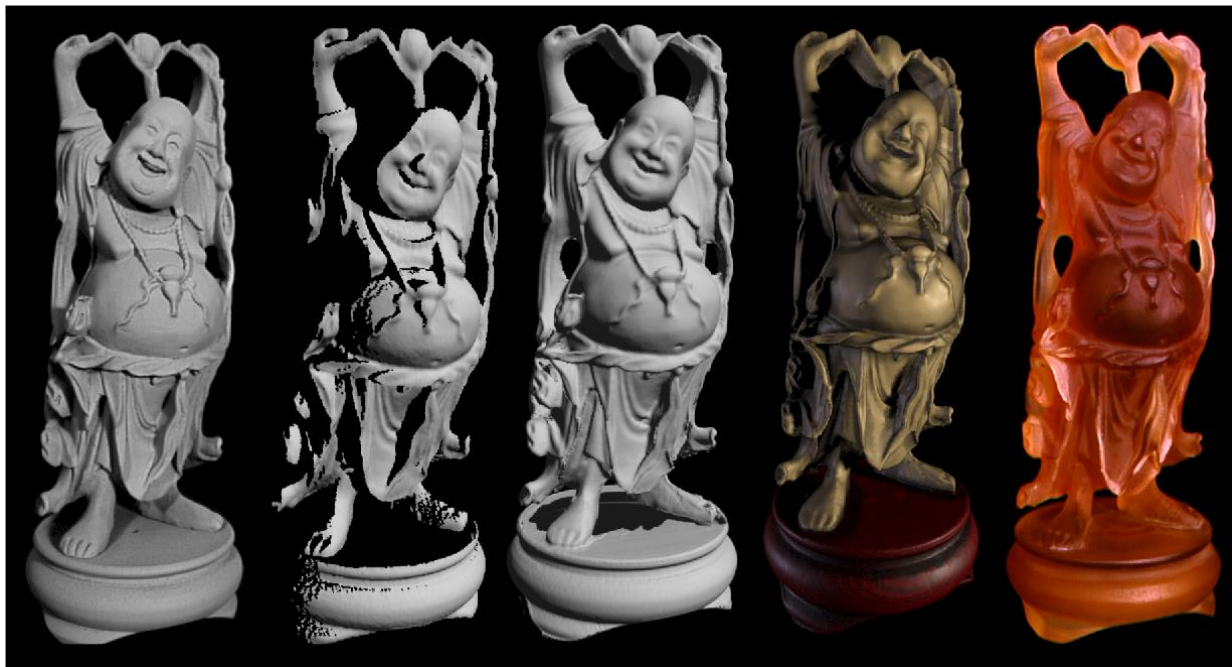
# Active Stereo

Some methods replace one of the two cameras with a projector:



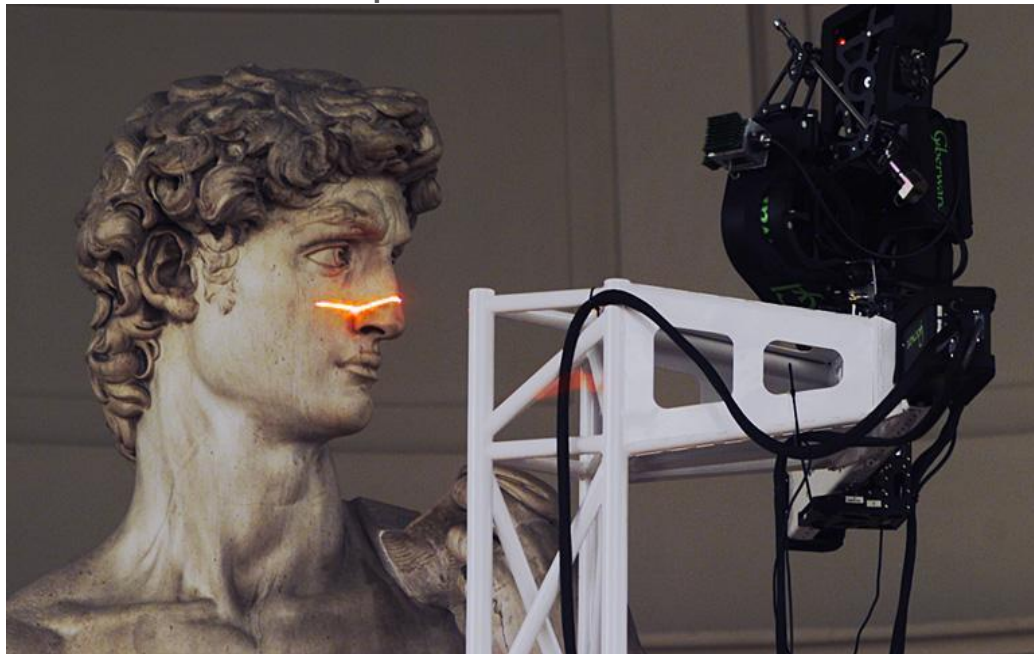
# Active Stereo

Some methods replace one of the two cameras with a projector:



# Active Stereo

Some methods replace one of the two cameras with a projector

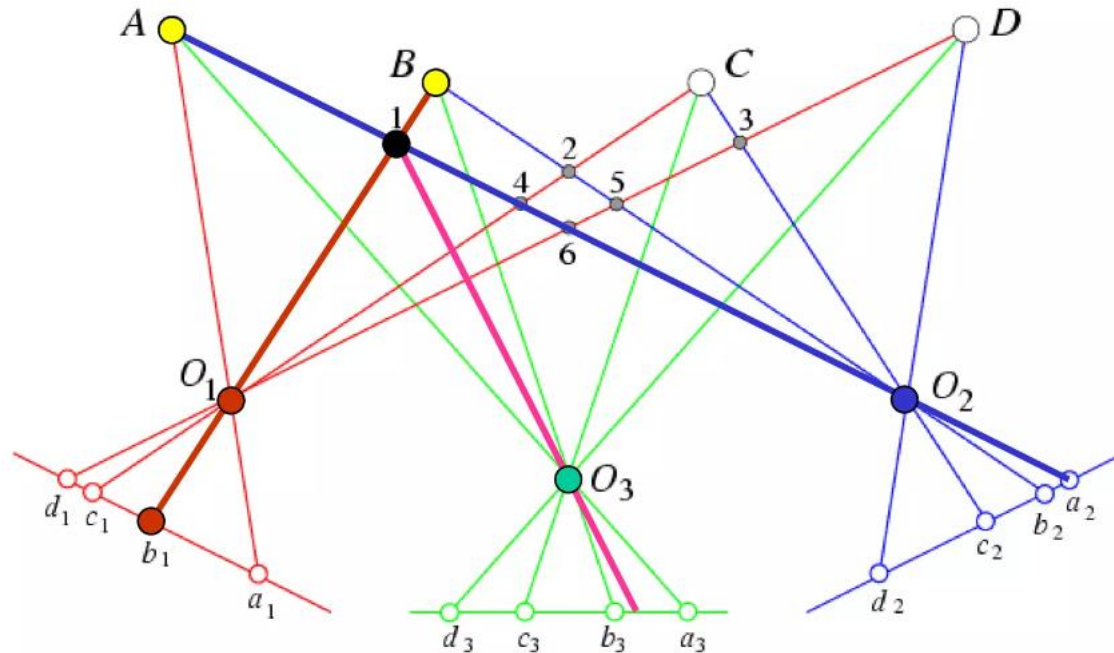


[https://graphics.stanford.edu/papers/digmich\\_falletti/](https://graphics.stanford.edu/papers/digmich_falletti/)

# Multiview Stereo Matching

# Multiview Stereo Matching

We can extend the problem of Stereo Matching to multiple cameras



# Multiview Stereo Matching

## Multiple-baseline Stereo:

- Problem: we have  $N$  images ( $N = 9$  for the example), and we want to reconstruct a 3D depth map



$I_0$



$I_1$

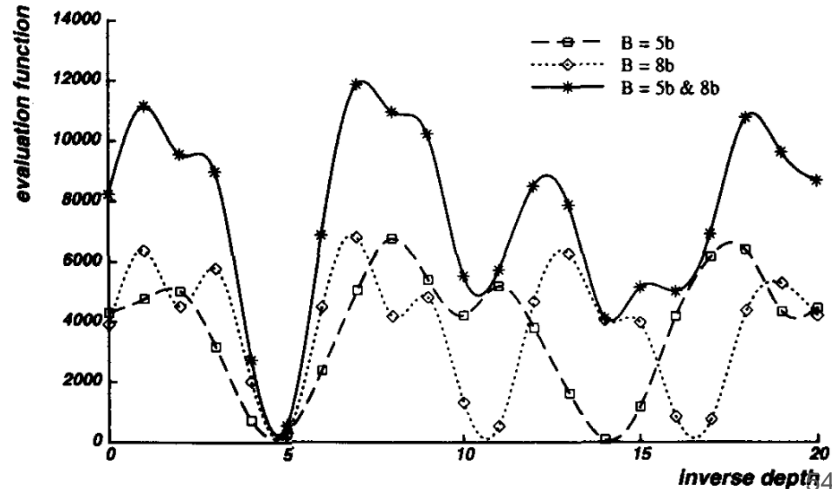
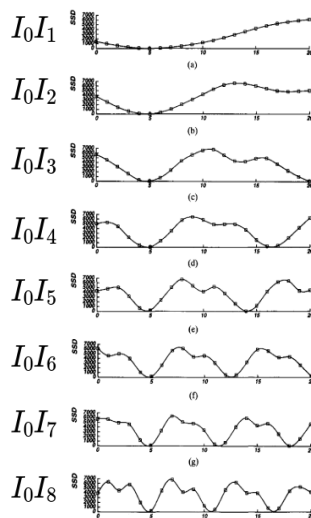


$I_8$

# Multiview Stereo Matching

## Multiple-baseline Stereo:

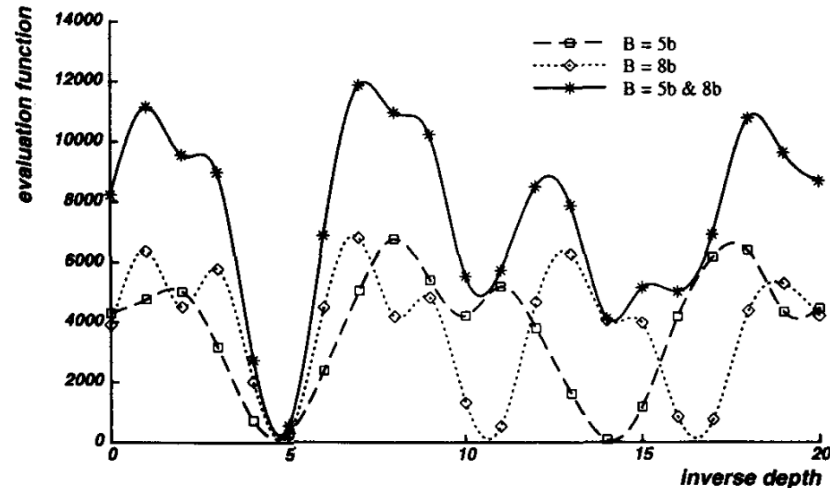
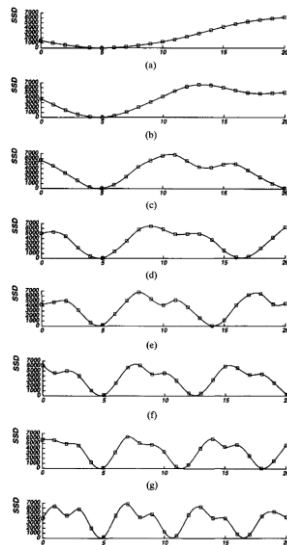
- Choose a view as reference, and slide a window across epipolar lines of other images. Store the **inverse depth** relative to the reference image  $I_0$
- Combine multiple baseline stereo pairs



# Multiview Stereo Matching

## Multiple-baseline Stereo:

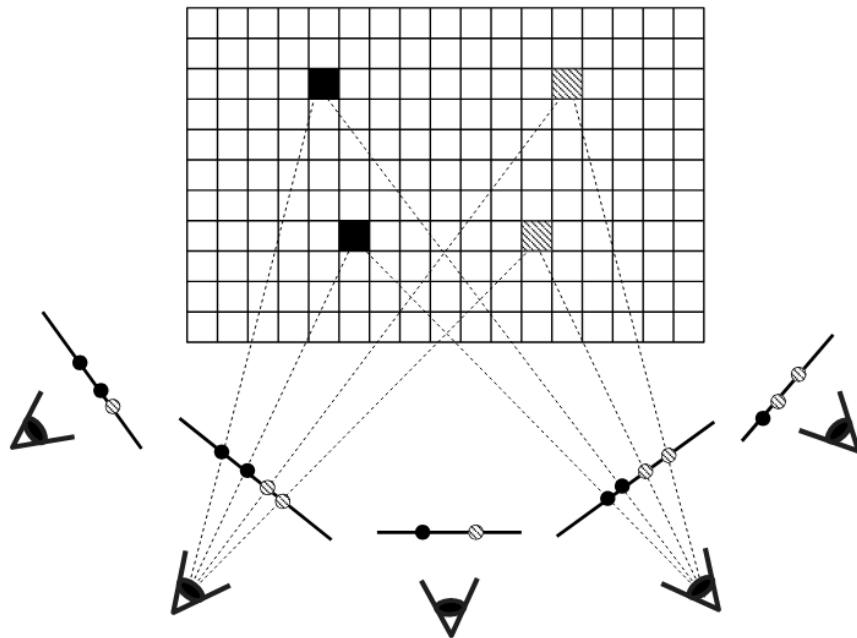
- Pros
  - Simple
- Cons
  - Computationally expensive
  - Sensible to light change and noise



# Multiview Stereo Matching

## Voxel Coloring:

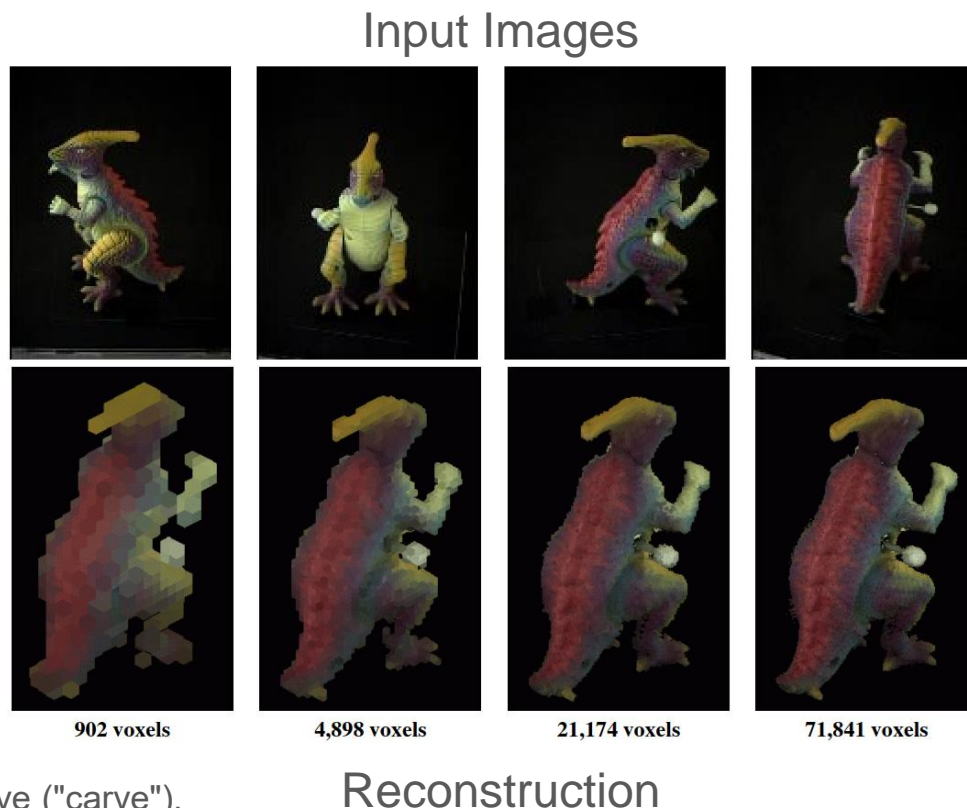
- Key idea:
  - Given a set of basis images and a grid of voxels, we wish to assign color values to voxels in a way that is consistent with all of the images.
  - By carving out inconsistent voxels, we retrieve a 3D reconstruction of the scene



# Multiview Stereo Matching

## Voxel Coloring:

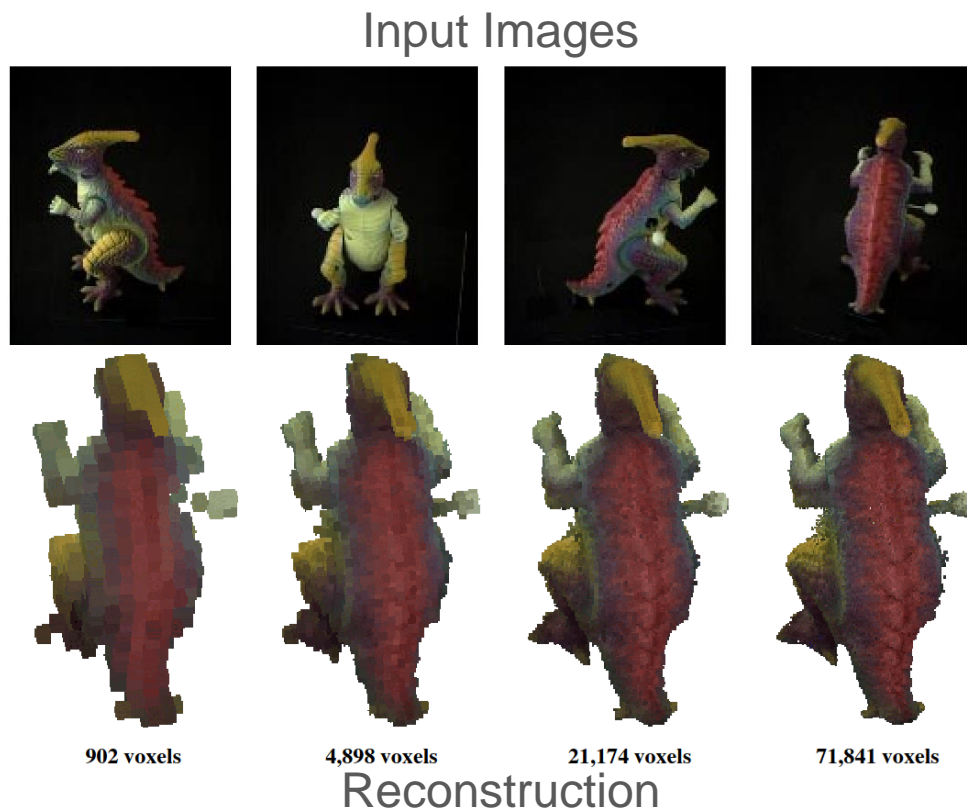
- Algorithm:
  - Initialize a 3D volume that encloses the scene.
  - For each voxel:
    - Project it into all images where it is visible.
    - Extract color values from those projections.
    - Compute photo-consistency (e.g., color variance).
  - If photo-consistent  $\rightarrow$  keep, else  $\rightarrow$  remove ("carve").
  - Proceed in visibility order (back-to-front w.r.t cameras).



# Multiview Stereo Matching

## Voxel Coloring:

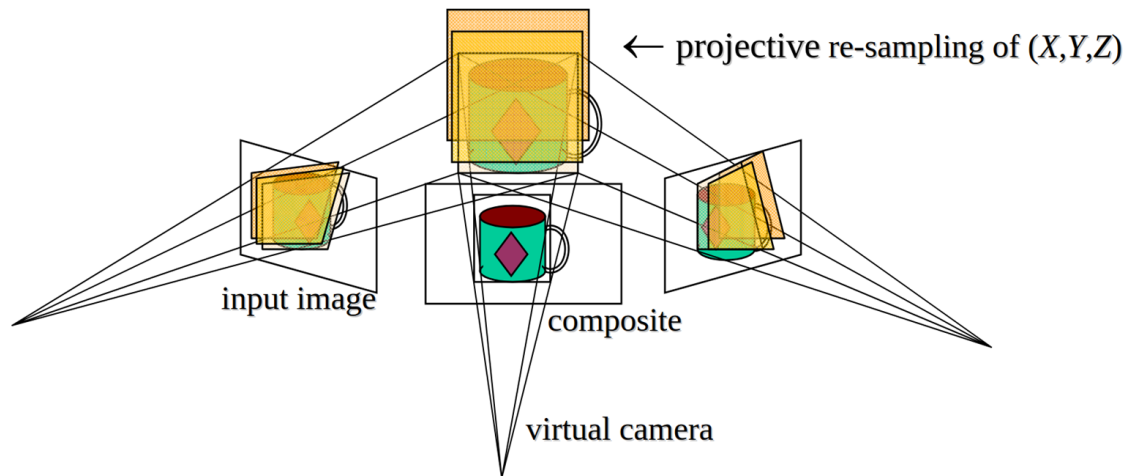
- **Pros:**
  - Simple and intuitive
  - Naturally handles occlusions via visibility reasoning.
  - No explicit feature matching needed.
- **Cons:**
  - Low resolution due to voxel size.
  - Computationally expensive (especially in dense grids).
  - Sensitive to calibration and lighting variation.



# Multiview Stereo Matching

## Plane-Sweep Stereo

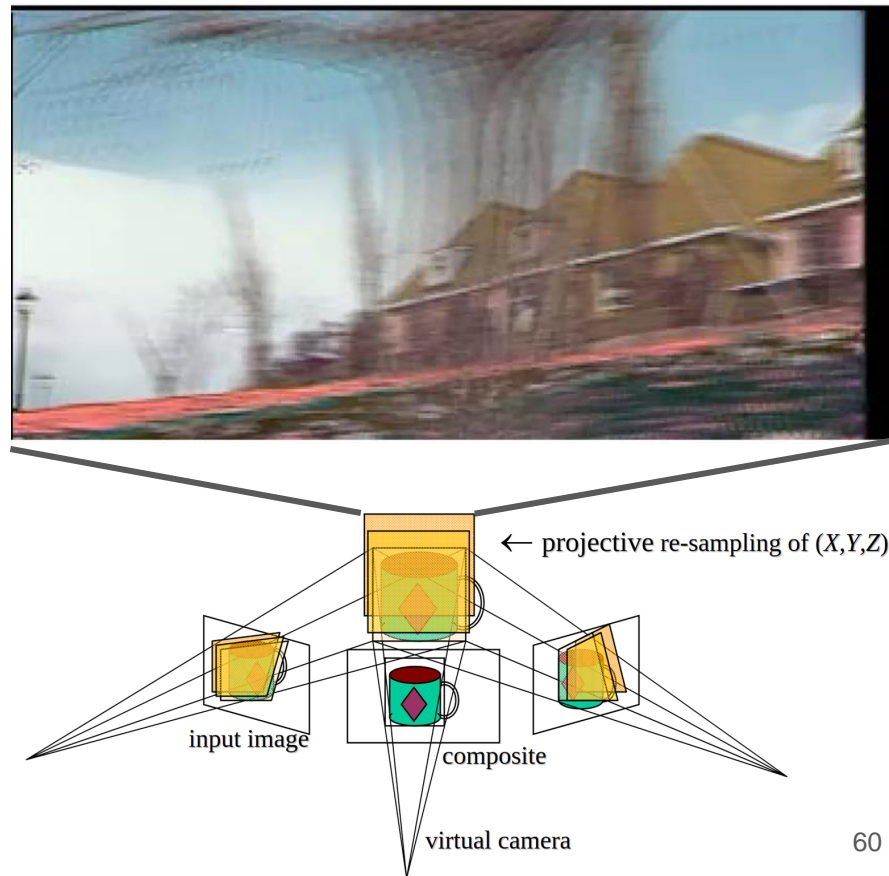
- Imagine taking a 3D scene and slicing it into a series of fronto-parallel planes at different depths.
- Projects all views onto these planes using known camera poses



# Multiview Stereo Matching

## Plane-Sweep Stereo

- For each pixel, measures photoconsistency (or feature consistency) for each plane, and choose the depth that gives the lowest (or highest) variance



# Multiview Stereo Matching

## Multi-View Stereopsis

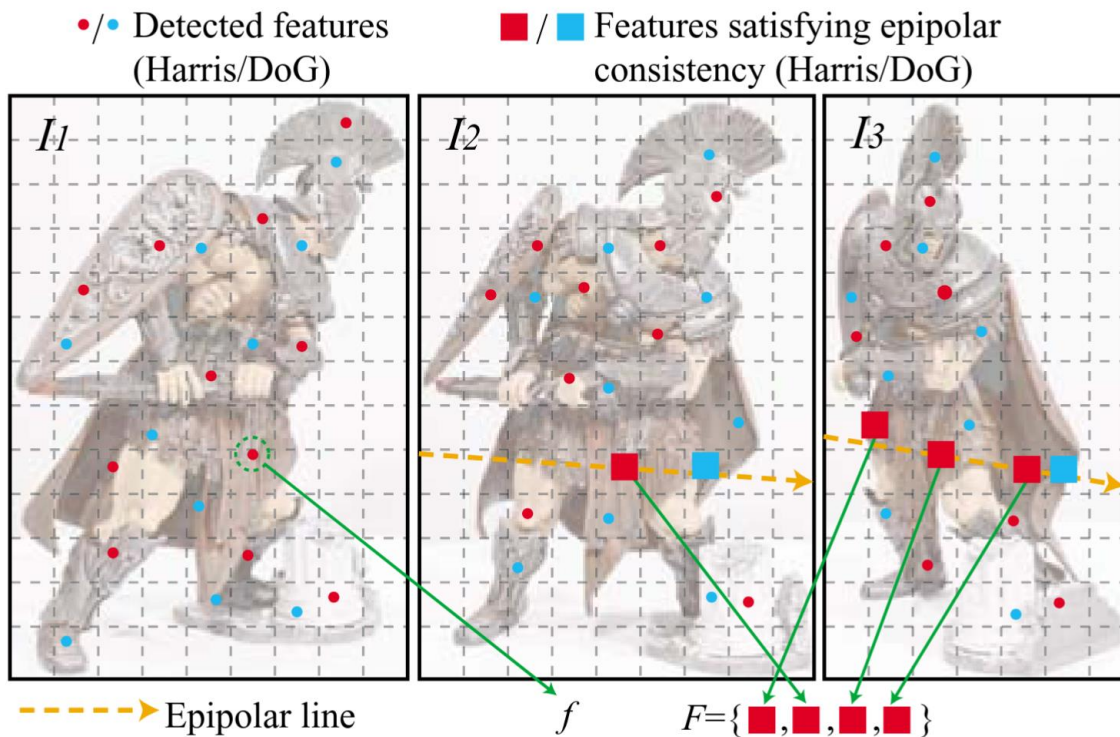


Accurate, Dense, and Robust Multi-View Stereopsis. Furukawa et al. CVPR '07

# Multiview Stereo Matching

## Multi-View Stereopsis

- Overall approach:
  - Divide the input images in patches, and find feature correspondence along epipolar lines
  - Triangulate 3D points based on feature matches
  - Reconstruct the surface from the pointcloud



# Modern Multiview Stereo Matching

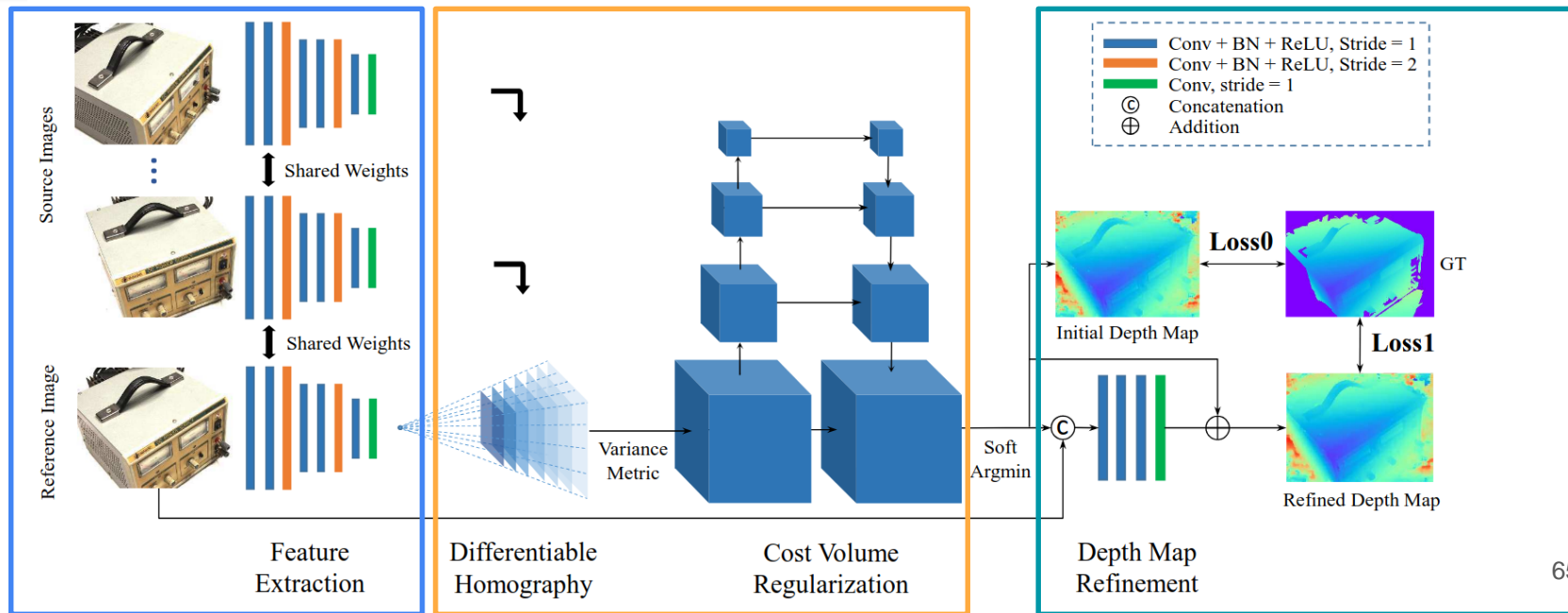
# Modern Multiview Stereo Matching

Modern Multiview Stereo Matching exploits data-driven deep learning techniques.

# Modern Multiview Stereo Matching

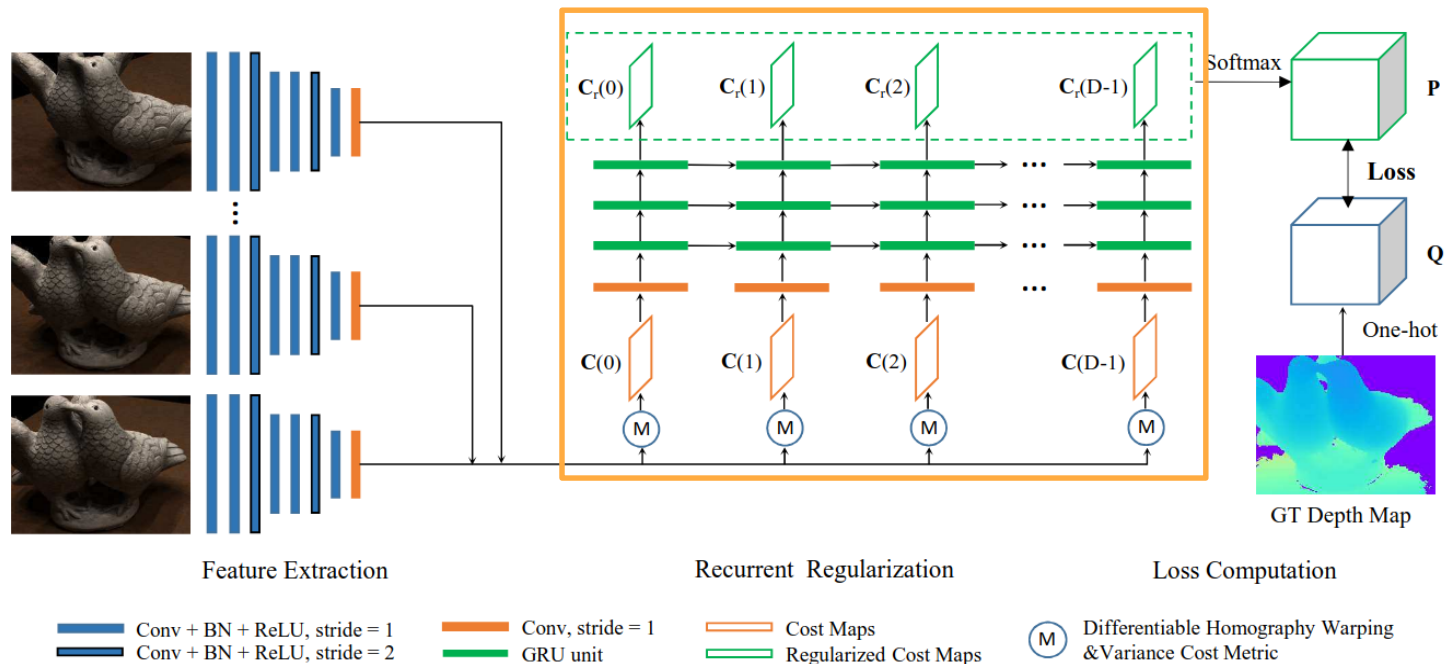
MVSNet proposes Cost Volume aggregation:

- Extract features, calculate cost volume and aggregate them with a cost metric, predict the depth map



# Modern Multiview Stereo Matching

R-MVSNet (CVPR2019) extends MVSNet by adding a recurrent neural network



# Modern Multiview Stereo Matching

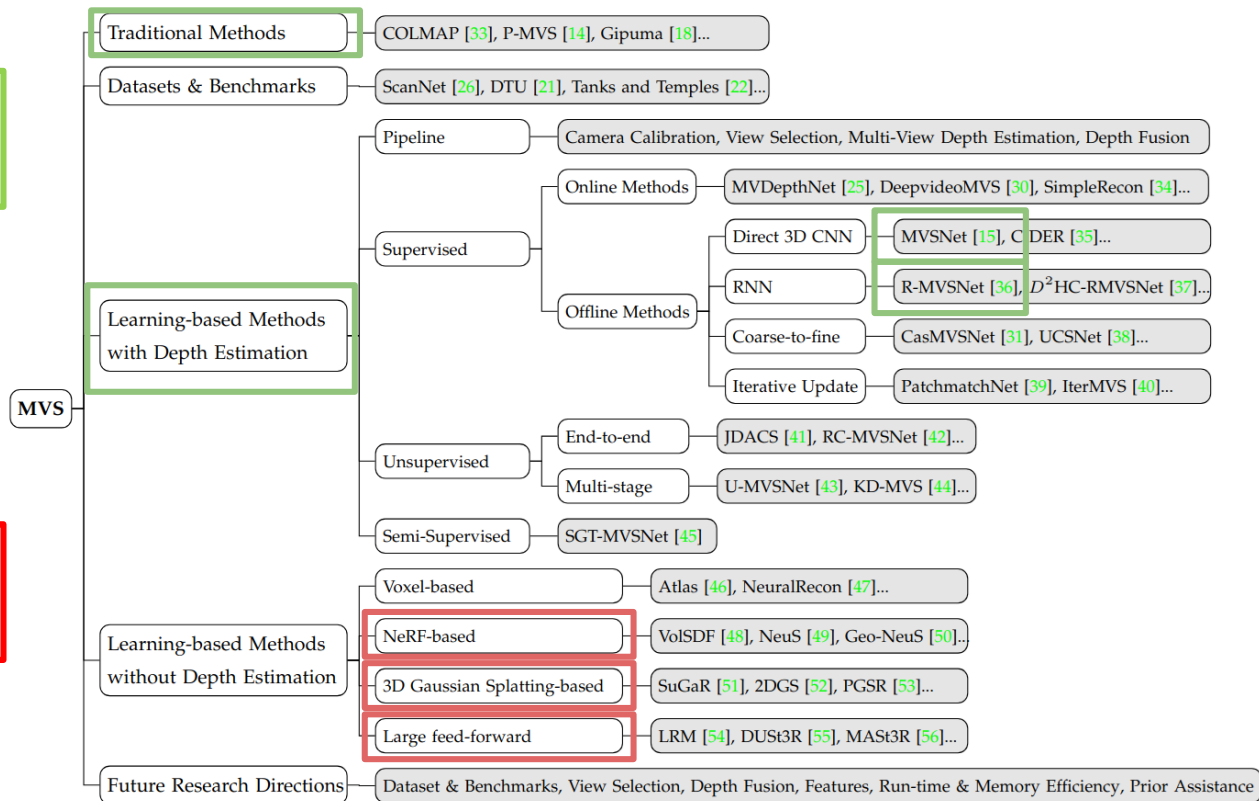
## Additional Readings:

- MVSAnywhere: Zero-Shot Multi-View Stereo, Izquierdo et al. CVPR 2025
- Stereo Anywhere: Robust Zero-Shot Deep Stereo Matching Even Where Either Stereo or Mono Fail. Bartolomei et al. CVPR 2025
- Selective-Stereo: Adaptive Frequency Information Selection for Stereo Matching. Wang et al. CVPR 2024
- Cross-spectral Gated-RGB Stereo Depth Estimation. Brucker et al. CVPR 2024
- MoCha-Stereo: Motif Channel Attention Network for Stereo Matching. Chen et al. CVPR 2024

# Modern Multiview Stereo Matching

What we saw until  
now

What we will see in  
future lectures



# Monocular Depth Estimation

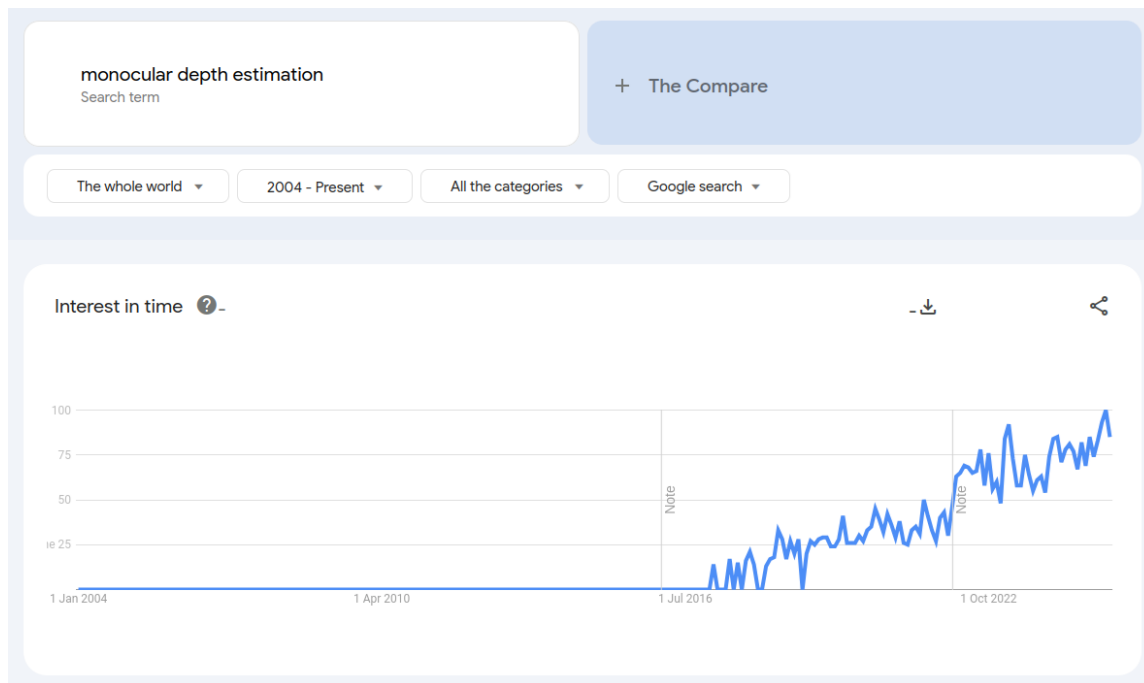
# Monocular Depth Estimation

**Monocular Depth Estimation** is the task of estimating the depth value (distance relative to the camera) of each pixel given a single (monocular) RGB image.



# Monocular Depth Estimation

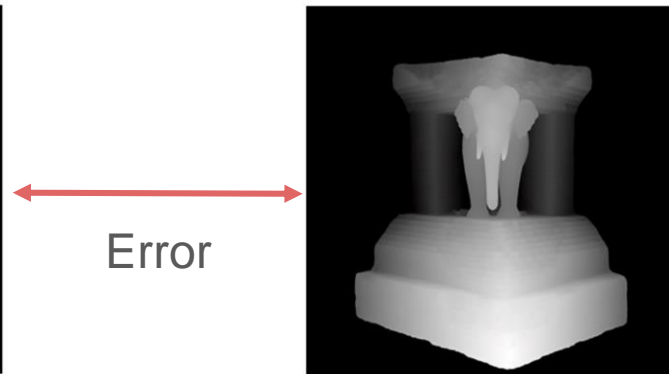
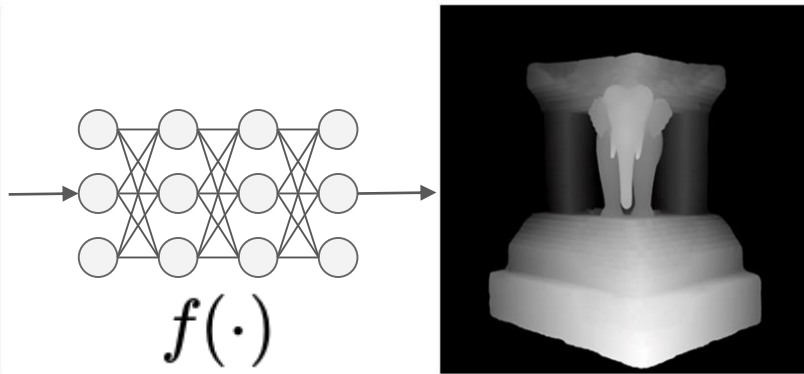
Recent Deep Learning advancements made Monocular Depth Estimation possible



# Monocular Depth Estimation

Learning from direct supervision:

- Data
- Architecture
- Training Objective



Error

Input Image  $x$

Predicted Depth  $y$

Ground Truth  $\hat{y}$

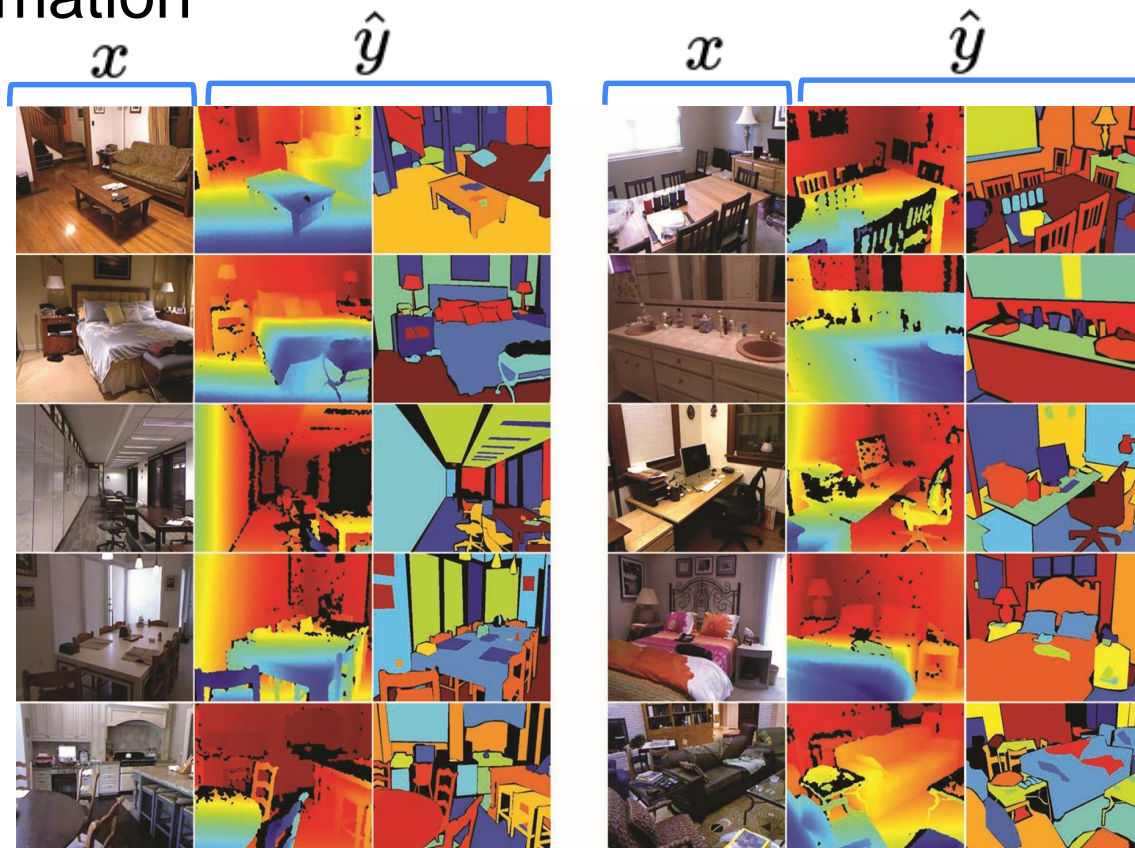
# Monocular Depth Estimation

- Collecting Real World Data



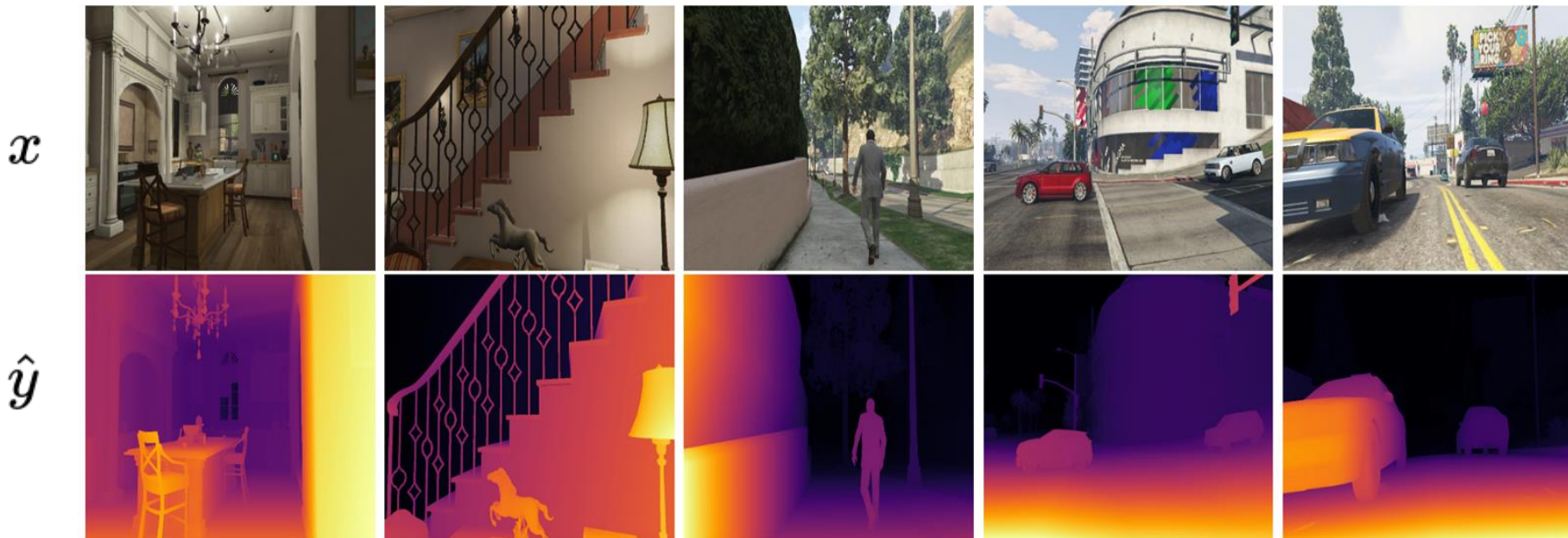
Kinect from Microsoft.

Cheap sensors made Monocular Depth Estimation possible.  
(stereo sensors, Time-of-Flight (ToF) sensors, Structured Light sensors...)



# Monocular Depth Estimation

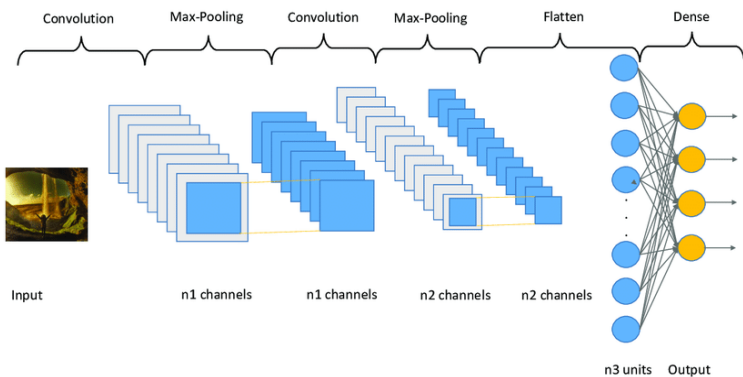
- Synthetic Data



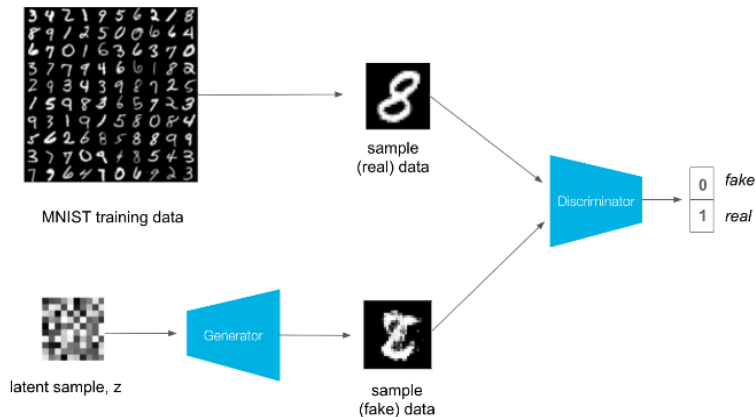
High-Resolution Synthetic RGB-D Datasets for Monocular Depth Estimation. Rajpal et al. CVPR 23

# Monocular Depth Estimation

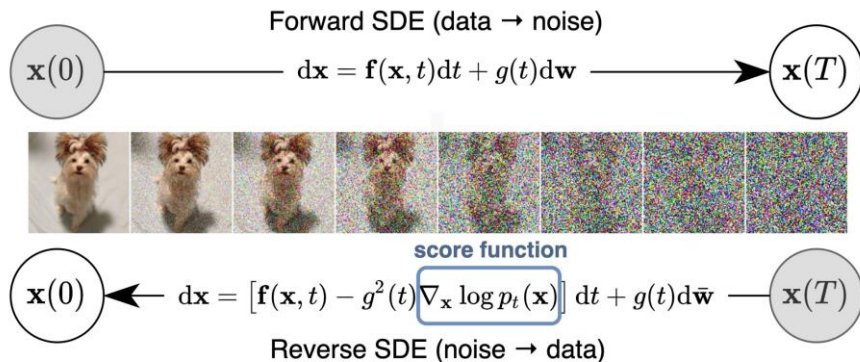
- Architecture
  - CNNs, GANs, Diffusion Models...



Convolutional Neural Networks

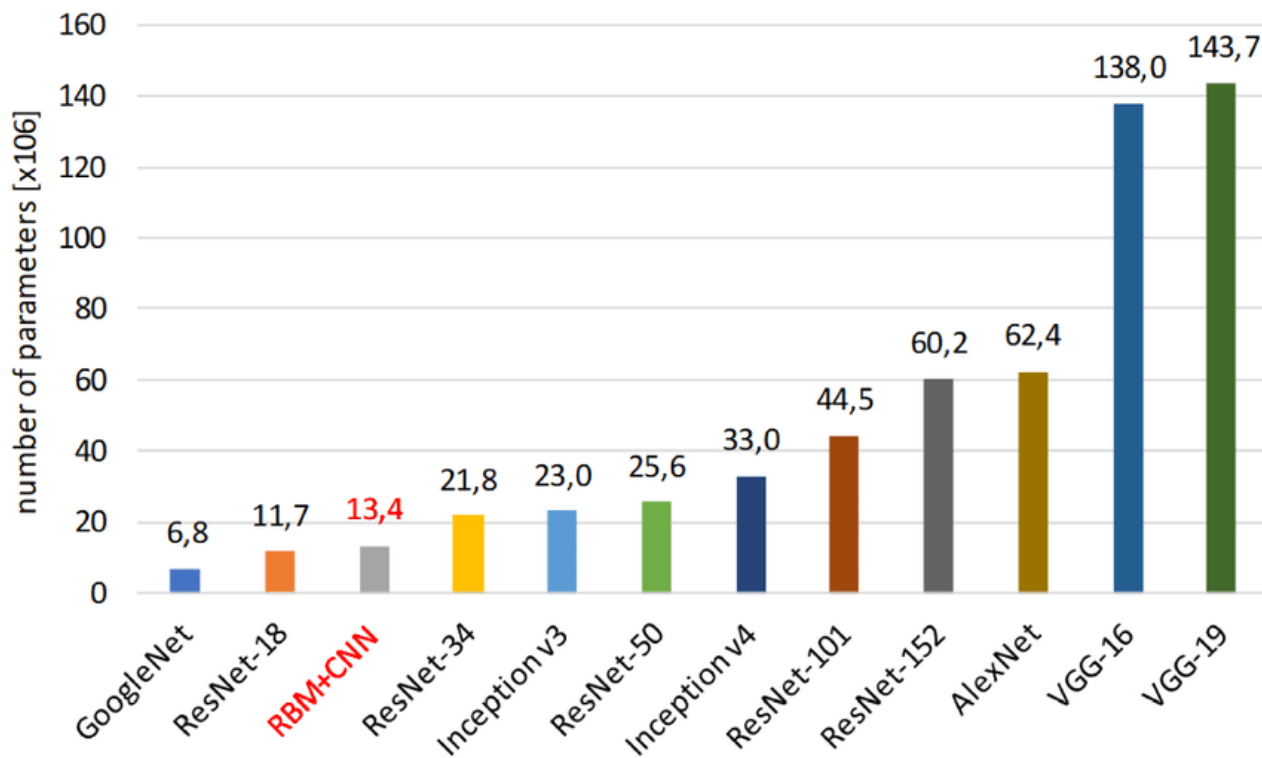


Generative Adversal Networks



Diffusion Models

# Monocular Depth Estimation



Sobczak et al. 2021

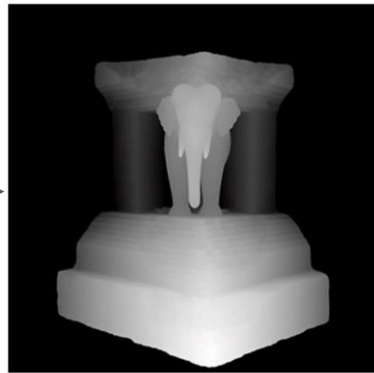
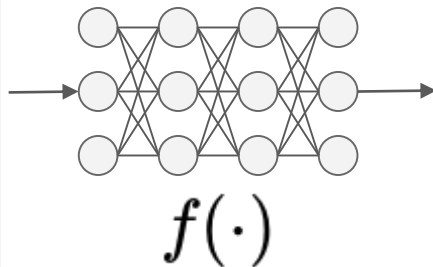
# Monocular Depth Estimation

- Training Objective

- A simple training objective could be to lower the error between prediction and ground truth



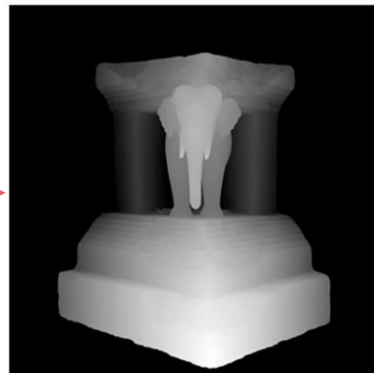
Input Image  $x$



Predicted Depth  $y$



Error



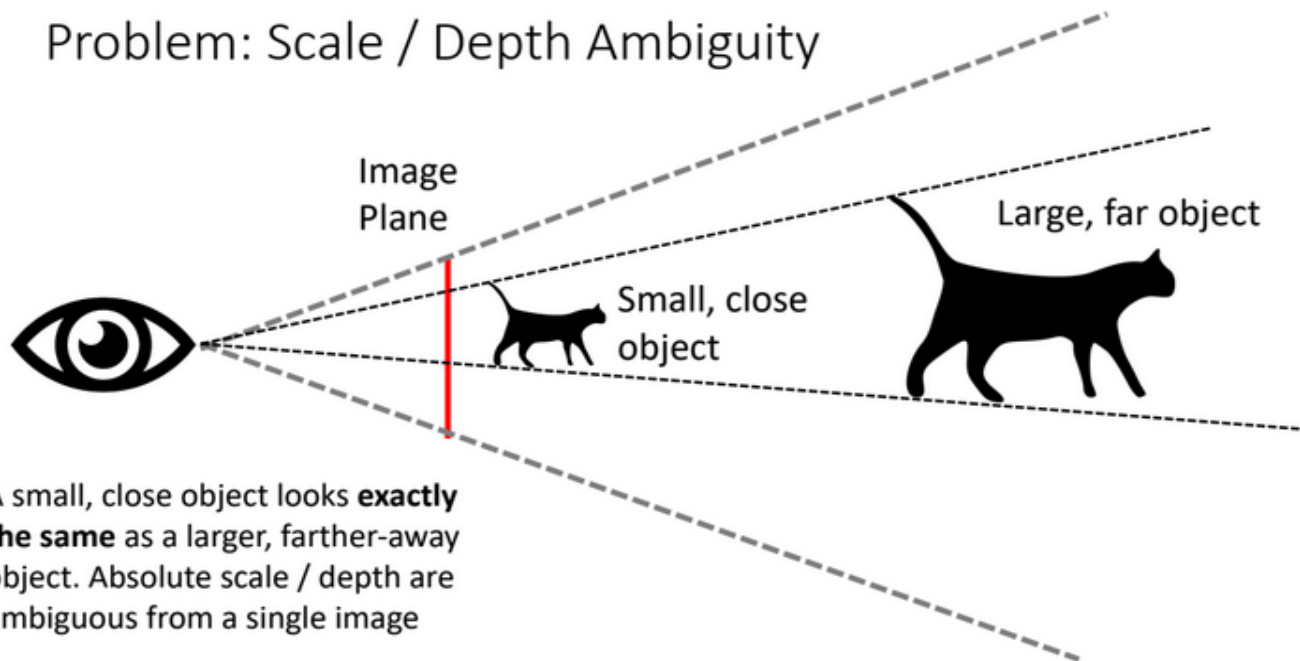
Ground Truth  $\hat{y}$

$$\mathcal{L}(y, \hat{y}) = \sum_i ||y_i - \hat{y}_i||^2, y = f(x)$$

# Monocular Depth Estimation

Ill-posed problem

Problem: Scale / Depth Ambiguity



# Monocular Depth Estimation

Sometimes we can fight back...

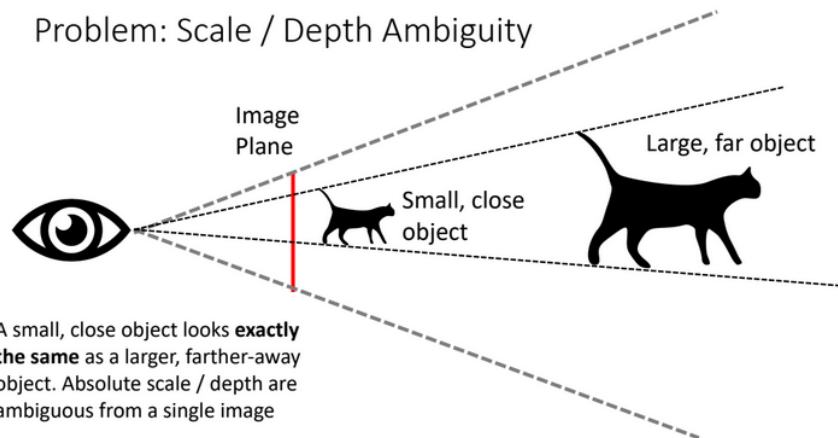
Scale-invariant mean square error:

$$L(y, \hat{y}) = \sum_i ||\log y_i - \log \hat{y}_i + \alpha(y, \hat{y})||^2$$

$$\alpha(y, \hat{y}) = \frac{1}{n} \sum_i (\log \hat{y}_i - \log y_i)$$

Use **mean depth** to measure **relationship between points** instead of their absolute value

Problem: Scale / Depth Ambiguity



# Monocular Depth Estimation

Sometimes we can fight back...

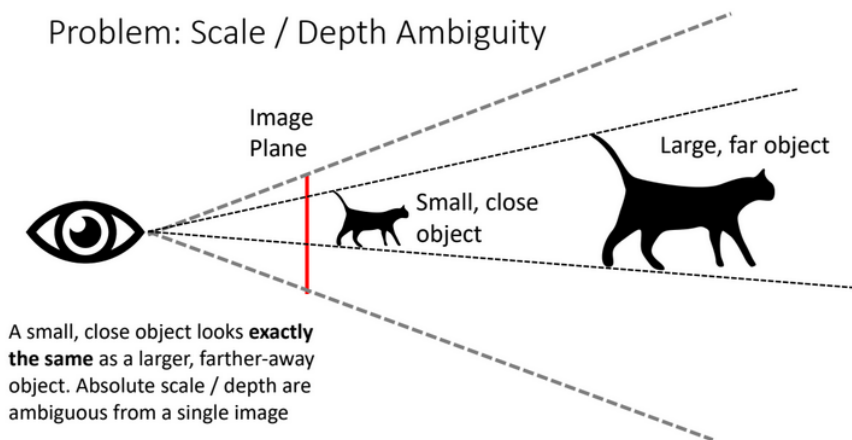
Scale-invariant mean square error:

$$L(y, \hat{y}) = \sum_i ||\log y_i - \log \hat{y}_i + \alpha(y, \hat{y})||^2$$

$$\alpha(y, \hat{y}) = \frac{1}{n} \sum_i (\log \hat{y}_i - \log y_i)$$

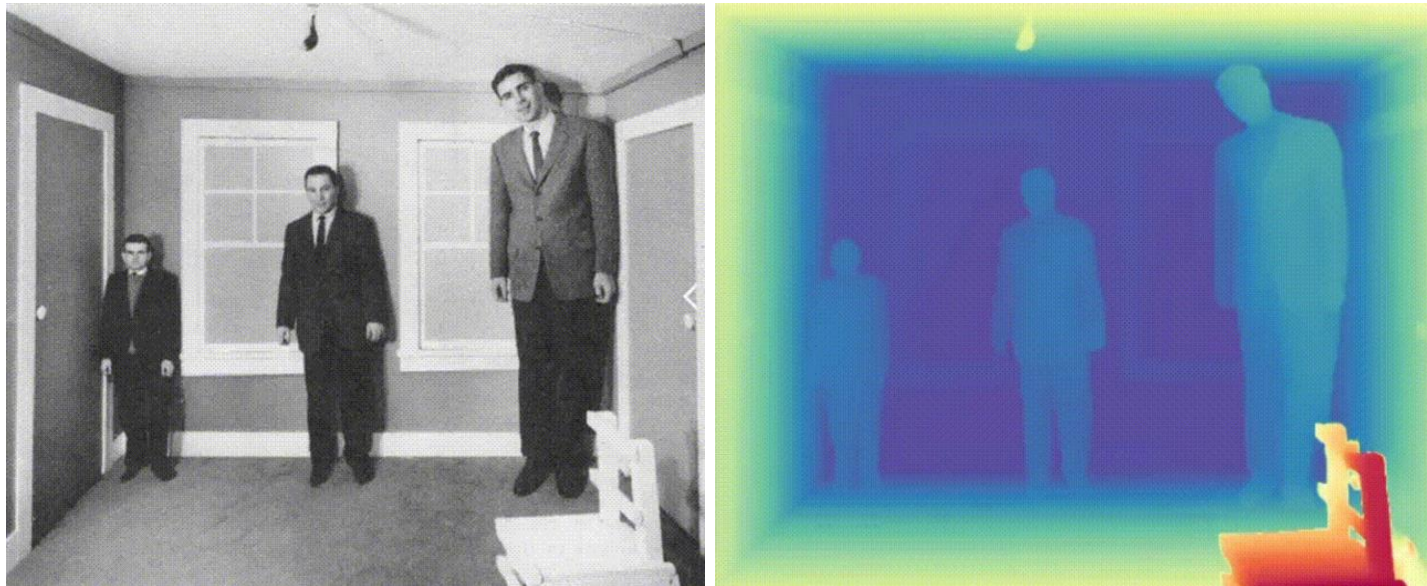
Why log-space?

If your output is off by a multiplicative constant  $y = s \cdot \hat{y}$ , by applying the loss in log-space we have  $\log(y) - \log(\hat{y}) = \log(s)$ , which is constant across all pixels so the loss ignores it, hence the scale invariance.



# Monocular Depth Estimation

Sometimes we can fight back... but we cannot do magic



Testing sota method “Depth Anything V2” (Yang et al. NeurIPS 2024) on Ames room illusion.

Demo: <https://huggingface.co/spaces/depth-anything/Depth-Anything-V2>

# Monocular Depth Estimation

Improve Monocular Depth Estimation:

- Acquire more data
- Improve network architecture
- Formulate better training objectives

# Monocular Depth Estimation

Today, Monocular Depth Estimation is a **very active field**

Method	Publication	Category	Inference	Dataset	Output	Source	
Zoedepth	Bhat et al. (2023)	Arxiv	discriminative	single	real	metric	open
Depth Anything	Yang et al. (2024a)	CVPR '24	discriminative	single	real	metric	open
Patch Fusion	Li et al. (2024a)	CVPR '24	discriminative	multiple	real	metric	open
Unidepth	Piccinelli et al. (2024)	CVPR '24	discriminative	single	real	metric	open
Marigold	Ke et al. (2024)	CVPR '24	generative	multiple	synthetic	relative	open
DMD	Saxena et al. (2023)	Arxiv	generative	multiple	real	metric	close
Depth Anything v2	Yang et al. (2024b)	NeurIPS '24	discriminative	single	real+synthetic	metric	open
GeoWizard	Fu et al. (2024)	ECCV '24	generative	multiple	real+synthetic	relative	open
Patch Refiner	Li et al. (2024b)	ECCV '24	discriminative	multiple	real+synthetic	metric	open
Depth pro	Bochkovskii et al. (2024)	Arxiv	discriminative	multiple	real+synthetic	metric	open
DAC	Guo et al. (2025)	Arxiv	discriminative	single	real+synthetic	metric	open

Survey on Monocular Metric Depth Estimation. Zhang, ArXiv preprint 2025

# Monocular Depth Estimation

- Depth Anything proposes a semi-supervised self-learning approach to enhance generalization

## Depth Anything: Unleashing the Power of Large-Scale Unlabeled Data

Lihe Yang<sup>1</sup> Bingyi Kang<sup>2†</sup> Zilong Huang<sup>2</sup> Xiaogang Xu<sup>3,4</sup> Jiashi Feng<sup>2</sup> Hengshuang Zhao<sup>1‡</sup>

<sup>1</sup>HKU

<sup>2</sup>TikTok

<sup>3</sup>CUHK

<sup>4</sup>ZJU

† project lead    ‡ corresponding author

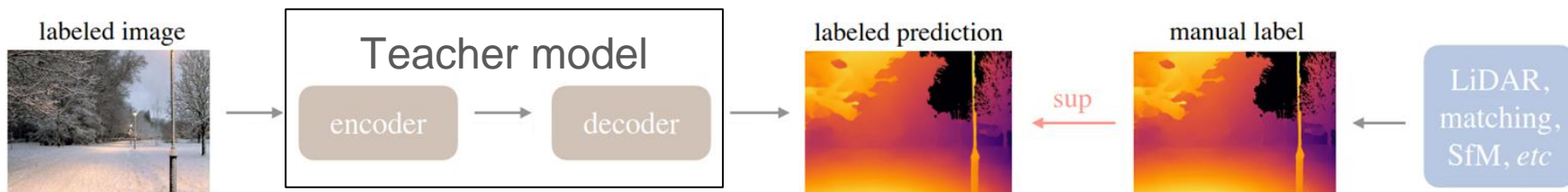
<https://depth-anything.github.io>



Figure 1. Our model exhibits impressive generalization ability across extensive unseen scenes. **Left two columns:** COCO [36]. **Middle two:** SA-1B [27] (a hold-out unseen set). **Right two:** photos captured by ourselves. Our model works robustly in low-light environments (1st and 3rd column), complex scenes (2nd and 5th column), foggy weather (5th column), and ultra-remote distance (5th and 6th column), etc.

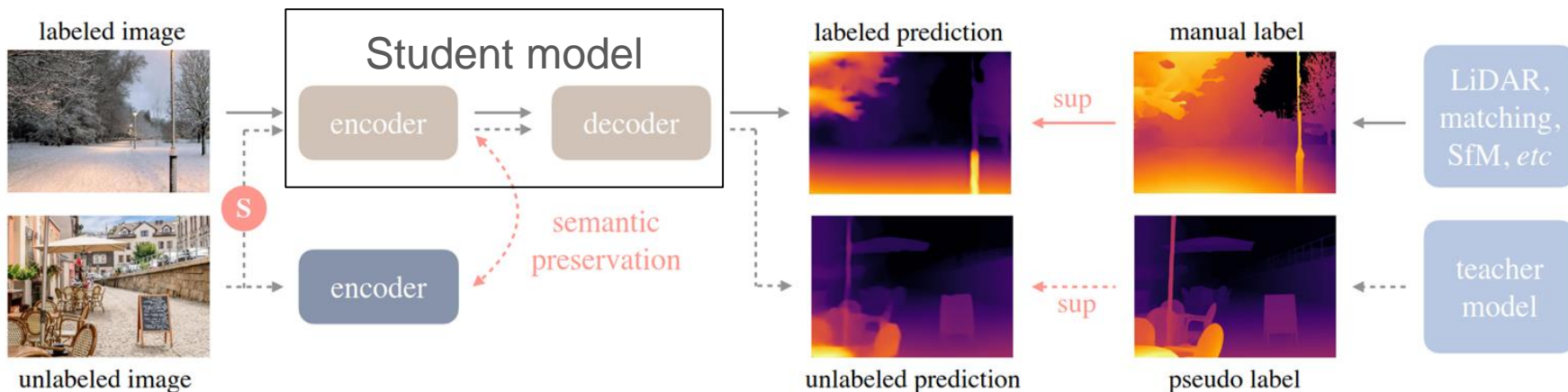
# Monocular Depth Estimation

- Depth Anything proposes a semi-supervised self-learning approach to enhance generalization
  - First, a **teacher model** learns monocular depth estimation (supervised).



# Monocular Depth Estimation

- Depth Anything proposes a semi-supervised self-learning approach to enhance generalization
  - First, a **teacher model** learns monocular depth estimation (supervised).
  - Then, a **student model** learns to mimic teacher's predictions on unlabeled images under input perturbation (color distortions, and CutMix [1])



# Monocular Depth Estimation

- Exploiting Feature Extraction Backbones

## DINOv2: Learning Robust Visual Features without Supervision

Maxime Oquab<sup>\*\*</sup>, Timothée Darcet<sup>\*\*</sup>, Théo Moutakanni<sup>\*\*</sup>,  
Huy V. Vo<sup>\*</sup>, Marc Szafraniec<sup>\*</sup>, Vasil Khalidov<sup>\*</sup>, Pierre Fernandez, Daniel Haziza,  
Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba,  
Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat,  
Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal<sup>1</sup>,  
Patrick Labatut<sup>\*</sup>, Armand Joulin<sup>\*</sup>, Piotr Bojanowski<sup>\*</sup>

Meta AI Research

<sup>1</sup>Inria

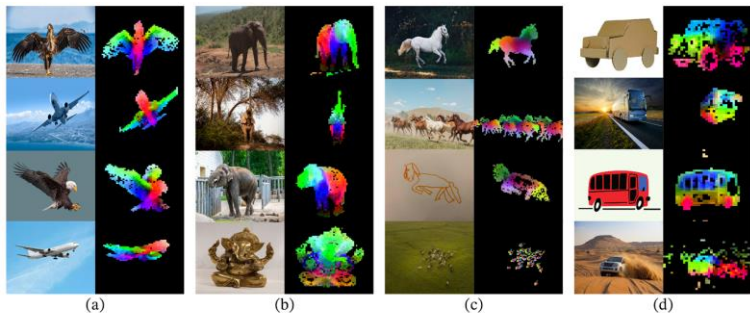


Figure 1: **Visualization of the first PCA components.** We compute a PCA between the patches of the images from the same column (a, b, c and d) and show their first 3 components. Each component is matched to a different color channel. Same parts are matched between related images despite changes of pose, style or even objects. Background is removed by thresholding the first PCA component.

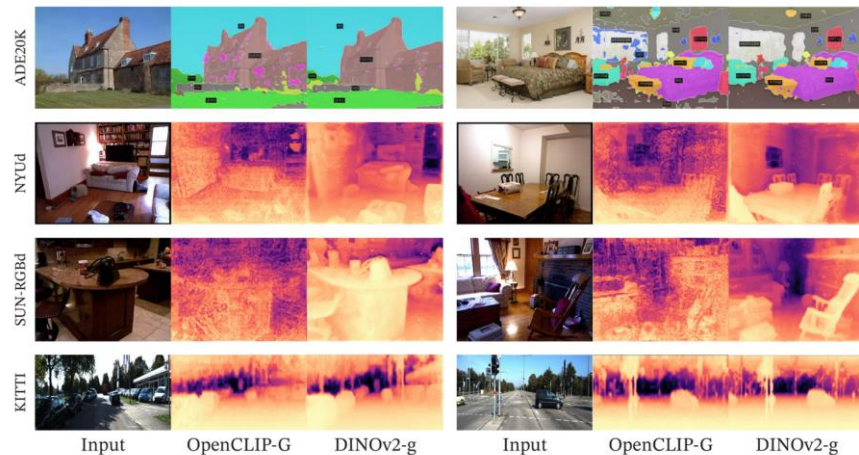
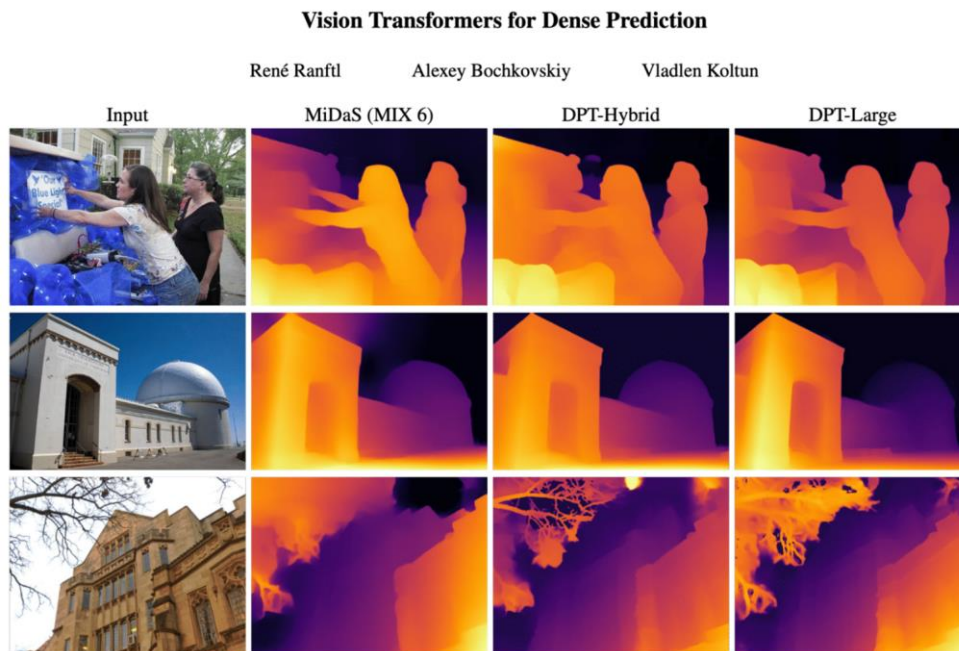


Figure 7: **Segmentation and depth estimation with linear classifiers.** Examples from ADE20K, NYUd, SUN RGB-D and KITTI with a linear probe on frozen OpenCLIP-G and DINOv2-g features.

# Monocular Depth Estimation

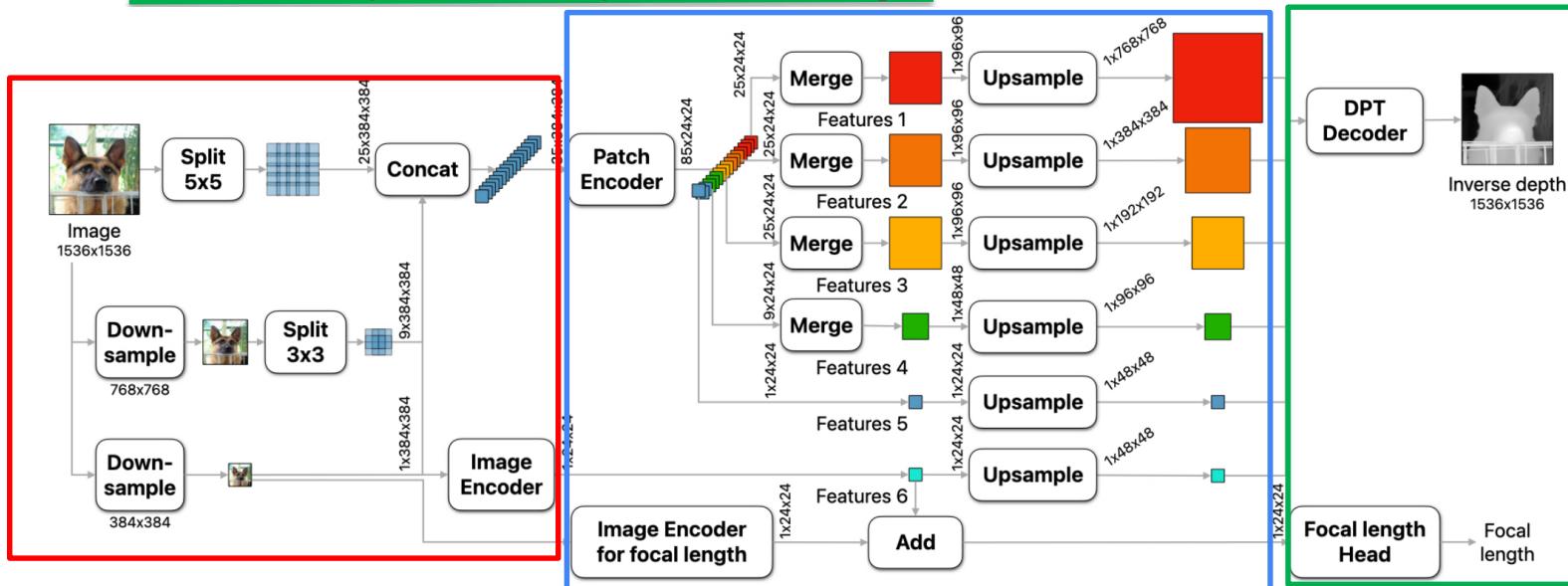
- Exploiting Feature Extraction Backbones



Vision Transformers for Dense Prediction. Ranftl et al. ICCV 2021

# Monocular Depth Estimation

- Depth Pro (ICLR 2025)
  - Key idea: 1) Extract patches at multiple scales 2) Extract features with ViT[1] encoder 3) decode the output inverse depth and focal length.



# Conclusions

# Conclusions

In this lecture we saw Stereo Matching, which consists of computing **disparity** maps given two reference image.

We saw how it is possible to compute depth from disparity maps, and reviewed both classical and recent methods for disparity map computation.

We then extended the problem formulation to Multiview Stereo Matching, and saw how to reconstruct a geometry from multiview images, and how to predict the depth from MVS.

We reviewed the problem of Monocular Depth Estimation, and analyzed some of the current state-of-the-art methods.

# References

- [http://vision.stanford.edu/teaching/cs131\\_fall1415/lectures/lecture9\\_10\\_stereo\\_cs131.pdf](http://vision.stanford.edu/teaching/cs131_fall1415/lectures/lecture9_10_stereo_cs131.pdf), Fei-Fei Li
- [https://cvgl.stanford.edu/teaching/cs231a\\_winter1415/lecture/lecture6\\_affine\\_SFM\\_notes.pdf](https://cvgl.stanford.edu/teaching/cs231a_winter1415/lecture/lecture6_affine_SFM_notes.pdf), Silvio Savarese
- <https://www.slideshare.net/slideshow/lec14-multiview-stereo/251201912>, Bali Thorat
- <https://www.slideshare.net/slideshow/lec13-stereo-converted/251201880>, Bali Thorat
- [https://drive.google.com/file/d/1QCDEM5l-wi7VLI4b8cw\\_ErmCAyJOxb10/view](https://drive.google.com/file/d/1QCDEM5l-wi7VLI4b8cw_ErmCAyJOxb10/view), Andreas Geiger
- <https://learning3d.github.io/>, Gkioulekas and Tulsiani

The End