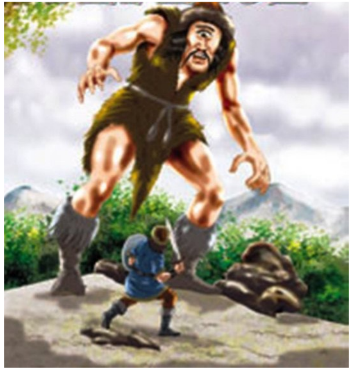
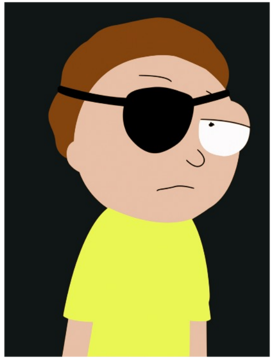
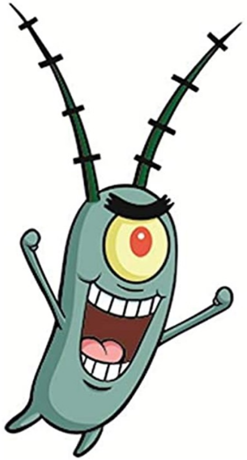
Learning Stereo

Nail Ibrahimli

What are these characters having in common?





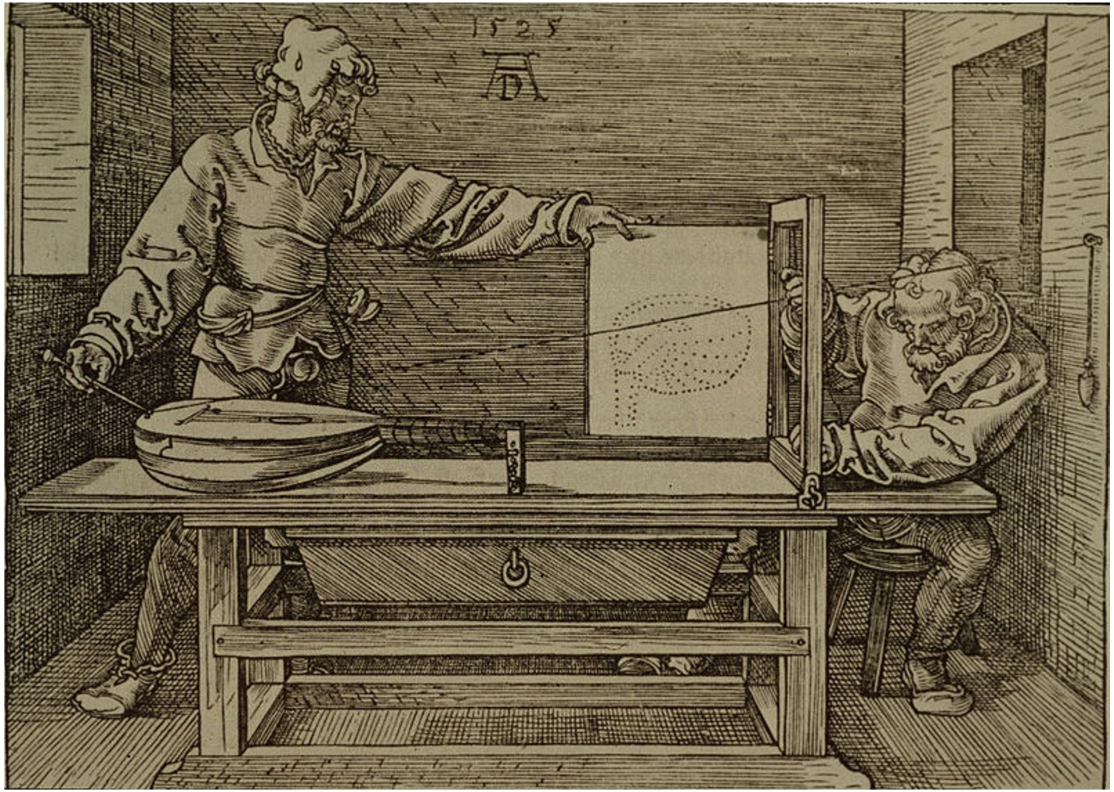
Cyclope
(Greek)

Hitotsume-kozō
(Japanese)

Tepegoz
(Turkic)

Eye of Sauron
(LOTR)

Imaging geometry of single eye (camera)



Albrecht Durer (Pinhole)

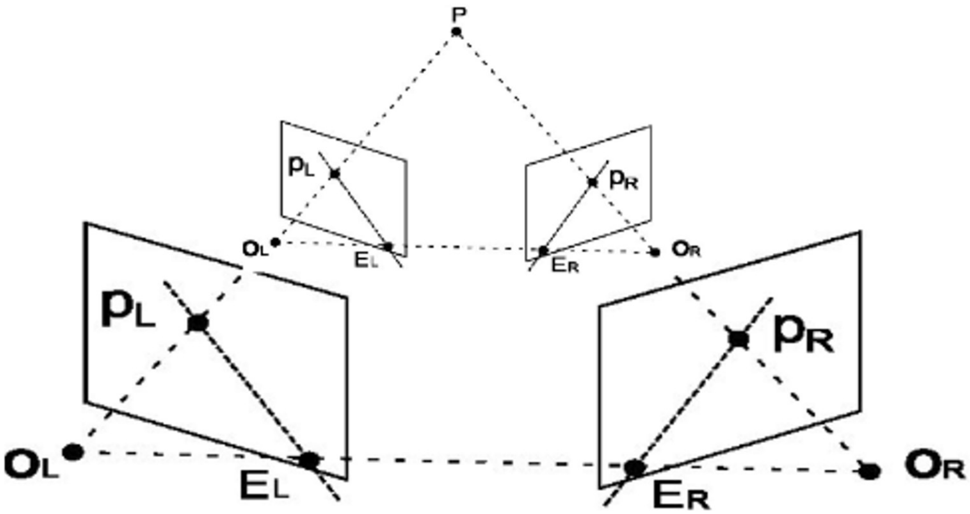
M.C. Escher(Omnidirectional)

Limitations of single eye

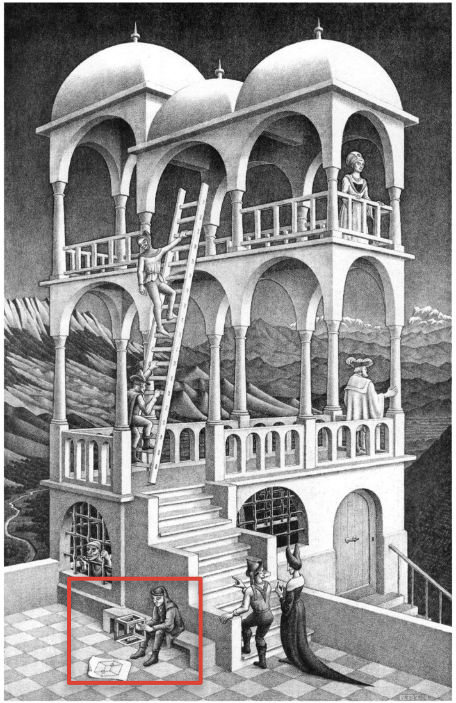
M.C. Escher

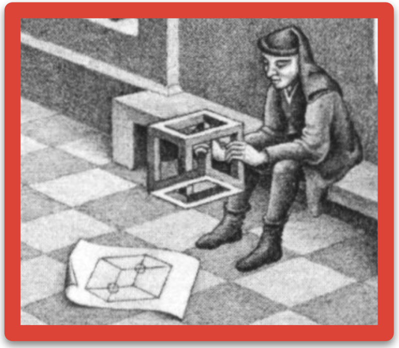
Limitations of single eye

M.C. Escher



Limitations of single eye





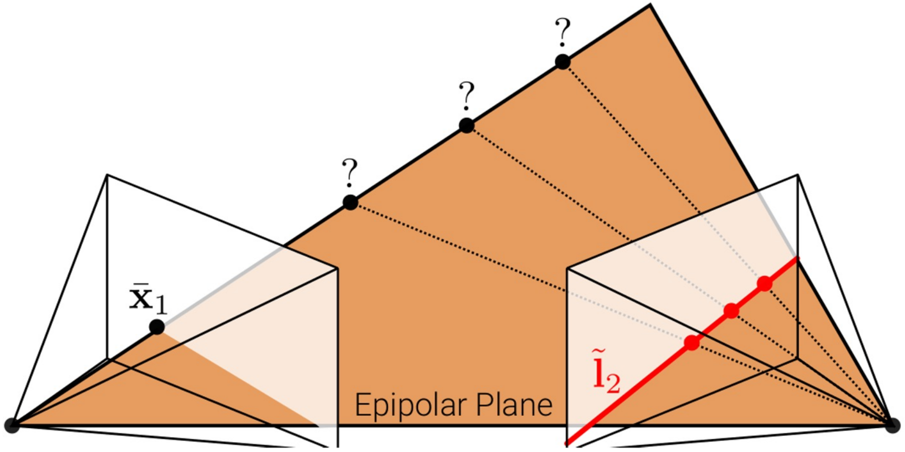
M.C. Escher

Internet

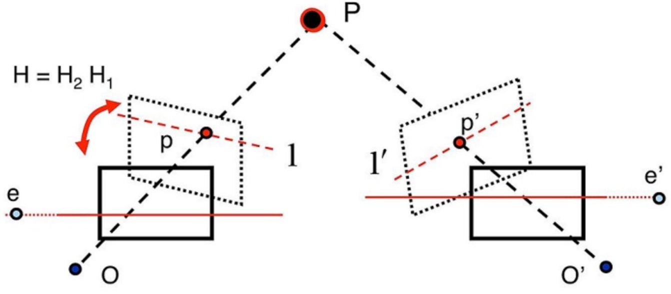
Why do we have two eyes?

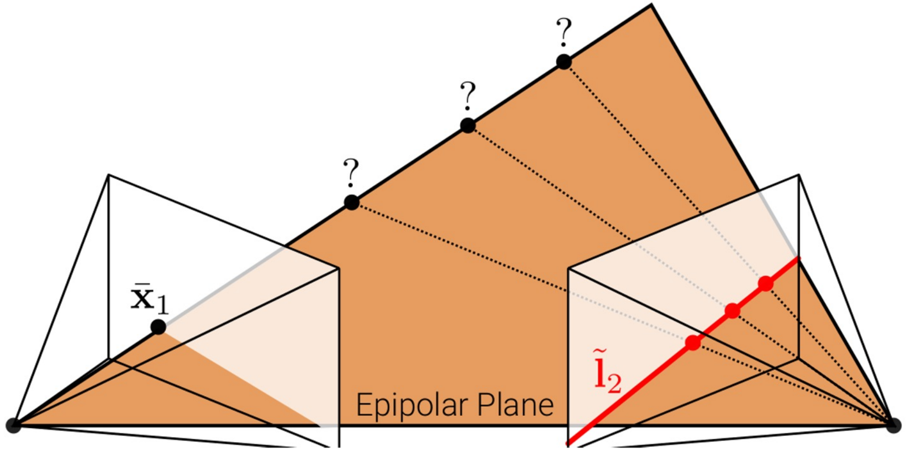
Pan’s Labyrinth

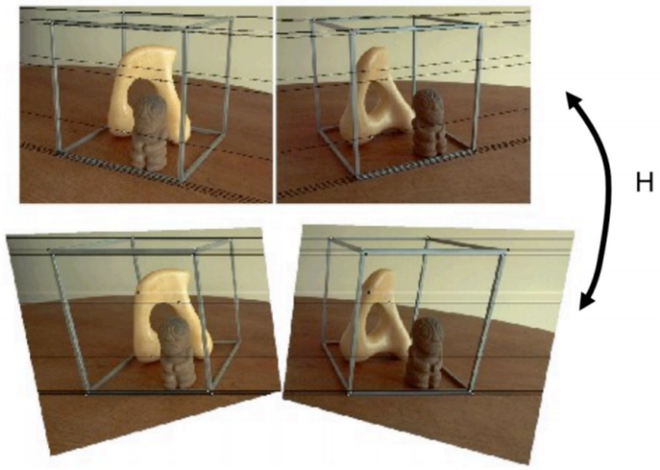
Why do we have two eyes?



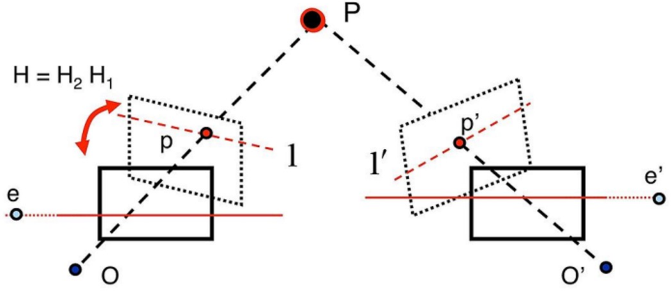
Why do we have two eyes?

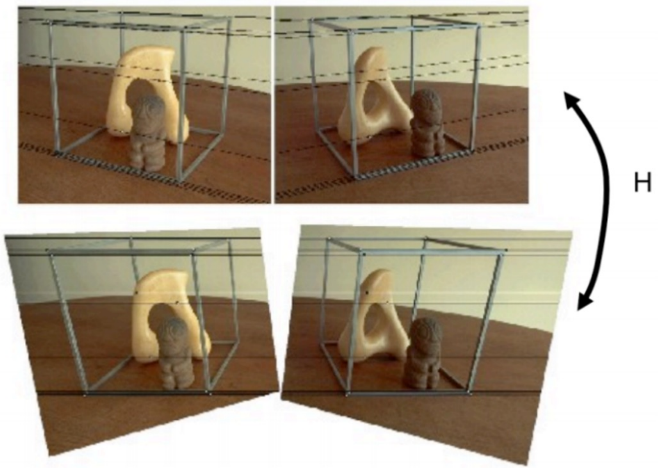


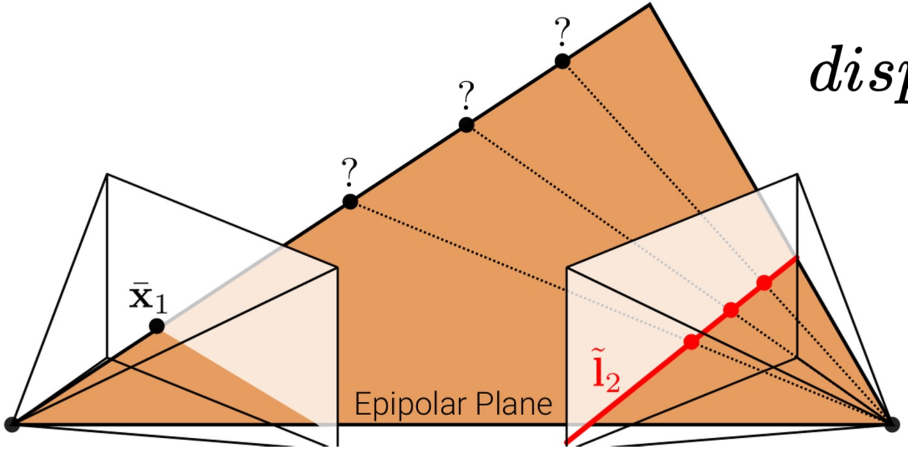




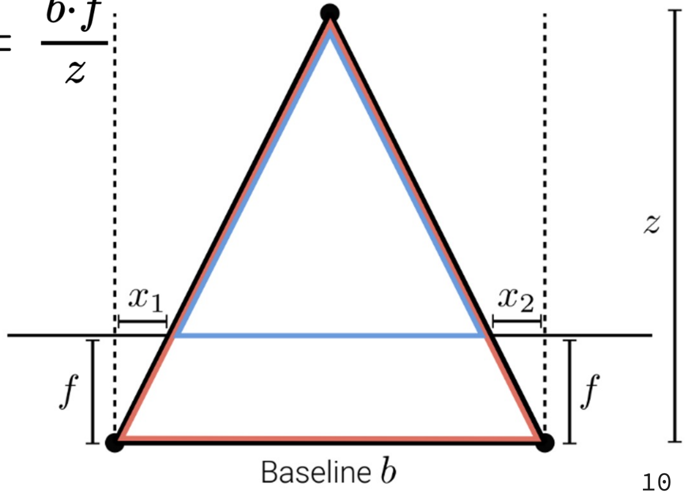
Why do we have two eyes?





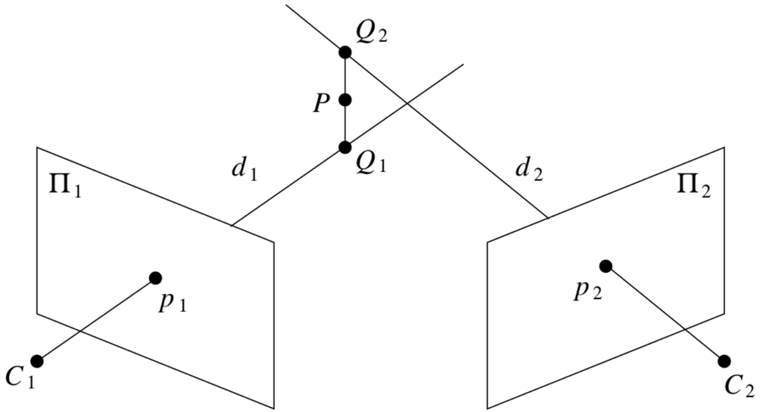


$$disparity = \frac{b \cdot f}{z}$$



Slide credit: Fei-fei Li, Andreas Geiger

Triangulation



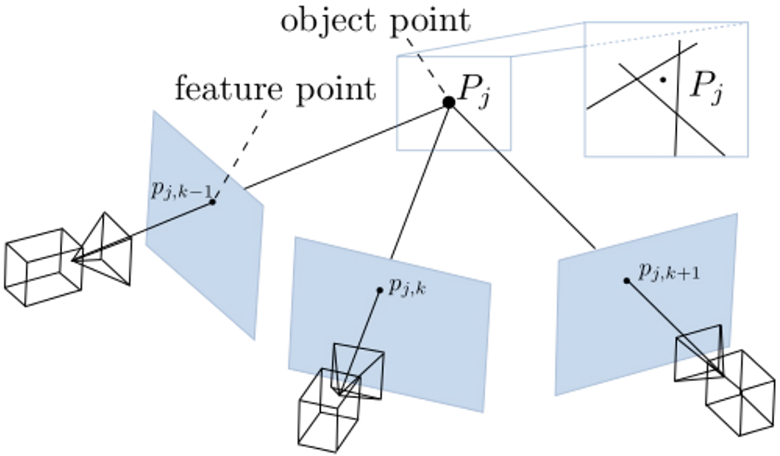
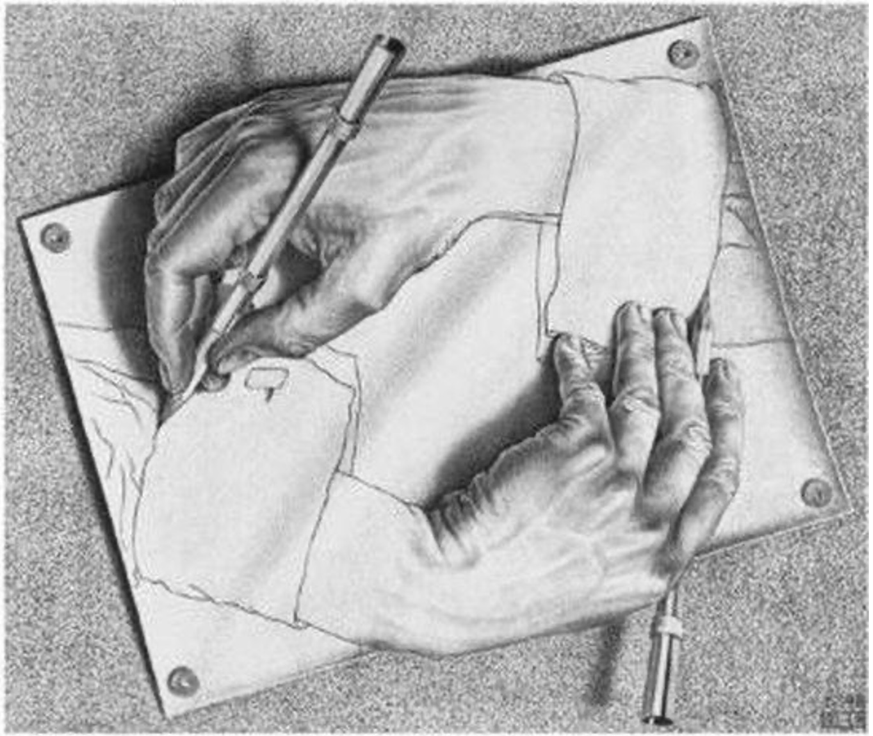


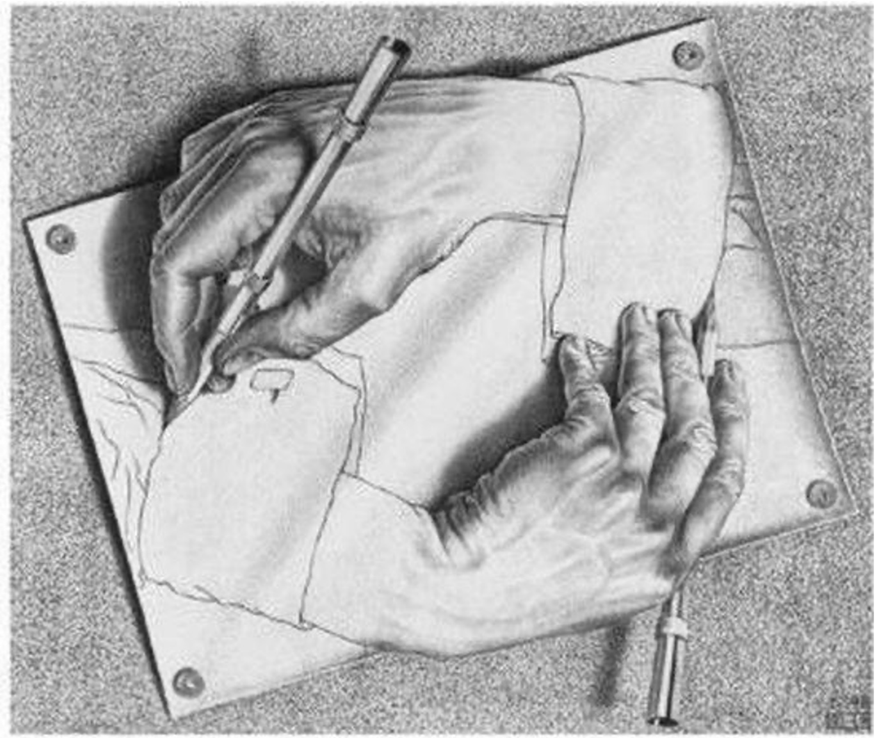
Image credit: OpenMVG

Visual cues for 3D: Shading

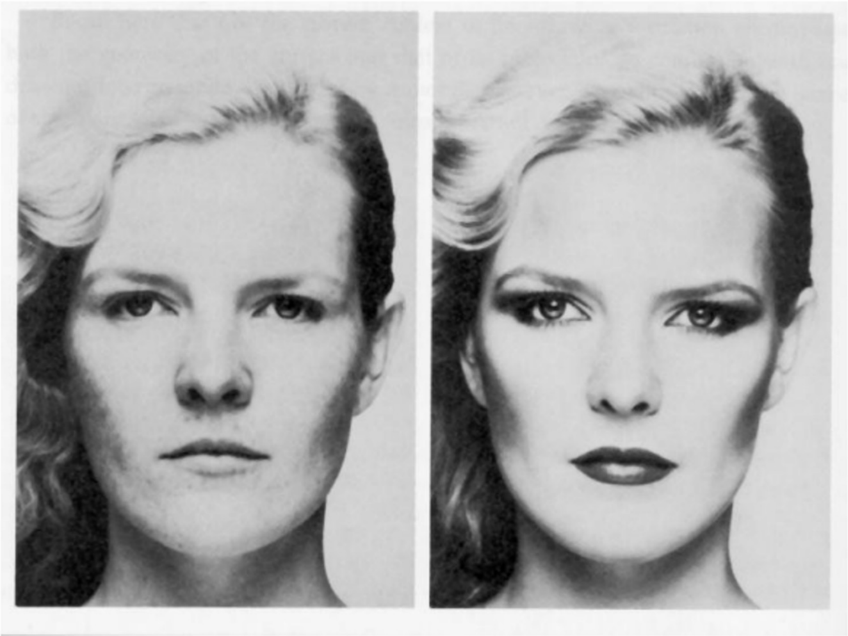


M.C. Escher

Visual cues for 3D: Shading

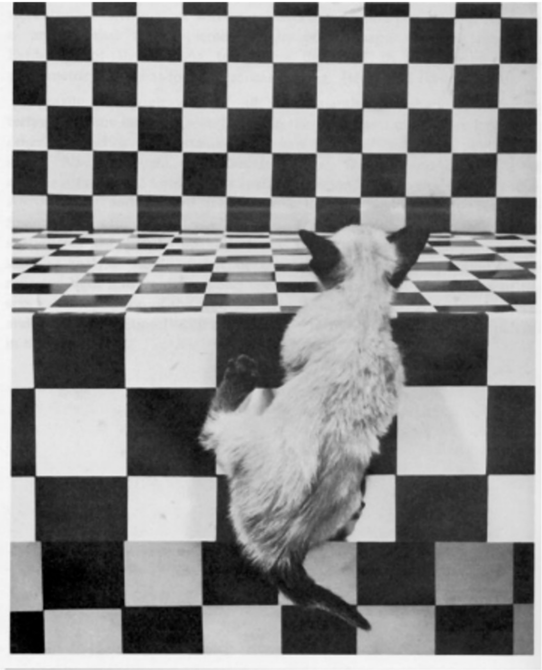


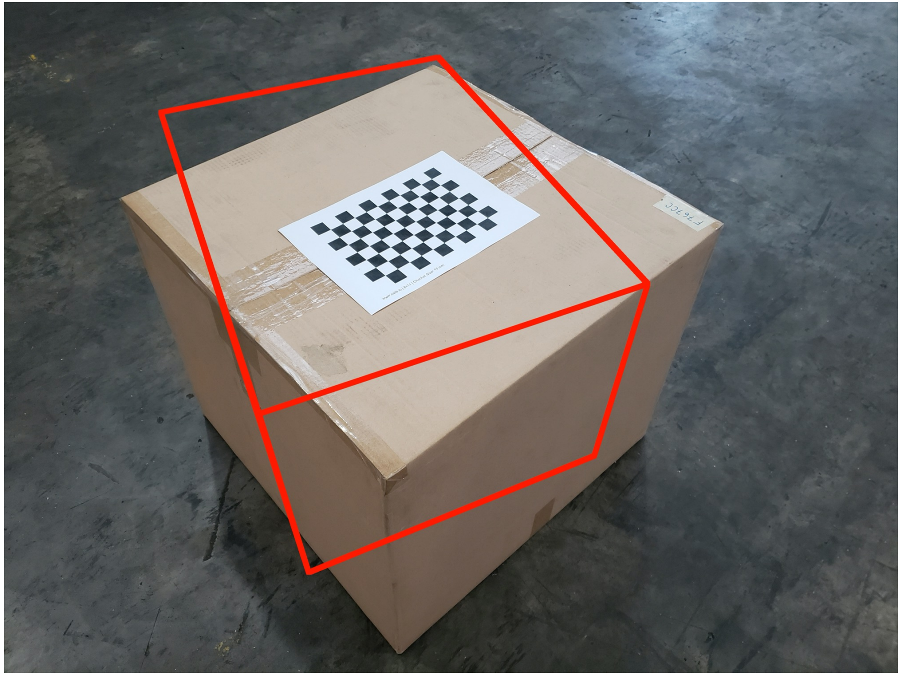
M.C. Escher



Merle Norman Cosmetics

Visual cues for 3D: Texture





The Visual Cliff by William Vandivert

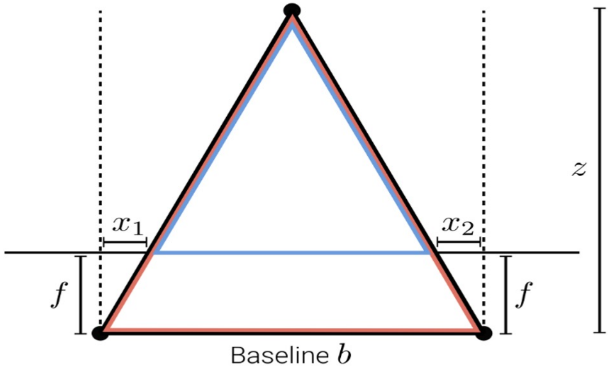
Visual cues for 3D: Focus, Motion

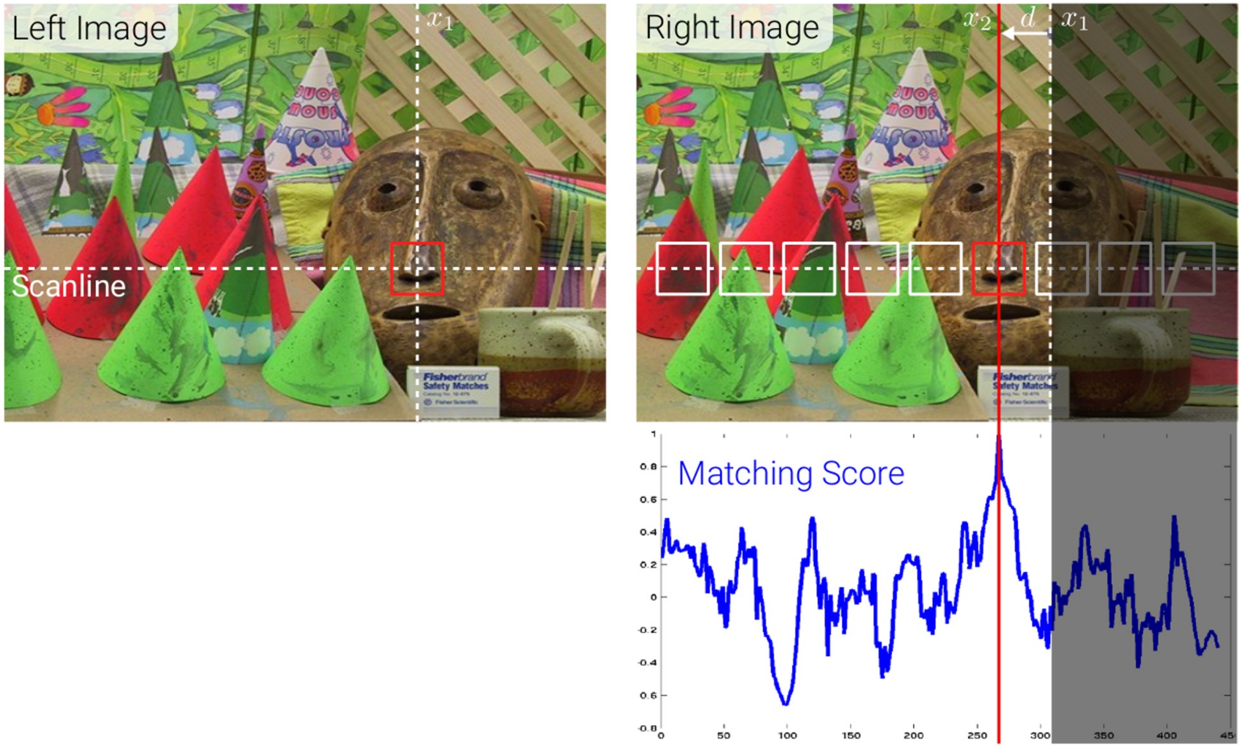
From Art of the Photography

Slide credit: James Hays

Stereo matching

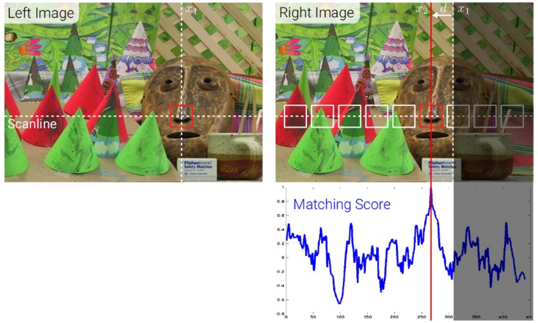
$$disparity = \frac{b \cdot f}{z}$$





Slide credit: Andreas Geiger

Block matching



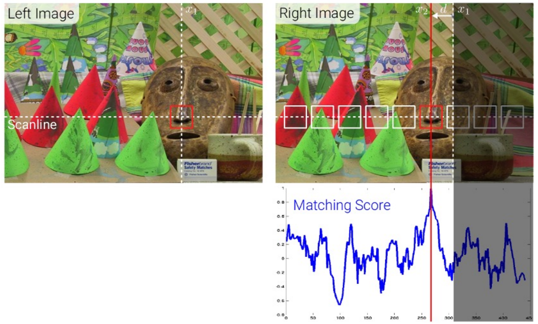
$$SSD = \sum\sum\left(I_{left} - I_{right}\right)^2$$ Sum of squares difference

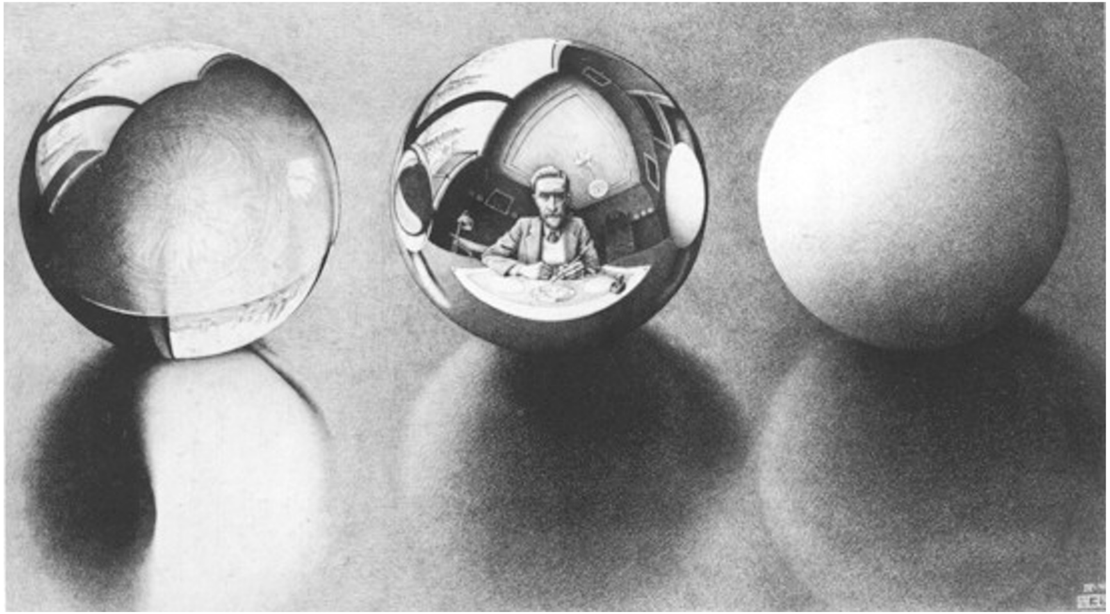
$$AD = \sum\sum\left|\left(I_{left} - I_{right}\right)\right|$$ Absolute difference

$$CC = \sum\sum I_{left} I_{right}$$ Cross correlation

$$NC = \frac{\sum\sum\left(I_{left} . I_{right}\right)}{\sqrt{\sum\sum I_{left} . I_{right}}}$$ Normalized Correlation

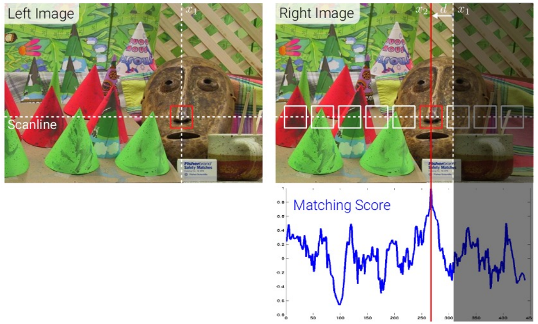
Block matching (Failure cases)

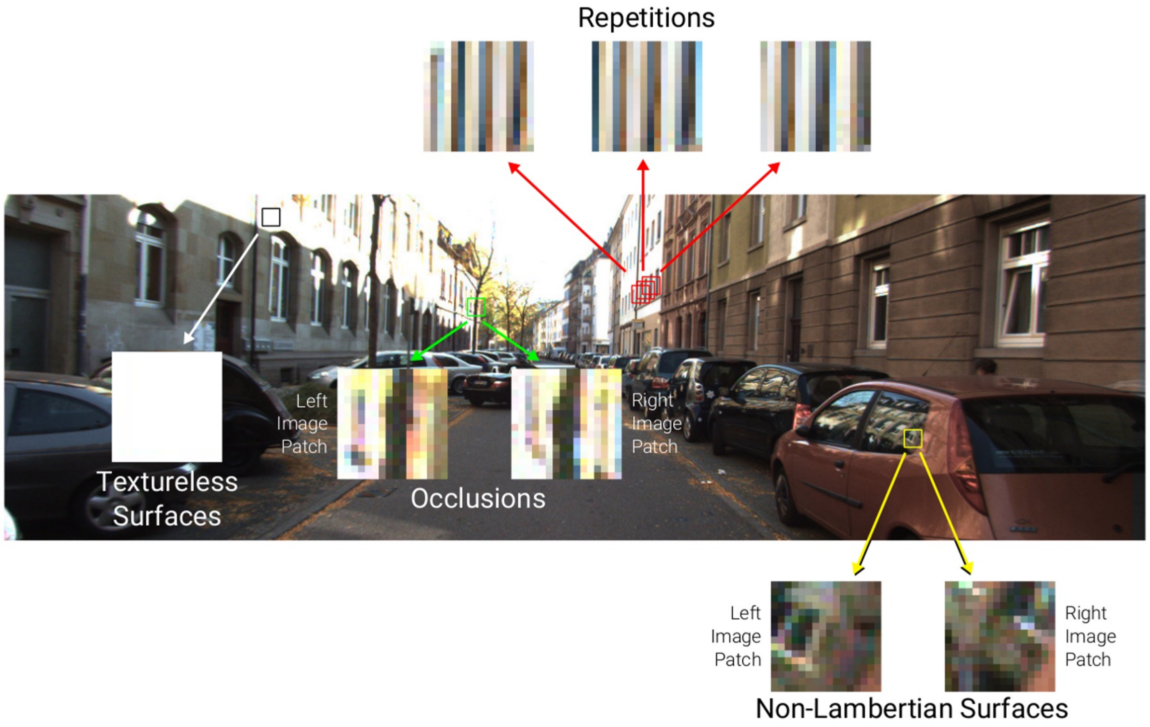




M.C. Escher

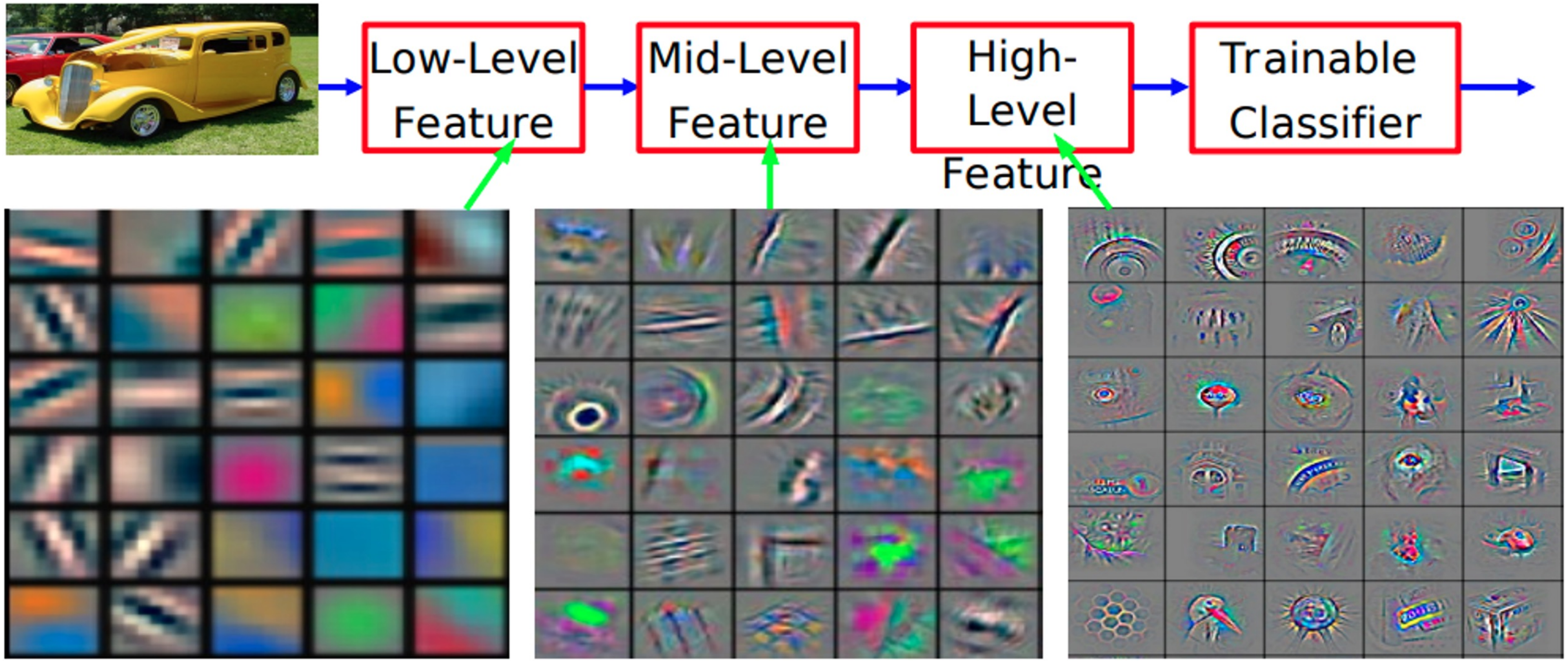
Block matching (Failure cases)





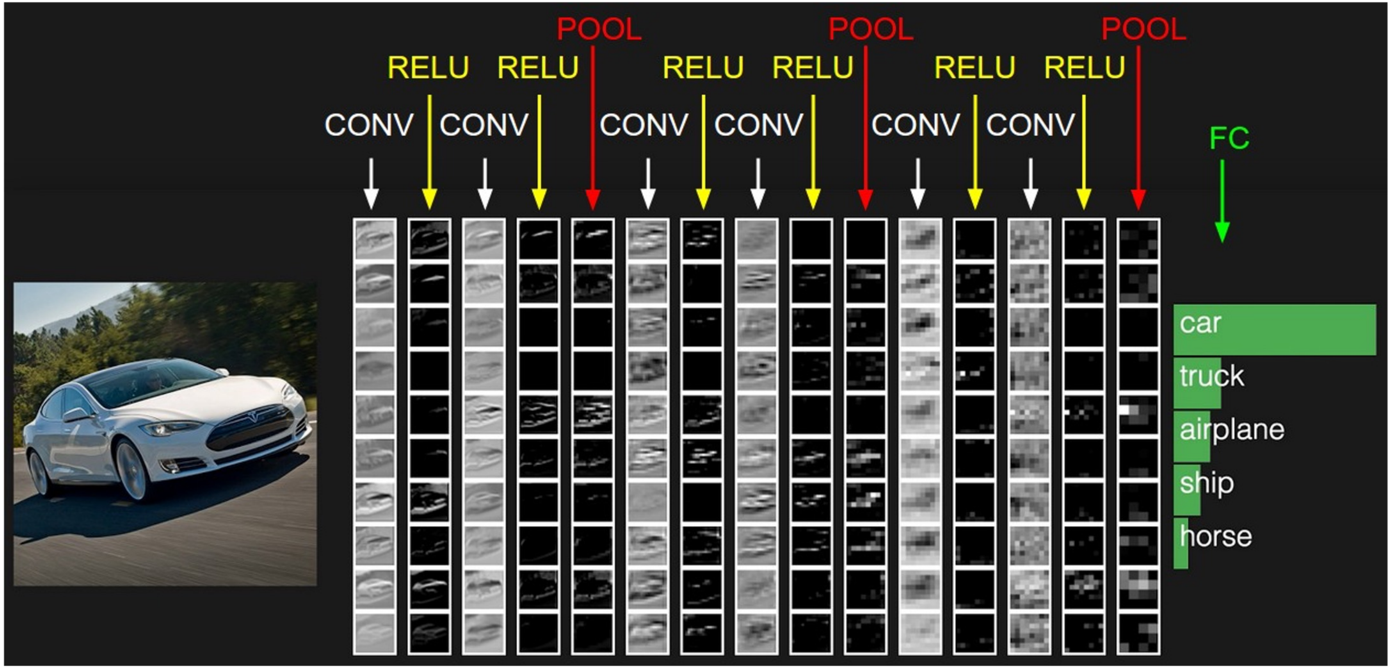
Slide credit: Andreas Geiger

Convolutional features



Slide credit: Yann Lecun
Image credit: Visualizing and Understanding Convolutional Networks (Zeiler & Fergus, 2013)

Convolutional features



Slide credit: Andrej Karpathy

Image convolution

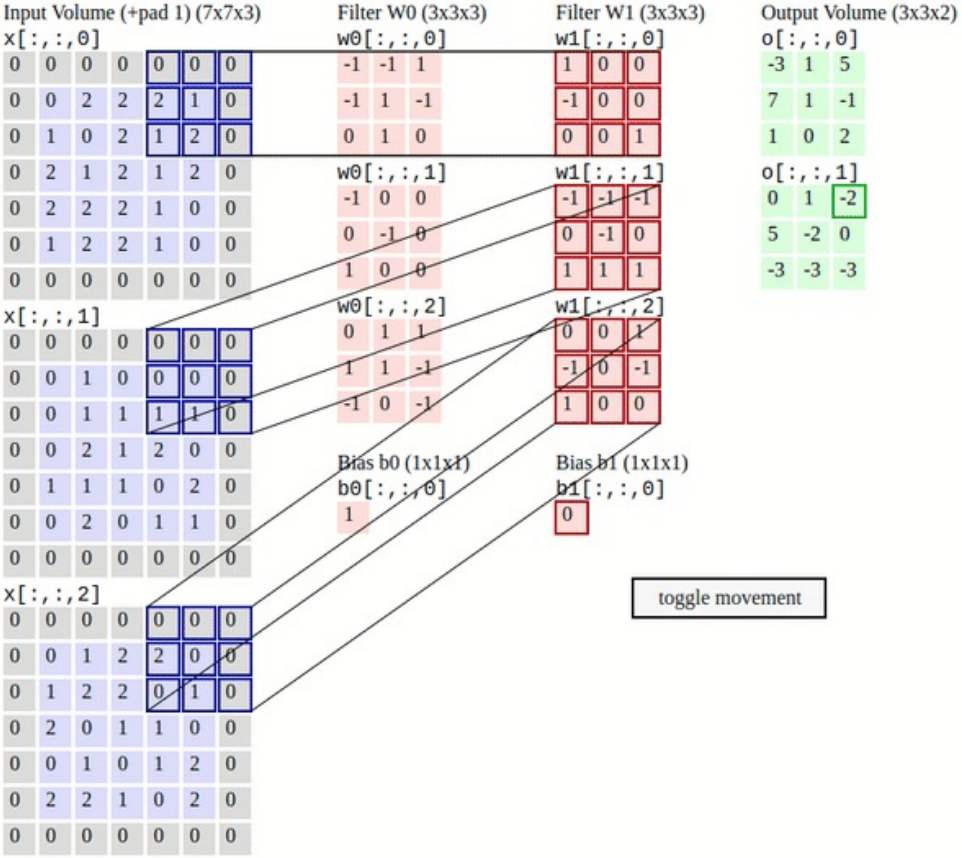


Image credit: Andrej Karpathy

2D and 3D convolutions

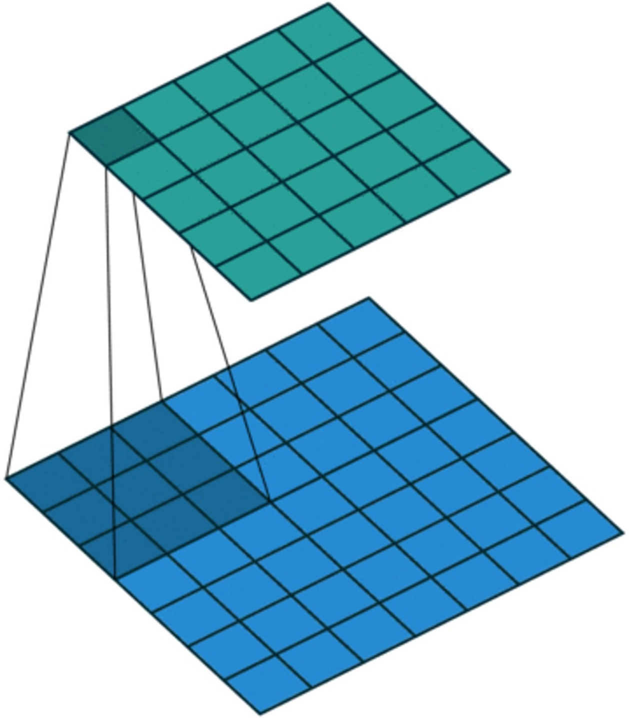
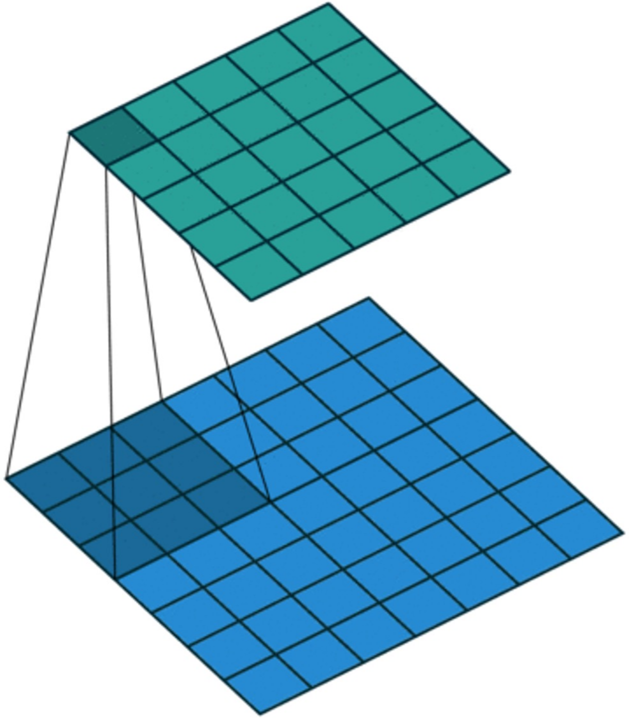


Image credit: https://iamaaditya.github.io/2016/03/one-by-one-convolution/

2D and 3D convolutions



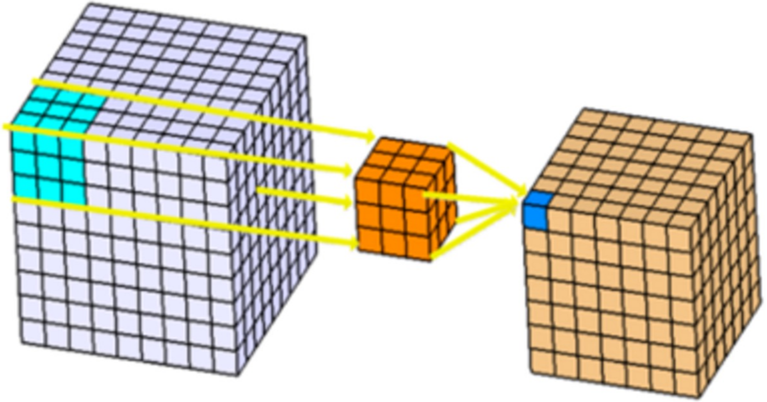
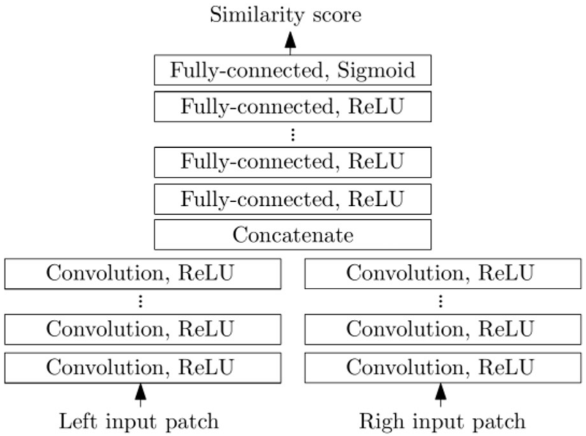


Image credit: https://biplabbarman097.medium.com/3d-convolutions-and-its-applications-6dd2d0e9e63f

Block matching

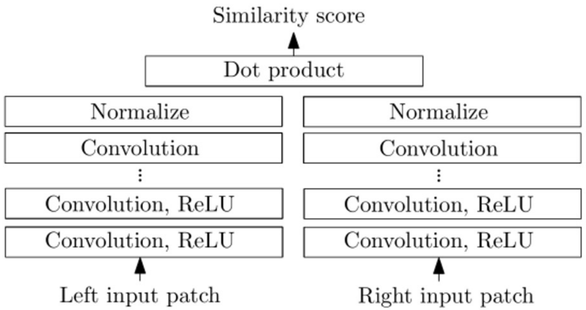
Learned Similarity:

- Learn features & sim. metric
- Potentially more expressive
- Slow (WxHxD MLP evaluations)



Cosine Similarity:

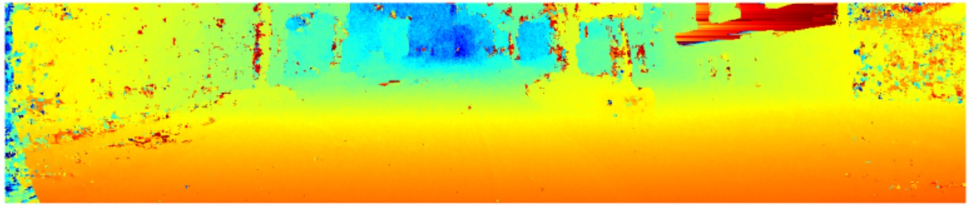
- Learn features & apply dot-product
- Features must do the heavy lifting
- Fast matching (no network eval.)



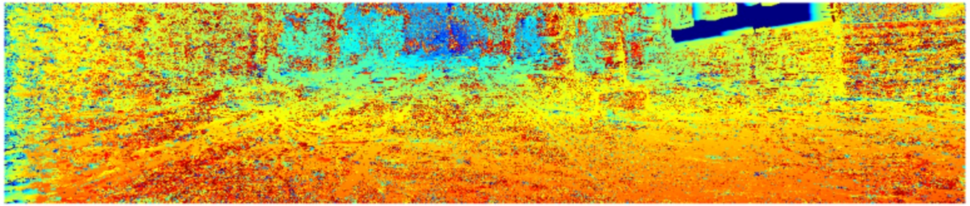
Zbontar and LeCun: Stereo Matching by Training a Convolutional Neural Network to Compare Image Patches. JMLR, 2016.

Block matching

Left Input Image



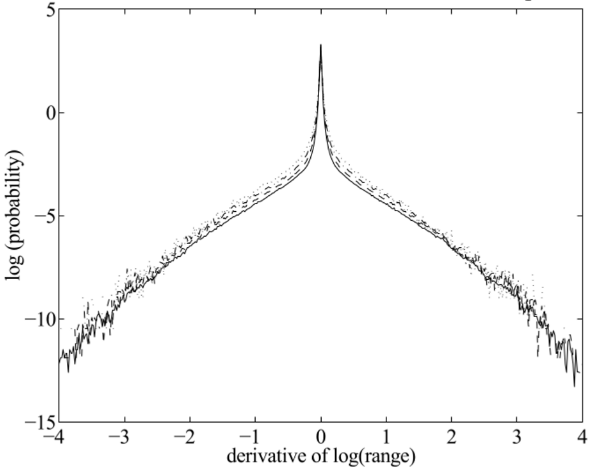
Siamese Network



Standard Block Matching

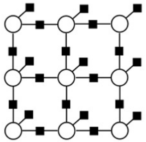
Zbontar and LeCun: Stereo Matching by Training a Convolutional Neural Network to Compare Image Patches. JMLR, 2016.

Block matching



Huang, Lee and Mumford: Statistics of Range Images. CVPR, 2000.

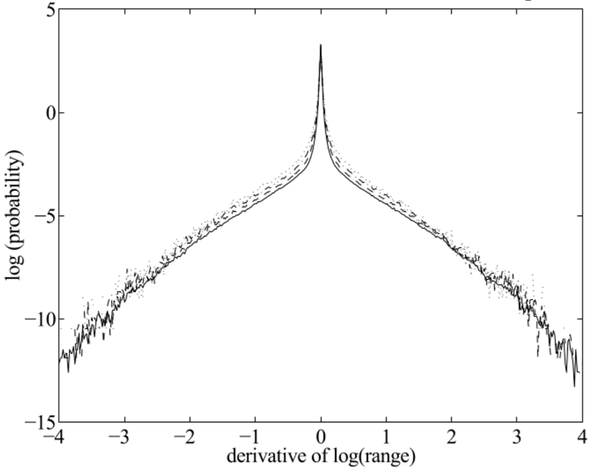
$$p(\mathbf{D}) \propto \exp\left\{-\sum_i \psi_{data}(d_i) - \lambda \sum_{i \sim j} \psi_{smooth}(d_i, d_j)\right\}$$



Y. Boykov, O. Veksler, and R. Zabih, “Fast approximate energy minimization via graph cuts”. PAMI(1999)

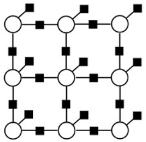
Zbontar and LeCun: Stereo Matching by Training a Convolutional Neural Network to Compare Image Patches. JMLR, 2016.

Block matching



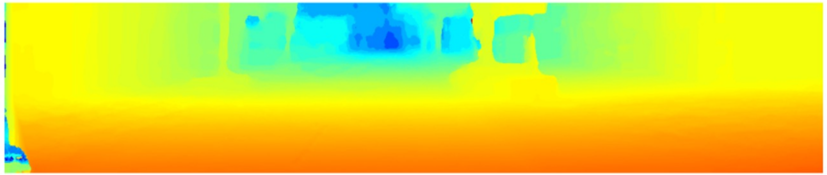
Huang, Lee and Mumford: Statistics of Range Images. CVPR, 2000.

$$p(\mathbf{D}) \propto \exp\left\{ -\sum_i \psi_{data}(d_i) - \lambda \sum_{i\sim j} \psi_{smooth}(d_i, d_j) \right\}$$

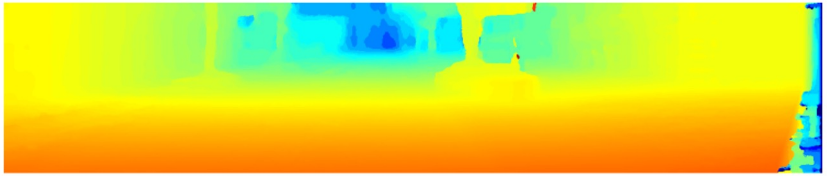


Y. Boykov, O. Veksler, and R. Zabih, “Fast approximate energy minimization via graph cuts”. PAMI(1999)

Semi-Global Matching Algorithm



Left Disparity Map

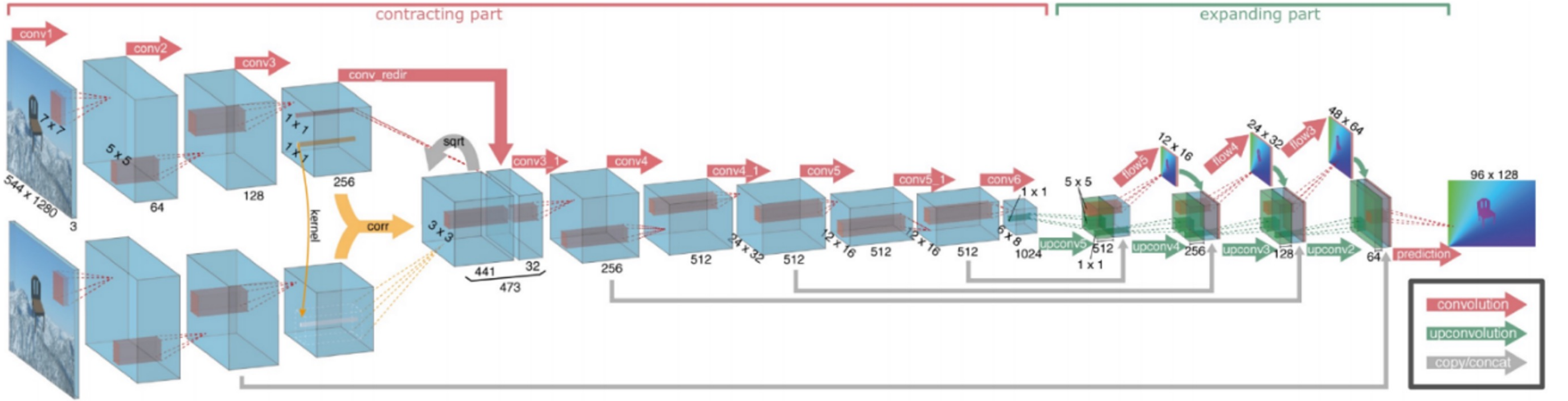


Right Disparity Map

Left-Right Consistency Test

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

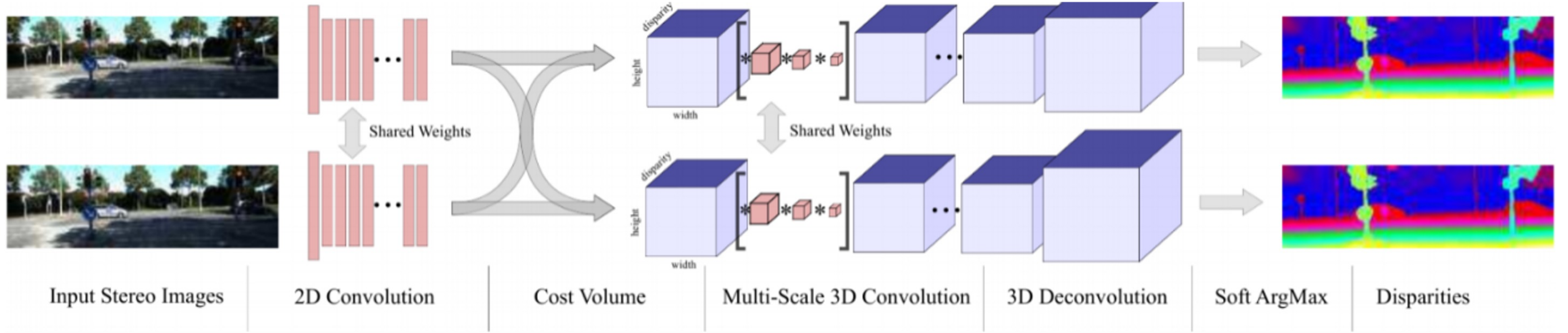
DISPNET



- DispNet was one of the first end-to-end trained deep neural network for stereo disparity
- It used a U-Net like architecture with skip-Connections to retain details
- It introduces correlation layer
- Multi-scale loss (disparity error in pixels), curriculum learning (easy-to-hard)

Mayer et al.: A Large Dataset to Train Convolutional Networks for Disparity, Optical Flow, and Scene Flow Estimation. CVPR, 2016.

GC-net

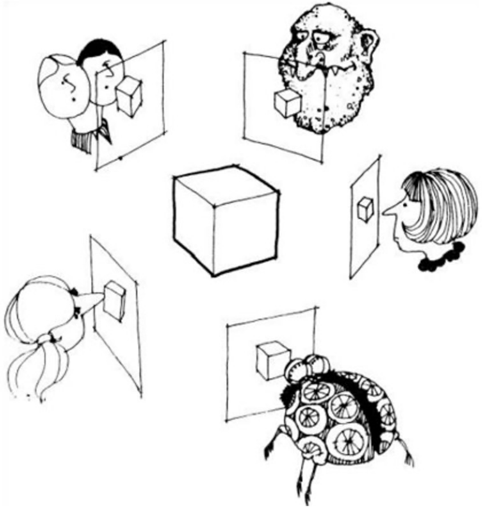


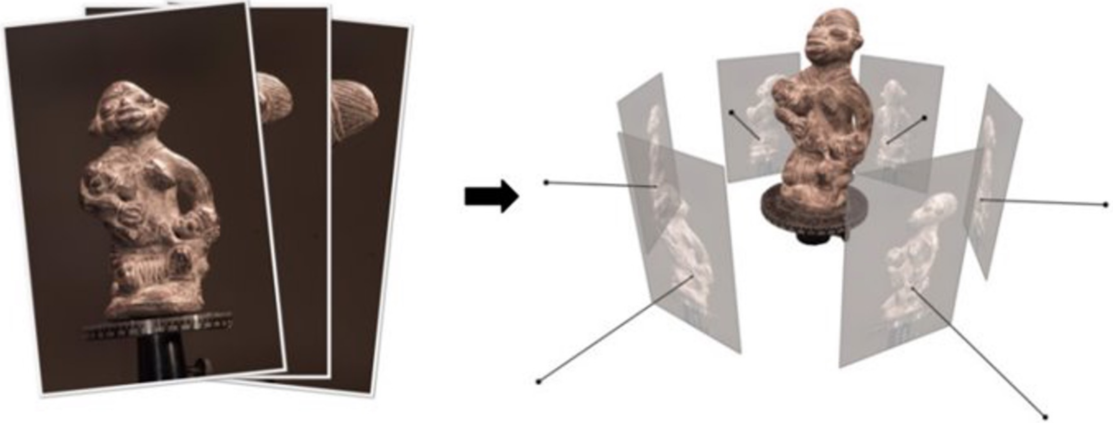
$$d^* = \mathbb{E}[d] = \sum_{d=0}^{D} \underbrace{\mathrm{softmax}(-c_\theta(d))}_{p(d)} \cdot d$$

- Key idea: calculate disparity cost volume and apply 3D convolutions on it
- Convert the learned matching cost c to disparity via the expectation(probability volume)
- Slightly better performance but large memory requirements (3D feature volume)

Kendall, Martirosyan, Dasgupta and Henry: End-to-End Learning of Geometry and Context for Deep Stereo Regression. ICCV, 2017.

Multi-view stereo





MVS Goal: To find a 3D shape that explains the images.

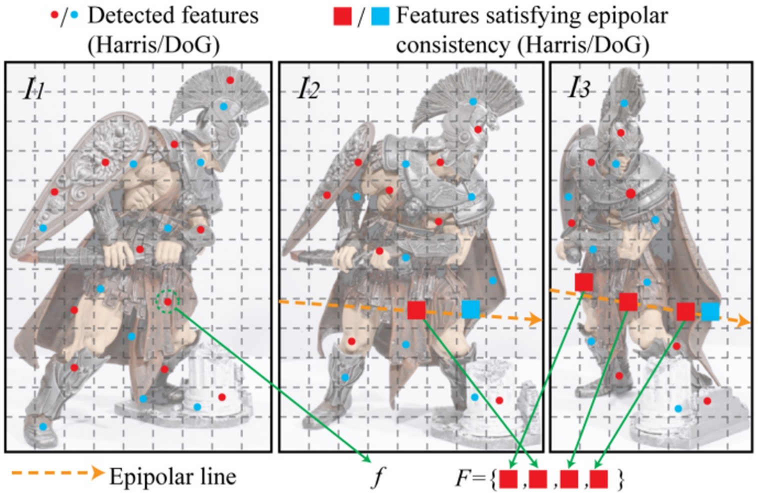
Image credit: Svetlana Lazebnik
Yasutaka Furukawa Carlos Hernandez: Multi-View Stereo: A Tutorial

PMVS in one slide :)

1. Detect keypoints
2. Triangulate a sparse set of initial matches
3. Iteratively expand matches to nearby locations
4. Use visibility constraints to filter out false matches
5. Perform surface reconstruction

Accurate, Dense, and Robust Multi-View Stereopsis CVPR07, Yasutaka Furukawa, Jean Ponce

Feature Detection

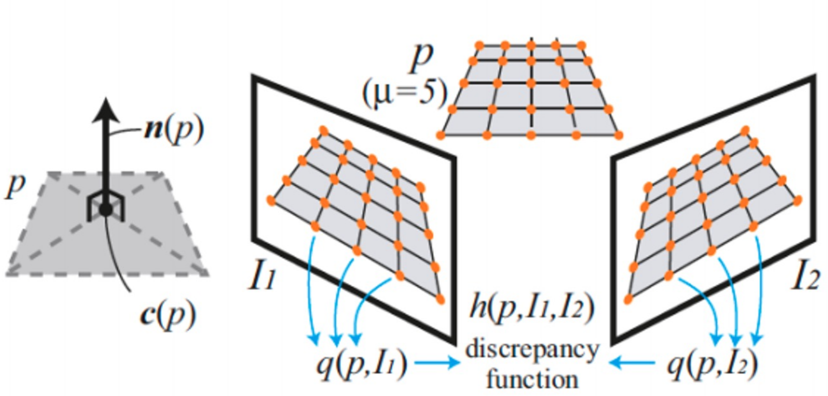


1. Divide grid to cells (32x32)

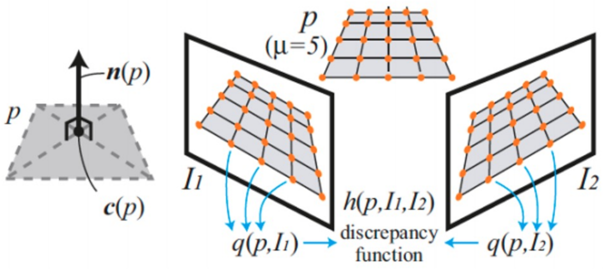
2. Use Harris Detector and DoG to find corners

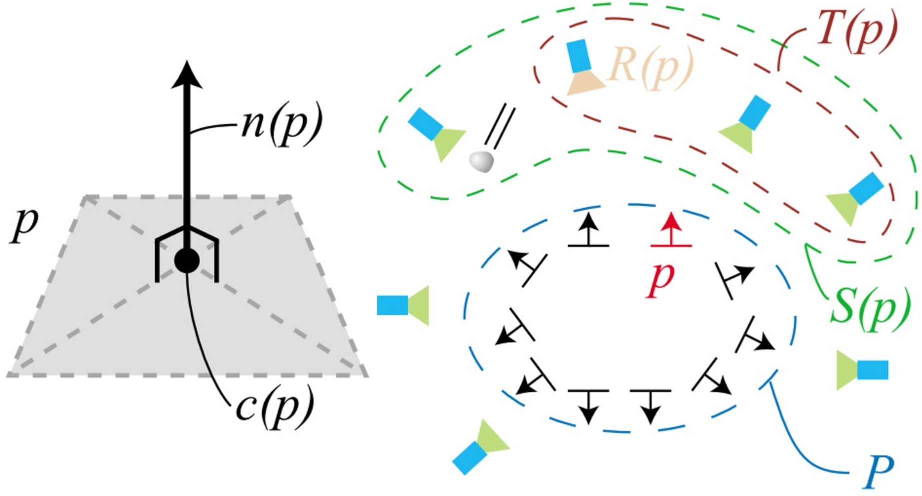
3. Try to find 4 good corners in each cell (uniform overage)

Patch Geometry



Patch Model

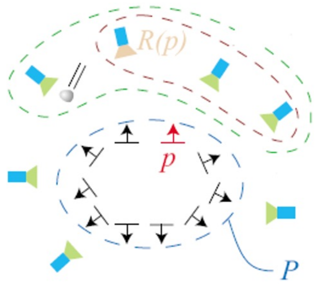




$$\mathbf{c}(p) \leftarrow \{\text{Triangulation from } f \text{ and } f'\},$$
$$\mathbf{n}(p) \leftarrow \overrightarrow{\mathbf{c}(p)O(I_i)} / |\overrightarrow{\mathbf{c}(p)O(I_i)}|,$$
$$R(p) \leftarrow I_i.$$

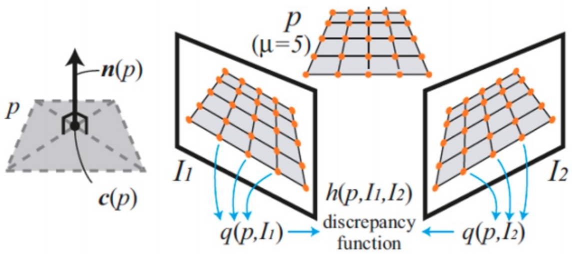
c(p): center of the patch
n(p): normal of the patch
R(p): reference image with p

Photometric Discrepancy Function



$$h(p, I, R(p)) = 1 - NCC(p, I, R(p))$$

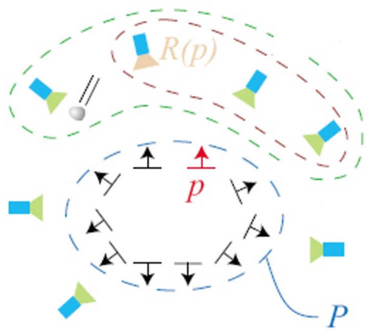
$$g(p) = \frac{1}{|V(p) \setminus R(p)|} \sum_{I \in V(p) \setminus R(p)} h(p, I, R(p))$$



V(p): initial set of images where patch p is potentially visible

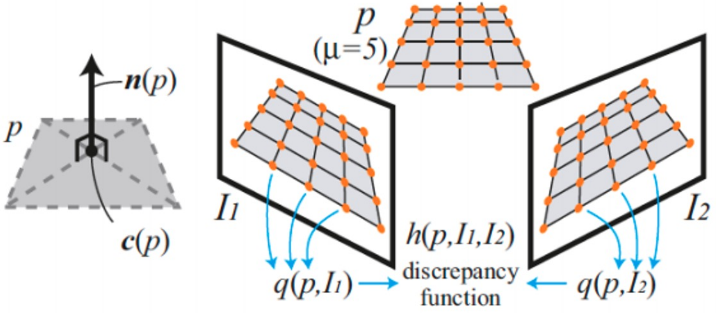
Photometric Discrepancy Function

$$V^*(p) = \{I \mid I \in V(p), h(p, I, R(p)) \leq \alpha\},$$
$$g^*(p) = \frac{1}{|V^*(p) \setminus R(p)|} \sum_{I \in V^*(p) \setminus R(p)} h(p, I, R(p)).$$



V(p): set of images where patch is truly visible

Patch optimization



$$h(p, I, R(p)) = 1 - NCC(p, I, R(p))$$

$$g^*(p) = \frac{1}{|V^*(p) \setminus R(p)|} \sum_{I \in V^*(p) \setminus R(p)} h(p, I, R(p))$$

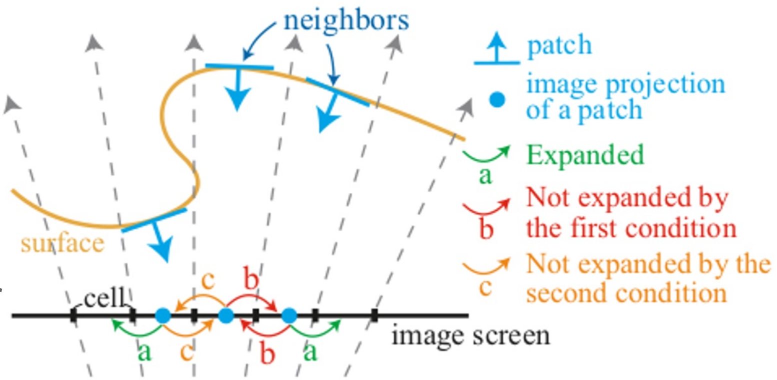
Optimize over c(p) and n(p) that minimizes g*(p)

Expansion

1. Identify neighboring cells for possible expansion
2. Test if there is already patch very close to that region
3. Test for depth discontinuity

Filtering

1. Photometric consistency filter
2. Geometric consistency filter
3. Occlusion check

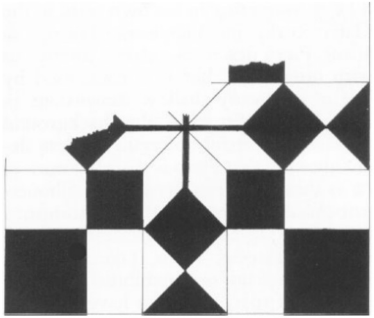


VisualSFM+PMVS

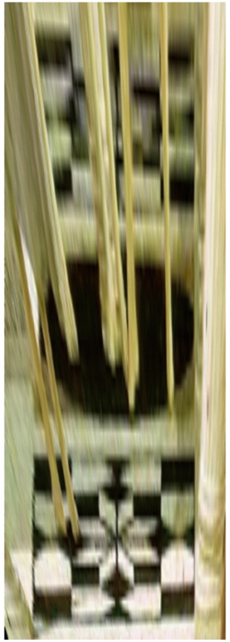
Differential homography

Flaggelation
Piero della Francesca

a



b



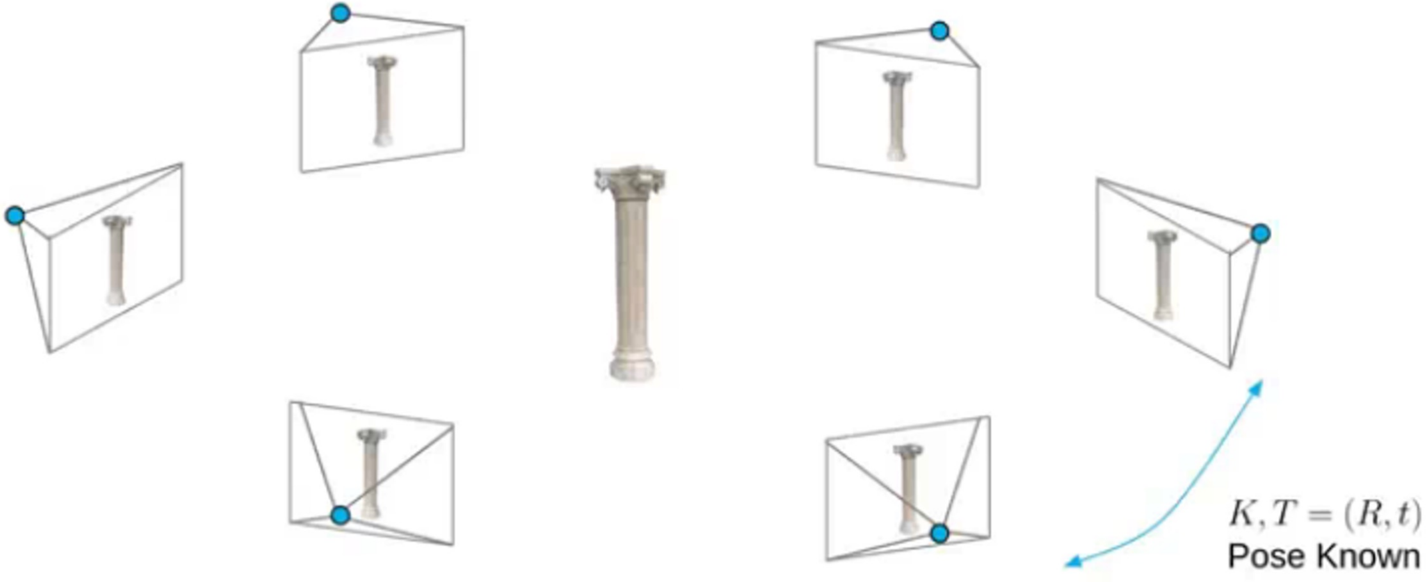
c

a) crop b) manual labeling c)homography

$$\mathbf{p}_{i,j} = \mathbf{K}_i \cdot (\mathbf{R}_{0,i} \cdot (\mathbf{K}_0^{-1} \cdot \mathbf{p} \cdot d_j) + \mathbf{t}_{0,i})$$

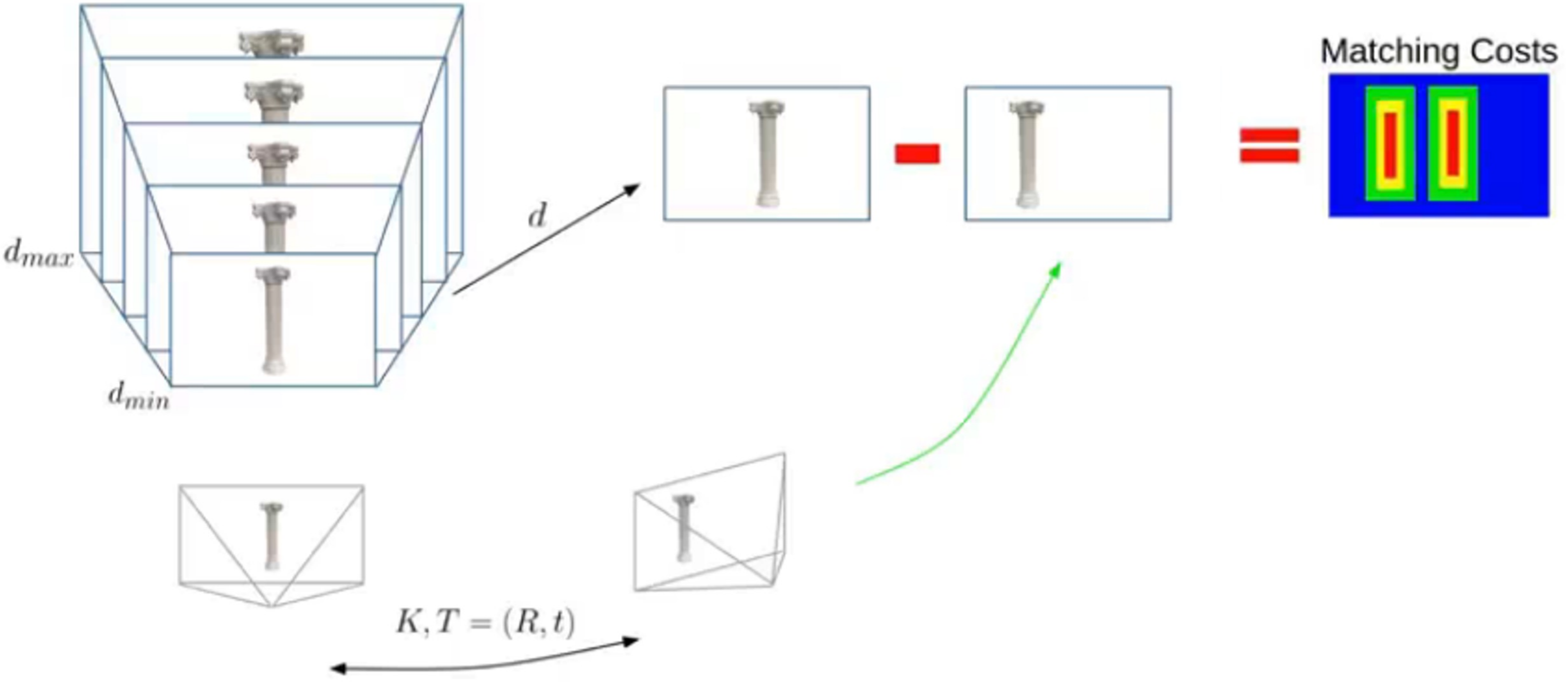
Criminisi et al. Bringing Pictorial Space to Life: Computer Techniques for the Analysis of Paintings. 2002.

Multi-view stereo - plane sweep stereo



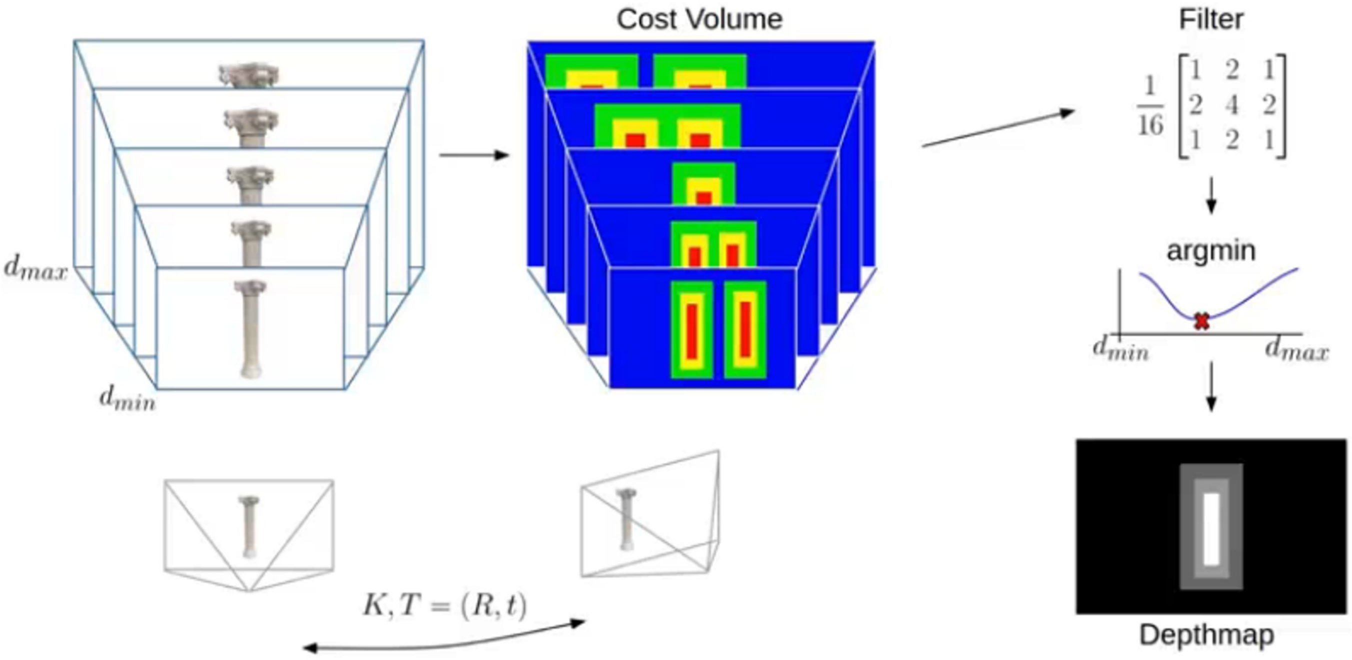
Slide credit: W. Nicholas Greene

Multi-view stereo - plane sweep stereo

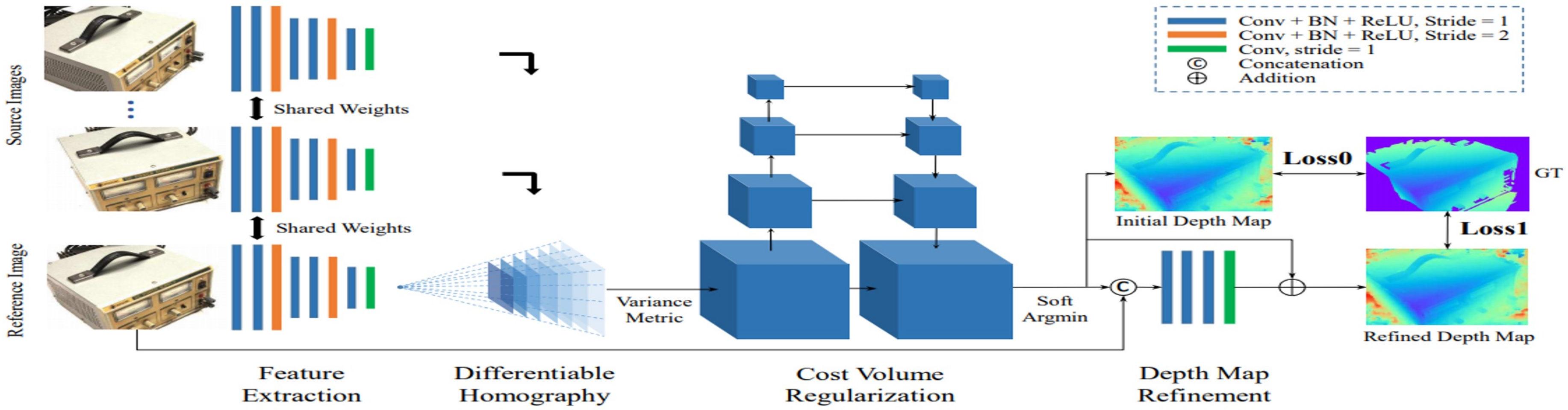


Slide credit: W. Nicholas Greene

Multi-view stereo - plane sweep stereo

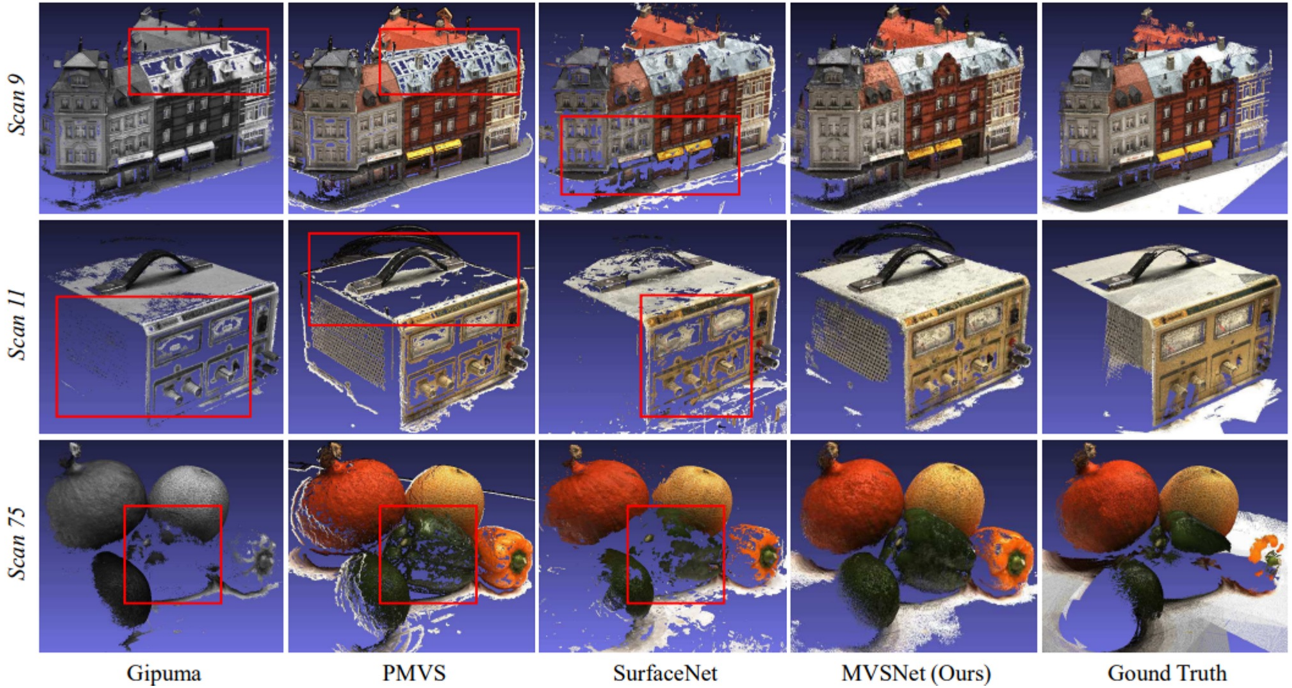


MVSNET



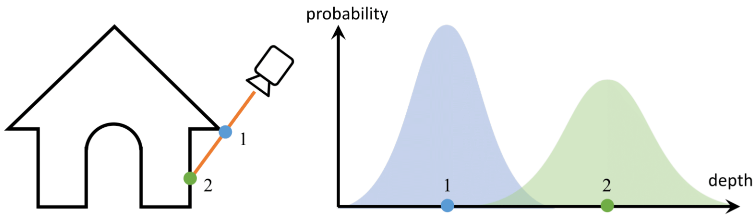
Yao Yao et. al.: MVSNet: Depth Inference for Unstructured Multi-view Stereo. ECCV 2018

MVSNET

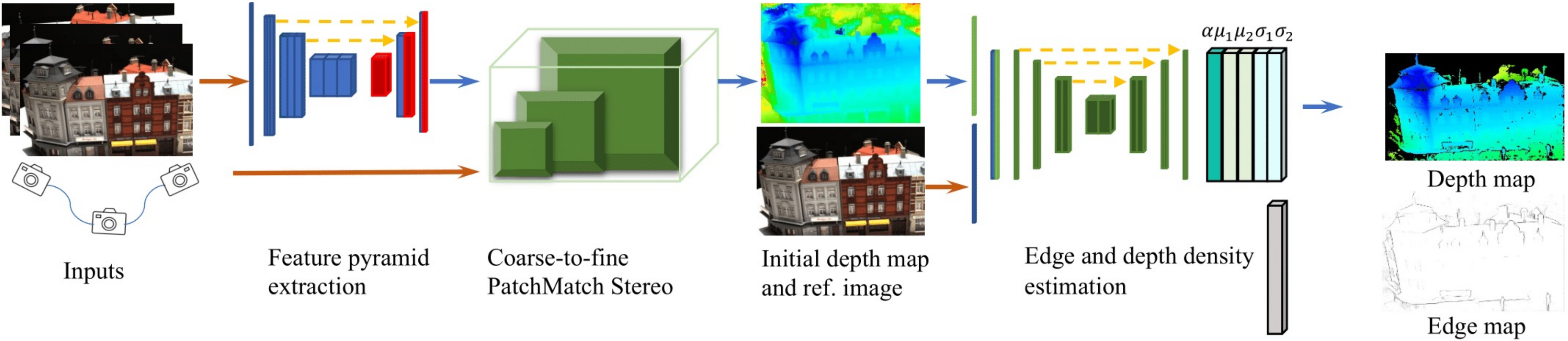


Yao Yao et. al.: MVSNet: Depth Inference for Unstructured Multi-view Stereo. ECCV 2018

DDLMVS



This video demonstrates visual comparisons with [COLMAP] and [PatchmatchNet]



MVCAST

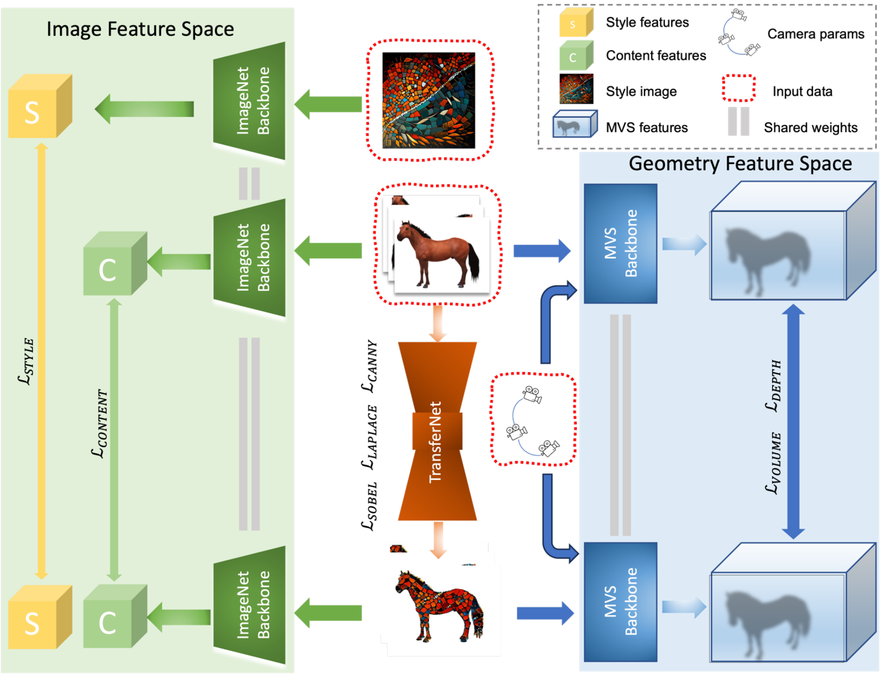


Image Feature Space
S
C
ImageNet Backbone
ImageNet Backbone
ImageNet Backbone
S
C
$\mathcal{L}_{STYLE}$
$\mathcal{L}_{CONTENT}$
$\mathcal{L}_{CANNY}$
$\mathcal{L}_{LAPLACE}$
$\mathcal{L}_{SOBEL}$
TransferNet
Style features
Content features
Style image
MVS features
Camera params
Input data
Shared weights
Geometry Feature Space
MVS Backbone
MVS Backbone
$\mathcal{L}_{DEPTH}$
$\mathcal{L}_{VOLUME}$

Thanks for Listening.