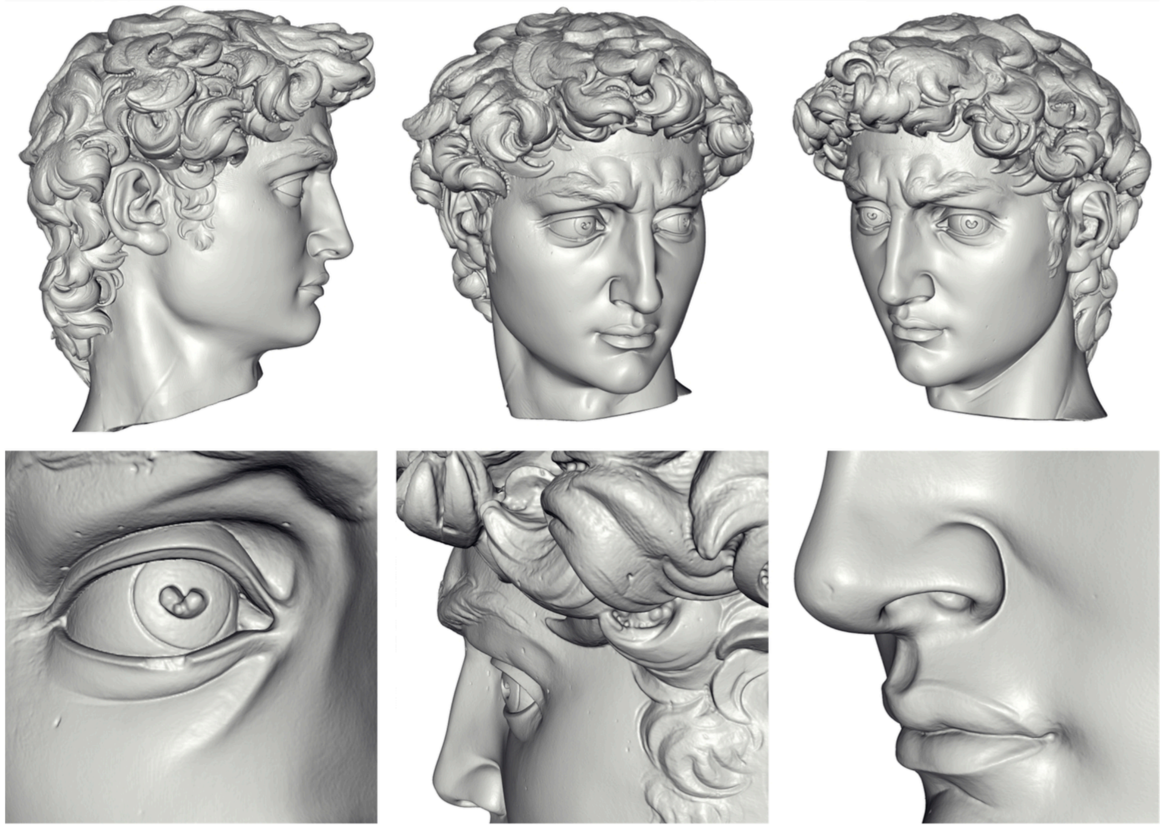


# 3D Reconstruction

Shanchen Jiang

# Film Industry

3D Camera + Virtual display

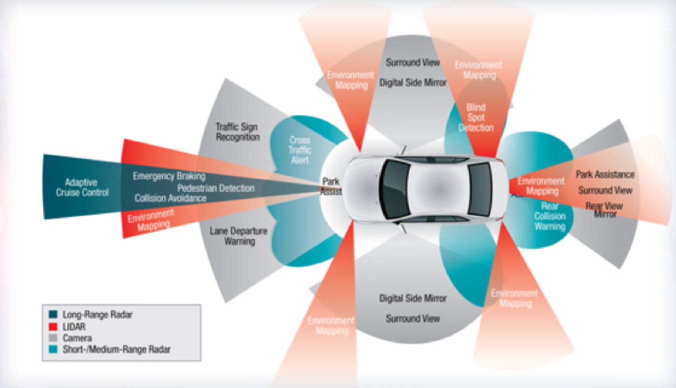
# Film Industry

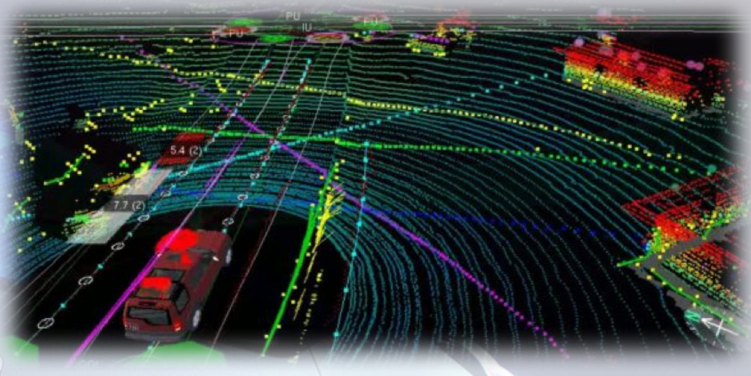

# City Planning


Taking Photos + Recovery from images

# City Planning

# Pilotless Automobile



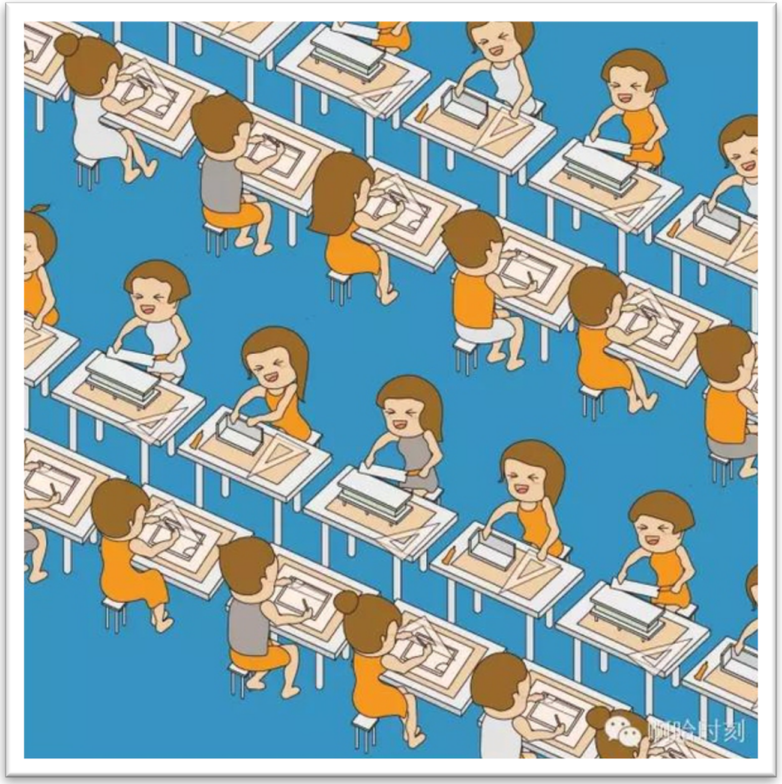



Sensors + Software

# Pilotless Automobile

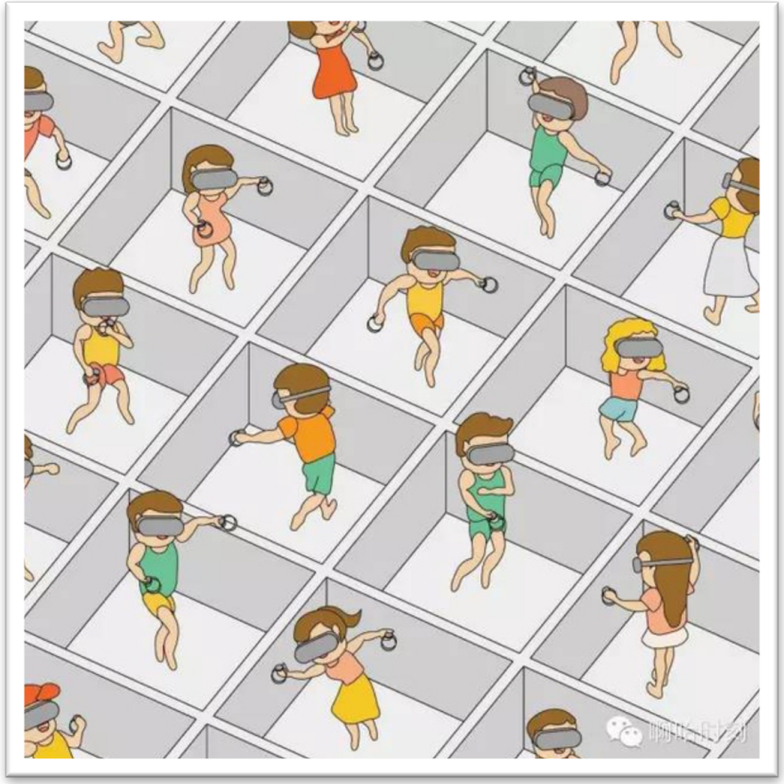

Everything is virtual

# Virtual Reality



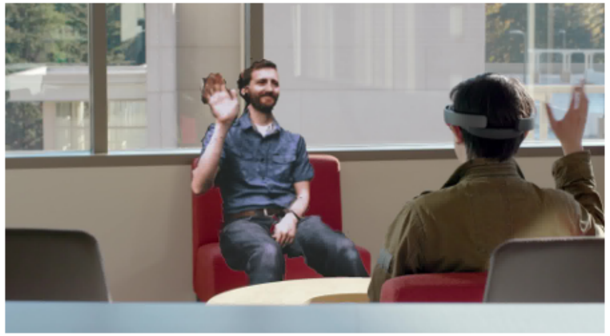
+

=



# Virtual Reality







# Augmented Reality

# Classification

GBM

Geometry Based Modeling

DSM

Depth Scanning Modeling

IBM

Image Based Modeling

In computer vision and computer graphics, 3D reconstruction is the process of capturing the shape and appearance of real objects.

*https://en.wikipedia.org/wiki/3D_reconstruction*

# History

1960s Problem Definition

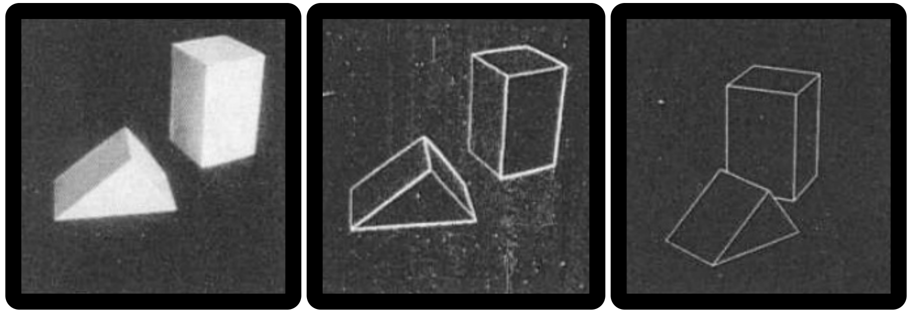
1970s Image Formulation

1980s

1990s Geometry

2000s Reconstruction

2010s Innovation



Input Gradient Output

Roberts L G. Machine perception of three-dimensional soups. *Massachusetts Institute of Technology*, 1963.

# History

1960s Problem Definition

**1970s Image Formulation**

1980s

1990s Geometry

2000s Reconstruction

2010s Innovation

# D.Marr

***Vision: A computational investigation into the human representation and processing of visual information*** **(ISBN 0-7167-1567-8)**


(1969) "A theory of cerebellar cortex." J. Physiol., 202:437-470.

(1970) "A theory for cerebral neocortex." Proceedings of the Royal Society of London B, 176:161-234.

(1971) "Simple memory: a theory for archicortex." Phil. Trans. Royal Soc. London, 262:23-81.

(1974) "The computation of lightness by the primate retina." Vision Research, 14:1377-1388.

(1975) "Approaches to biological information processing." Science, 190:875-876.

(1976) "Early processing of visual information." Phil. Trans. R. Soc. Lond. B, 275:483-524.

(1976) "Cooperative computation of stereo disparity." Science, 194:283-287. (with Tomaso Poggio)

(1976, March) "Artificial inte[illegible] Technical Report AIM 355, MIT AI Laboratory, Cambridge, MA.

(1977) "Artificial intelligence: A personal view." Artificial Intelligence 9(1), 37–48.

(1977) [illegible] Neurosciences Res. Prog. Bull., 15:470-488. (with Tomaso Poggio)

(1978) [illegible]ensional tructure [illegible] Nishihara)

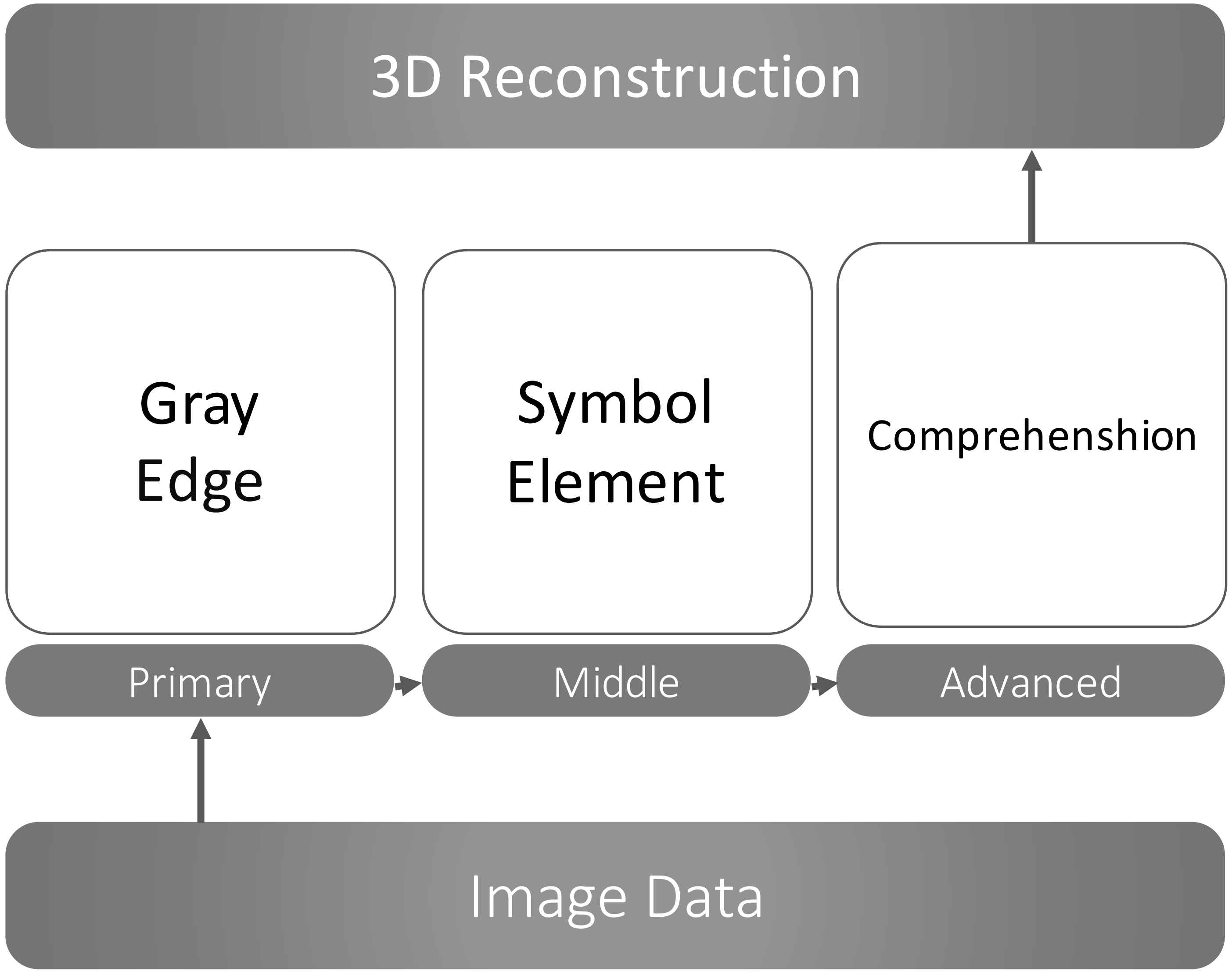
(1979) "A computational theory of human stereo vision." Proceedings of the Royal Society of London B, 204:301-328. (with Tomaso Poggio)

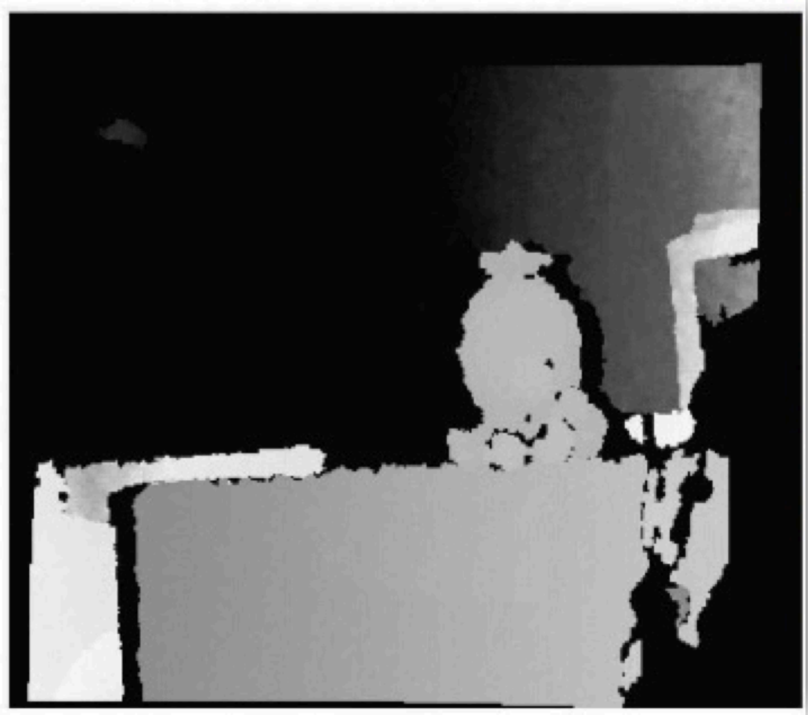
(1980) "Theory of edge detection." Proc. R. Soc. Lond. B, 207:187-217. (with E. Hildreth)

(1981) "Artificial intelligence: a personal view." In Haugeland, J., ed., Mind Design, chapter 4, pages 129-142. MIT Press, Cambridge, MA.

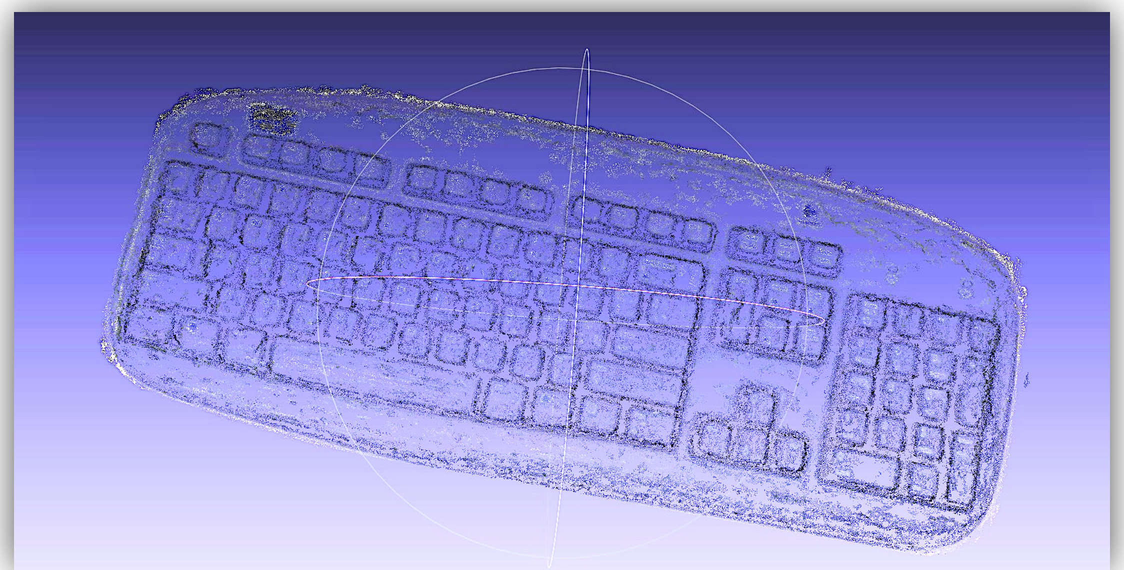
(1982) "Representation and recognition of the movements of shapes." Proceedings of the Royal Society of London B, 214:501-524. (with L. M. Vaina)

(1982). Vision: A Computational Investigation into the HumanRepresentation and Processing of Visual Information. New York: Freeman.

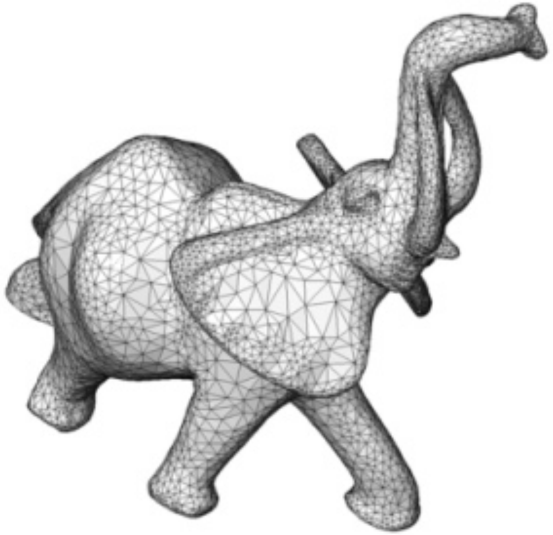

3D Reconstruction
Gray
Edge
Symbol
Element
Comprehenshion
Primary
Middle
Advanced
Image Data



2.5D information



Point cloud



Grid chart

# History

1960s Problem Definition

1970s Image Formulation

**1980s**

**1990s Geometry**

2000s Reconstruction

2010s Innovation

Essential Matrix: $X'_{\mu} Q_{\mu\upsilon} X_{\upsilon} = 0$

$$\mathbf{X}'_{\mu}\mathbf{Q}_{\mu\nu}\mathbf{X}_{\nu} = 0 \quad (11)$$

Dividing equation (11) by $X'_3X_3$ we arrive at the desired relationship between the image coordinates:

$$\mathbf{x}'_{\mu}\mathbf{Q}_{\mu\nu}\mathbf{x}_{\nu} = 0 \quad (12)$$

Longuet-Higgins H C. A computer algorithm for reconstructing a scene from two projections. *Readings in Computer Vision: Issues, Problems, Principles, and Paradigms, MA Fischler and O. Firschein,* 1987.

# History

1960s Problem Definition

1970s Image Formulation

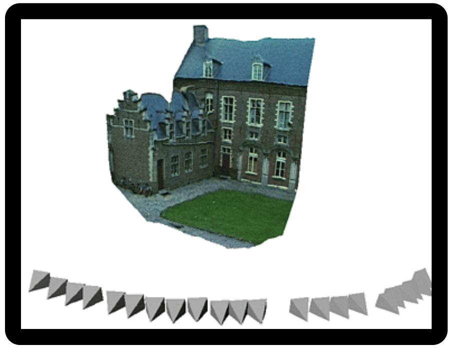
1980s

1990s Geometry

**2000s Reconstruction**

2010s Innovation

Structure from Motion



Pollefeys et al

Multi-view Stereo

Furukawa & Ponce

# History

| | |
|---|---|
| 1960s | Problem Definition |
| 1970s | Image Formulation |
| 1980s | |
| 1990s | Geometry |
| 2000s | Reconstruction |
| **2010s** | **Innovation** |

Big data
Real time
Deep learning

# History

1960s Problem Definition

1970s Image Formulation

1980s

1990s Geometry

2000s Reconstruction

2010s Innovation

Goals of Computer Vision:

- Let machines see
- Let humans see better

basis of 3D reconstruction

# DEPTH

# Classification

## Active methods

Structure light ...

## Passive methods

Recover from shadow

Recover from stereoscopic

Structure from motion

Photometric three-dimensional ...

# Active methods

Physical means

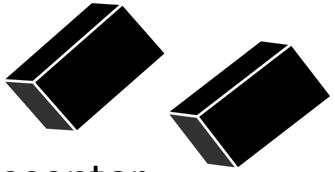
light

acoustic

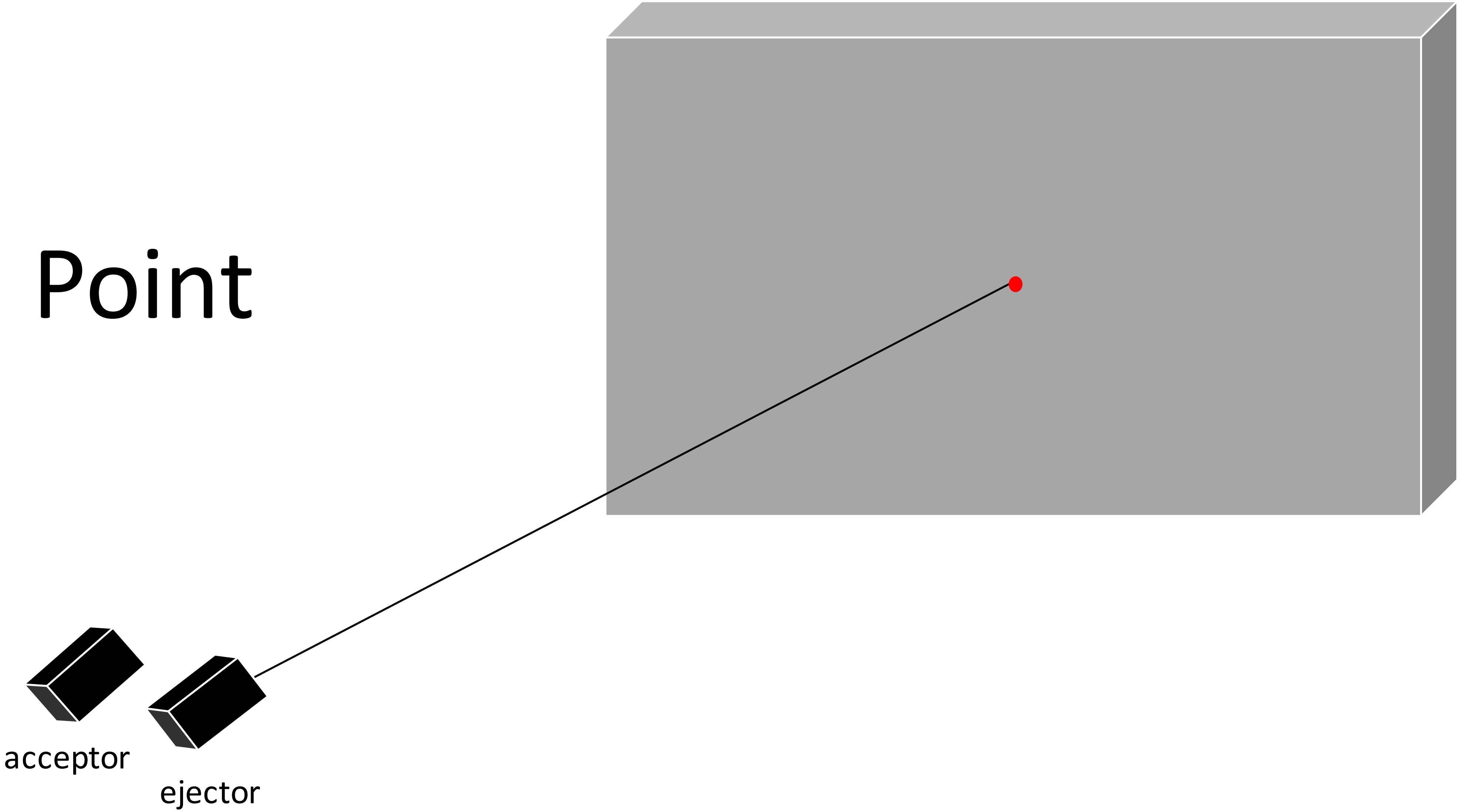
# Active methods

## Physical means

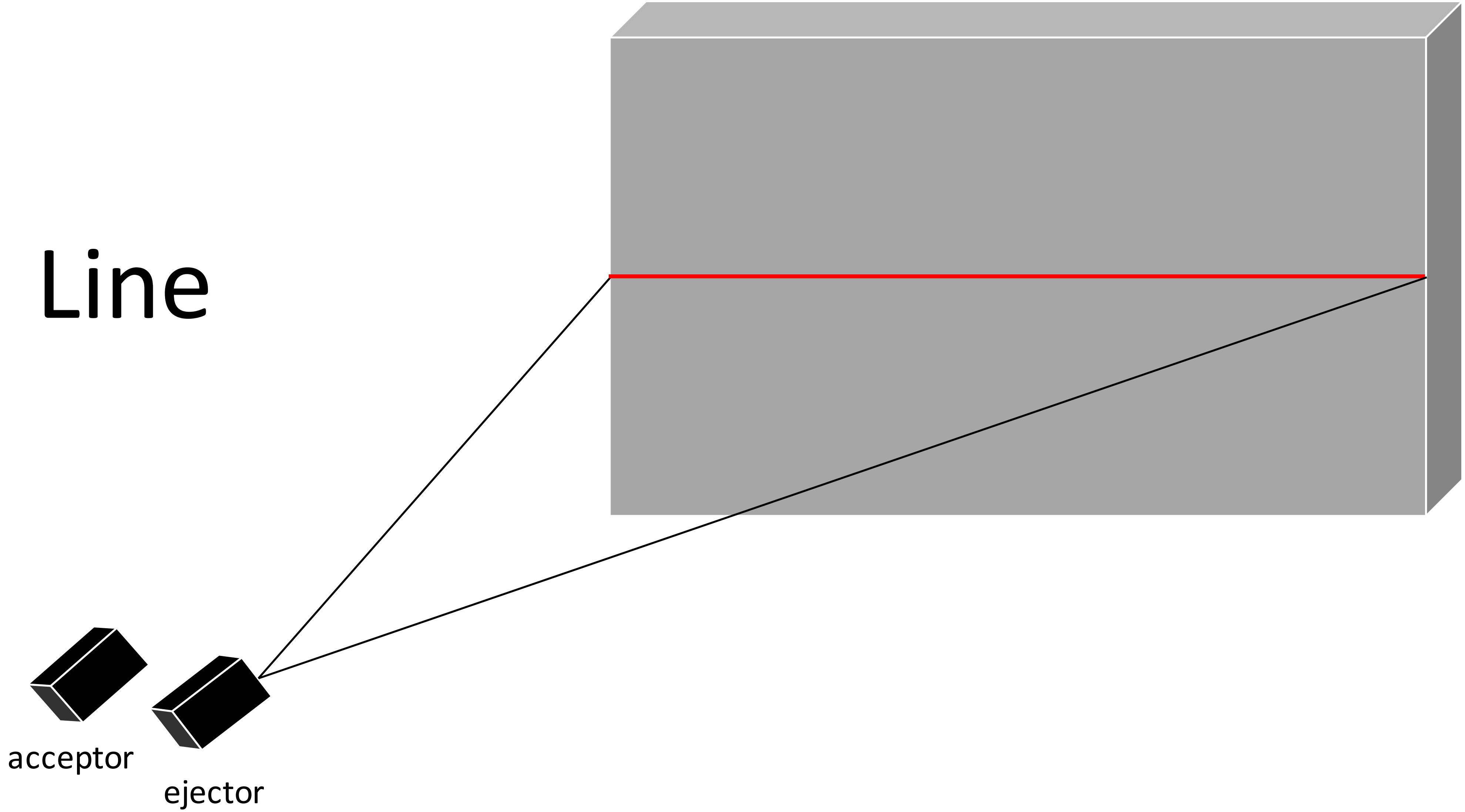
light



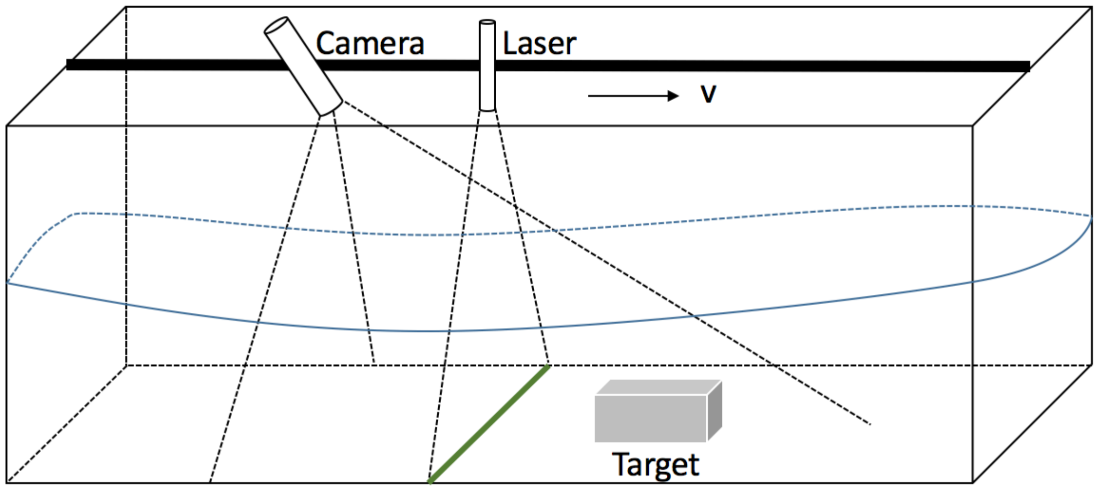
# Point



# Line



# Laser Line Scanning




Calibration

Acquisition

Extraction

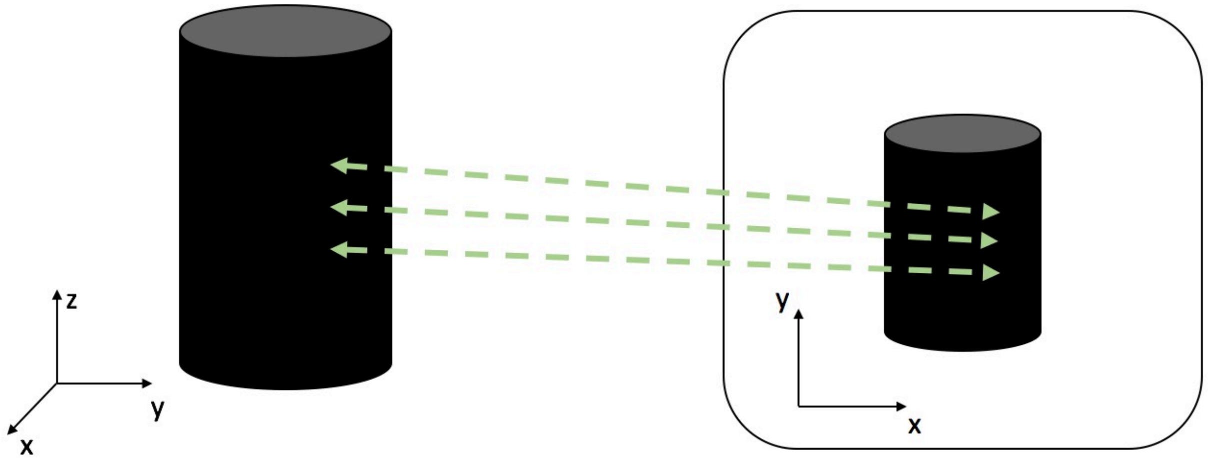
Transformation

Modeling

# Calibration

Calibration → Correction → Match → 3D Recover

| Internal reference | External reference | Distortion |
|---|---|---|

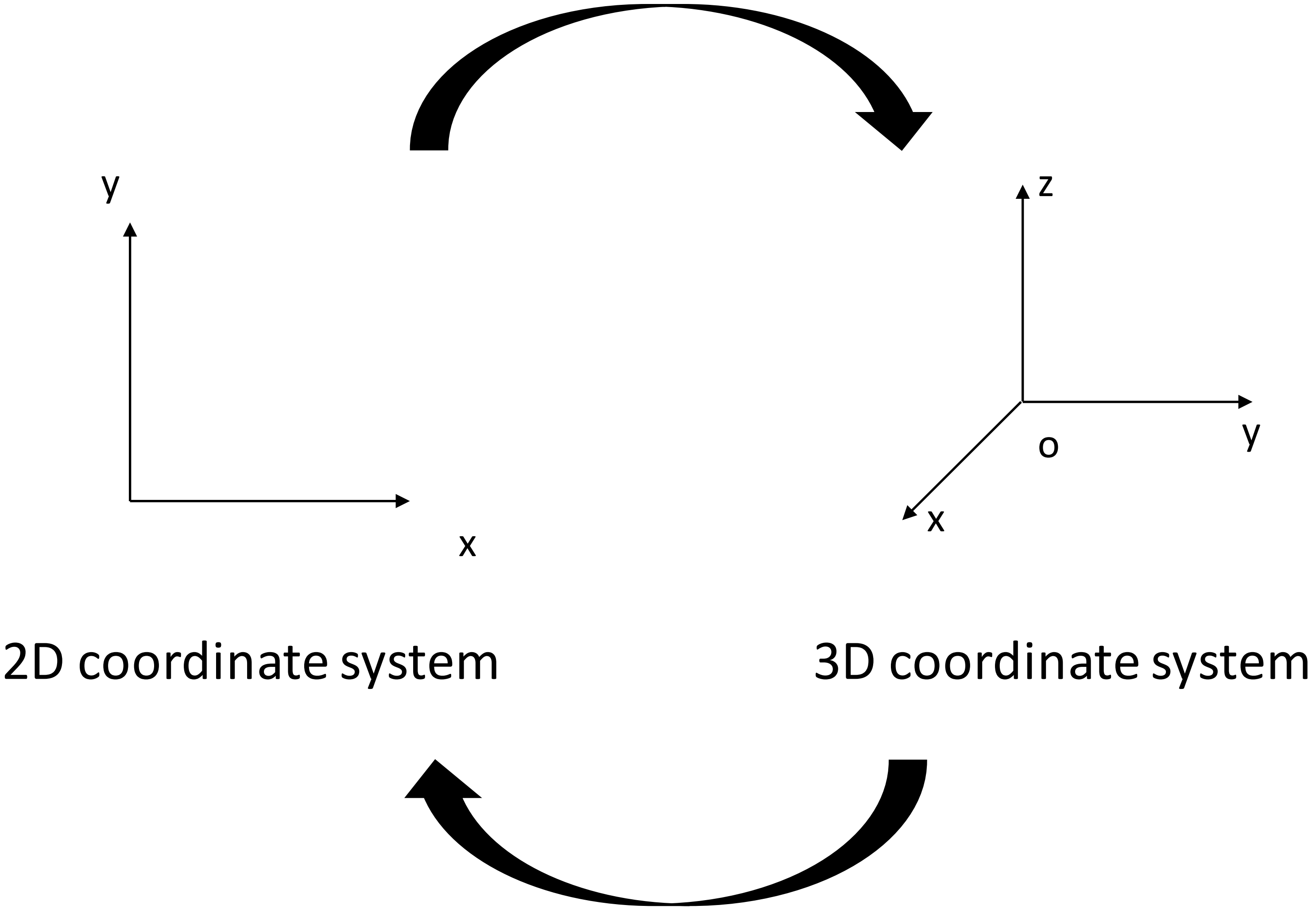


# Calibration

$$s\tilde{x} = K[R \quad t]\tilde{X} \tag{1}$$

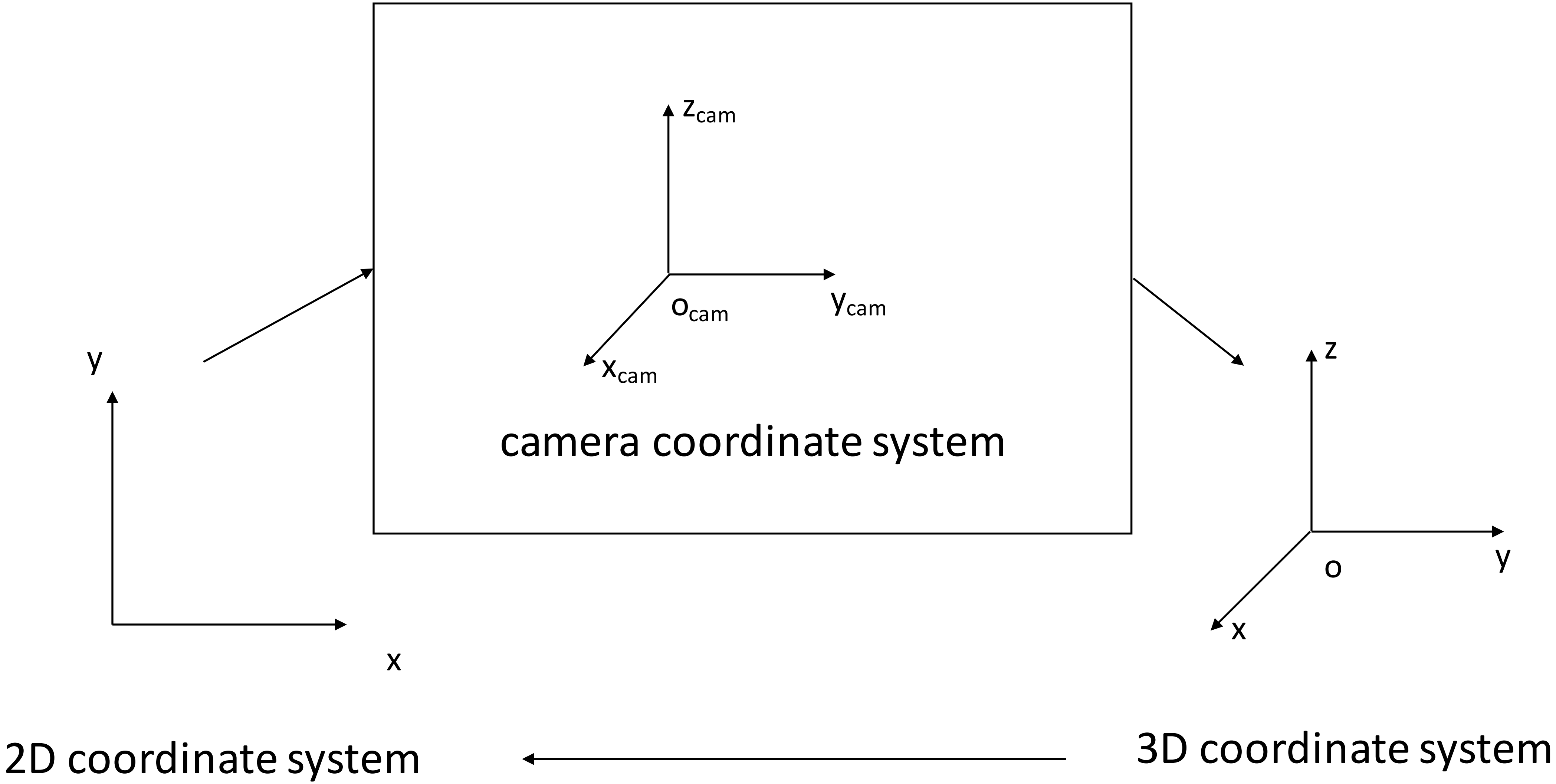
$$H = [\, h_1 \quad h_2 \quad h_3] = \lambda K[\, r_1 \quad r_2 \quad t] \tag{2}$$

[1] Faugeras O. Three-Dimensional Computer Vision: A Geometric Viewpoint . *MIT Press*, 1993.
[2] Hartley R I. An algorithm for self calibration from several views. *CVPR*, 1994.
[3] Luong Q T. Self-calibration of a moving camera from point correspondences and fundamental matrices. *International Journal of computer vision* , 1997.




Jianxiong Xiao
PRINCETON VISION GROUP

*calibration*




*calibration*

# I am a handsome boy

calibration

# I am a 3D Point

# I am a 3D Point

•

$$\mathbf{X} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

# I am cool

People loves to take picture of me.

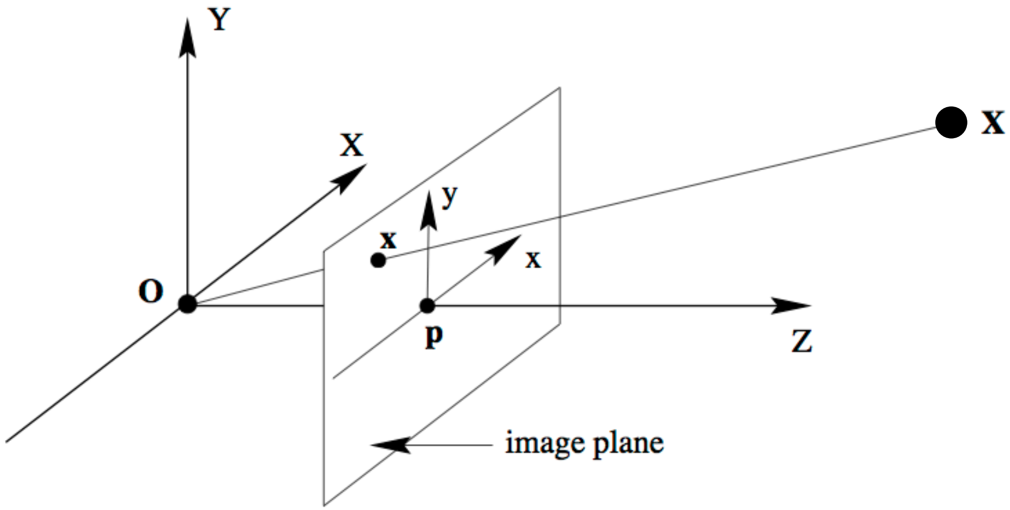
•

$$\mathbf{X} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

*calibration*

# I am cool

People loves to take picture of me.

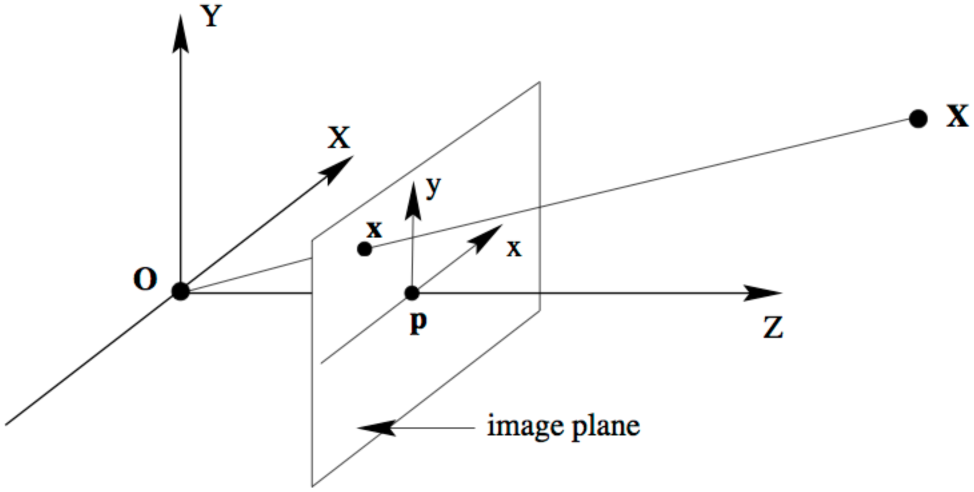



$$\mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix} \qquad \mathbf{X} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

2D 3D

*calibration*

# When they take a picture




$$
\begin{bmatrix} x \\ y \\ \boxed{f} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}
$$

2D 3D

$$\mathbf{x} = \mathbf{K}\left[\mathbf{R} \middle| \mathbf{t}\right]\mathbf{X}$$

*calibration*

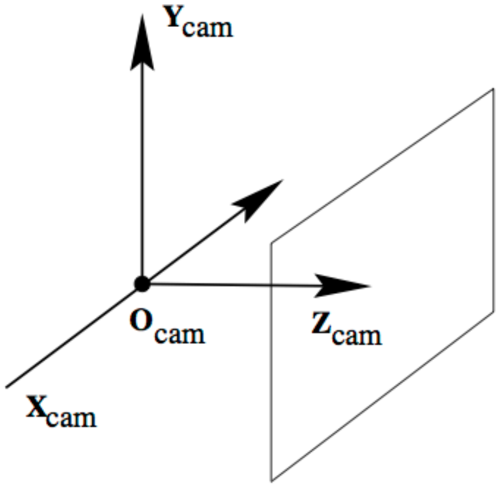
# When they take a picture

$$\begin{bmatrix} x \\ y \\ f \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \Rightarrow \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

$$\mathbf{x} = \mathbf{K}\left[\mathbf{R} \middle| \mathbf{t}\right]\mathbf{X}$$

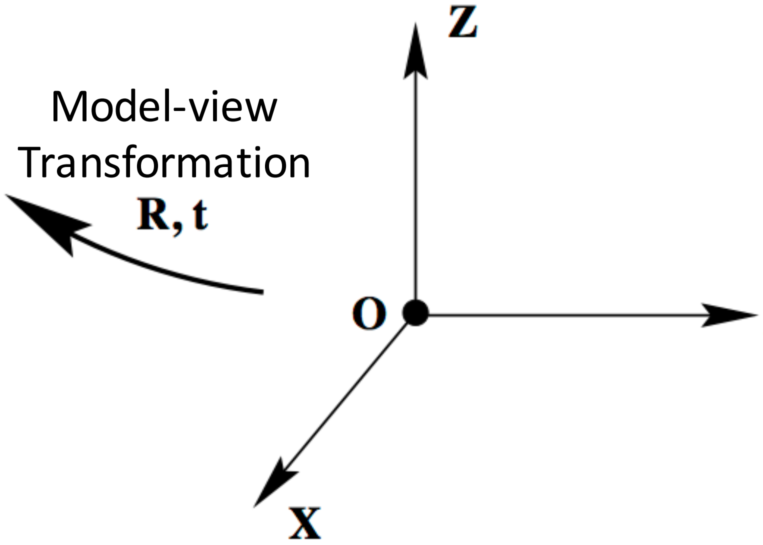
*calibration*

# I live in a real world




Camera Coordinate System

Model-view
Transformation
**R, t**




World Coordinate System

$$\begin{pmatrix} X_{cam} \\ Y_{cam} \\ Z_{cam} \\ 1 \end{pmatrix} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

3D 3D

$$\mathbf{x} = \mathbf{K}\left[\mathbf{R} \middle| \mathbf{t}\right]\mathbf{X}$$

*calibration*

# World Coor. → Camera Coor.

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

$$\mathbf{P} = \mathbf{K}\left[\mathbf{R}|\mathbf{t}\right]$$

Intrinsic Extrinsic

Camera Parameter
Camera Projection Matrix

$$\mathbf{K} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \mathbf{x} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad \mathbf{X} = \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

$$\mathbf{x} = \mathbf{K}\left[\mathbf{R}|\mathbf{t}\right]\mathbf{X}$$

$$\mathbf{x} = \mathbf{P}\mathbf{X}$$

*calibration*

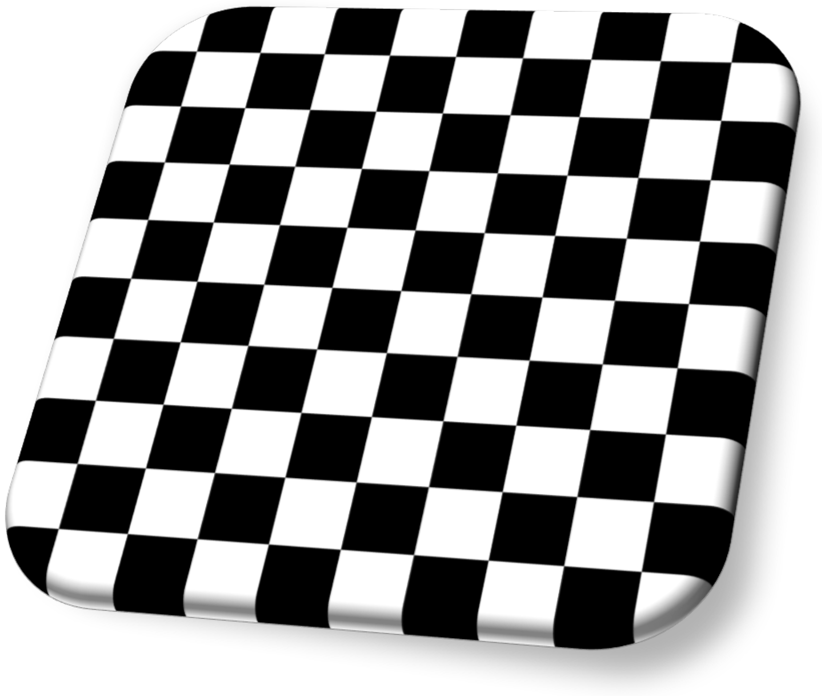
# Calibration

Traditional object-based calibration[1]

Self-calibration[2][3]

[1] Faugeras O. Three-Dimensional Computer Vision: A Geometric Viewpoint . *MIT Press*, 1993.
[2] Hartley R I. An algorithm for self calibration from several views. *CVPR*, 1994.
[3] Luong Q T. Self-calibration of a moving camera from point correspondences and fundamental matrices. *International Journal of computer vision* , 1997.

# Zhang Zhengyou calibration method



Closed-Form solution

↓ *maximum-likelihood criterion*

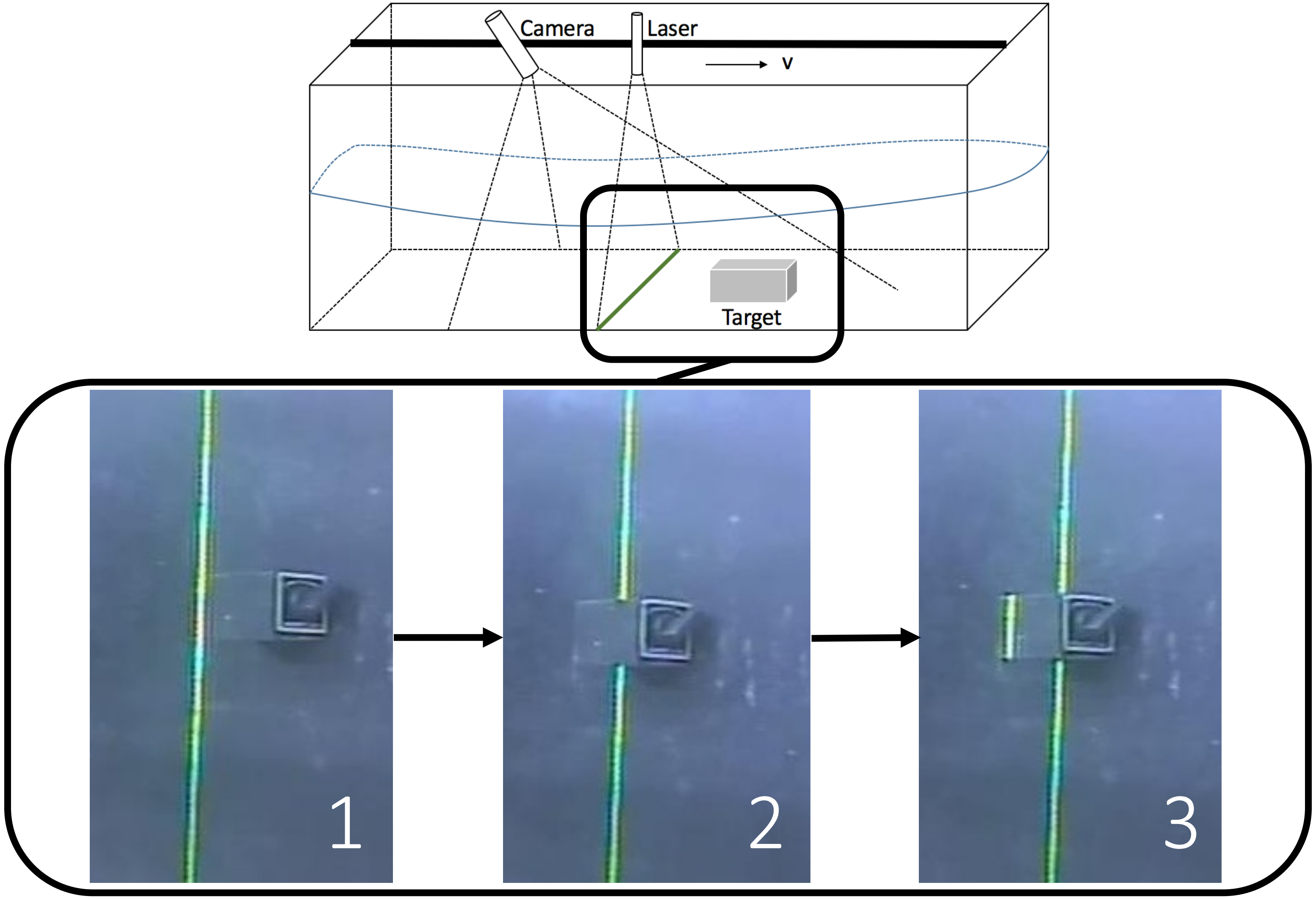
Nonlinear optimization solution

↓ *radial distortion*

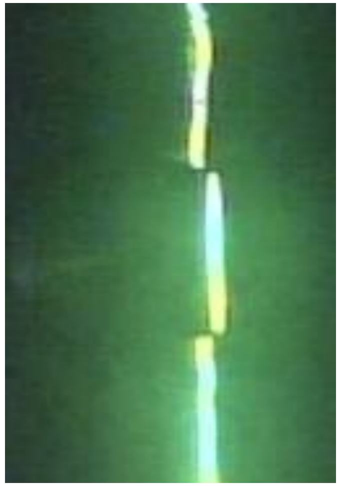
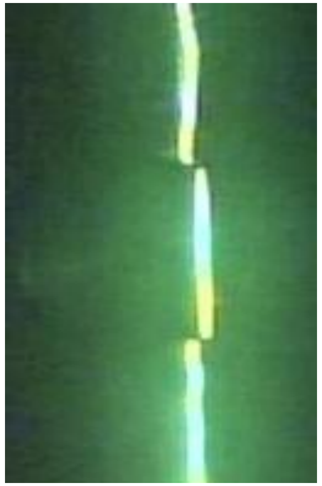
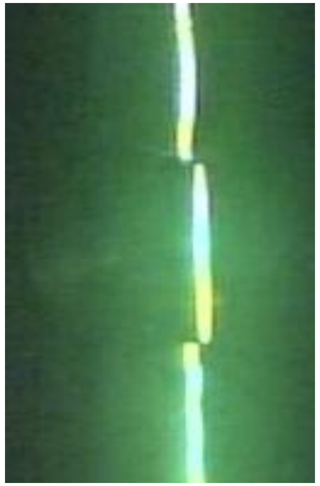
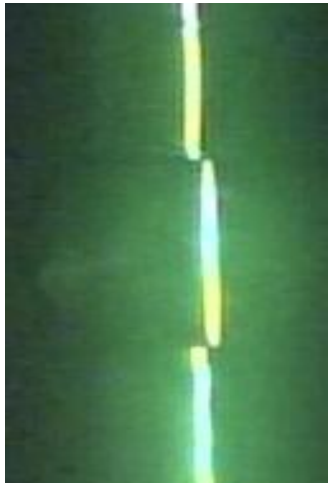
Analytical solution + Nonlinear solution

Zhang Z Y. A flexible new technique for camera calibration. *IEEE Transactions on pattern analysis and machine intelligence*, 2000.

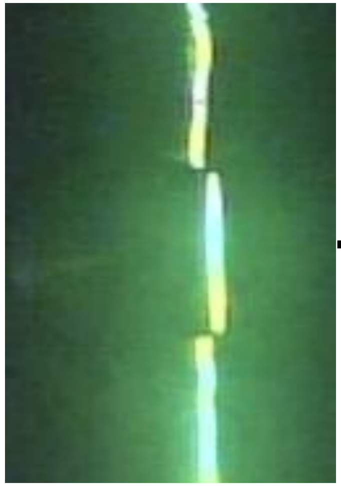
# Acquisition

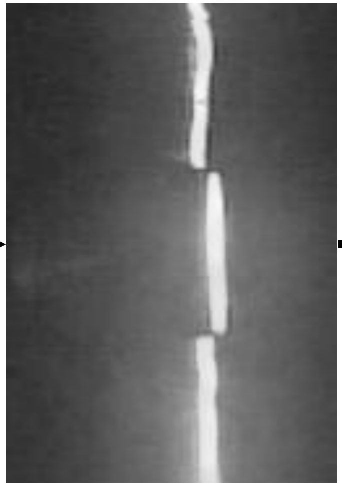


# Acquisition

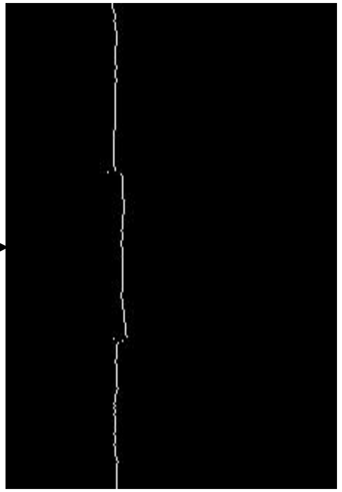

# Extraction

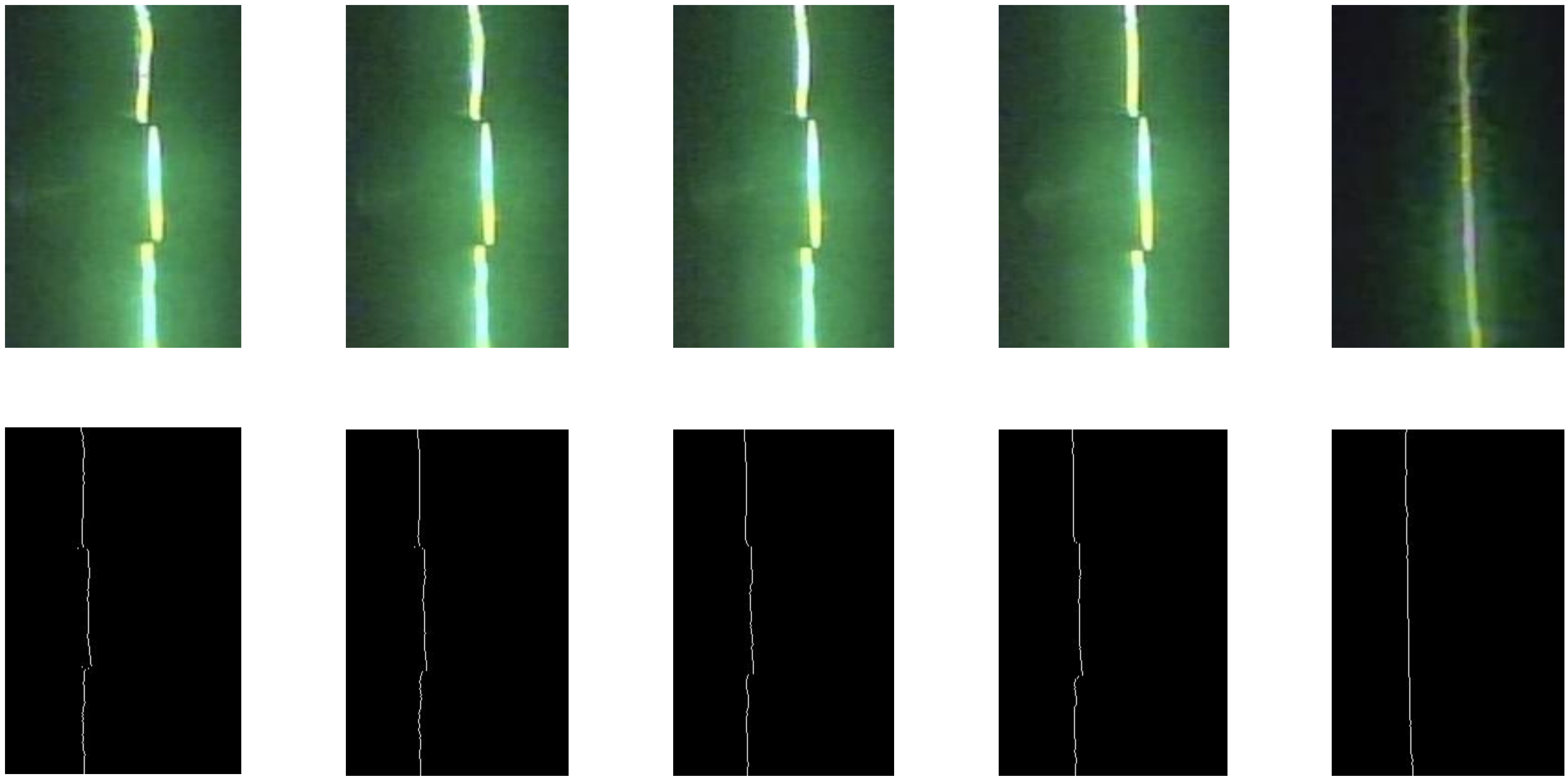




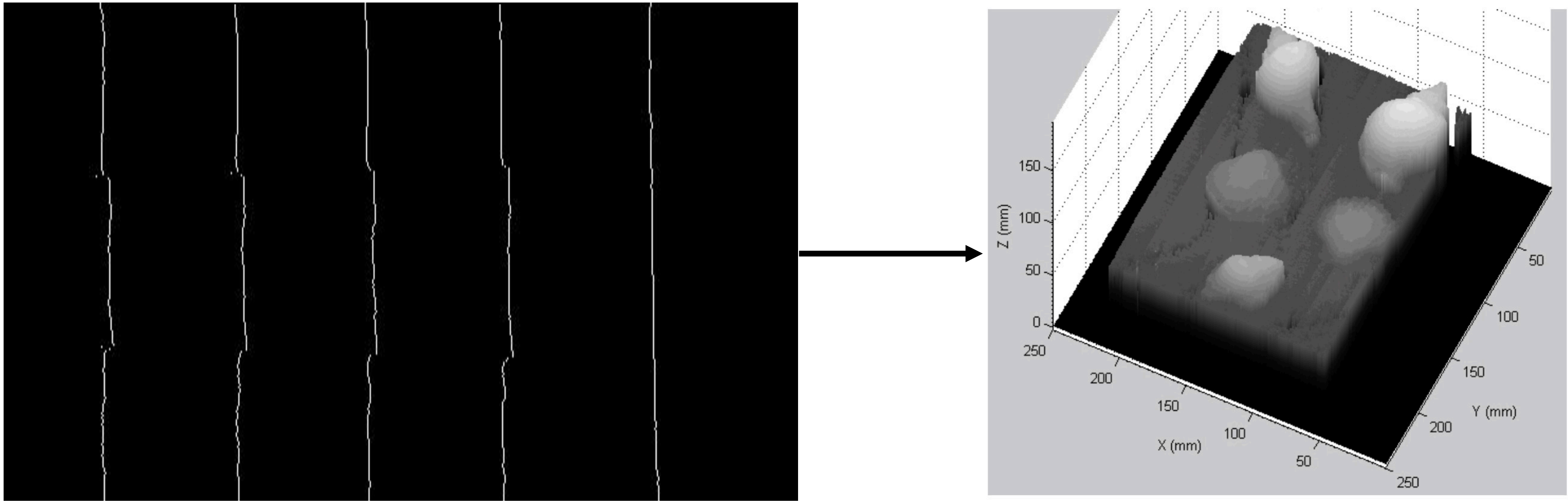
$$x_0 = \frac{\sum_x x \cdot \ I_x}{\sum_x \ I_x}$$

Center of gravity

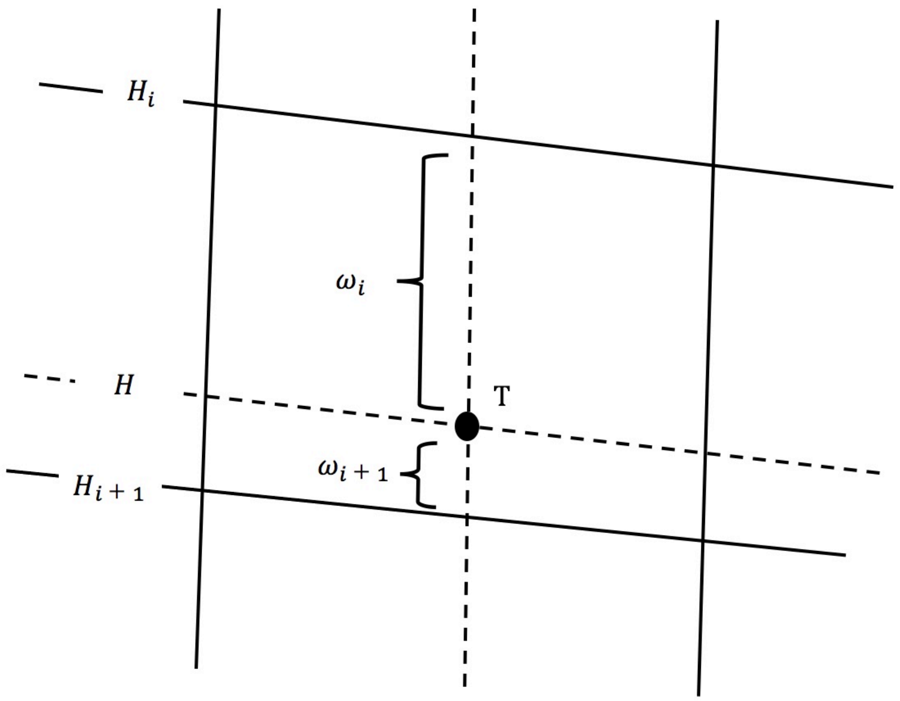




# Transformation



# Transformation

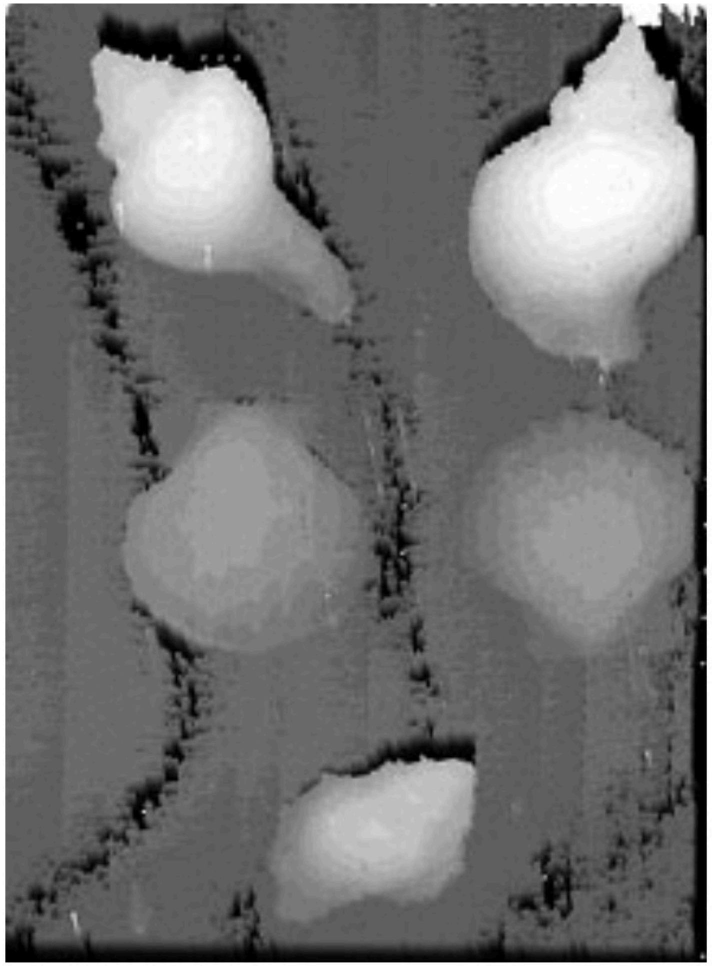



$$\omega_i = \frac{H - H_i}{H_{i+1} - H_i}$$

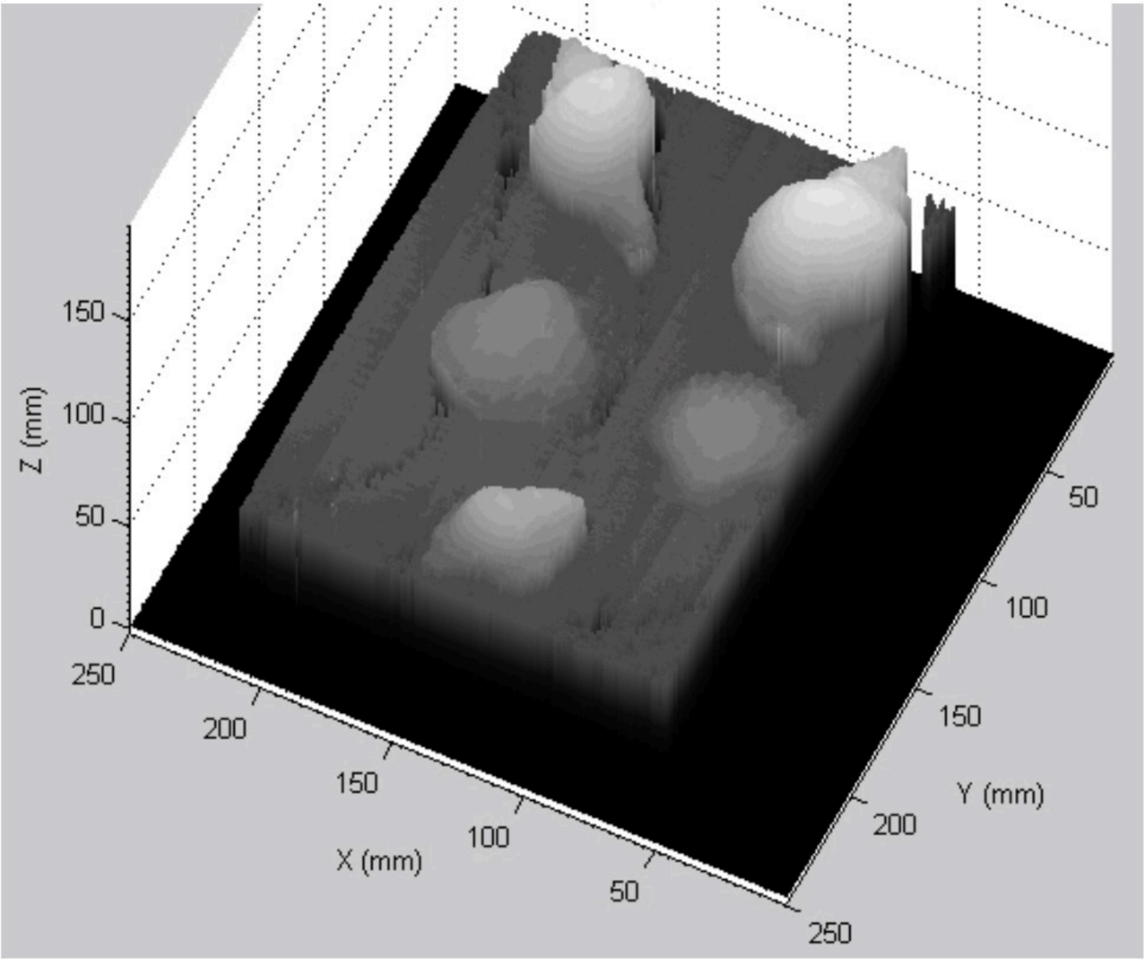
$$\omega_i + \omega_{i+1} = 1$$

$$h = \omega_i H_i + \omega_{i+1} H_{i+1}$$

Wang C C, Cheng M S. Nonmetric camera calibration for underwater laser scanning system. *IEEE Journal of Oceanic Engineering* , 2007.

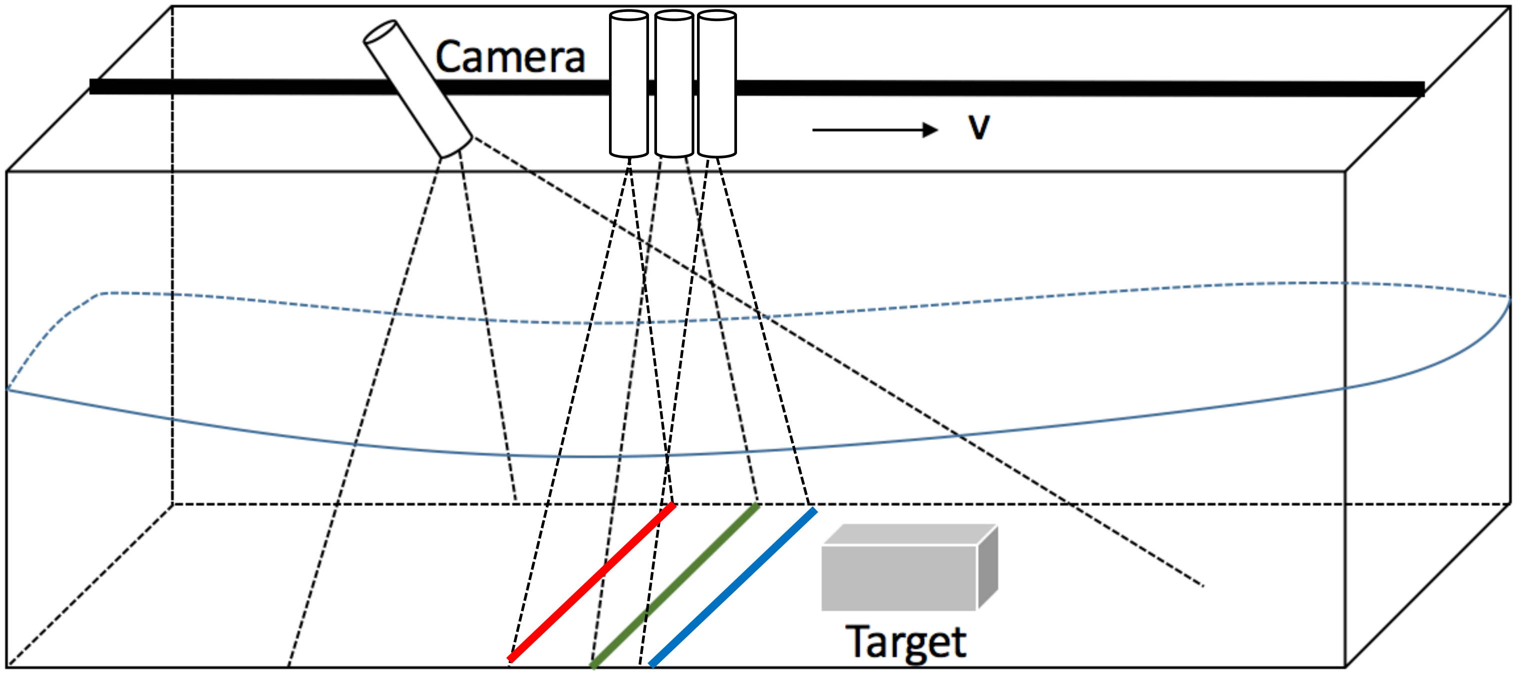


(a)



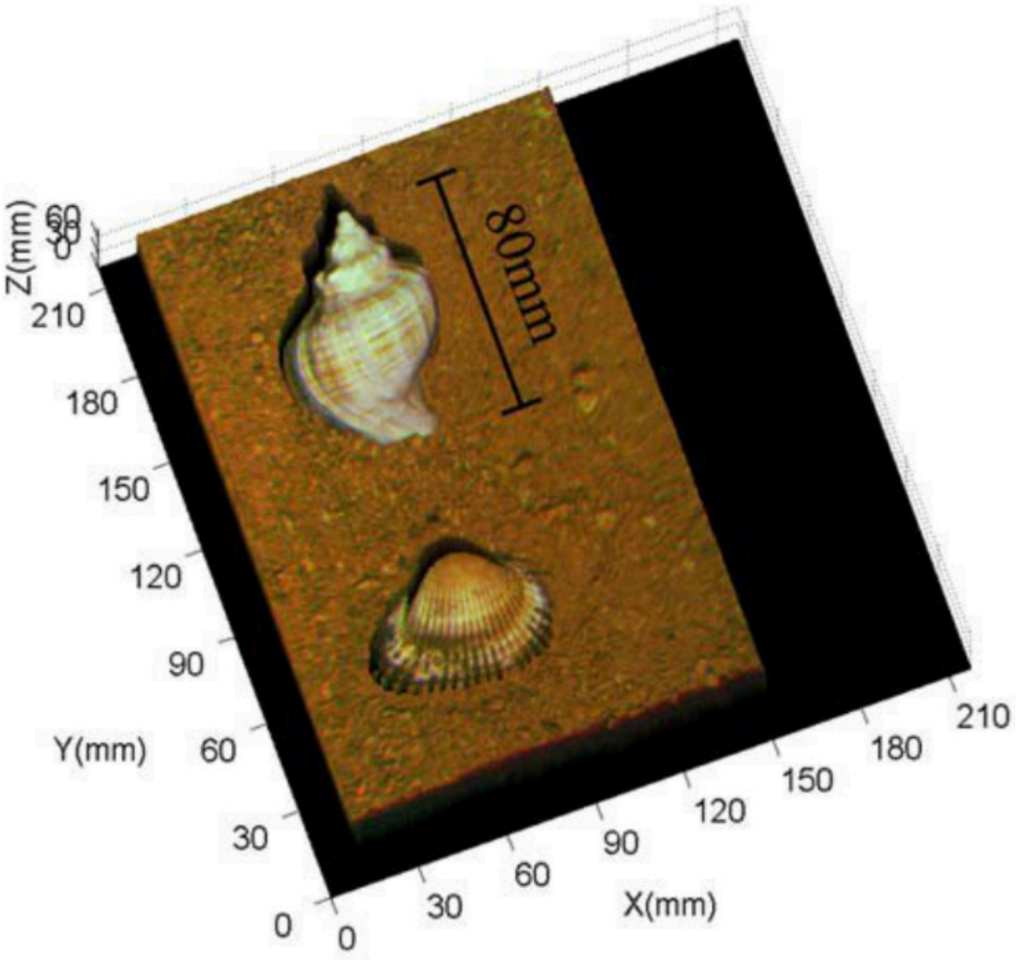

(b)

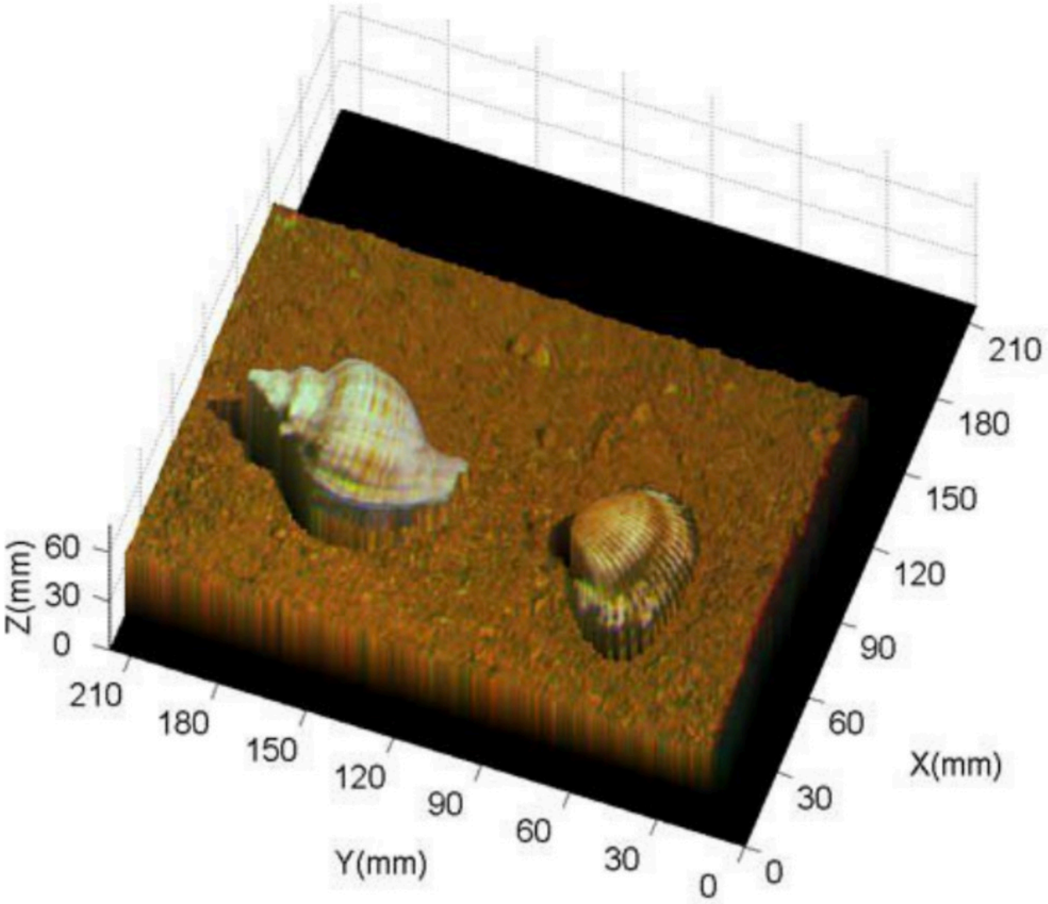
Yang Y, *et al*. 3D reconstruction for underwater laser line scanning. *OCEANS-Bergen,* 2013.




Calibration + Color calibration

Transformation + Color compound


80mm
Z(mm)
60
30
0
Y(mm)
210
180
150
120
90
60
30
0
X(mm)
0
30
60
90
120
150
180
210
（a）


Z(mm)
60
30
0
Y(mm)
210
180
150
120
90
60
30
0
X(mm)
0
30
60
90
120
150
180
210
（b）

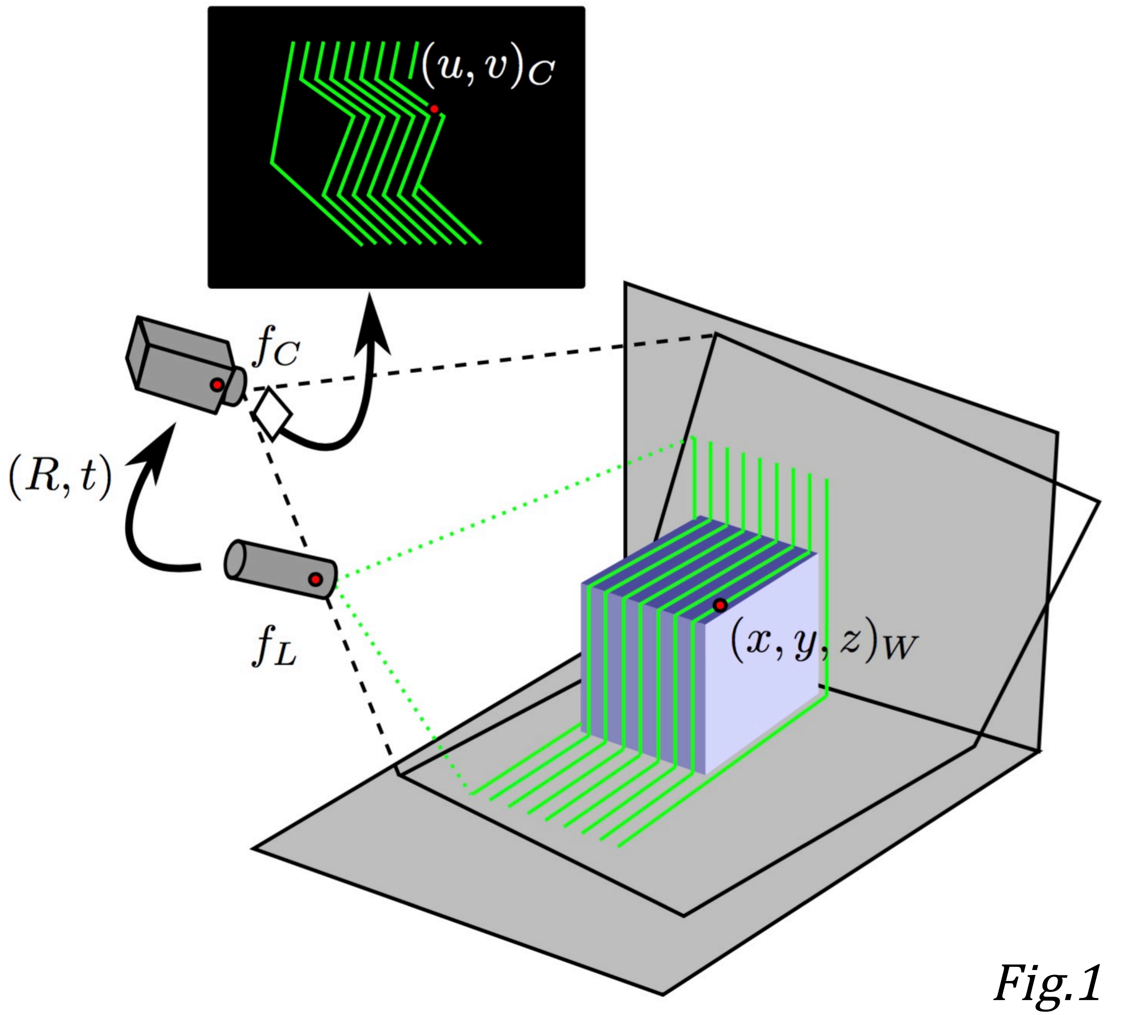


*Fig.1*

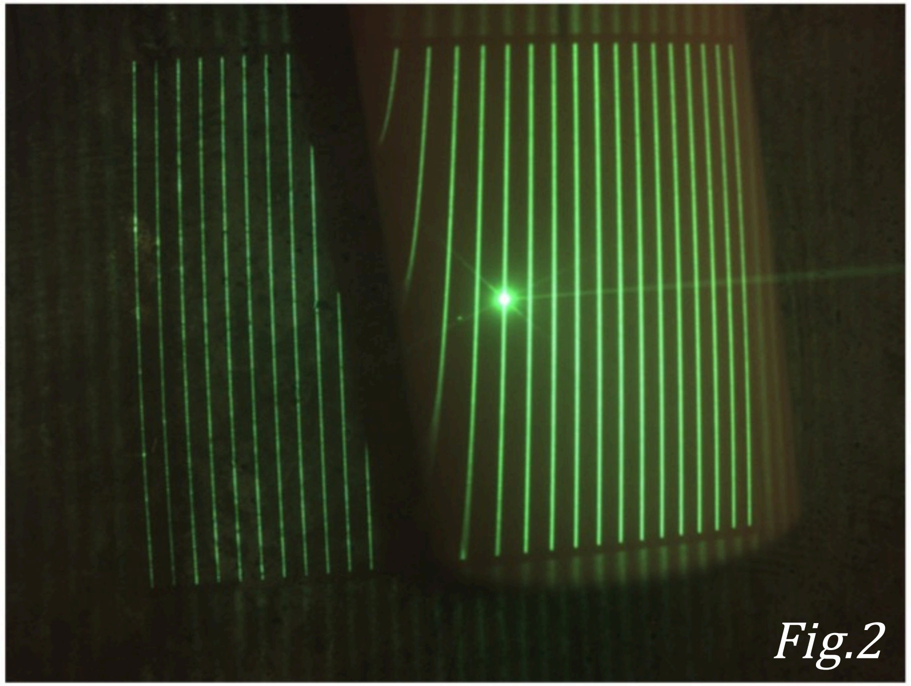


*Fig.2*

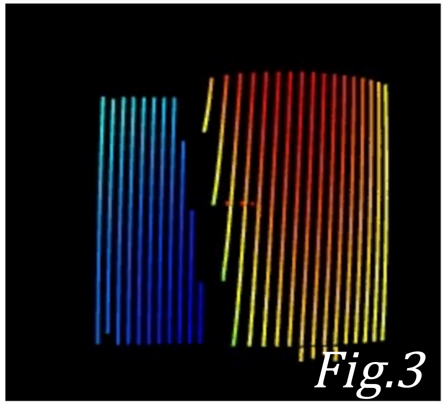


*Fig.3*

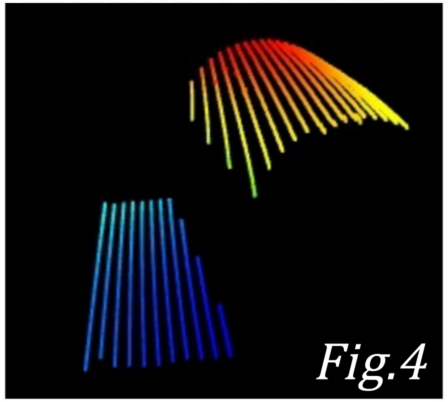
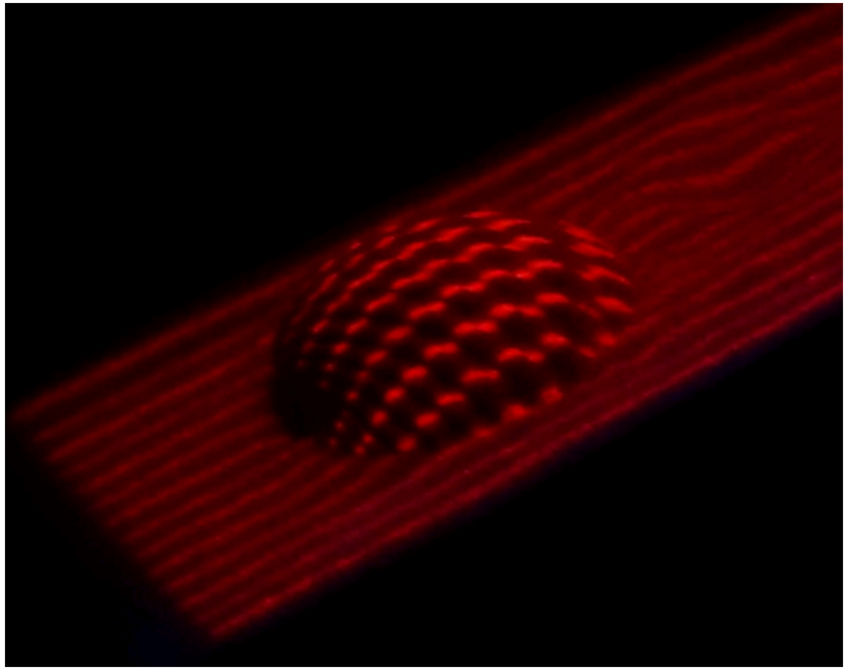
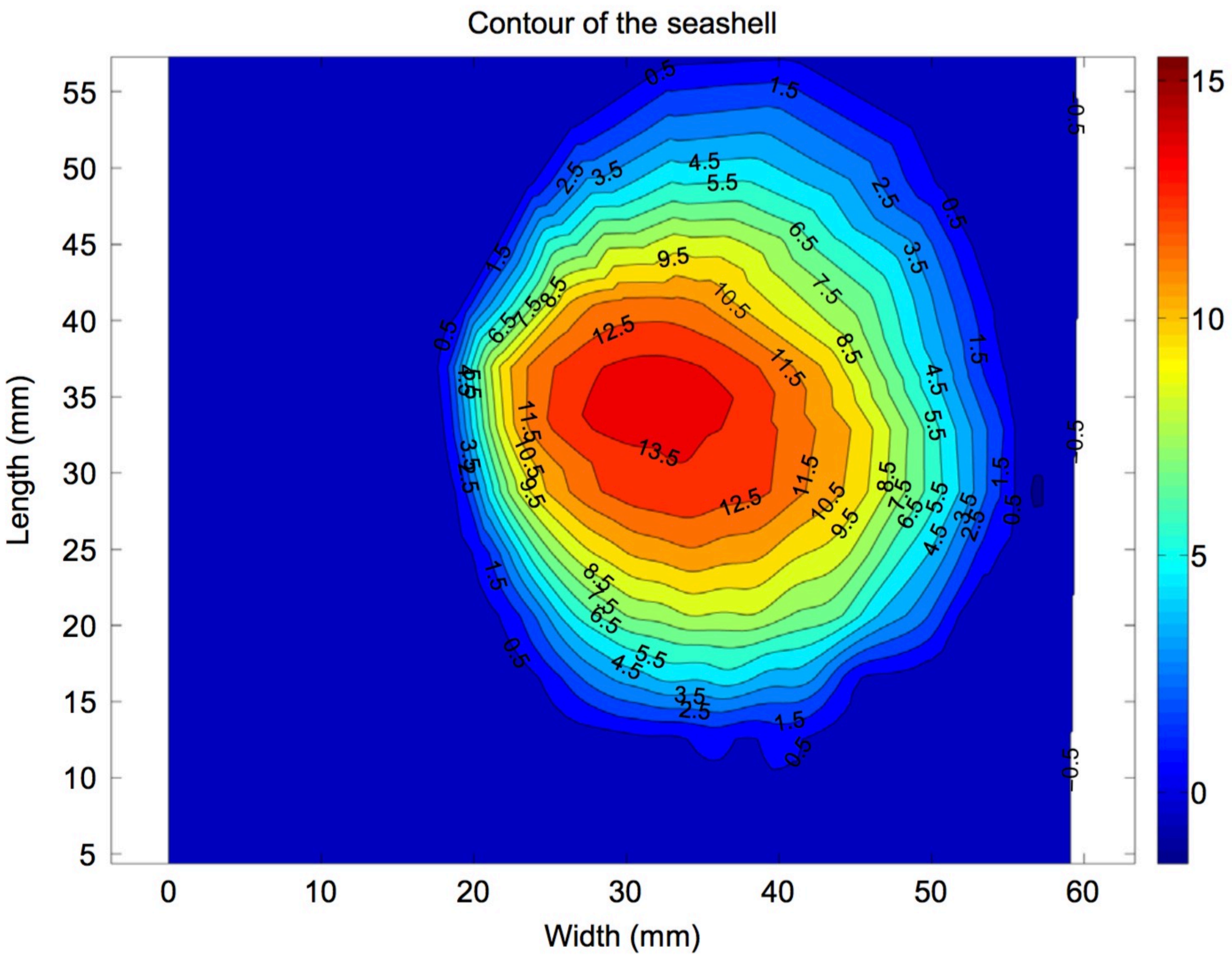


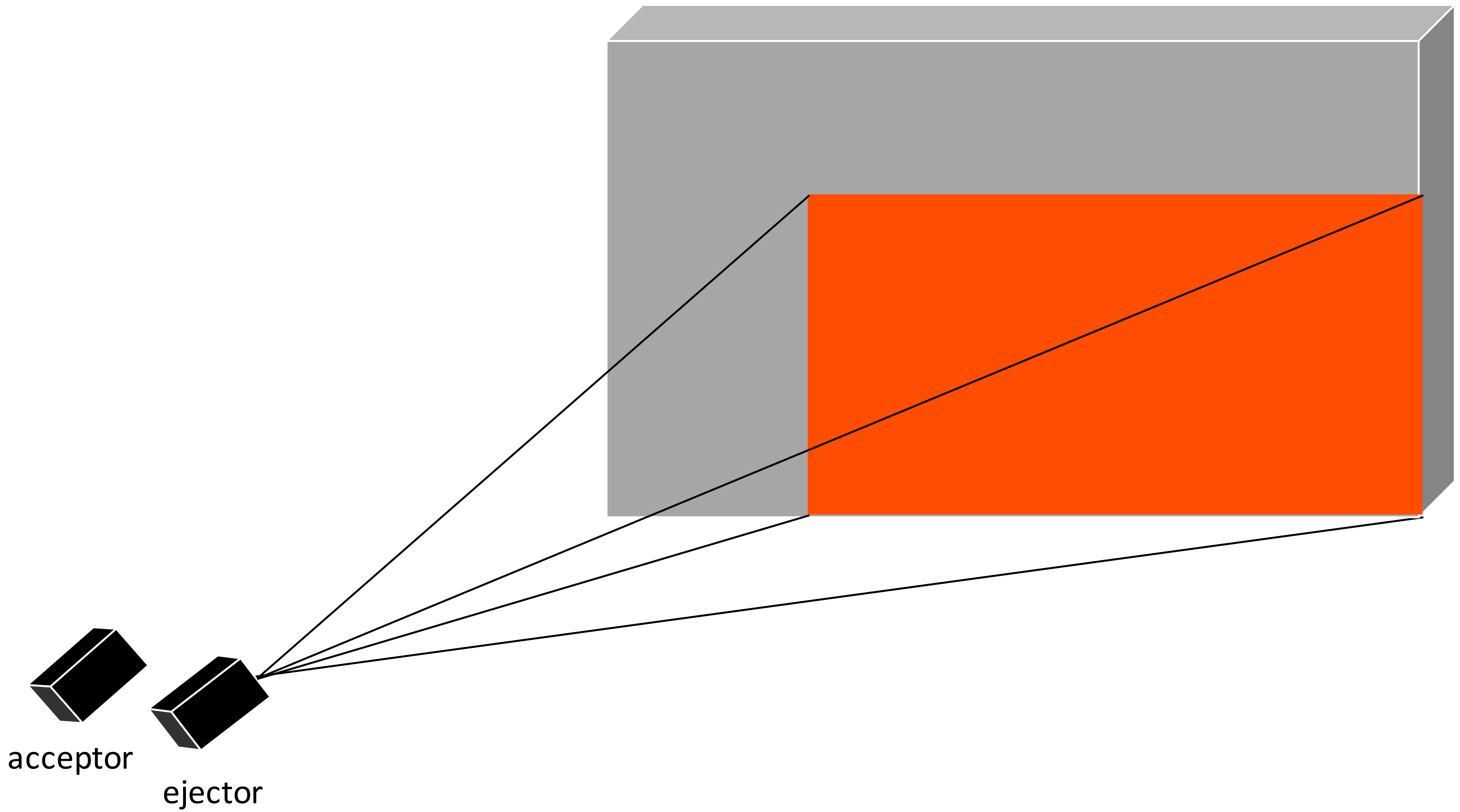
*Fig.4*

Massot-Campos M, and Oliver-Codina G. One-shot underwater 3D reconstruction. *Emerging Technology and Factory Automation (ETFA), 2014.*






Cebrián-Robles D, Ortega-Casanova J. Low cost 3D underwater surface reconstruction technique by image processing. *Ocean Engineering*, 2016.



acceptor
ejector

# Modified method

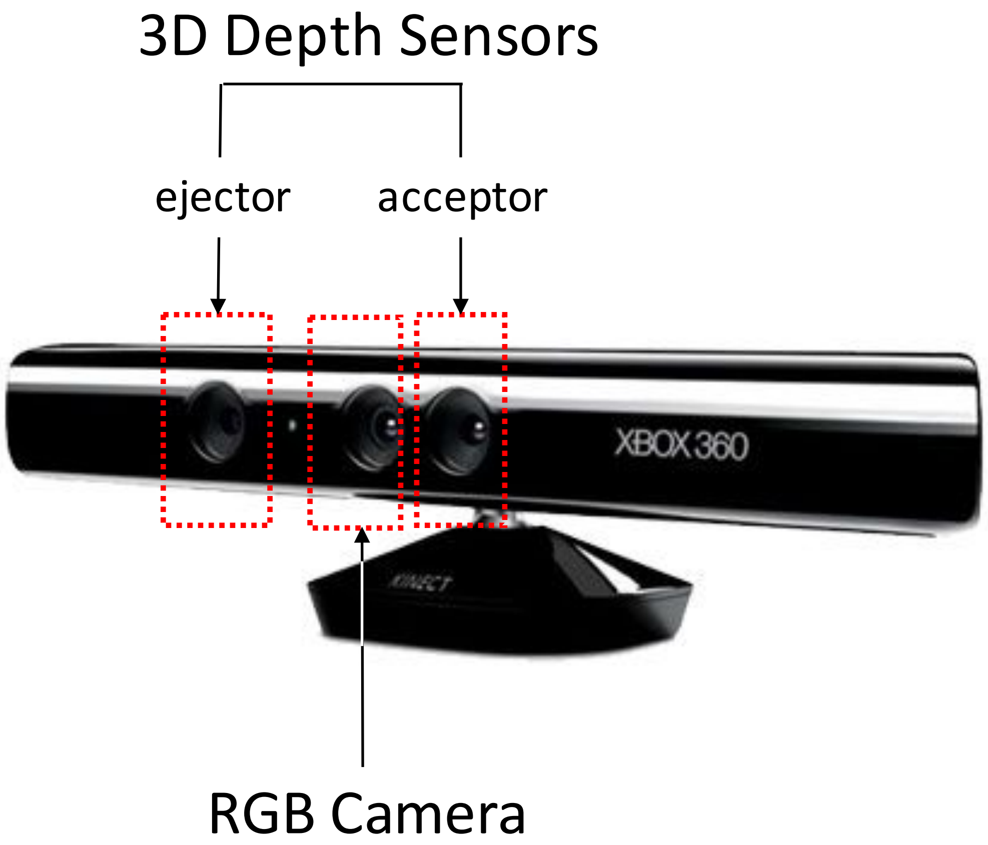
Kinect

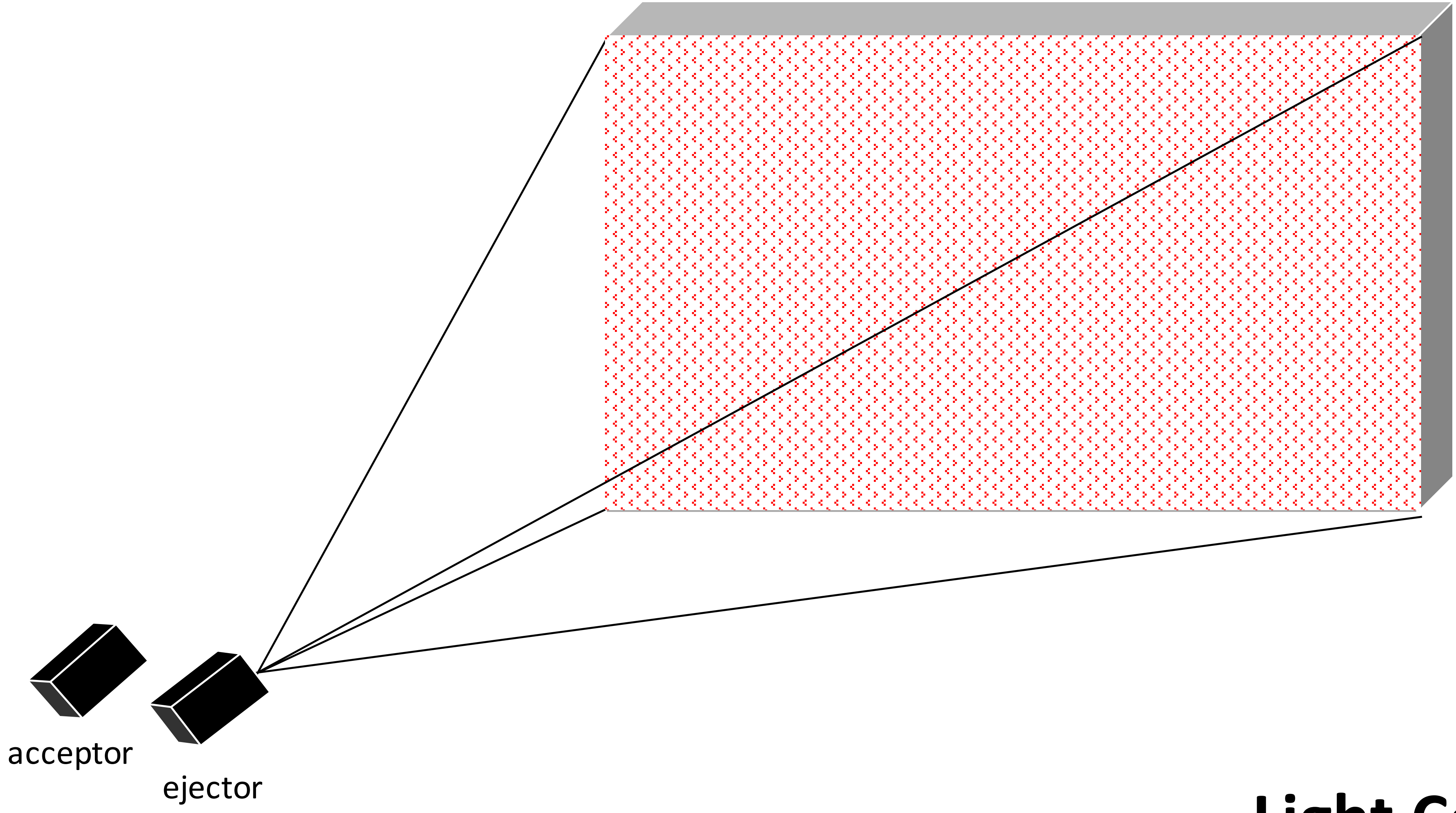

First generation


Second generation

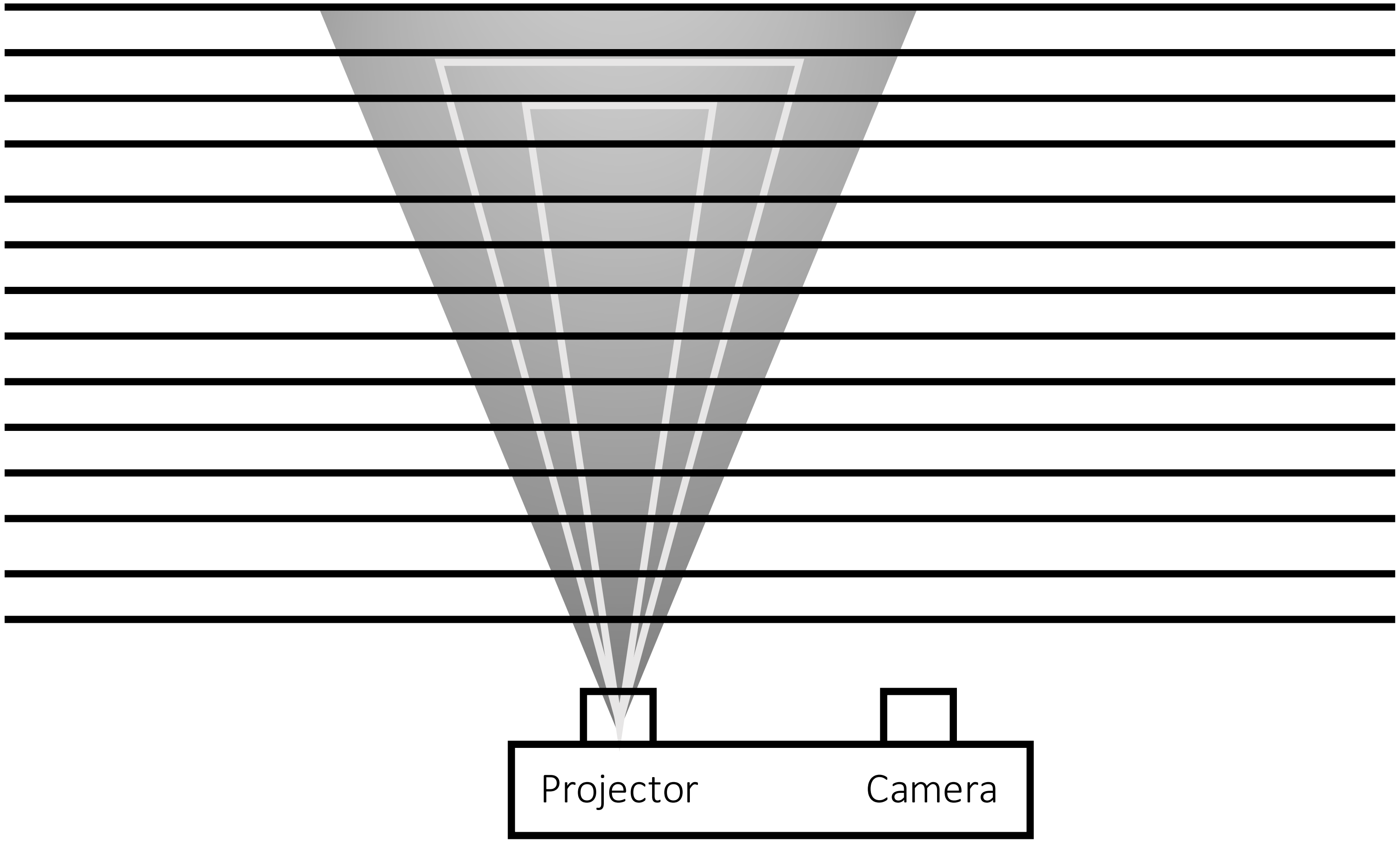
**Light Coding**

**Time of Flight**

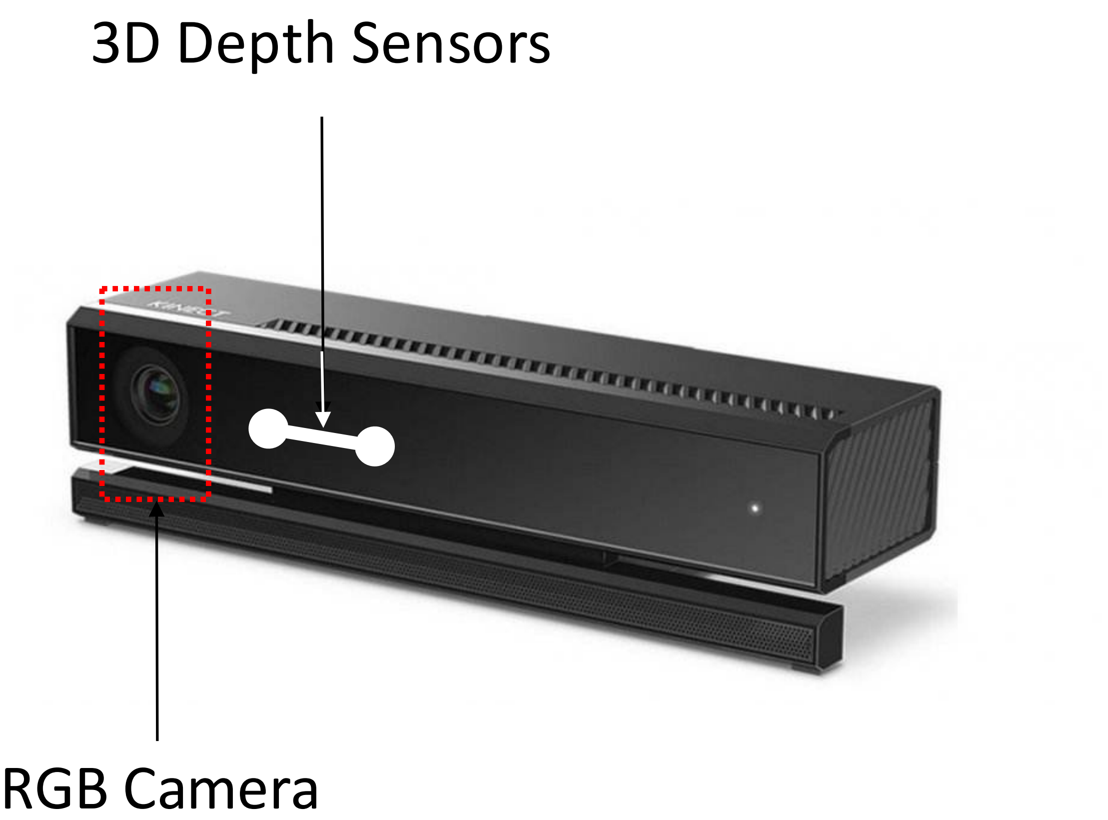

3D Depth Sensors
ejector
acceptor
XBOX 360
RGB Camera



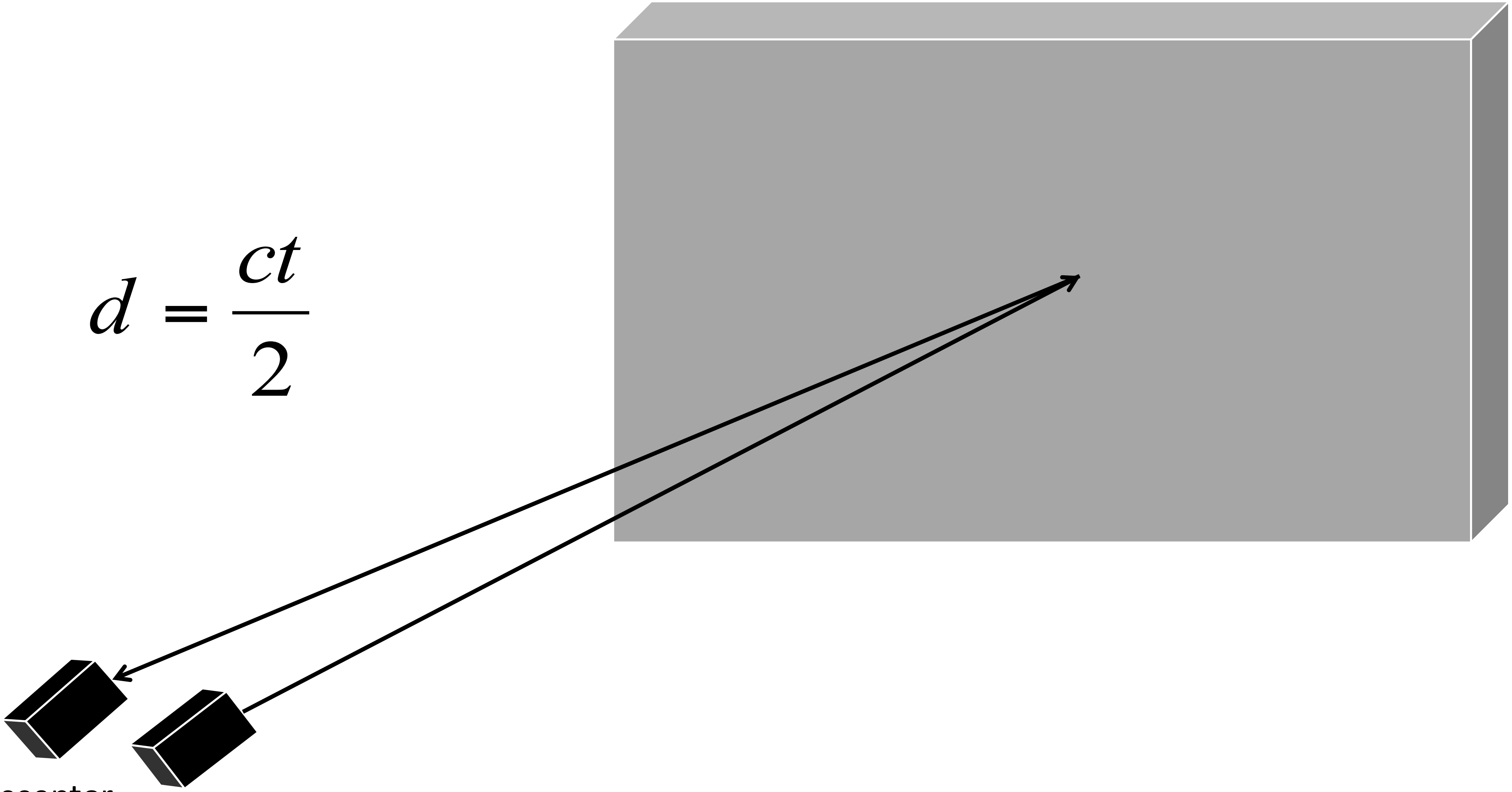

**Light Coding**




# Light Coding


3D Depth Sensors
RGB Camera

$$d = \frac{ct}{2}$$




**Time of Flight**

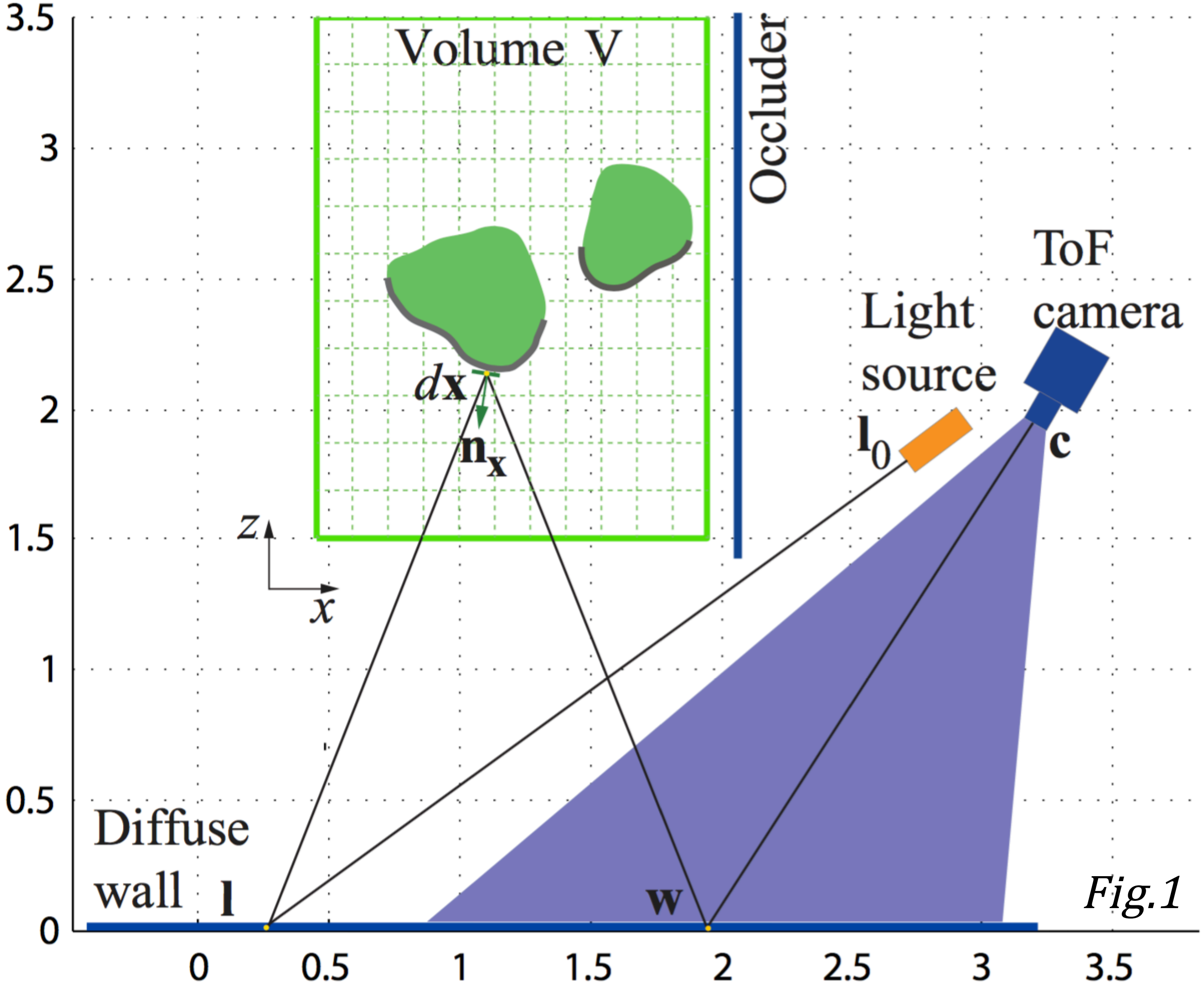


*Fig.1*

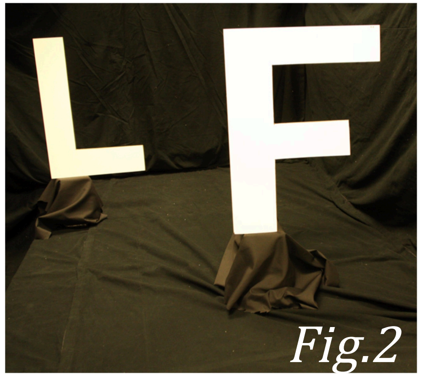


*Fig.2*

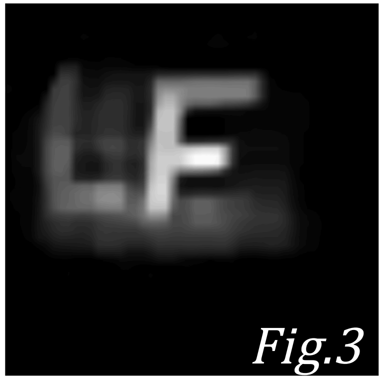


*Fig.3*

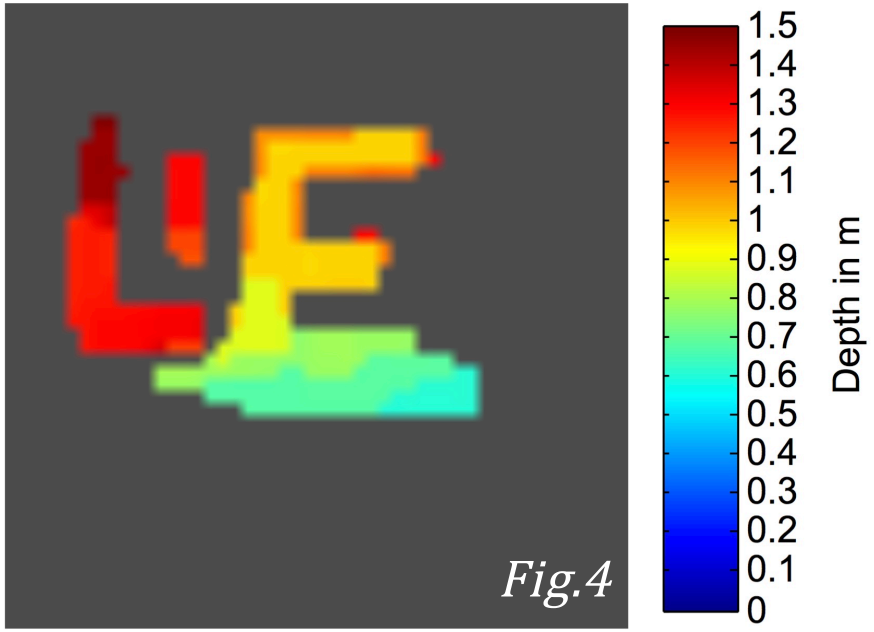


*Fig.4*

Felix H, *et al*. Diffuse Mirrors: 3D Reconstruction from Diffuse Indirect Illumination Using Inexpensive Time-of-Flight Sensors. *ICCV*, 2014.

# Passive methods

# Photos

## Monocular vision

One camera

## Binocular vision

Two parallel camera

# Structure from Motion

two-dimensional image sequences

three-dimensional structures

# Structure from Motion

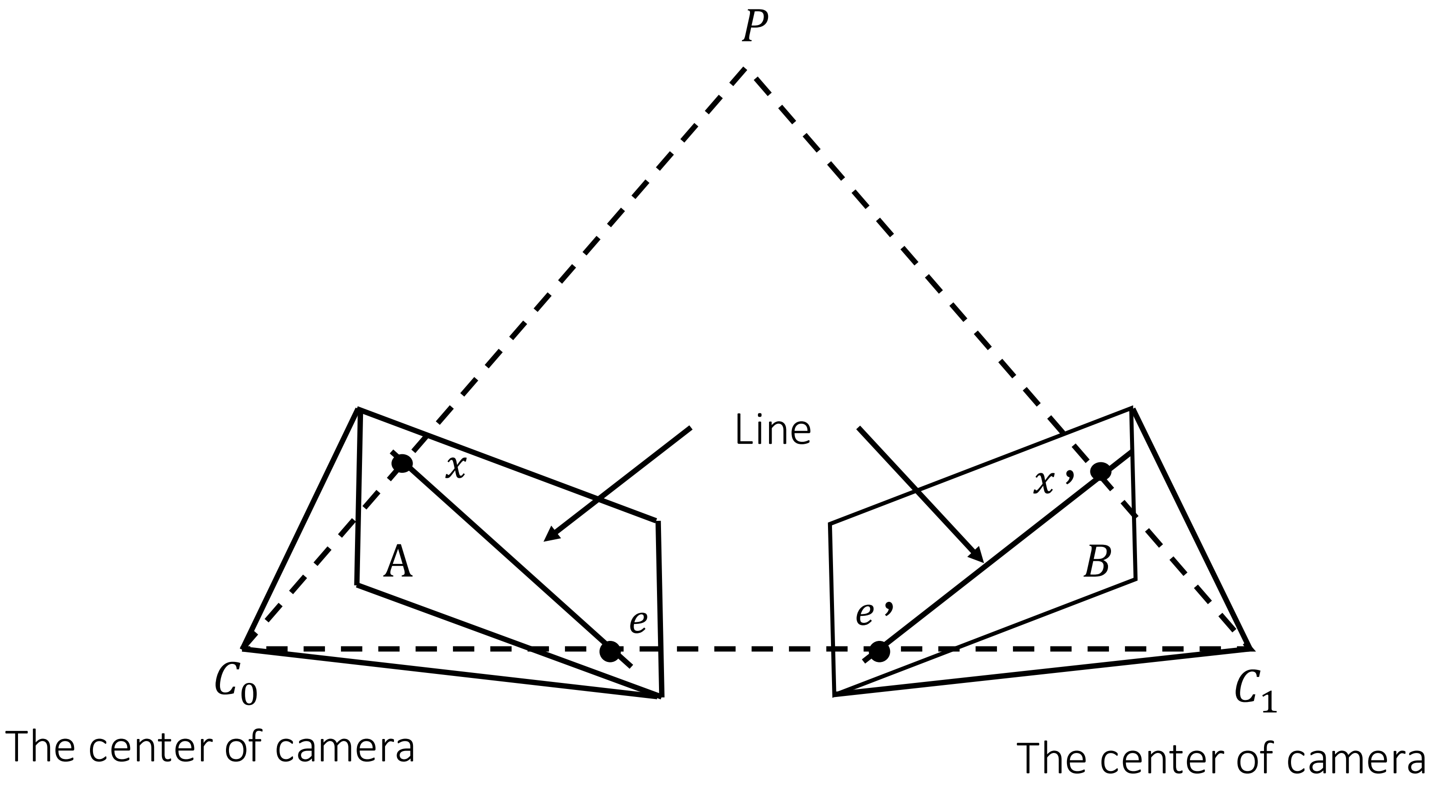
**advantage**

1. Calibration
2. Acquisition
3. Scene

**disadvantage**

1. Arithmetic
2. Accuracy
3. Time

# Introduction



# Introduction

Image acquisition → 

Sparse reconstruction

Dense reconstruction

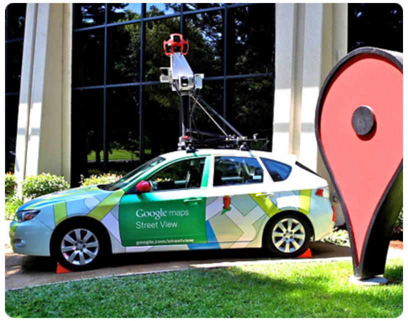
Surface reconstruction

# Acquisition

UAV


Digital camera

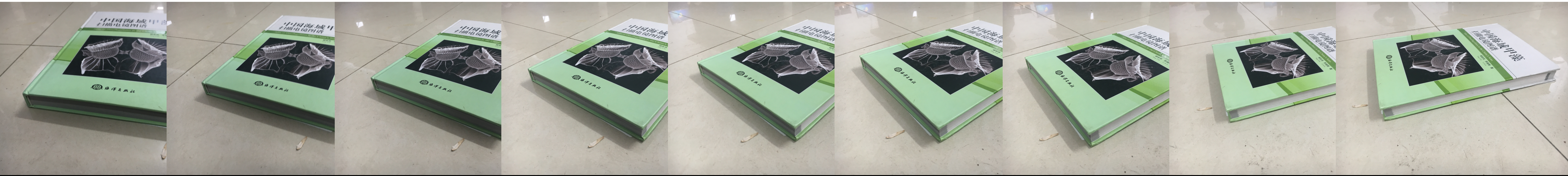
Camera



Street car

Mobile phone


中国海域甲藻
扫描电镜图谱
Atlas of Dinoflagellates in the China's Seas
杨世民 李瑞香 著



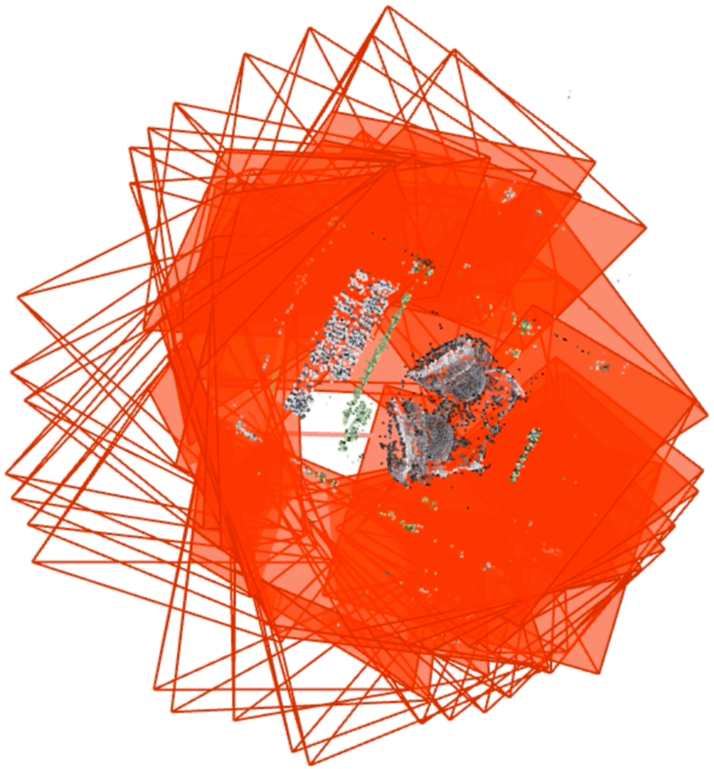
vision@ouc

# Sparse reconstruction

Feature point matching

Calculation of camera parameters

Bundle adjustment



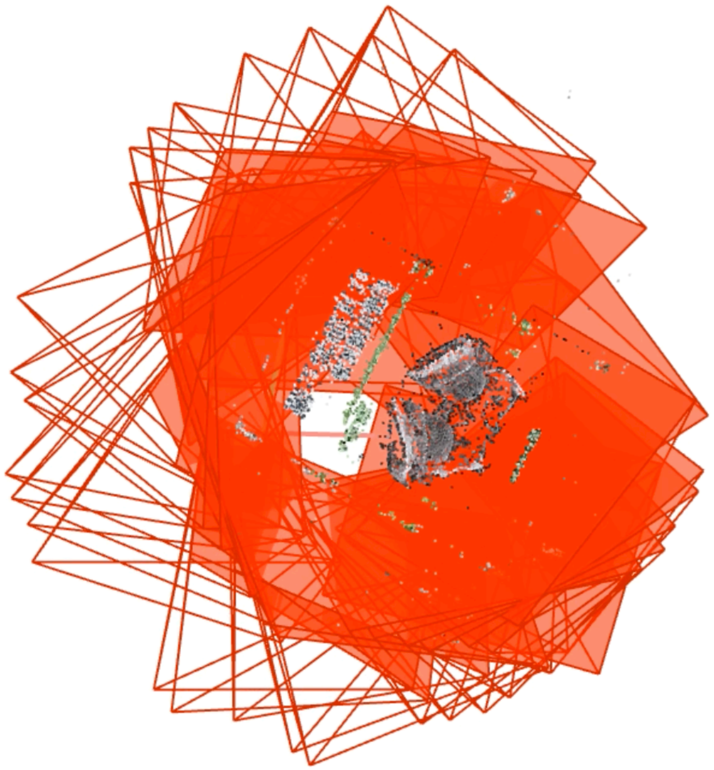
The generated process of Sparse reconstruction

# Sparse reconstruction

Feature point matching

Calculation of camera parameters

Bundle adjustment



The generated process of Sparse reconstruction

# Feature point matching

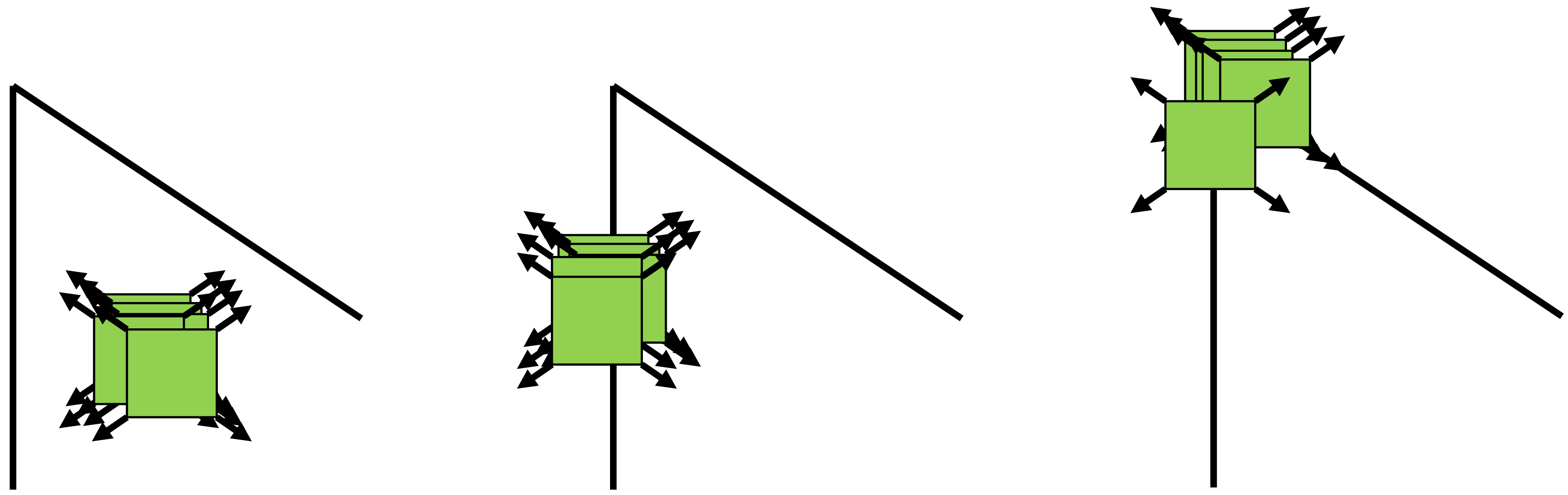
Harris corner detection[1]

SIFT（Scale-invariant feature transform）[2]

SURF（Speeded Up Robust Features）[3]

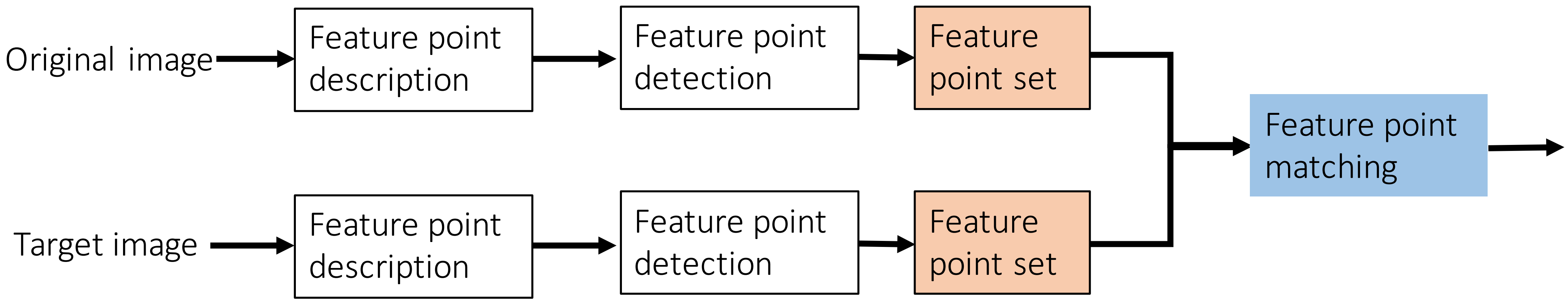
[1] Chris H and Stephens M. A combined corner and edge detector. *Alvey vision conference*. 1988.
[2] Lowe D G. Object recognition from local scale-invariant features. *Computer vision, 1999.*
[3] Bay H , Tinne T, and Luc V G. Surf: Speeded up robust features. *ECCV*, 2006.

# Harris corner detection



Chris H and Stephens M. A combined corner and edge detector. *Alvey vision conference*. 1988.

# SIFT




Lowe D G. Object recognition from local scale-invariant features. *Computer vision, 1999.*

# SIFT vs SURF

The establishment of scale space

The extraction of feature points

The generation of feature descriptors

The matching of feature points

similarity

The way of convolution

The detection of feature points

The way of finding direction

The descriptor of feature point

difference

# Comparison

| Method | Time | Scale | Rotation | Blur | Illumination | Affine |
| --- | --- | --- | --- | --- | --- | --- |
| SIFT | common | best | best | best | common | good |
| PCA-SIFT | good | common | good | common | good | good |
| SURF | best | good | common | good | best | good |

Graffiti dataset

Juan L, Gwun O. A comparison of sift, pca-sift and surf. *International Journal of Image Processing*, 2009.

# Calculation of camera parameters

3D reconstruction

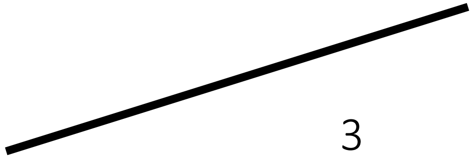
motion estimation

fundamental matrix

self-calibration

Mapping & tracking

# The redundancy of feature point



3

5

[1] Hartley R I. In defense of the eight-point algorithm. *IEEE Transactions on pattern analysis and machine intelligence*, 1997.
[2] Torr P H , and W. Murray D. The development and comparison of robust methods for estimating the fundamental matrix. *International journal of computer vision* ,1997.

# Bundle adjustment

iterative optimization

parameter optimization → nonlinear optimization

Bill T, *et al*. Bundle adjustment—a modern synthesis. *International workshop on vision algorithms*. 1999.

# Bundler Noah Snavely

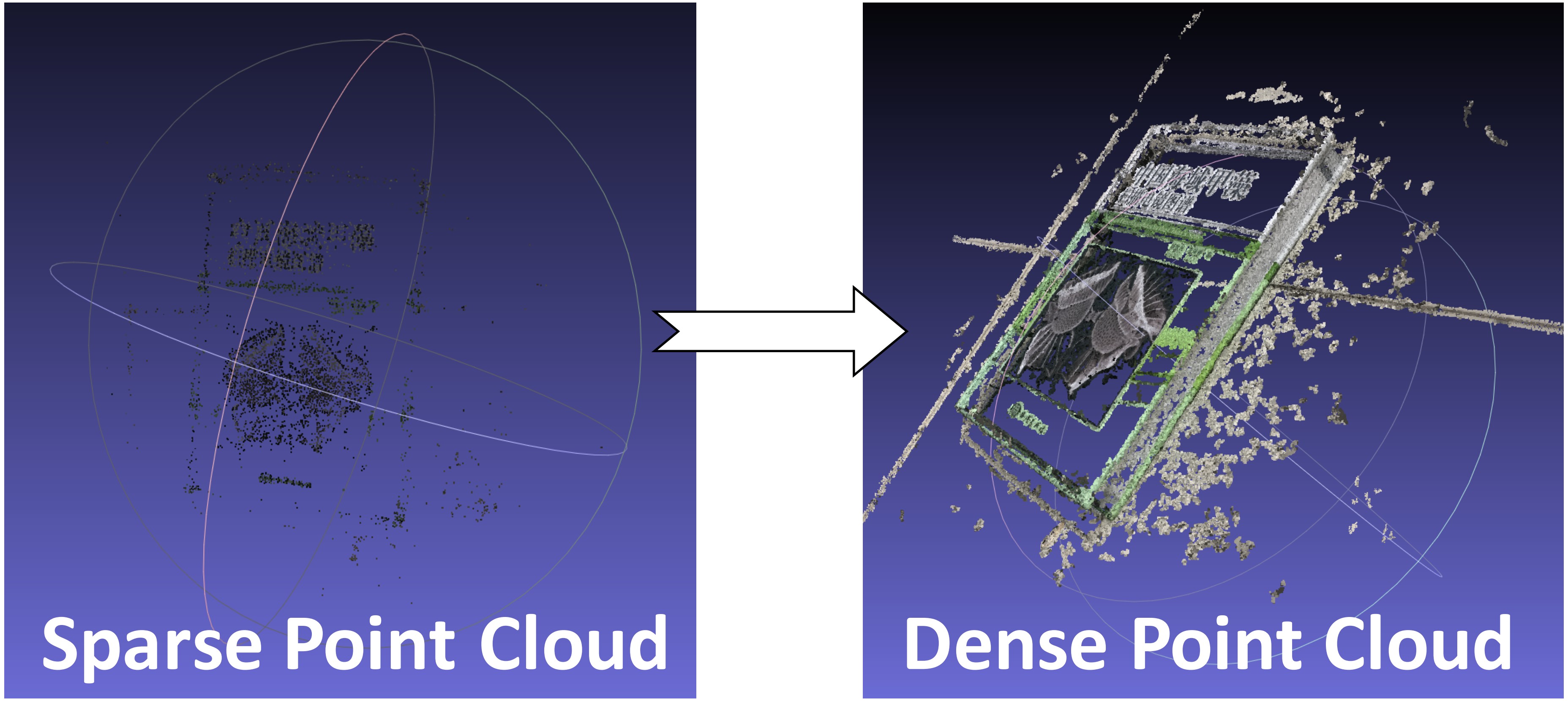
C → Sparse point clouds

C++ → Camera parameters

Introduction: http://www.cs.cornell.edu/%7Esnavely/bundler
Code: https://github.com/snavely/bundler_sfm

# Video

# Dense reconstruction



# Dense reconstruction

light and shade[1]

geometry

surface profile[2]

[1] Durou J D, Falcone M, and Sagona M. Numerical methods for shape-from-shading: A new survey with benchmarks. *Computer Vision and Image Understanding* ,2008.
[2] Boyer E, and Franco J S. A hybrid approach for computing visual hulls of complex objects. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*,2003.

# PMVS2 Yasutaka Furukawa& Jean Ponce

Sparse point clouds
Camera parameters

Dense point clouds

Introduction & code: http://www.di.ens.fr/pmvs

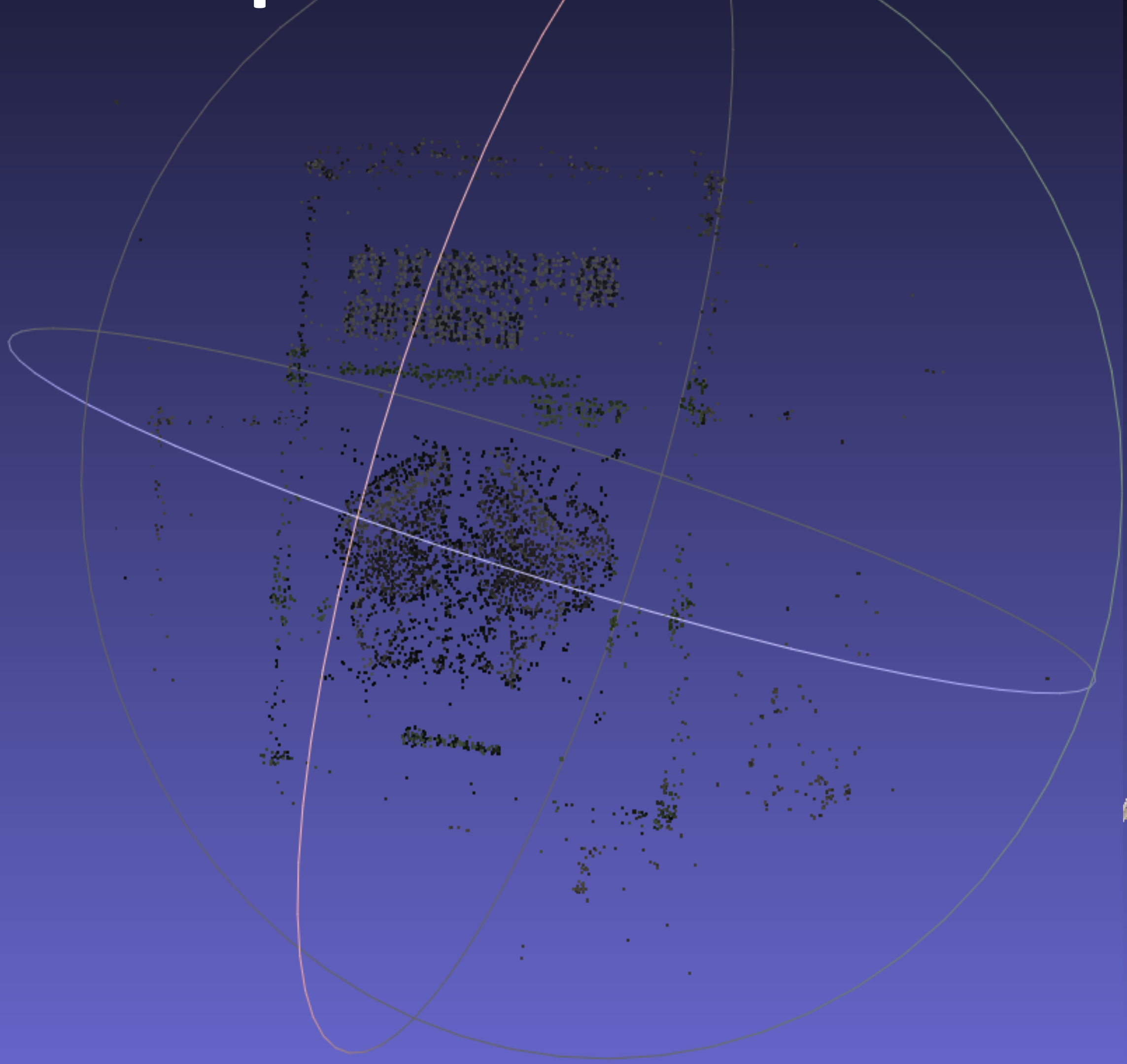
# Meshlab

Open source software

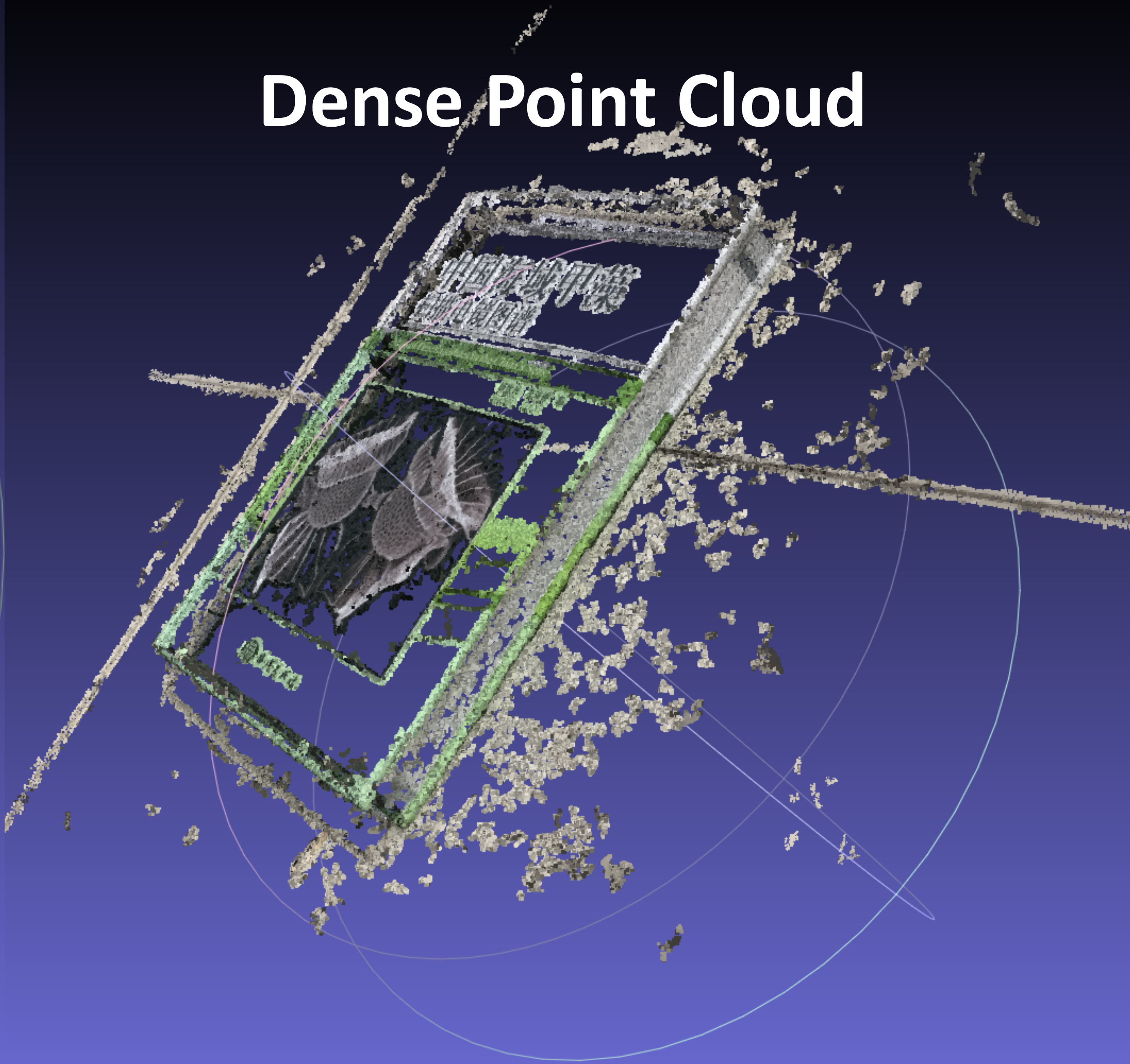
3D reconstruction processing tool

http://www.meshlab.net

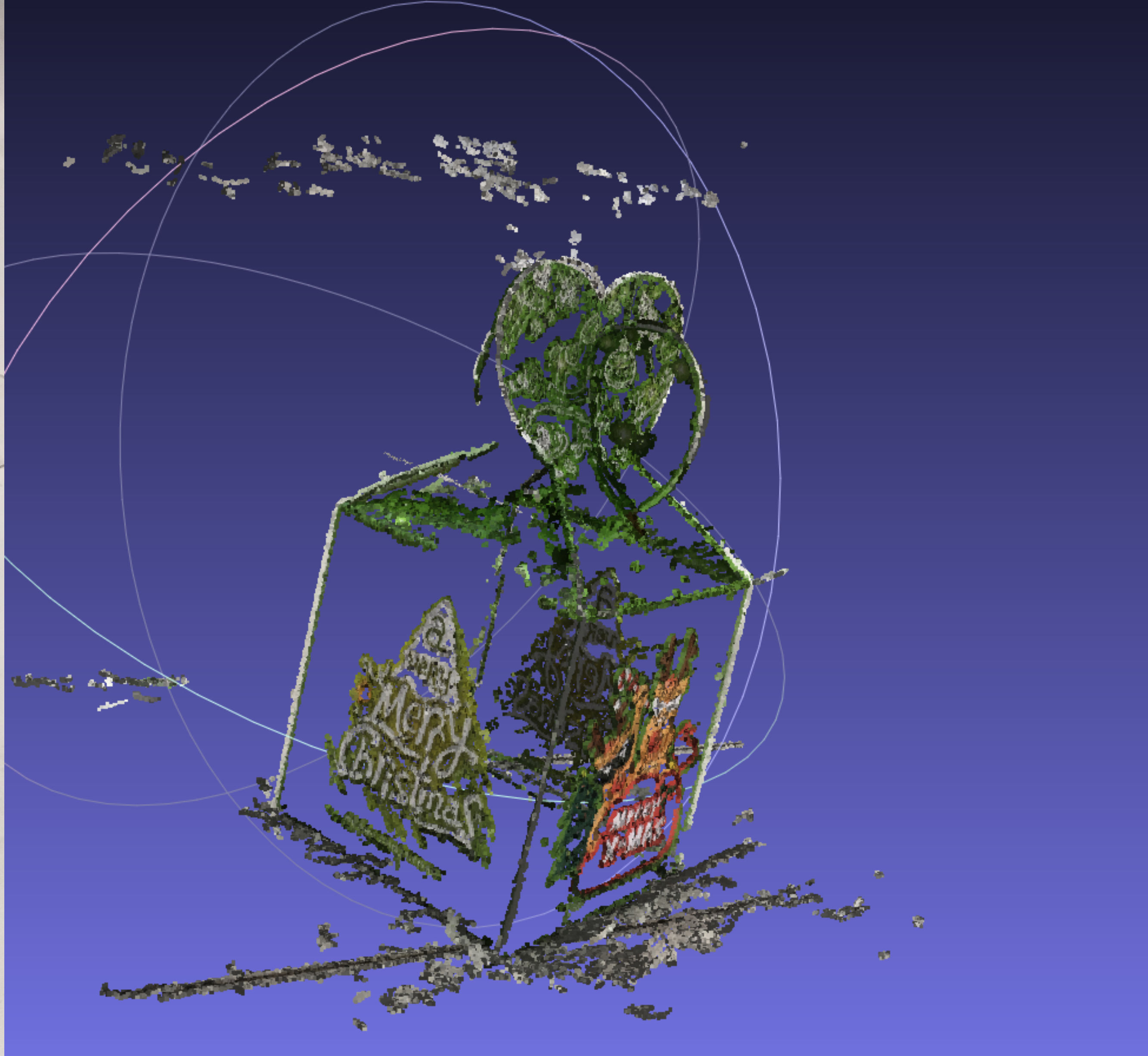
## Sparse Point Cloud



## Dense Point Cloud

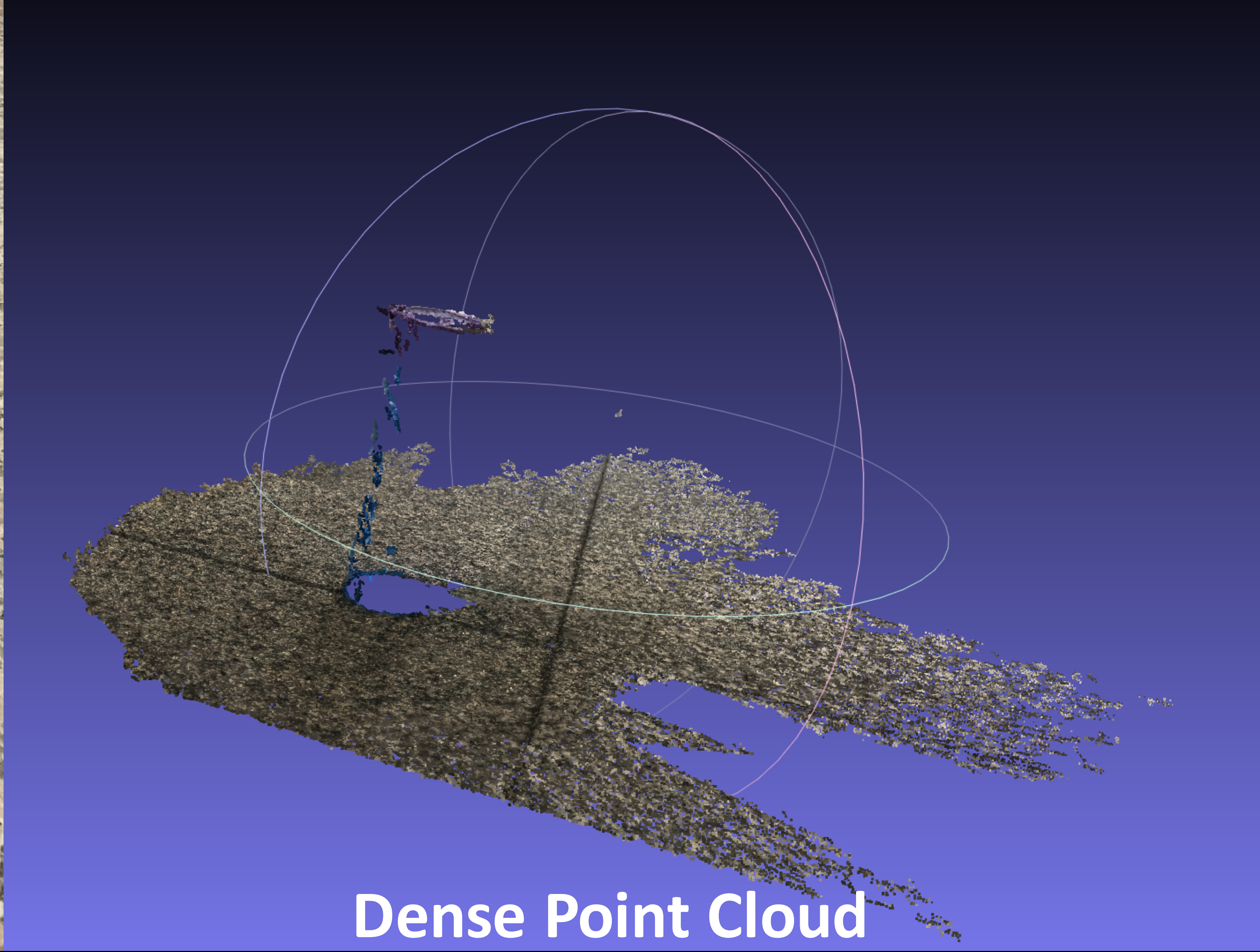



RGB

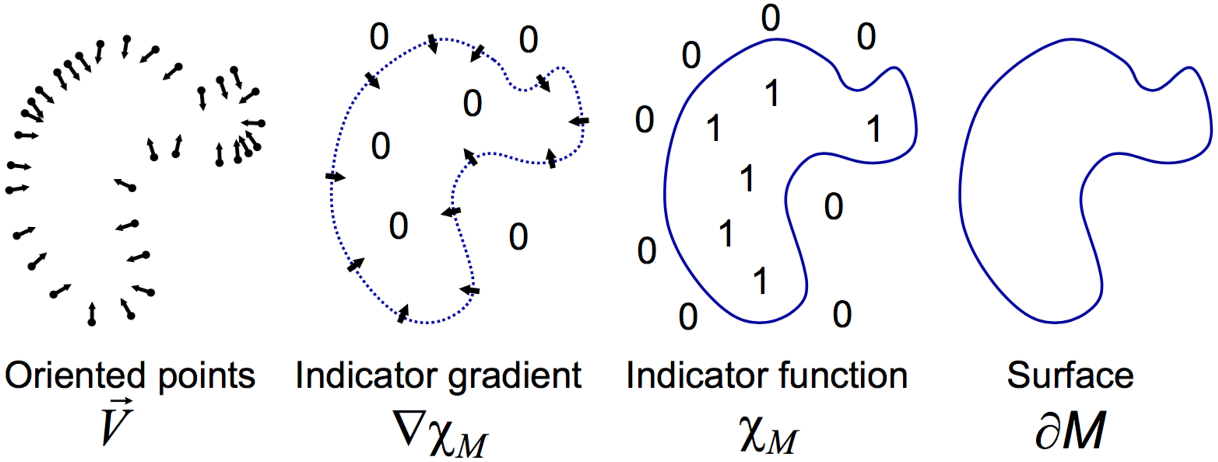


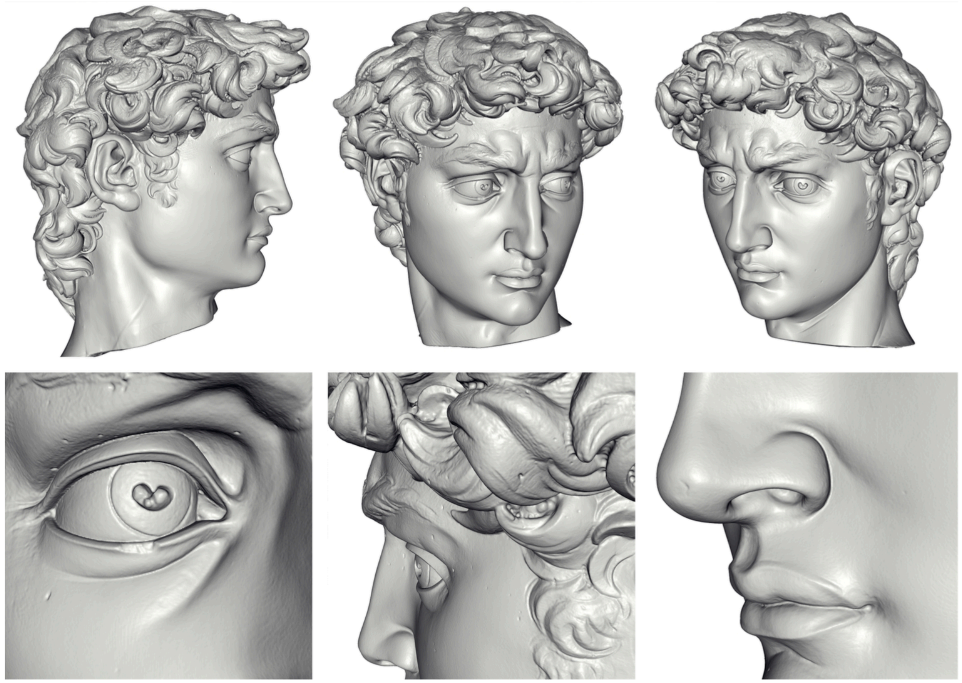
Dense Point Cloud


RGB


Dense Point Cloud

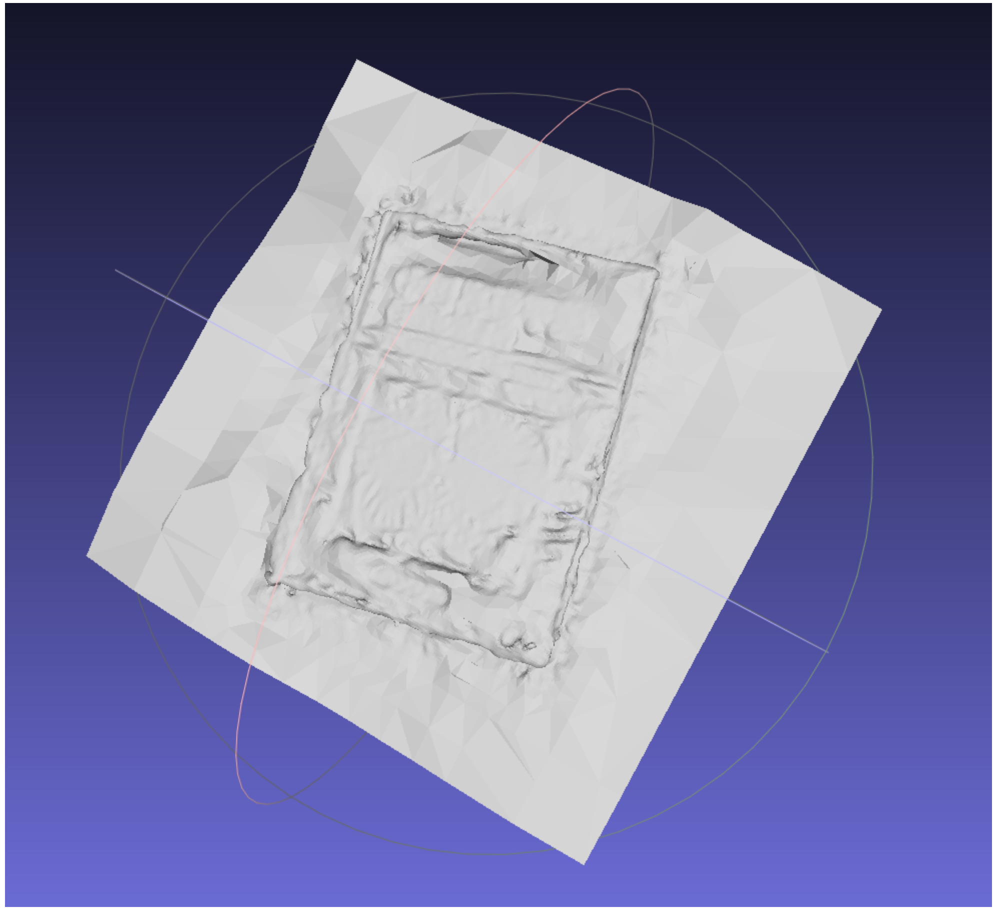
# Poisson Surface Reconstruction

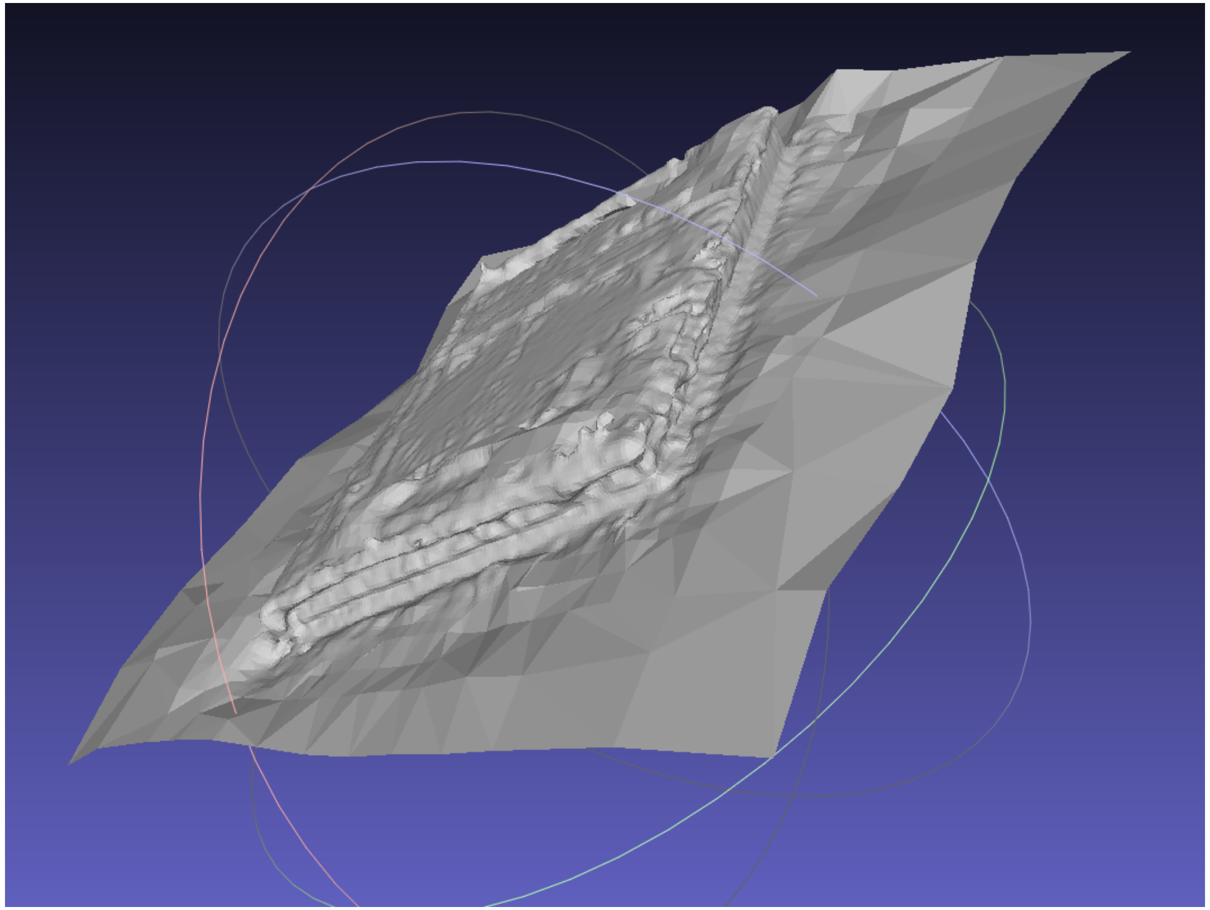


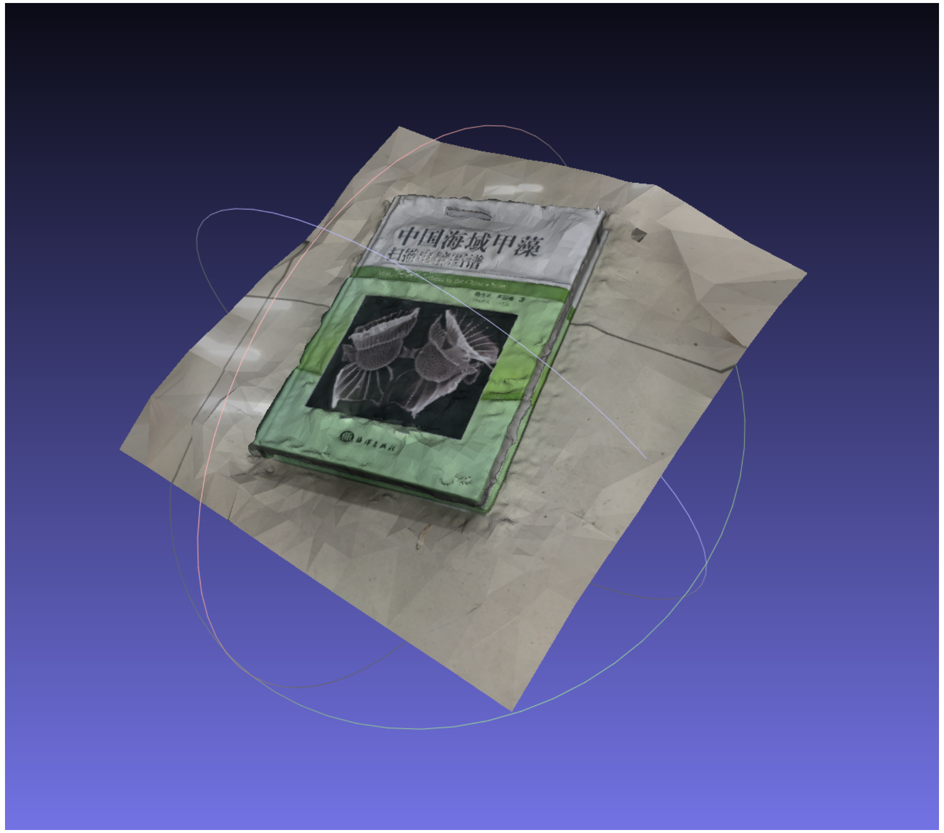
Kazhdan M, Hoppe H. Screened poisson surface reconstruction. *ACM Transactions on Graphics (TOG),* 2013.

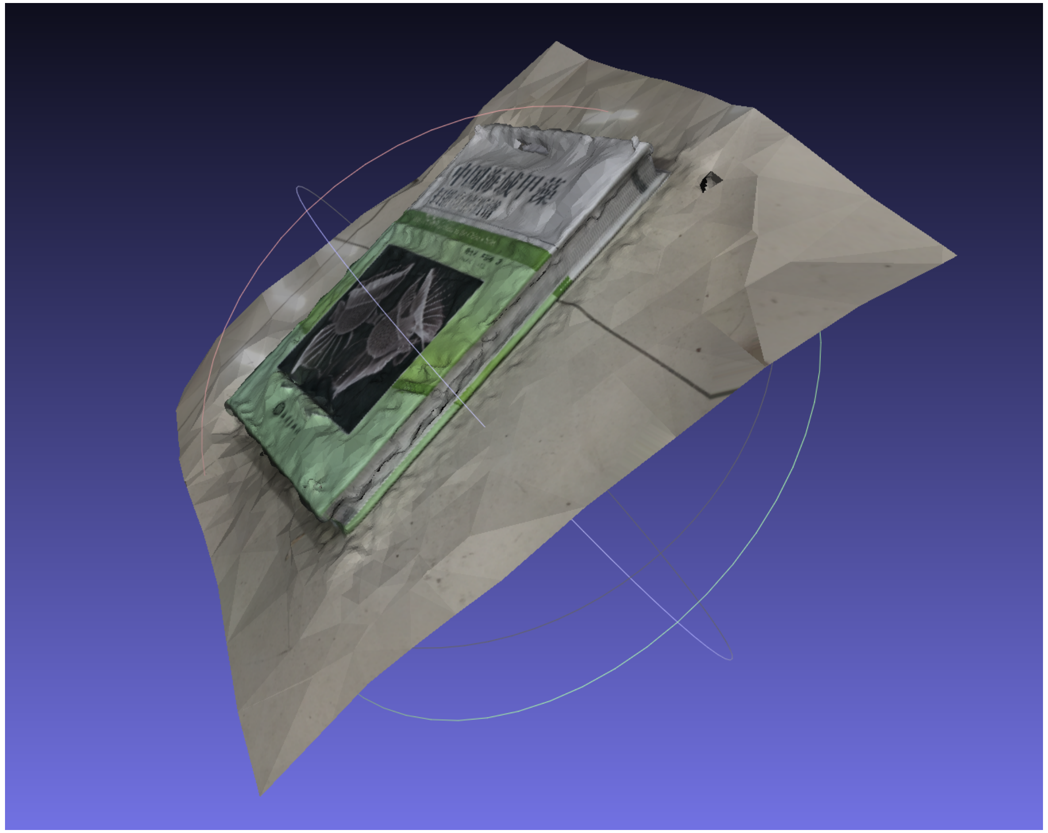
# Result



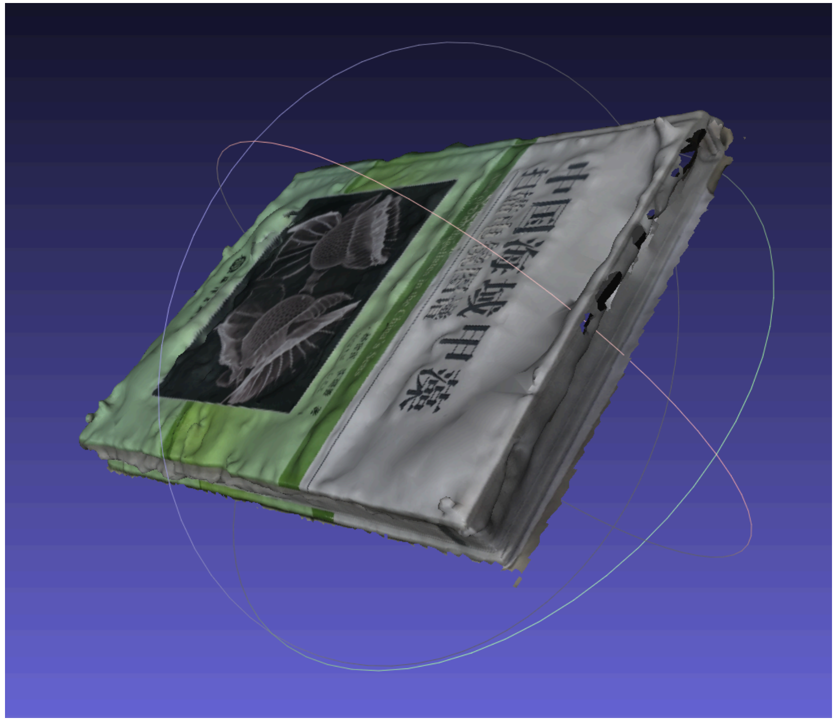


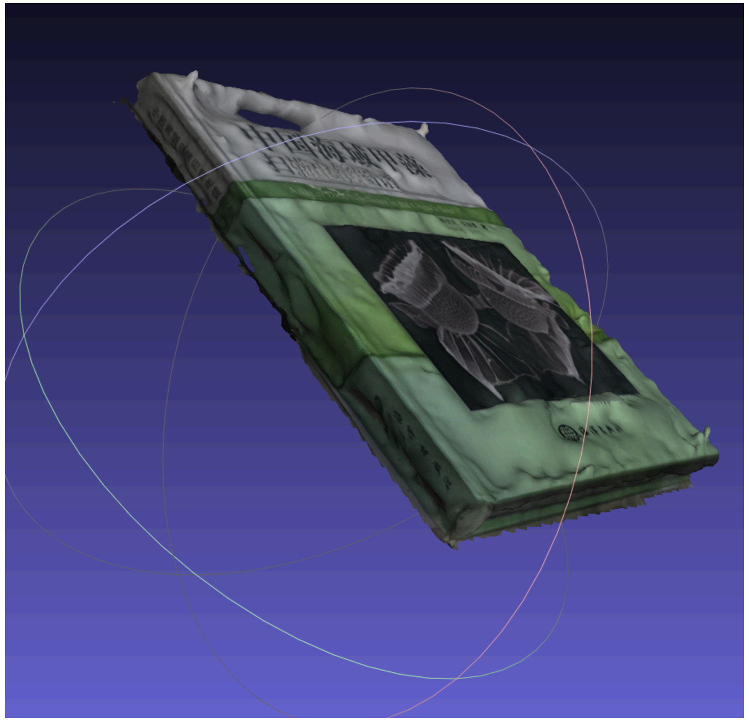
# Texture mapping

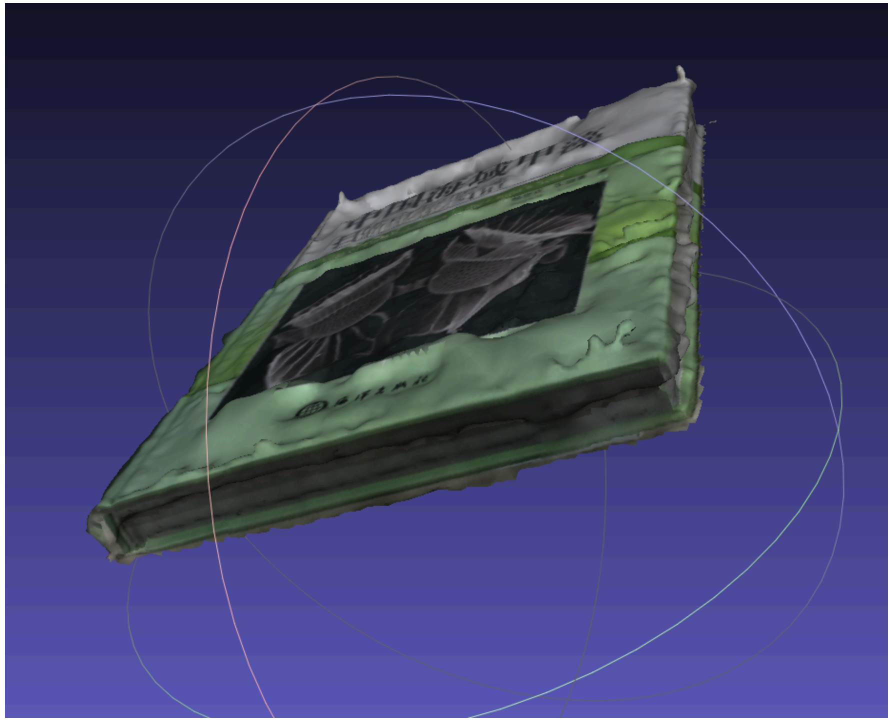


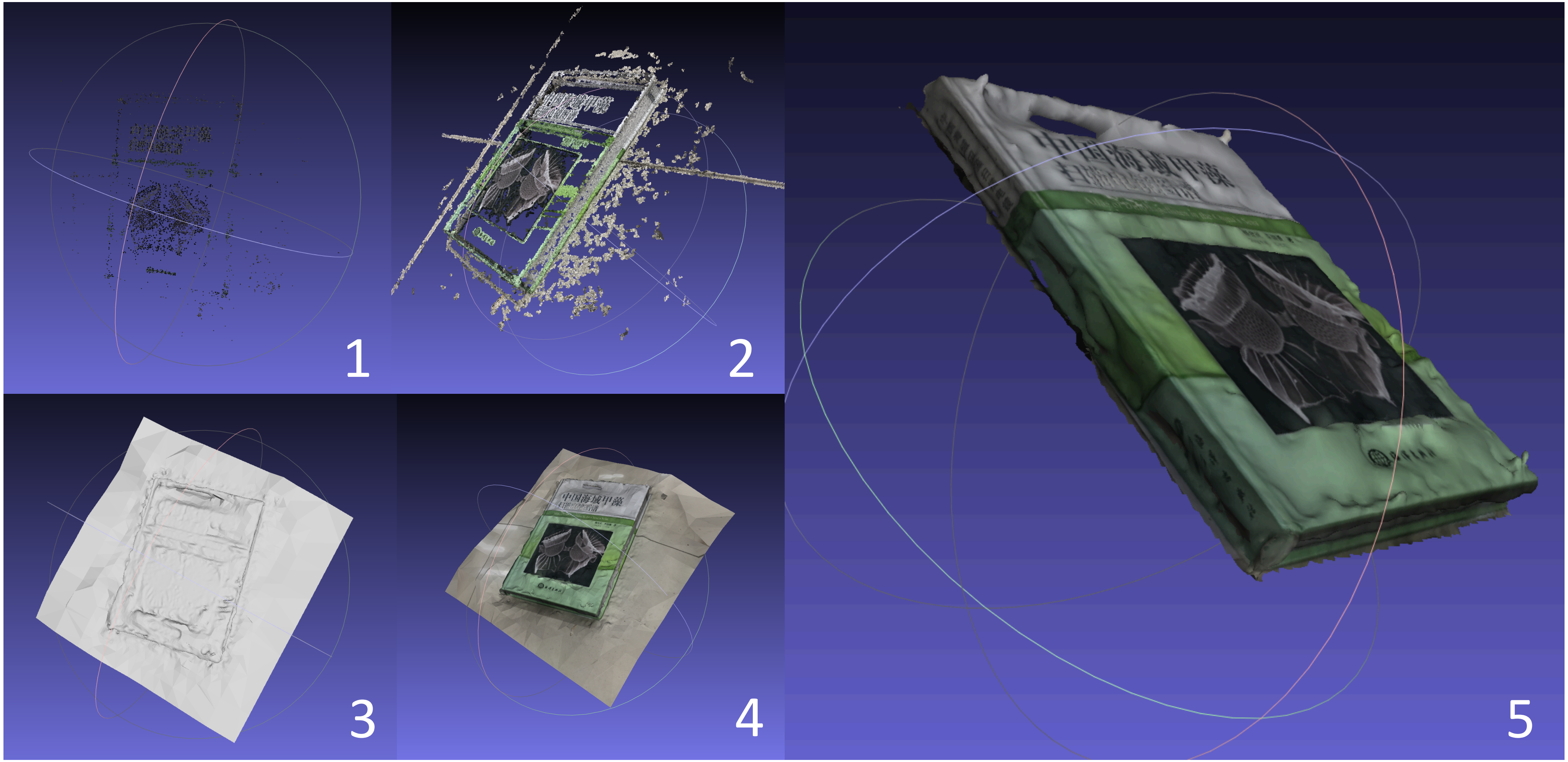



# Final result



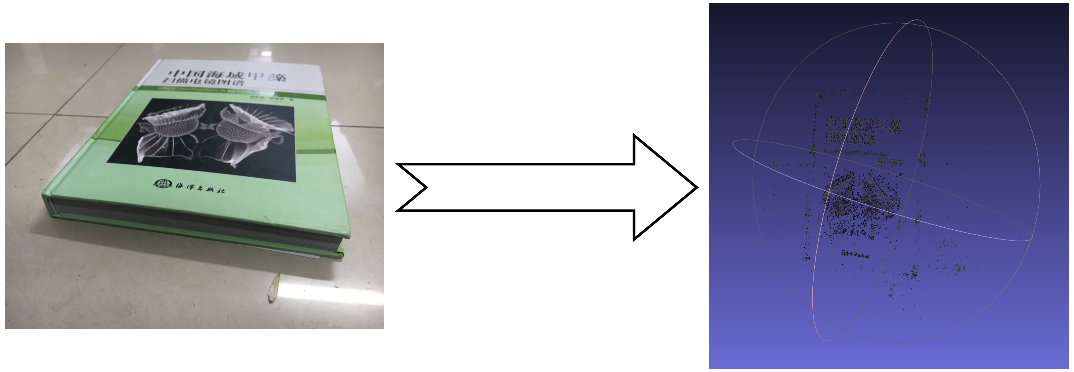





1
2
3
4
5

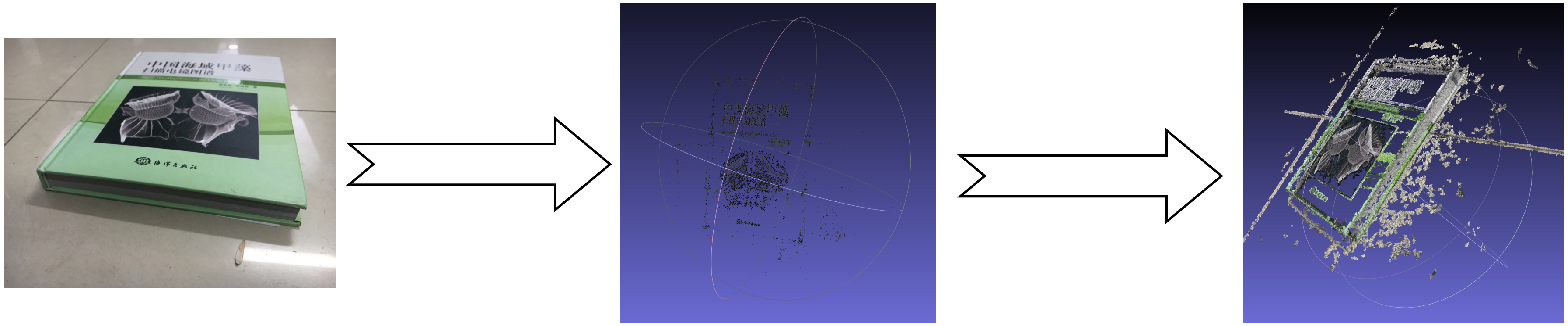
# Steps

| | | |
|---|---|---|
| | Images → Points: | Structure from Motion |
| | Points → More points: | Multiple View Stereo |
| | Points → Meshes: | Model Fitting |
| + | Meshes → Models: | Texture Mapping |
| = | Images → Models: | Image-based Modeling |



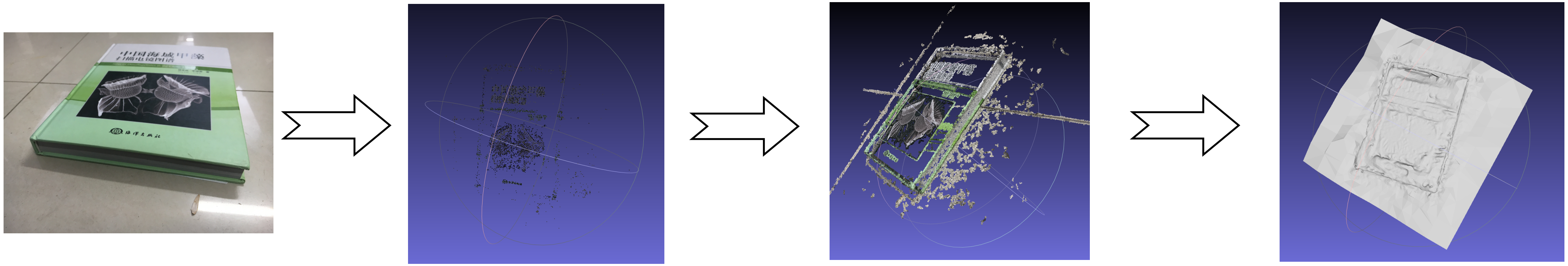
# Steps

| | | |
|---|---|---|
| | Images → Points: | Structure from Motion |
| | Points → More points: | Multiple View Stereo |
| | Points → Meshes: | Model Fitting |
| + | Meshes → Models: | Texture Mapping |
| = | Images → Models: | Image-based Modeling |



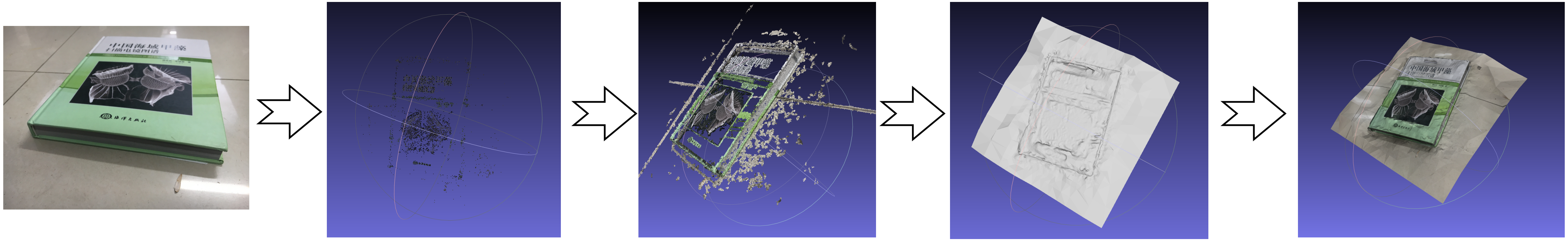
# Steps

| | | |
|---|---|---|
| | Images → Points: | Structure from Motion |
| | Points → More points: | Multiple View Stereo |
| | Points → Meshes: | Model Fitting |
| + | Meshes → Models: | Texture Mapping |
| = | Images → Models: | Image-based Modeling |



# Steps

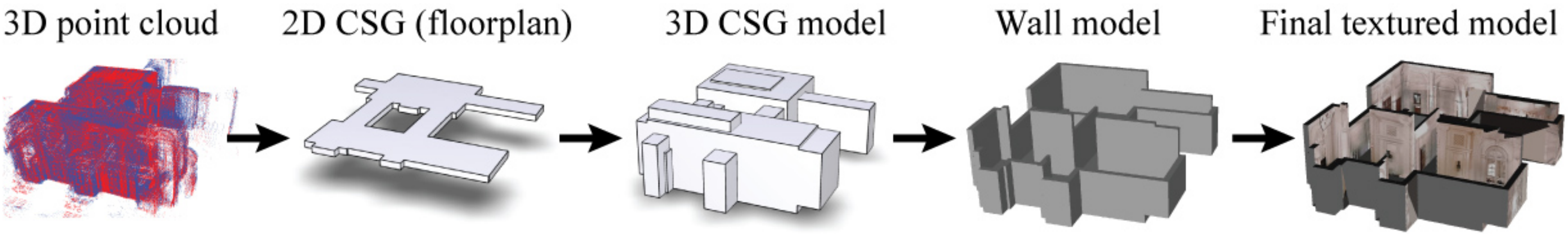
| | | |
|---|---|---|
| | Images → Points: | Structure from Motion |
| | Points → More points: | Multiple View Stereo |
| | Points → Meshes: | Model Fitting |
| + | Meshes → Models: | Texture Mapping |
| = | Images → Models: | Image-based Modeling |



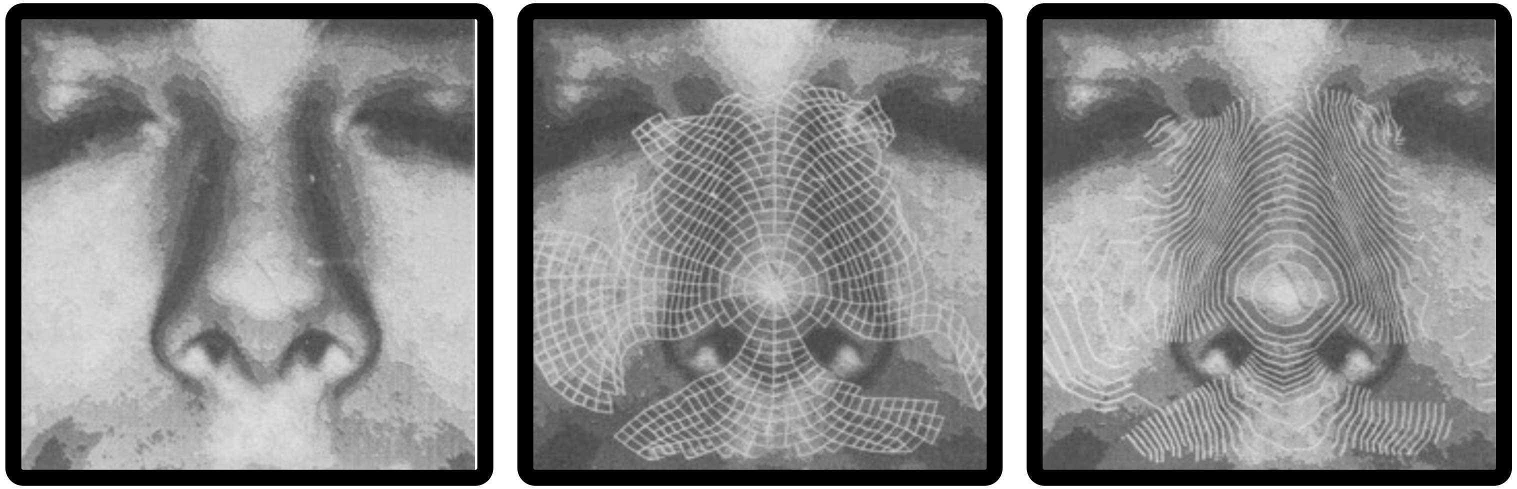
# Other tools

- Surface Reconstruction
  - Marching Cubes: http://en.wikipedia.org/wiki/Marching_cubes
  - Poisson Surface Reconstructionhttp://research.microsoft.com/en-us/um/people/hoppe/proj/poissonrecon/
- Model Fitting
  - RANSAC & J-linkage http://www.diegm.uniud.it/fusiello/demo/jlk/
  - InverseCSG
  - GlobFit http://code.google.com/p/globfit/
  - Face Model Fitting http://research.microsoft.com/apps/pubs/default.aspx?id=73211

# Application



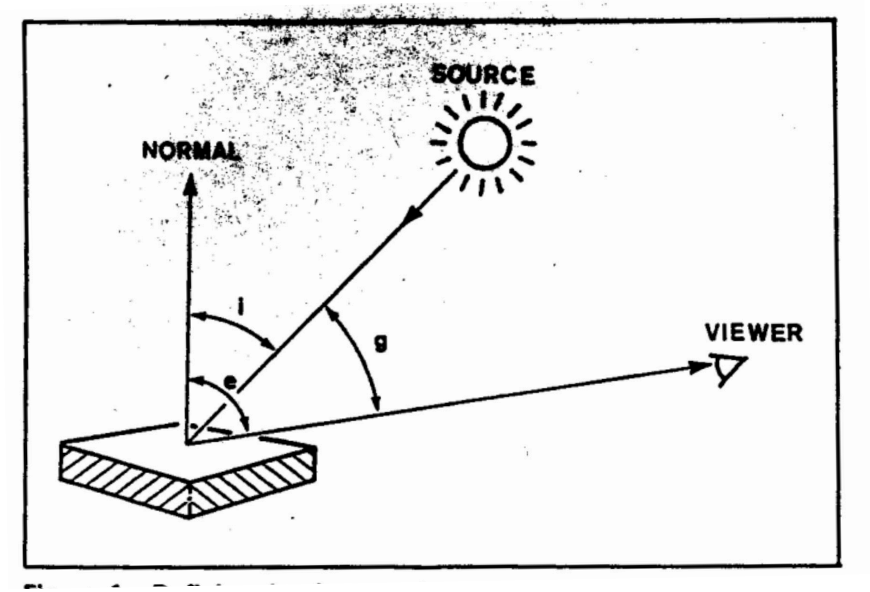
# Shape from Shading



1.Bronte model
2. Infinite point
3. Rectangular projection

Berthold KP H. Shape from shading: A method for obtaining the shape of a smooth opaque object from one view,1970.

# Photometric Stereo

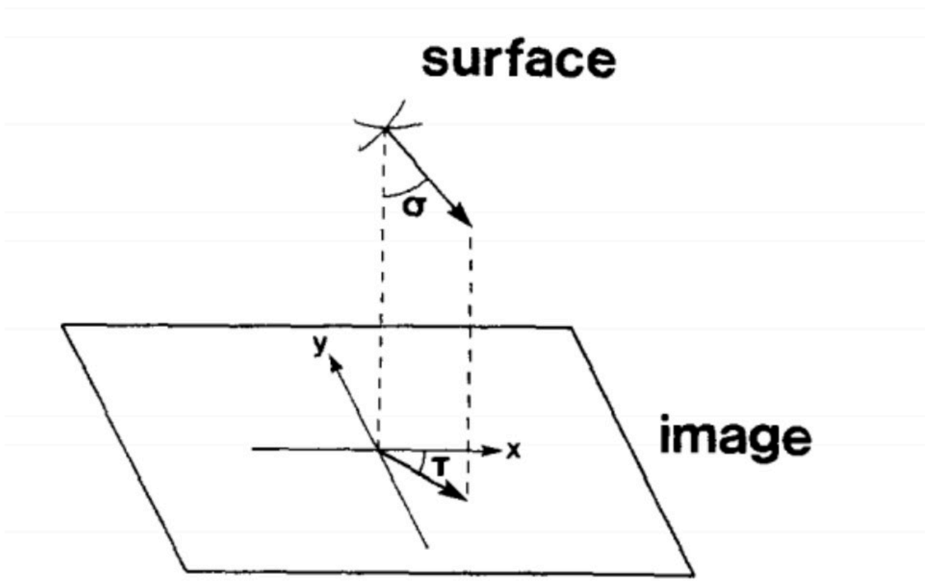



Nonlinearity and Noise Reduction 2003;
Gradient Control Point 2004;
Credibility Passed with Markov Random Field 2005;
Three - dimensional reconstruction of light conditions under unknown conditions in 2007;
Non - Lambert 2007;
Reconstruction of colored light in 2007;

Woodham R J. Photometric method for determining surface orientation from multiple images. *Optical engineering*,1980.

# Other methods

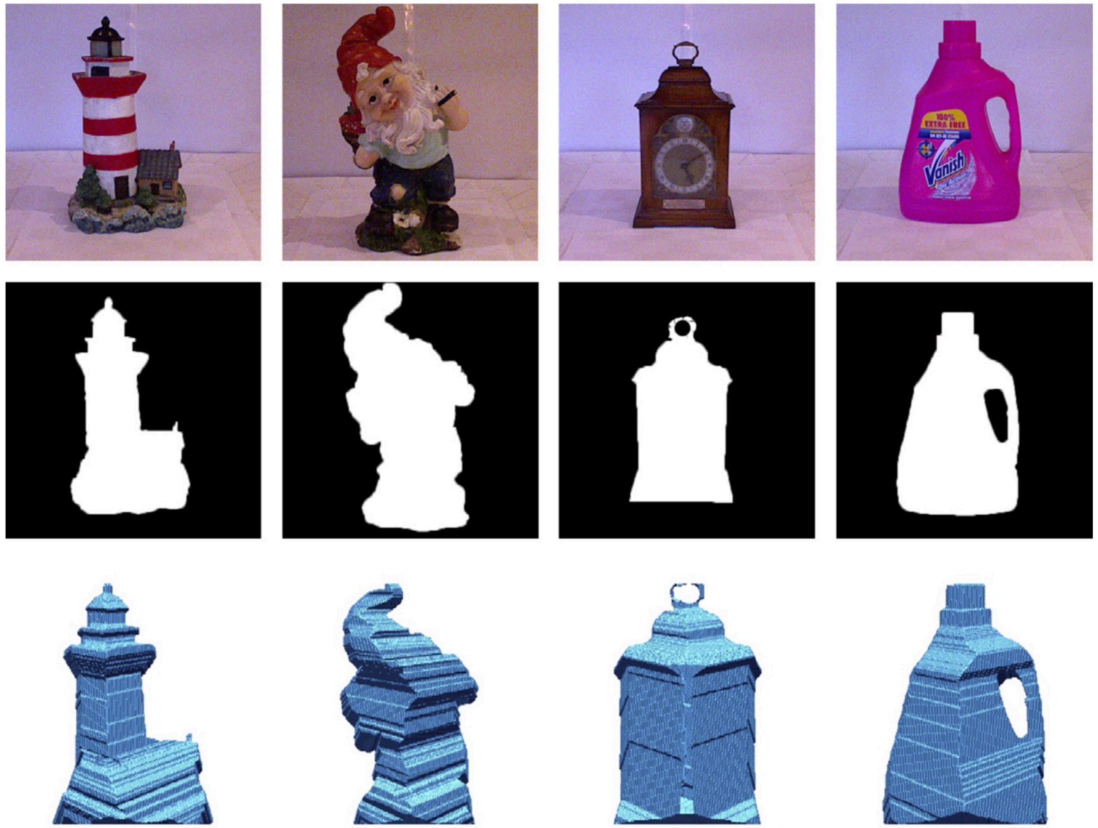
## Shape from Texture





Witkin A P. Recovering surface shape and orientation from texture. *Artificial intelligence*, 1981.

# Other methods



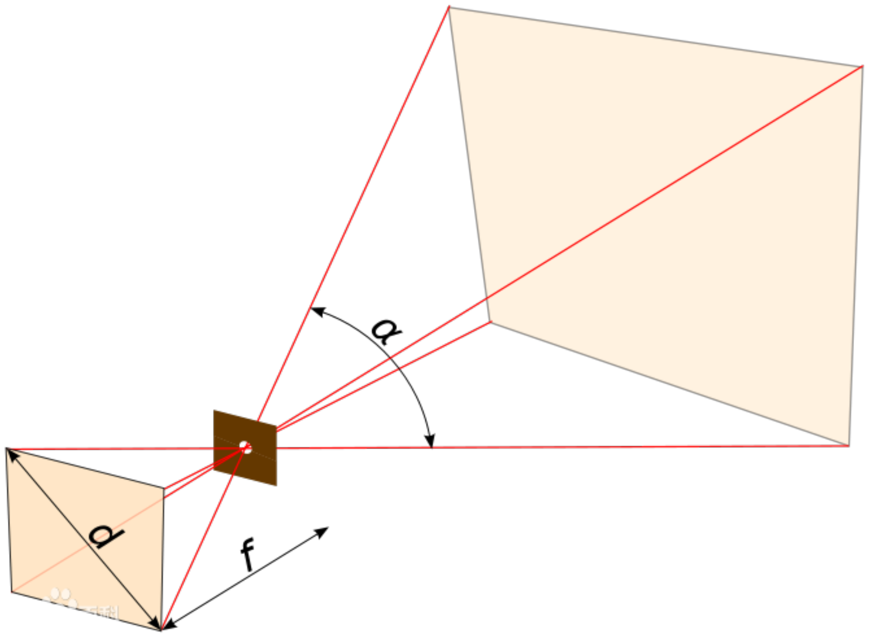

## Shape from Silhouettes

Divide the three-dimensional space into voxels

Martin W N, and Jagdishkumar K A . Volumetric descriptions of objects from multiple views. *IEEE transactions on pattern analysis and machine intelligence*, 1983.

# Other methods

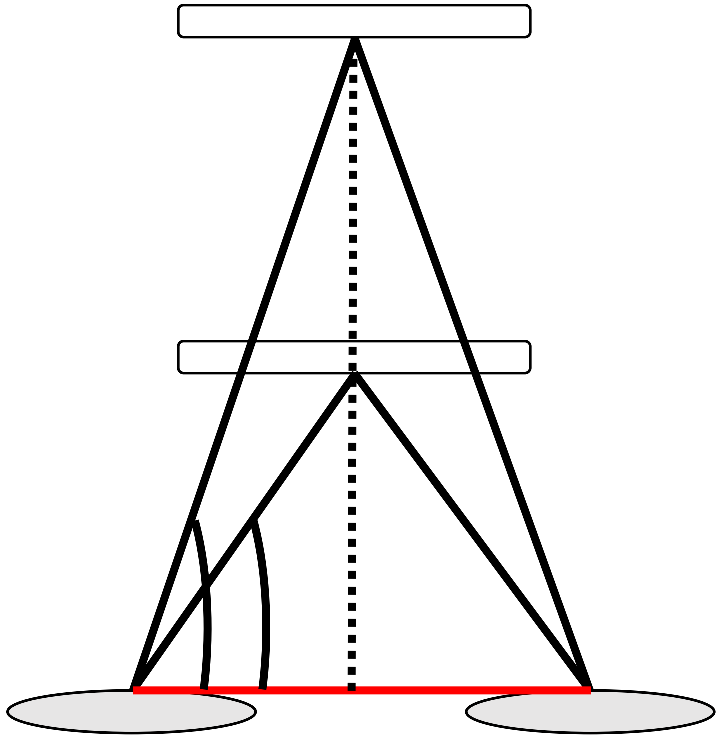
## Shape from Focus




Rajagopalan A N, Chaudhuri S. Optimal selection of camera parameters for recovery of depth from defocused images. *CVPR, 1997.*

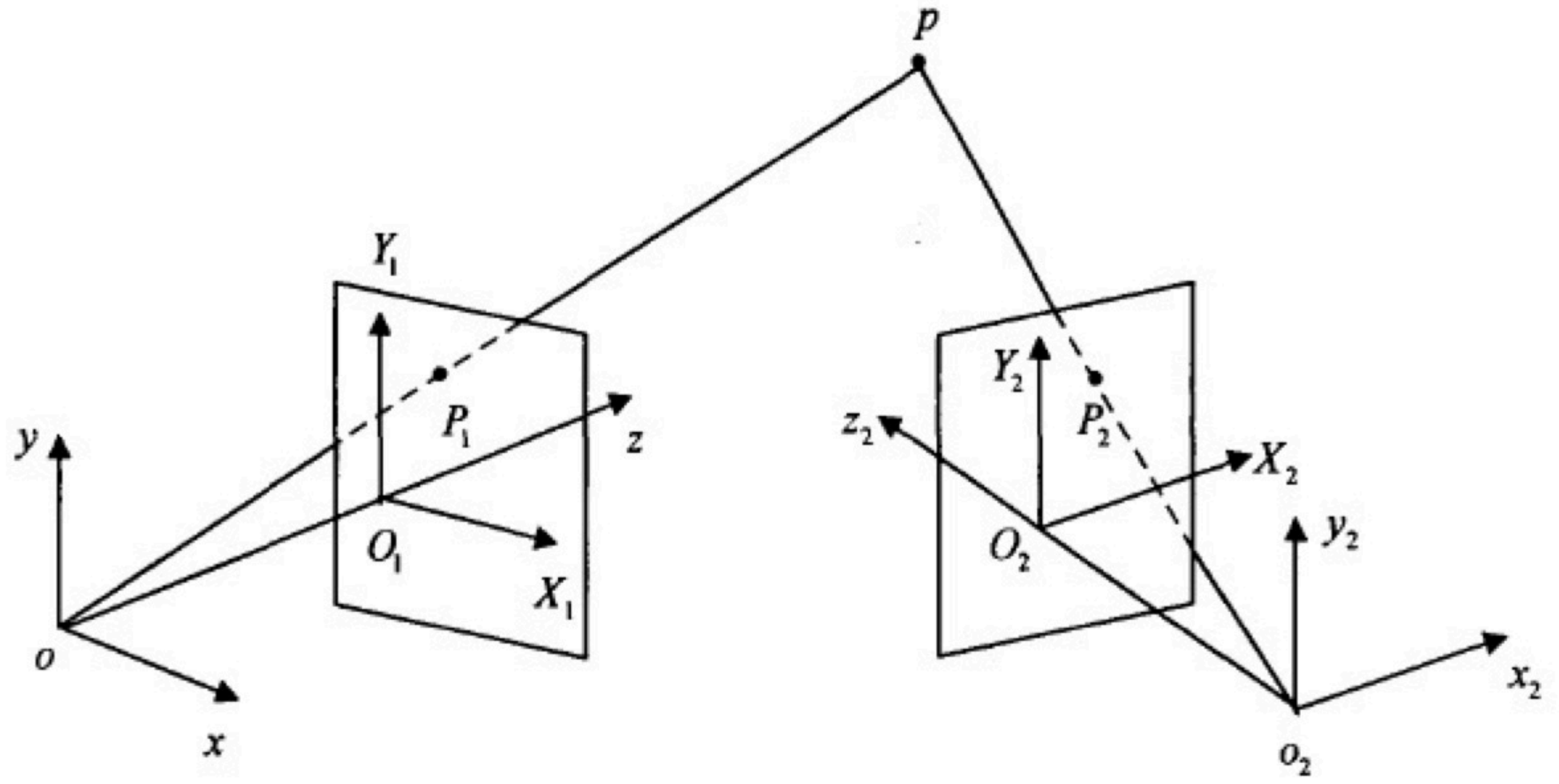
# Binocular vision

How to work？

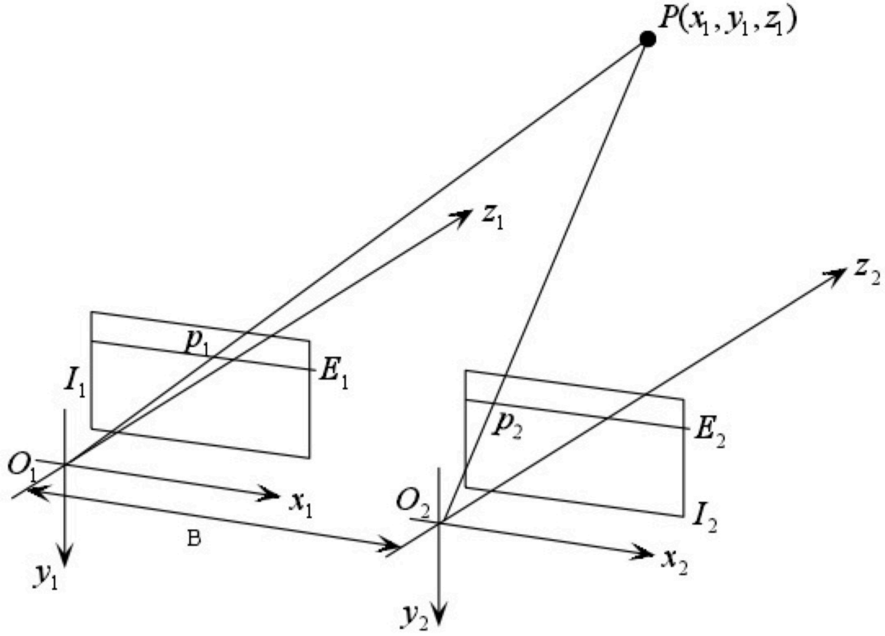


# GAME

# Binocular vision



# Parallel binocular stereo vision model




$$Z = \frac{Bf}{x_1 - x_2}$$

# Steps

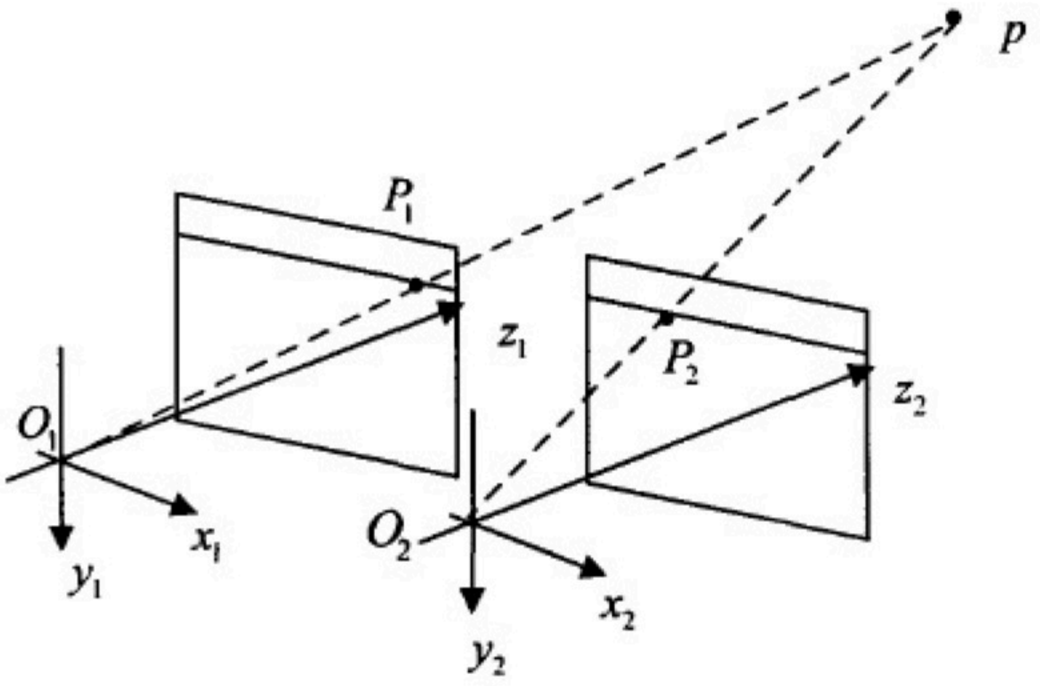
Calibration

Acquisition

Pretreatment

Stereo matching

Recovering



# Acquisition & Calibration

Traditional object-based calibration[1]

Self-calibration[2][3]


[1] Faugeras O. Three-Dimensional Computer Vision: A Geometric Viewpoint . *MIT Press*, 1993.
[2] Hartley R I. An algorithm for self calibration from several views. *CVPR*, 1994.
[3] Luong Q T. Self-calibration of a moving camera from point correspondences and fundamental matrices. *International Journal of computer vision* , 1997.

# Pretreatment

Image enhancement

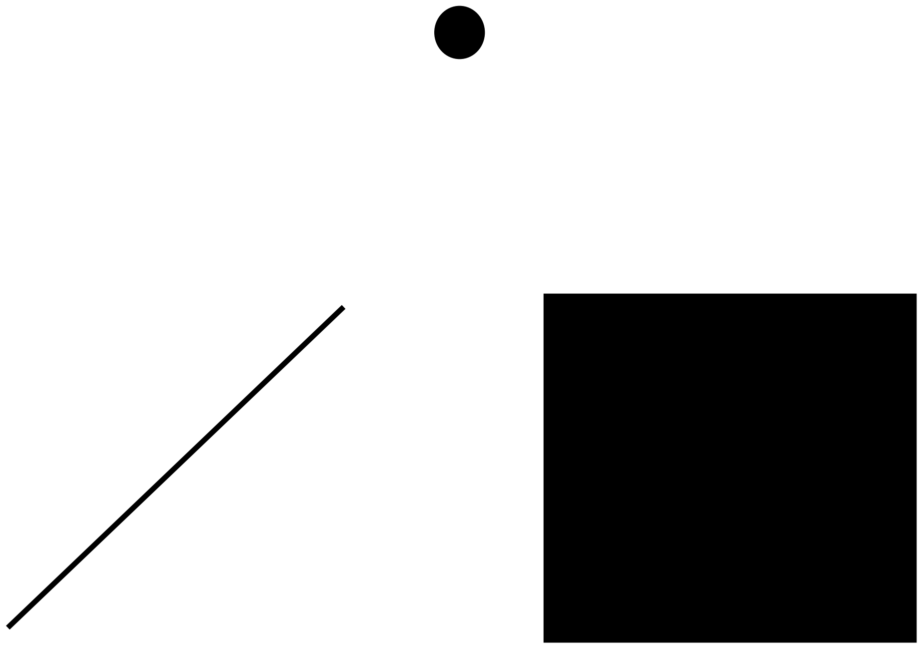
Geometric correction

Smoothing filtering

Image restoration

# Stereo matching

element



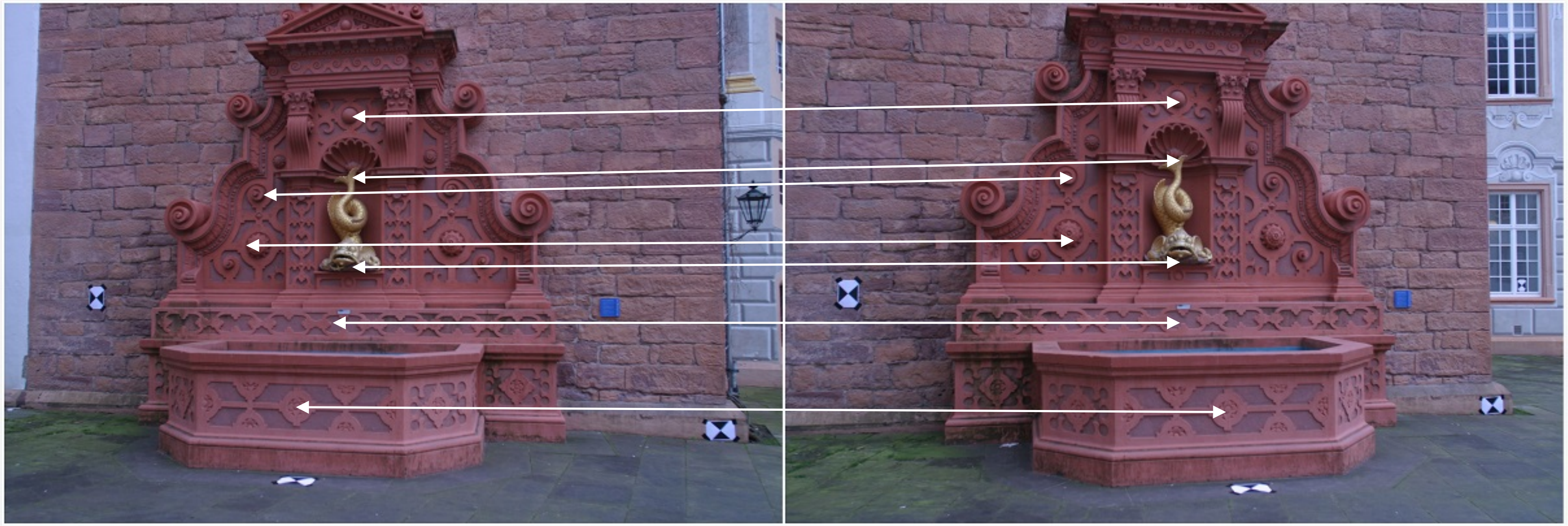
Gray matching

Feature matching

Phase matching

# Constraint condition

uniqueness

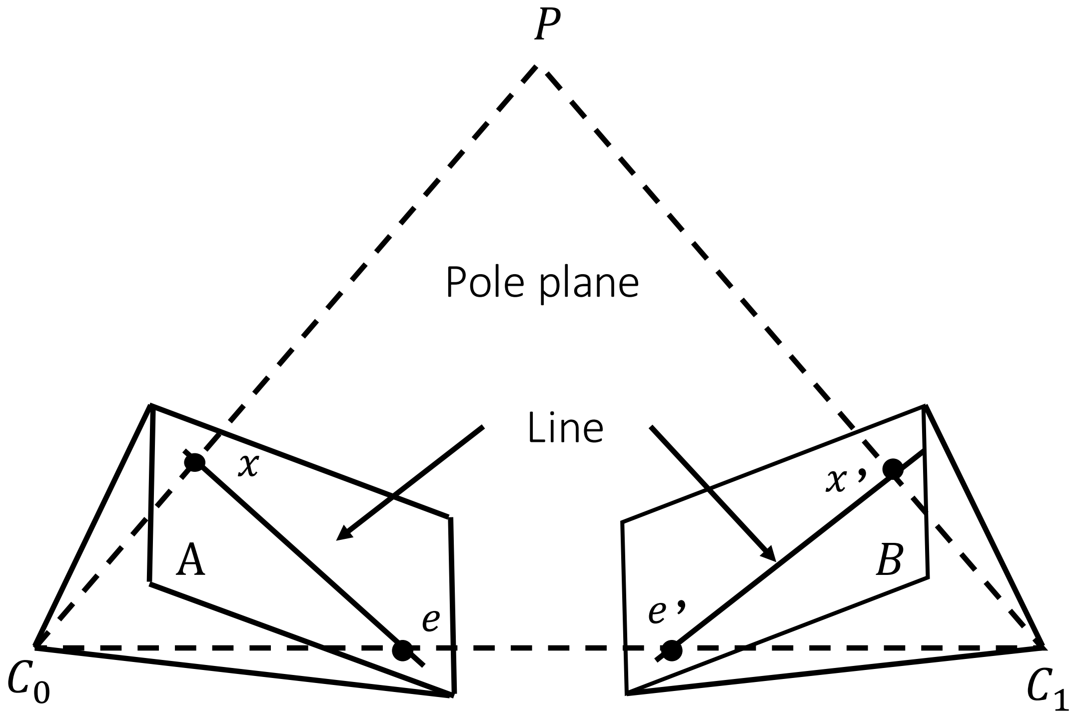


# Constraint condition

## Continuity

# Constraint condition

epipolar constraint




The center of camera

The center of camera

# Constraint condition