COMPUTER VISION AND PHOTOGRAMMETRY

INTRODUCTION

TODAY

- Definitions
    - What this course is about
    - What this course is not about
- Contents
- Objectives
- Some practicalities
- Interesting stuff
- Warm up: Image formation
- Lab 0

DEFINITIONS

COMPUTER VISION AND PHOTOGRAMMETRY

- Computer Vision
  - it is a broad topic
  - we will cover the part oriented to photogrammetry
- Photogrammetry
  - “use photography in surveying and measuring objects”
  - extracting 3D information from 2D images
  - 3D reconstruction

COMPUTER VISION AND PHOTOGRAMMETRY

- Image Formation and Camera Calibration
- Feature detection and matching
- Stereo and Structure from Motion
- Registration, Meshing and Texturing
- Shape from Shading and Photometric Stereo

COMPUTER VISION AND PHOTOGRAMMETRY

- I don’t plan to cover but we might overview:
  - image, video, color, compression/codification
  - tracking
  - motion flow
  - machine learning - deep learning

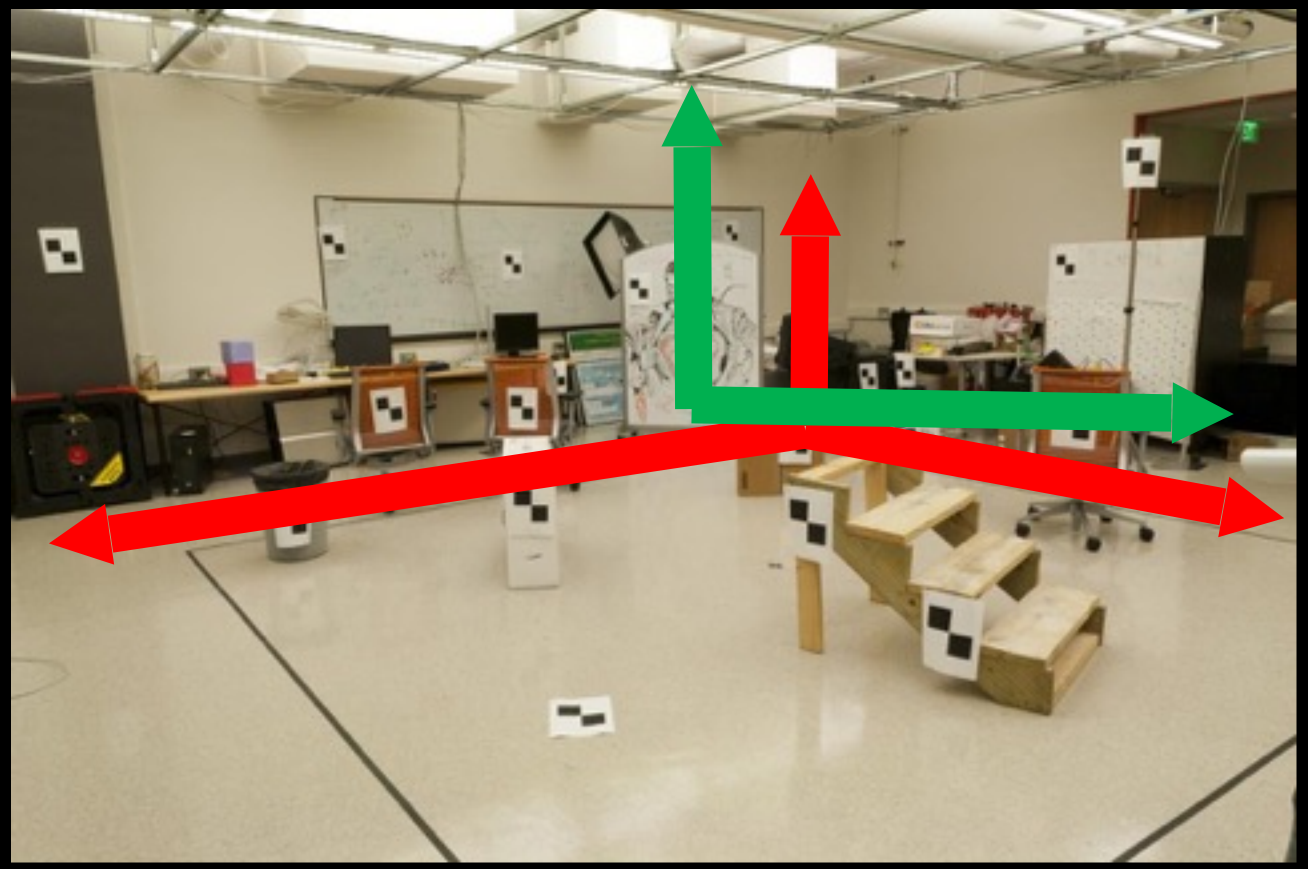
MOTIVATION

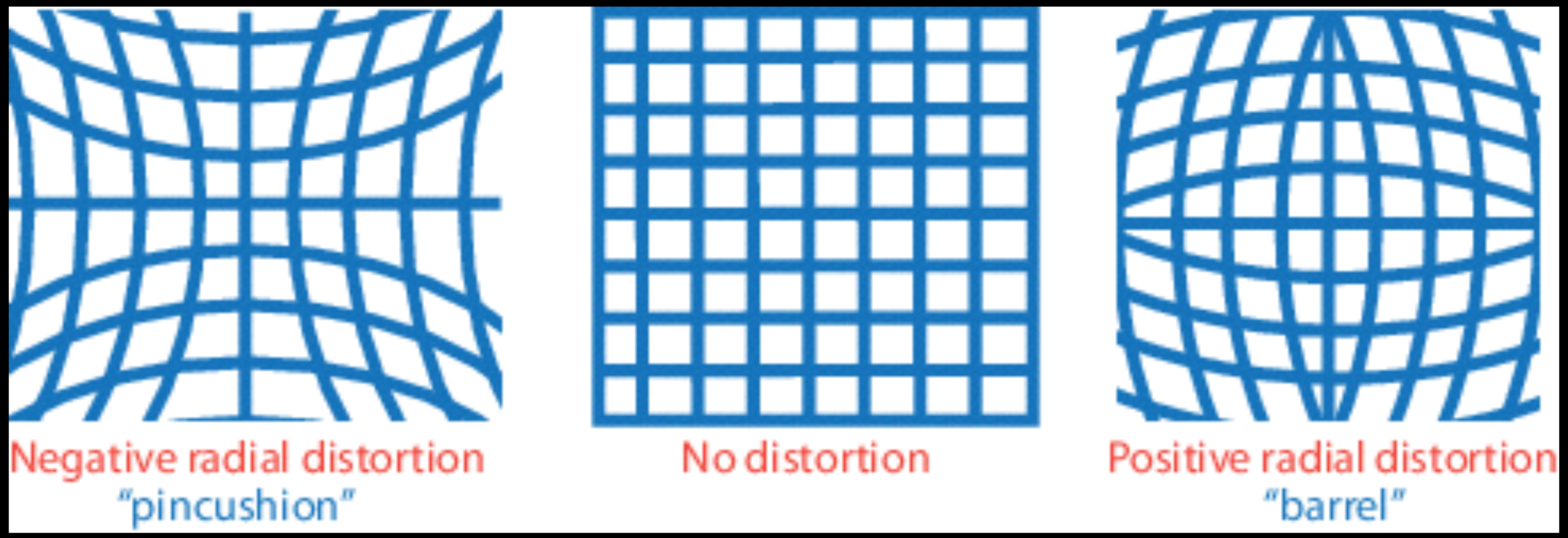
PHOTOGRAMMETRY IS ALL AROUND THE PLACE

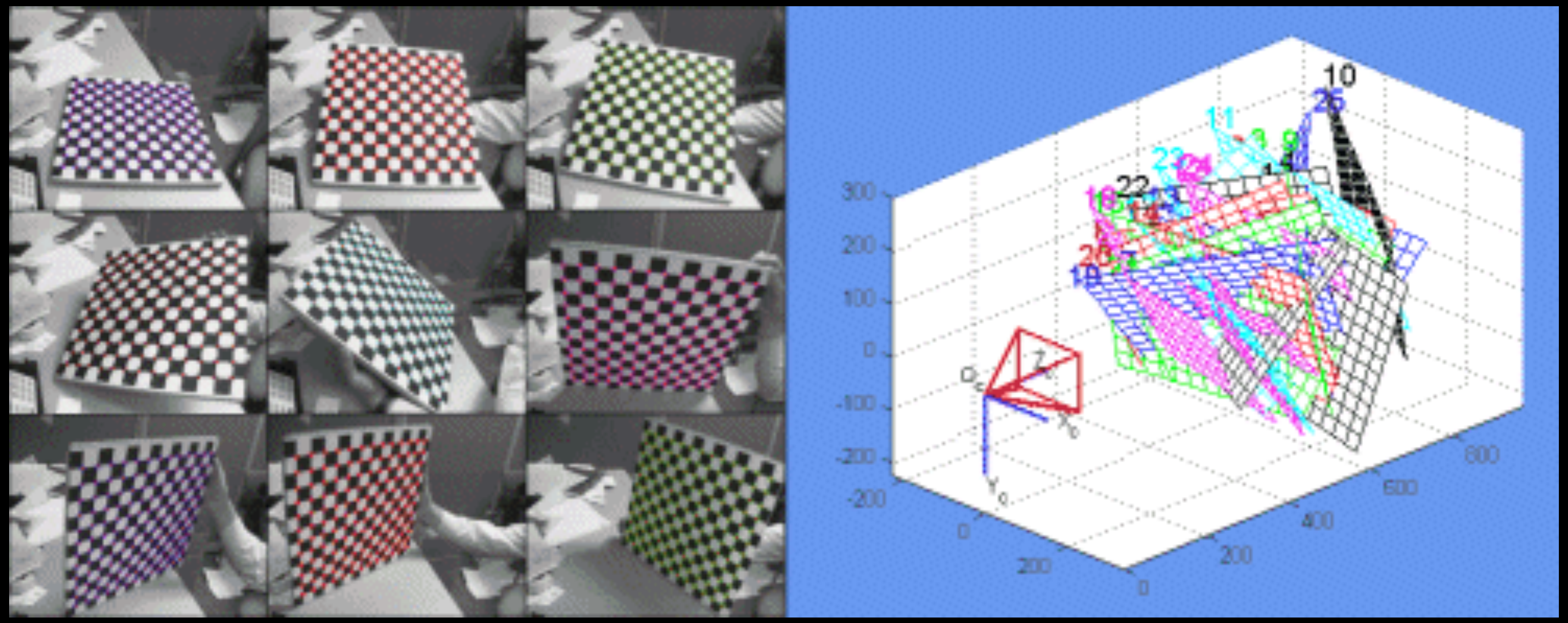
- 3D assets…
  - Video games
  - Film production
  - Cultural heritage
- Architecture and Urban planning
- Arts and multimedia

CONTENTS

IMAGE FORMATION AND CALIBRATION

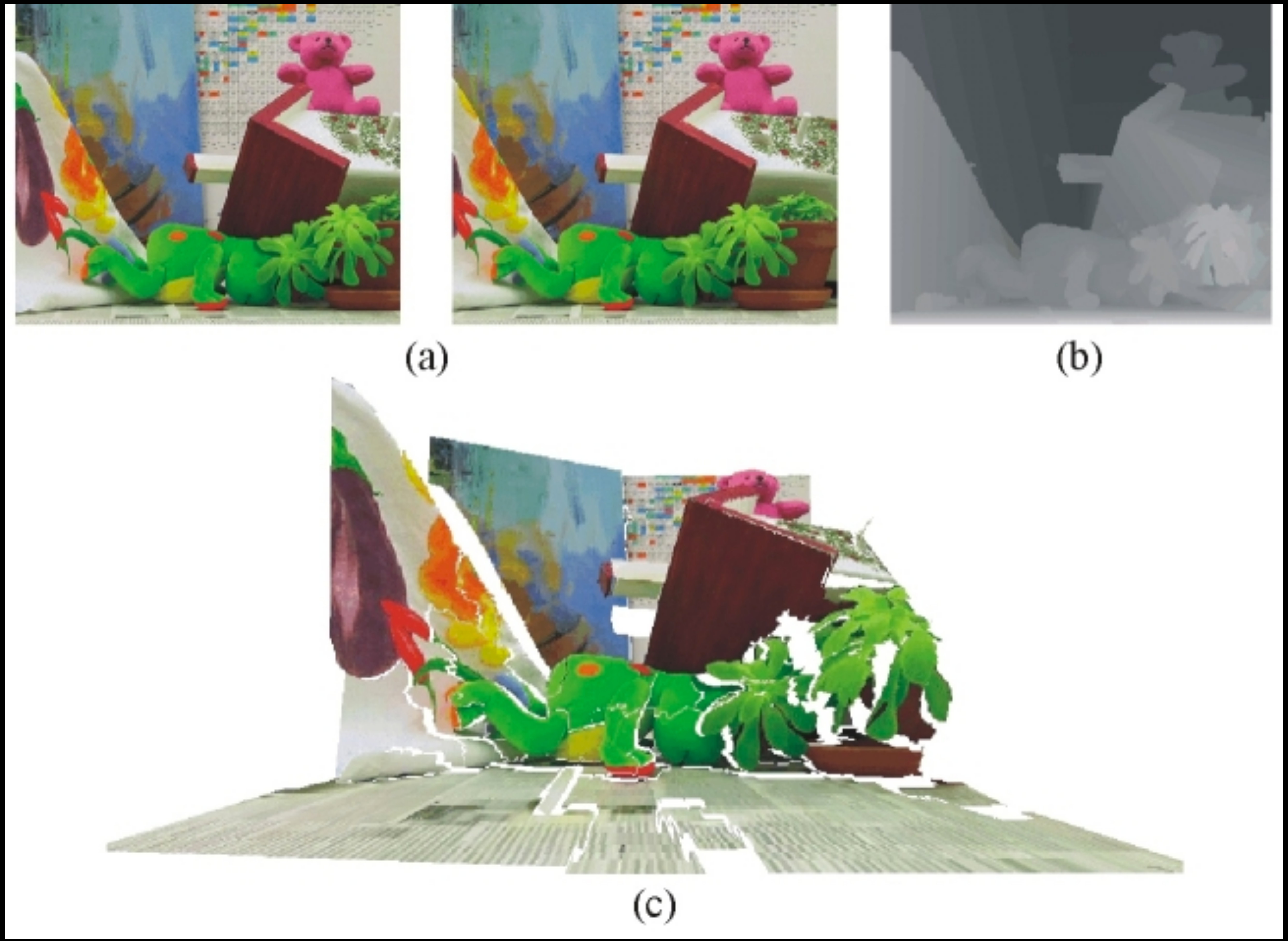






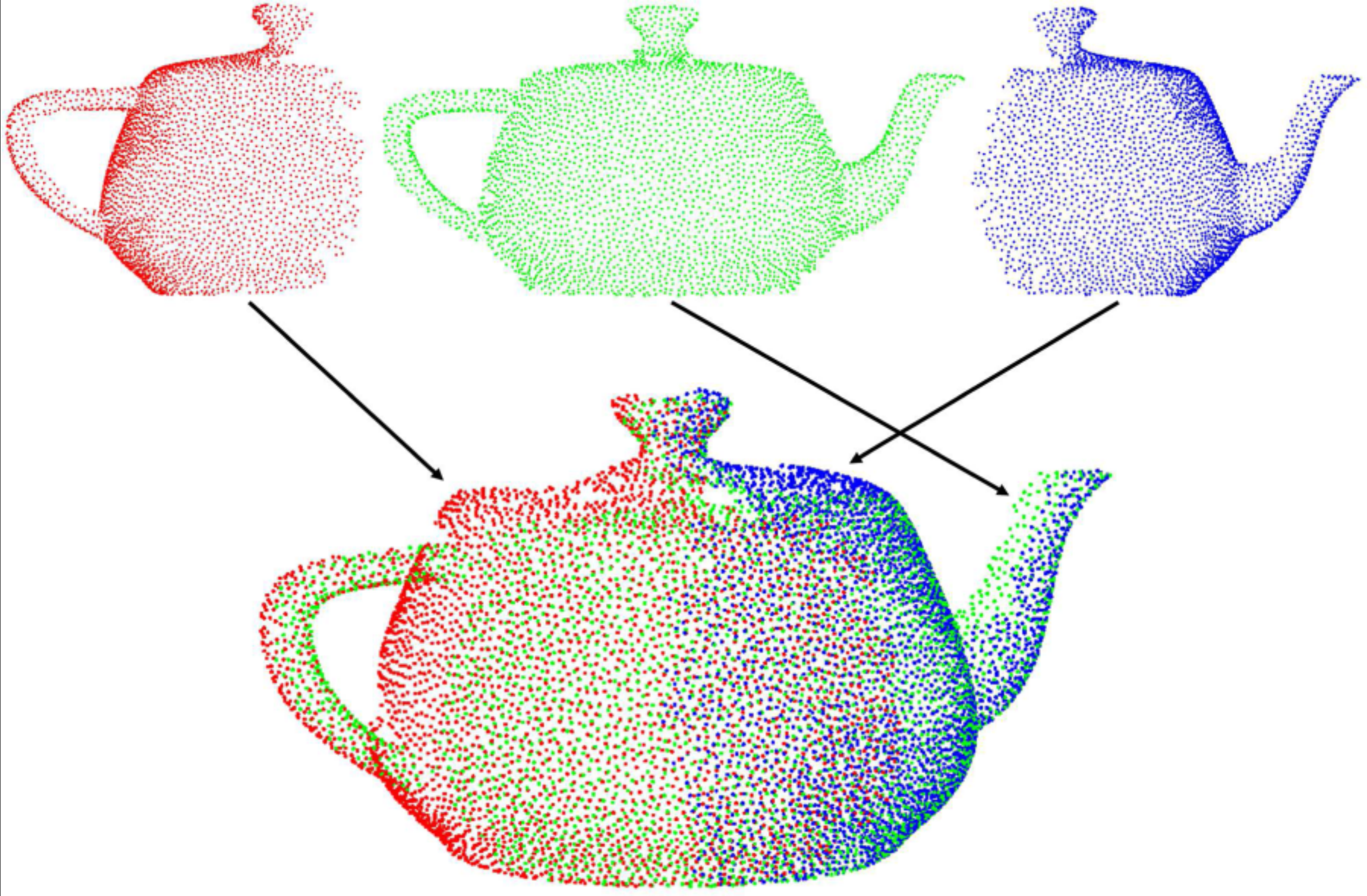
FEATURE DETECTION AND MATCHING

STEREO

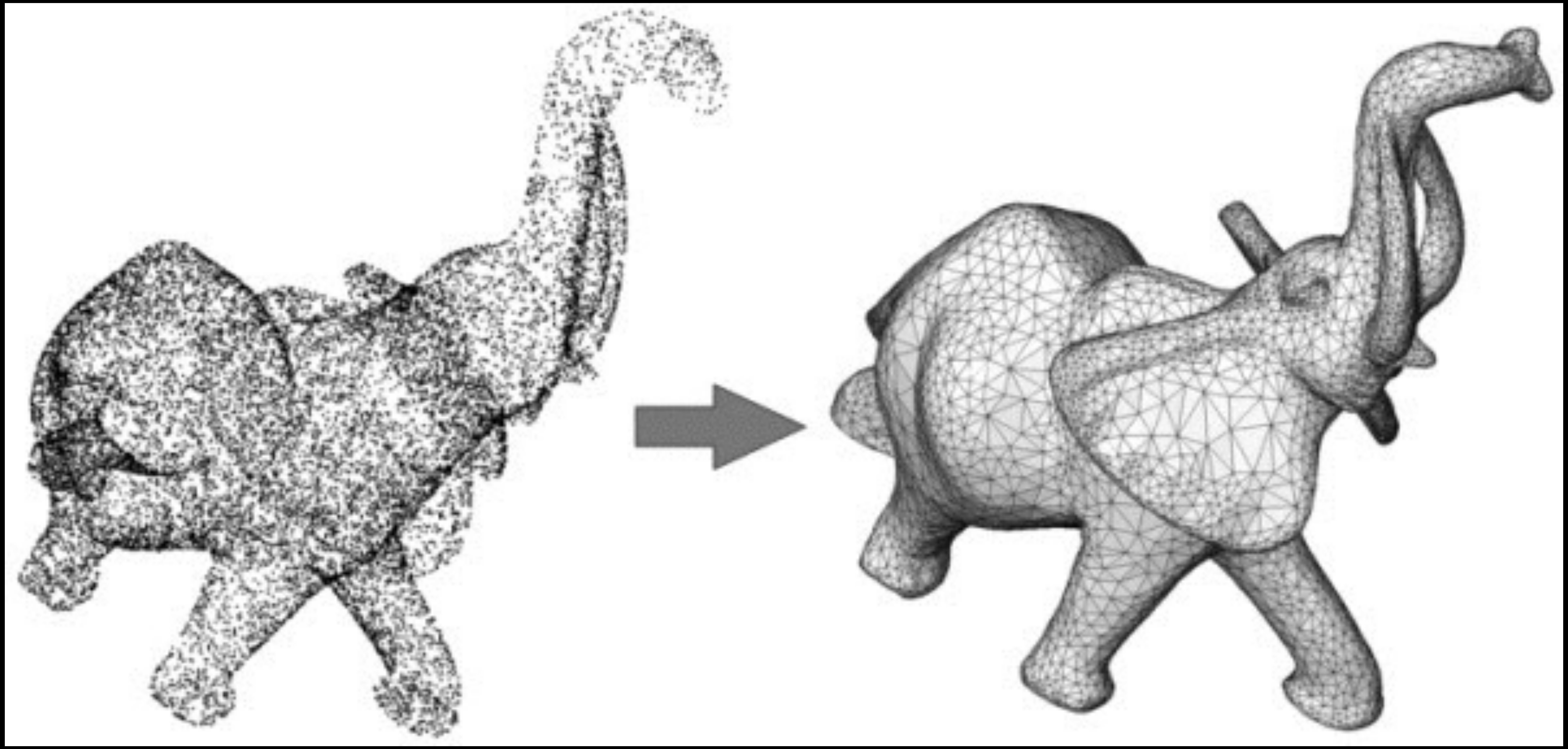


STRUCTURE FROM MOTION (SFM)

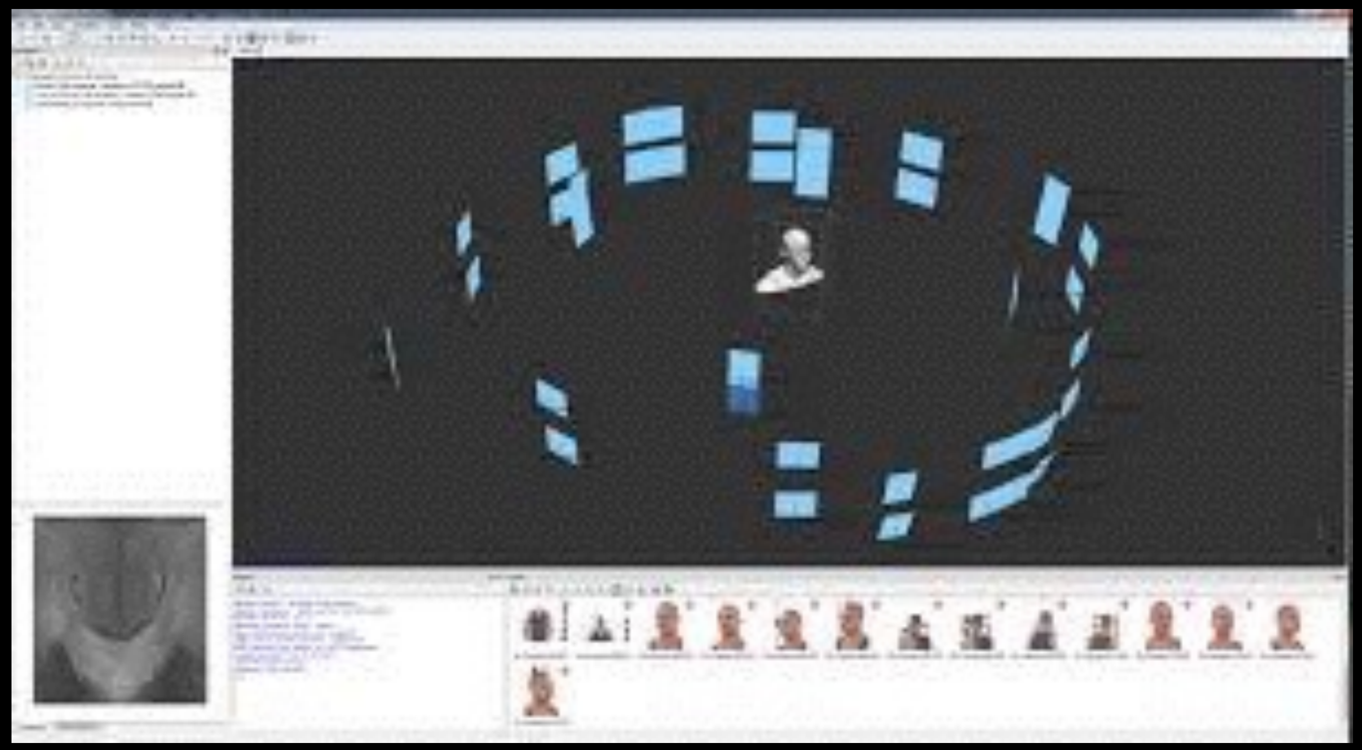
REGISTRATION



MESHING

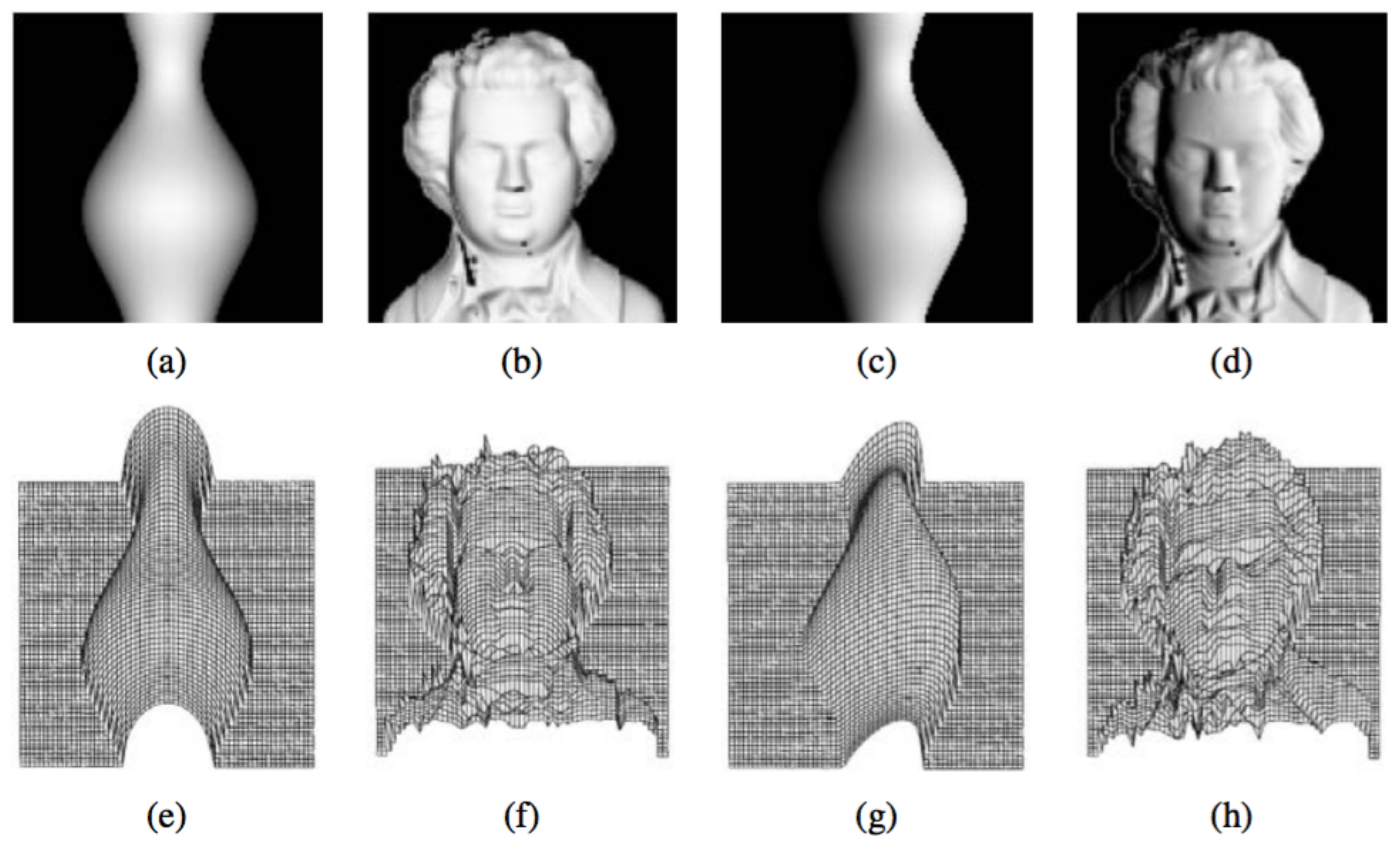


TEXTURING

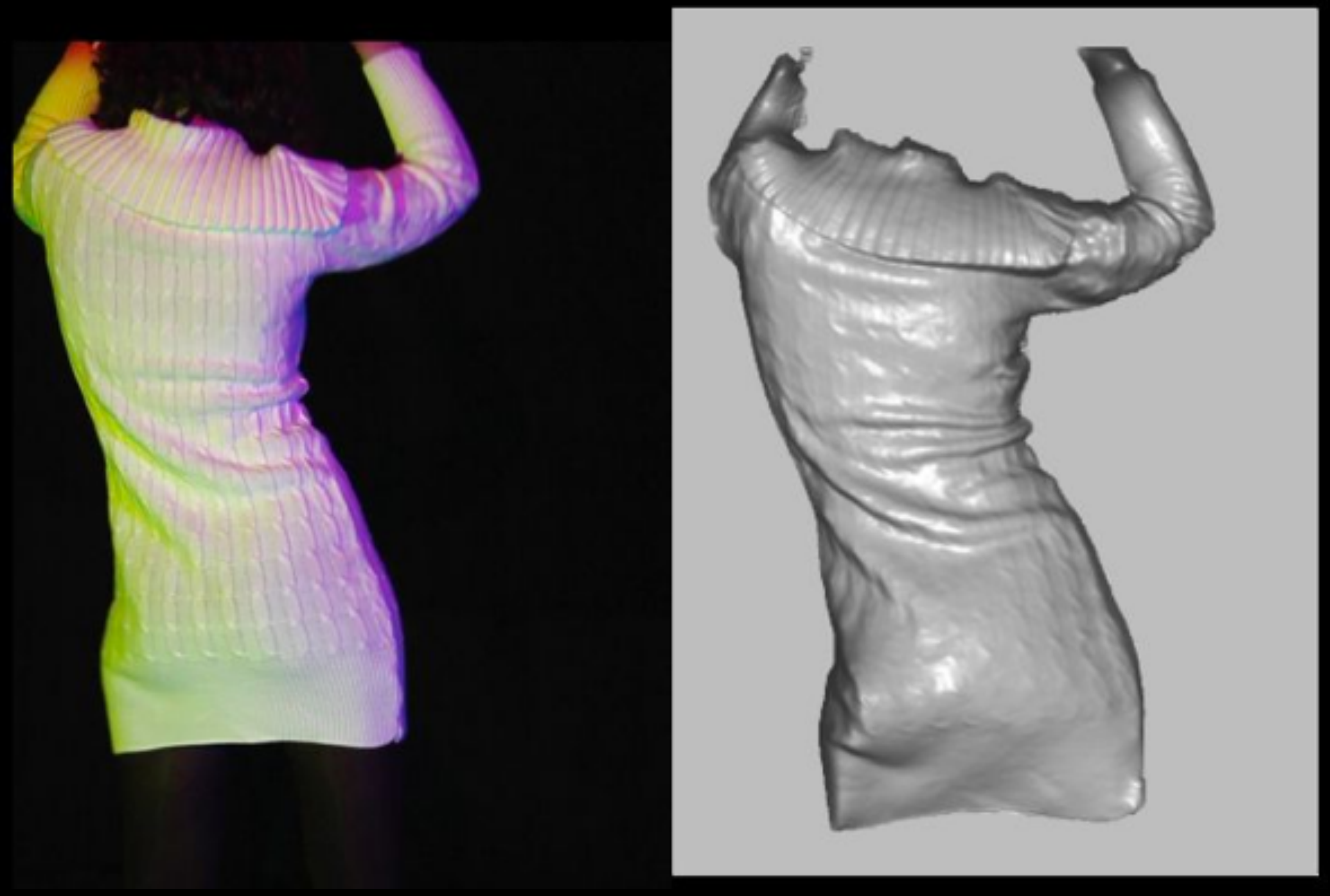


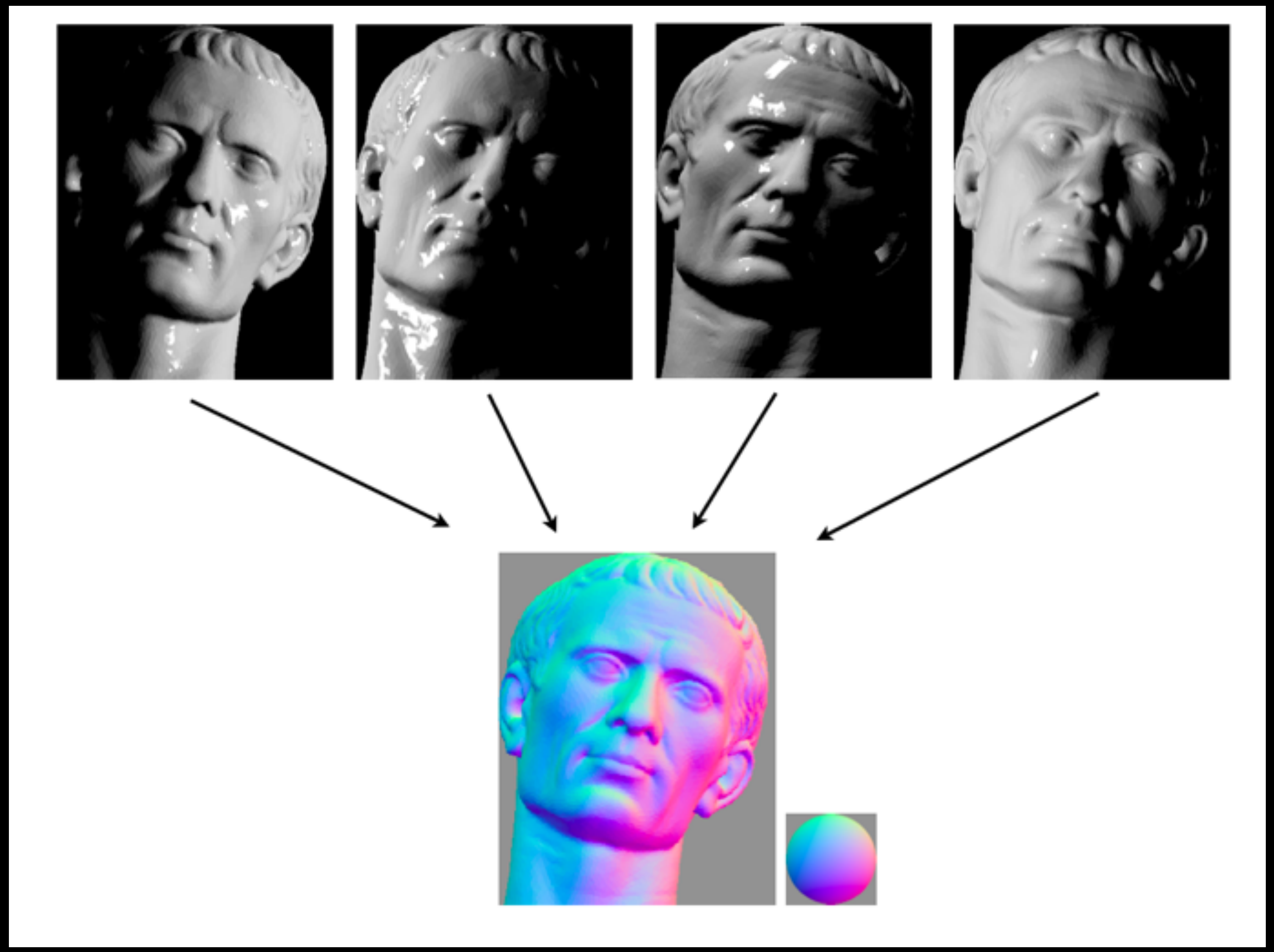
TEXTURING

SHAPE FROM SHADING

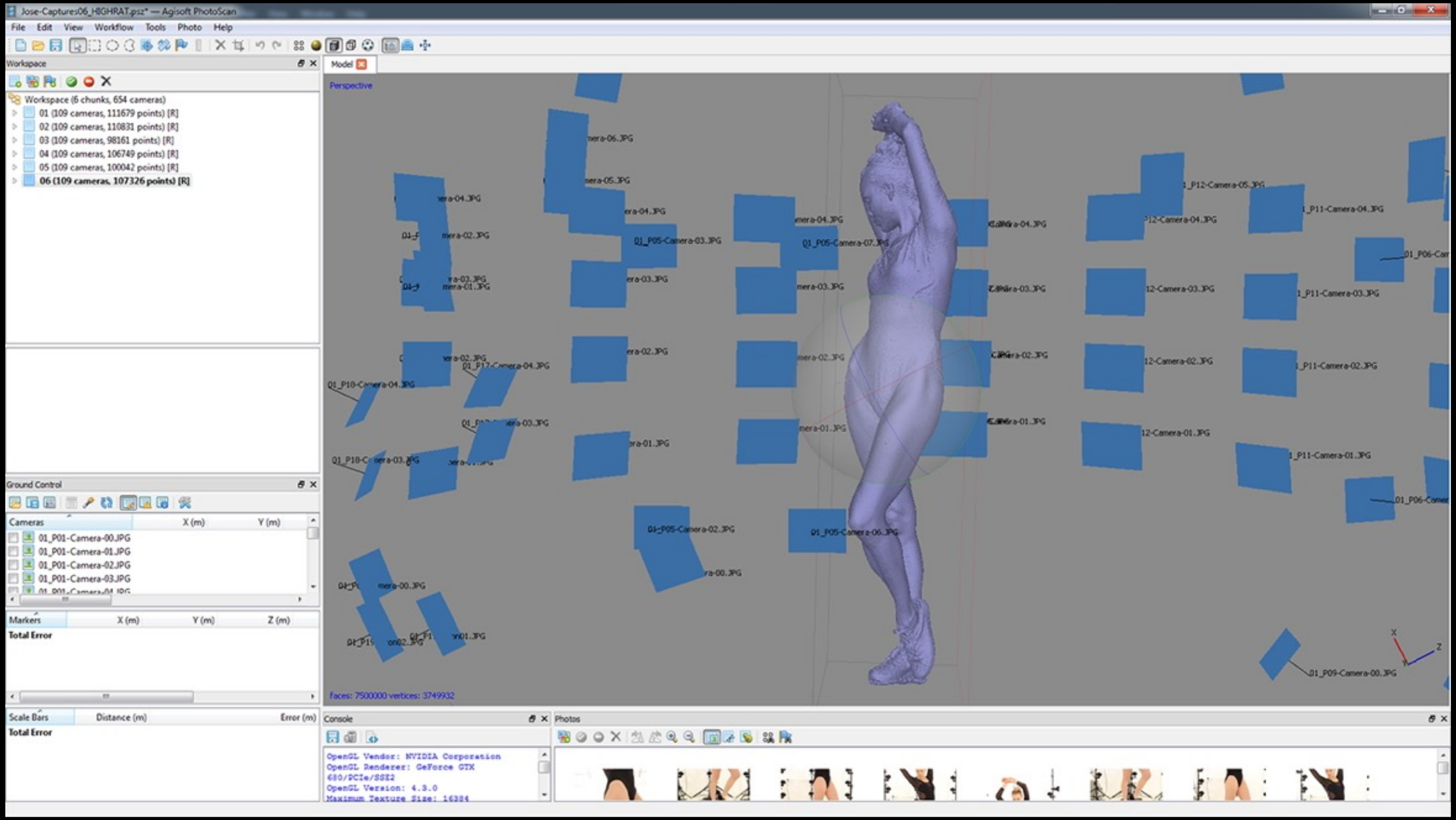


PHOTOMETRIC STEREO

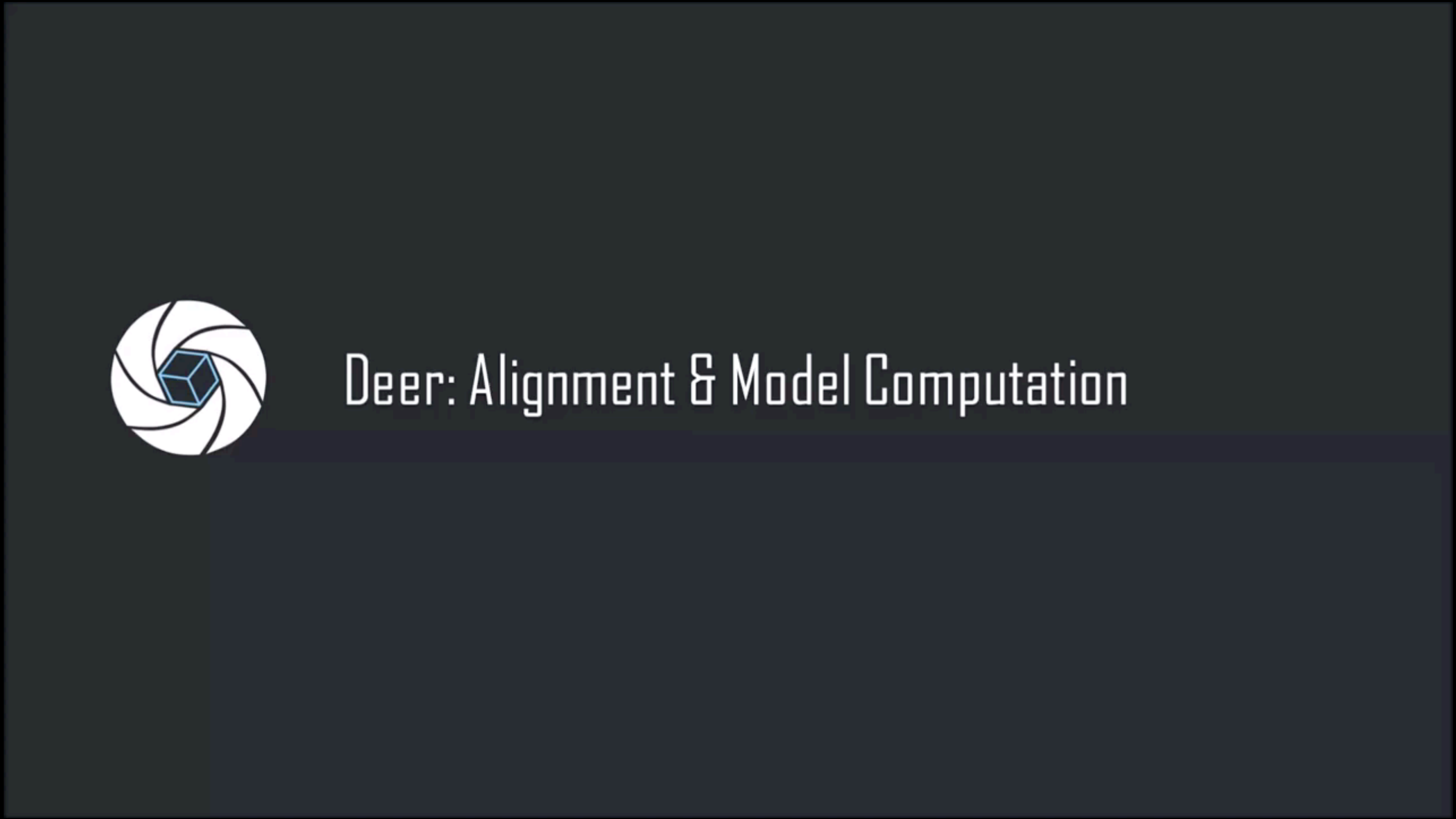




COMERCIAL TOOLS: AGISOFT PHOTOSCAN



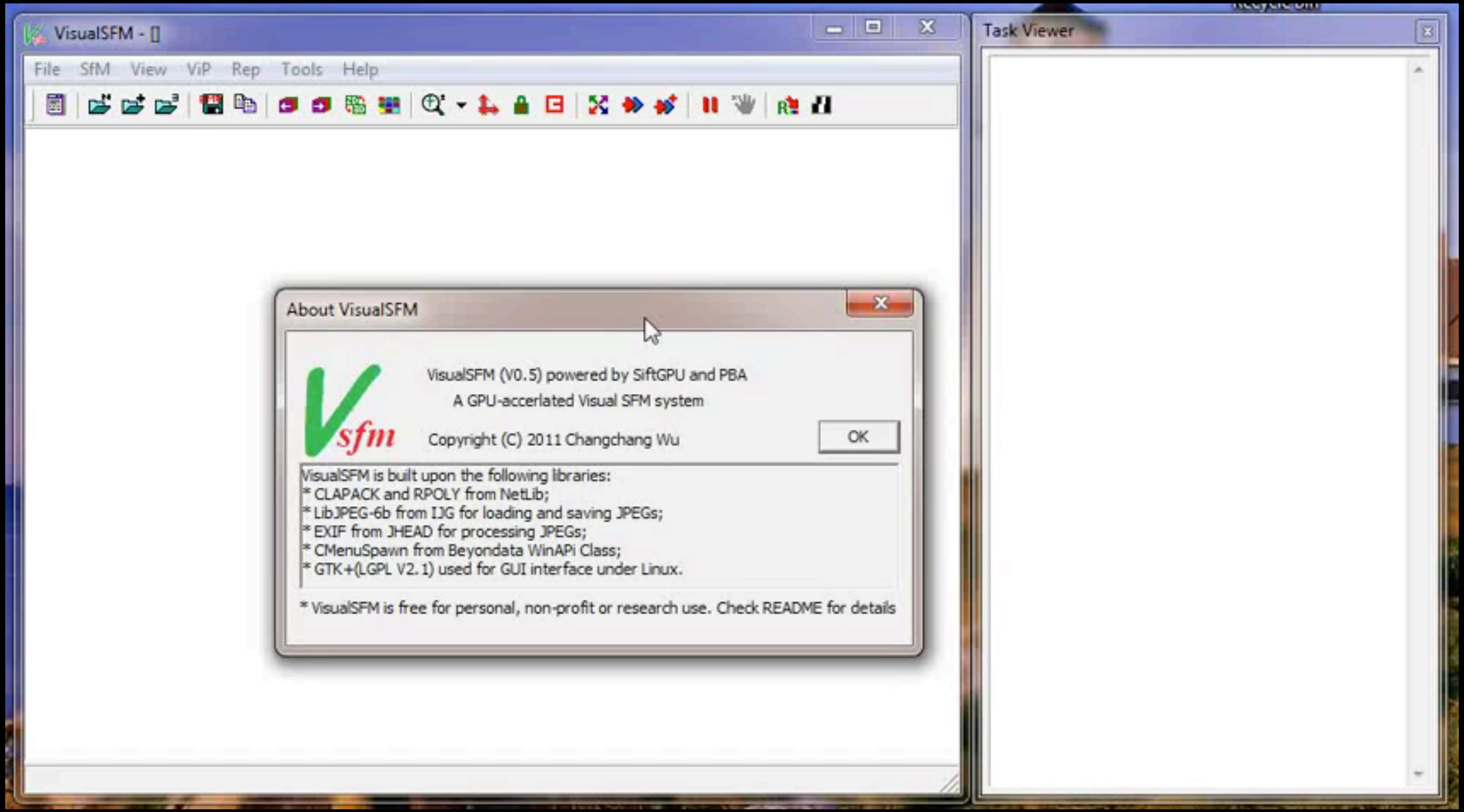
COMERCIAL TOOLS: CAPTURING REALITY



COMERCIAL TOOLS: OTHERS

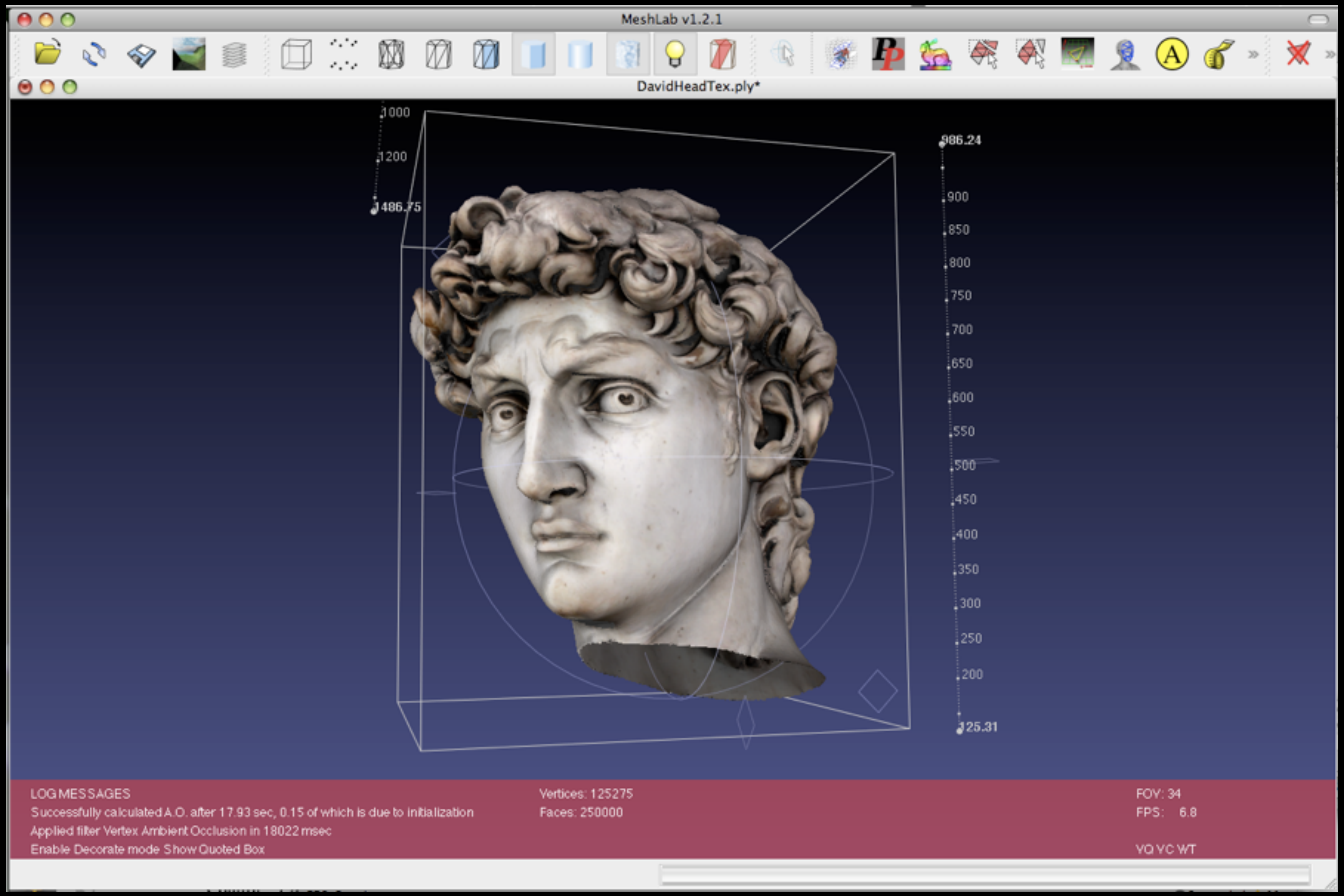
- Autodesk 123D Catch
- Autodesk Recap 360
- Professional Features
  - Post-process and cleaning
  - production-ready models
  - merge with laser scans

RESEARCH TOOLS



RESEARCH TOOLS

- http://meshlab.sourceforge.net/



OBJECTIVES

KNOWLEDGE

- Theoretical background regarding 3d reconstruction
- Hands-on knowledge about 3d reconstruction
- Understanding the problems involved in 3d reconstruction
- and an overview of possible solutions

SKILLS

- Experience implementing image processing algorithms
- Experience with algebra and image processing libraries

PRACTICALITIES

LECTURES

- 15 Lectures: Tuesdays 16:00
- Bring a coffee
- There is no exam so… Lectures are Compulsory
  - If you are not coming - SEND ME AN EMAIL

LABS

- 9-10 Labs
- Every week/two weeks.
- Submitted and marked on lab hours
- The labs are thought for Python.
  - You can use other languages at your own risk

INTERESTING STUFF

MASTERS AND PHDS

- 360 Depth Panoramas
- Other Stuff with Kinects… Performance Capture / dynamic scenes capture
- Stylisation

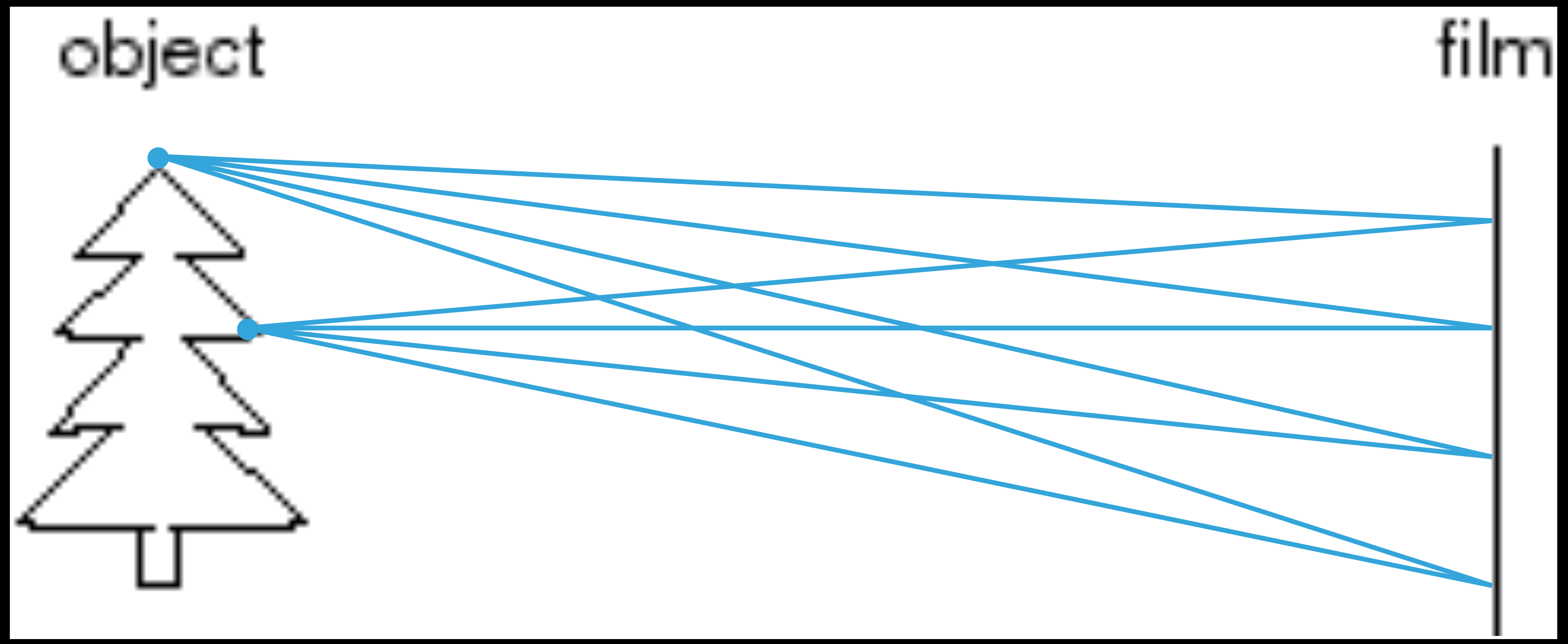
SIGGRAPH CHAPTER

- Come to meet-ups
- Volunteer in SIGGRAPH
- Help organising events

IMAGE FORMATION

IMAGE FORMATION

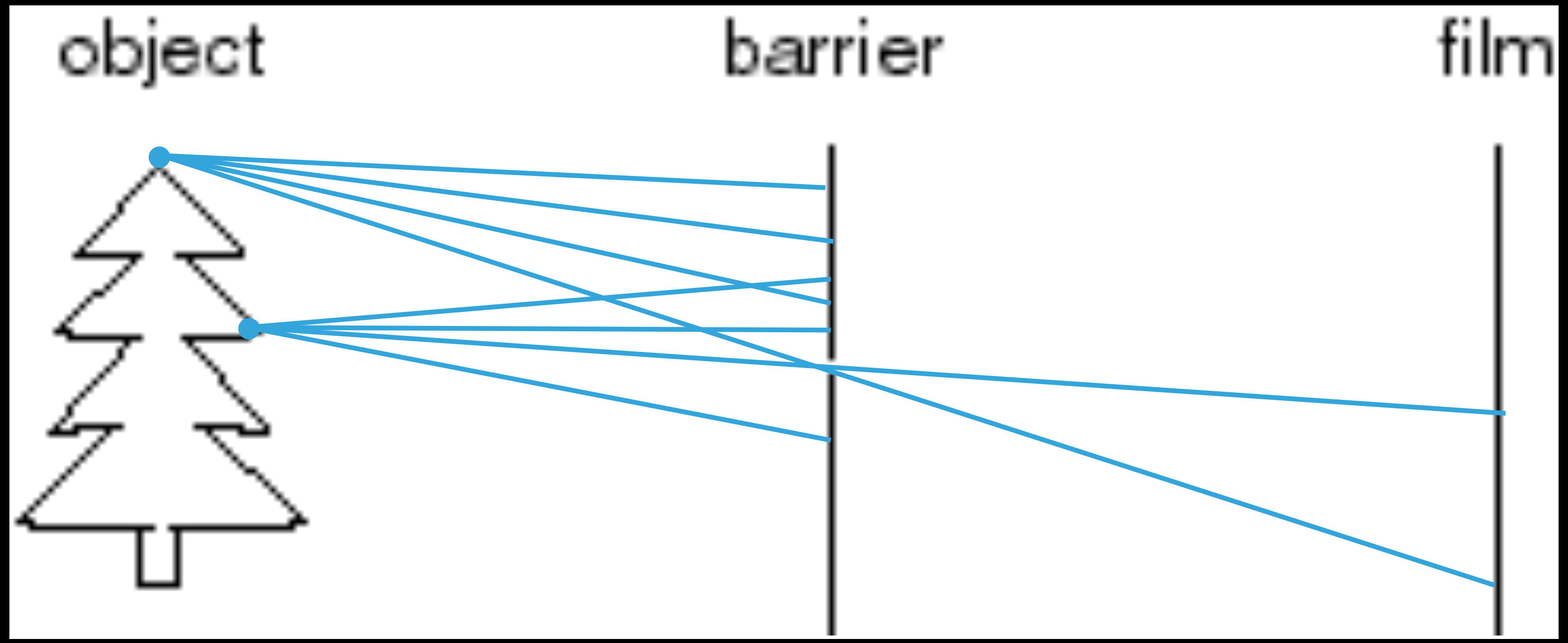
- Let's design a camera
  - Idea 1: put a piece of film in front of an object
  - Do we get a reasonable image?



Slide source: Seitz

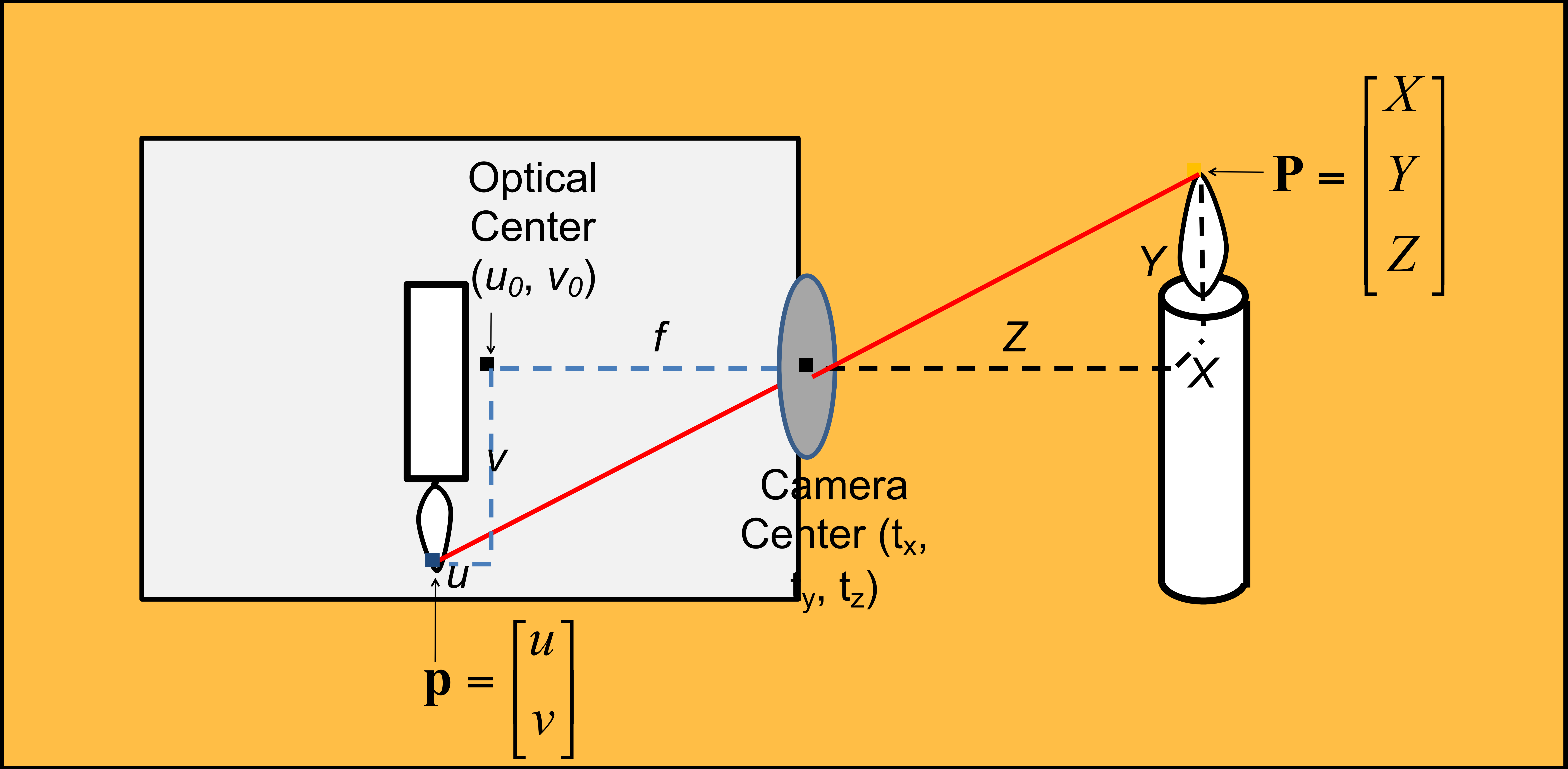
PINHOLE CAMERA

- Idea 2: add a barrier to block off most of the rays
  - This reduces blurring
  - The opening known as the aperture



Slide source: Seitz

PROJECTION: WORLD COORDINATES→IMAGE COORDINATES



HOMOGENEOUS COORDINATES

Converting to homogeneous coordinates

- Conversion

$$(x, y) \Rightarrow \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

homogeneous image coordinates

$$(x, y, z) \Rightarrow \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

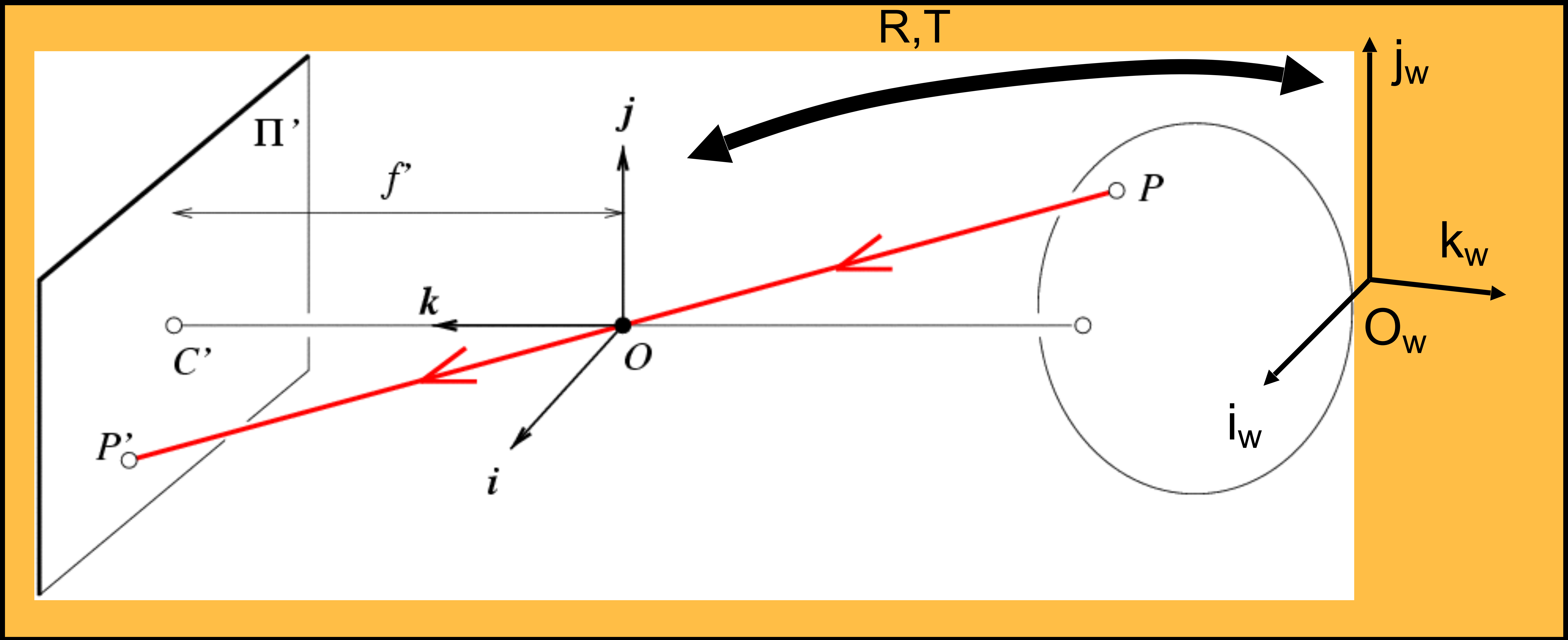
homogeneous scene coordinates

Converting from homogeneous coordinates

$$\begin{bmatrix} x \\ y \\ w \end{bmatrix} \Rightarrow (x/w, y/w)$$

$$\begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix} \Rightarrow (x/w, y/w, z/w)$$

PROJECTION MATRIX



$$\mathbf{x} = \mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix}\mathbf{X}$$

x: Image Coordinates: (u,v,1)

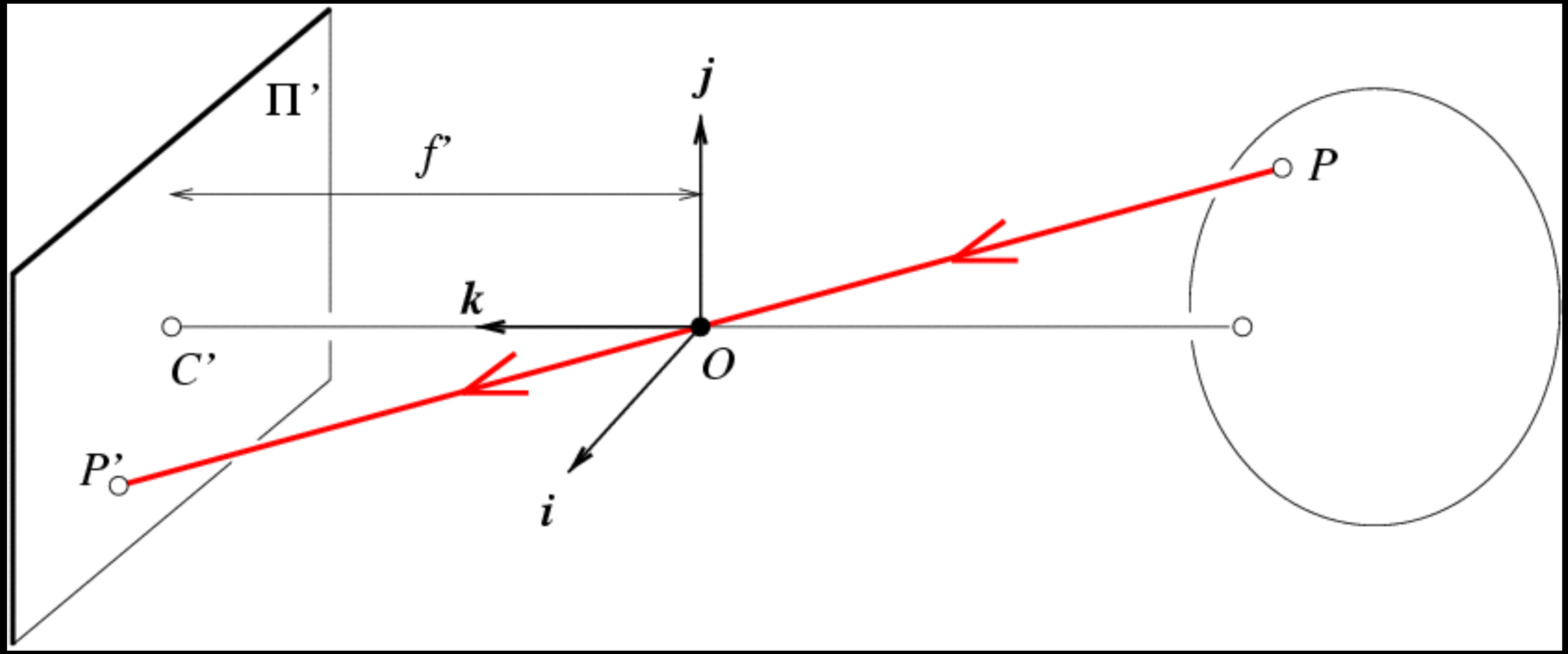
K: Intrinsic Matrix (3x3)

R: Rotation (3x3)

t: Translation (3x1)

X: World Coordinates: (X,Y,Z,1)

PROJECTION MATRIX



Intrinsic Assumptions

Unit aspect ratio

Optical center at (0,0)

No skew

Extrinsic Assumptions

No rotation

Camera at (0,0,0)

$$\mathbf{x} = \mathbf{K}\begin{bmatrix}\mathbf{I} & \mathbf{0}\end{bmatrix}\mathbf{X} \Rightarrow w\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Slide Credit: Saverese

REMOVE ASSUMPTION: KNOWN OPTICAL CENTER

Intrinsic Assumptions

Unit aspect ratio

No skew

Extrinsic Assumptions

No rotation

Camera at (0,0,0)

$$\mathbf{x} = \mathbf{K}\begin{bmatrix}\mathbf{I} & \mathbf{0}\end{bmatrix}\mathbf{X} \Rightarrow w\begin{bmatrix}u\\v\\1\end{bmatrix} = \begin{bmatrix}\boxed{\begin{matrix}f & 0 & u_0\\0 & f & v_0\\0 & 0 & 1\end{matrix}} & \begin{matrix}0\\0\\0\end{matrix}\end{bmatrix}\begin{bmatrix}x\\y\\z\\1\end{bmatrix}$$

REMOVE ASSUMPTION: KNOWN OPTICAL CENTER

Intrinsic Assumptions

No skew

Extrinsic Assumptions

No rotation

Camera at (0,0,0)

$$\mathbf{x} = \mathbf{K}\begin{bmatrix}\mathbf{I} & \mathbf{0}\end{bmatrix}\mathbf{X} \Rightarrow w\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha & 0 & u_0 & 0 \\ 0 & \beta & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

REMOVE ASSUMPTION: KNOWN OPTICAL CENTER

Intrinsic Assumptions

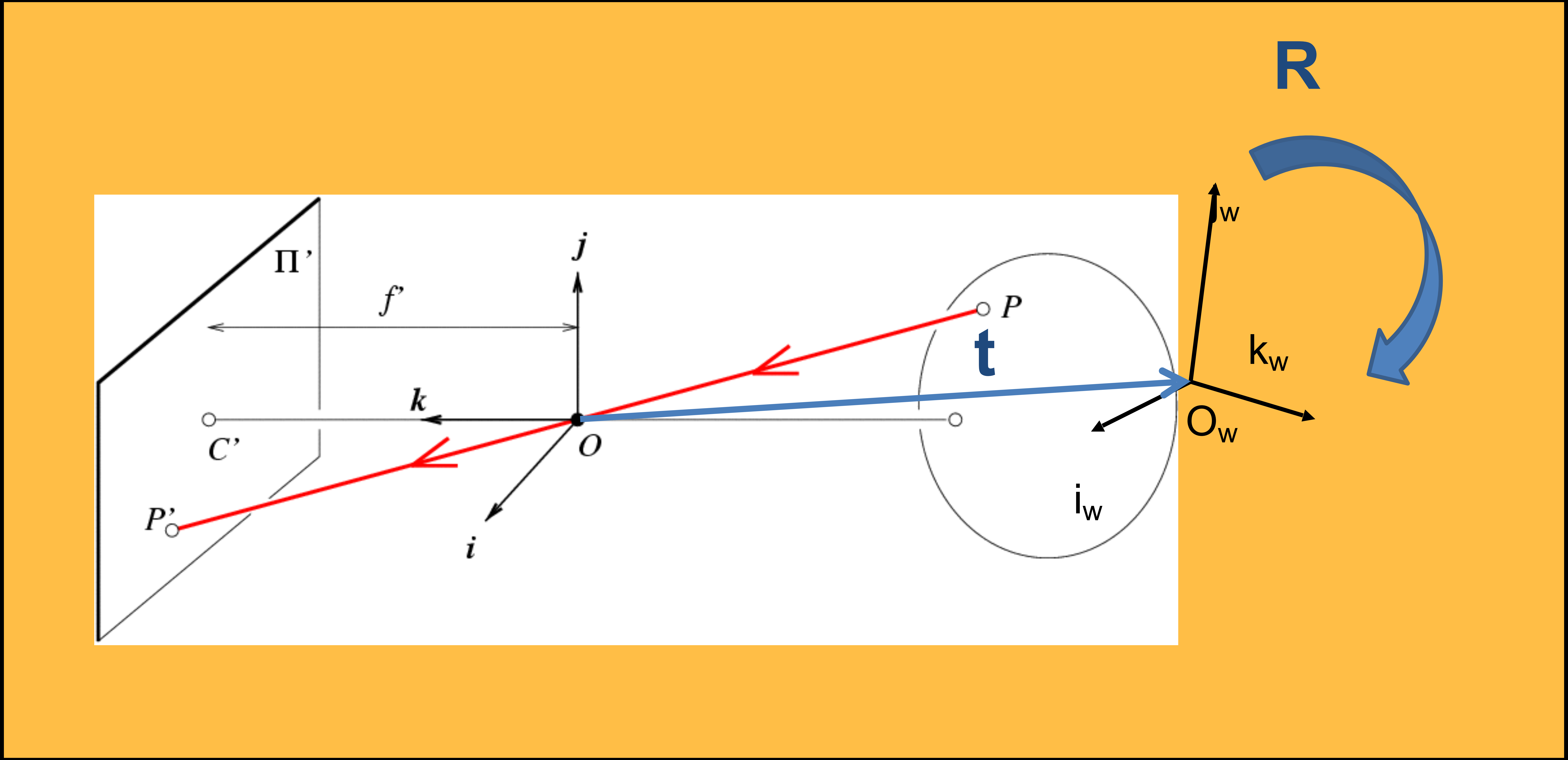
Extrinsic Assumptions

No rotation

Camera at (0,0,0)

$$\mathbf{x} = \mathbf{K}\begin{bmatrix}\mathbf{I} & \mathbf{0}\end{bmatrix}\mathbf{X} \Rightarrow w\begin{bmatrix}u\\v\\1\end{bmatrix} = \begin{bmatrix}\alpha & s & u_0 & 0\\0 & \beta & v_0 & 0\\0 & 0 & 1 & 0\end{bmatrix}\begin{bmatrix}x\\y\\z\\1\end{bmatrix}$$

ORIENTED AND TRANSLATED CAMERA



REMOVE ASSUMPTION: KNOWN OPTICAL CENTER

Intrinsic Assumptions

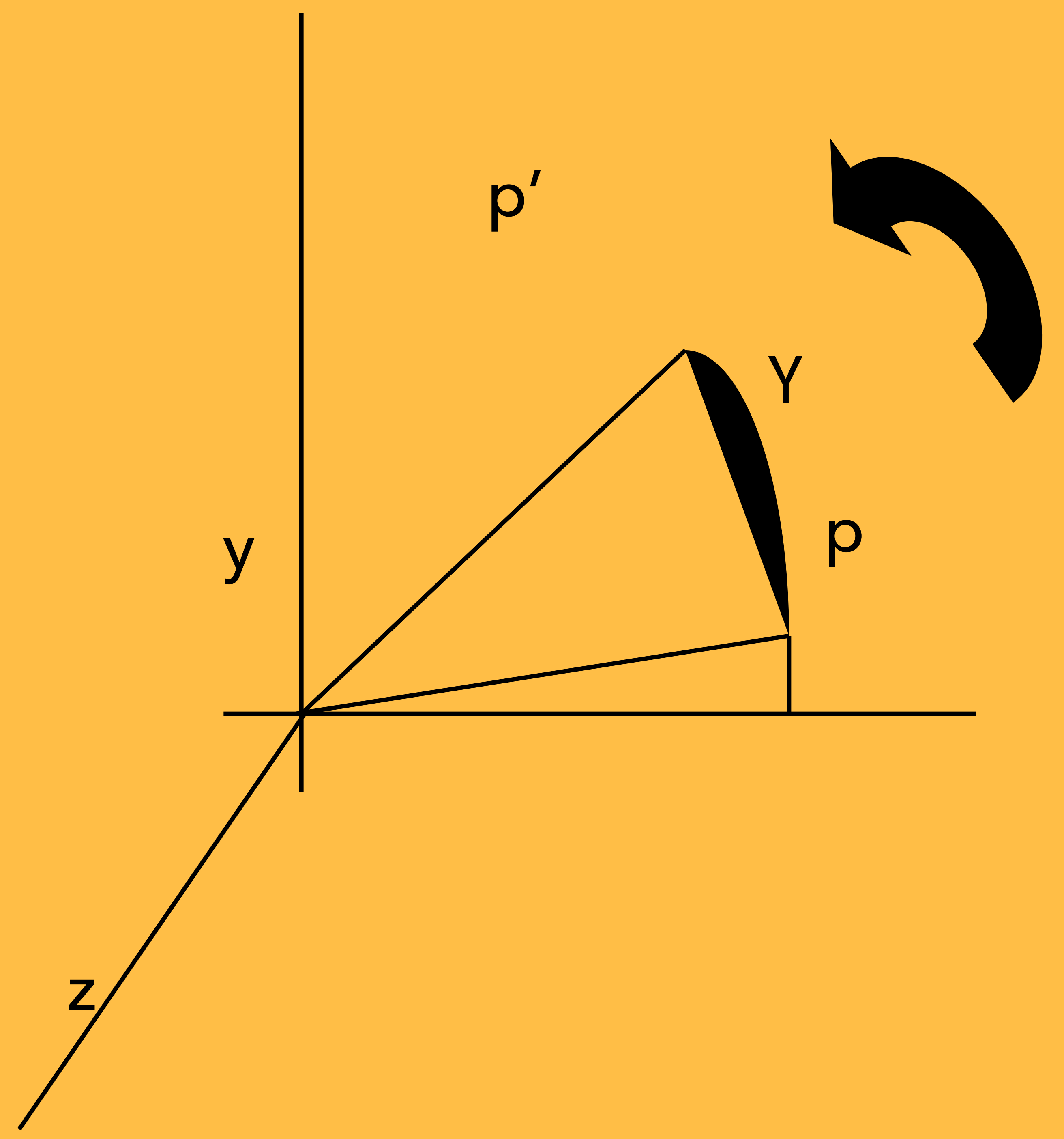
Extrinsic Assumptions

No rotation

$$\mathbf{x} = \mathbf{K}\begin{bmatrix}\mathbf{I} & \mathbf{t}\end{bmatrix}\mathbf{X} \Rightarrow w\begin{bmatrix}u\\v\\1\end{bmatrix} = \begin{bmatrix}\alpha & 0 & u_0\\0 & \beta & v_0\\0 & 0 & 1\end{bmatrix}\begin{bmatrix}1 & 0 & 0 & t_x\\0 & 1 & 0 & t_y\\0 & 0 & 1 & t_z\end{bmatrix}\begin{bmatrix}x\\y\\z\\1\end{bmatrix}$$

3D ROTATION OF POINTS

Rotation around the coordinate axes, counter-clockwise:



$$R_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix}$$

$$R_y(\beta) = \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix}$$

$$R_z(\gamma) = \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

ALLOW CAMERA ROTATION

$$\mathbf{x} = \mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix}\mathbf{X}$$

$$\Downarrow$$

$$w\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha & s & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

HOW TO CALIBRATE THE CAMERA?

$$\mathbf{x} = \mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix}\mathbf{X}$$

$$\begin{bmatrix} su \\ sv \\ s \end{bmatrix} = \begin{bmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & * \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

DEGREES OF FREEDOM

$$\mathbf{x} = \mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix}\mathbf{X}$$

$$w\begin{bmatrix}u\\v\\1\end{bmatrix} = \overset{5}{\begin{bmatrix}\alpha & s & u_0\\0 & \beta & v_0\\0 & 0 & 1\end{bmatrix}}\overset{6}{\begin{bmatrix}r_{11} & r_{12} & r_{13} & t_x\\r_{21} & r_{22} & r_{23} & t_y\\r_{31} & r_{32} & r_{33} & t_z\end{bmatrix}}\begin{bmatrix}x\\y\\z\\1\end{bmatrix}$$

BEYOND PINHOLES: RADIAL DISTORTION

- Common in wide-angle lenses or for special applications (e.g., security)
- Creates non-linear terms in projection
- Usually handled by through solving for non-linear terms and then correcting image

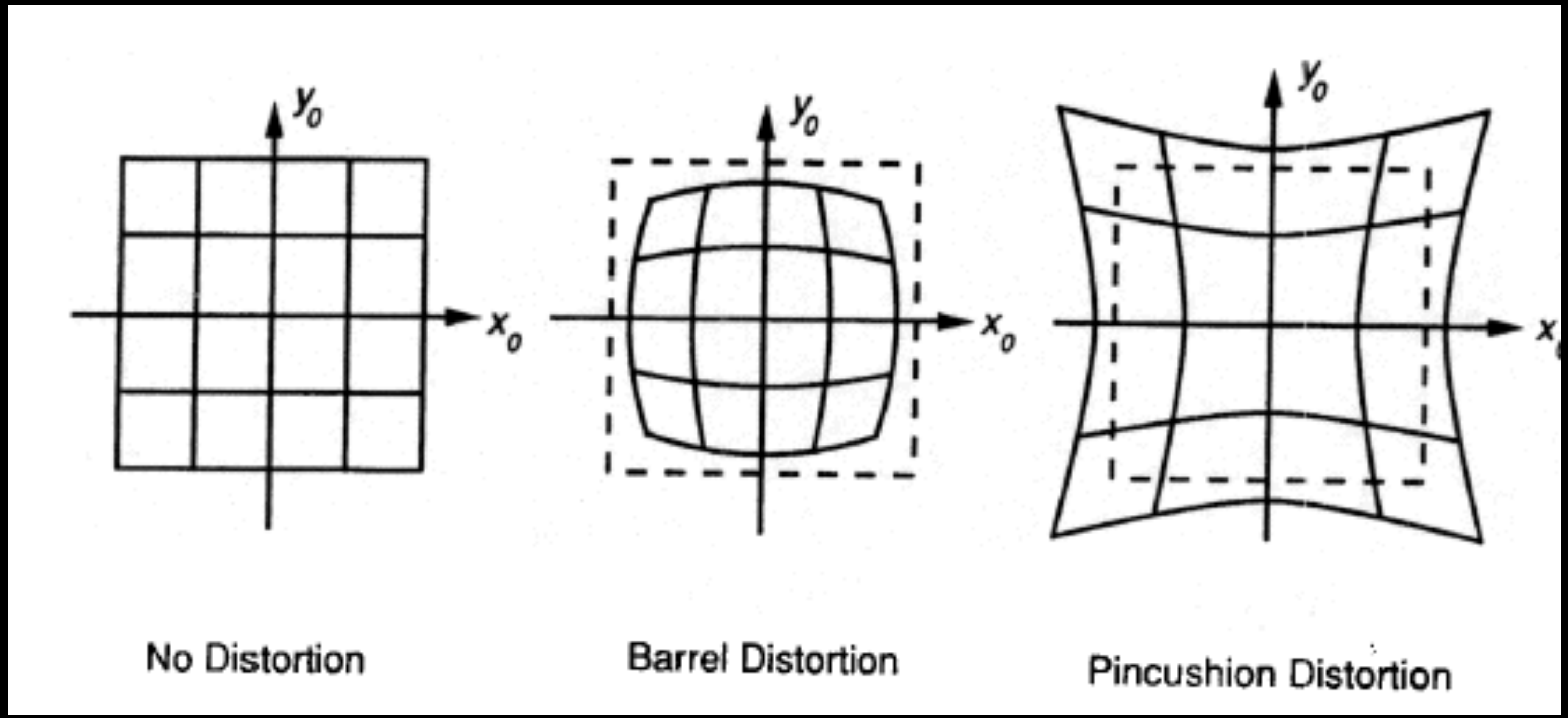


Image from Martin Habbecke

Corrected Barrel Distortion

LAB 0

SOME NICE LINK

- Dissecting the Camera Matrix, Part 3
  - http://ksimek.github.io/2013/08/13/intrinsic/

PYTHON AND IMAGES

- Install python. I recommend Anaconda.
- Try Jupyter! Write your report over your code!
- You most likely will use:
  - numpy
  - imageio, Scikit-image, scipy.ndimage
  - Read an image, save an image, do some tutorials, try some filters, so some manipulation on the data.

REFERENCES

- Some slides from James Hays, Steve Seitz, and Silvio Savarese