# VIDEO SENSORS

Catalin Golban, Anca Foloba

BOSCH

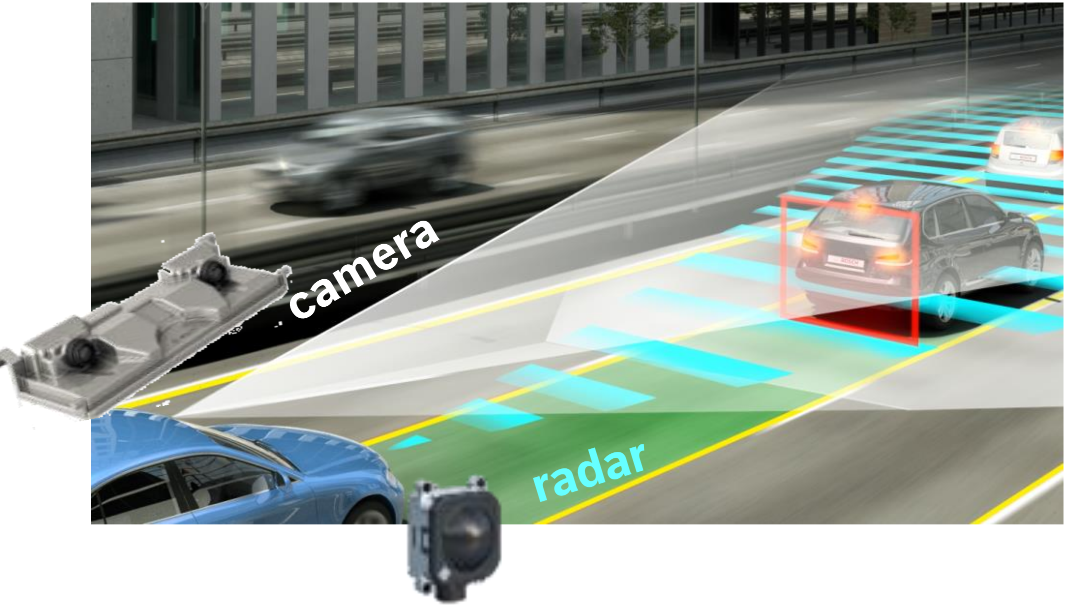
# CONTENTS

# CONTENTS
## Slide structure

# INTRODUCTION TO VIDEO SENSORS

# Bosch Engineering Center Cluj
## Automated driving activities

SOFTWARE ENGINEERING




Radar Systems

Video Systems

Connectivity

Central processing unit

Ultrasonic Systems

Electric Power Steering

# INTRODUCTION TO VIDEO SENSORS
## What are we doing in Cluj?

- main responsibilities on video area
  - Software development and pre-development for mono and stereo video systems
  - Computer vision, image processing and machine learning algorithms development for driver assistance and automated driving


stereo camera

mono camera

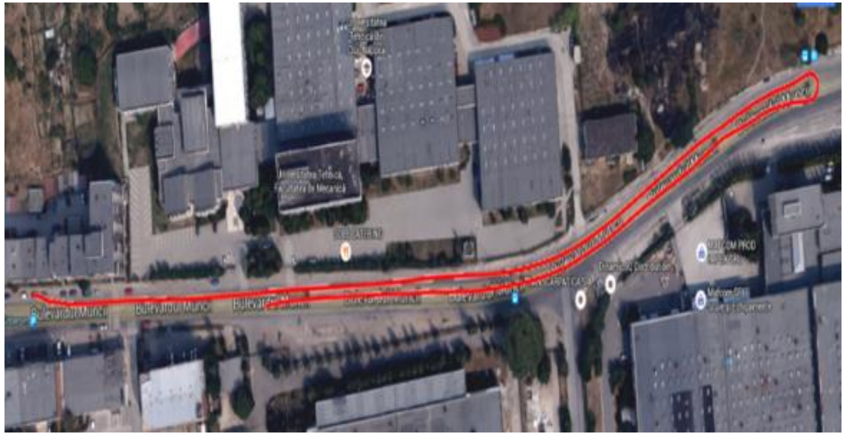

RBRO/ESA1 | 30/08/2017

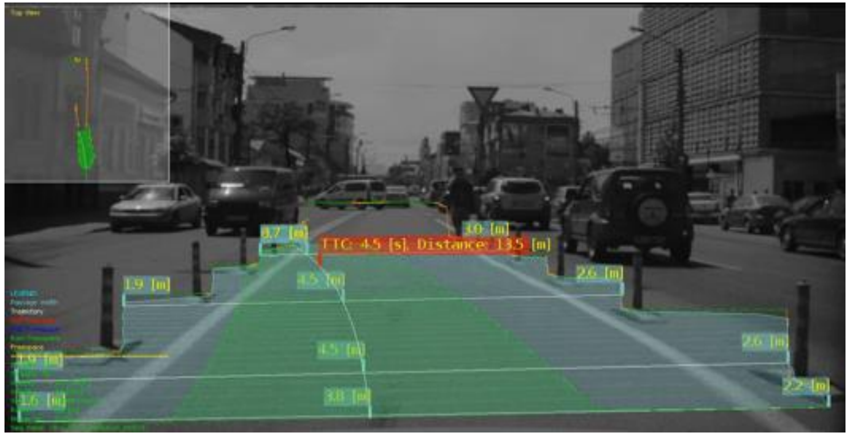
BOSCH

# INTRODUCTION TO VIDEO SENSORS

## What algorithms are we developing in Cluj – some examples

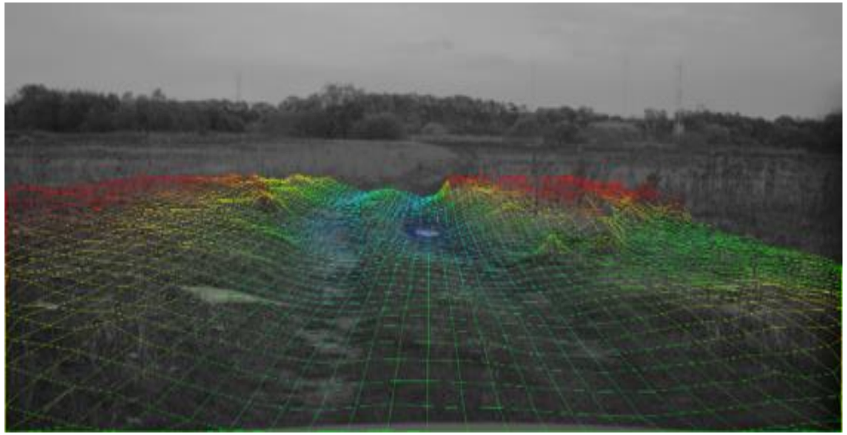
visual odometry





3D free-space

off-road



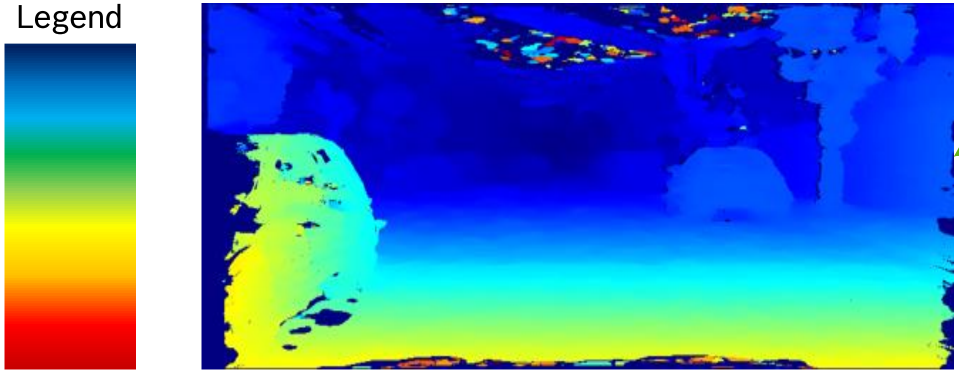
# INTRODUCTION TO VIDEO SENSORS

## Goal for today

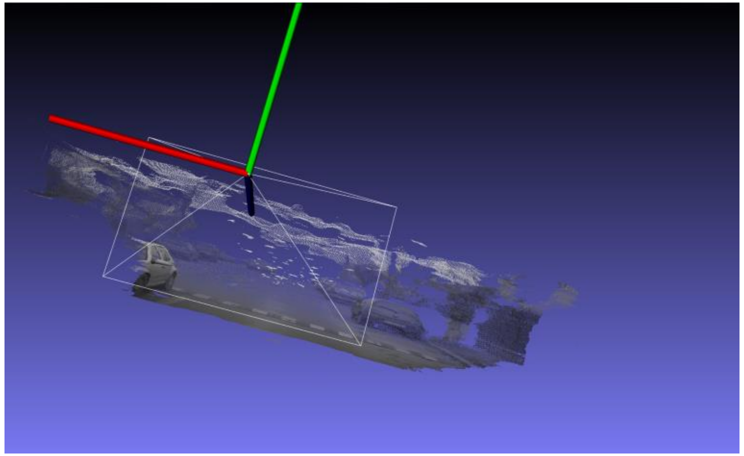
Legend

Far objects

Close objects

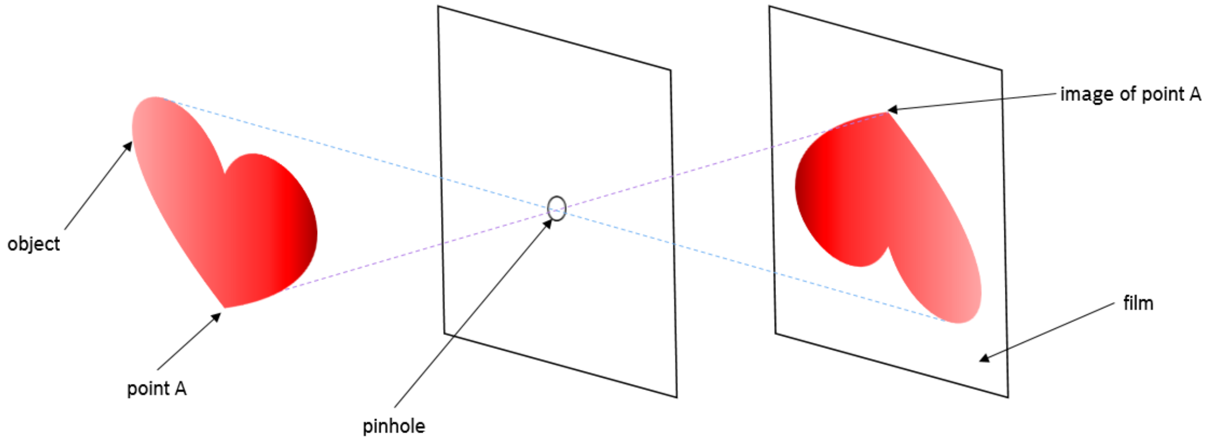


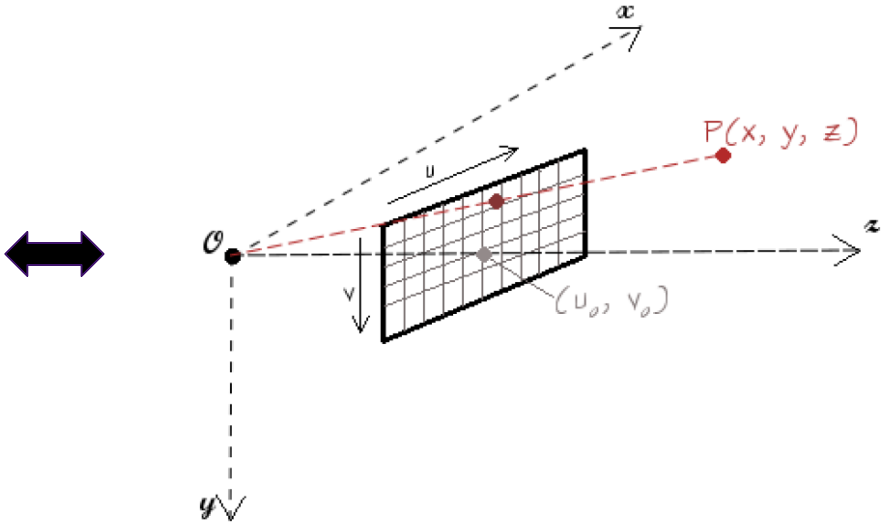
3D reconstruction



# INTRODUCTION TO VIDEO SENSORS
## Pinhole camera & geometric model

- pinhole camera model – the light passes through the pinhole and projects an inverted image → it has small amount of light




our model, no inversion needed




BOSCH

# WHAT ARE IMAGES AND HOW ARE THEY REPRESENTED?

# WHAT ARE IMAGES AND HOW ARE THEY REPRESENTED?

## What is an image?

- image – depicts the visual perception which has the appearance of some object, person, landscape, etc.
- an image can be two-dimensional such as a photograph, or three-dimensional such as a hologram or statue
- moving image – a video

Hologram – Source – https://www.technobuffalo.com/wp-content/uploads/2015/12/Hologram-Pyramid-1.jpeg


RBRO/ESA1 | 30/08/2017

# How is an image represented?

## Photograph

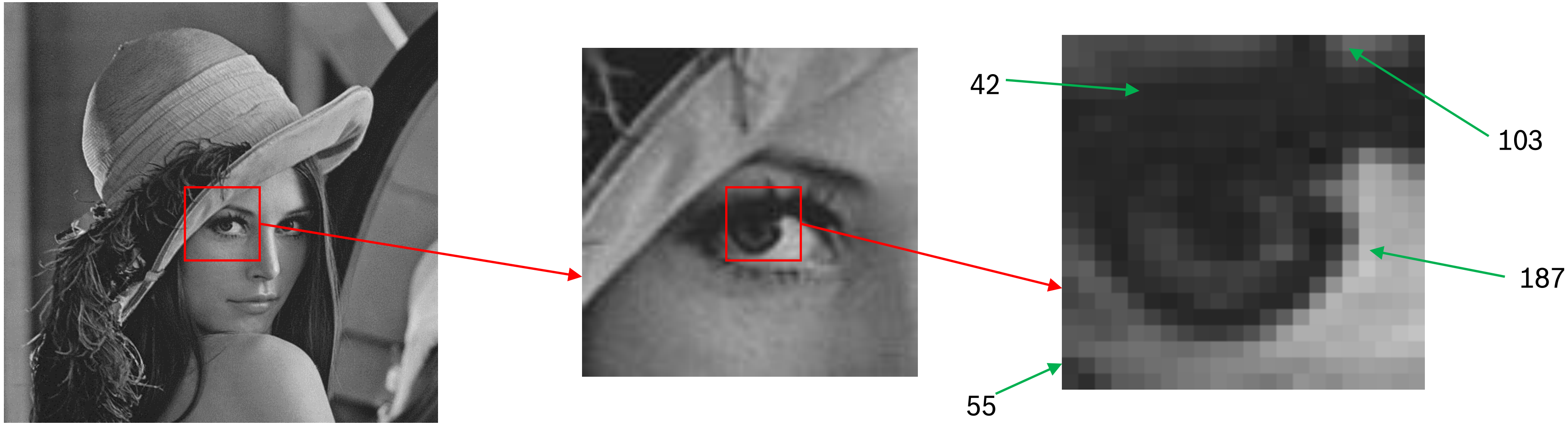
- photograph – image created using a camera
- the light is captured by the camera – it falls on a light-sensitive surface and then it is encoded in a digital format

Photograph

# How is an image represented?

## Digital image

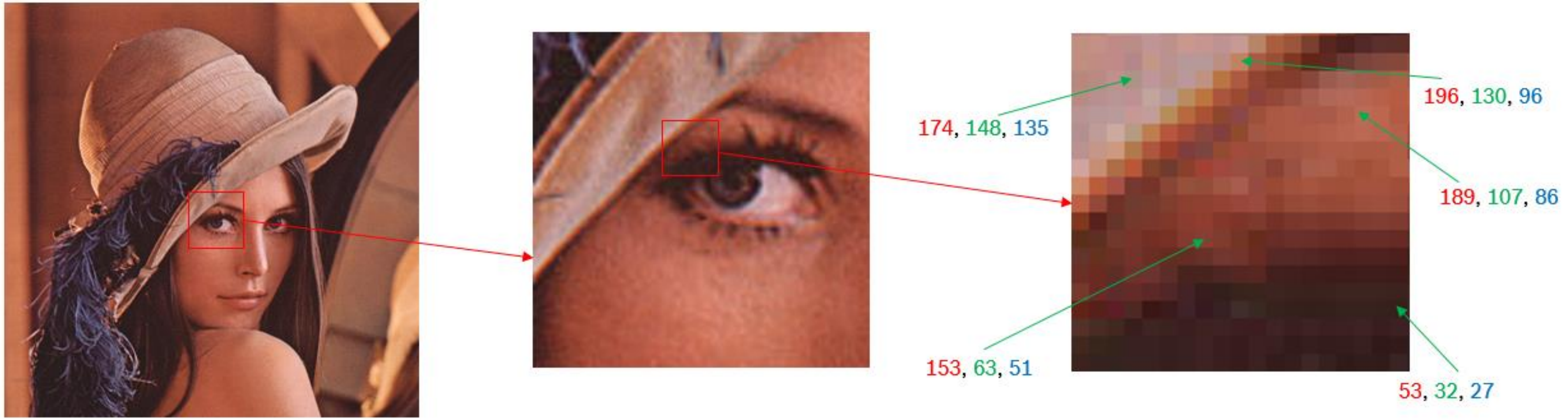
- numeric representation of an image
- represented as a vector or as a matrix
- each numerical value corresponds to a single pixel from the image



# How is an image represented?

## Digital image

- most common model to represent a color image is **RGB** (**R**ed, **G**reen, **B**lue) → each pixel is represented by three values – amount of red, green and blue
- this will use more amount of memory than gray-scale images



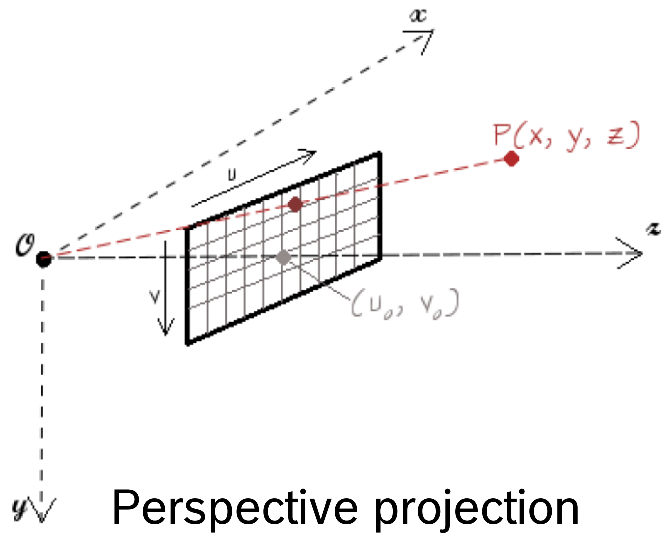
# WHAT ARE IMAGES AND HOW ARE THEY REPRESENTED?

## How is an image represented?

# CALIBRATION

# CALIBRATION
## General concepts

- **Computer vision** – getting 3D information from 2D image data → **the parameters of the model** used for transformation of the 3D points to 2D pixels must be known
- Calibration needed in order to obtain a good projection of the world in the image plane
  - **Intrinsic** parameters
  - **Extrinsic** parameters
- Calibration
  - **Static calibration** (happens before the actual usage of the video sensor)
  - Online (while driving the de-calibration is detected)
    - geometric model is not fixed, it changes in time => **online calibration**




Perspective projection

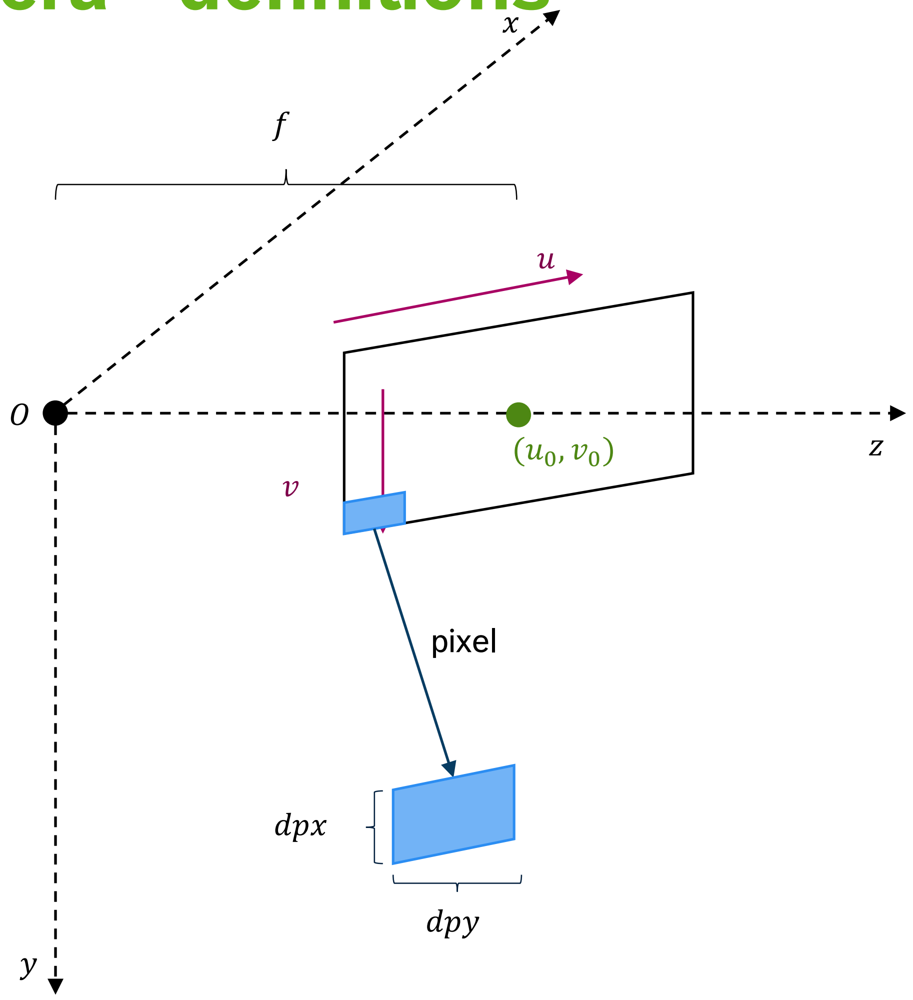

RBRO/ESA1 | 30/08/2017

BOSCH

# CALIBRATION

## Intrinsic parameters of the camera - definitions

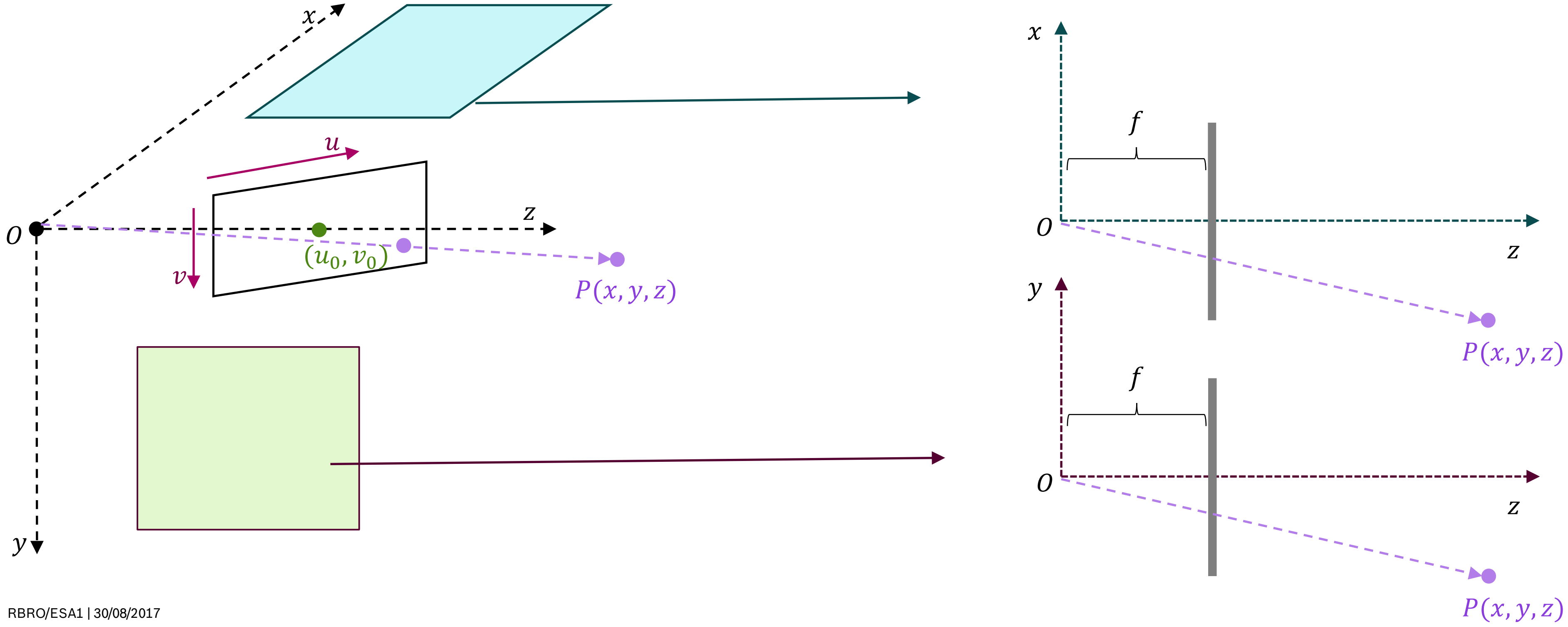
- $O$ – camera center
- $u_0, v_0$ - principal point
- $u, v$ – image coordinates
- $f$ – focal length – distance from $O$ to principal point
- $dpx, dpy$ – size of pixels
- $f_x, f_y$ - focal length in pixels units



# 3D RECONSTRUCTION

## Intrinsic parameters of the camera – image plane

- the problem presented in the $XOZ$ plane – compute the $u$ coordinate for the 3D point
- the problem presented in the $YOZ$ plane – compute the $v$ coordinate for the 3D point

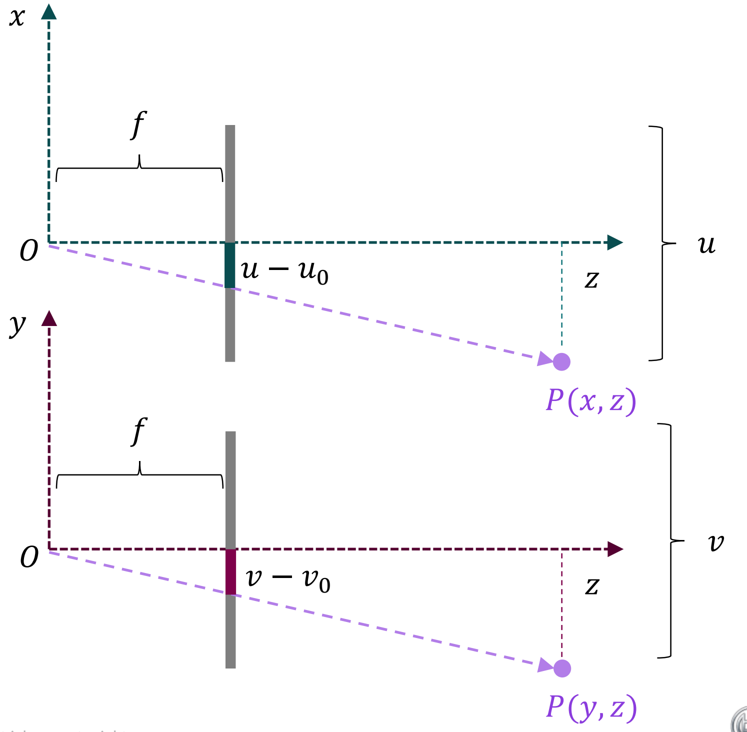


# Intrinsic parameters of the camera – image plane

- the $u$ and $v$ can be obtained using similar triangles

$$\frac{(u-u_0)\cdot dpx}{f}=\frac{x}{z} \qquad \frac{(u-u_0)}{f_x}=\frac{x}{z}$$

$$\frac{(v-v_0)\cdot dpy}{f}=\frac{y}{z} \qquad \frac{(v-v_0)}{f_y}=\frac{y}{z}$$



# Intrinsic parameters of the camera – matrix form

## Intrinsic parameters

- there are 5 intrinsic parameters, which describes the internal geometry and optical characteristics of the camera – focal length, image sensor format and principal point

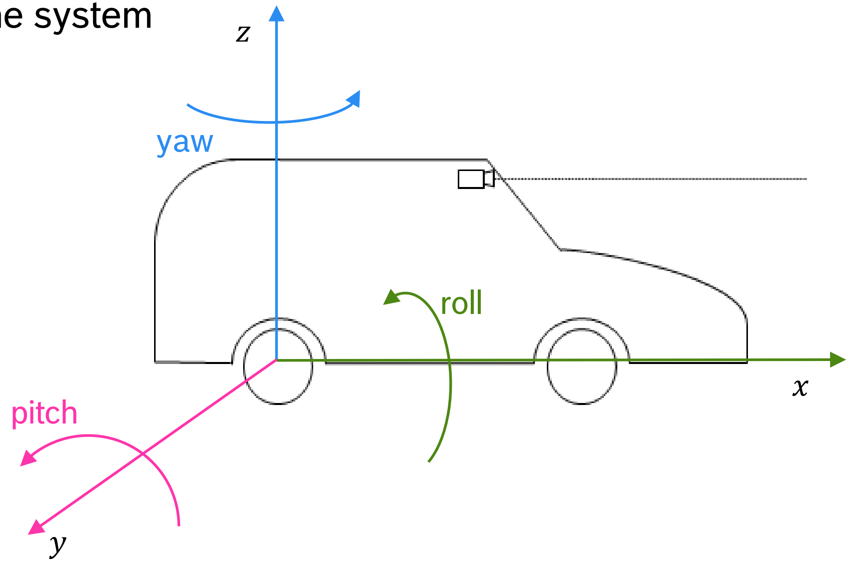
- it can be modelled using the following matrix $K = \begin{bmatrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix}$, where:
  - $f_x = f \cdot dpx$ and $f_y = f \cdot dpy$ represent the focal length in term of pixels, with $dpx$ and $dpy$ being the scale factors and $f$ the focal length
  - $u_0$ and $v_0$ represent the principal point
  - Matrix based projection formula (equivalent to the one from the previous slide): $\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \sim K \cdot \begin{bmatrix} x \\ y \\ z \end{bmatrix}$

# CALIBRATION
## Parameters of the camera

### Extrinsic parameters

- represent position and orientation of the camera in the car (world) coordinate system
  - Convention – use the center of the rear axis
- $R$ matrix for each axis and $T$ represents the translation vector applied on the system
- they can be written in the following form $[R|T] = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix}$
- there are three rotations applied in 3D space – pitch, roll and yaw

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \sim K \cdot \left( R \cdot \begin{bmatrix} x \\ y \\ z \end{bmatrix} + T \right) = K \cdot [R|T] \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$



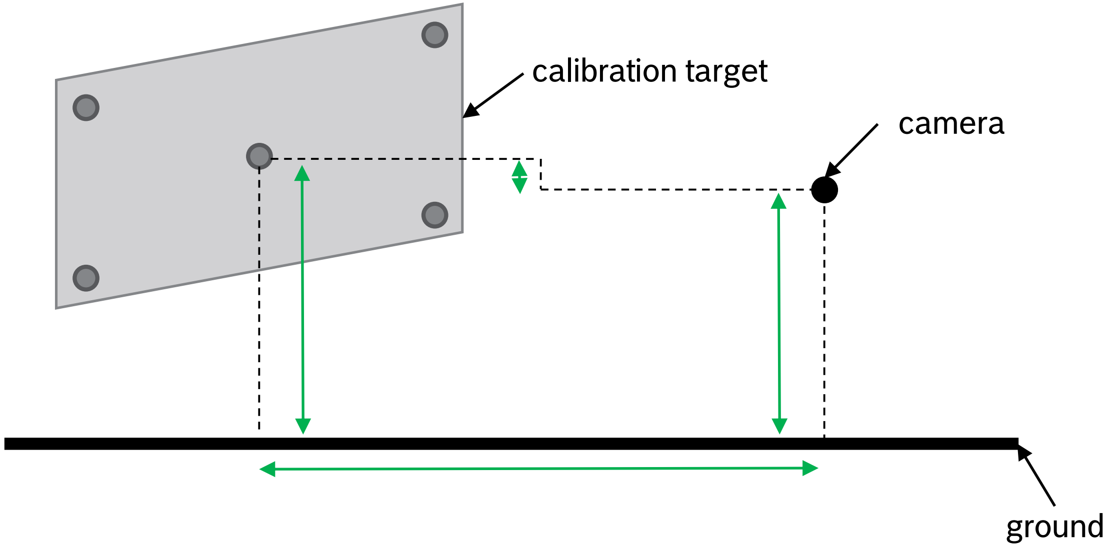


RBRO/ESA1 | 30/08/2017

BOSCH

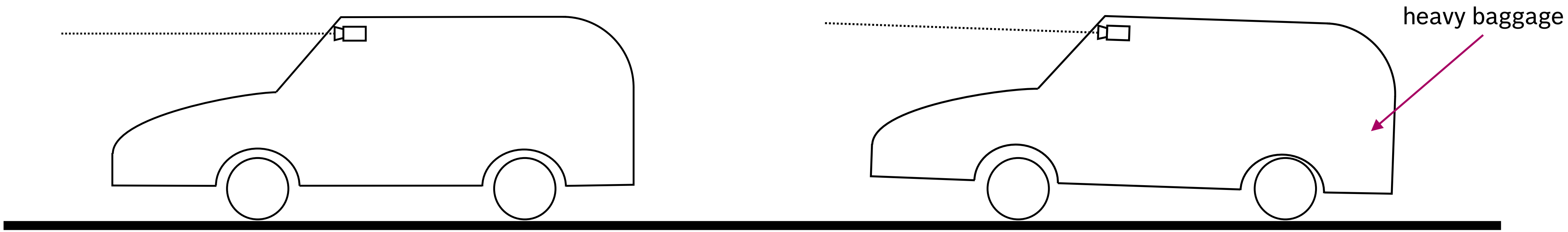
# CALIBRATION
## Static calibration

- align the camera towards a static calibration target with a specific visual pattern
- camera detects the calibration target in the image – sets the detected position in relation to the target's expected position – computes the intrinsic and extrinsic parameters



# CALIBRATION
## Online calibration

- starts from knowing the exact position of the camera with respect to the world system of coordinates
- detect a de-calibration of the camera as soon as possible since an initially determined calibration may change dynamically due to external influences (ex; heavy baggage)

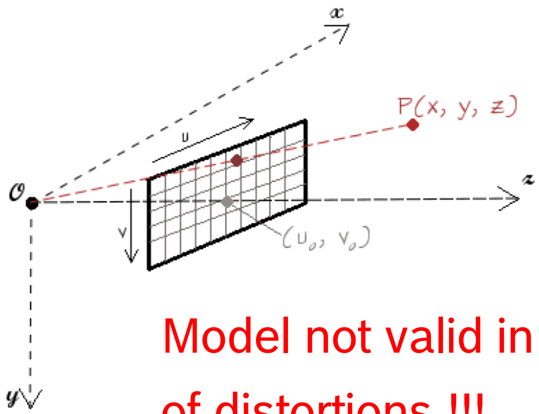


# DISTORTION

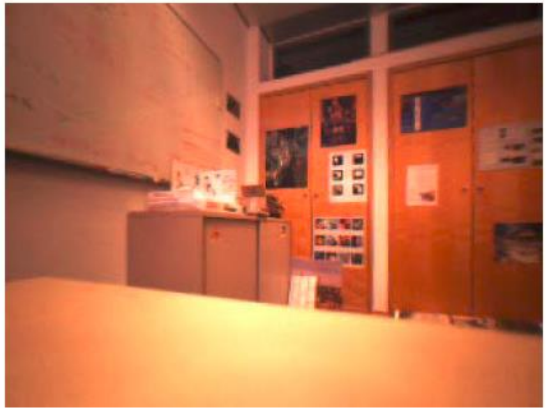
# DISTORTION
## General aspects

- distortion – alteration of the original image
- In the presence of distortions the

perspective projection model does not hold




Model not valid in the presence of distortions !!!

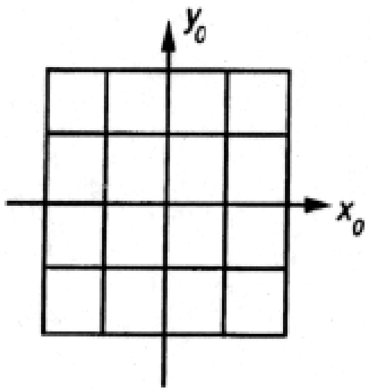


Radial Distortion – Source – Dynamic Vision, T. Schon

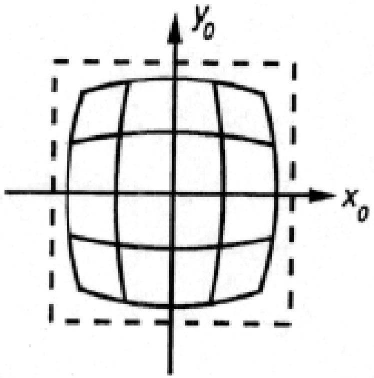
# DISTORTION

## Types of distortion

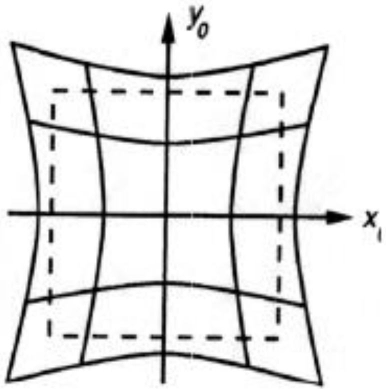
1. <u>Radial distortion</u>
   - induced by the curve of the lens
   - there are two types – barrel distortion or pincushion distortion



no distortion



barrel distortion



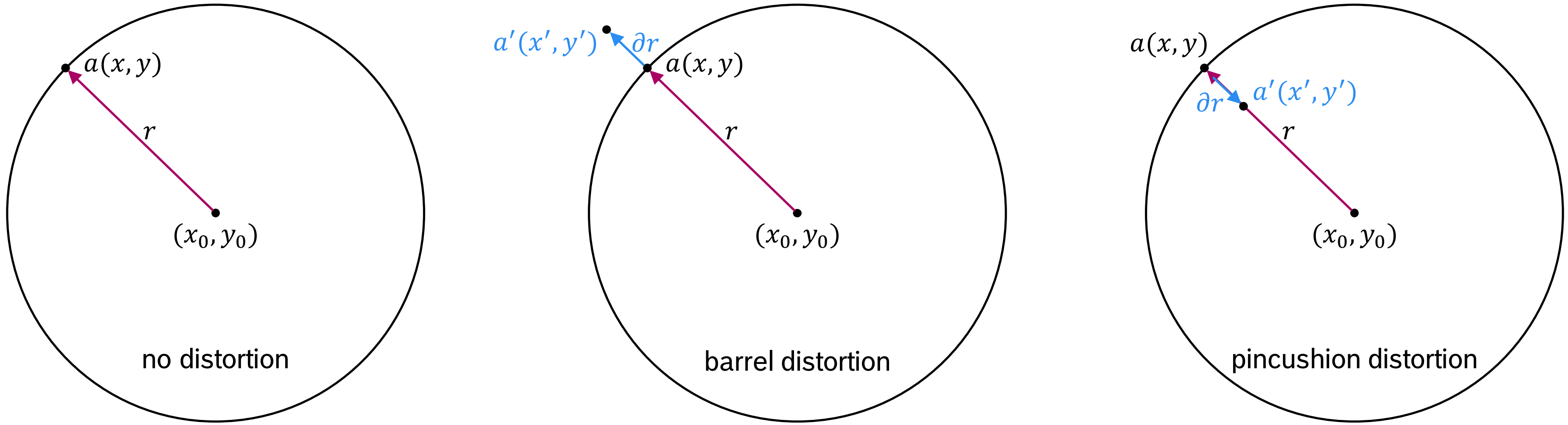
pincushion distortion

# DISTORTION

## Types of distortion

1. <u>Radial distortion</u>

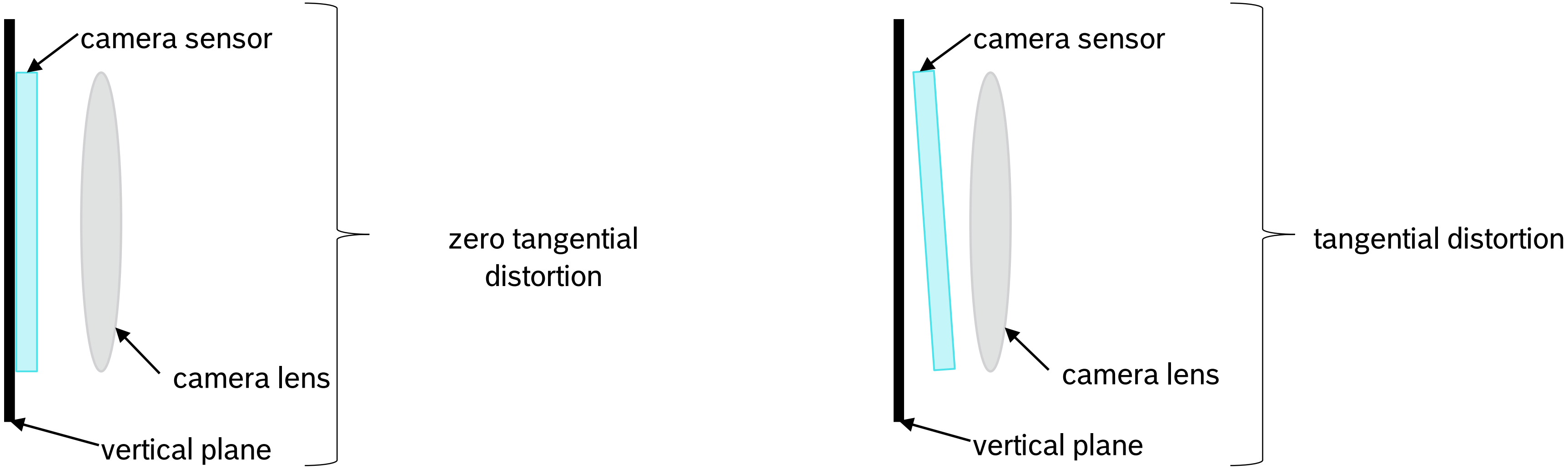
- it can be modeled using the following equation: $\begin{bmatrix} x_{undistorted} - x_{distorted} \\ y_{undistorted} - y_{distorted} \end{bmatrix} = \begin{bmatrix} x \cdot (k_1 \cdot r^2 + k_2 \cdot r^4 + \cdots) \\ y \cdot (k_1 \cdot r^2 + k_2 \cdot r^4 + \cdots) \end{bmatrix}$, where $r^2 = x^2 + y^2$ and $k_1, k_2, \dots$ are the coefficients of radial distortion



# DISTORTION
## Types of distortion

2. <u>Tangential distortion</u>

- induced by the misalignment of the center of camera lens and camera sensors

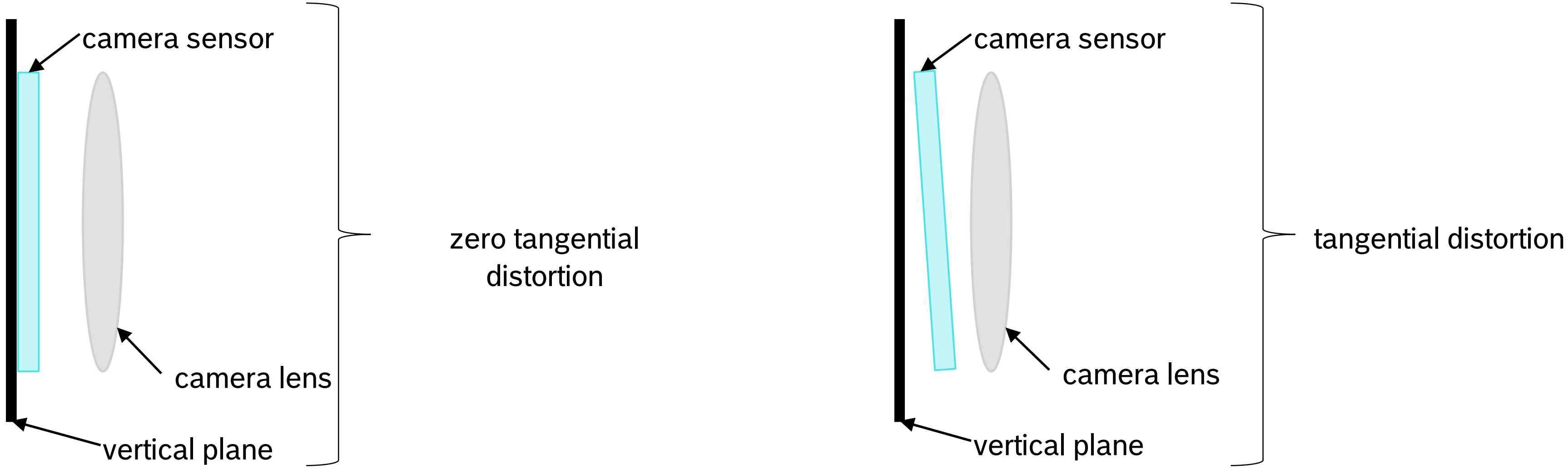


# DISTORTION

## Types of distortion

### 2. Tangential distortion

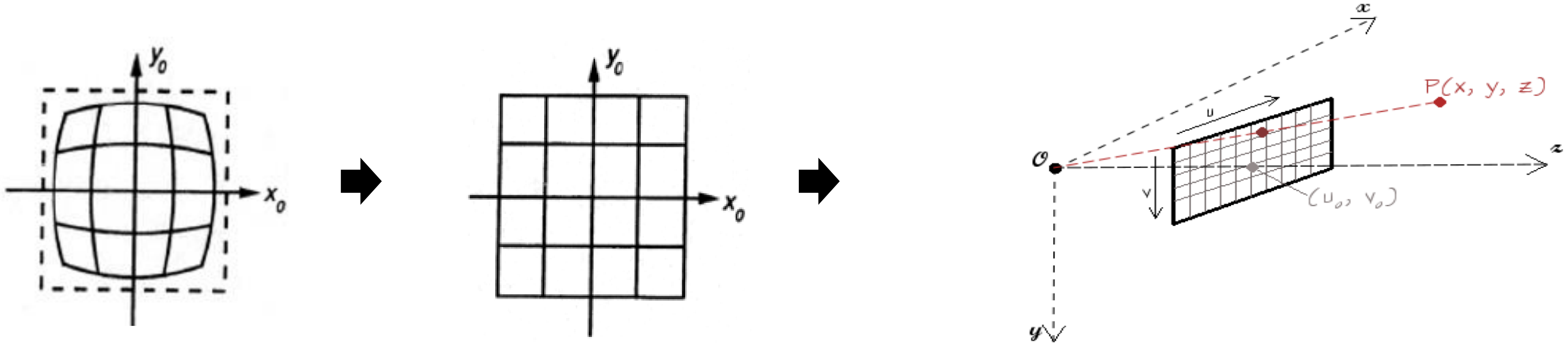
- it can be modeled using: $\begin{bmatrix} x_{undistorted} - x_{distorted} \\ y_{undistorted} - y_{distorted} \end{bmatrix} = \begin{bmatrix} 2 \cdot p_1 \cdot x \cdot y + p_2 \cdot (r^2 + 2 \cdot x^2) \\ p_1 \cdot (r^2 + 2 \cdot y^2) + 2 \cdot p_2 \cdot x \cdot y \end{bmatrix}$, where $p_1, p_2$ are the coefficients of tangential distortion



# PROJECTIVE GEOMETRY

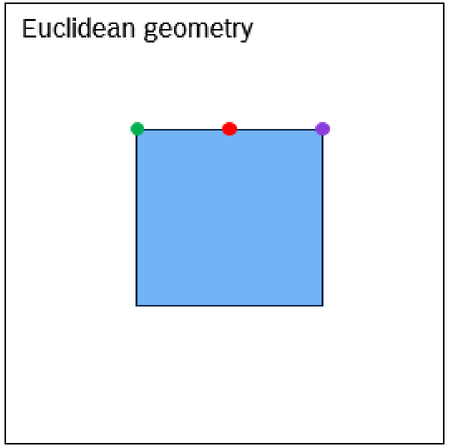
# PROJECTIVE GEOMETRY
## General aspects

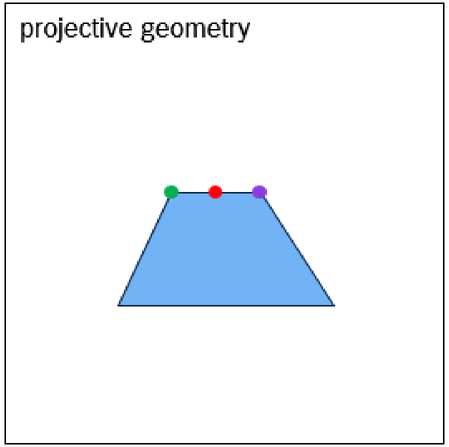
- a branch of geometry which deals with the properties of the objects that are invariant under projective transformations
- this can be applied after the elimination of the lens distortions (i.e. when the image formation model can be modeled based on a perspective projection)
- projective geometry is a very useful tool in computer vision



# PROJECTIVE GEOMETRY
## General aspects

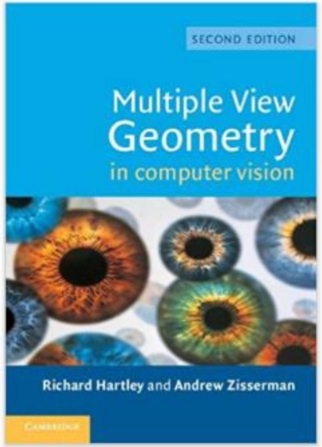
- difference between Euclidean and projective geometry:
  - Euclidean geometry – describe objects “as they are” (unchanged by rigid motions)
  - projective geometry – describe objects “as they appear”

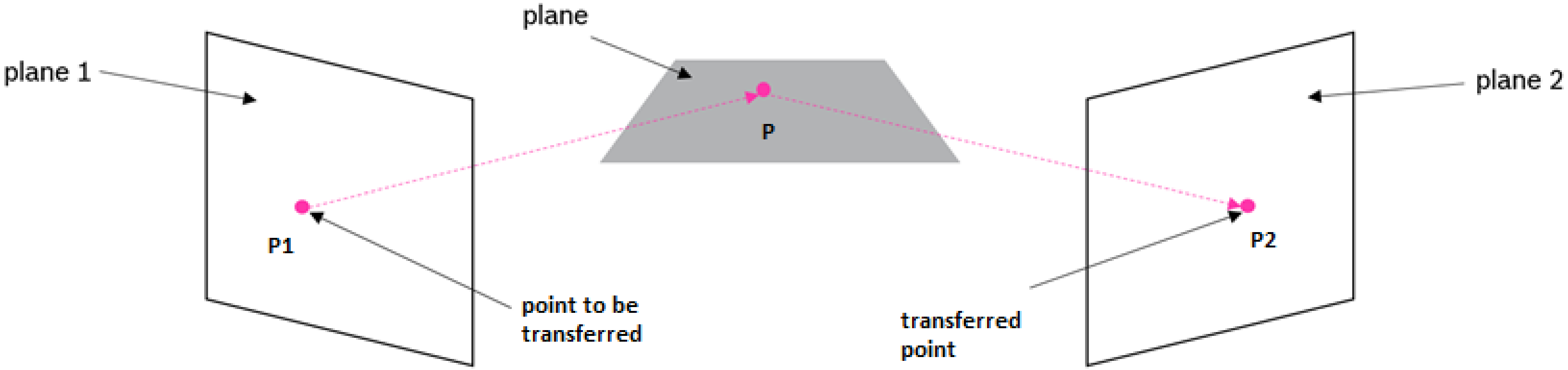



# PROJECTIVE GEOMETRY
## References and example result from projective geometry

- https://www.amazon.com/Multiple-View-Geometry-Computer-Vision/dp/0521540518
- http://robotics.stanford.edu/~birch/projective/
- http://robotics.stanford.edu/~birch/projective/projective.pdf
- https://www.robots.ox.ac.uk/~vgg/hzbook/hzbook2/HZepipolar.pdf
- http://mathworld.wolfram.com/ProjectiveGeometry.html




$$P_1 = H_1 \cdot P$$

$$P_2 = H_2 \cdot P$$

$$H_{image} = H_2 \cdot H_1^{-1}$$

$$P_2 = H_{image} \cdot P_1$$

$H_{image}, H_1, H_2$ **– homography matrices**

**Transfer via plane**

**For details see the references**

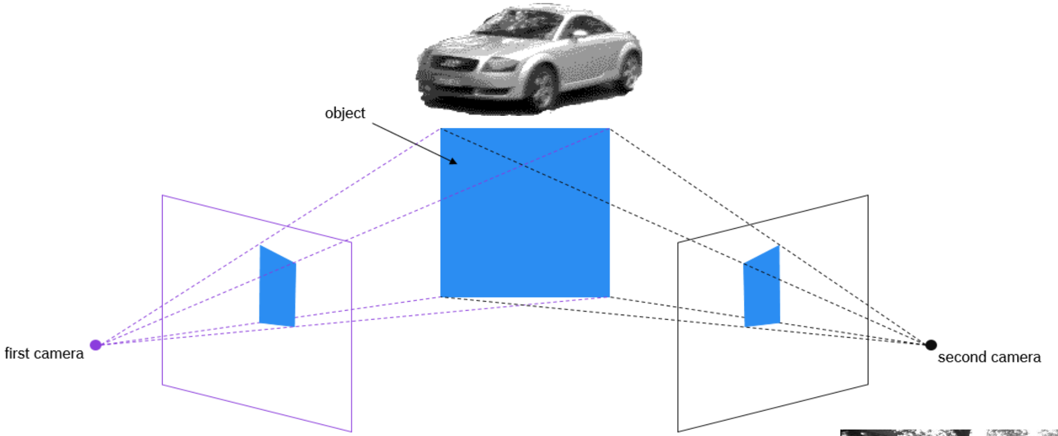
# 3D RECONSTRUCTION

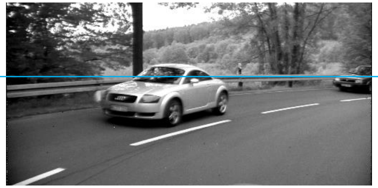
# 3D RECONSTRUCTION
## General concepts

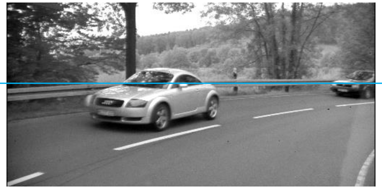
**stereo camera**

**mono camera**

- 3D reconstruction – the process of creating the 3D shape and position of real objects from images
- in computer vision for automated driving
  - using the stereo system - two cameras from different positions, targeting the same scene
  - using the mono system - same camera, targeting the same scene at different points in time





Rectification for 3D reconstruction => corresponding image points are on the same scan-line







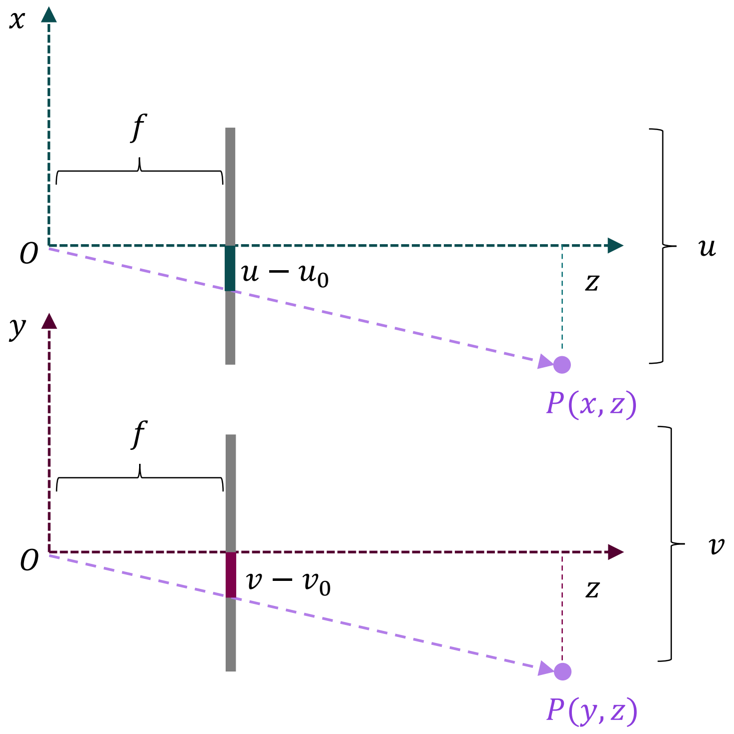
BOSCH

# 3D RECONSTRUCTION
## Inverse problem

- compute $x$ and $y$ coordinates

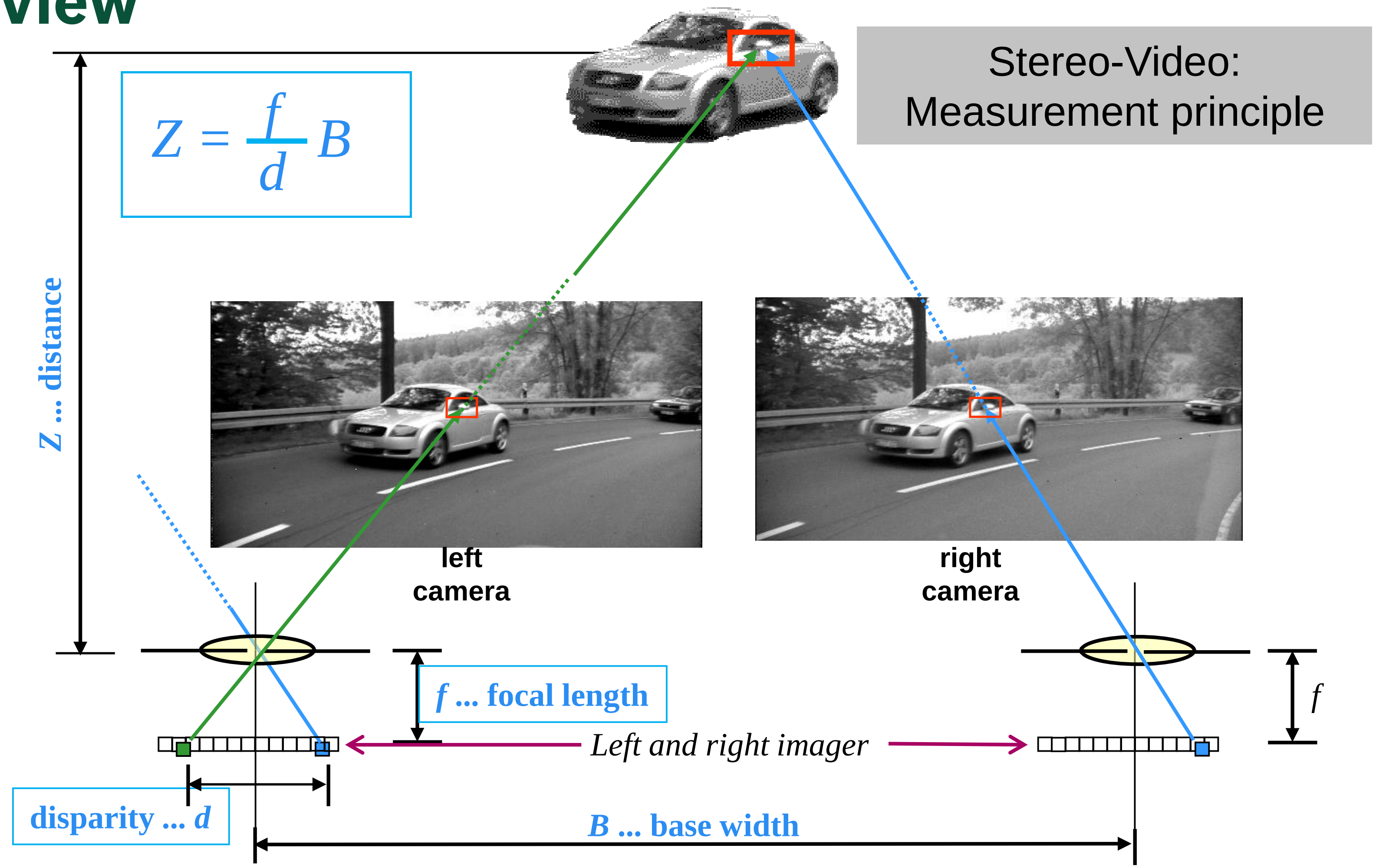
$$\frac{(u-u_0)}{f_x} = \frac{x}{z} \qquad \frac{(u-u_0)\cdot z}{f_x} = x$$

$$\frac{(v-v_0)}{f_y} = \frac{y}{z} \qquad \frac{(v-v_0)\cdot z}{f_y} = y$$



# 3D RECONSTRUCTION

## Disparity review

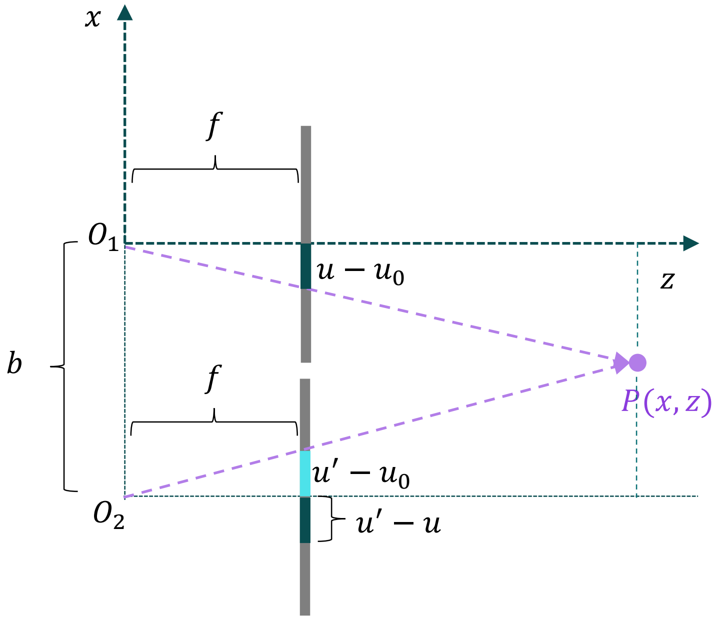





BOSCH
38

# 3D RECONSTRUCTION
## Disparity

- used to determine the depth of an object using the stereo system – we assume that the two cameras are already rectified




$$\frac{(u - u_0)}{f_x} = \frac{x}{z}$$

$$\frac{(u' - u_0)}{f_x} = \frac{x + b}{z}$$

$$\frac{(u' - u)}{f_x} = \frac{b}{z}$$

$$\frac{f_x \cdot b}{u' - u} = z$$

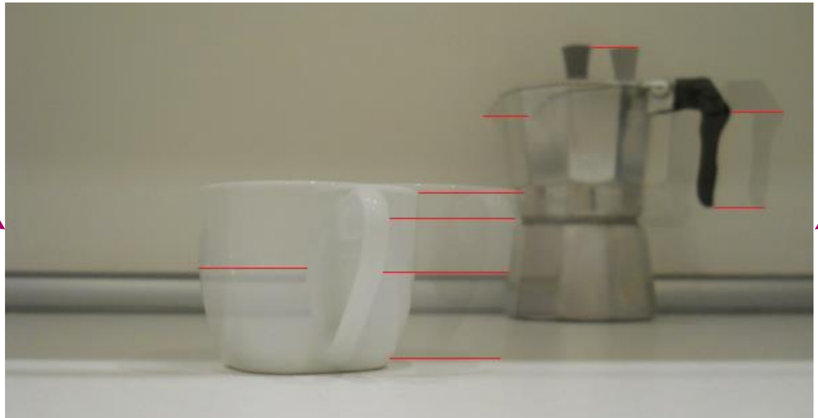
$$\frac{\boldsymbol{f_x \cdot b}}{\boldsymbol{d}} = \boldsymbol{z}$$

d -> disparity -> displacement of the projection between the left & right image

# 3D RECONSTRUCTION
## Disparity

left image

right image



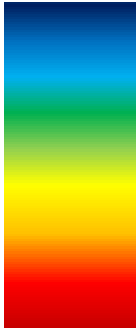
Disparity = displacement between left & right images

Disparity is small in far range and big in close range

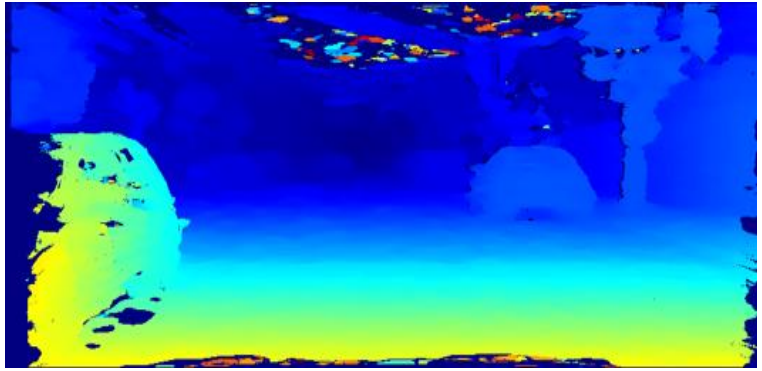
# 3D RECONSTRUCTION
## Disparity

Legend

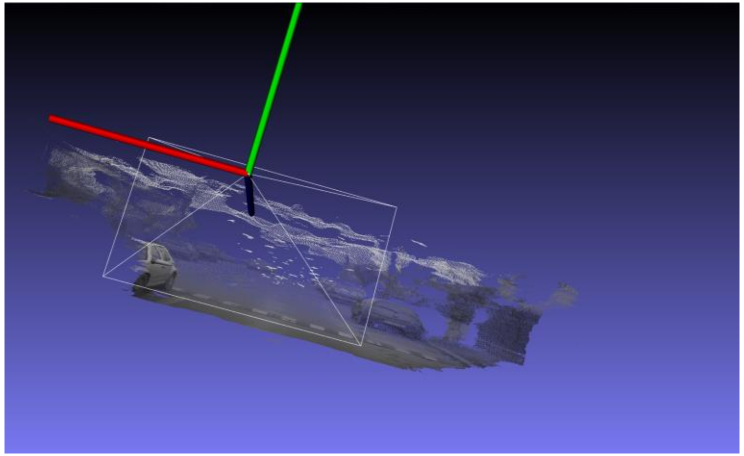
Far objects



Close objects



3D reconstruction



Disparity is small in far range
and big in close range

# Thank you for your attention!