Lecture

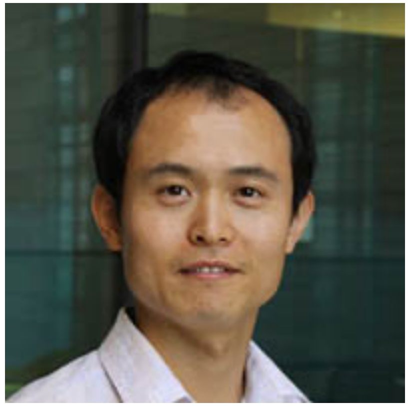
Introduction

Liangliang Nan

Agenda

- The teacher and teaching assistant
- Overview of the course
  - What the course is about
  - What you will learn
  - Topics/Lectures
  - Assignments
  - Final exam
  - Communication method
- Review linear algebra basics

Teacher & teaching assistant



Liangliang Nan
LiangliangNan#0976

Nail Ibrahimli
nibrahimli#5857

What the course is about

- Photogrammetry
- Computer Vision
- 3D Computer Vision

What are the differences?

What the course is about

- Photogrammetry
  - General
    - Obtaining info about objects or environment
      - Recording: digital image capturing
      - Measuring: generating 2D or 3D measurements
      - Interpreting: detecting interested objects
    - Output can be
      - A map
      - A drawing
      - 3D model
      - …

What the course is about

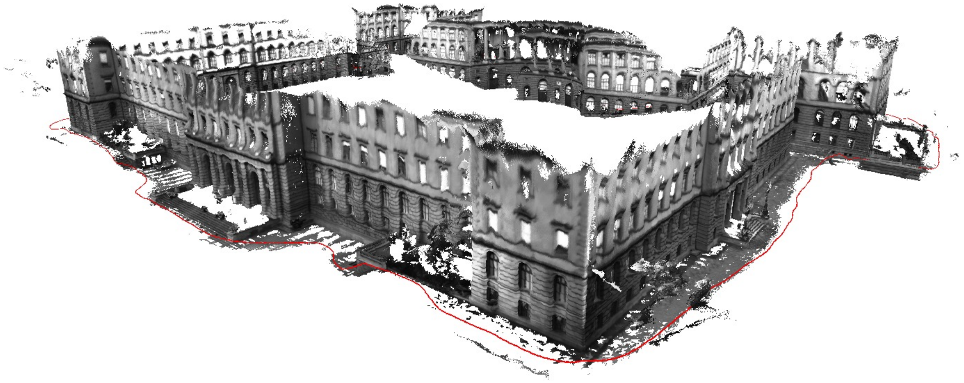
- Photogrammetry
  - General
  - Specific
    - Extracting 3D geometry from images
    - Goal: generating 3D digital models of an object
    - Using well defined photogrammetric methods
      - e.g., structure from motion

What the course is about

- Photogrammetry
- Computer vision
  - Training computers to interpret and understand the visual world
  - Using digital images or videos
  - Mimics the human visual system
    - Video tracking
    - Object detection/recognition
    - Scene reconstruction
    - …

What the course is about

- Photogrammetry
- Computer vision
- 3D computer vision
  - Scene reconstruction/modeling
  - 3D data processing (e.g., semantic segmentation, classification)



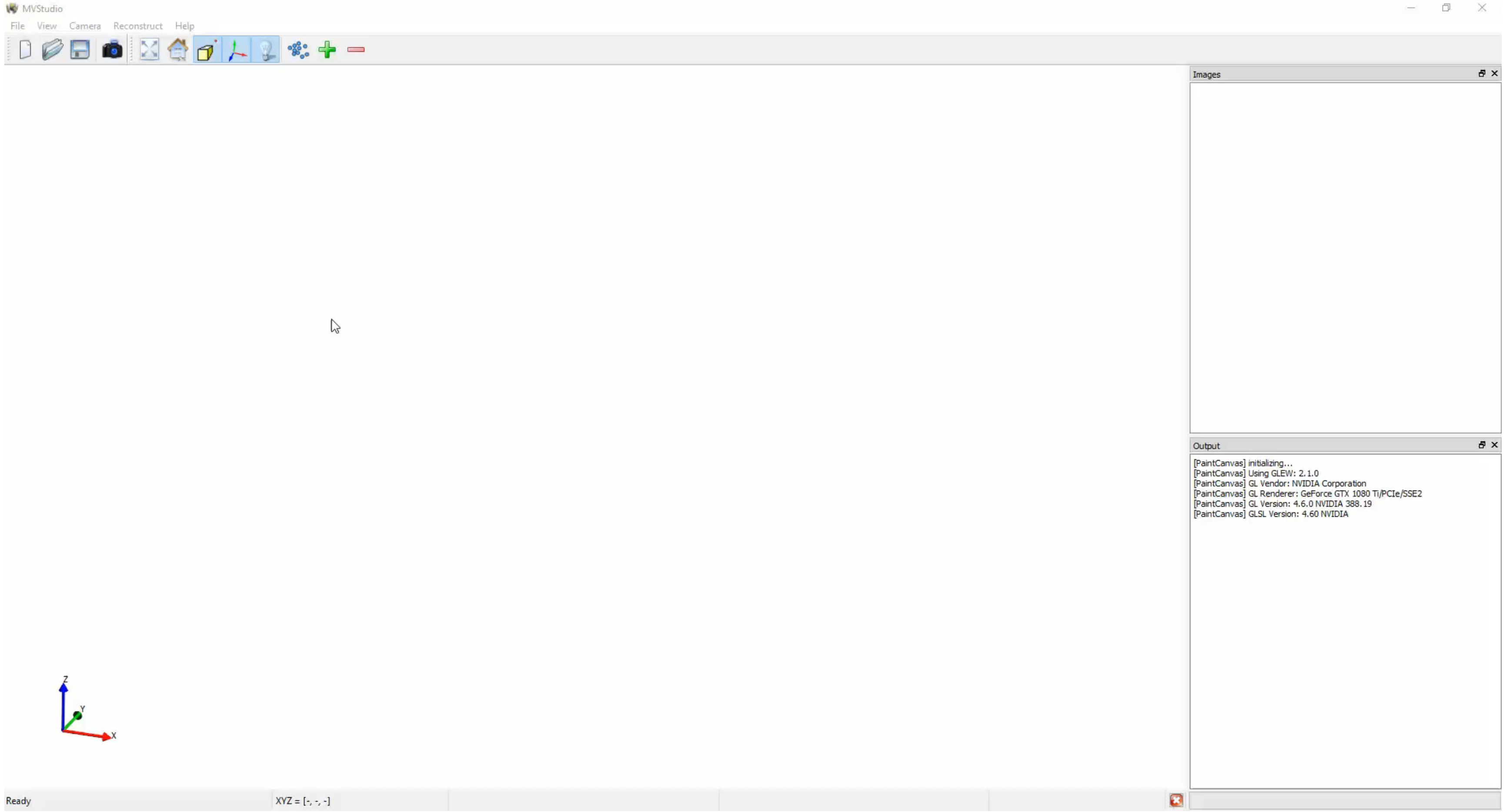
What the course is about

- Photogrammetry
- Computer vision
- 3D computer vision
  - Scene reconstruction/modeling
  - 3D data processing (e.g., semantic segmentation, classification)

3D computer vision == Photogrammetry

What the course is about

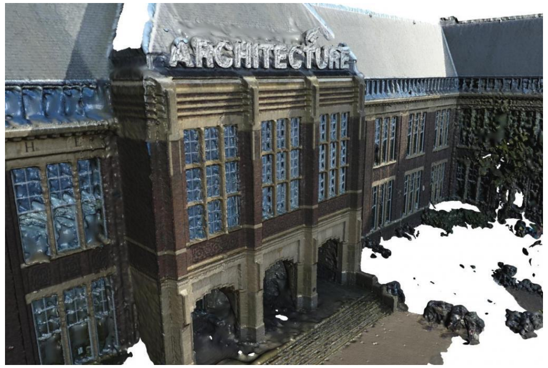
- Demo



Code & data: https://github.com/LiangliangNan/MVStudio

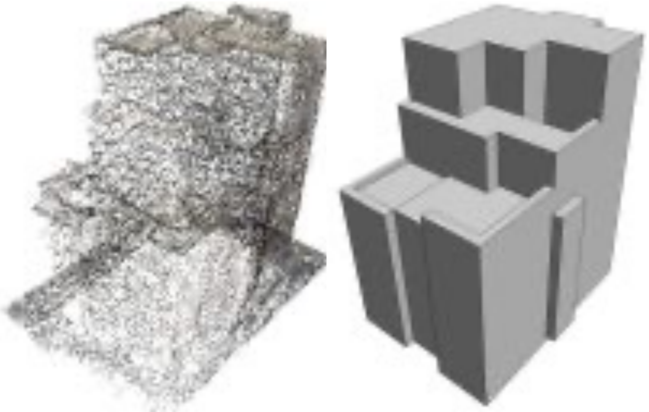
What the course is about

- Photogrammetry
- Computer vision
- 3D computer vision
  - Scene reconstruction/modeling
  - 3D data processing (e.g., semantic segmentation, classification)
  - Urban objects



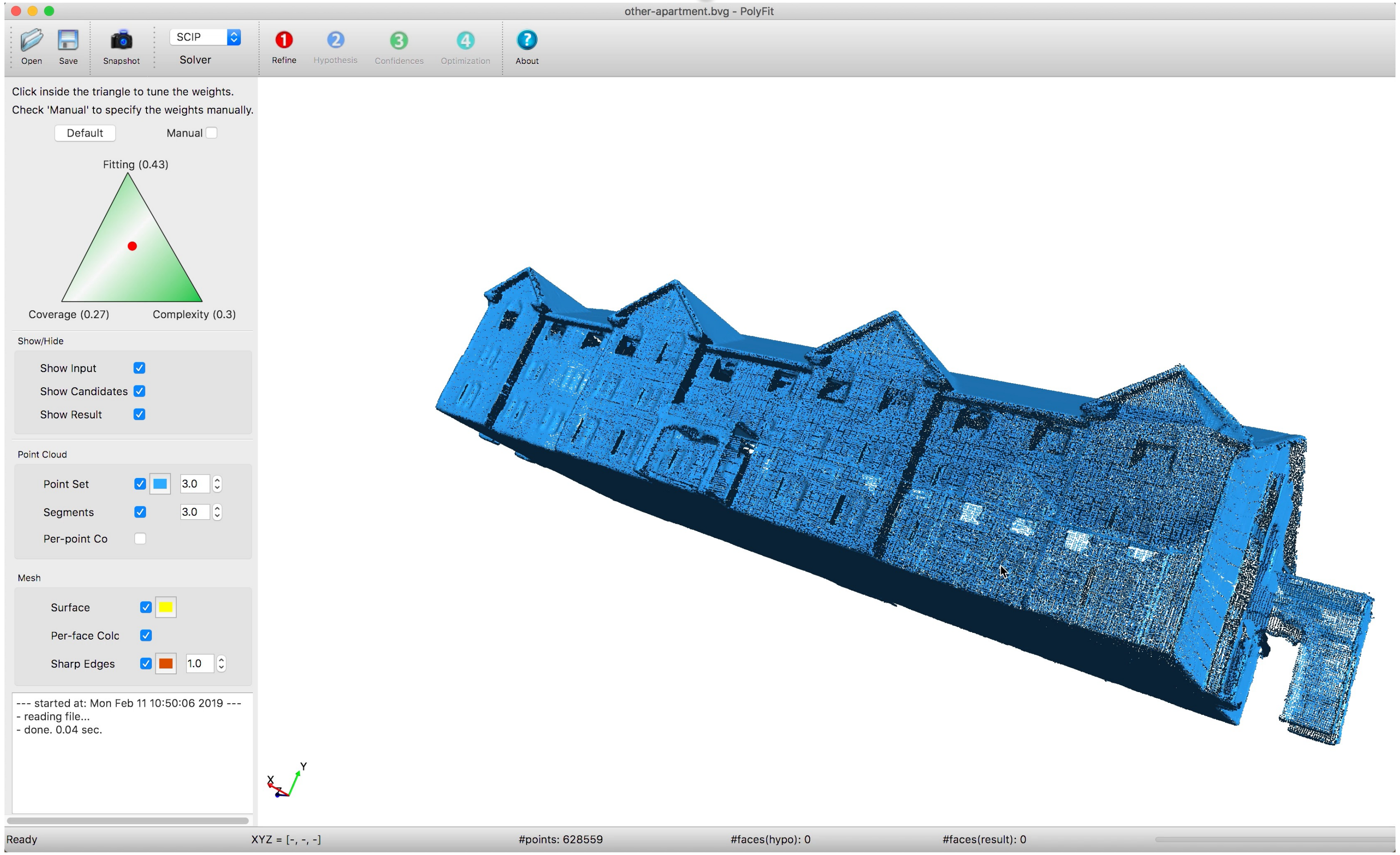
What the course is about

- Photogrammetry
- Computer vision
- 3D computer vision
  - Scene reconstruction/modeling
  - 3D data processing (e.g., semantic segmentation, classification)
  - Urban objects
  - Point clouds vs. Surface models



What the course is about

- Demo



Code & data: https://github.com/LiangliangNan/PolyFit

Applications of 3DV

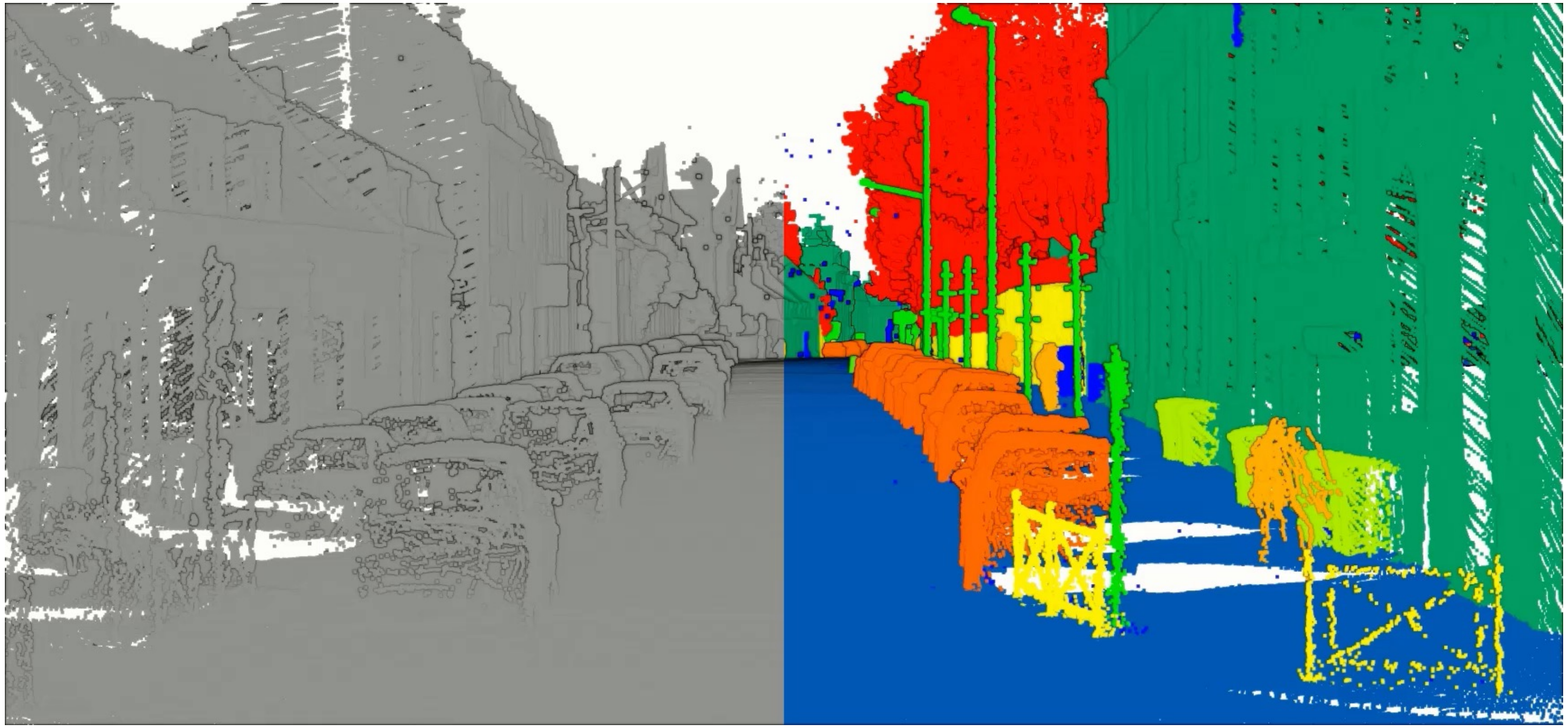
- Façade parsing and 3D modeling

(a) (b) (c) (d) (e) (f) (g)

Nan et al. Template Assembly for Detailed Urban Reconstruction. *Computer Graphics Forum, Vol. 34, No. 2, 2015*

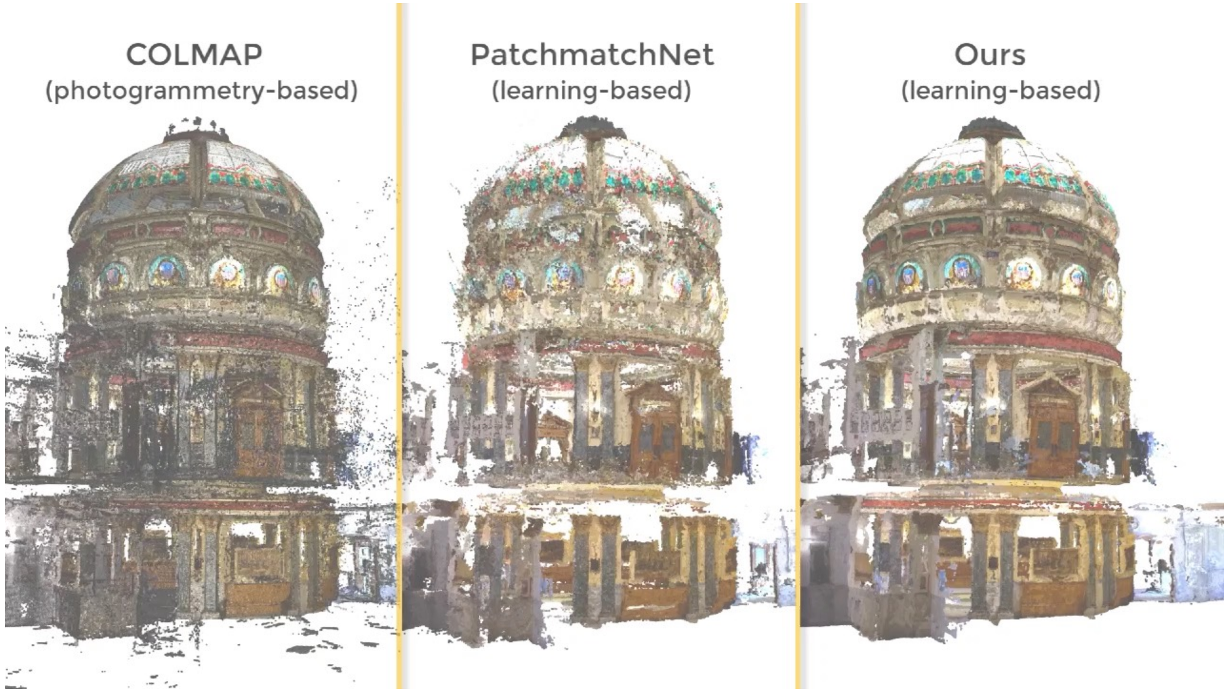
Applications of 3DV

- Semantic segmentation



Applications of 3DV

- 3D reconstruction from images



What you will learn

- Theory, methodology, and algorithms
- The complete pipeline for modelling real-world objects (mainly buildings)
  - Data acquisition
  - Processing
  - Reconstruction

Topics/Lectures

- 1,2: Introduction & Linear algebra
- 3,4: Camera models
- 5,6: Camera calibration
- 7,8: Epipolar geometry
- 9,10: Triangulation & structure from motion
- 11,12: Multi-view stereo [by Nail]
- 13,14: Surface reconstruction

Learning activities

- Lectures
  - 2 hours per week
- Assignments
  - 2 x 2-hour sessions/week
  - Teachers available at Geolab
    - Data acquisition
    - Install/Use software
    - Programming
    - Debugging
    - Discussion with teammates and with teachers

Assessment

- 3 group assignments (40 %)
  - – Group performance
  - – Personal contribution/Peer reviews
- Final exam (60%):
  - – Lectures, handouts, assignments, lab exercises
    - Multiple-choice questions
    - Open questions

Assessment

- 3 group assignments (40 %)
- Final exam (60 %)
- Pass?
  - Assignments >= 5.5
  - Exam >= 5.5
  - Total of 6.0 or above

Assignments

- Programming
  - 3D reconstruction and point cloud processing
- Each assignment released after the lecture
- Work in groups (3 students per group)

Assignments

- Programming
- Each assignment released after the lecture
- Work in groups (3 students per group)
- C++ source code framework provided
- If you fail one of the assignment?
  - Depending code and data will be provided

Assignments

- What to submit?
  - Report
    - < 3 pages (excluding figures, tables, references)
    - Individual contribution
    - See an example on Course Webpage.

Isaac Newton (75 %)

- Compared the reconstruction results from method [1] and method [2];
- Implemented the function reorient_normals ();
- Came up with a novel reconstruction method and implemented it in function reconstruct();
- Wrote the "Methodology" section of the report.

Albert Einstein (20 %)

- Preparing and pre-processing of the point clouds, i.e., taking photos, run SfM and MVS, cropping the buildings from the messy point clouds, and normal estimation;
- Wrote the "Implementation Details" section of the report.

Thomas Edison (5 %)

- Wrote the "Abstract" section of the report.

Assignments

- What to submit?
  - Report
  - Code
    - Collaboration using GitHub
    - Include the link to the GitHub repository in the report

Assignments

- What to submit?
  - Report
  - Code
    - Collaboration using GitHub
    - Include the link to the GitHub repository in the report
    - Reproduce the results
      - **Doesn’t compile:** -10%
      - **Doesn’t reproduce the result:** -10%

Assignments

- What to submit?
- We allow multiple submissions
  - Incorporating comments from teachers/peers
  - Evaluation based on 1st submission + 0.5 maximum

Example:

First submission 6, then final mark will be <= 6.5

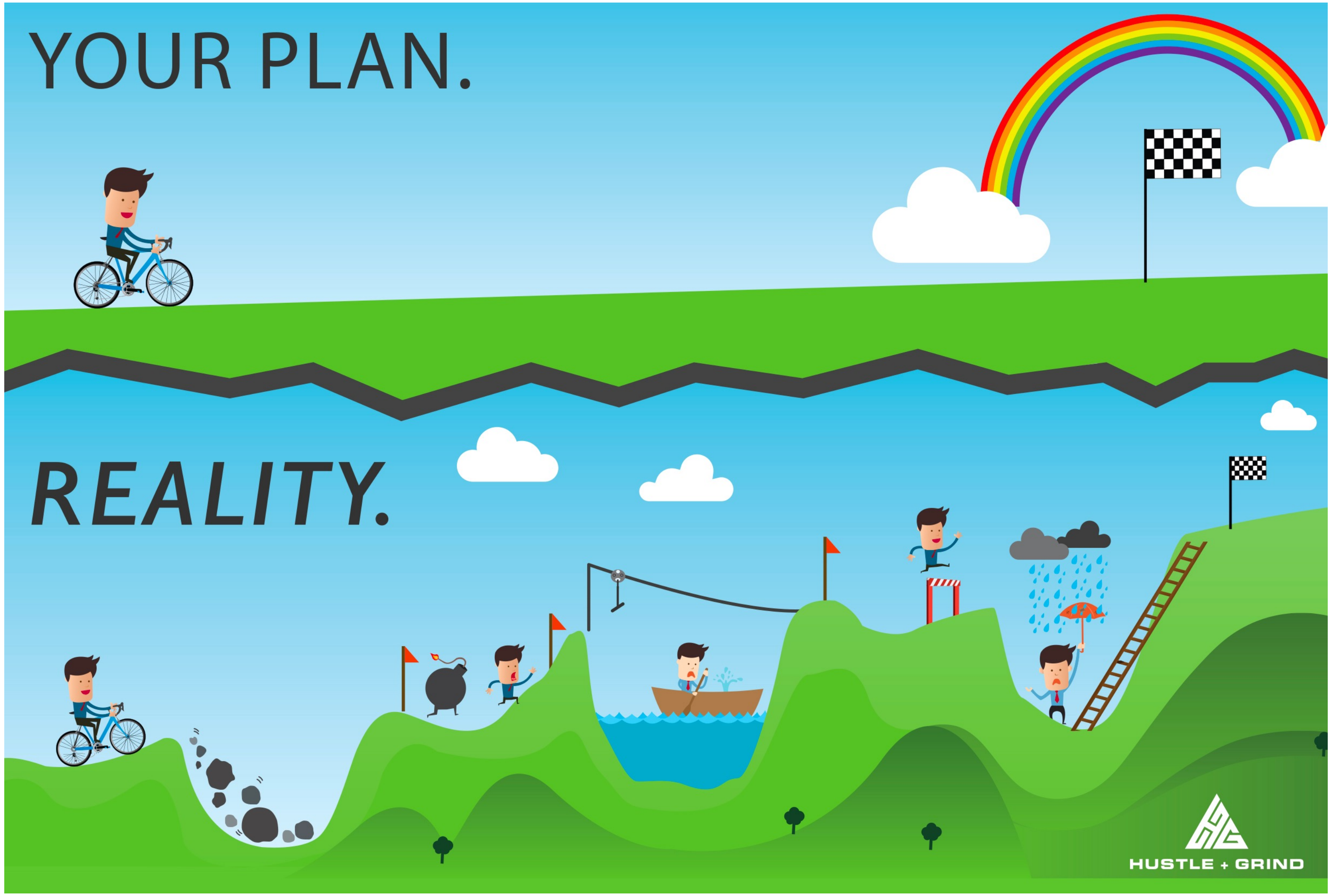
Assignments

- What to submit?
- We allow multiple submissions
- Strict deadline
  - Late submission
    - 10% deducted per day late
    - Note acceptable after 3 days late.

Assignments

- What to submit?
- We allow multiple submissions
- Strict deadline
- Teamwork
  - **Everyone active in coding/discussion/report**
  - **We strongly discourage**
    - report writing to one person and code writing to another
    - one person working on course A and another on course B
    - strategically provide perfectly equal individual contributions

Assignments: start earlier



Plagiarism

- Copy from others/internet, or use ChatGPT
  - Code
  - Sentences
  - Figures
  - …
- Plagiarism checker
  - http://www.icto.tudelft.nl/tools/turnitin/check-for-originality/
  - https://smallseotools.com/plagiarism-checker/
  - https://www.duplichecker.com/
  - …

Communication method

- Course website
  - https://3d.bk.tudelft.nl/courses/geo1016/

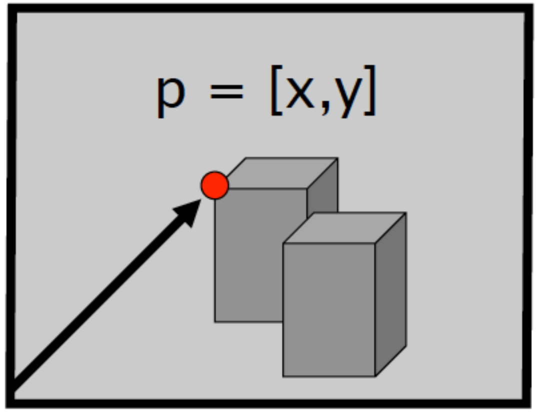
Grouping

- Find your teammates …
  - 3 students per team
  - Click on following link and put your name, student ID, and email address

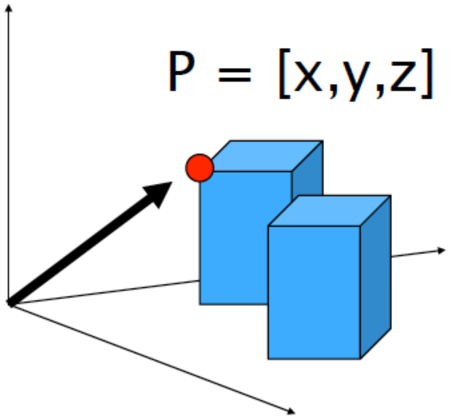
  https://docs.google.com/document/d/1OWDYkJDDq5JFBhp04Ken22KWe6Isb44JtXBqGSnM9As/edit?usp=sharing

Review linear algebra

Vectors (i.e., 2D and 3D vectors)



Image

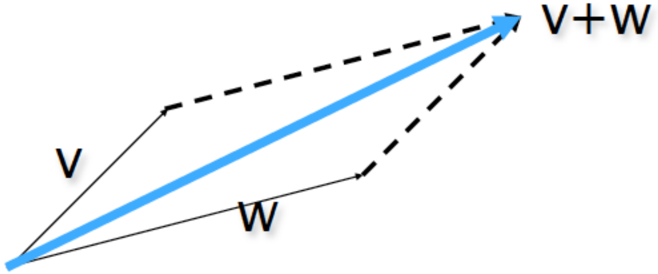


3D world

Vector arithmetic

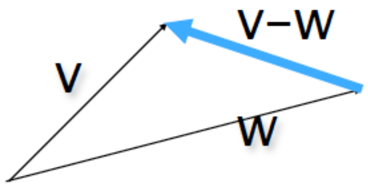
- Addition

$$\mathbf{v} + \mathbf{w} = (x_1, x_2) + (y_1, y_2) = (x_1 + y_1, x_2 + y_2)$$



- Subtraction

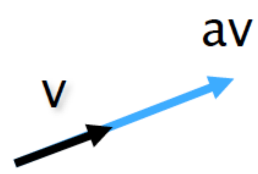
$$\mathbf{v} - \mathbf{w} = (x_1, x_2) - (y_1, y_2) = (x_1 - y_1, x_2 - y_2)$$



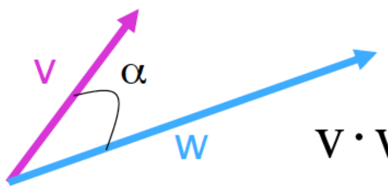
Vector arithmetic

- Scalar Product

$$a\mathbf{v} = a(x_1, x_2) = (ax_1, ax_2)$$



- Dot (inner) product



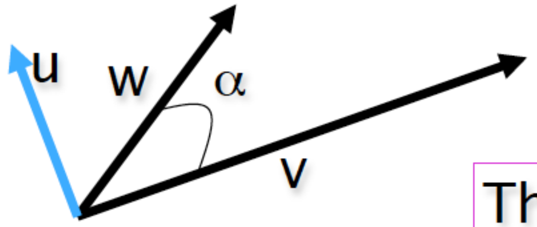
$$v \cdot w = (x_1, x_2) \cdot (y_1, y_2) = x_1 y_1 + x_2 y_2$$

The inner product is a SCALAR!

$$v \cdot w = (x_1, x_2) \cdot (y_1, y_2) = \| v \| \cdot \| w \| \cos\alpha$$

Vector arithmetic

- Cross (vector) Product

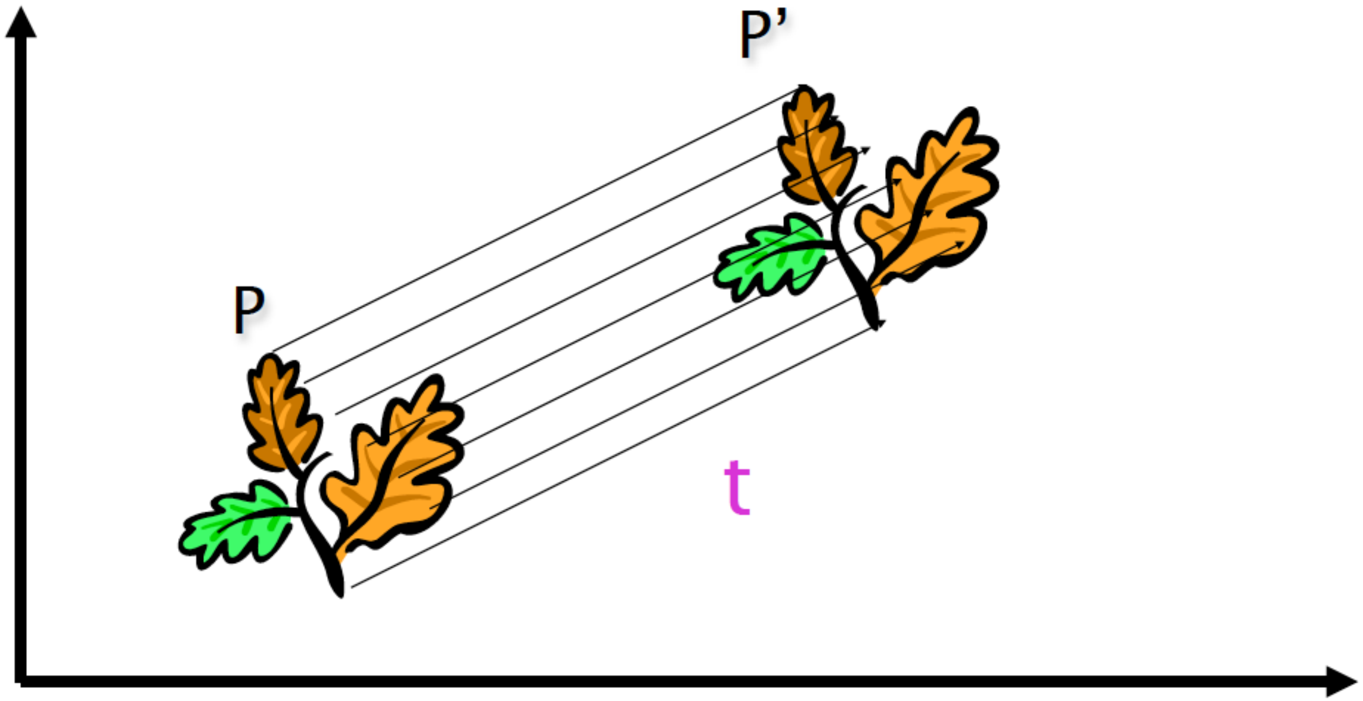


$$u = v \times w$$

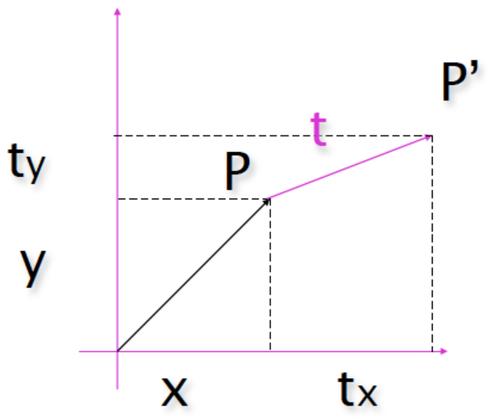
The cross product is a VECTOR!

$$\text{Magnitude:} \|u\| = \|v \times w\| = \|v\| \|w\| \sin \alpha$$

Translation



Translation



$$\mathbf{P} = (x, y) \rightarrow (x, y, 1)$$

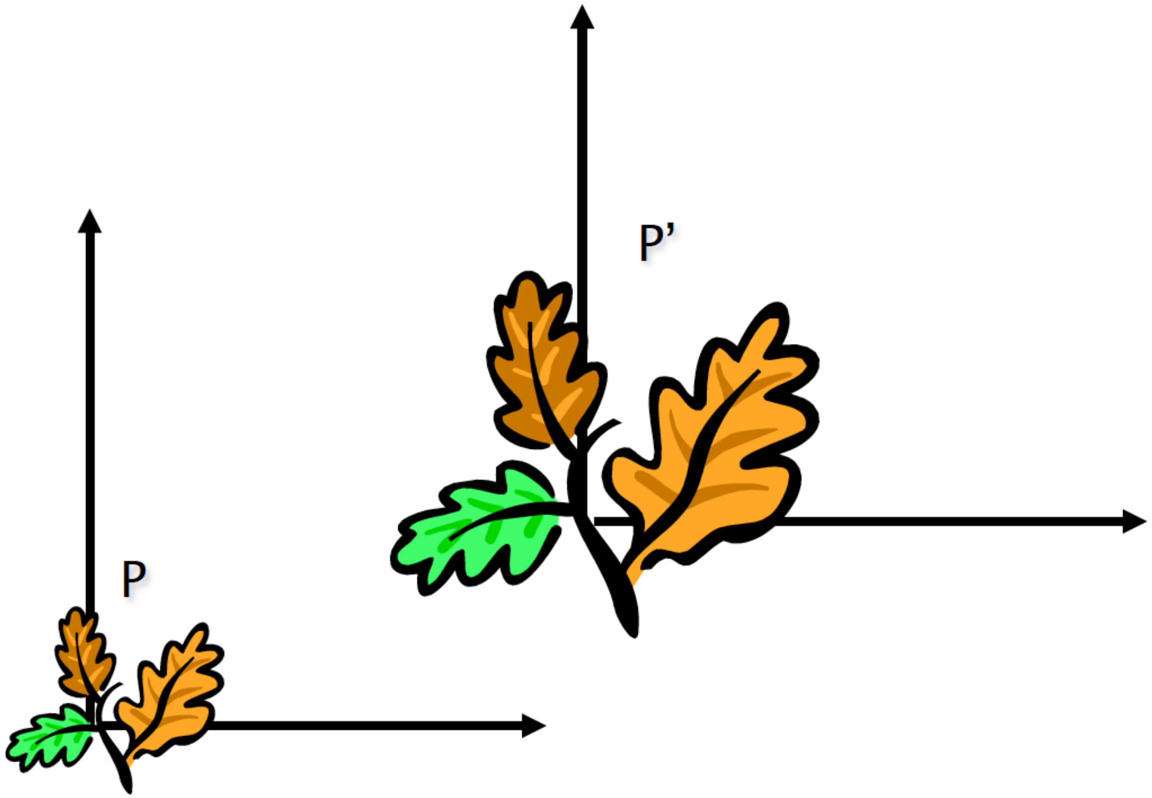
$$\mathbf{t} = (t_x, t_y) \rightarrow (t_x, t_y, 1)$$

t P

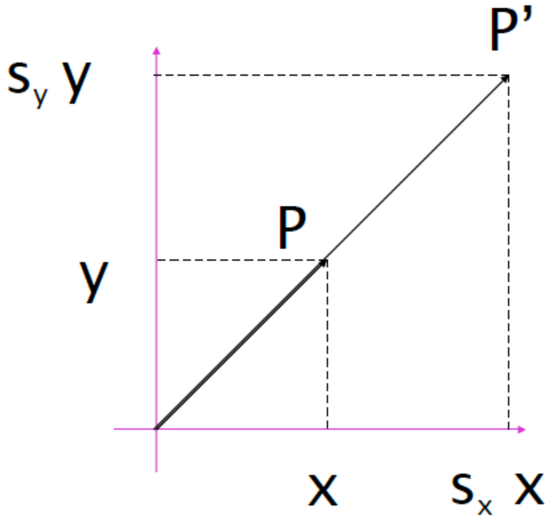
$$\mathbf{P}' \rightarrow \begin{bmatrix} x + t_x \\ y + t_y \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{I} & \mathbf{t} \\ 0 & 1 \end{bmatrix} \cdot \mathbf{P} = \mathbf{T} \cdot \mathbf{P}$$

Scaling



Scaling



$$\mathbf{P} = (\mathrm{x}, \mathrm{y}) \rightarrow \mathbf{P}' = (\mathrm{s_x x}, \mathrm{s_y y})$$

$$\mathbf{P} = (x, y) \rightarrow (x, y, 1)$$

$$\mathbf{P}' = (s_x x, s_y y) \rightarrow (s_x x, s_y y, 1)$$

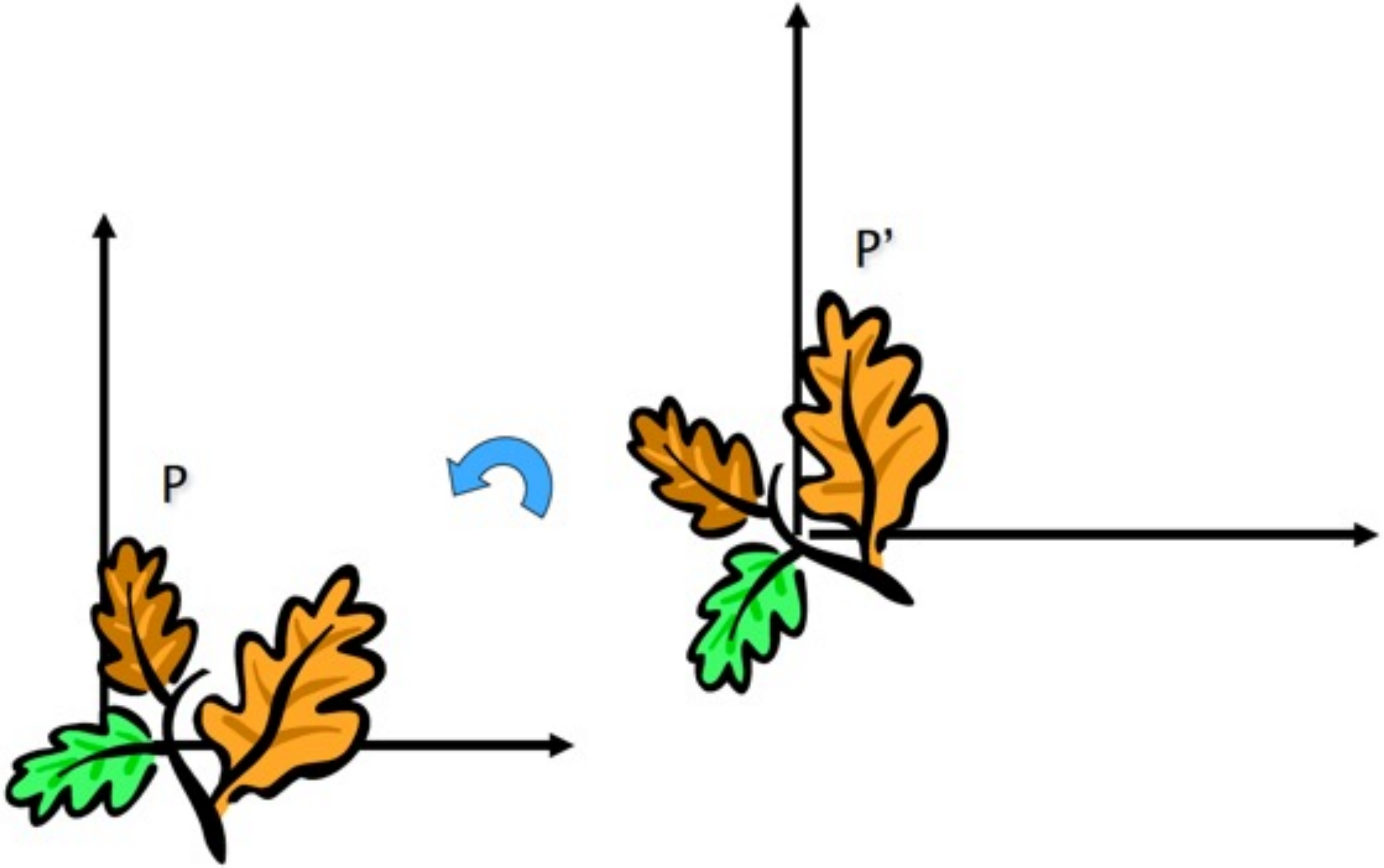
$$\mathbf{P}' \rightarrow \begin{bmatrix} s_x x \\ s_y y \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\mathbf{S}} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{S}' & \mathbf{0} \\ \mathbf{0} & \mathbf{1} \end{bmatrix} \cdot \mathbf{P} = \mathbf{S} \cdot \mathbf{P}$$

Scaling & Translation

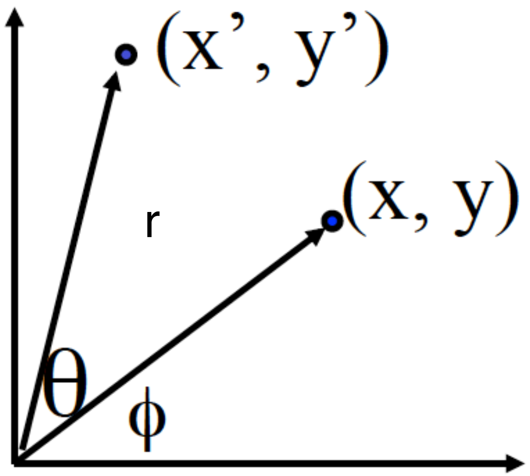
$$\mathbf{P}'' = \mathbf{T} \cdot \mathbf{S} \cdot \mathbf{P} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} =$$

$$= \underbrace{\begin{bmatrix} s_x & 0 & t_x \\ 0 & s_y & t_y \\ 0 & 0 & 1 \end{bmatrix}}_{A} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Rotation



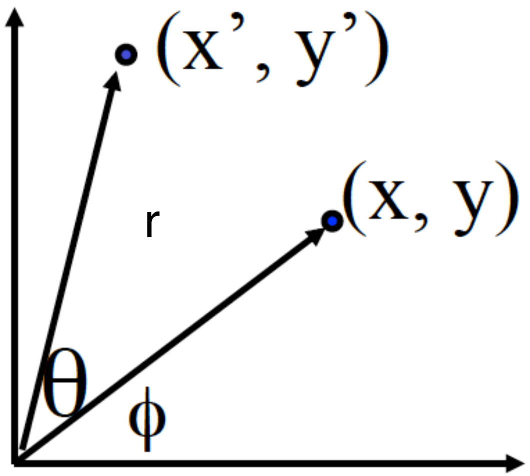
Rotation



$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \underbrace{\begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}}_{\mathbf{R}} \begin{bmatrix} x \\ y \end{bmatrix}$$

What is the inverse transformation

Rotation



$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \underbrace{\begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}}_{\mathbf{R}} \begin{bmatrix} x \\ y \end{bmatrix}$$

What is the inverse transformation

- Rotation by –θ

R has many interesting properties:

$\mathbf{R}^{-1} = \mathbf{R}^{T}$ $\quad \mathbf{R} \cdot \mathbf{R}^{\mathrm{T}} = \mathbf{R}^{\mathrm{T}} \cdot \mathbf{R} = \mathbf{I}$ $\quad \det(\mathbf{R}) = 1$

Translation + Rotation + Scaling

$$\mathbf{P}' = \mathbf{T}\cdot\mathbf{R}\cdot\mathbf{S}\cdot\mathbf{P} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} =$$

$$= \begin{bmatrix} \cos\theta & -\sin\theta & t_x \\ \sin\theta & \cos\theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} =$$

$$= \begin{bmatrix} R' & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} S & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} R'S & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

If $s_x = s_y$, this is a similarity transformation

Next lecture

- Camera models

$$\mathbf{p} = M\mathbf{P}$$
$$= K \begin{bmatrix} R & \mathbf{t} \end{bmatrix} \mathbf{P}$$

Internal (intrinsic) parameters

External (extrinsic) parameters