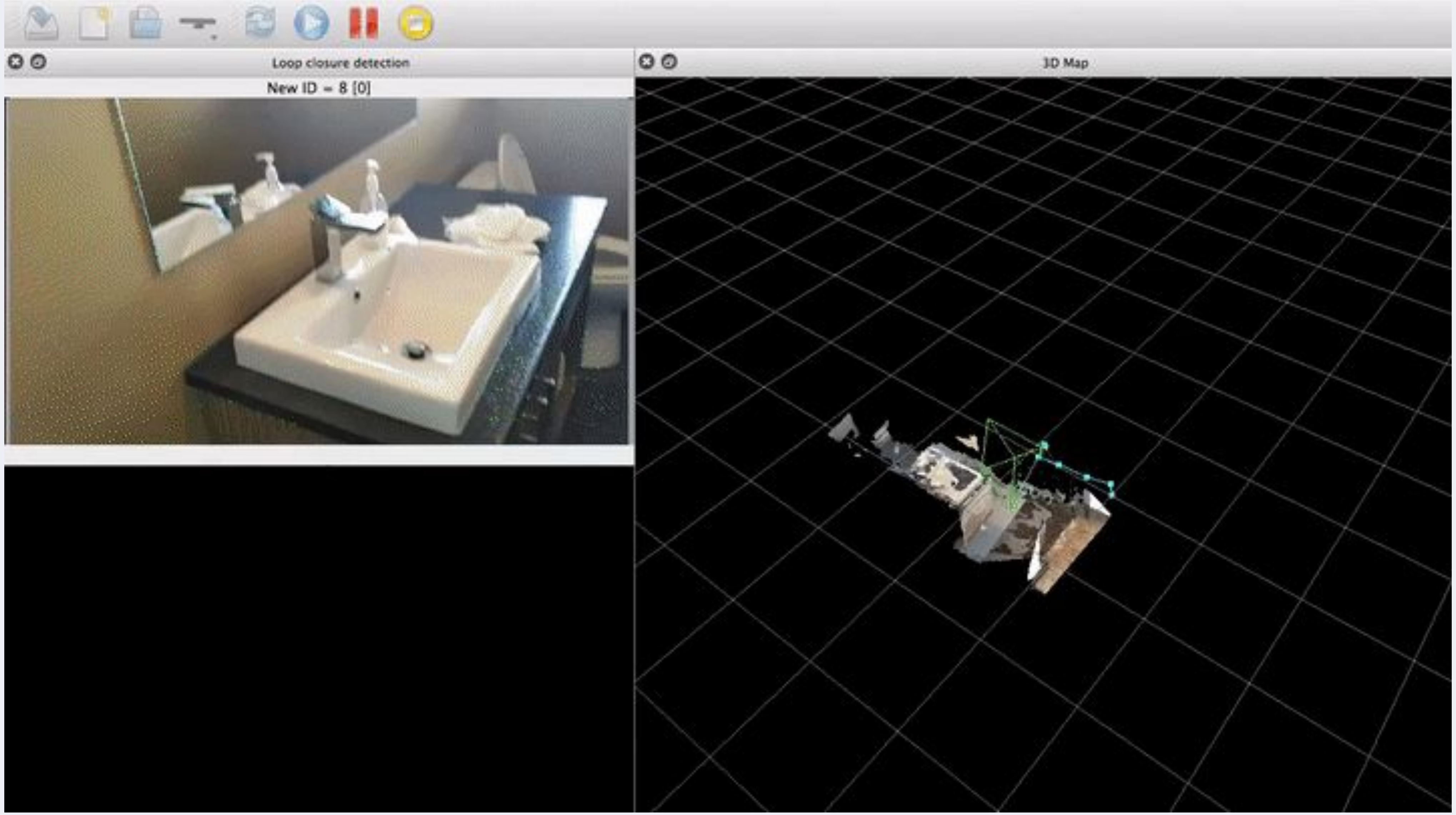
Computer Vision project: Indoor SLAM

Shalygin Igor, Tsoy Anton, Kozlov Kiril

NSU FIT, 2023

What Indoor SLAM is

Method of constructing the most accurate indoor map by processing video captured by a moving camera



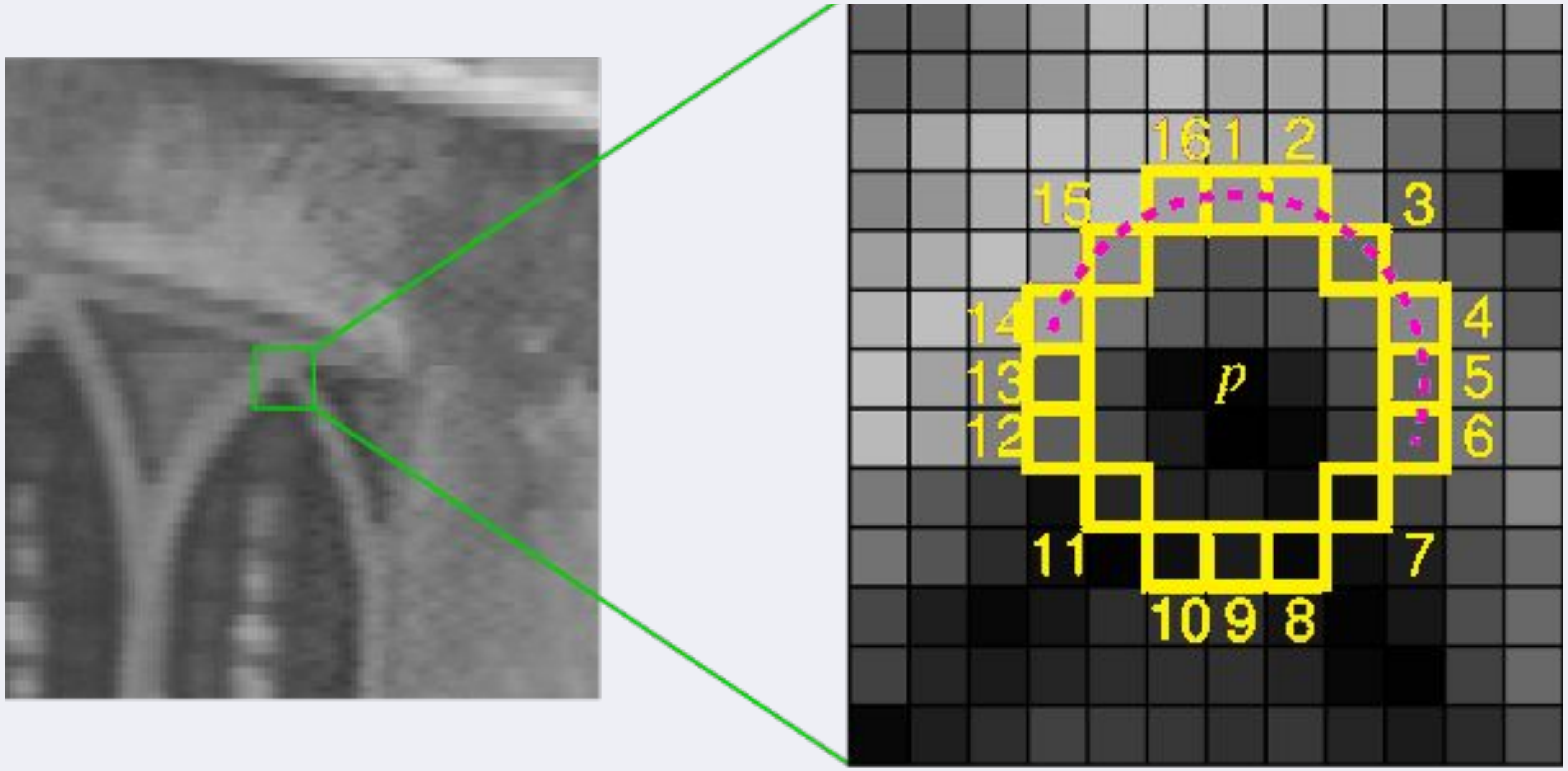
Implement Indoor SLAM

OpenCV

Feature extracting

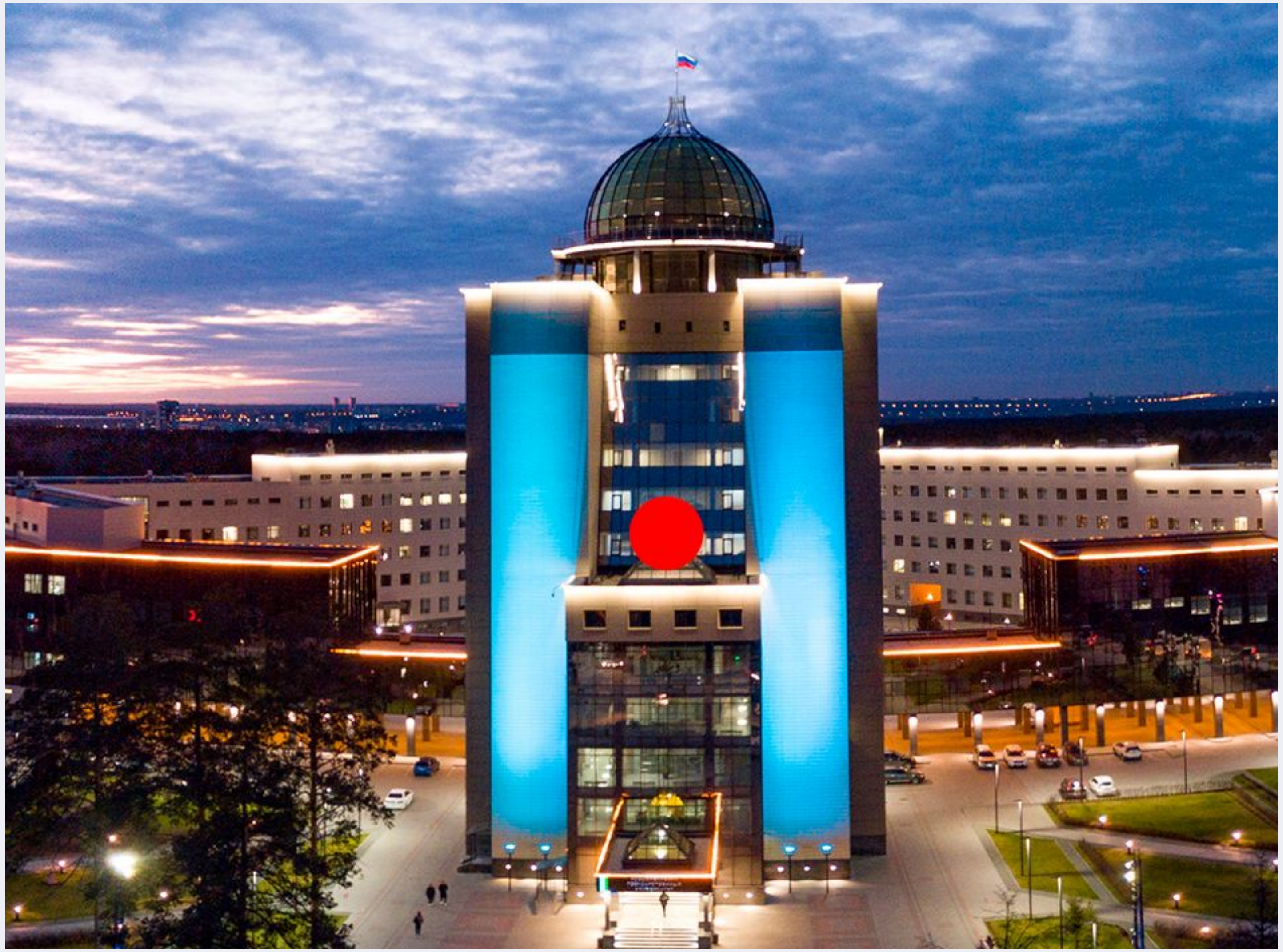
Choose candidate pixel P → Compare the intensity with circle → EXTRACT

SKIP



Feature tracking using *Least Sum Difference* algorithm

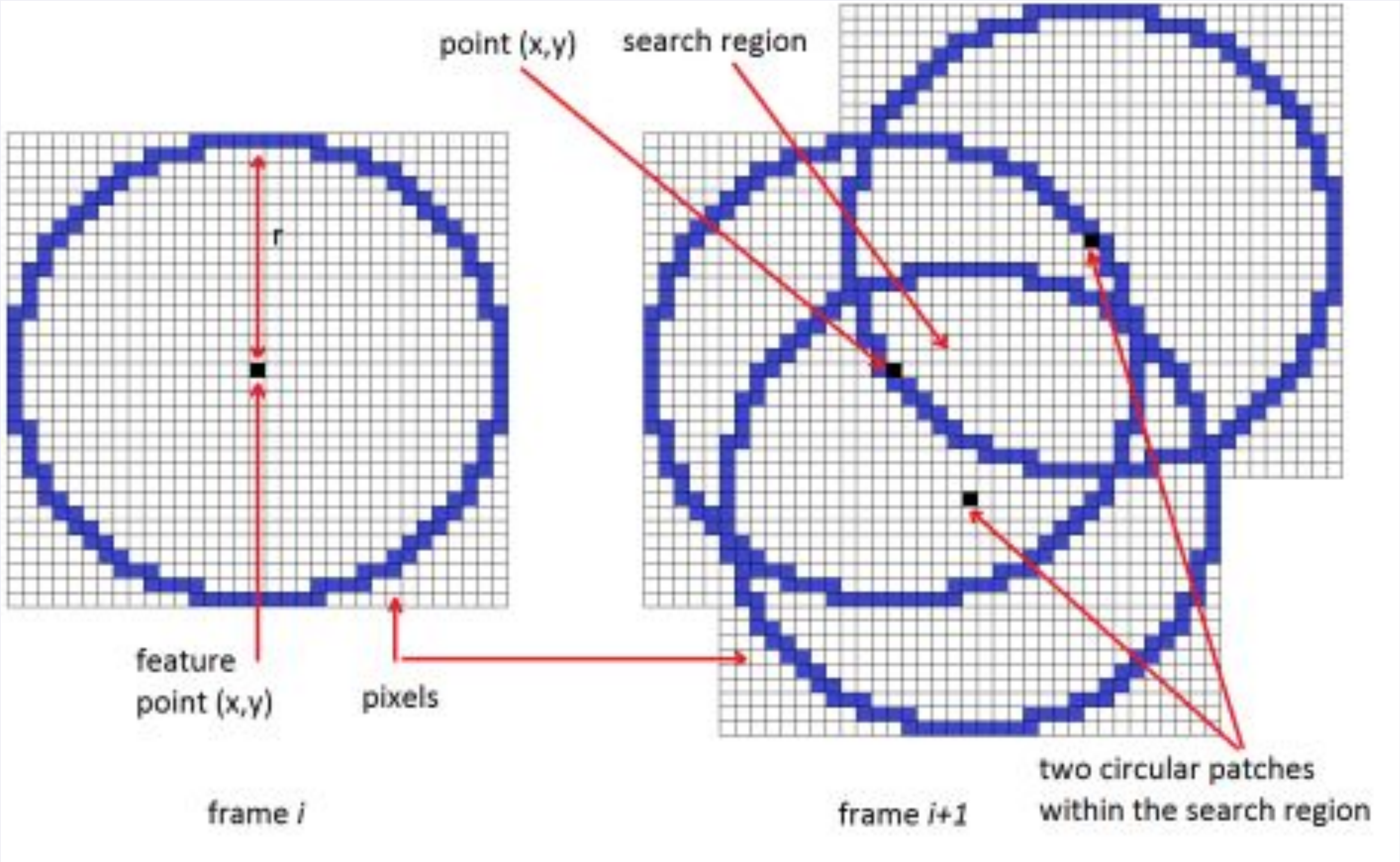
We propose feature in the second frame moved within some **search region** with radius **R**.



The feature's search region.

Computing circular patch pixels

For each pixel within the **search region** we find the **circular patch** with radius **R** in the second frame.



How the Least Sum Difference algorithm works.

Then we compute the **SAD** (sum of absolute differences) between every **circular patch β,** and our **search region σ**.

$$\mathrm{SAD}(\beta,\sigma) = \sum_{u=-r}^{u=r} \sum_{v=-r}^{v=r} \mathbf{G}(u+r+1, v+r+1) \cdot |(I_i(x+u, y+v) - I_{i+1}(x+u, y+v))|$$

The patch with the **least SSD** will contain our feature at the center.

First frame **extracted** features

Second frame **tracked** features

World positions

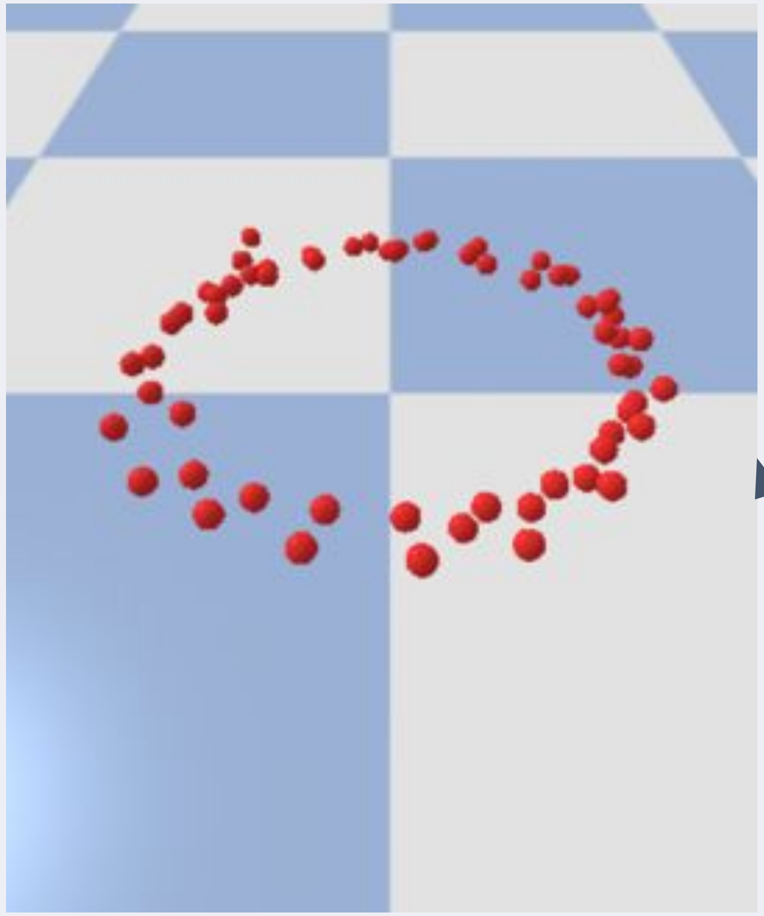
$$R_{world} = R_{new} R_{world}$$
$$t_{world} = t_{world} + t_{new} R_{world}$$

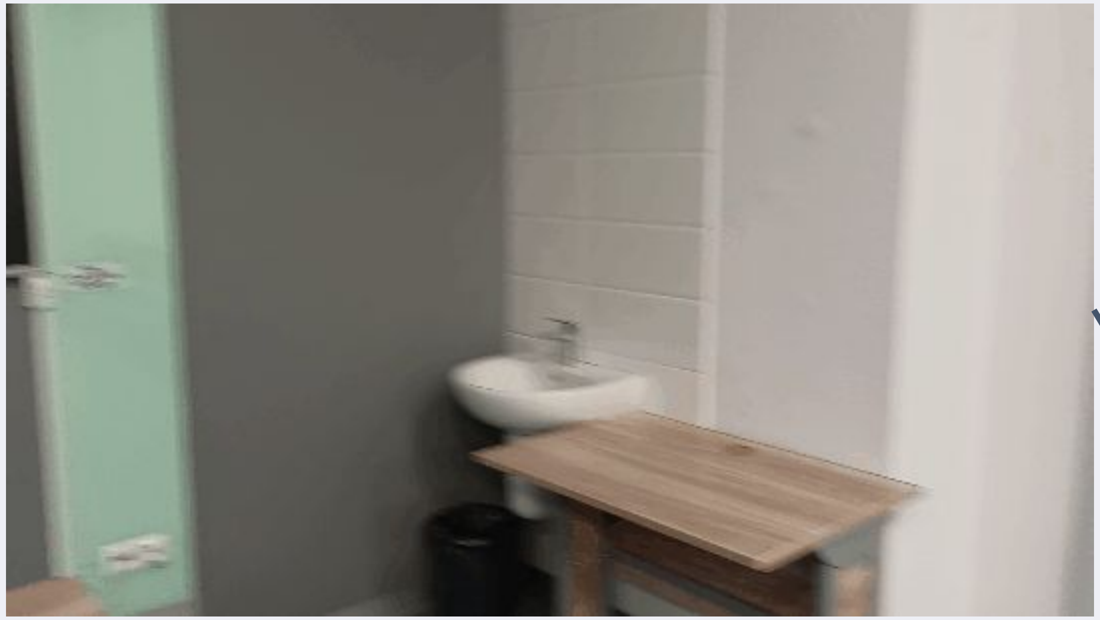
New world transition vector also can be assumed as world camera position

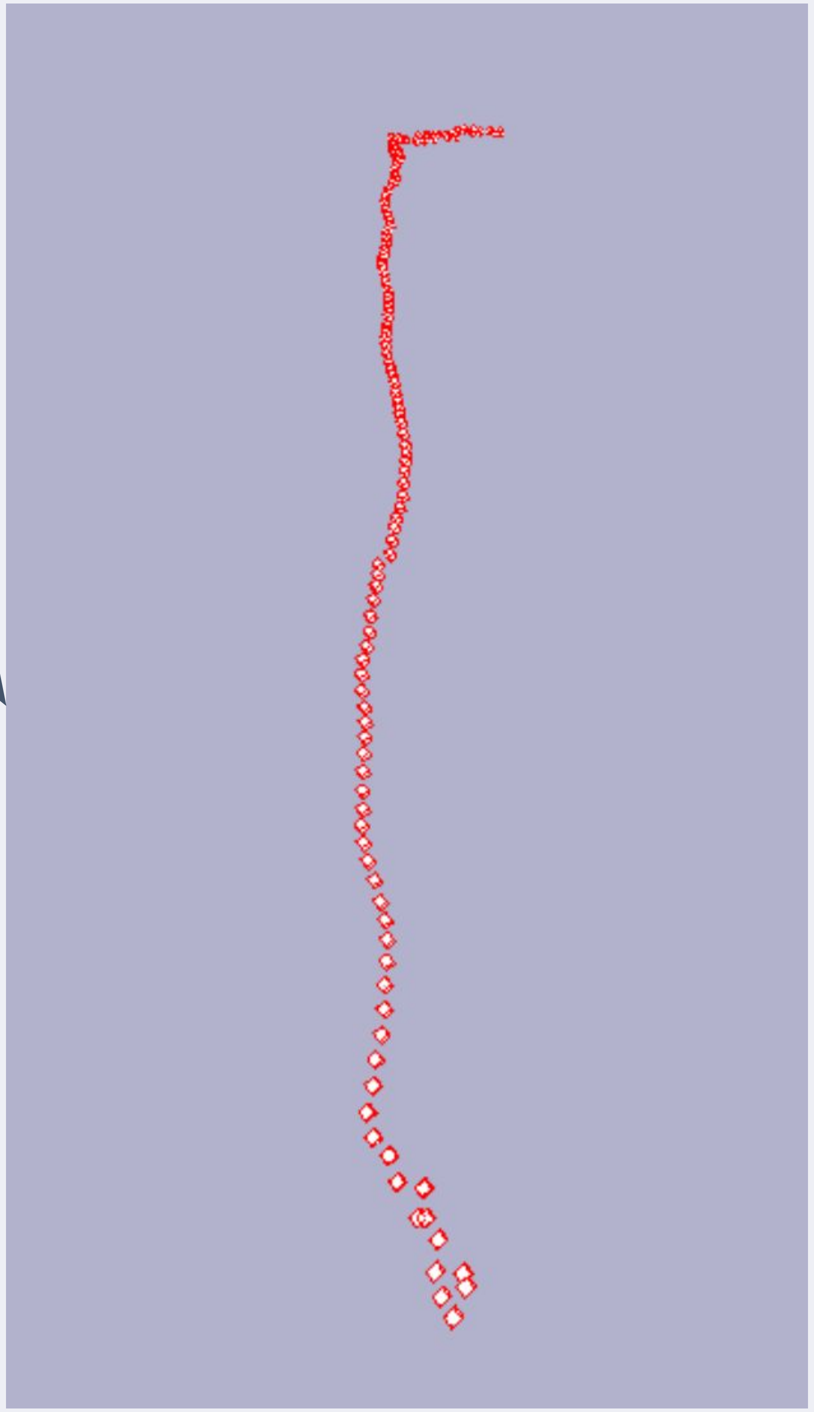
$$P = [R|t^T]$$

Projection matrix is a simple combination of rotation and transition which can be more convenient in some cases

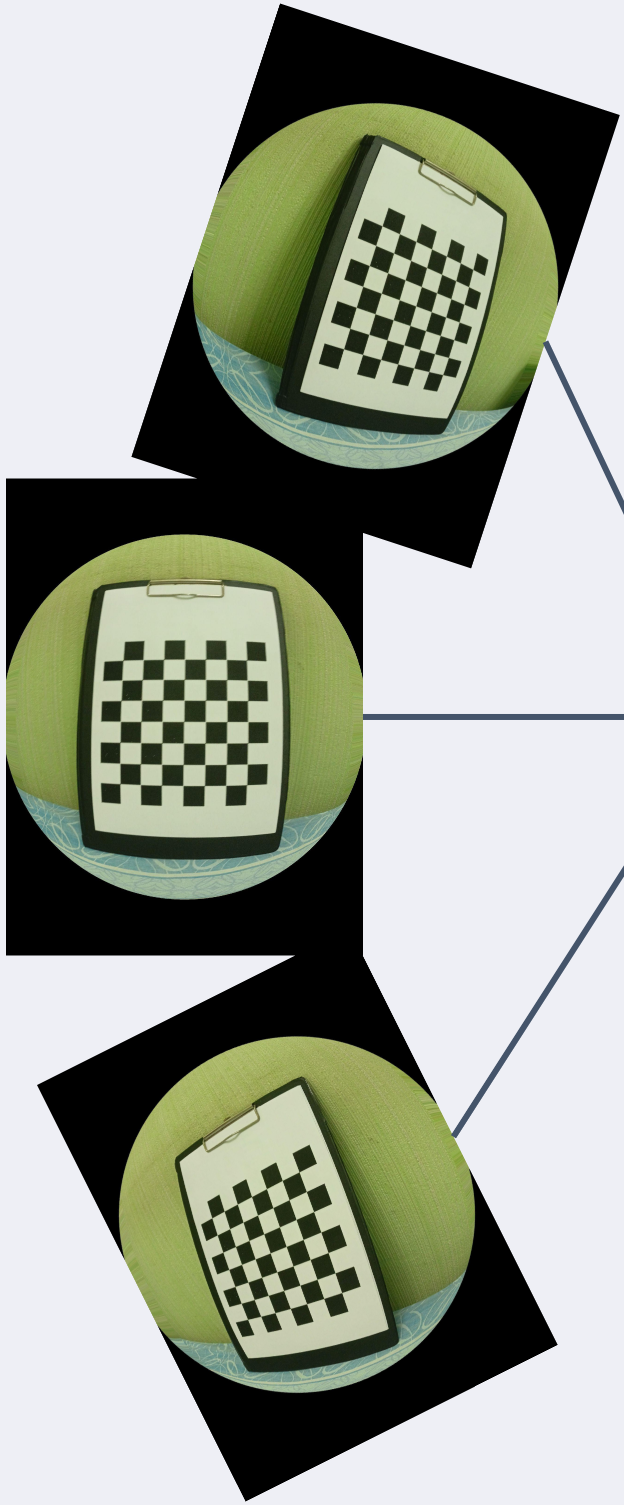
How it looks



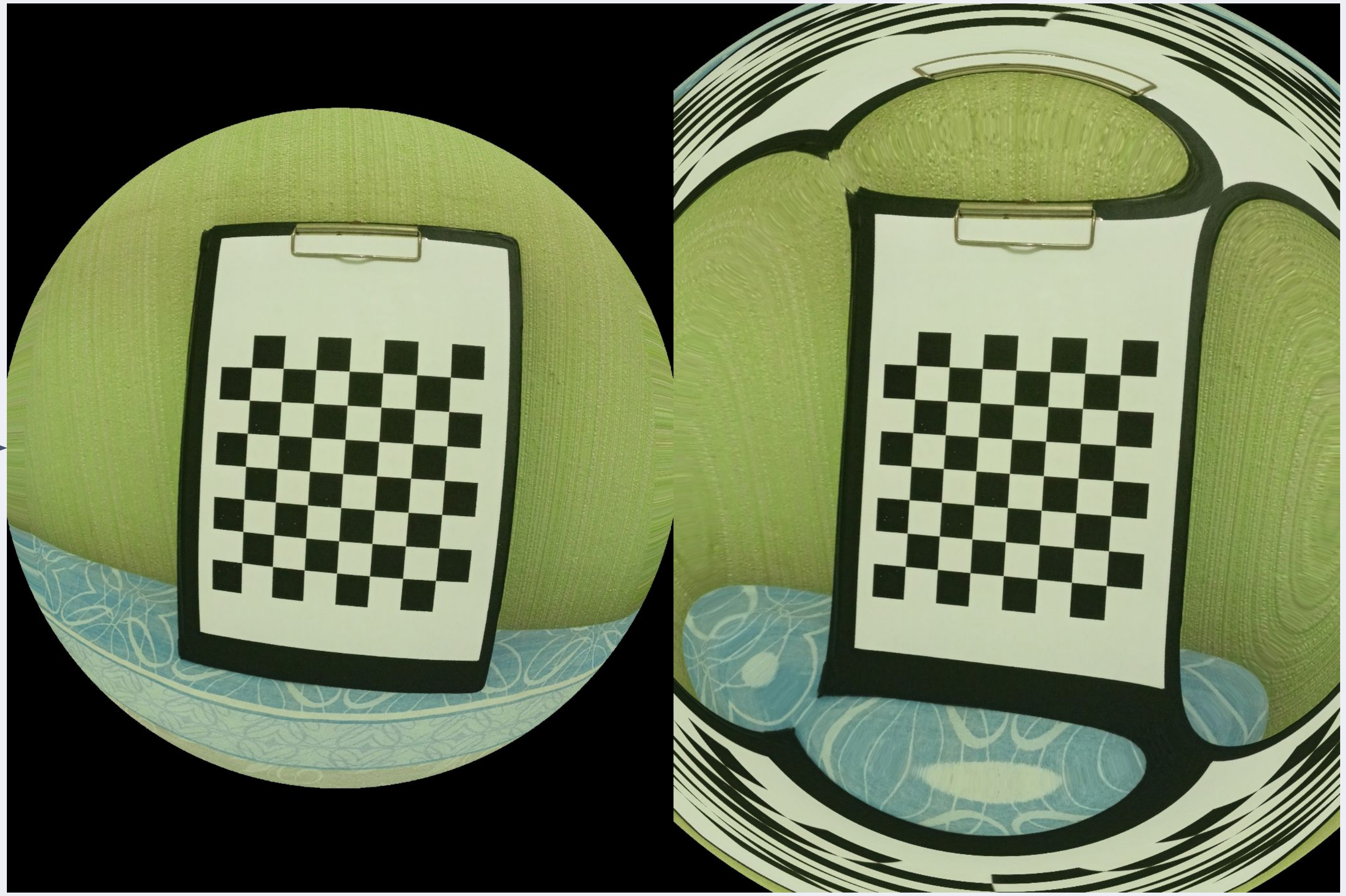




Calibration matrix



$$\begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

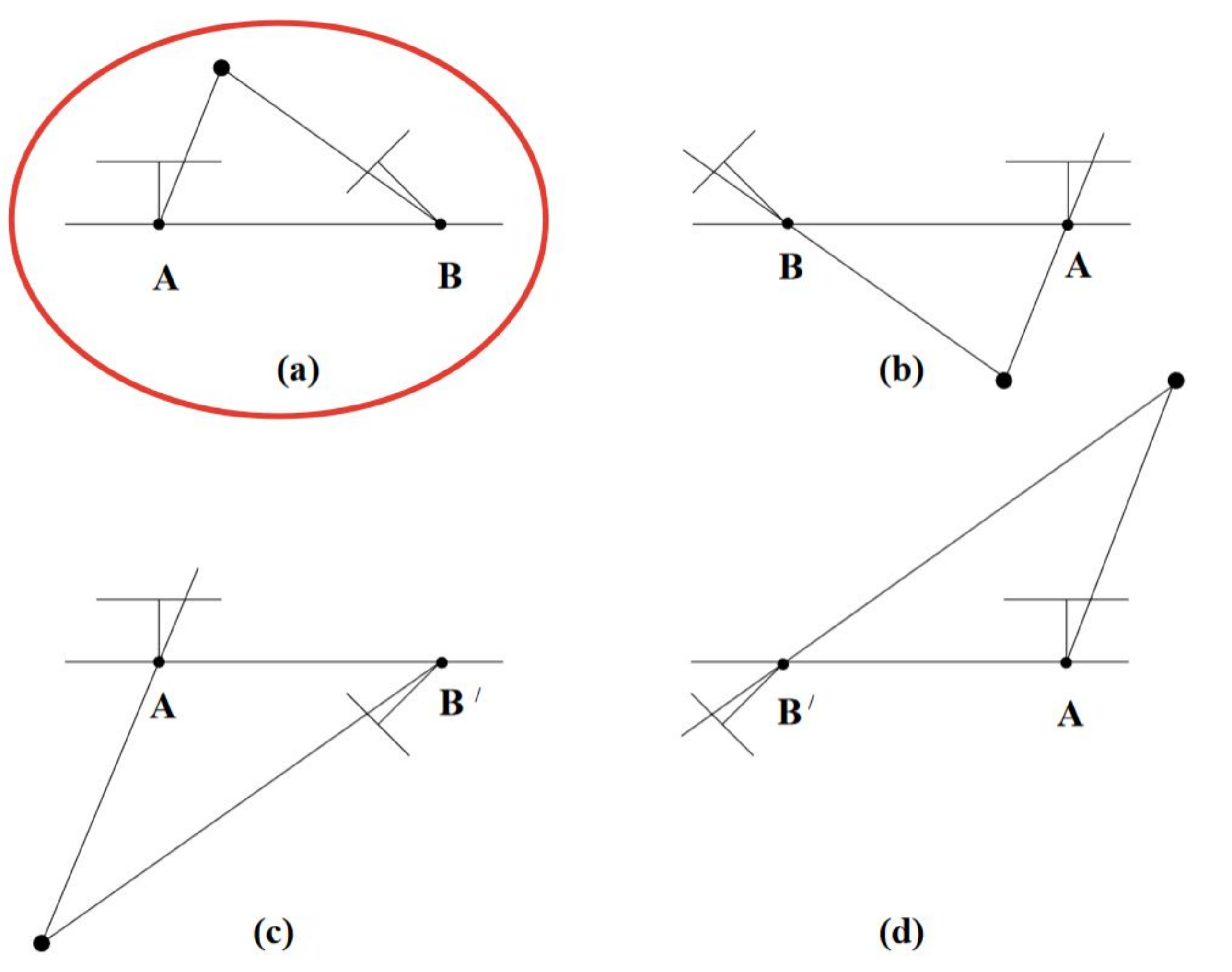


How to find rotation and transition?

$$[p_{2i}; 1]^T K^{-T} E K^{-1} [p_{1i}; 1] = 0$$

Essential matrix decomposition

$$[R_1, t], [R_1, -t], [R_2, t], [R_2, -t].$$



Projections have same logic for 3D

Triangulation

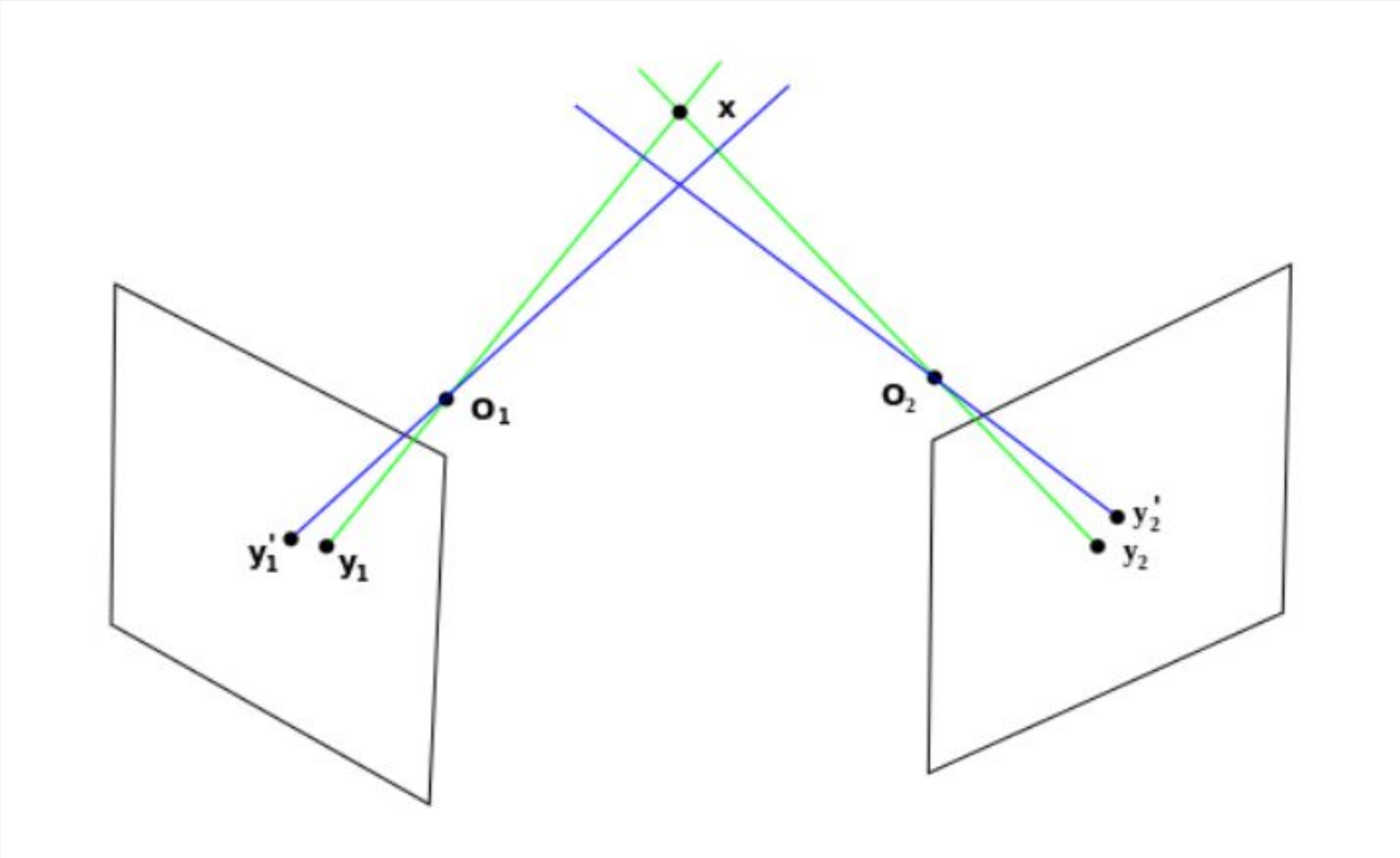
2D point Y1
(extracted point)

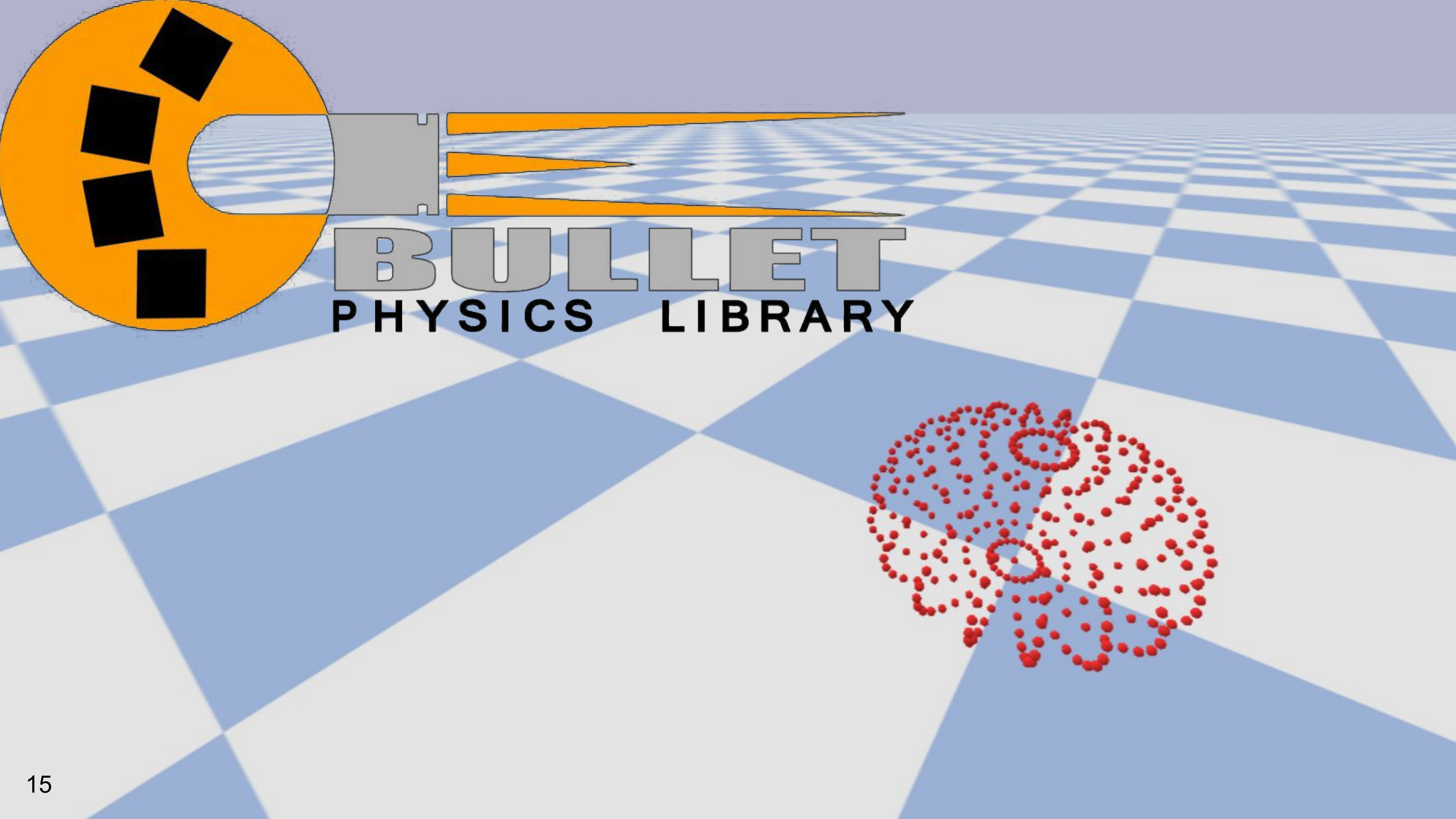
2D point Y2
(tracked point)

Projection
matrices

Complex linear decomposition

3D point X




BULLET
PHYSICS LIBRARY

Next term’s plans

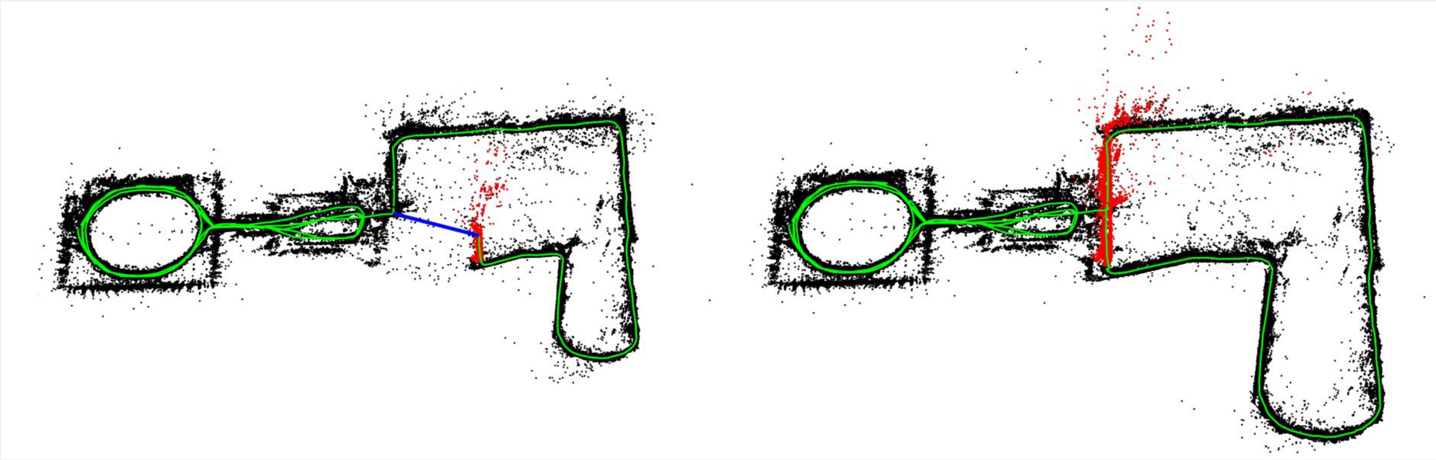
1 **Improve accuracy**

We hope to find more accurate and suitable algorithm of triangulation

2 **Optimizations**

For example, multithreading for main program and visualizer

3 **Detection of cycles**



Thanks for attention!

Task board

Github