Announcements

Final project upcoming deadlines:

- **Nov 13,** a short video of your progress.
  Submit via Compass, 10% of the final project total.

- **Dec 16,** 7-11pm in Siebel 4240.
  Final project presentations and Open House for press!

VR Lab Closed on Nov 16th ! (Pilots are using it)

Position Tracking: Visibility Methods

FEATURES:

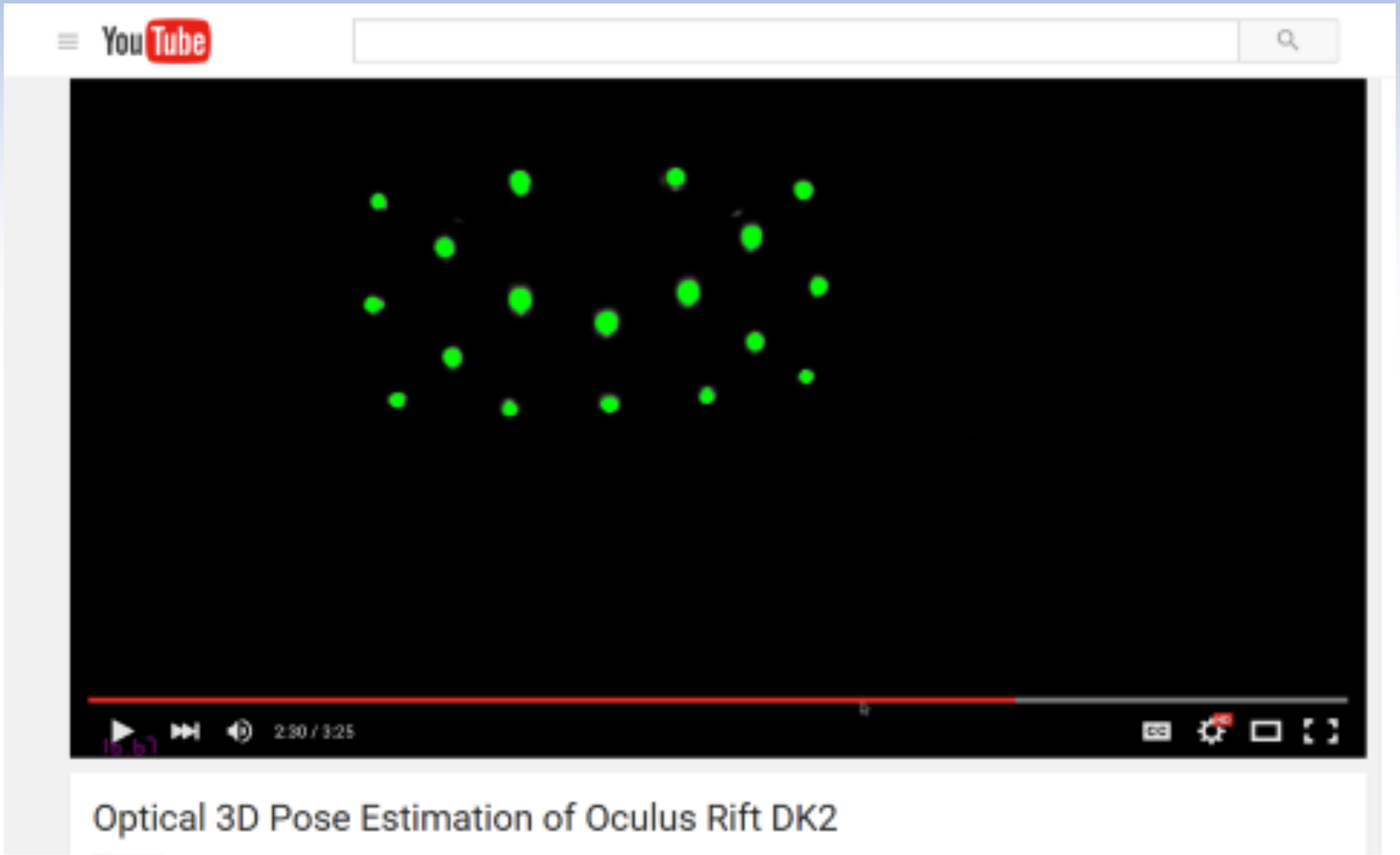
1) Natural

- Hard computer vision
- Extract and maintain from natural scenes
- Remove moving objects
- Reliability low

2) Artificial

- Trivial computer vision (blob detection)
- QR tags, retro reflective markers, LEDs, laser projections
- Can stay in IR spectrum (invisible to humans)

Position Tracking: Visibility Methods



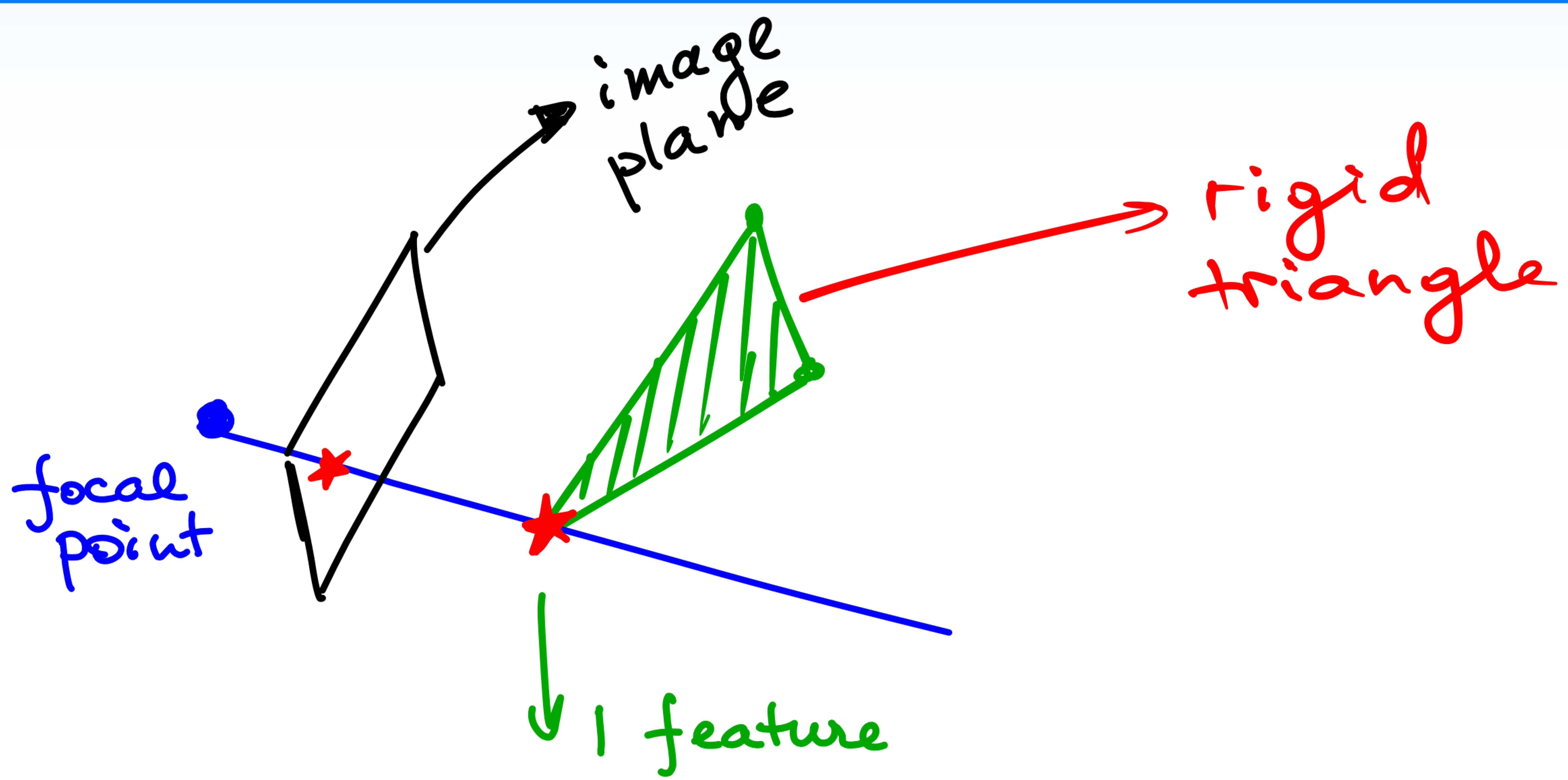
https://www.youtube.com/watch?v=X4G6_zt1qKY

Position Tracking: Blob Detection

PnP Problem:

Determine rigid body ____________________ from _________________________ features on a rigid body.

P1P Problem:



DOF analysis:

- Start with 6 DOFs (rigid body)
- Each feature subtracts 2 DOFs

DOFs left:

DOFs for rotations +

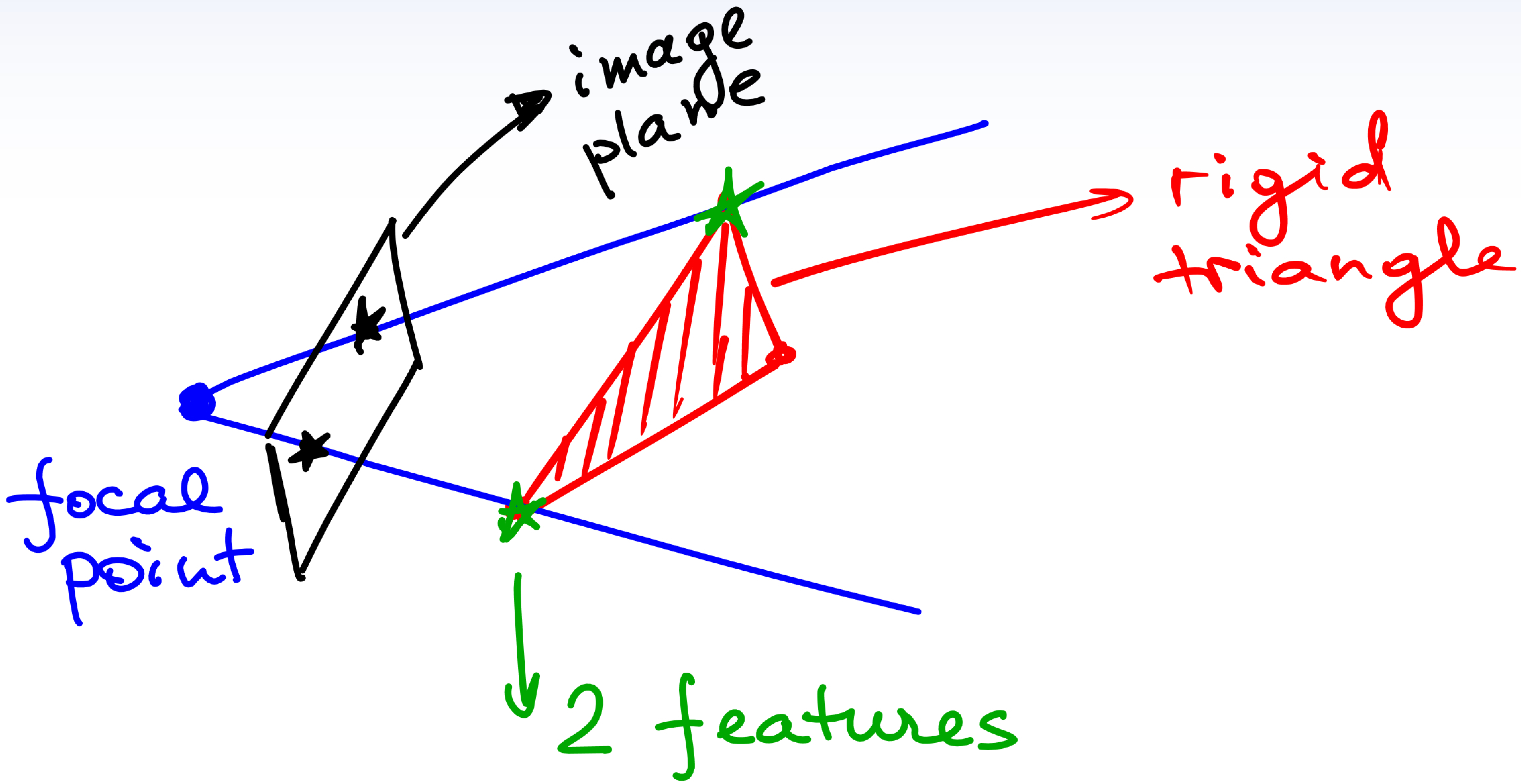
DOF for sliding along perspective line

Position Tracking: Blob Detection

P2P Problem:

identified

Determine position and orientation of triangle from 2 features in image.



DOFs left: DOF for sliding +

DOF for rotation around edge

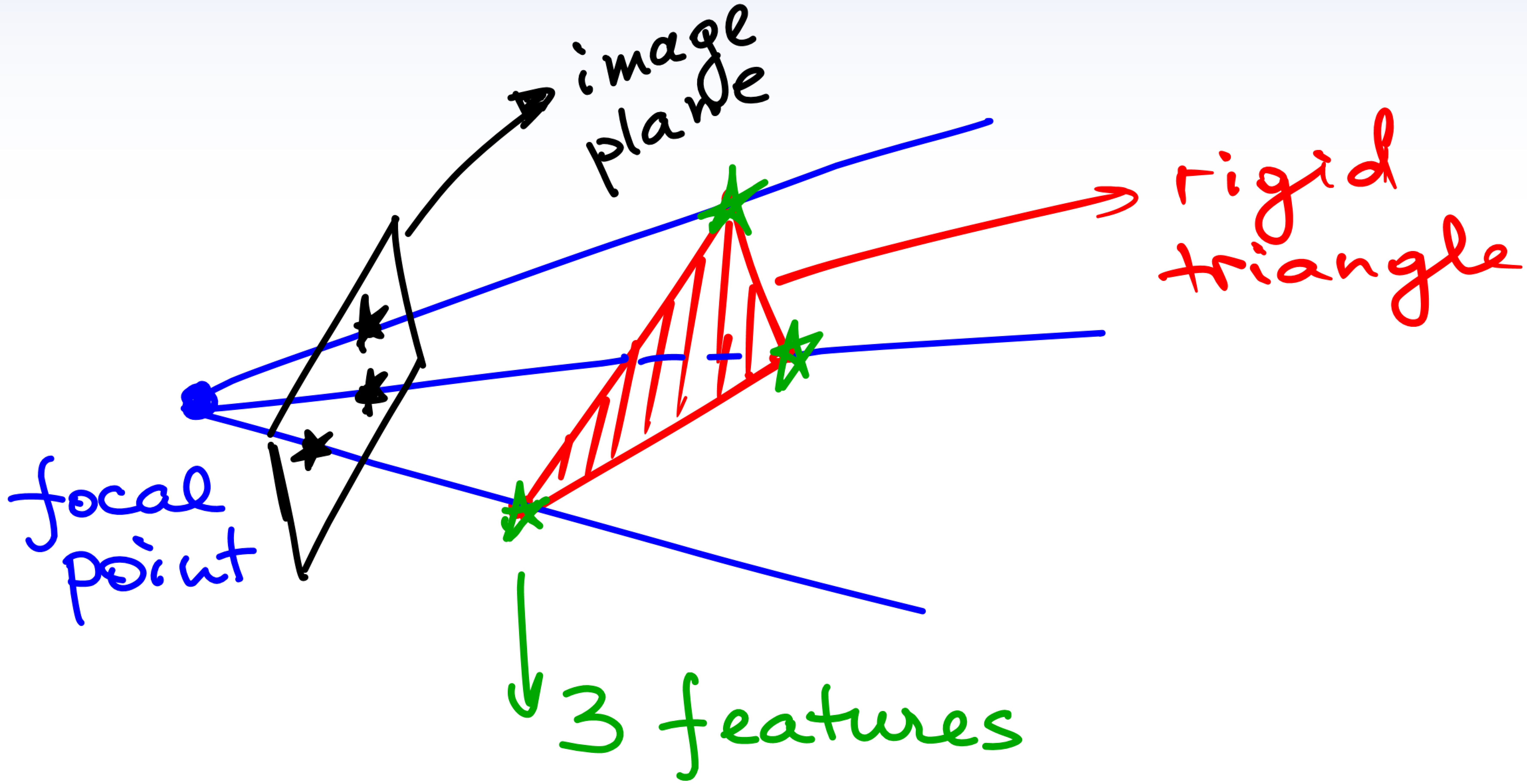
Position Tracking: Blob Detection

P3P Problem:

identified

Determine position and orientation of triangle from 3 features in image.

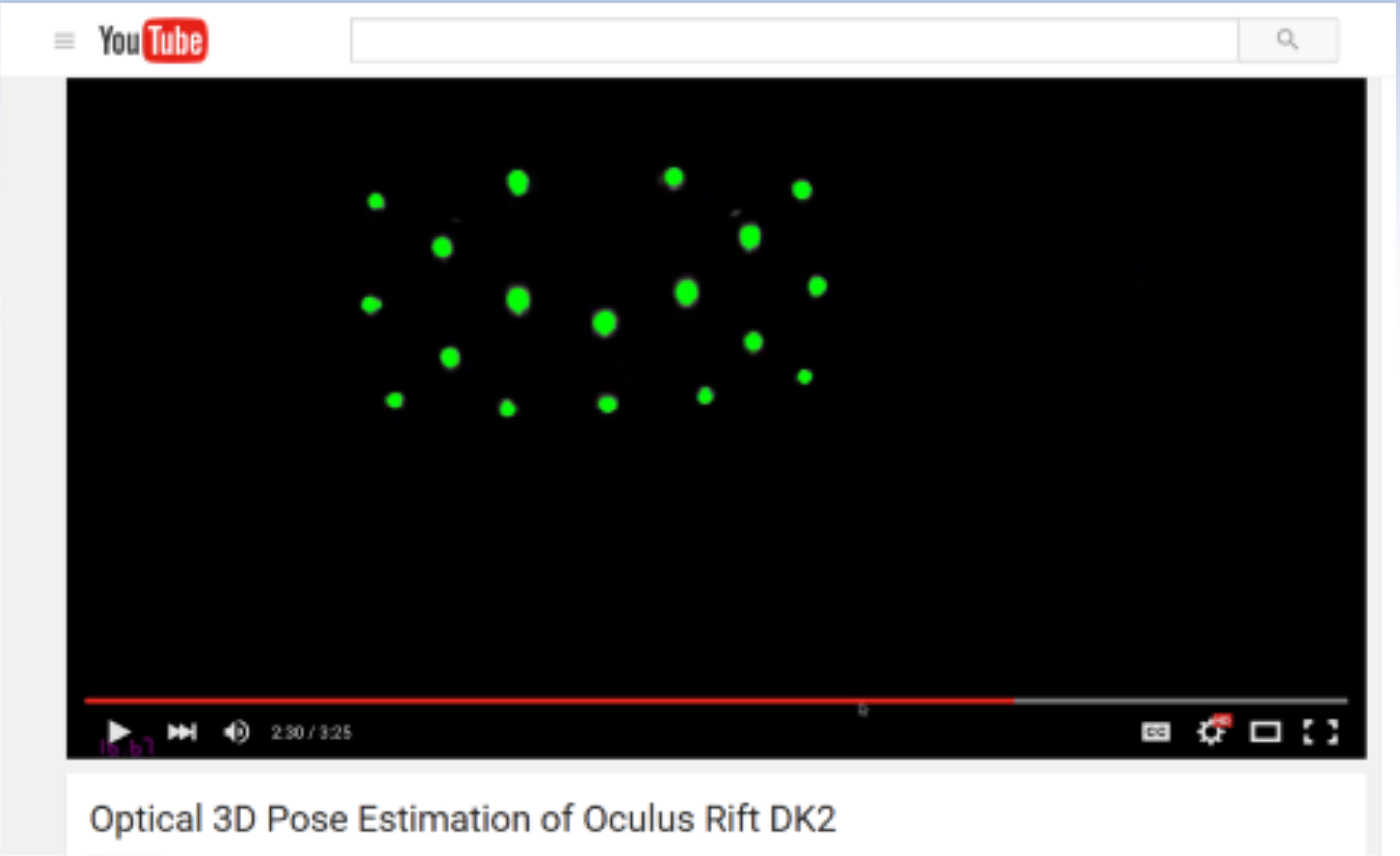
DOFs left



Solution: A system of polynomial equations leads to 8 solutions (but only 4 in front of the camera). The beginnings of computational real algebraic geometry.

P4P, P5P, ... PnP

Position Tracking: Visibility Methods



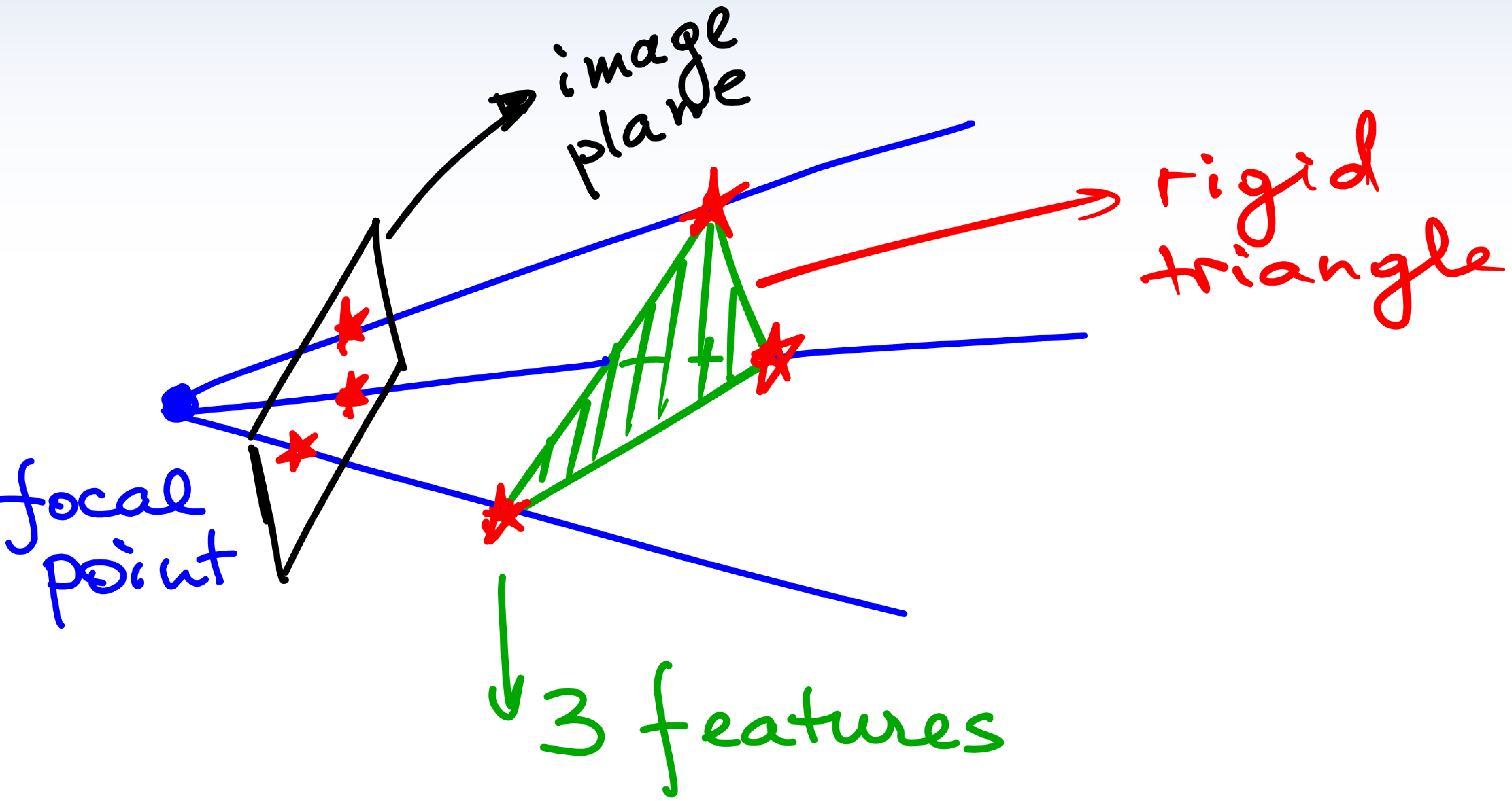
https://www.youtube.com/watch?v=X4G6_zt1qKY

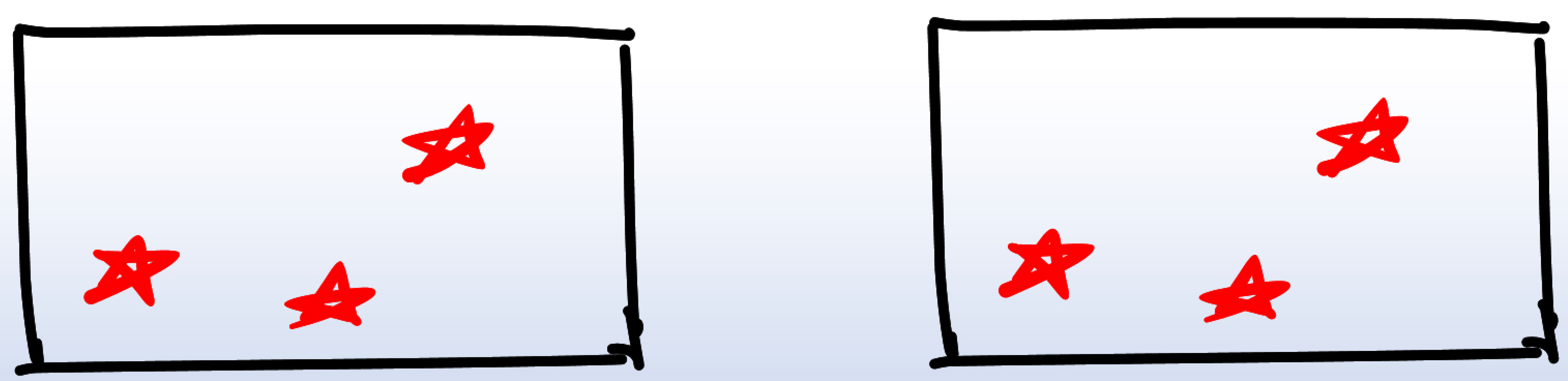
Position Tracking: Incremental Blob Detection

Incremental PnP Problem:

Faster than PnP!

Determine rigid body tranformation from identified features in image, given current estimate.





Position Tracking: Sensor Fusion

Sensors:

- Gyroscope
- Accelerometer
- Magnetometer
- Camera-LEDs

$$\hat{x}_{k+1} = \hat{x}_k +$$

$$\hat{v}_{k+1} = \hat{v}_k +$$

Position Tracking: Sensor Fusion

Sensors:
- Gyroscope
- Accelerometer
- Magnetometer
- Camera-LEDs

$$\hat{x}_{k+1} = \hat{x}_k + \hat{v}_k \Delta t + \frac{1}{2}\hat{a}\Delta t^2 + \alpha \Delta x_k$$

$$\hat{v}_{k+1} = \hat{v}_k + \hat{a}\Delta t + \beta \Delta v_k$$

$$\Delta x_k = \hat{x}_{vis} - \hat{x}_k$$

$$\Delta v_k = \hat{v}_{vis} - \hat{v}_k$$

How to pick $\alpha, \beta \in (0, 1)$?

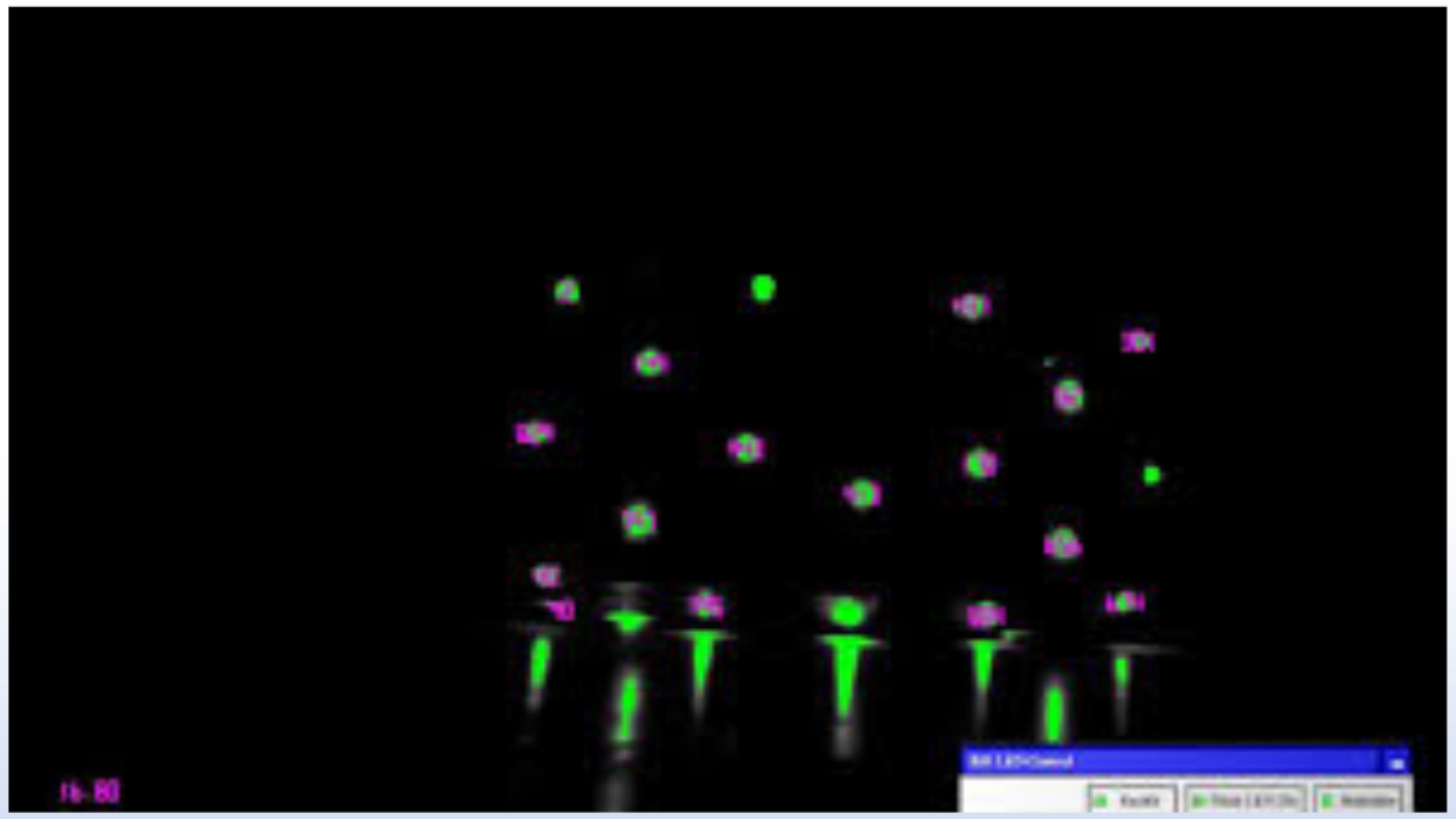
1) Empirically - complementary filter
2) Kalman filter - general purpose black box

σ_w = process noise varience

σ_v = measurement noise varience

3) Perceptually tuned gains

Position Tracking: Sensor Fusion



Position Tracking: LIghthouse (Valve/HTC Valve)

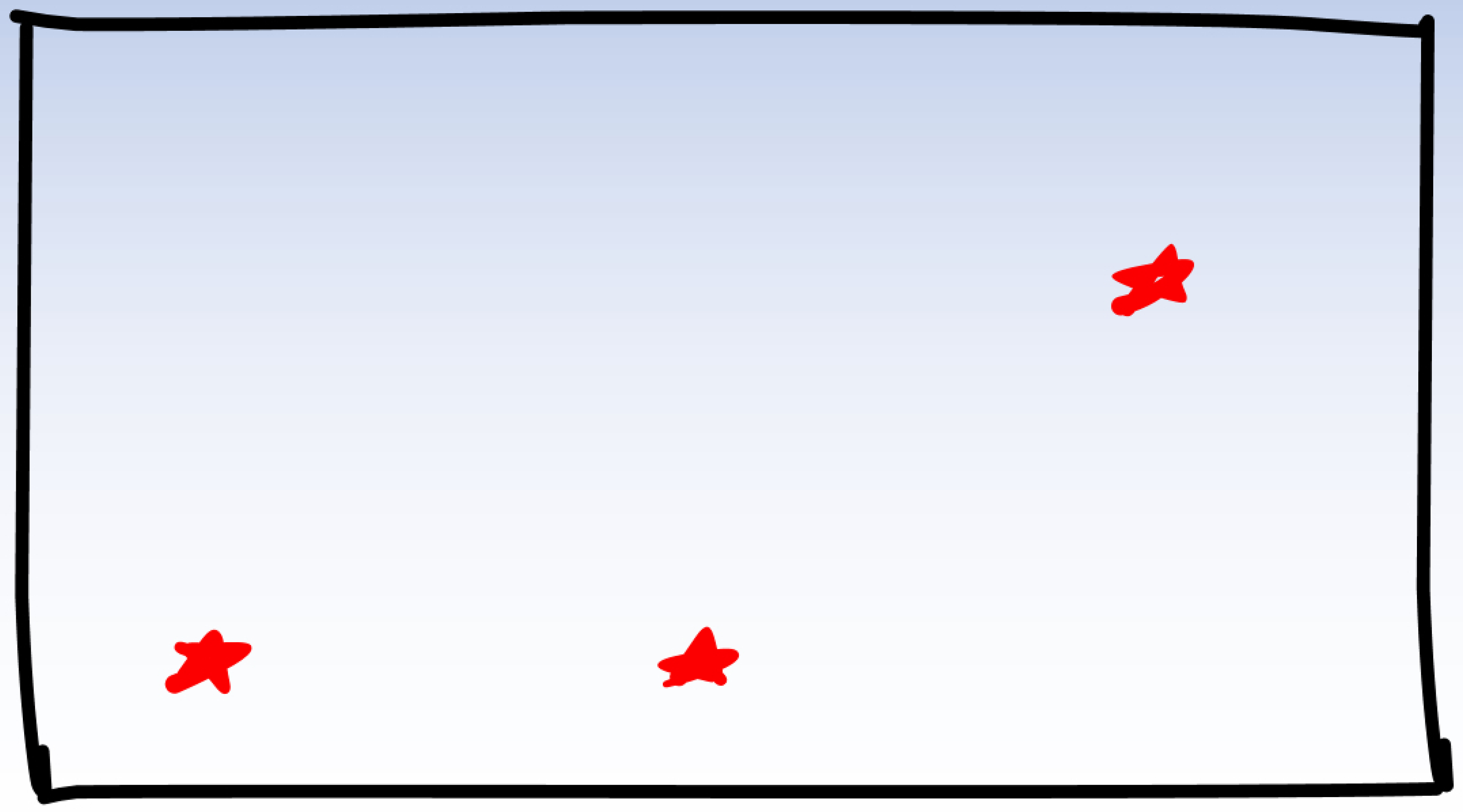


Image processing steps:

1)

2)

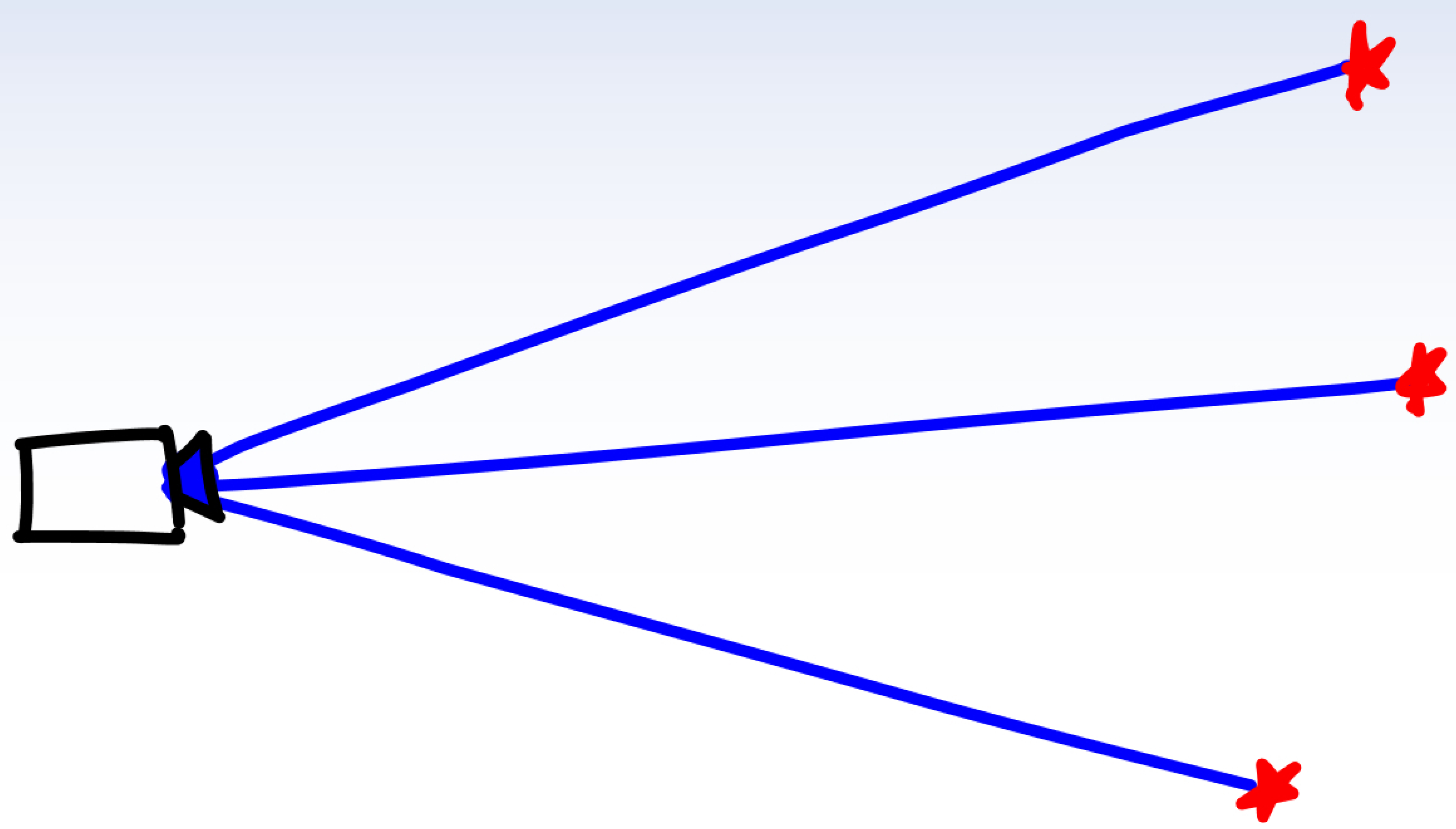
3)

4)

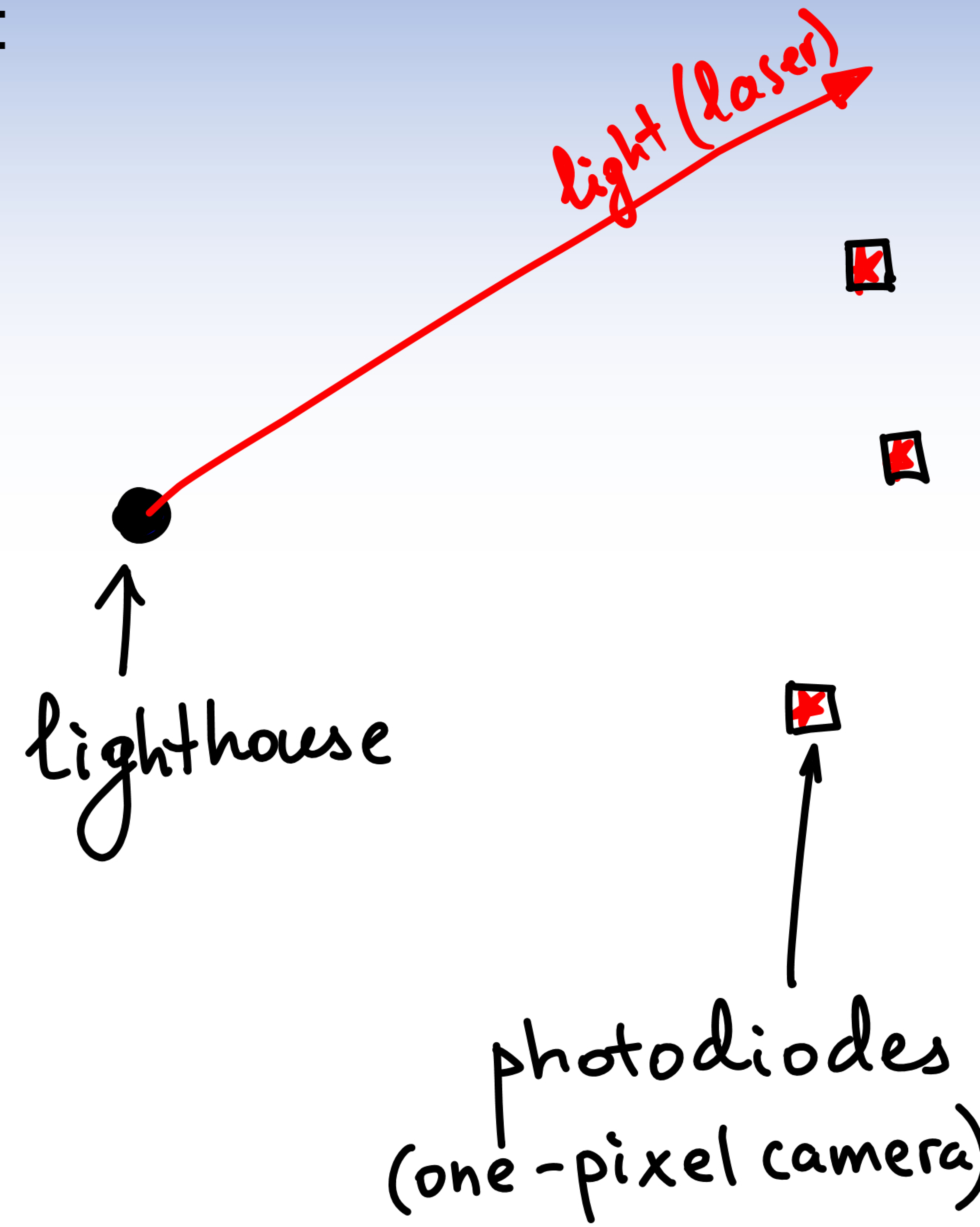
Why not bypass taking an image and image processing (scanning for features)?

Position Tracking: LIghthouse (Valve/HTC Valve)

Mathematically equivalent models in 2D:



vs.



Pros:

- Identification of features is

- Need to know rotation

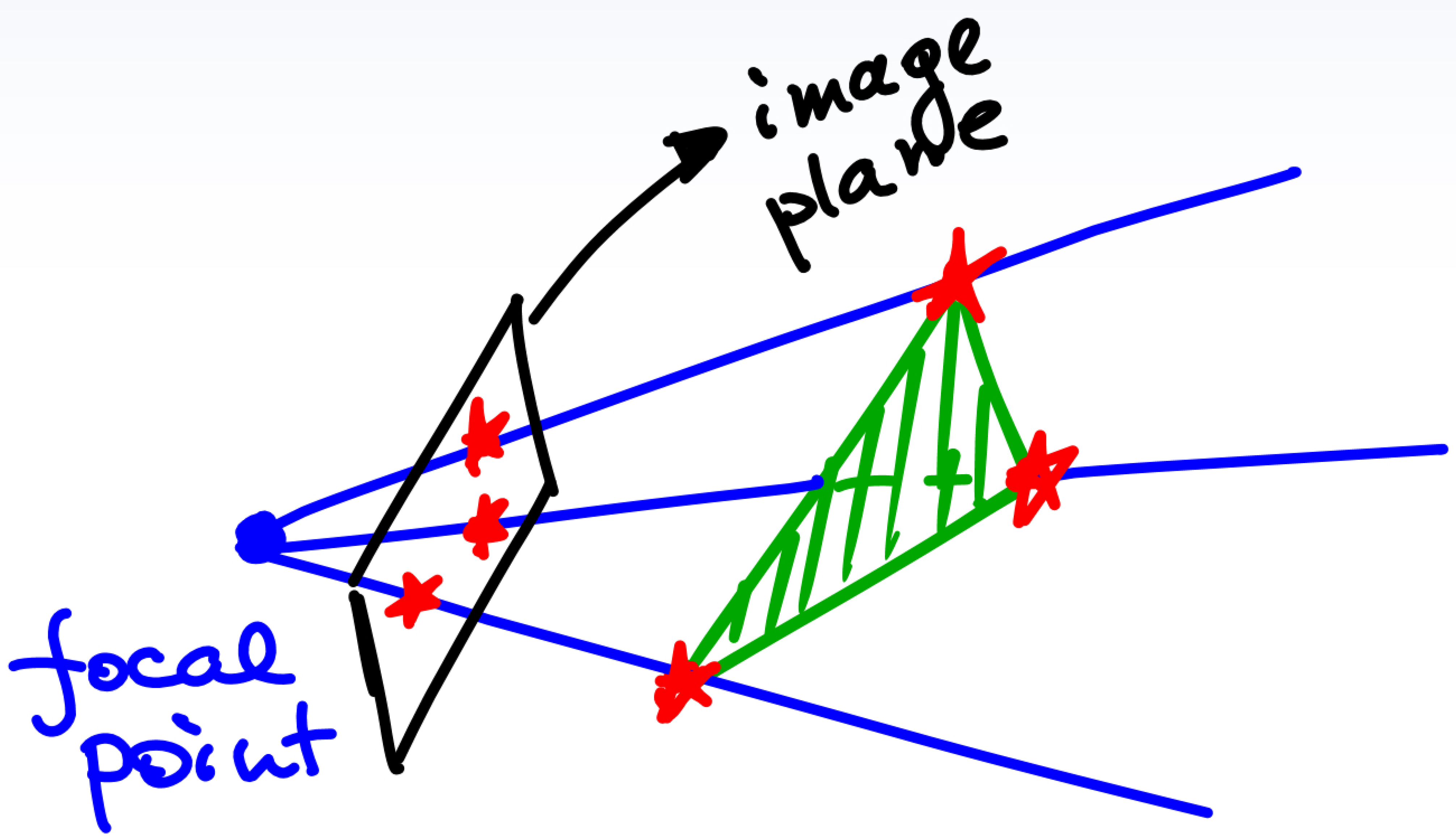
- No

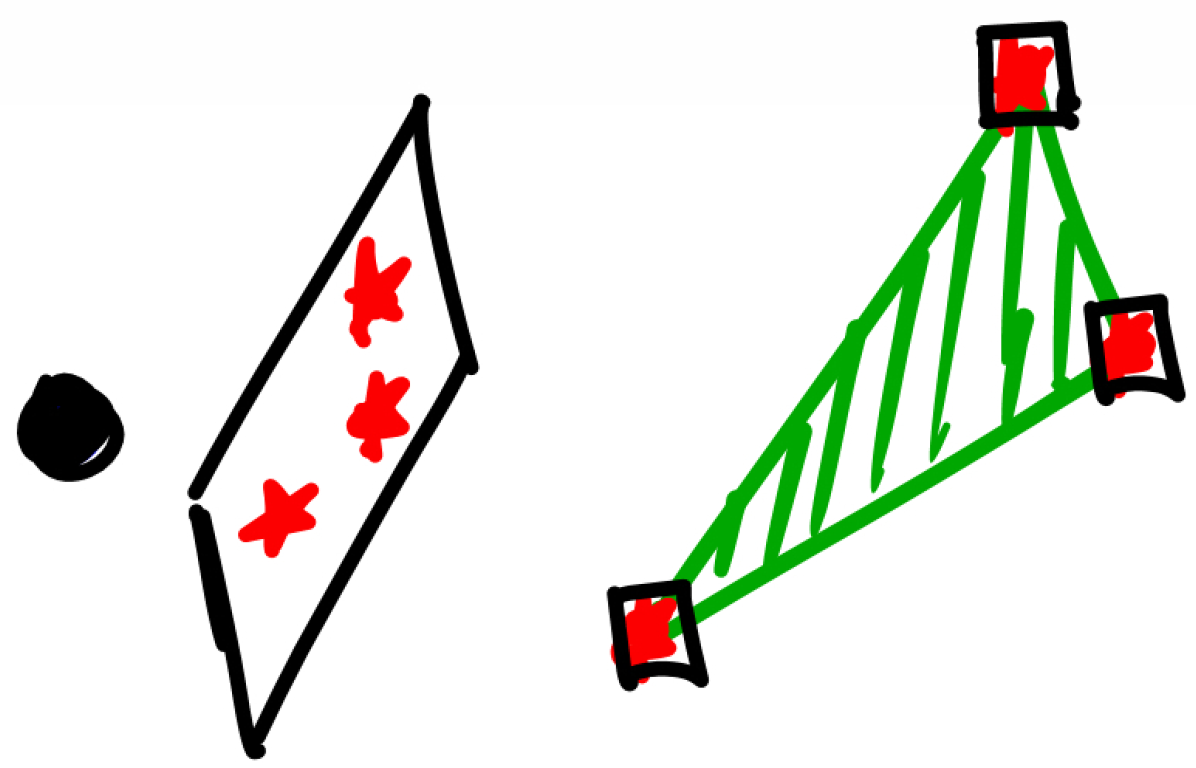
Cons:

- Does

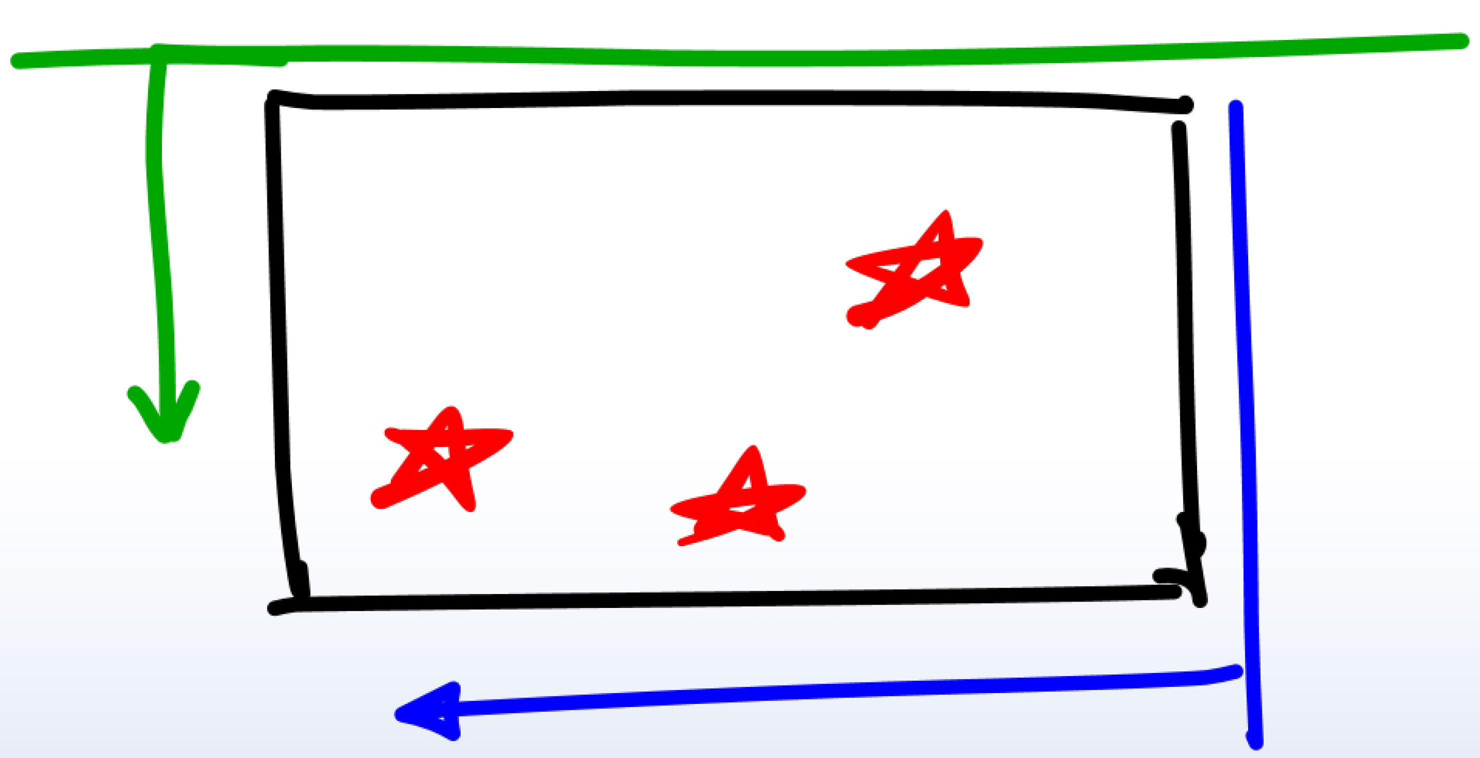
Position Tracking: LIghthouse (Valve/HTC Valve)

Mathematically equivalent models in 3D: need horizontal and vertical beams



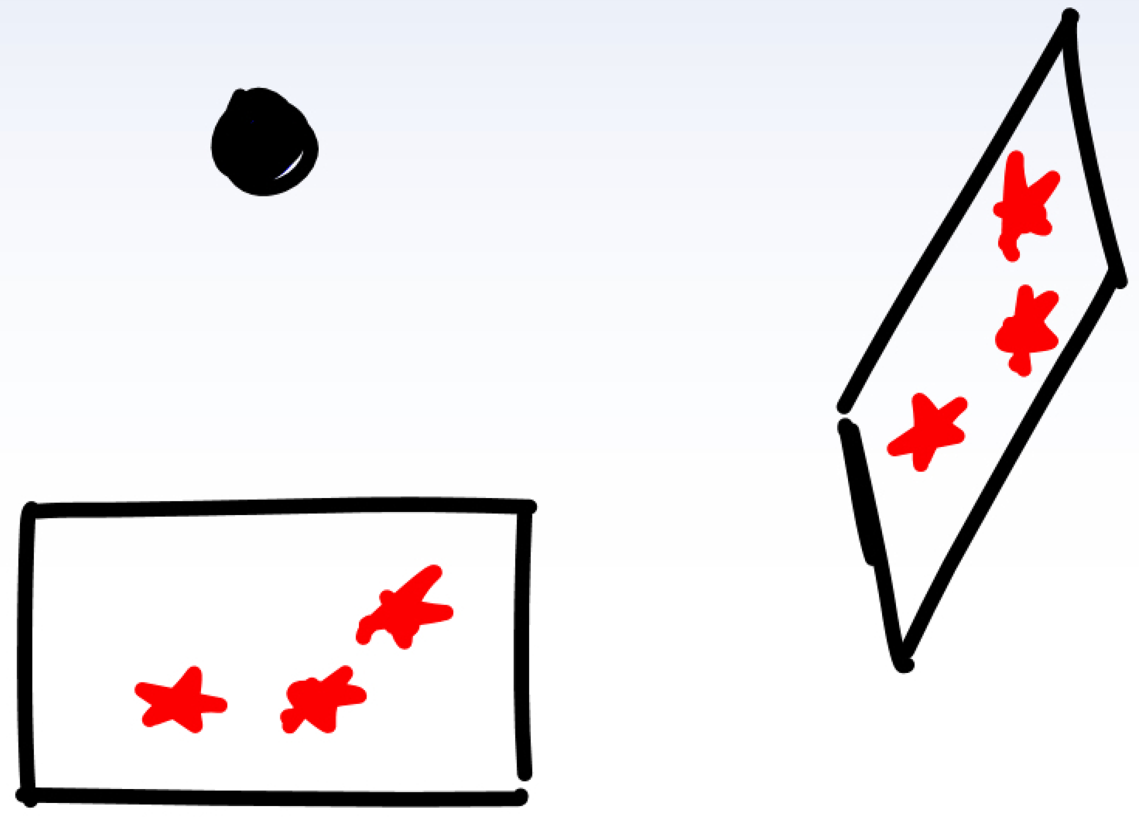


Problems:
- Maintaining timing to avoid angular drift
- Two beams hitting the same sensor at the same time



Position Tracking: LIghthouse (Valve/HTC Valve)

Correcting for angular drift due to timing error:



Solution:

- Flash a bright flash (in IR)

- Received by

- Reset/sync

Rendering

$$T = \overset{T_{dist}}{T_{vp}} \cdot \overset{T_L \text{ or } T_R}{T_{can}} \cdot T_{eye} \cdot T_{rB}$$