AHEAD

3D AR HUD

1. Traditional rendering

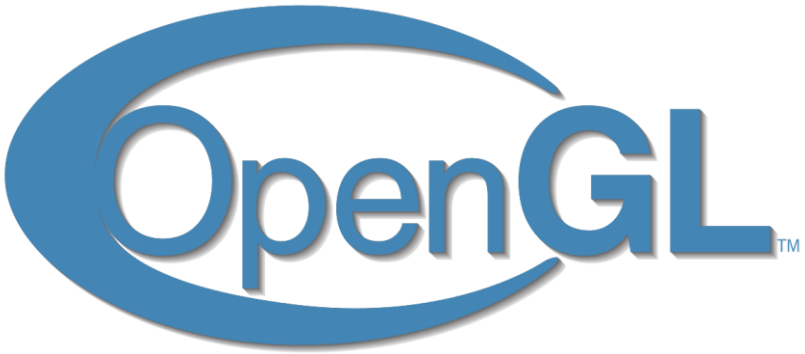
Rendering

- Rendering: 3D space -> 2D display

Source: Unity

Rendering

- Graphics Library for rendering
  - OpenGL: C/C++, GLSL(shader language)
  - DirectX : C/C++, HLSL(shader language)



Khronos Group

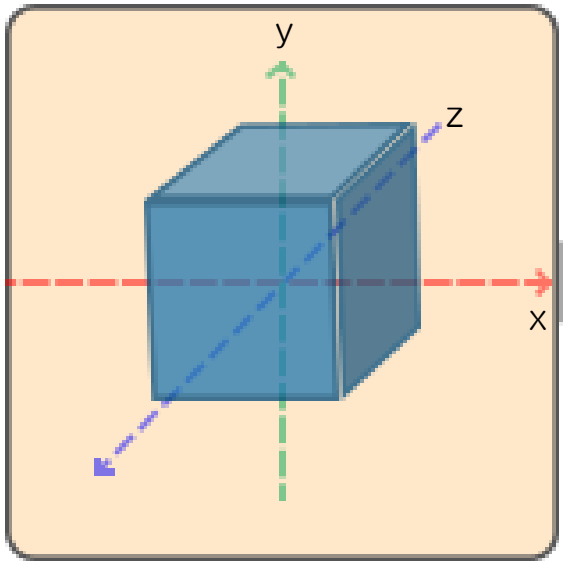
Microsoft

Source: wikipedia

Rendering Pipeline

- 1. Local space
  - Local coordinates are the coordinates of your object relative to its local origin.

Vertex Data (x,y,z)

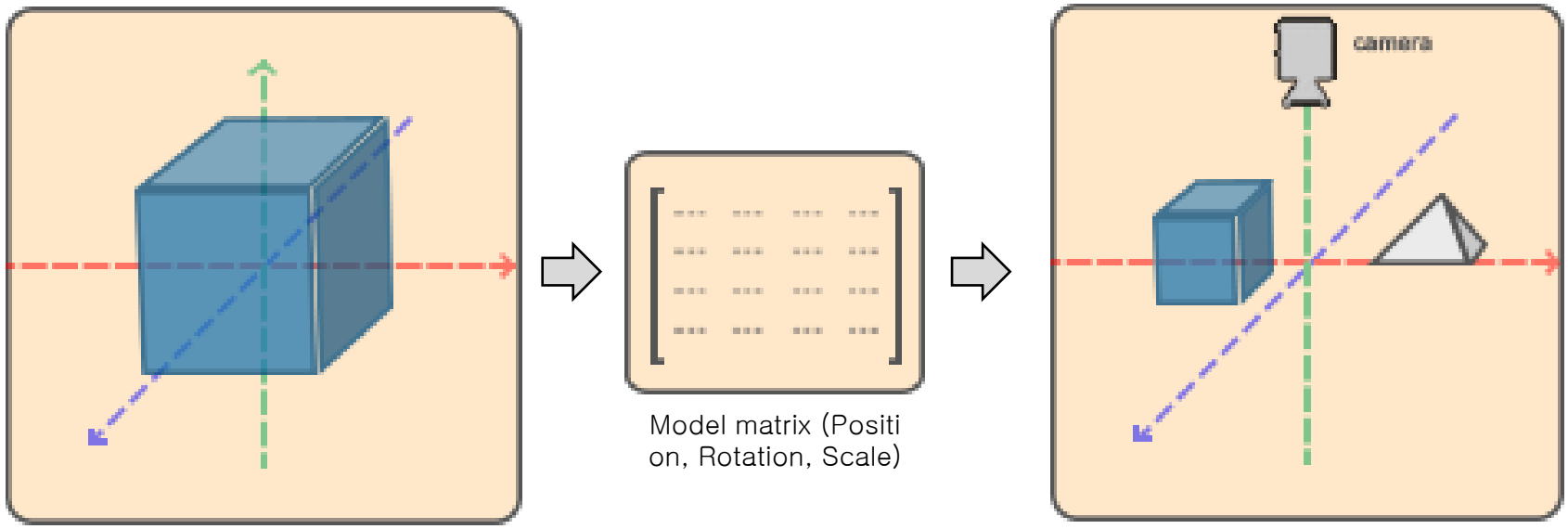


Local space

Source: learnopengl

Rendering Pipeline

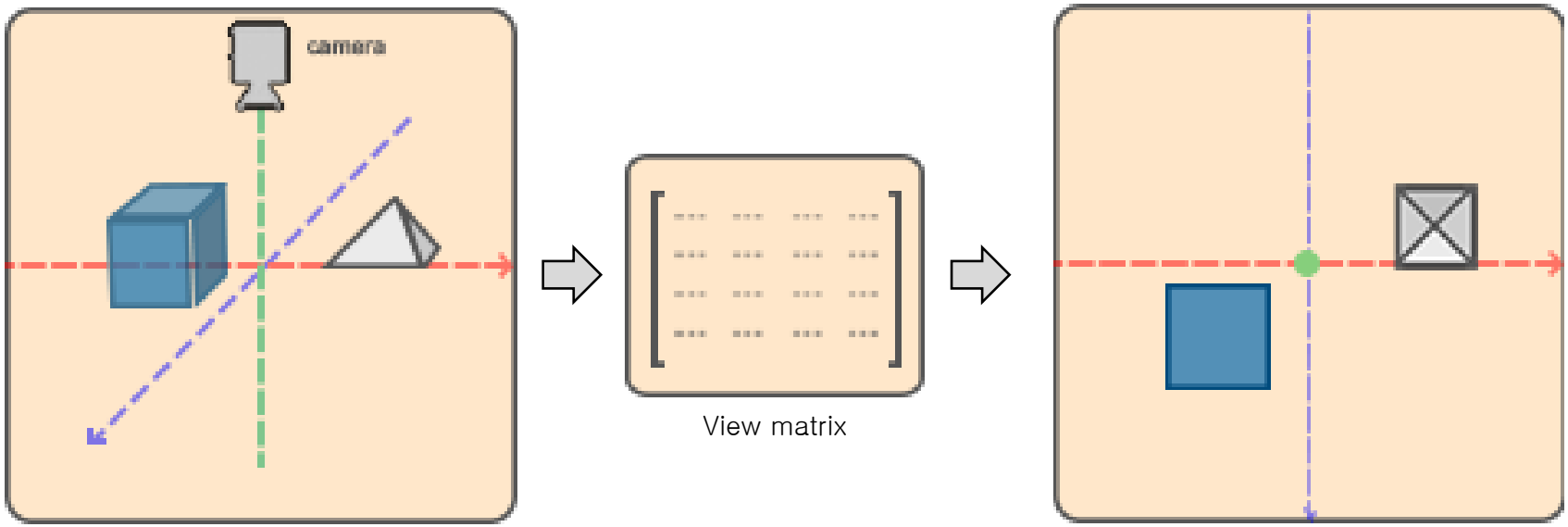
- 2. World space
  - These coordinates are relative to some global origin of the world.



Source: learnopengl

Rendering Pipeline

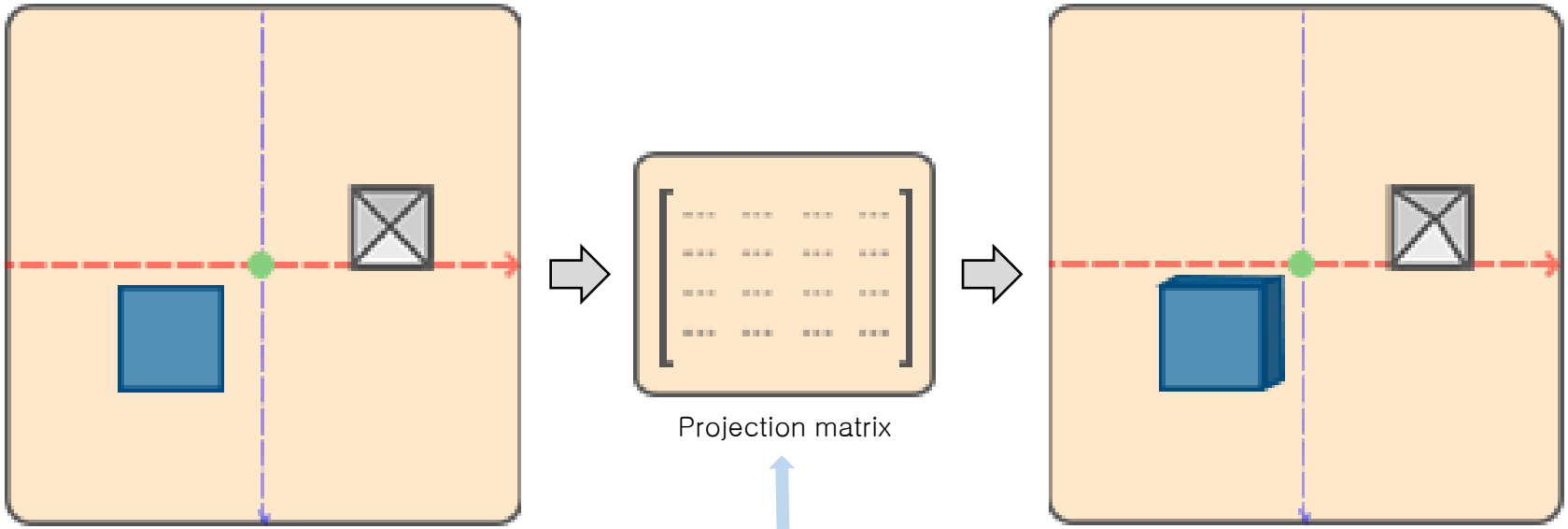
- 3. View space
  - Coordinate is as seen from the camera point of view.



Source: learnopengl

Rendering Pipeline

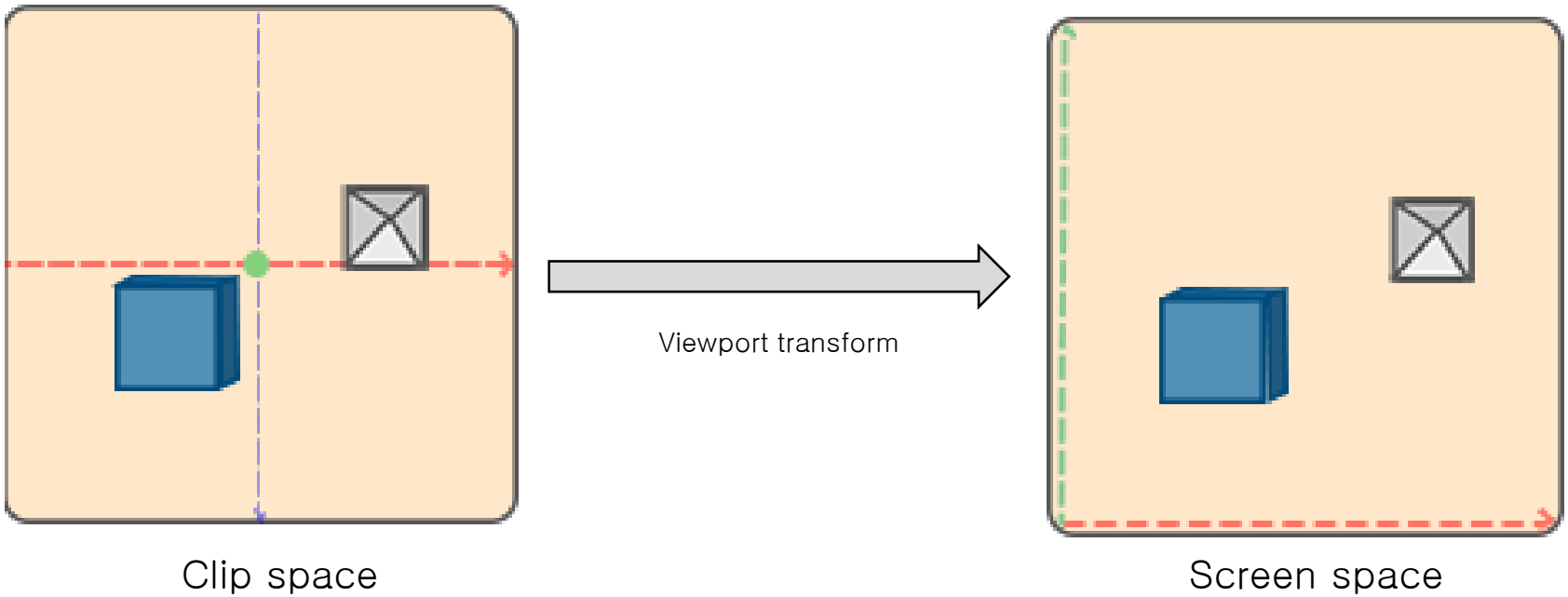
- 4. Clip space
  - Clip coordinates are processed to the -1.0 and 1.0 range.
    - (optional) perspective projection: Adding perspective.



Source: learnopengl

Rendering Pipeline

- 5. Screen space
  - Clip coordinate(-1.0~1.0) transform to screen coordinate(ex. 1980x1080).

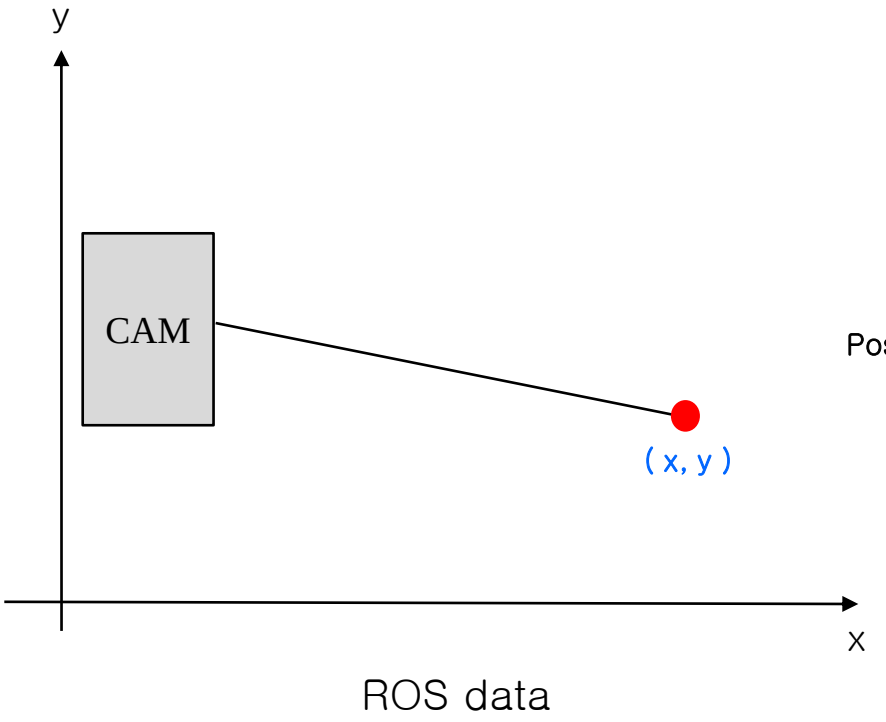


Source: learnopengl

2. AHEAD rendering

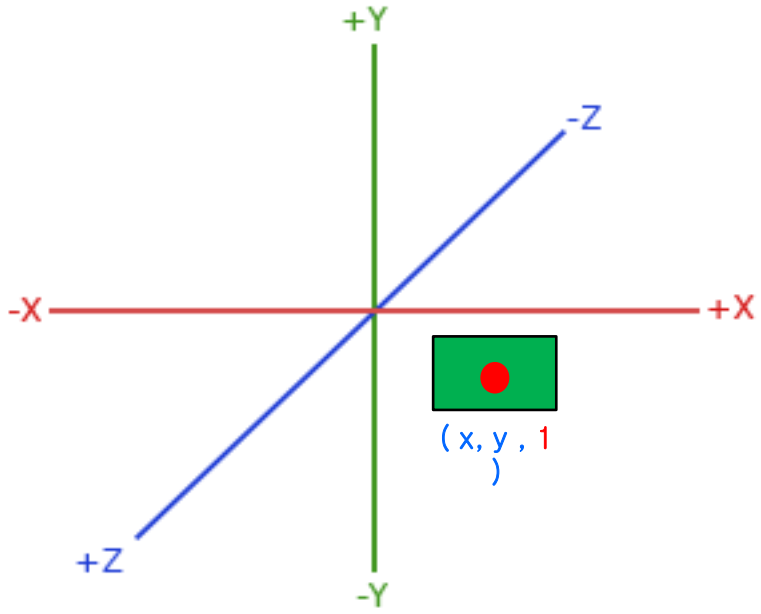
AHEAD rendering

- Vehicle Detection rendering process
  - Bird eye view:



ROS data

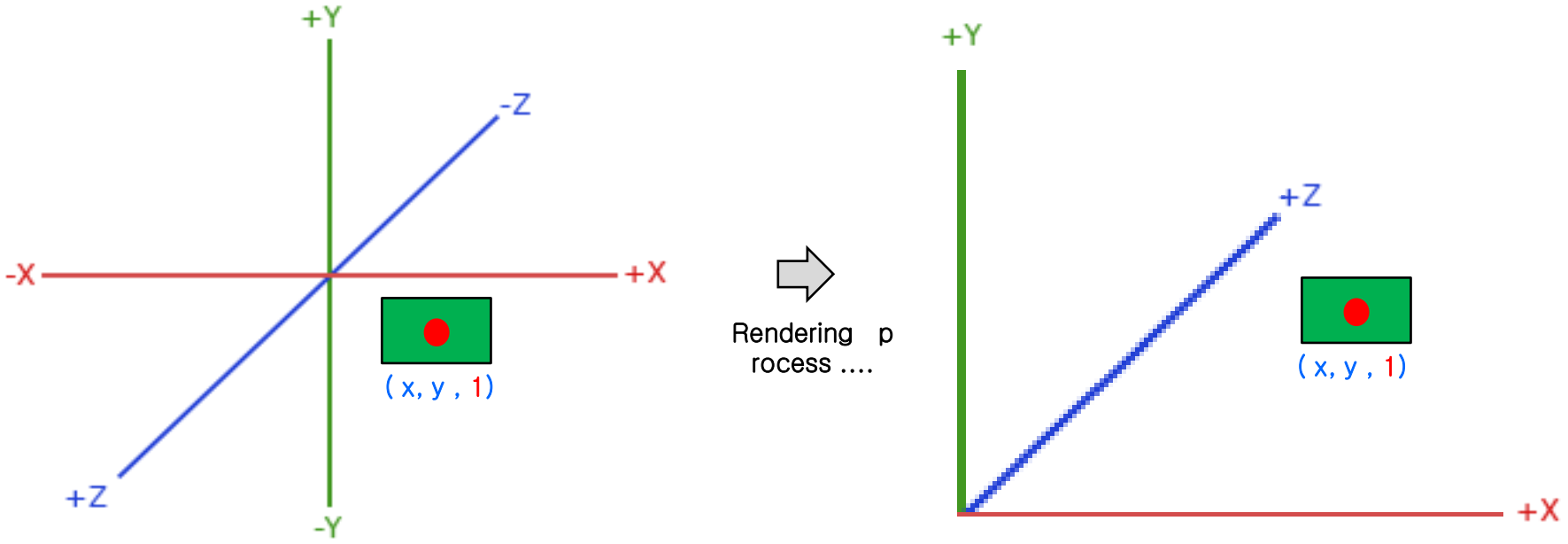
Position(x, y , 1)



World space

AHEAD rendering

- Vehicle Detection rendering process
  - Bird eye view:

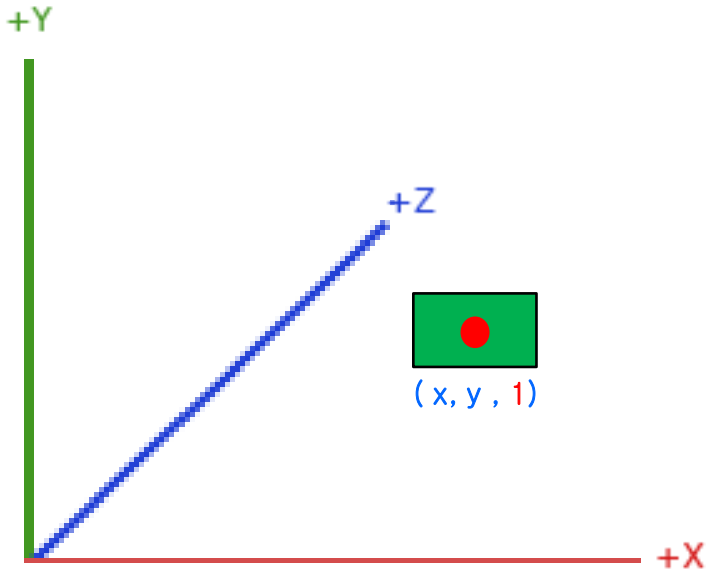


World space

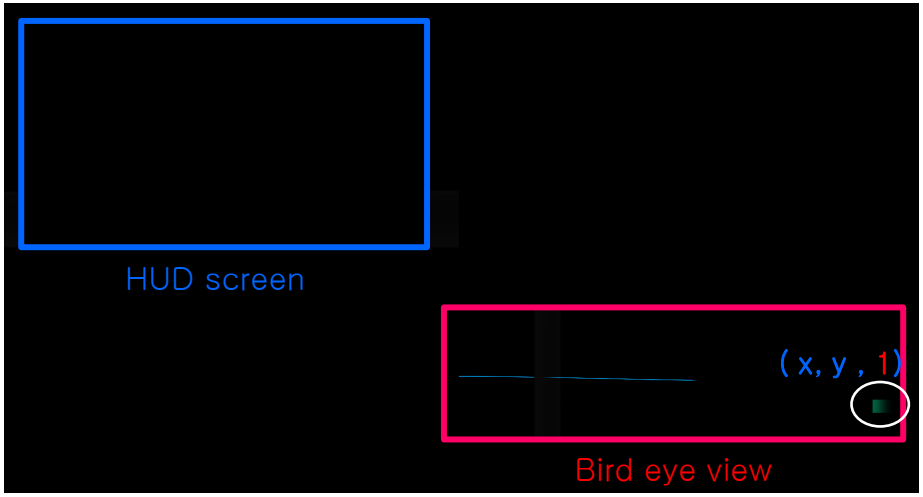
Screen space

AHEAD rendering

- Vehicle Detection rendering process
  - Bird eye view:



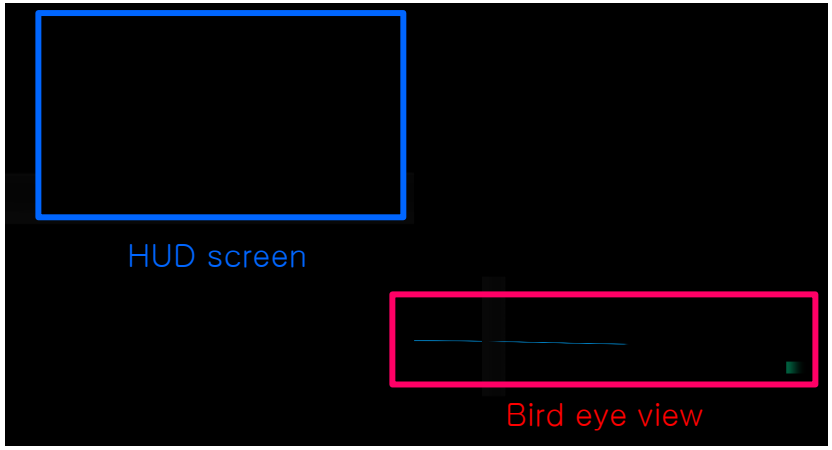
Screen space

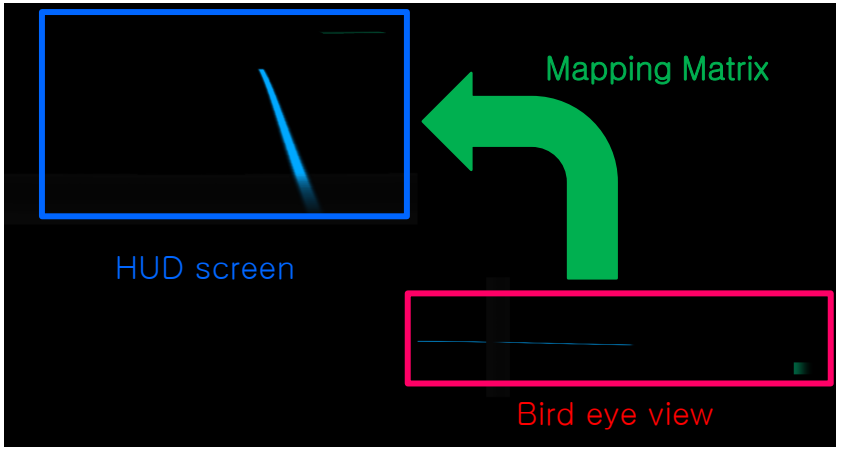


Bird eye view complete!

AHEAD rendering

- Vehicle Detection rendering process
  - HUD screen: Move 2D screen coordinates(bird eye view screen) to HUD screen
    .



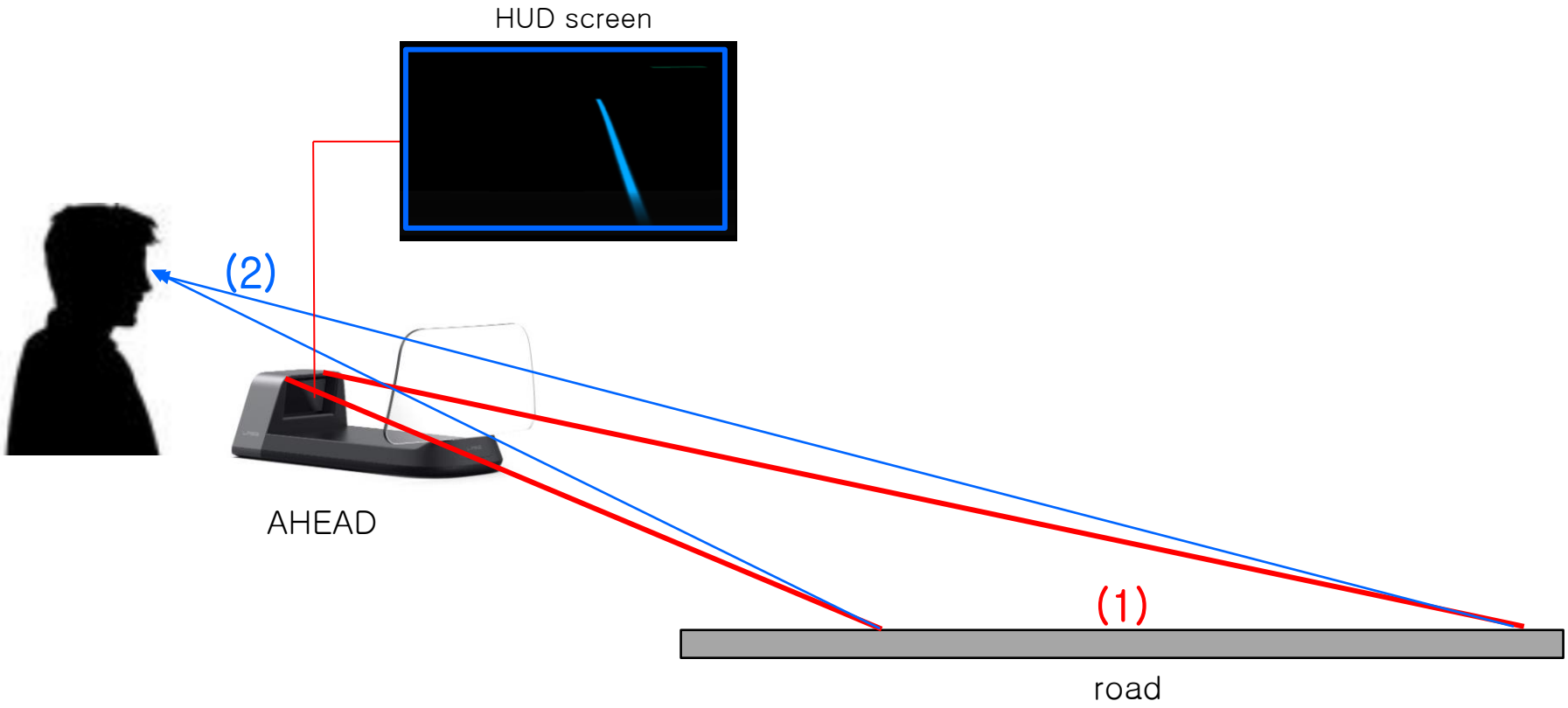


Bird eye sceen complete

HUD complete!

AHEAD

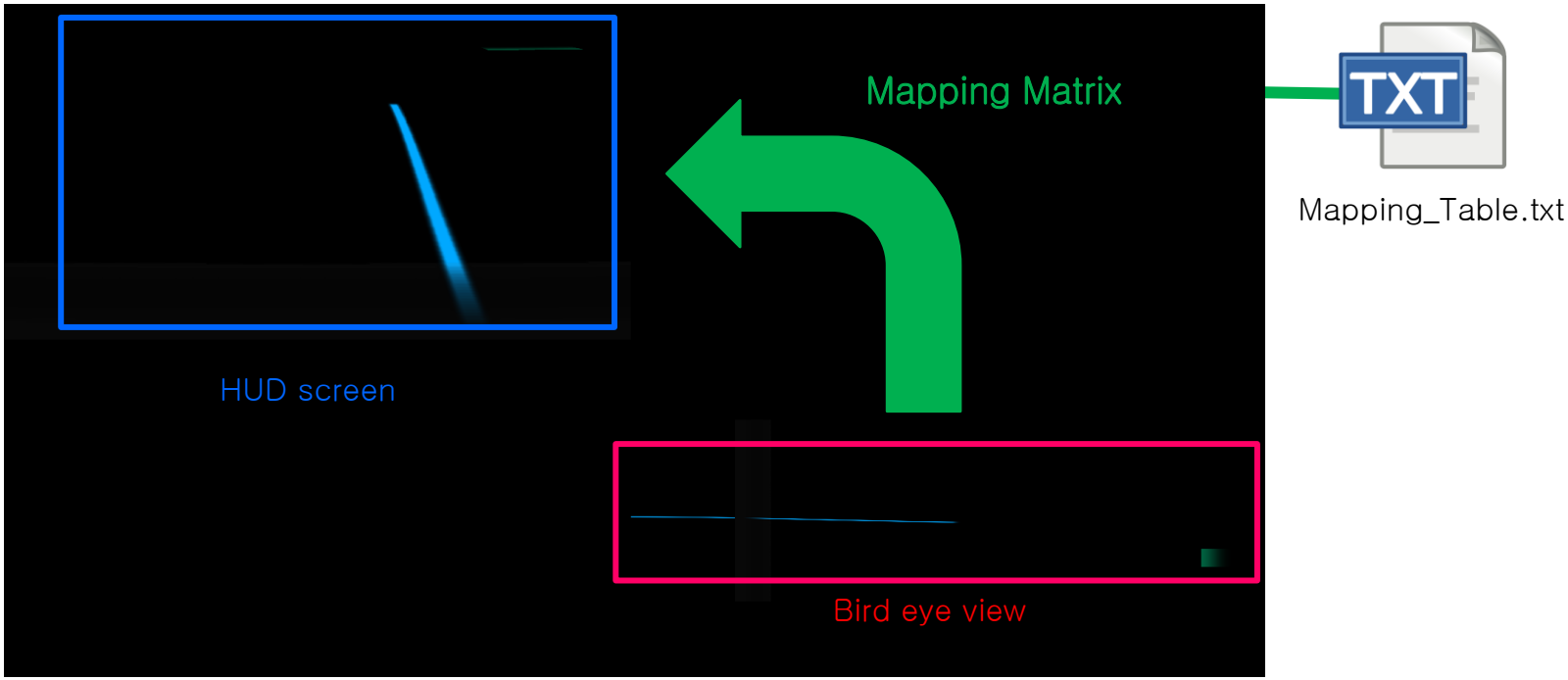
- 3D AR HUD process



3. Project progress

AHEAD Display

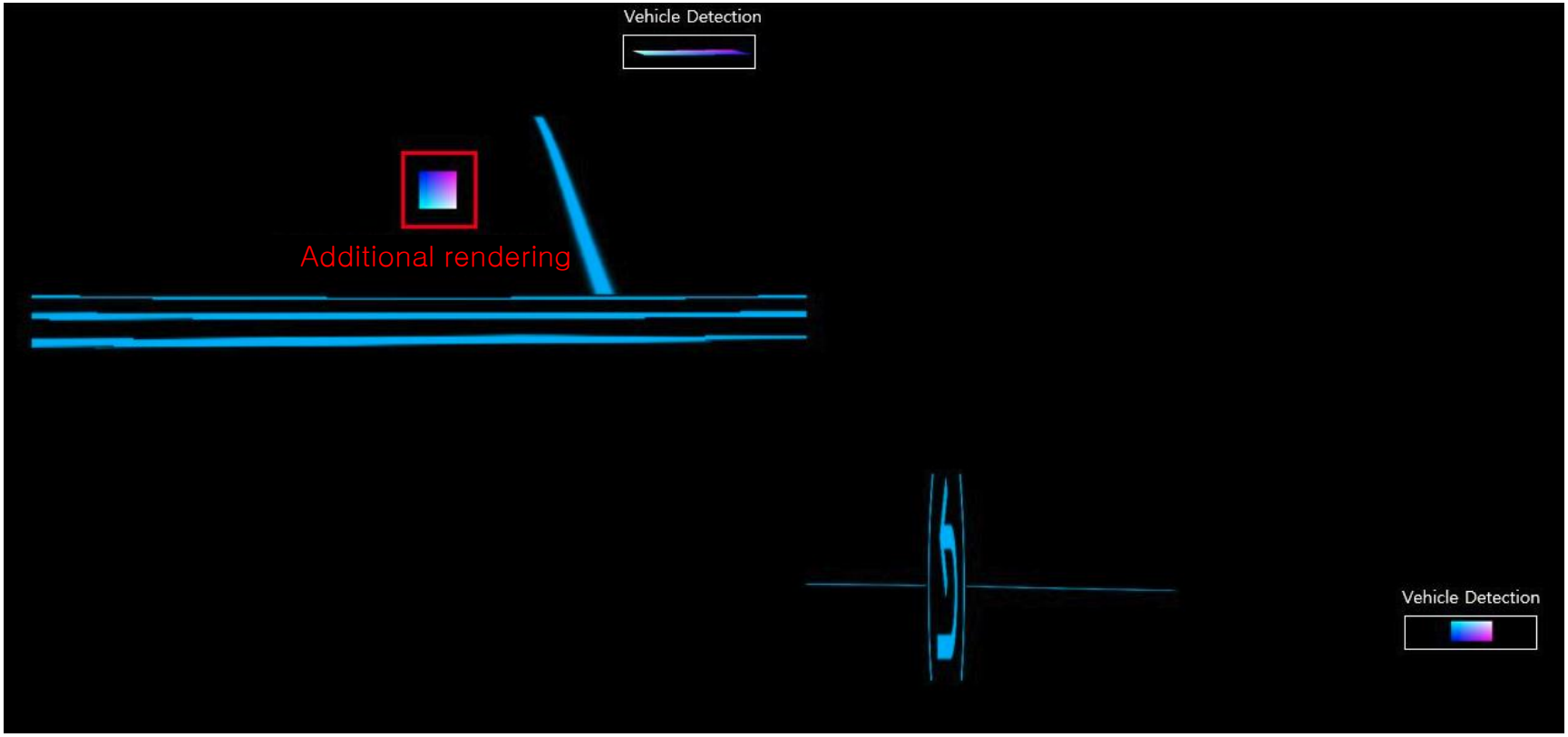
- Problem
  - Rendered 2D pixel coordinates(bird eye view) -> Pixel coordinates of 2D Hud screen
    => **Difficult to stand up UI**



Example of moved pixel coordinates

AHEAD Display

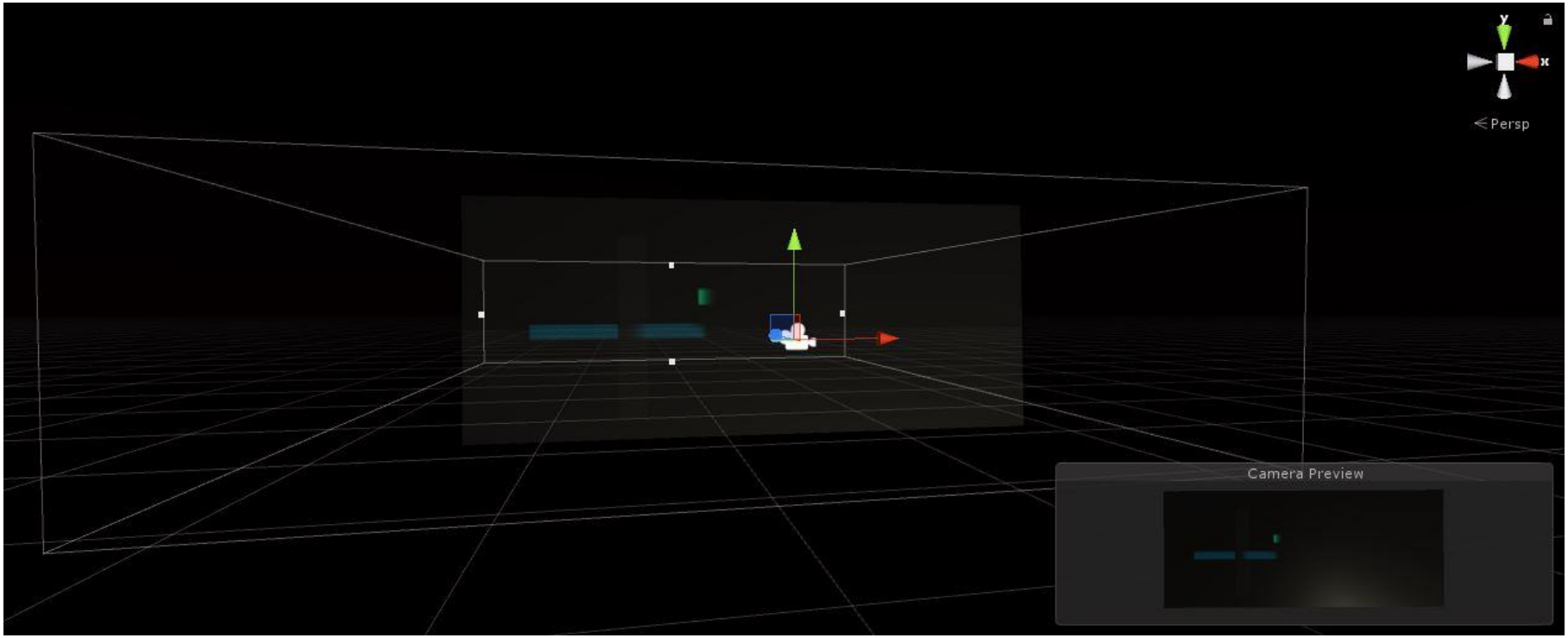
- Related work
  - Rendering method to add UI on HUD screen



Example of additional UI rendering

AHEAD Display

- Our approach
  - Bird eye view rendering(Orthographic projection)

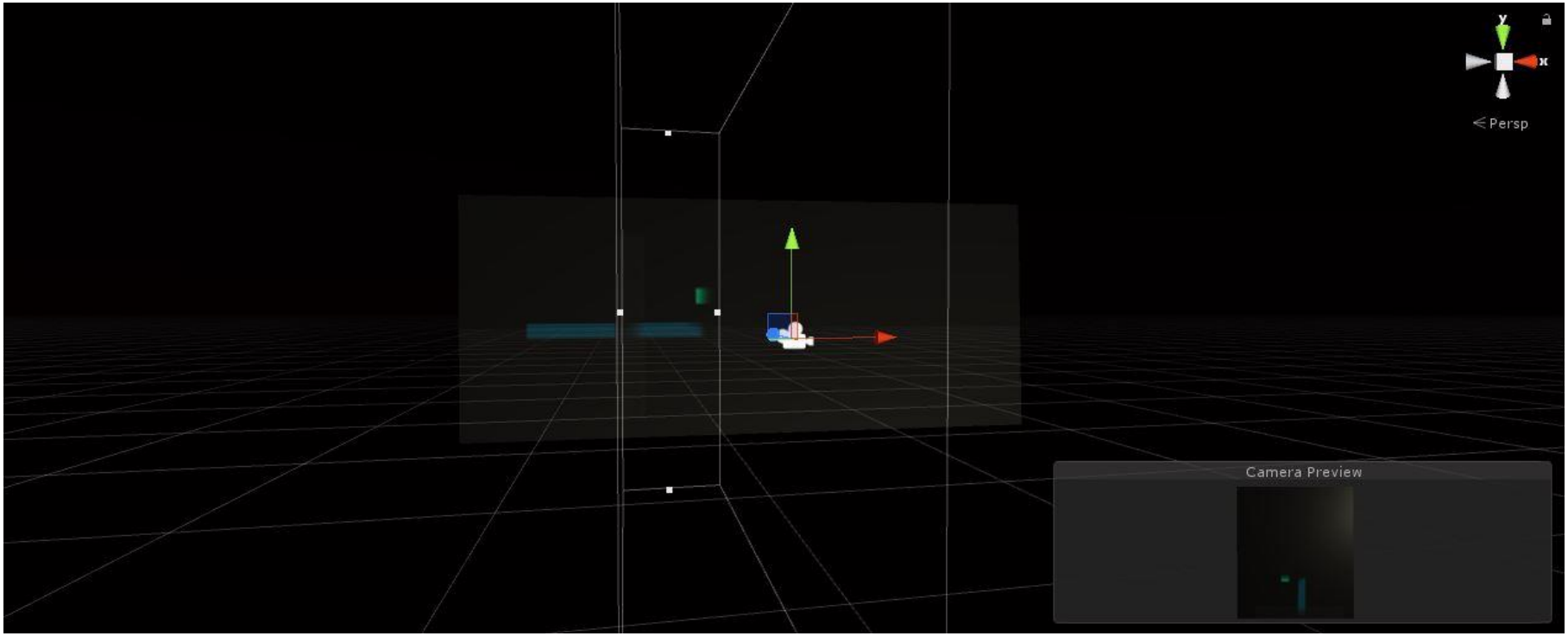


Example of bird eye view rendering

Source: Unity

AHEAD Display

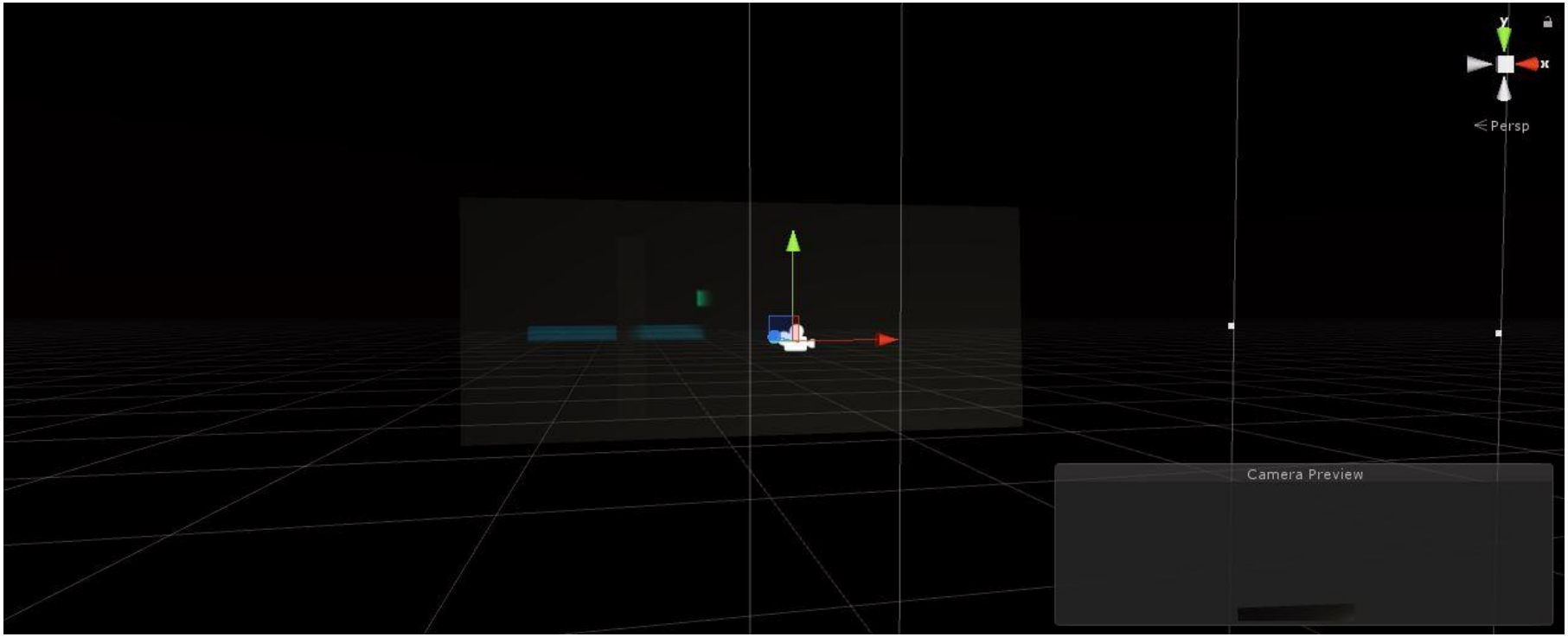
- Our approach
  - Step 1) Rotation in Z axis



Source: Unity

AHEAD Display

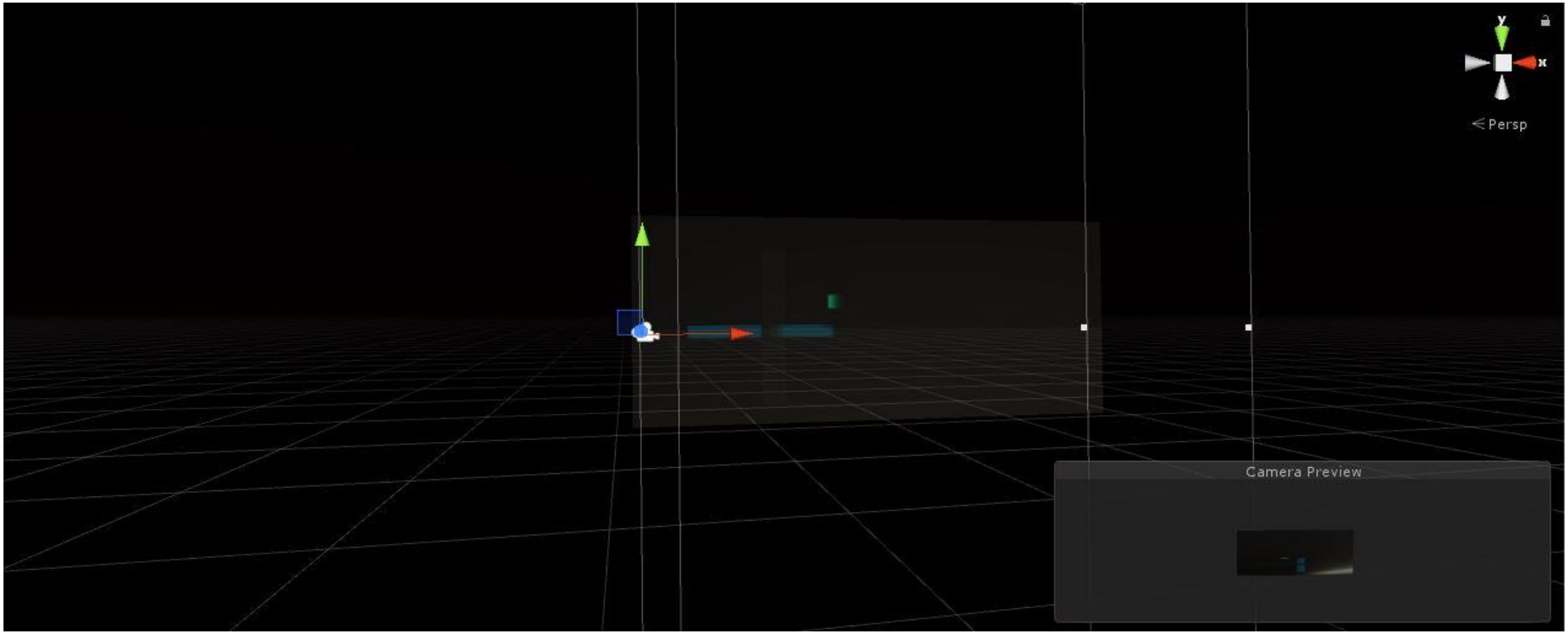
- Our approach
  - Step 2) Rotation in X axis



Source: Unity

AHEAD Display

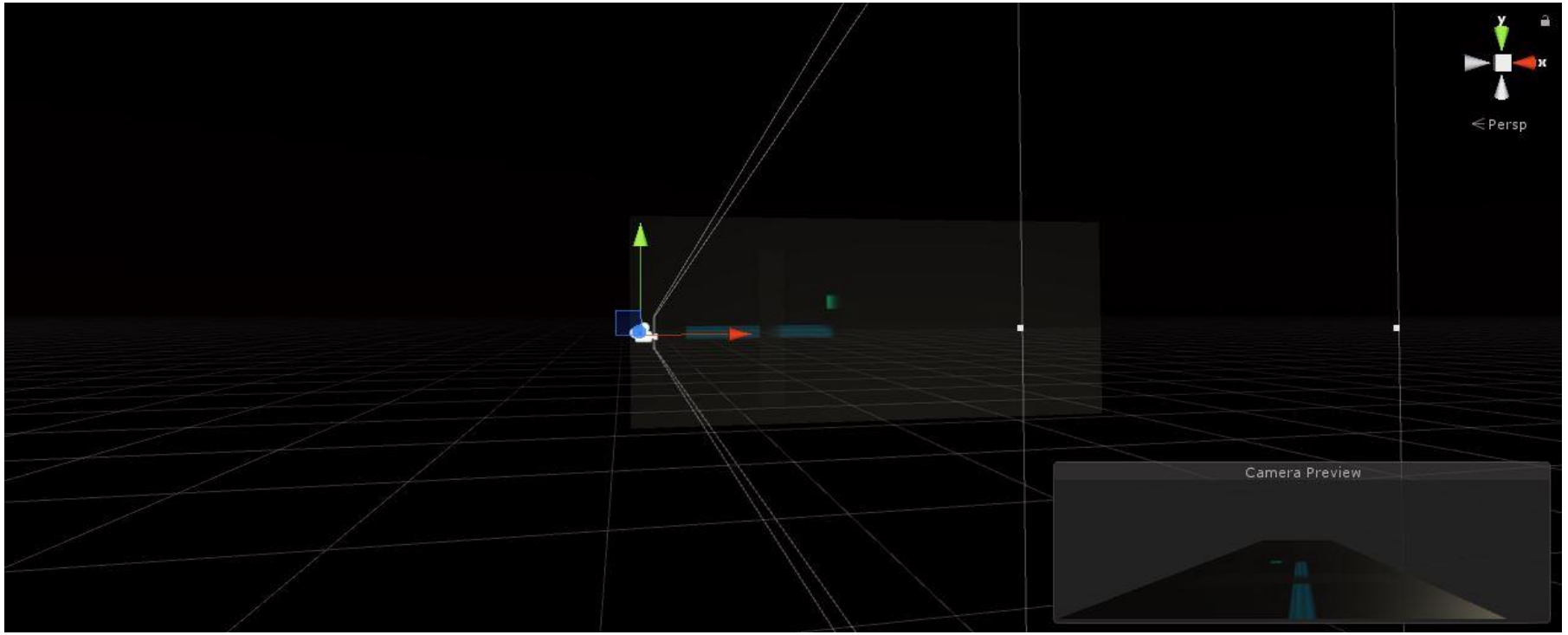
- Our approach
  - Step 3) Moving camera position



Source: Unity

AHEAD Display

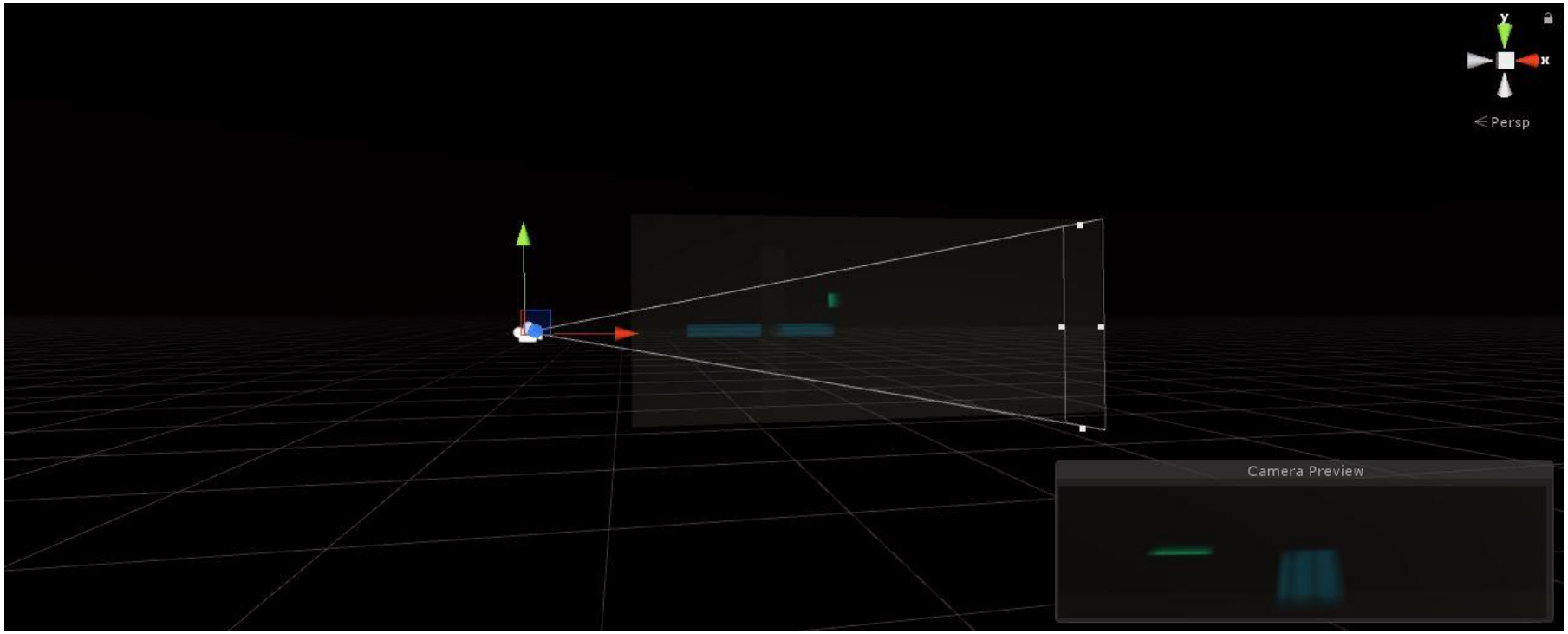
- Our approach
  - Step 4) Replacing perspective projection



Source: Unity

AHEAD Display

- Our approach
  - Step 5) Adjusting Fov(field of view)



Source: Unity

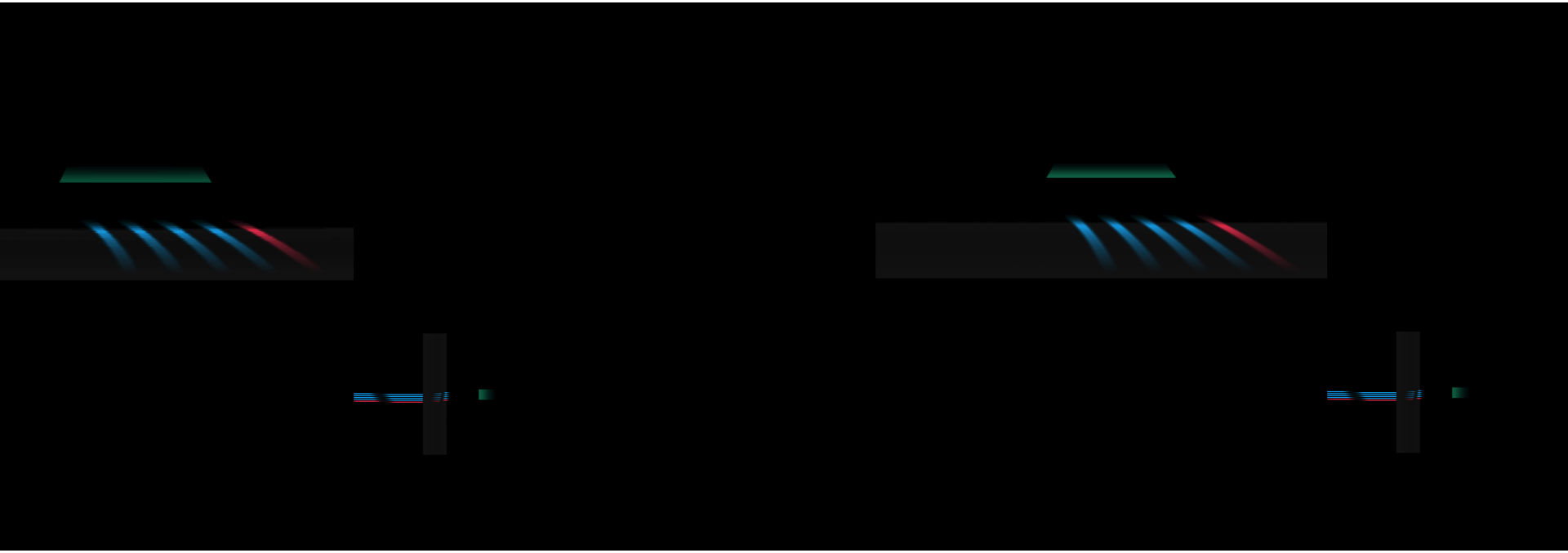
AHEAD Display

- Results
  - RoutePath

Example of RoutePath rendering (Left: Mapping matrix method, Right: Propose method)

AHEAD Display

- Results
  - FCW(Forward Collision Warning)



Example of FCW rendering (Left: Mapping matrix method, Right: Propose method)

AHEAD Display

- Results
  - LDW(Lane Departure Warning)

Example of LDW rendering (Left: Mapping matrix method, Right: Propose method)

AHEAD Display

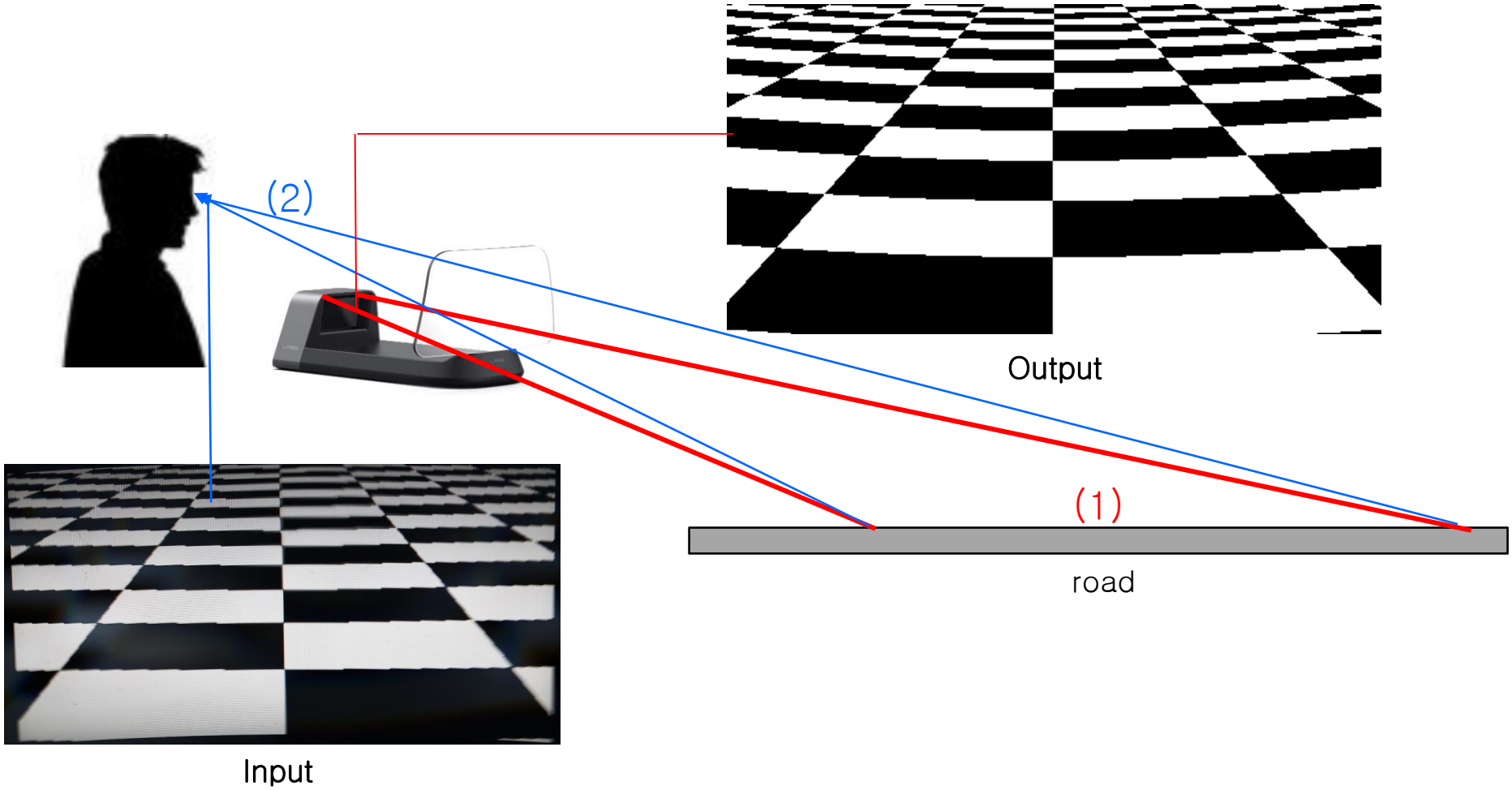
- Results
  - Turn-Left UI
    - UI scale change with distance

Example of UI rendering (Left: Mapping matrix method, Right: Propose method)

Results (Propose method +Switching Mobileye)

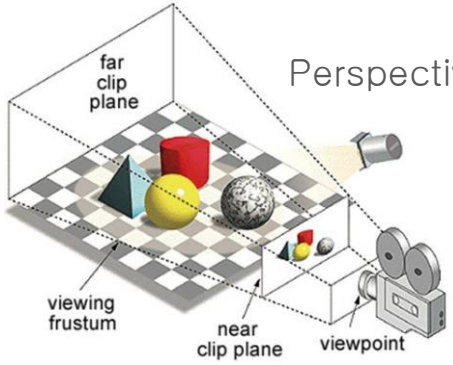
AHEAD Display

- Limitations
  - Output: Circular sector shape(부채꼴)
  - Input: Trapezoid shape(사다리꼴)

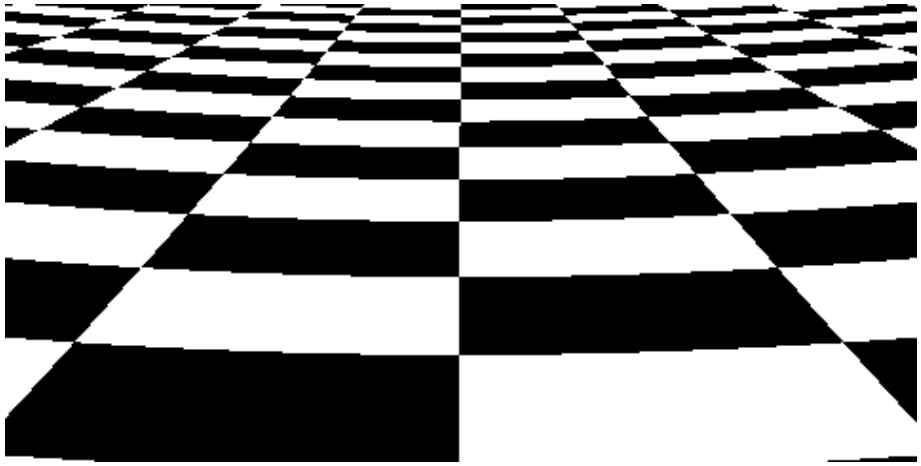


AHEAD Display

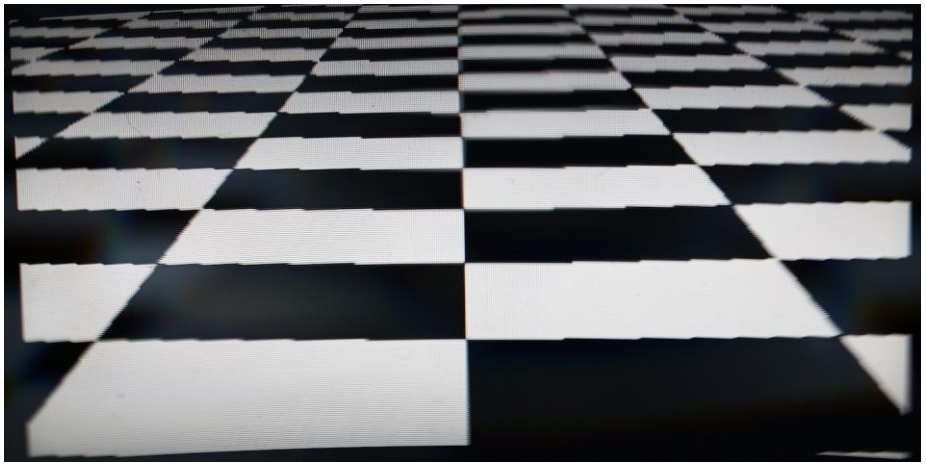
- Limitations



Perspective projection



Mapping matrix method



Perspective matrix method

Source:slideplayer

Future work

- Continue research
  - Solution 1) Pre-distortion method
  - Solution 2) Hybrid method
    - Bottom: Mapping matrix
    - Top: propose method

- Driving test
  - Ver 1) UI version of the propose method
  - Ver 2) Mobileye version

Q n A

Switching Mobileye

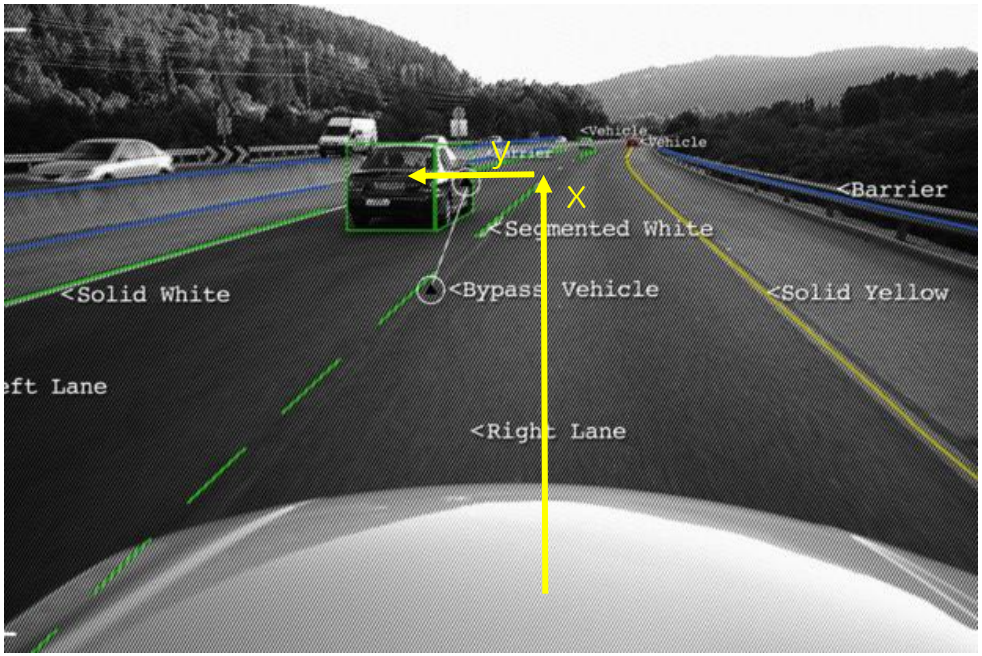
- LDW(Lane Departure Warning)
- Message
  - Physical distance between lane marl and camera on the lateral position
    - PerceptionInfo::P3(double) -> rightLane::C0(double)
    - PerceptionInfo::P4(double) -> leftLane::C0(double)

Lateral position(left)

Source:hasso hering

Switching Mobileye

- FCW(Forward Collision Warning)
- Message
  - The longitude position of the obstacle relative to the reference point.
    - Float64MultiArray::xpos(float) -> obstacles::pos_x(float)
    - Float64MultiArray::ypos(float) -> obstacles::pos_y(float)
    - . . .

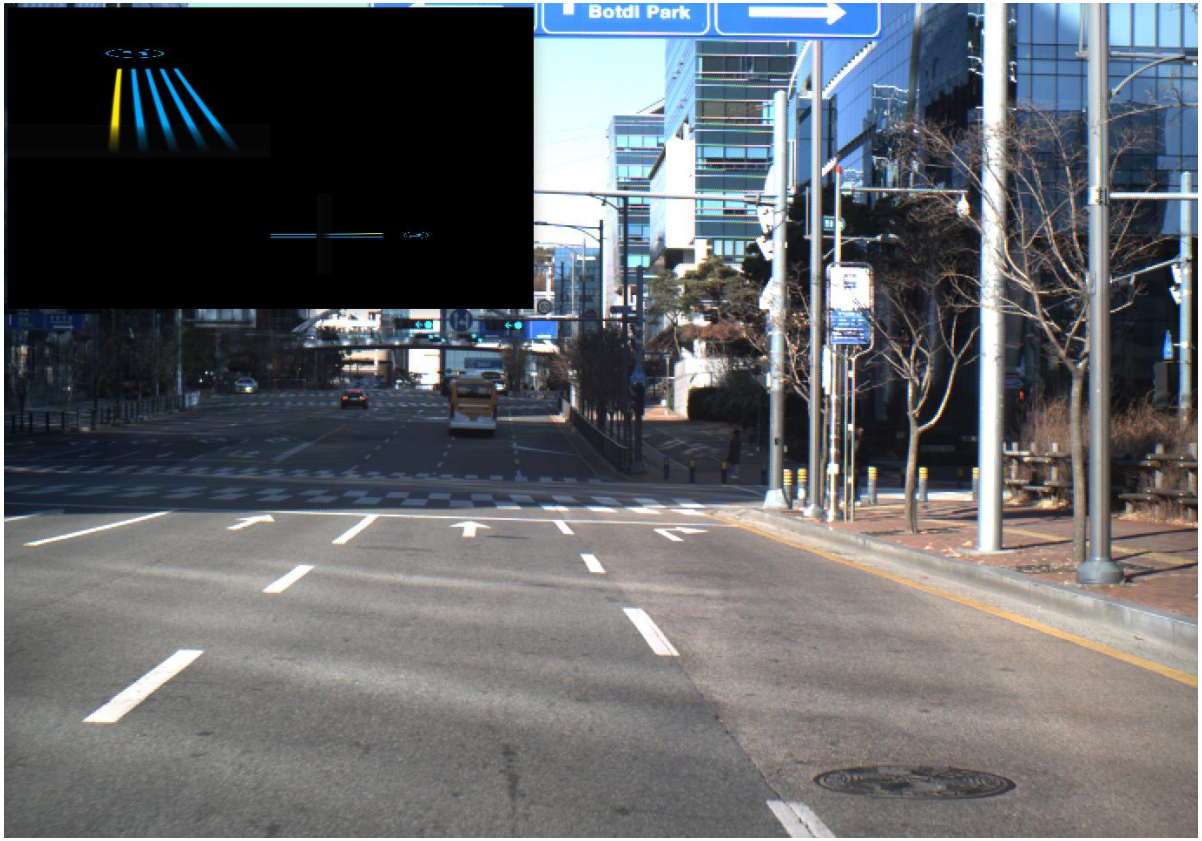


position of the obstacle relative

Source: computerworld

Switching Mobileye

- Results
  - LDW



Switching Mobileye

- Results
  - FDW