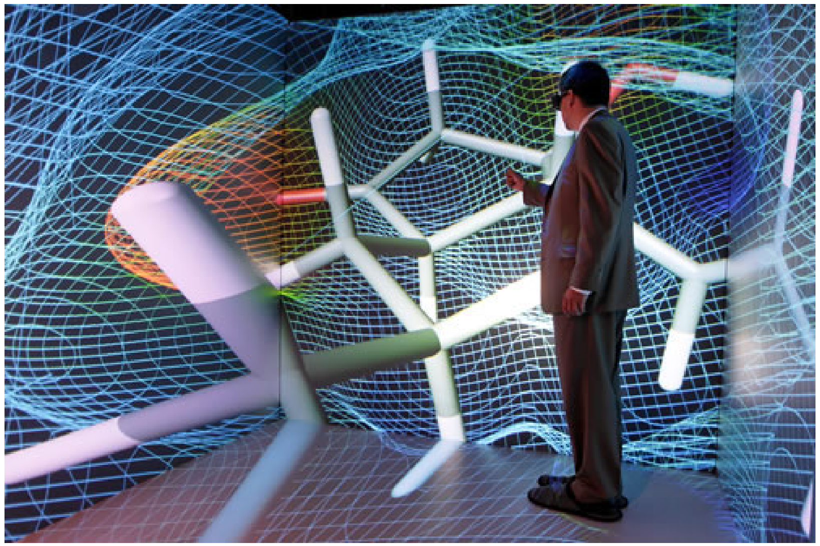
Stereoscopy

C. Andujar

Sep 2020

Motivation

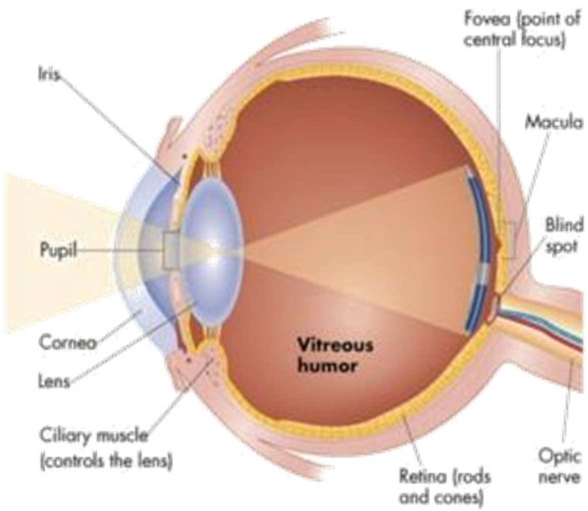
- One of the most important depth cues
- Key in all VR applications
- Influences both software and hardware



DEPTH PERCEPTION

Depth perception

- Our retinas only capture planar images
- The HVS is able to recover depth information by combining multiple depth cues.
- Depth cues = features that help the HVS obtain depth information.



Depth cues

- In the real-world, all depth cues agree.
- In VR, some depth cues might disagree!
  – Poor stereo → poor depth, eye strain (visual fatigue)
  – Conflicting stimuli → motion sickness

Depth cues

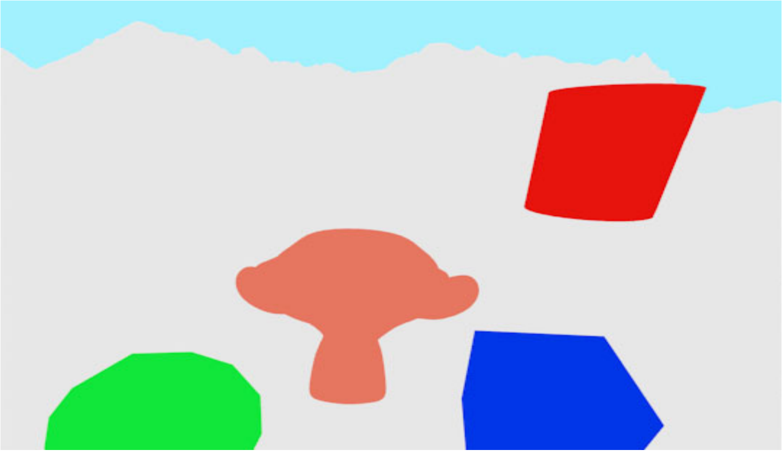
Depth cue classification

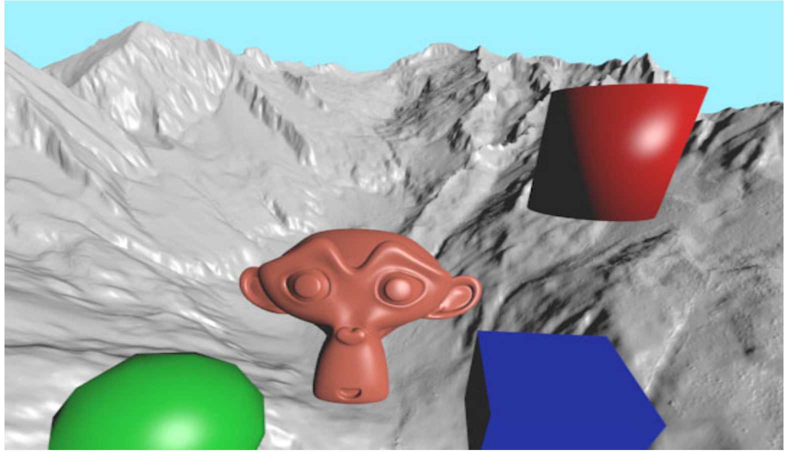
- Monocular
  - Shading, Shadows, Relative size, Perspective, Texture gradient, Atmospheric perspective, Motion parallax, Occlusion, Accommodation
- Binocular
  - Convergence, retinal disparity

1

Shading

Shading provides shape and depth information because it largely depends on the surface orientation

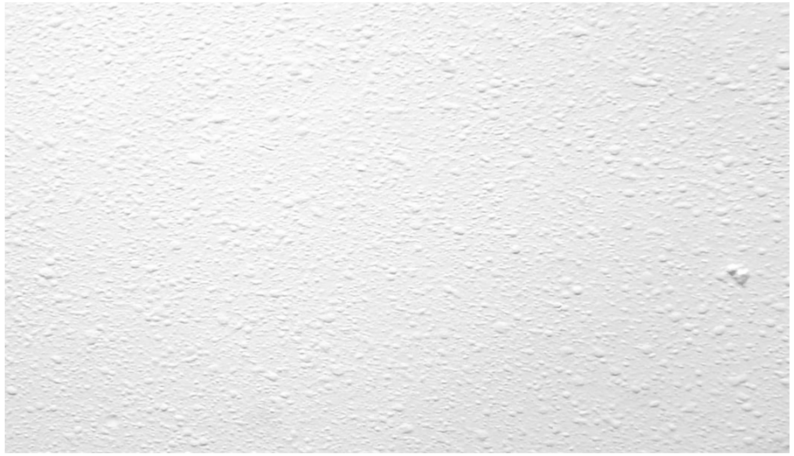


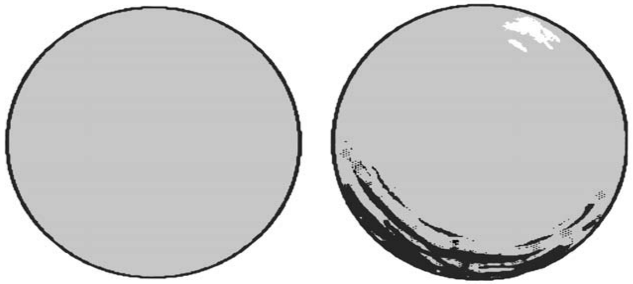


1

Shading

Shading provides shape and depth information because it largely depends on the surface orientation

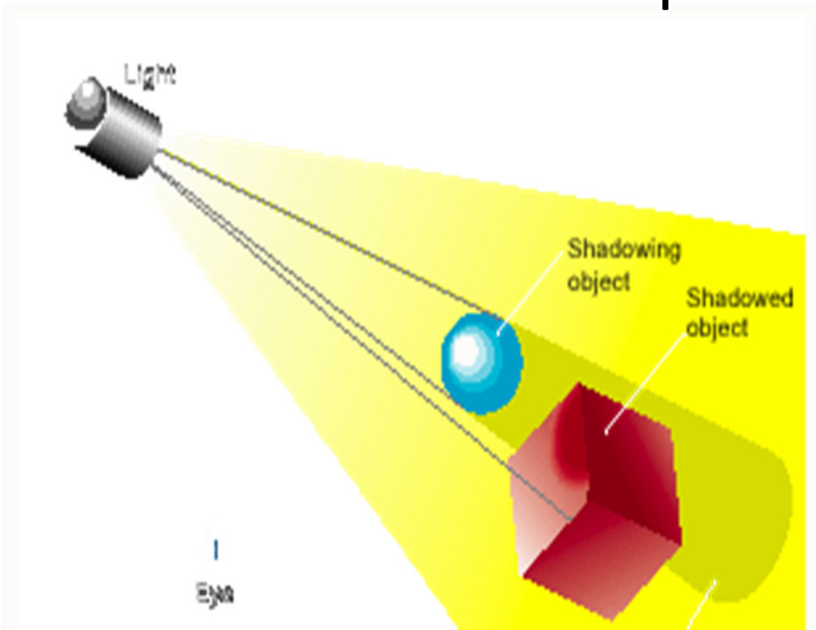




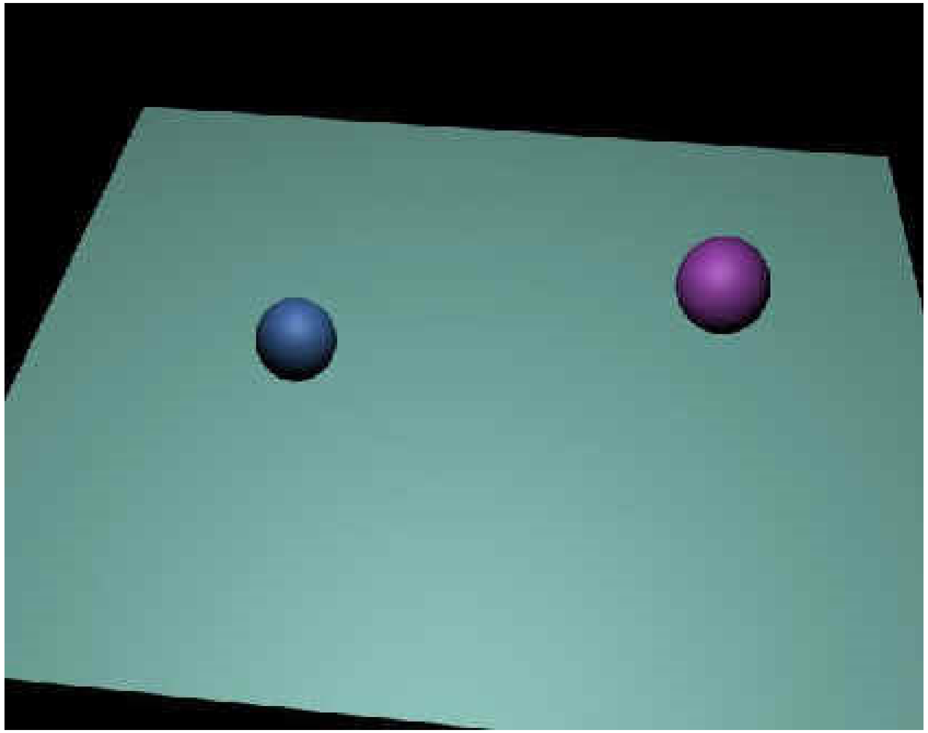
2

Shadows

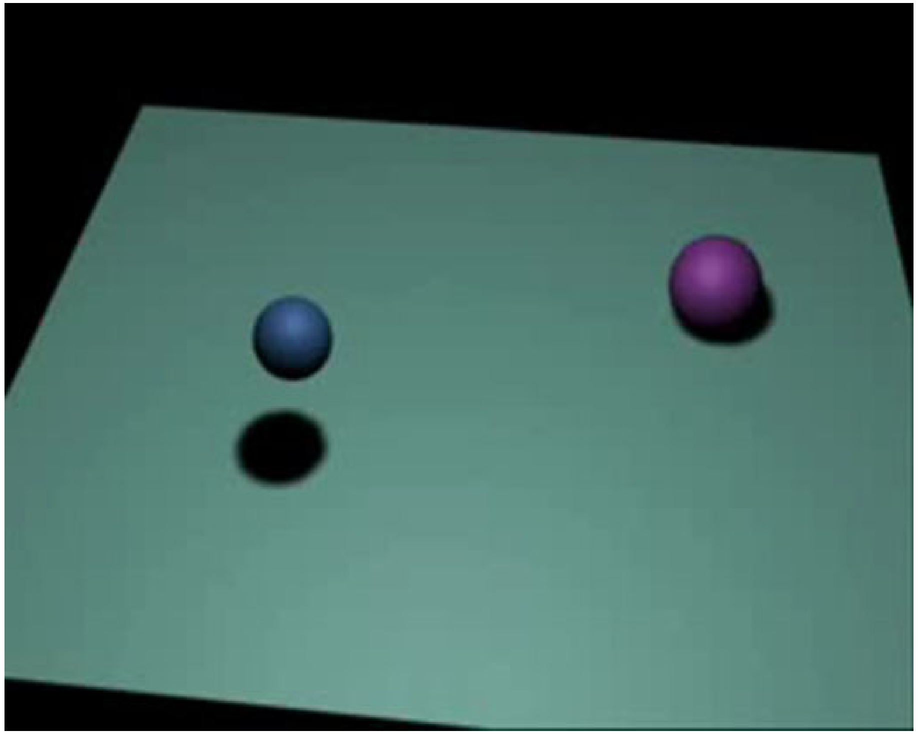
Shadows cast by objects play an important role in depth perception



Shadows



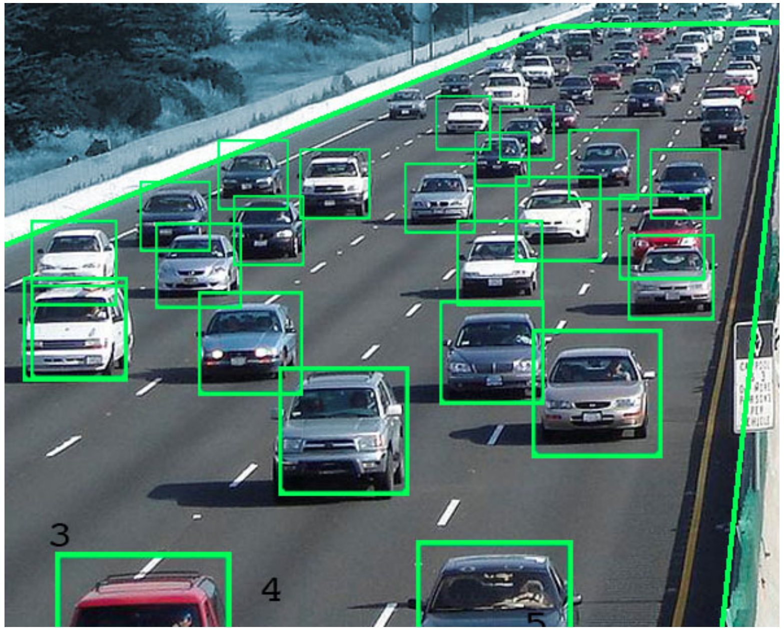
Shadows



3

Retinal image size

The visual system exploits the relative size of familiar objects to judge distance.



4

Perspective (linear perspective)

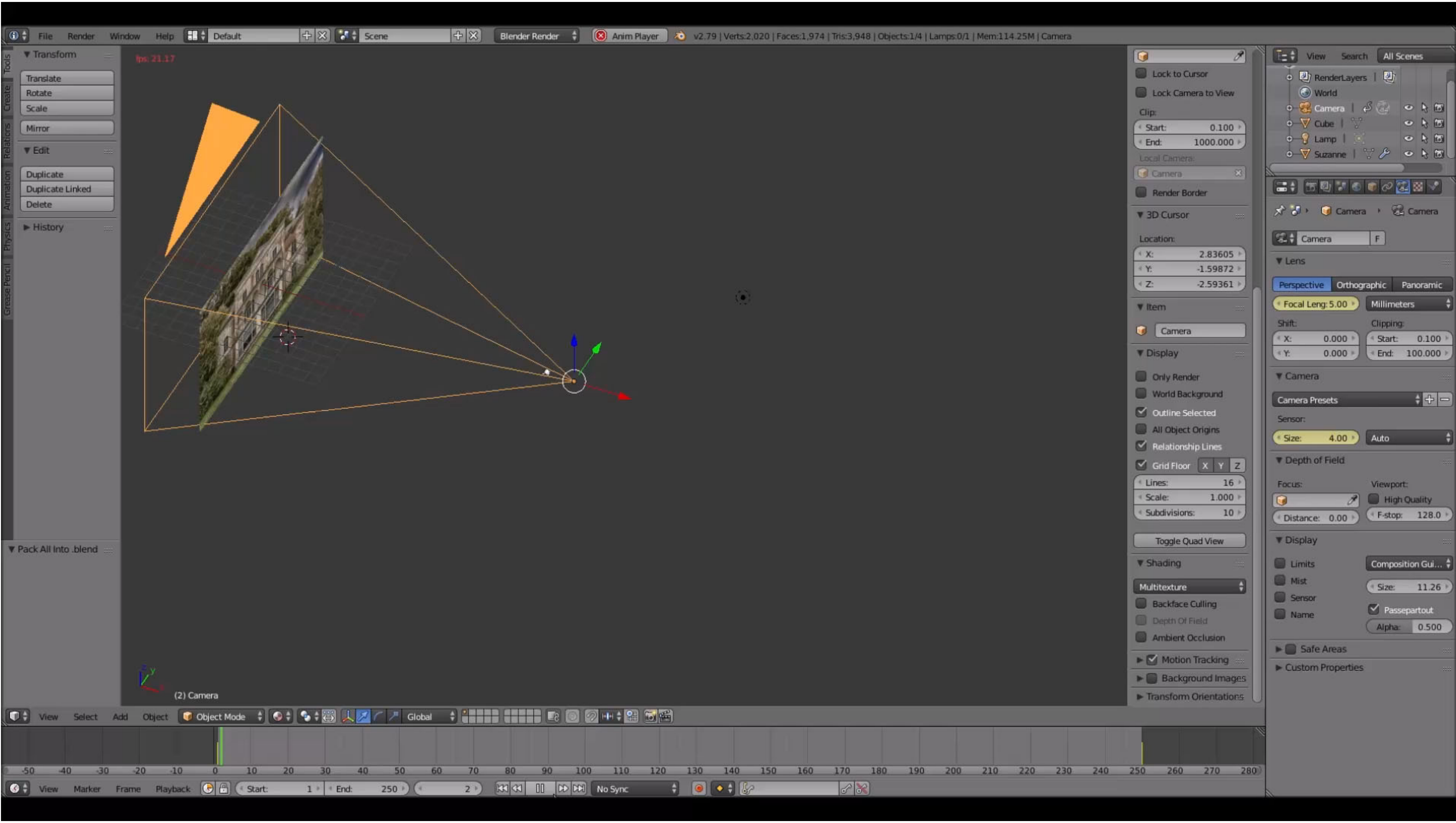
- As objects become more distant, they appear smaller because their visual angle decreases.
- Parallel lines appear to be meeting at a distant point (the vanishing point) on the horizon.

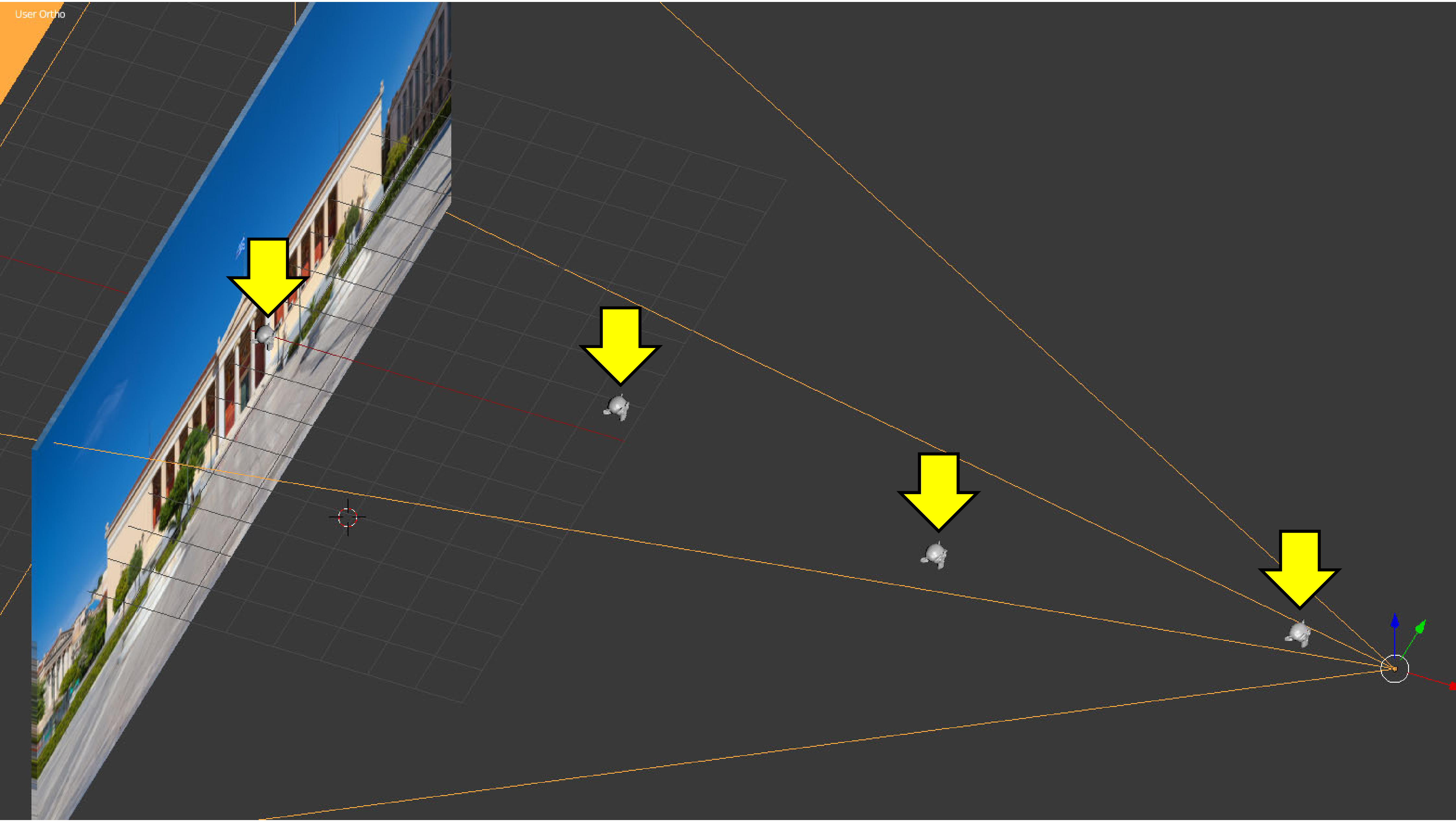
Perspective (linear perspective)

We can compress depth by increasing focal length (photos) or decreasing field-of-view (non-VR renders)

Photo: artandstructure.com

18mm "wide" 50mm "normal" 120mm "telephoto"


File Render Window Help
Default
Scene
Blender Render
Anim Player
v2.79 | Verts:2,020 | Faces:1,974 | Tris:3,948 | Objects:1/4 | Lamps:0/1 | Mem:114.25M | Camera
Transform
Translate
Rotate
Scale
Mirror
Edit
Duplicate
Duplicate Linked
Delete
History
Pack All Into .blend
fps: 21.17
(2) Camera
Lock to Cursor
Lock Camera to View
Clip:
Start: 0.100
End: 1000.000
Local Camera:
Camera
Render Border
3D Cursor
Location:
X: 2.83605
Y: -1.59872
Z: -2.59361
Item
Camera
Display
Only Render
World Background
Outline Selected
All Object Origins
Relationship Lines
Grid Floor X Y Z
Lines: 16
Scale: 1.000
Subdivisions: 10
Toggle Quad View
Shading
Multitexture
Backface Culling
Depth Of Field
Ambient Occlusion
Motion Tracking
Background Images
Transform Orientations
View Search All Scenes
RenderLayers
World
Camera
Cube
Lamp
Suzanne
Camera › Camera
Camera F
Lens
Perspective Orthographic Panoramic
Focal Leng: 5.00
Millimeters
Shift:
X: 0.000
Y: 0.000
Clipping:
Start: 0.100
End: 100.000
Camera
Camera Presets
Sensor:
Size: 4.00
Auto
Depth of Field
Focus:
Distance: 0.00
Viewport:
High Quality
F-stop: 128.0
Display
Limits
Mist
Sensor
Name
Composition Gui...
Size: 11.26
Passepartout
Alpha: 0.500
Safe Areas
Custom Properties
View Select Add Object
Object Mode
Global
View Marker Frame Playback
Start: 1
End: 250
2
No Sync


User Ortho

Telephoto lenses

Some lenses reach 500 mm focal length!

5

Texture gradient

The further apart an object, the higher the spatial frequency of its texture

6

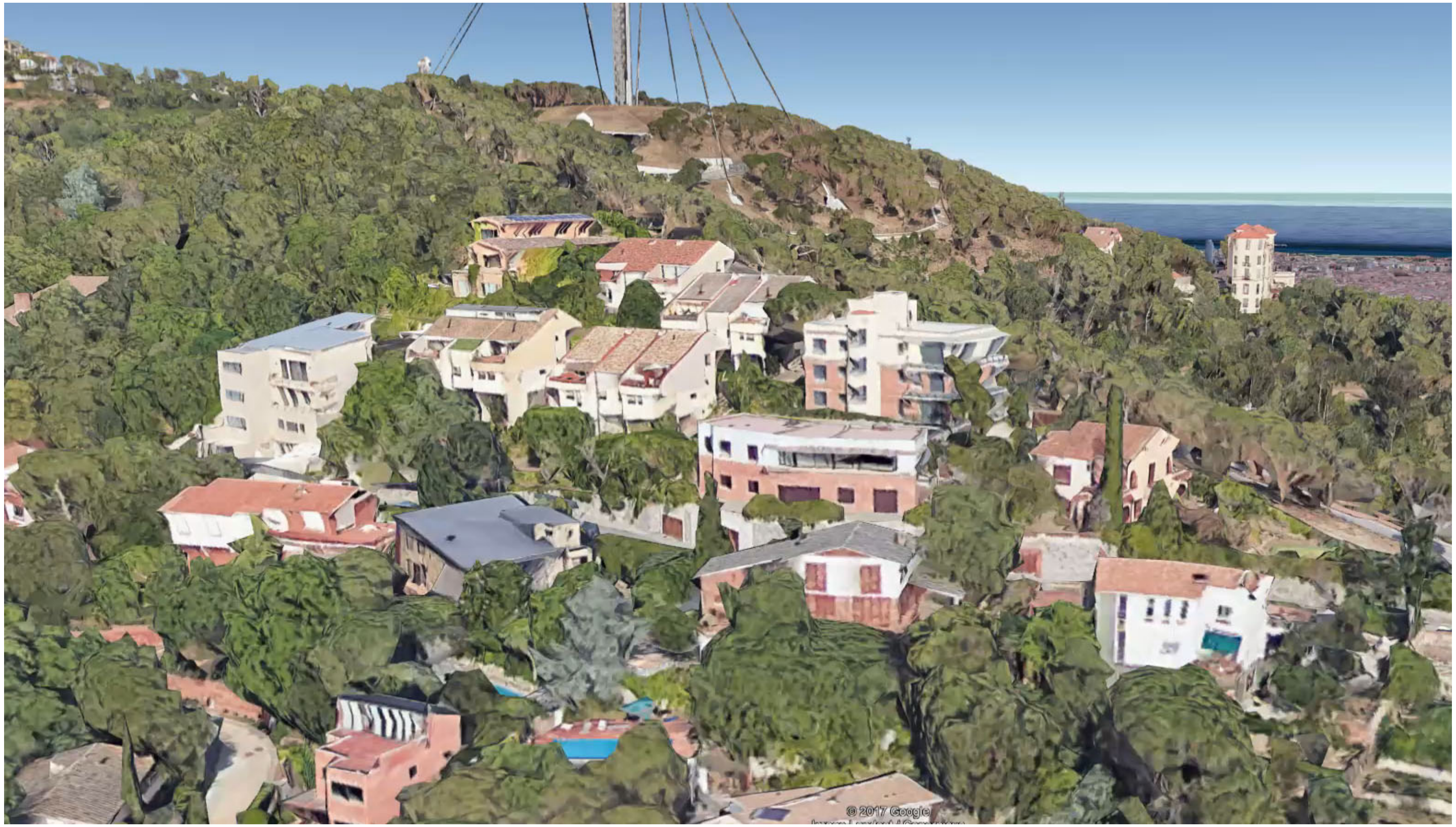
Atmospheric perspective (fog)

Objects that are a great distance away look hazier due to light scattering by the atmosphere

Atmospheric perspective (fog)

7

Motion parallax



8

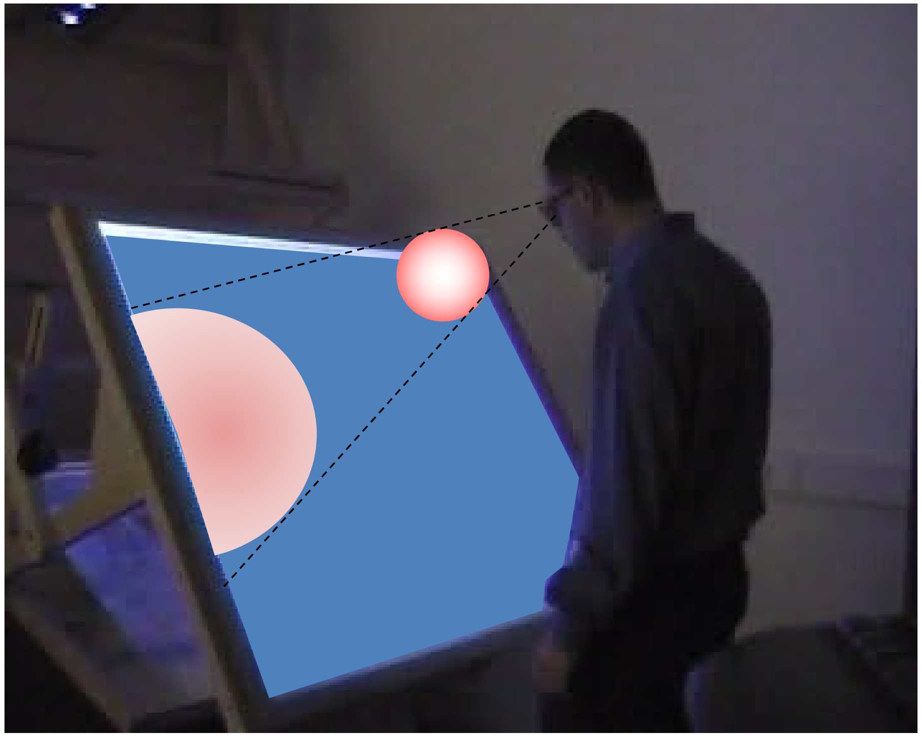
Occlusion (interposition)

Objects that block other objects appear to be closer.

Occlusion – screen surround

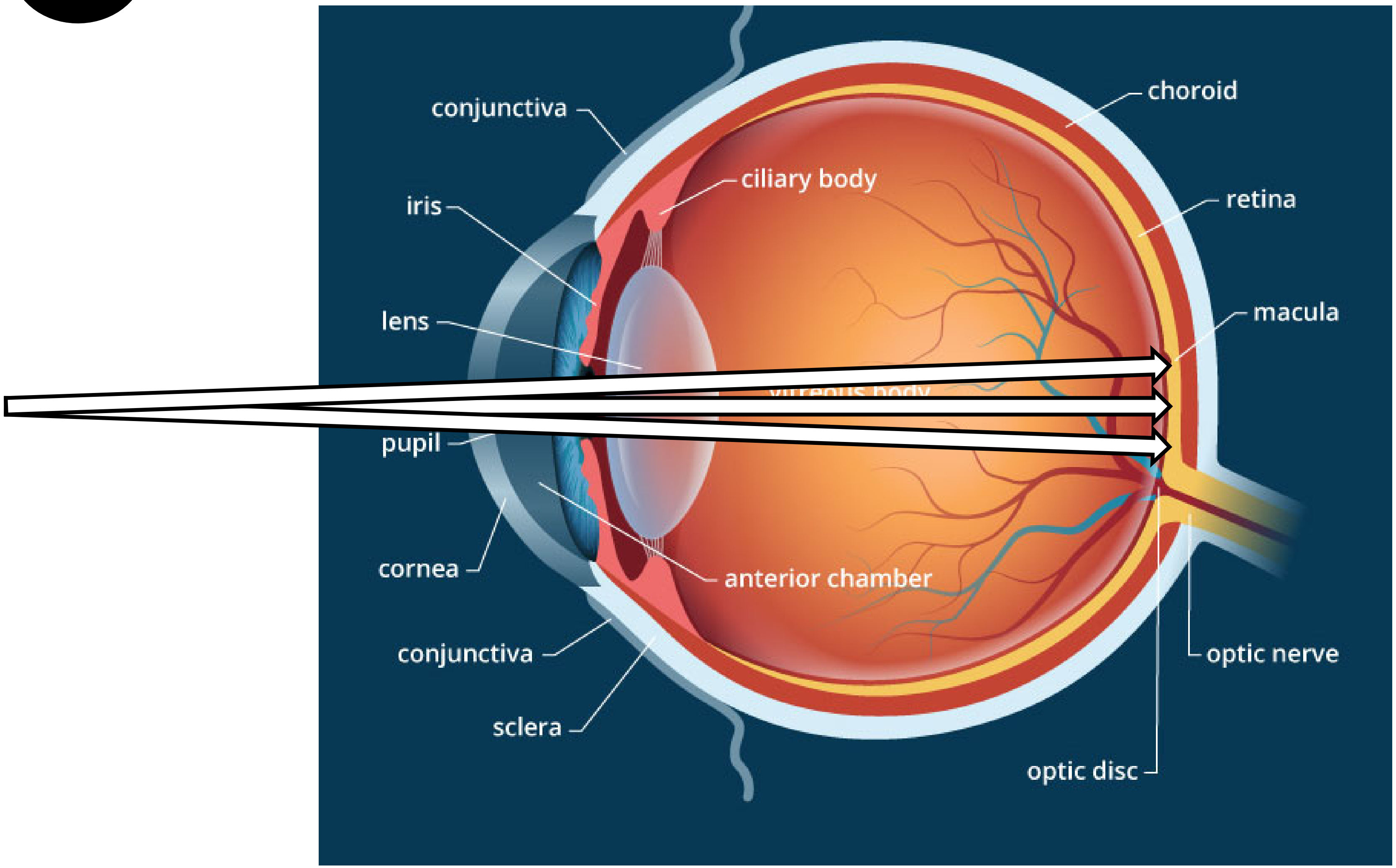
Occlusion – screen surround

Occlusion – screen surround



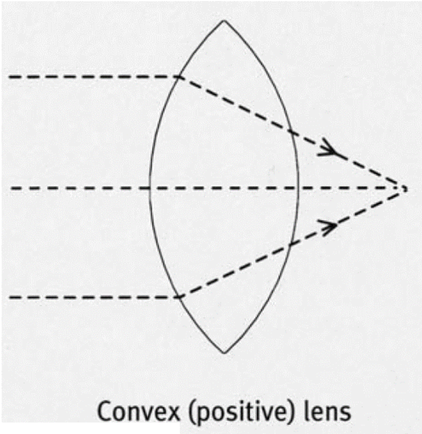
9

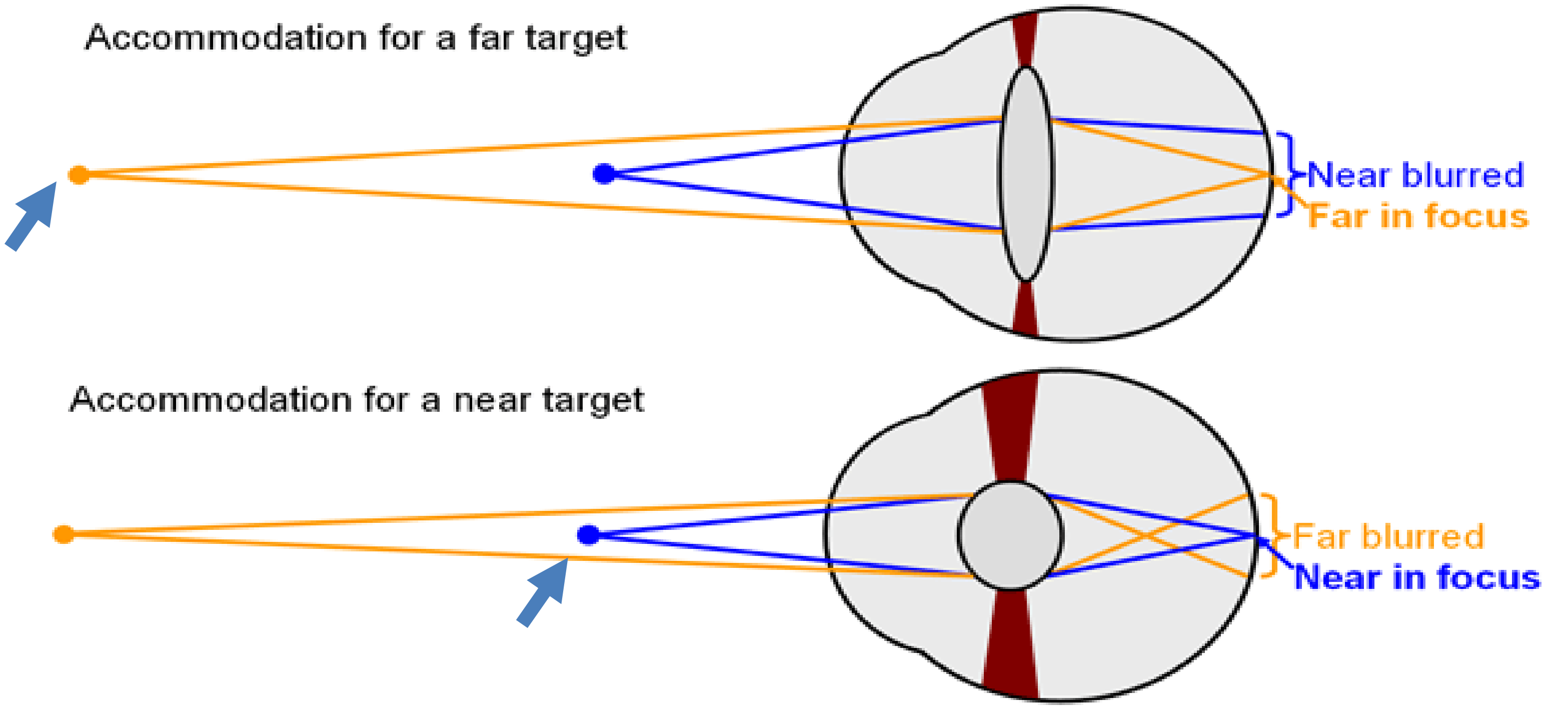
Accommodation



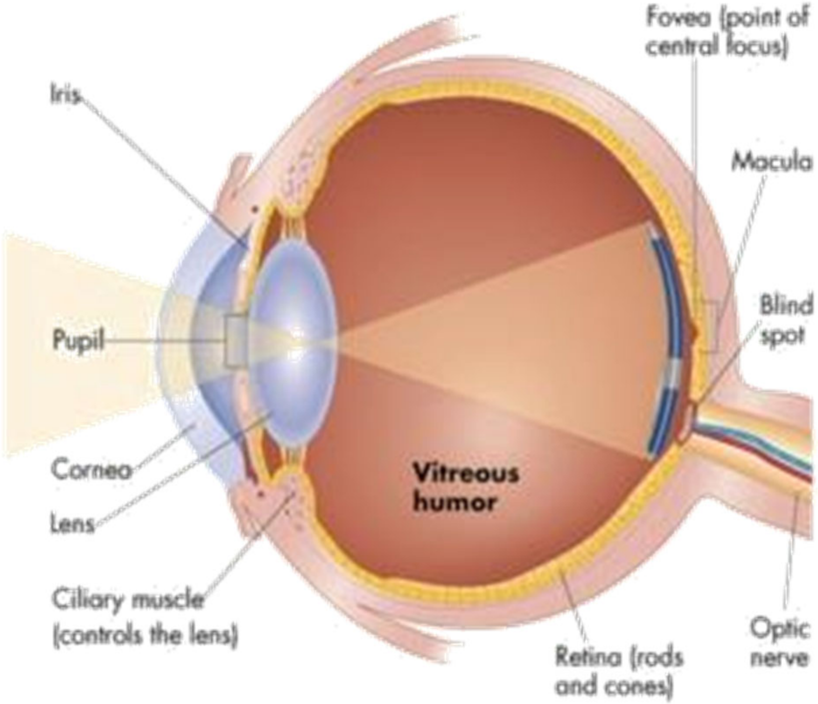
Accommodation

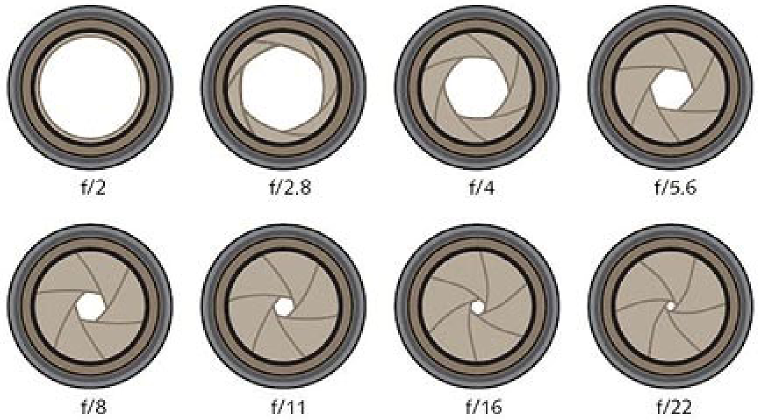
- **Deformation of the eye lens** (*cristalino*)
- Allows to **focus objects** at some distance



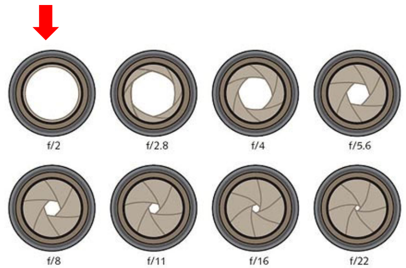


Accommodation vs pupil size (camera analog)

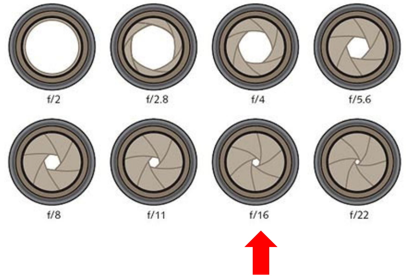




Accommodation



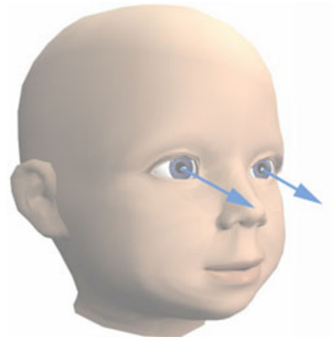
Accommodation

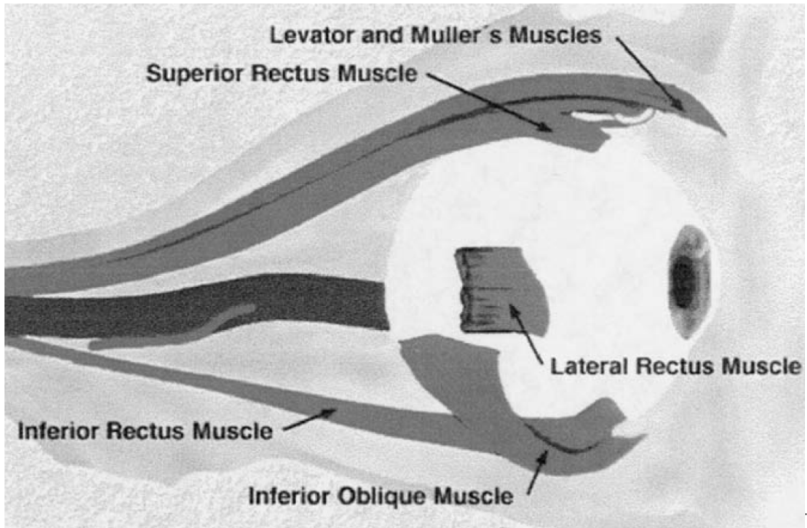


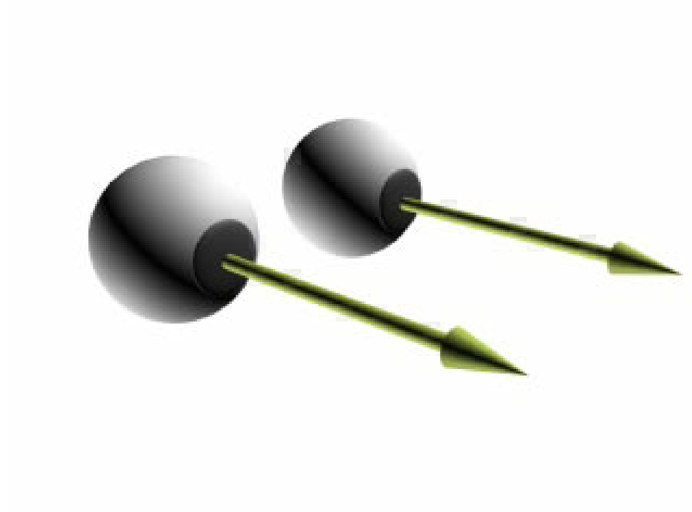
10

Convergence

- **Rotation of the eye balls**
- Allows us to **place the target at the fovea**







Accommodation & convergence

Points in common:

- Both depend on the depth of the fixation point
- Both are depth cues
- There is a natural relationship between accomodation and convergence
- In most VR stereo displays, this relationship is broken → **vision decoupling**

RETINAL DISPARITY

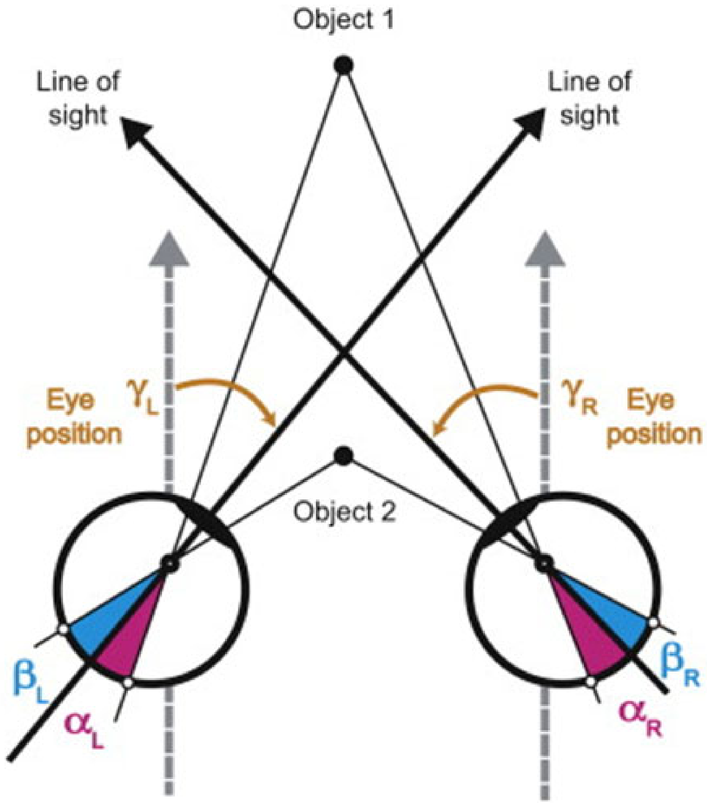
11 Retinal disparity

Difference in the L/R images of an object due to the eyes' horizontal separation

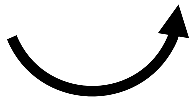
Fusion and stereopsis

- The HVS combines two images with disparity into a single image with depth.
- This ability is called **fusion** and the resulting sense is called **stereopsis**.

Retinal disparity (more formally)



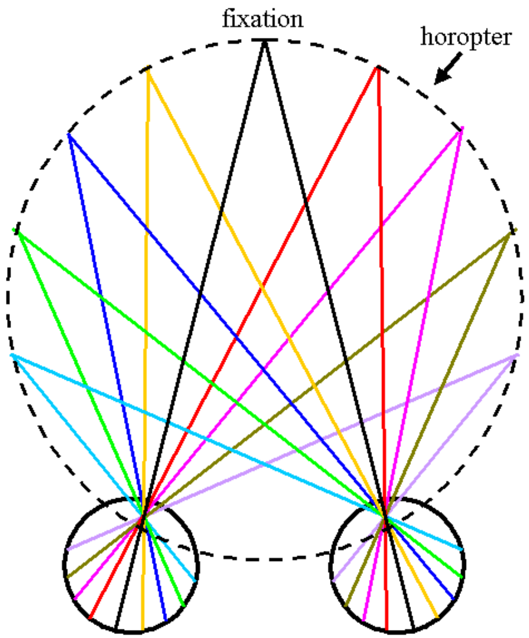
	Retinal Disparity
Object 1	$\alpha_L - \alpha_R$
Object 2	$\beta_L - \beta_R$



Positive angles

Retinal disparity (more formally)

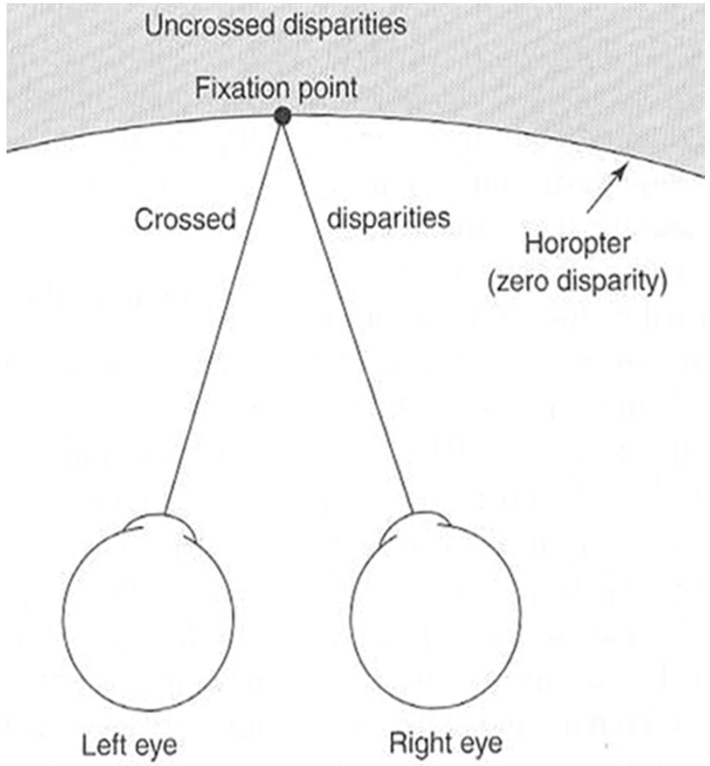
Horopter: surface defined by points that, for a given convergence, are projected onto points with **no disparity**



Retinal disparity (more formally)

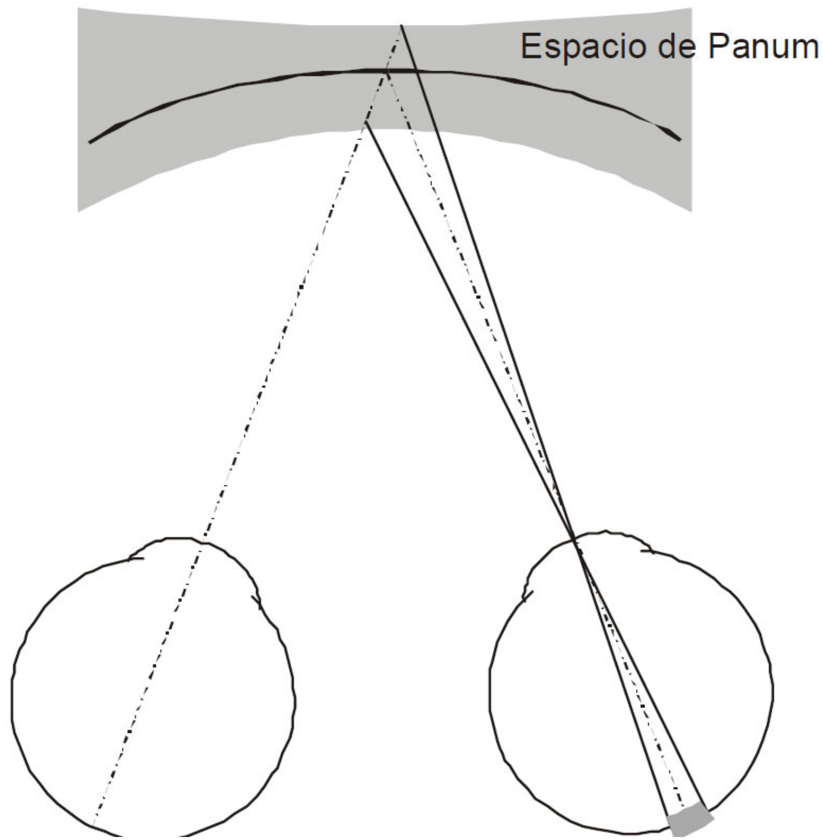
Points closer than the Horopter have **crossed disparity** (negative disparity)

Points farther than the Horopter have **uncrossed disparity** (positive disparity)



Retinal disparity (more formally)

- **Panum's fusion area:** Space around the horopter where fusion is feasible.
- **Within Panum** points can be fused (resulting in a single image with depth).
- **Outside Panum** fusion fails, producing double images.
- Near the fovea, the maximum binocular disparity resulting in fusion corresponds to a visual angle of about **1/6 of one degree.**



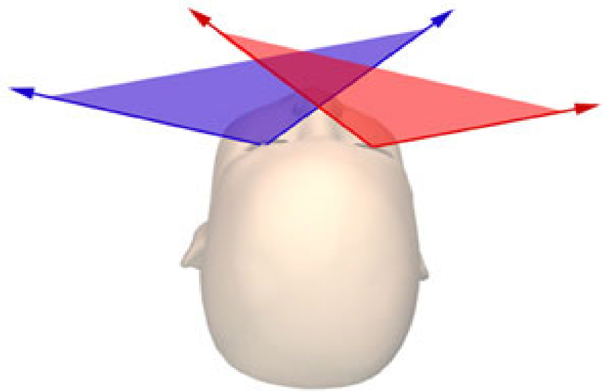
Retinal disparity

Remember:

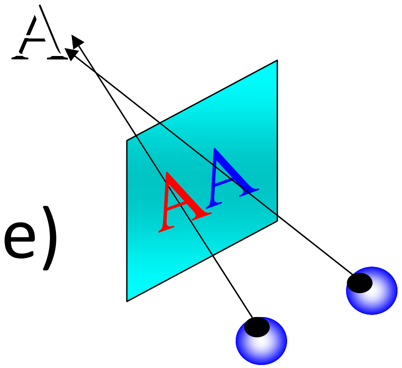
- Every scene point has its own retinal disparity.
- If we change fixation point, we change retinal disparities.

Human eye fact sheet

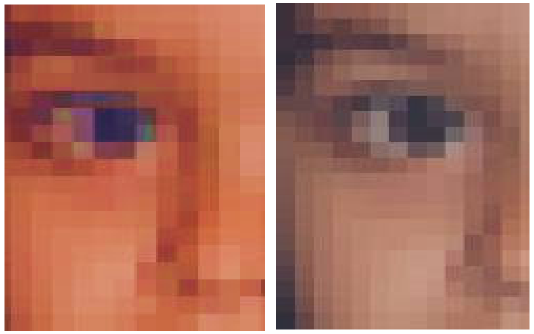
- Range IOD: 5 - 7.5 cm
- Average IOD: 6.3 – 6.6 cm
- Eye field-of-view: 160°-180° hor, 130° vert
- Horizontal overlap: 140°



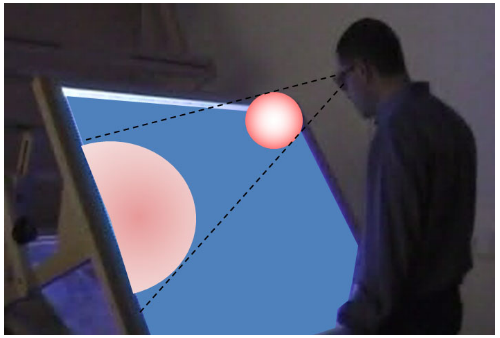
Summary of issues



- Vision-decoupling (accommodation vs convergence)



- Image congruence (correspondence problem)



- Depth cue conflicts (e.g. Screen surround)

Two screens, one screen, no screen

TAXONOMY OF 3D DISPLAYS

According to number of screens

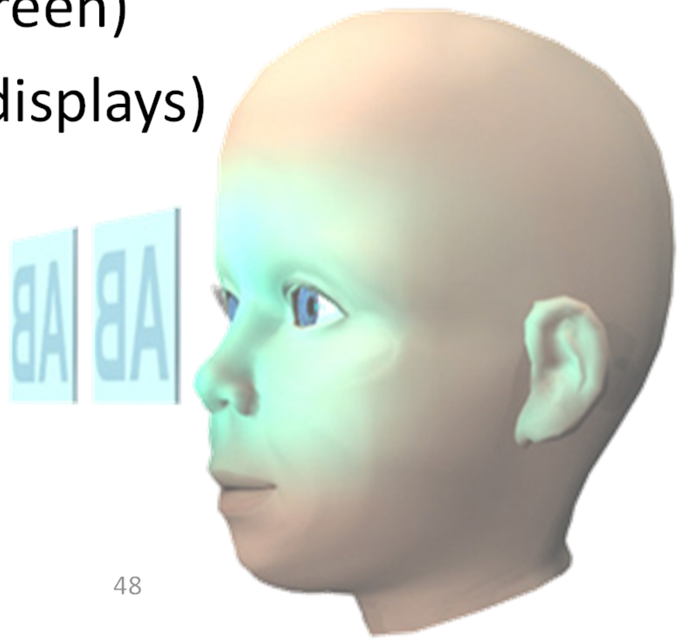
Where are the L/R images projected onto?

- Two independent screen regions (eg HMD)
- A single screen (3D TV, 3D cinema, CAVE…)
- Directly onto the retina (no screen)
- Spinning screen (holographic displays)

According to number of screens

Where are the L/R images projected onto?

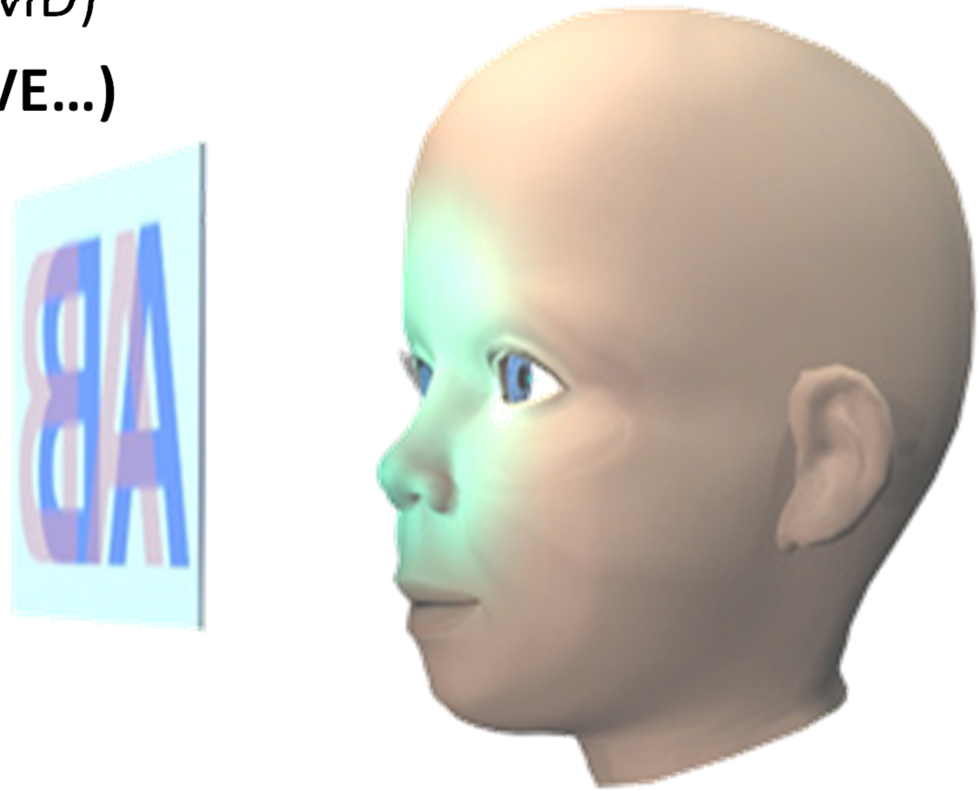
- **Two independent screen regions (eg HMD)**
- A single screen (3D TV, 3D cinema, CAVE…)
- Directly onto the retina (no screen)
- Spinning screen (holographic displays)



According to number of screens

Where are the L/R images projected onto?

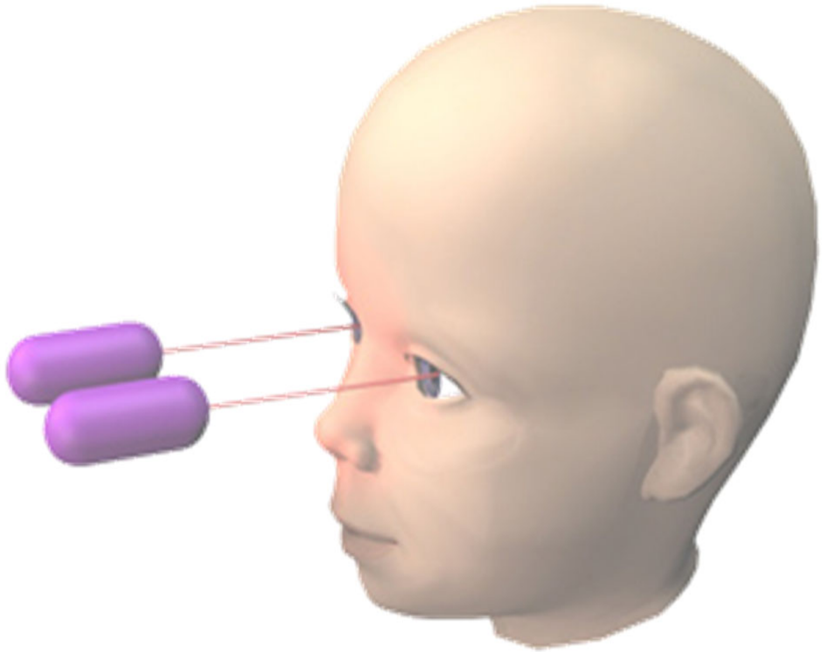
- Two independent screen regions (eg HMD)
- **A single screen (3D TV, 3D cinema, CAVE…)**
- Directly onto the retina (no screen)
- Spinning screen (holographic displays)



According to number of screens

Where are the L/R images projected onto?

- Two independent screen regions (eg HMD)
- A single screen (3D TV, 3D cinema, CAVE…)
- **Directly onto the retina (no screen)**
- Spinning screen (holographic displays)

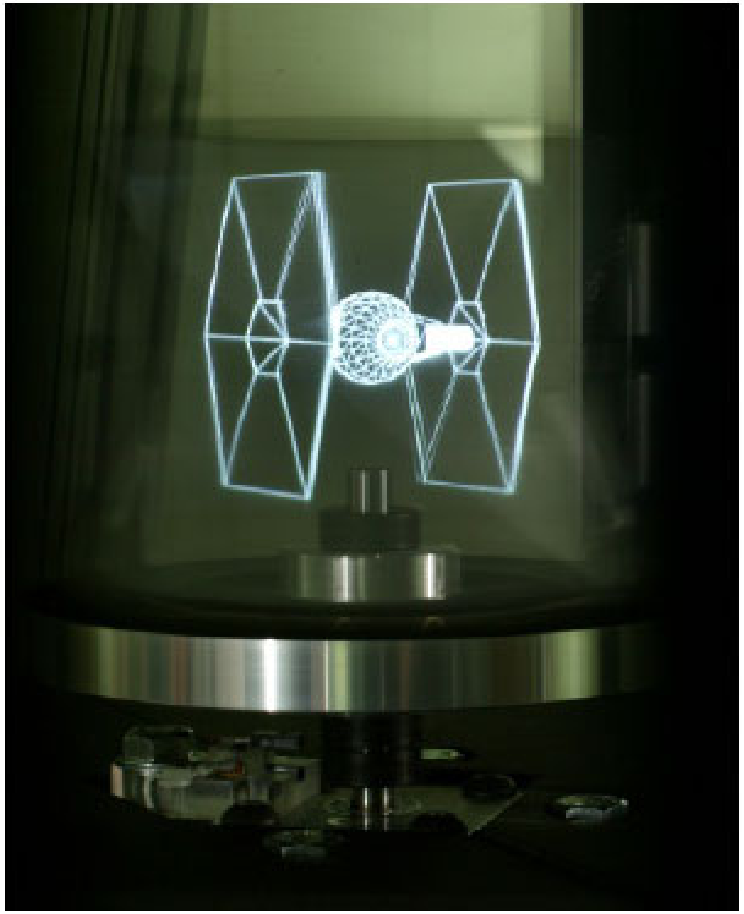


- Laser emitters
- Virtual Retinal Display, VRD http://www.hitl.washington.edu/projects/vrd/project.html

According to number of screens

Where are the L/R images projected onto?

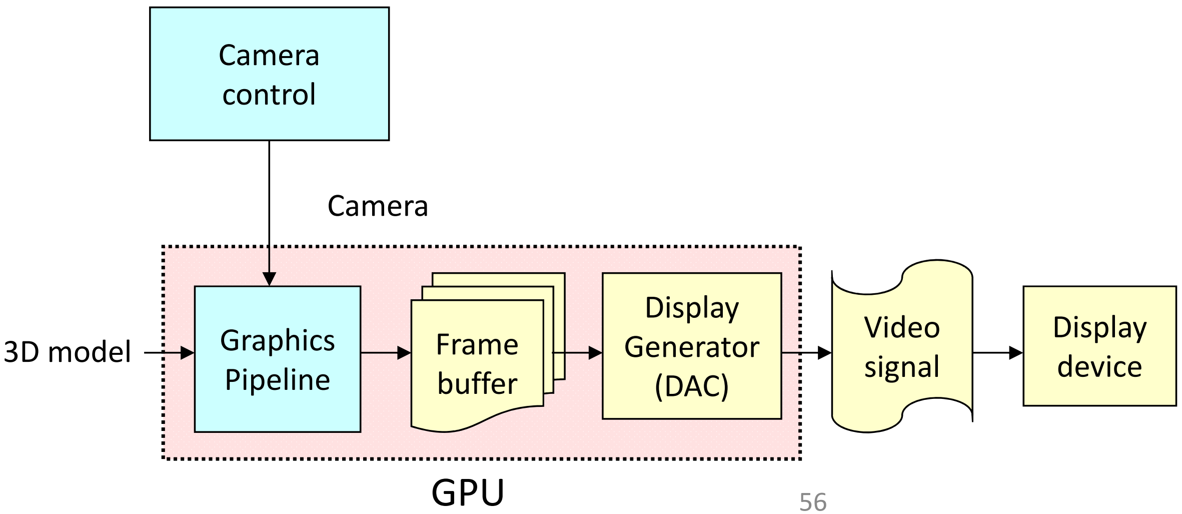
- Two independent screen regions (eg HMD)
- A single screen (3D TV, 3D cinema, CAVE…)
- Directly onto the retina (no screen)
- **Spinning screen (holographic displays)**
  - Lab prototypes
  - Example: spinning mirror with a high-speed projector
    https://www.youtube.com/watch?v=8gvPS1m40gw



SYNTHESIS OF STEREO IMAGES

Synthesis of non-stereo images

- Input: 3D model, tracking data, display system data
- Output: images with retinal disparity



Synthesis of stereo images

- Input: 3D model, tracking data, display system data
- Output: images with retinal disparity

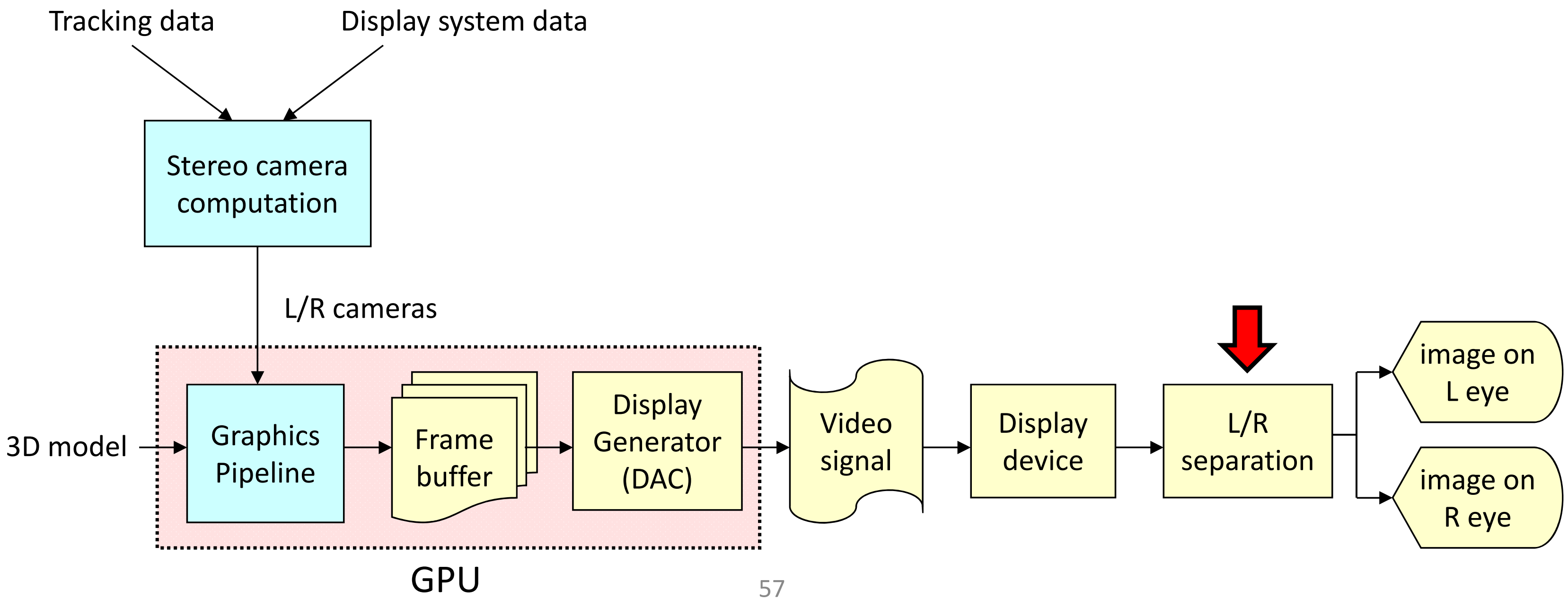


IMAGE SEPARATION

Image separation

- Separation = means for providing each eye with its needed image (and rejecting the unnecessary image).
- Required when a single screen surface contains both images.

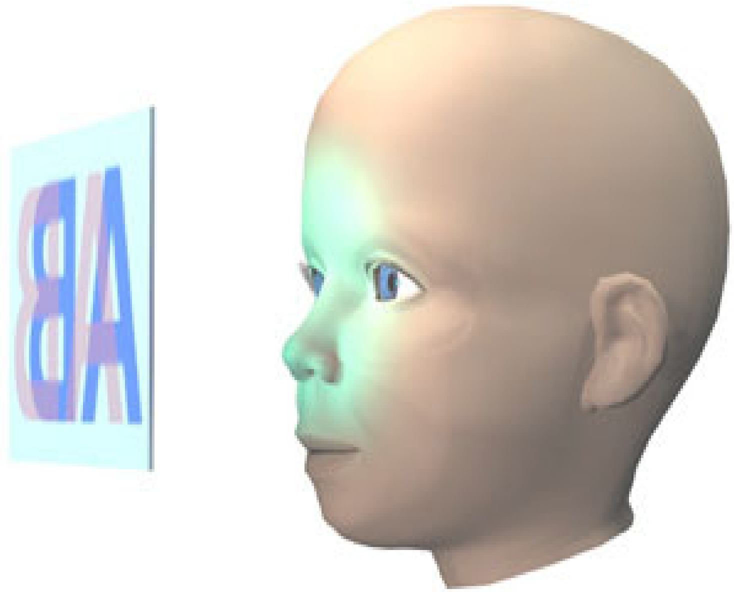


Image separation

Classification

- Autostereoscopic displays
- Glasses
  - Anaglyph glasses
  - Infitec glasses
  - Polarizing glasses
  - Shutter-glasses

Passive stereo

Active stereo

Image separation

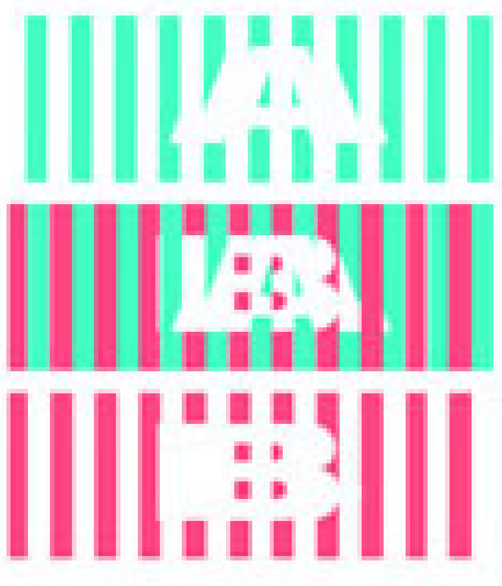
Classification

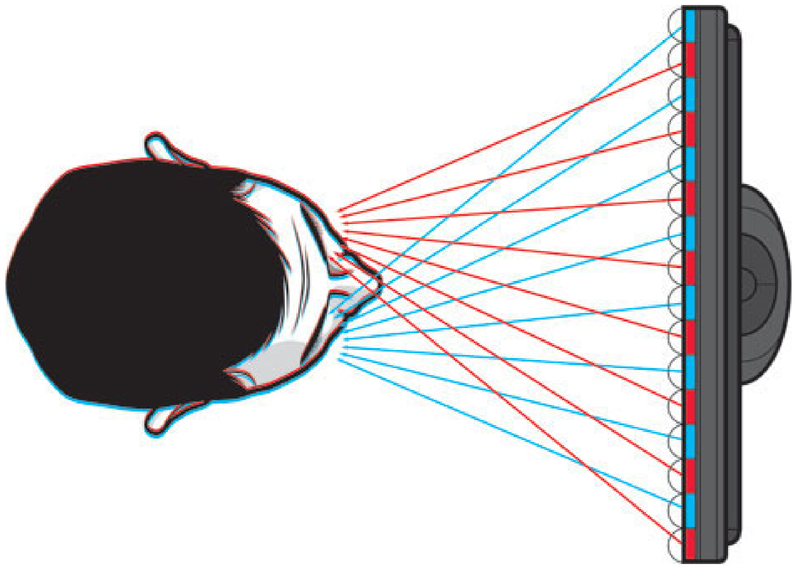
- **Autostereoscopic displays**
- Glasses
  - Anaglyph glasses
  - Infitec glasses
  - Polarizing glasses
  - Shutter-glasses

Passive stereo

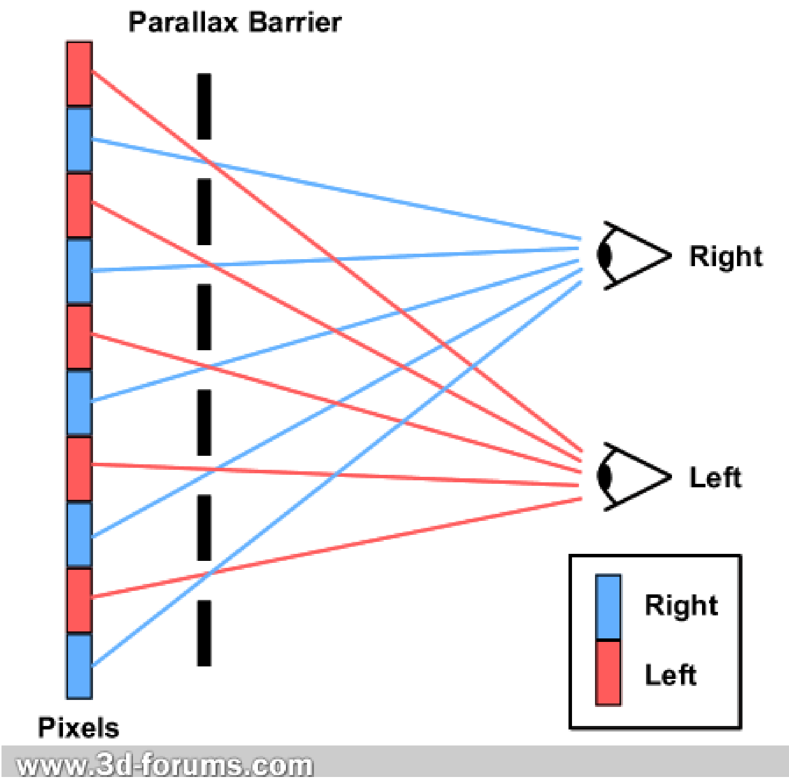
Active stereo

Autostereoscopic displays





Autostereoscopic displays



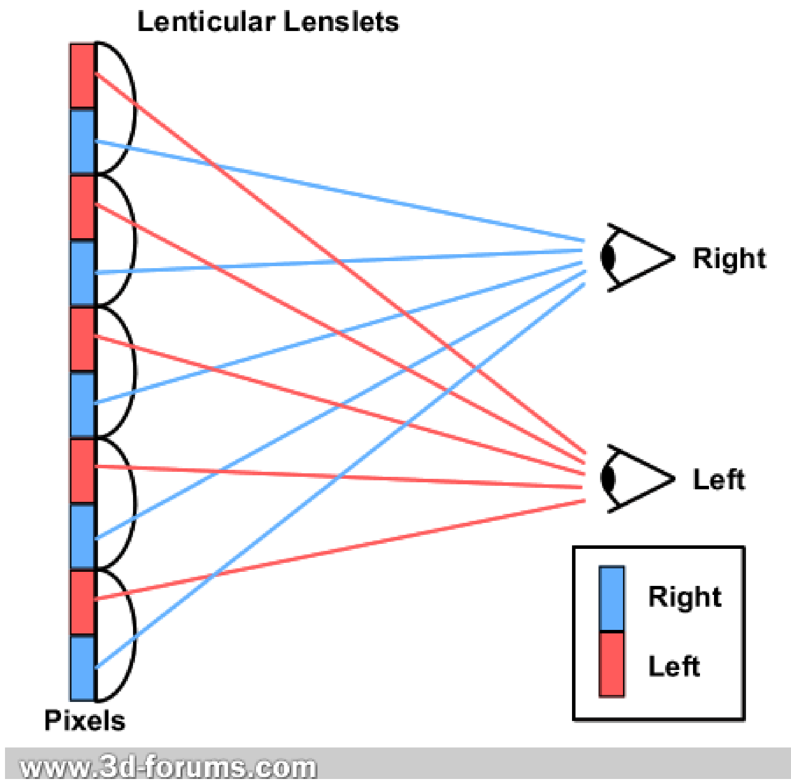


Image separation

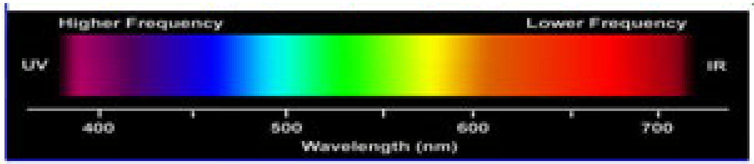
Classification

- Autostereoscopic displays
- Glasses
  - **Anaglyph glasses**
  - Infitec glasses
  - Polarizing glasses
  - Shutter-glasses

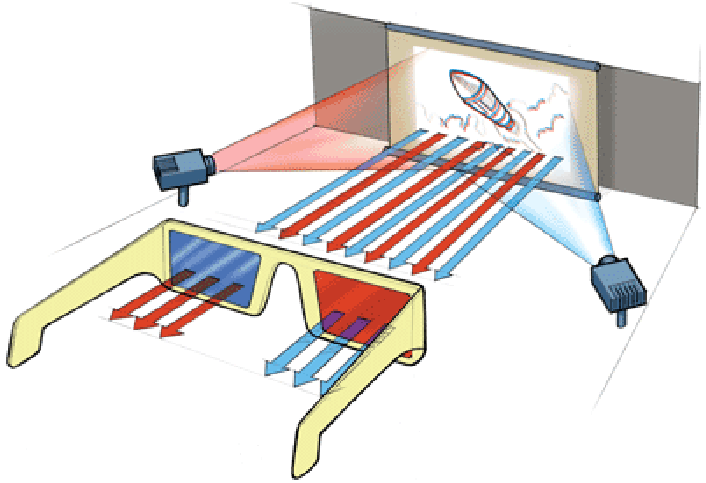
Passive stereo

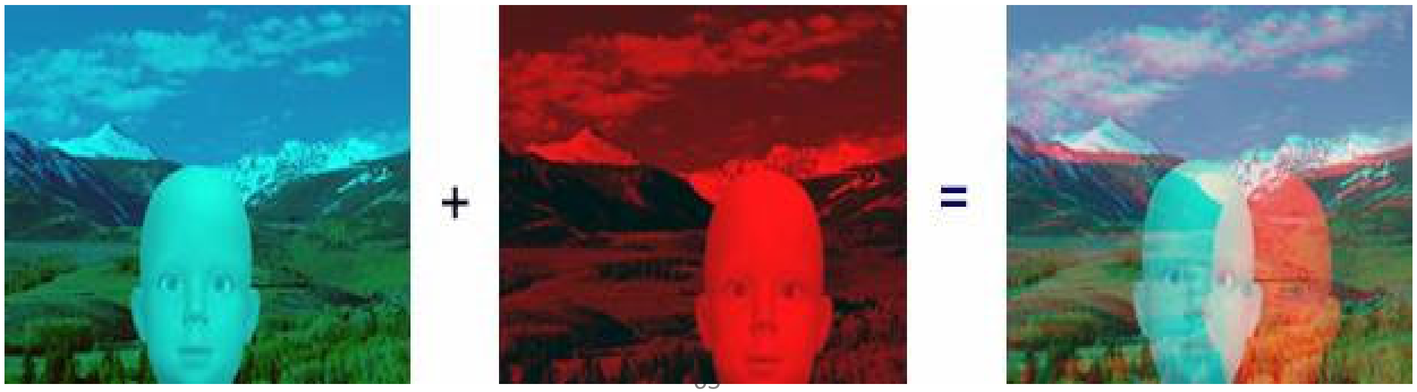
Active stereo

Anaglyph filters



Visible Light





Anaglyph filters

- Based on complementary colors:
  - R-GB, G-RB, B-RG
- Example: red-cyan, left filter blocks G and B; leaves R.
- Cheap
- Obvious color problems

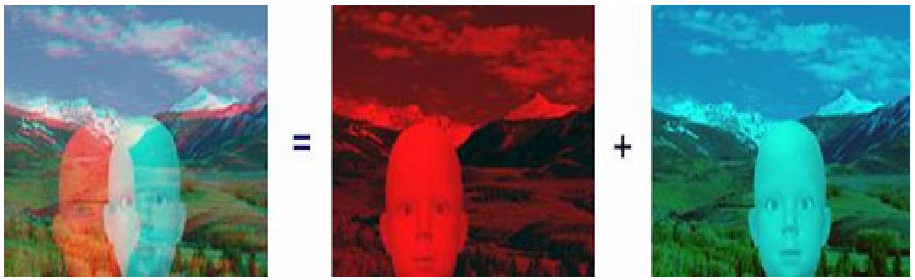


Image separation

Classification

- Autostereoscopic displays
- Glasses
  - Anaglyph glasses
  - **Infitec glasses**
  - Polarizing glasses
  - Shutter-glasses

Passive stereo

Active stereo

Infitec glasses (Dolby 3D digital cinema)

Improved version of anaglyph glasses

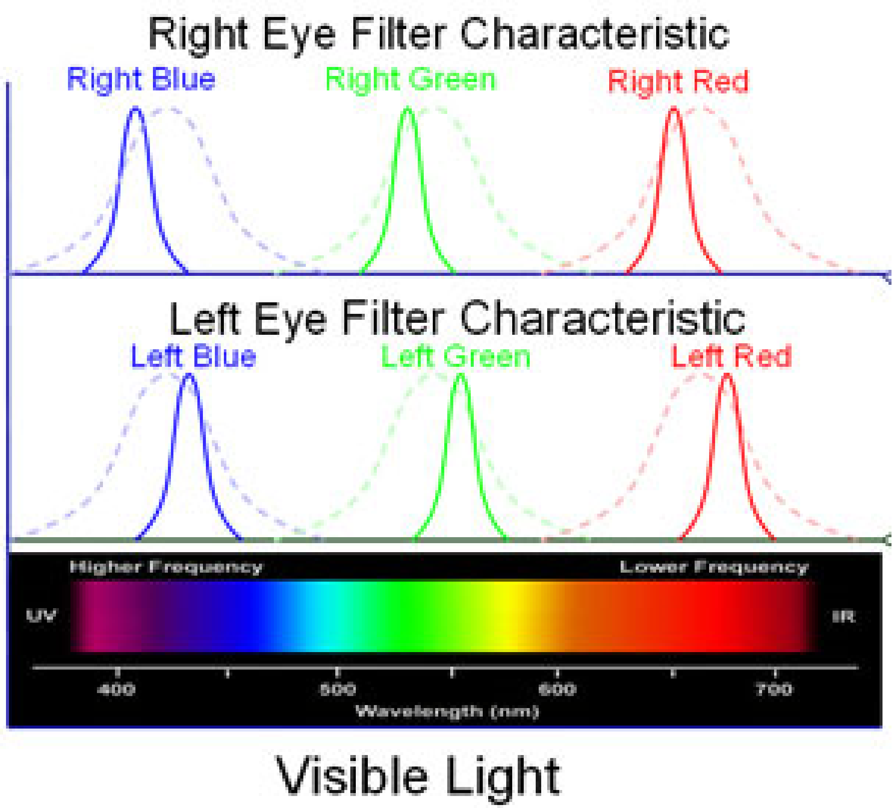


Image separation

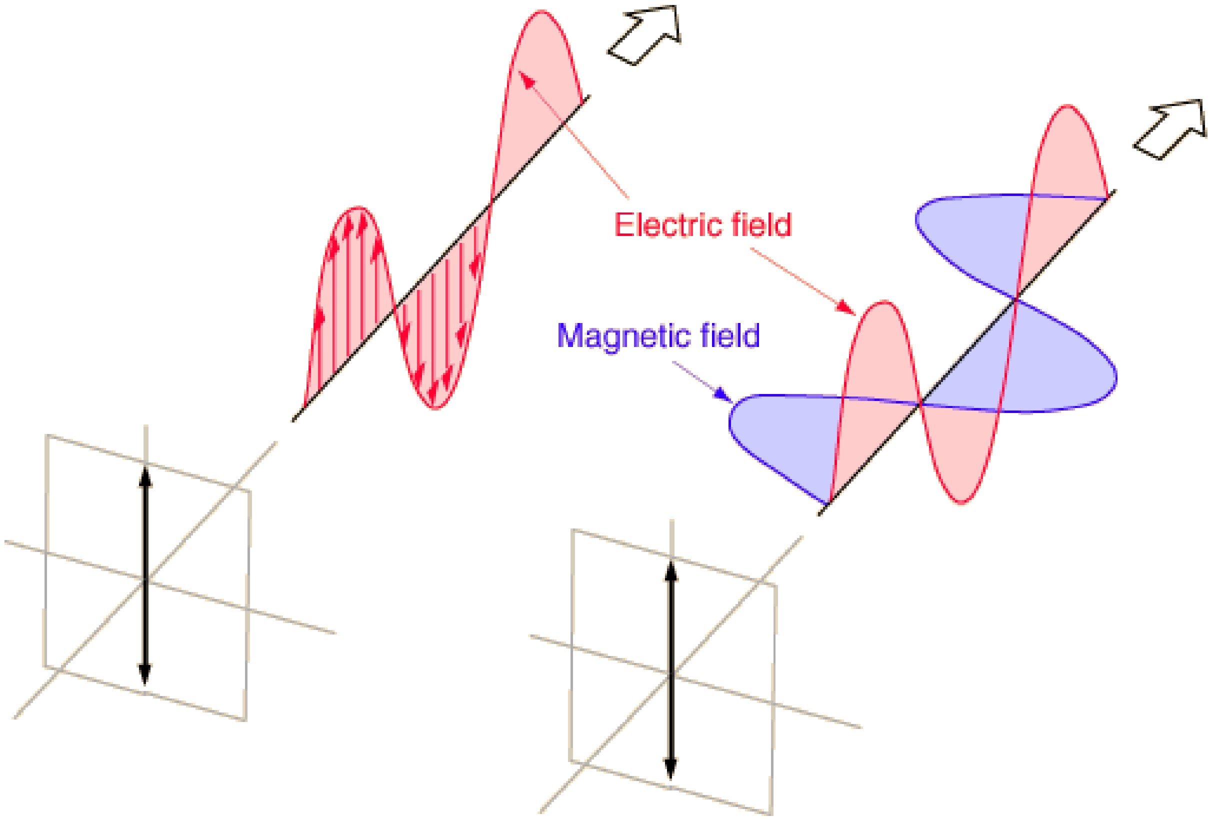
Classification

- Autostereoscopic displays
- Glasses
  - Anaglyph glasses
  - Infitec glasses
  - **Polarizing glasses**
  - Shutter-glasses

Passive stereo

Active stereo

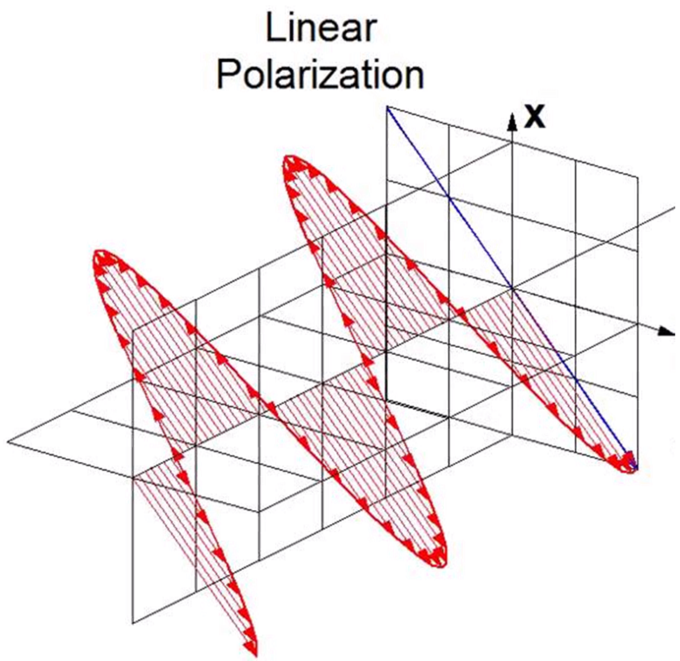
Polarization



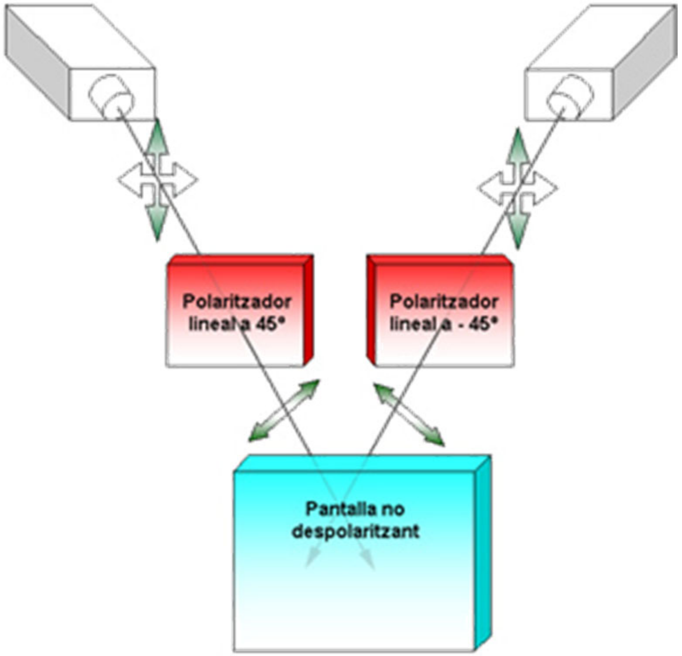
See this video: https://www.youtube.com/watch?v=8YkfEft4p-w

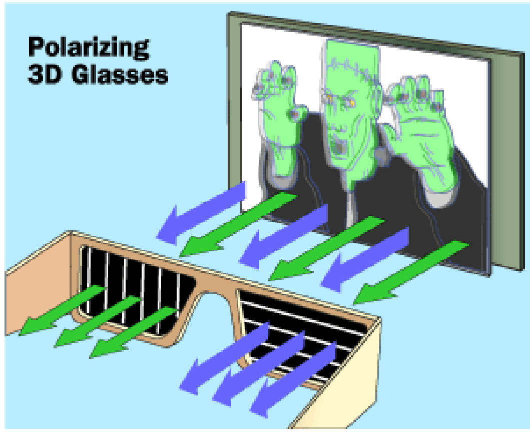
Linear polarizing glasses

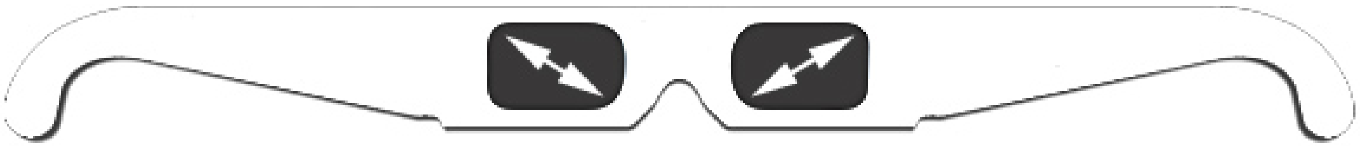
- Based on linear polarization filters
  - In the projectors
  - In the glasses
- Oriented at 90º
- Cheap
- Sensible to head tilt



Linear polarizing glasses







Circular polarizing glasses

- Based on circular polarization filters (projectors & glasses)
- Cheap
- Almost head-tilt insensitive

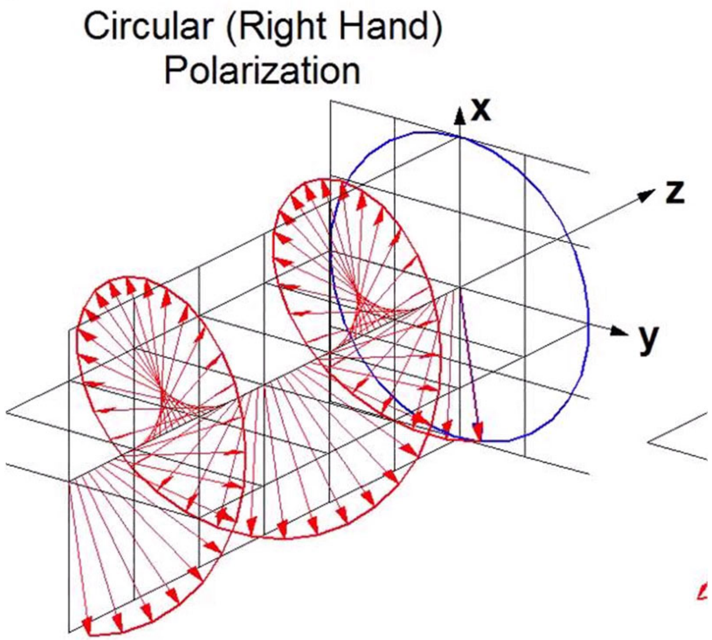


Image separation

Classification

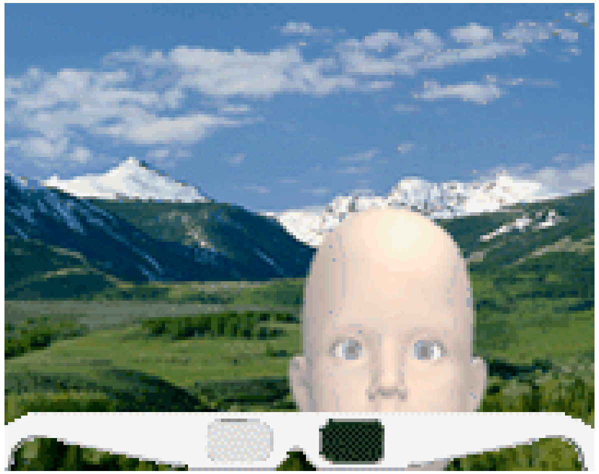
- Autostereoscopic displays
- Glasses
  - Anaglyph glasses
  - Infitec glasses
  - Polarizing glasses
  - **Shutter-glasses**

Passive stereo

Active stereo

Shutter glasses

- LC shutter blocks light, sync with the video signal.
- In any given time, one filter is in *transparent state* and the other in *opaque state*.
- Requires a display running at >100 Hz
- Sync types:
  - Infrarred signal
  - Wired



Shutter glasses

Sync source:

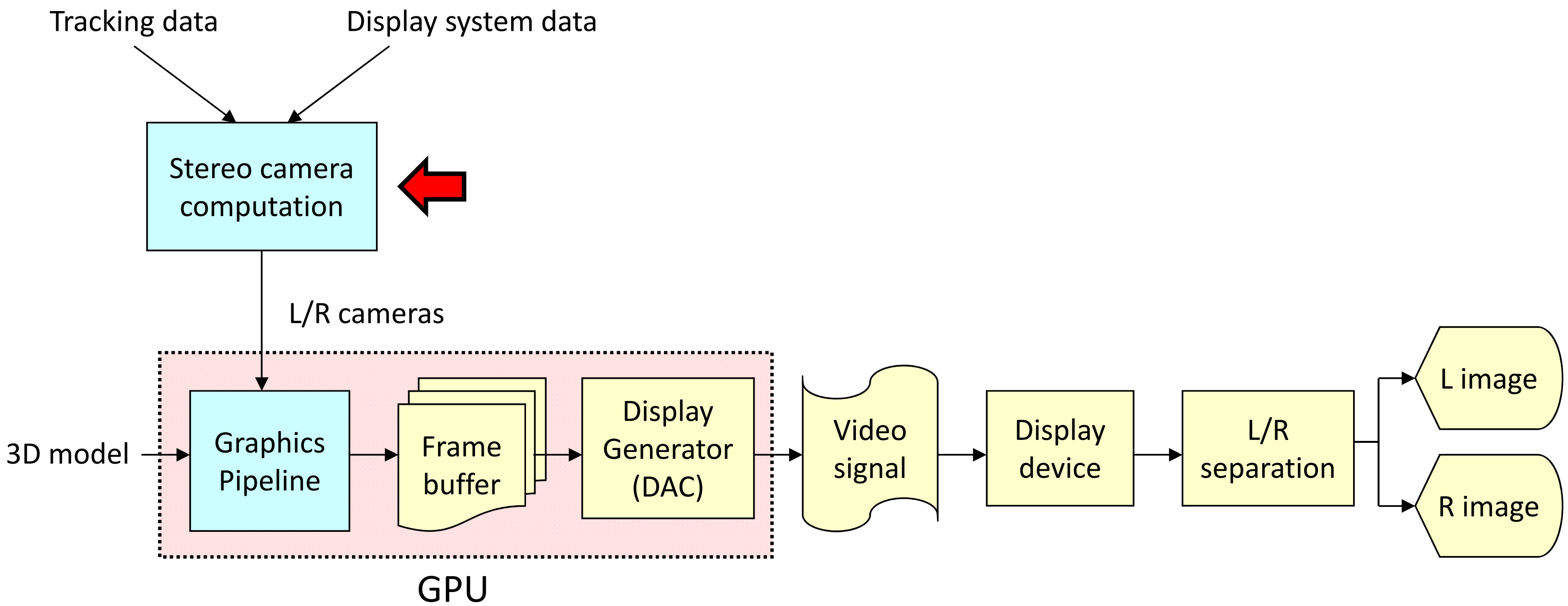
- Stereo-readycards
  - Dedicated signal (eg VESA mini DIN-3)
- Non-stereo-ready cards:
  - VGA connector (pass-through)

Comparison

Technology	Glasses	Sensitive to…		Glasses Cost	Suitable for
		Head Pos	Head Tilt		
Lenticular	No glasses	Y	Y	-	TV, Nintendo-like
Anaglyph	Red/Cyan	N	N	Low	Photos
Infitec	Infitec	N	N	High	VR, cinema
Passive stereo	Polarizing glasses	N	Y	Mid	VR, TV, cinema
Active stereo	Shutter glasses	N	N	High	VR, TV

STEREO CAMERA COMPUTATION

Synthesis of stereo images



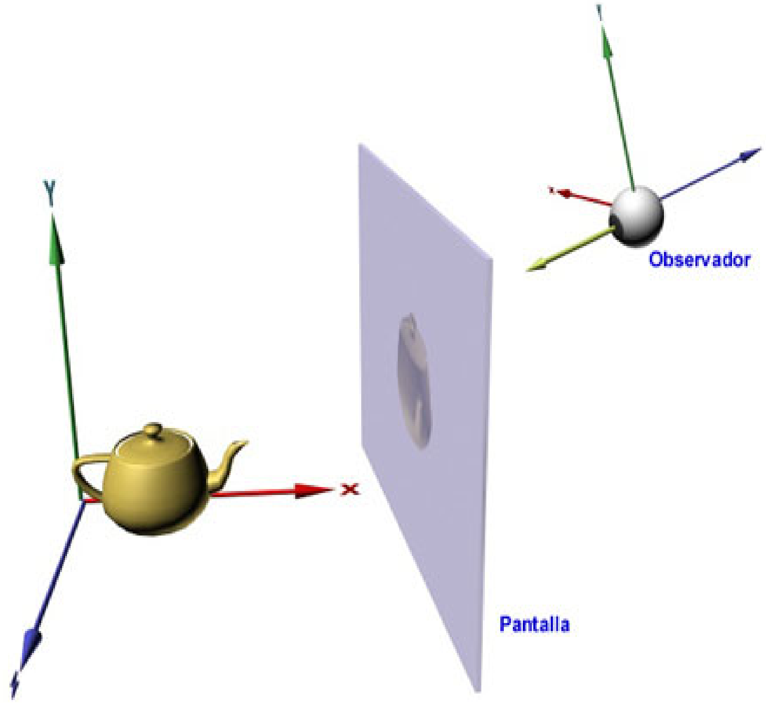
Stereo camera computation

Output: Left and Right cameras:

- Extrinsic parameters: Eye (OBS), target (VRP), up (VUV) → lookAt(eye.x, eye.y, eye.z, target.x, target.y, target.z, up.x, up.y, up.z);
- Intrinsic parameters: view frustum geometry → frustum(left, right, bottom, top, near, far);

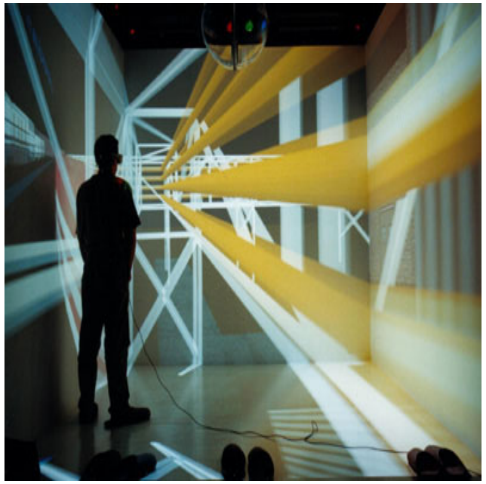
Stereo rendering

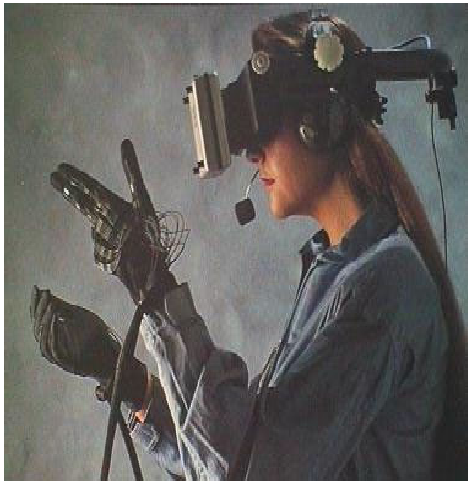
- The virtual camera must be computed taking into account:
  - Screen geometry (size, position, orientation)
  - The eye position with respect to the screen.
- This is absolutely required:
  - On tiled-displays
  - On multi-screen displays
  - On head-tracked displays



Configurations

- Projection-based VR (+ head-tracking)
- VR monitor (without head tracking)
- Head-Mounted displays





Projection-based systems

Applies to CAVEs, Stereowalls…

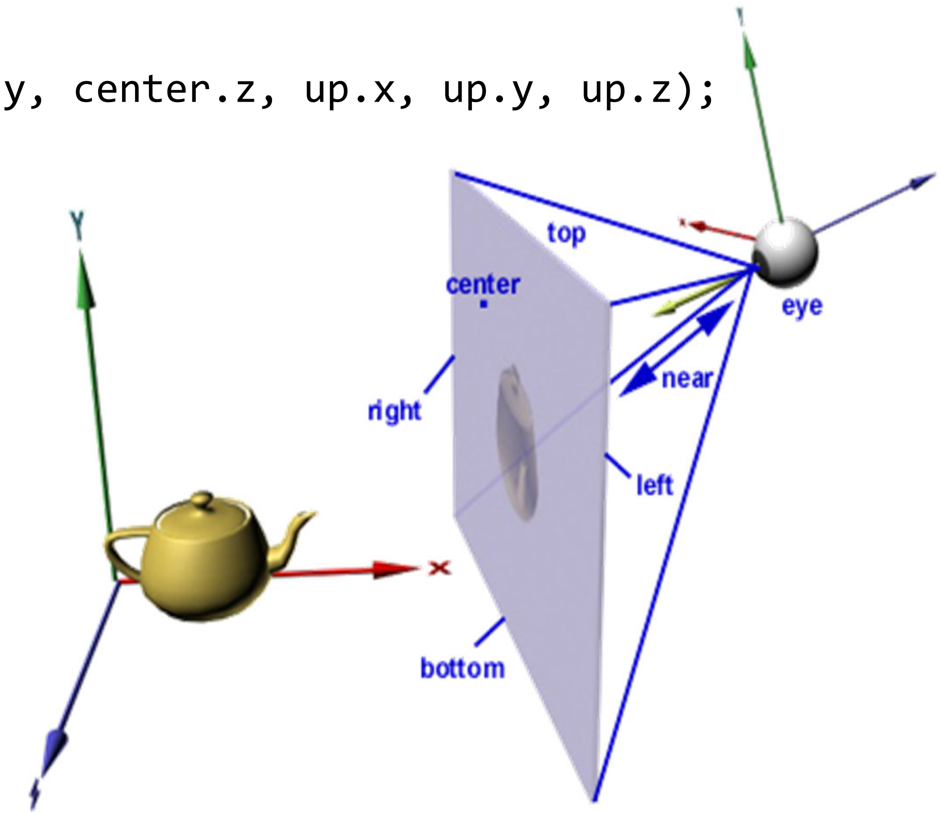
Parameters:

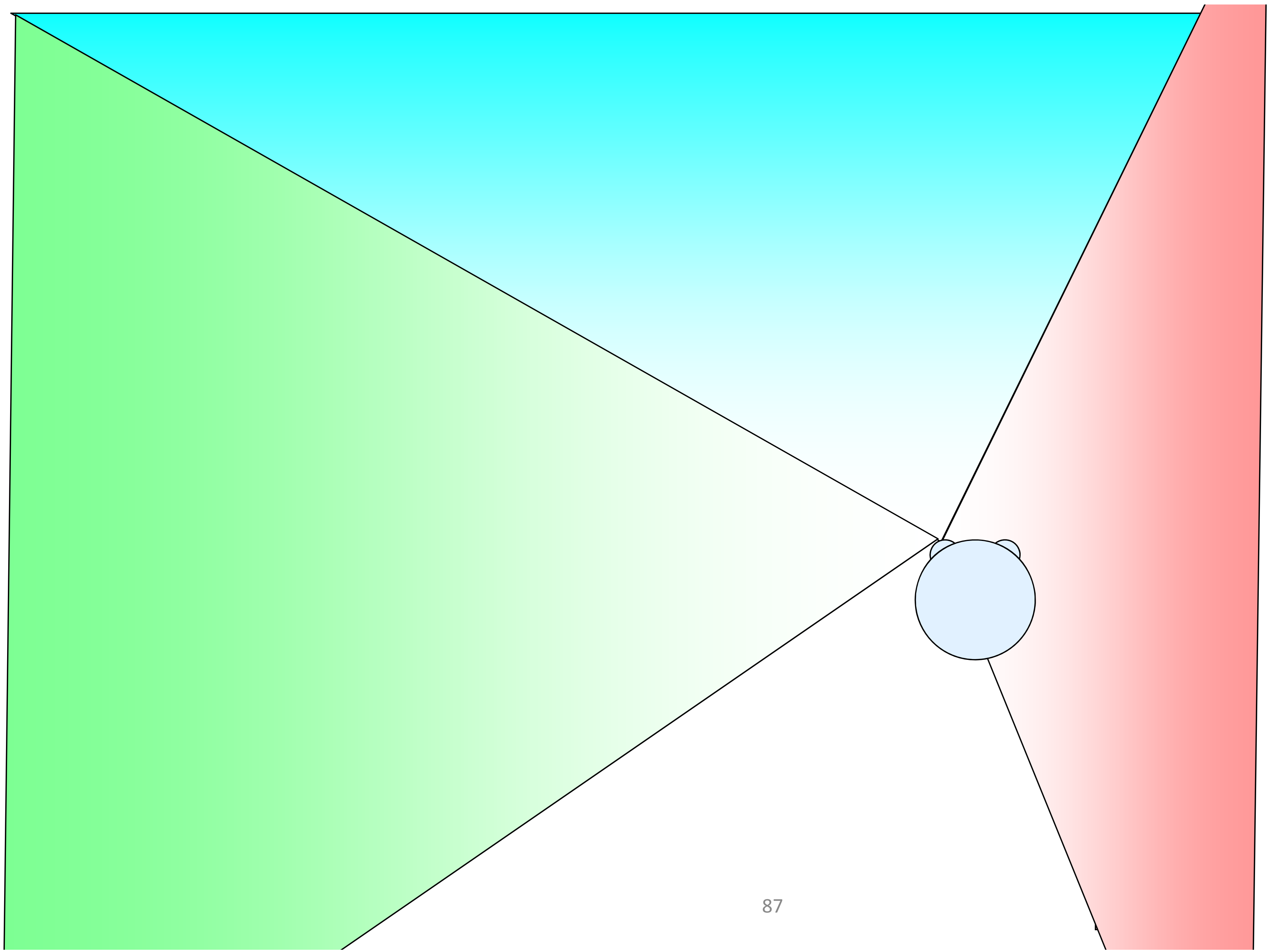
- Tracking data: L/R eye position
  - Two position trackers (3DOF each)
  - One 6DOF tracker (head, glasses…)
- Display system data
  - Screen geometry

Projection-based systems

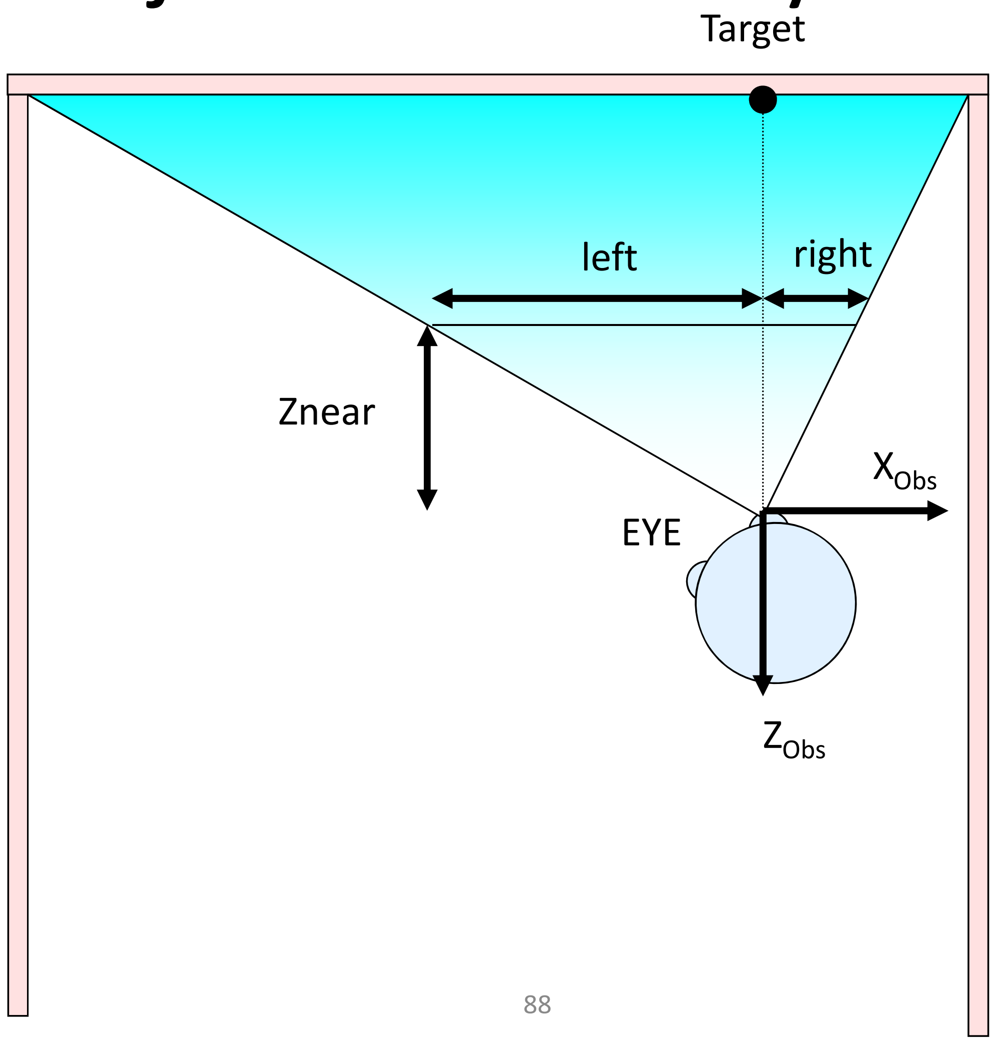
```
// Using OpenGL Compatibility mode for clarity...
glMatrixMode(GL_MODELVIEW);
glLoadIdentity();
gluLookAt(eye.x, eye.y, eye.z, center.x, center.y, center.z, up.x, up.y, up.z);

glMatrixMode(GL_PROJECTION);
glLoadIdentity();
glFrustum(left, right, bottom, top, near, far);
```

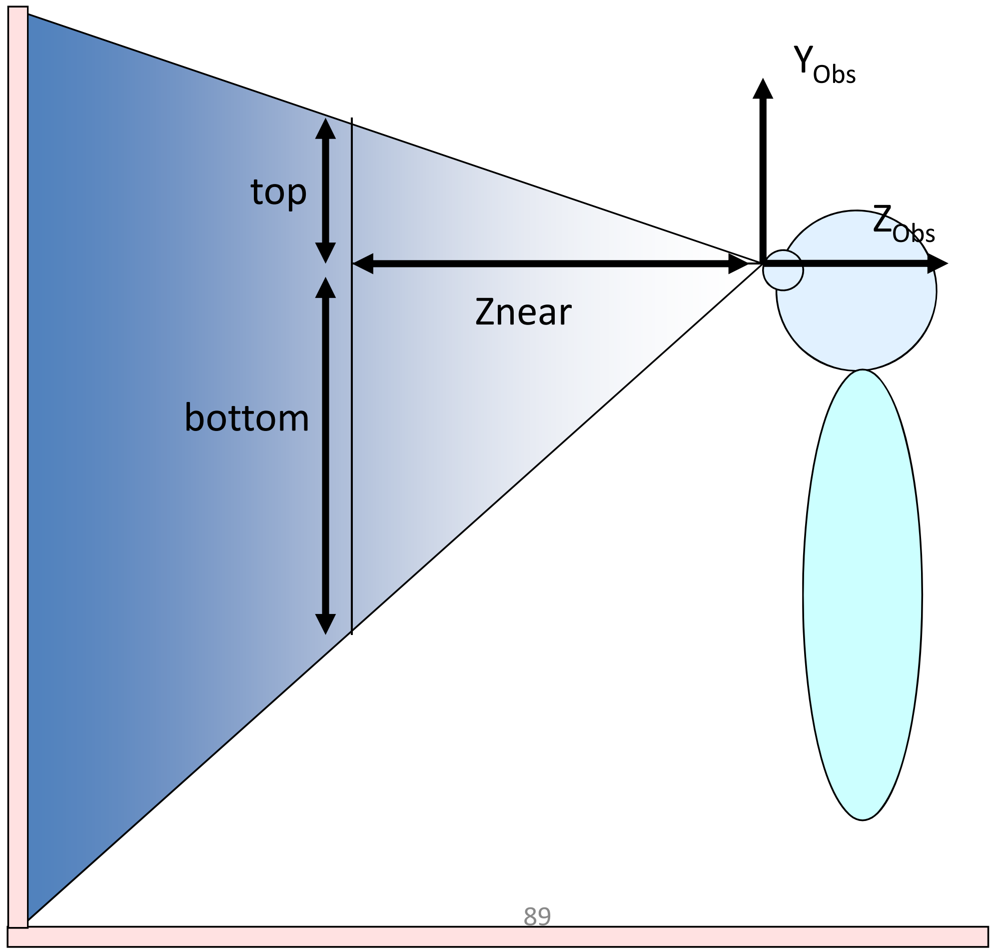




Projection-based systems

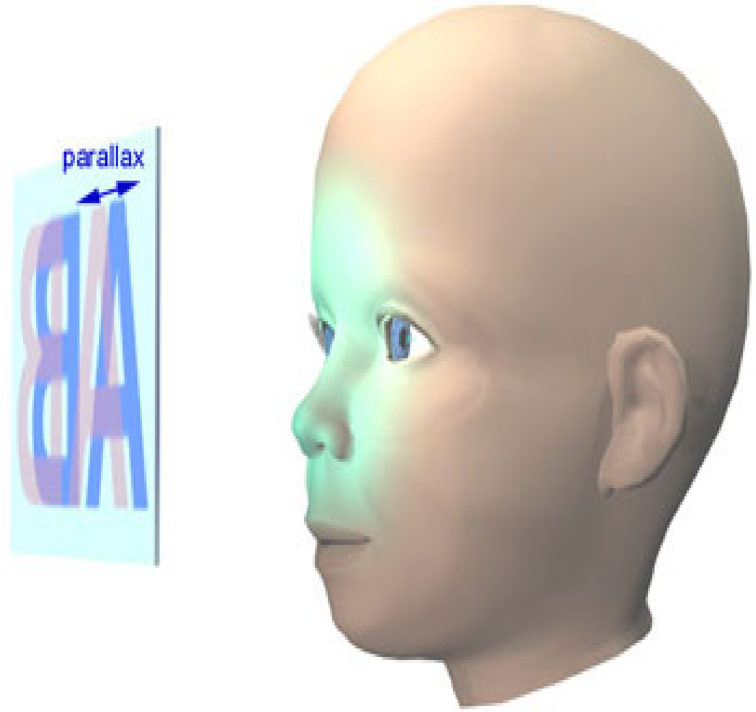


Projection-based systems



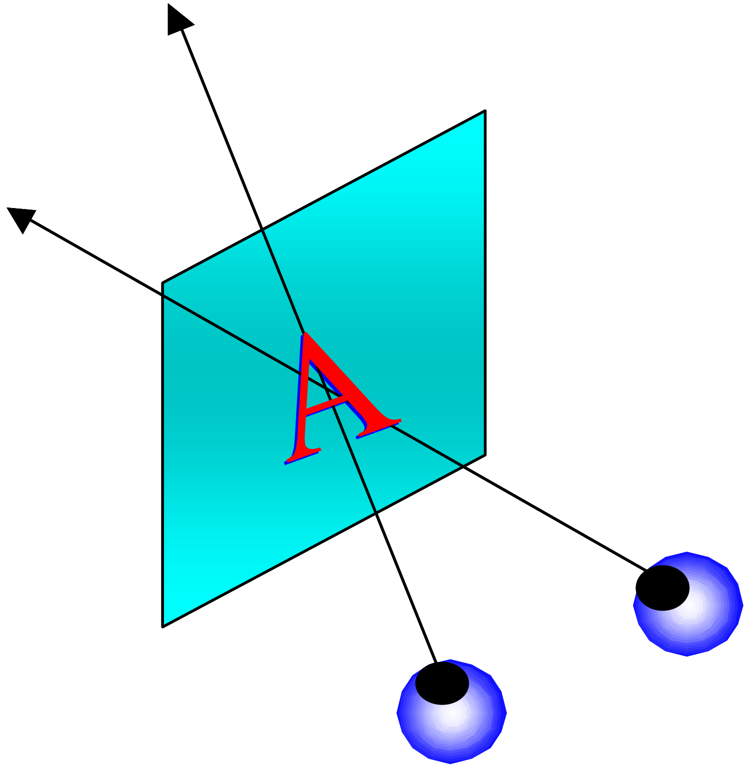
Parallax

- **Parallax** is a 3D stereo concept.
- Related to (but distinct from!) **retinal disparity**.
- **Parallax**: distance between L/R points **measured on the screen**.
- Unlike retinal disparity, **parallax does not depend on eye convergence**.



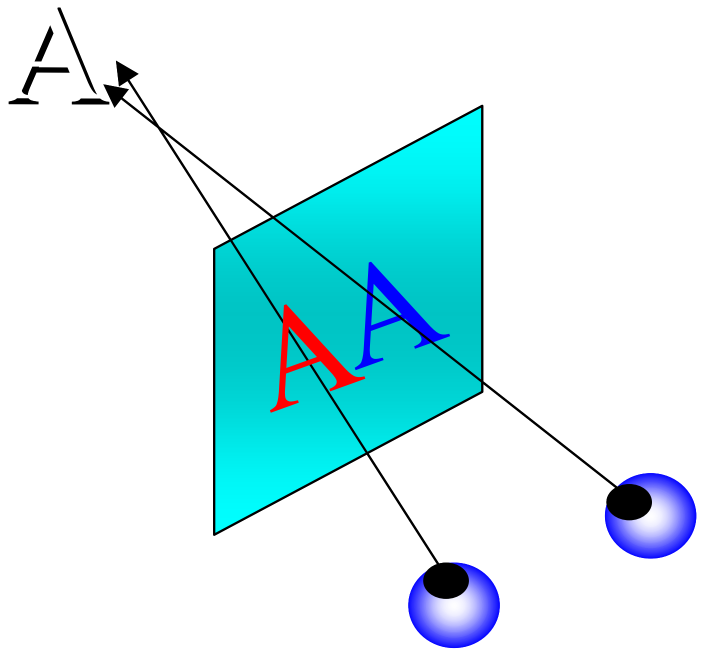
Zero parallax

- An object has zero parallax when the corresponding L/R points overlap.
- When looking at the object, the eyes converge in the screen plane.
- The object appears to be **on the screen plane**.



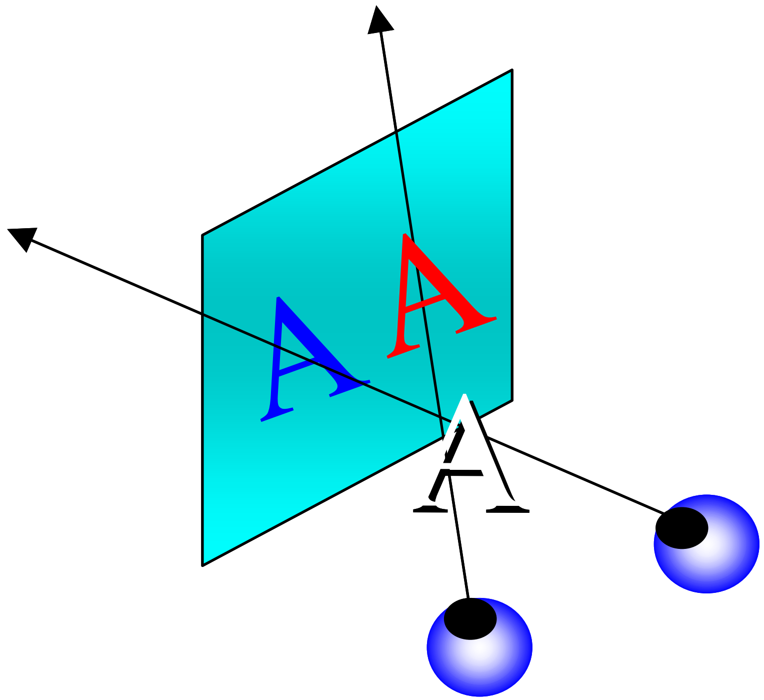
Positive parallax

- An object has **positive parallax** when the corresponding L/R points have positive distance (**L point is to the left** of the R point).
- When looking at the object, the eyes converge at a point **behind the screen plane**.
- The object appears to be behind the screen.
- It should be in [0, 6.3cm]
- If stereo cameras are computed correctly using actual eye positions, parallax wont be greater than the IOD.



Negative parallax

- An object has **negative parallax** when the corresponding L/R points have negative distance (**L point is to the right** of the R point).
- When looking at the object, the eyes converge at a point **in front of the screen plane**.
- The object appears to be **in front of the screen**.
- Some studies recommend not to exceed 1.5 degrees (object is too close).



Desktop system

- Used in low-cost systems (fish-tank VR)
- As in projection-based systems but guessing user’s location.

HMD

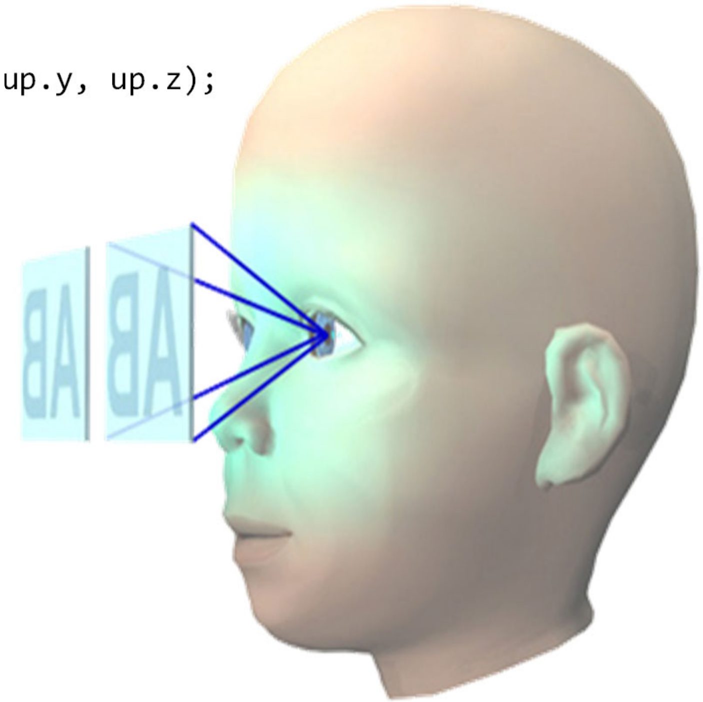
- The screens follow the head movements, so they are fixed with respect to the eyes.
- Parameters:
  - Head orientation
  - Head position (optional)
  - HMD frustum + distortion

HMD

If we could neglect lens distortion:

For each eye

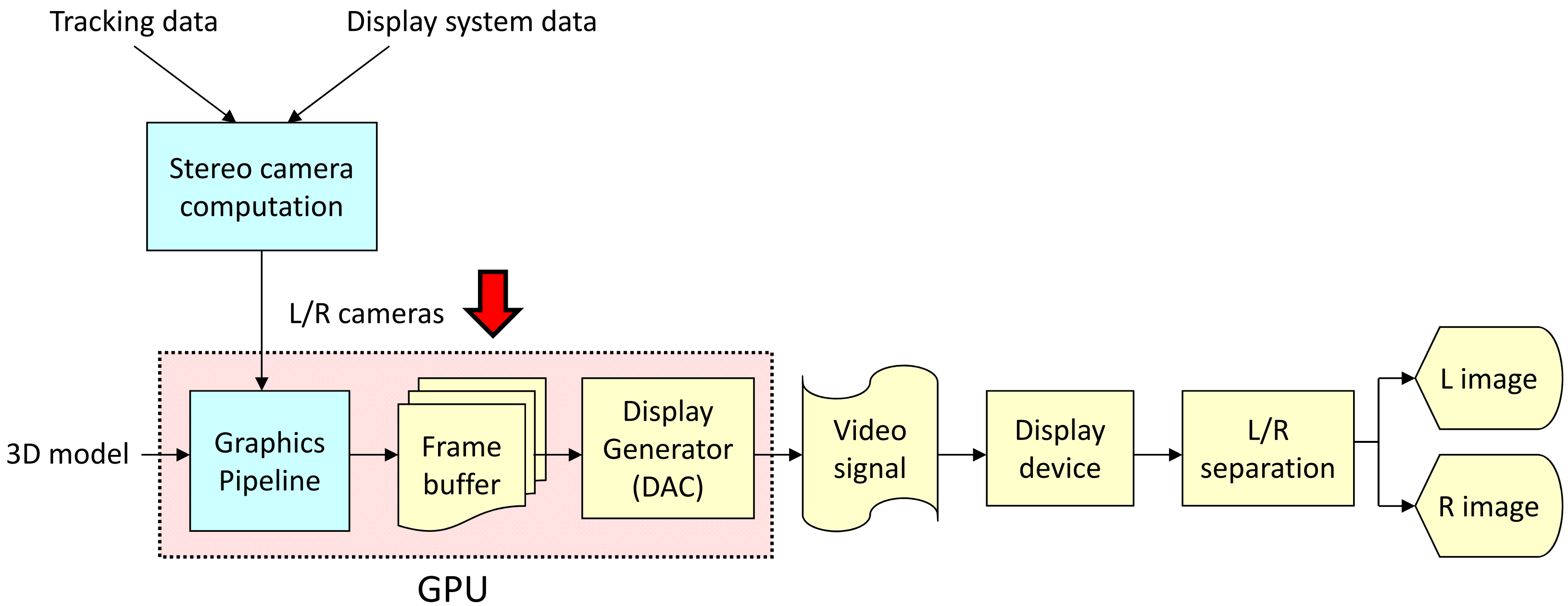
```
glMatrixMode(GL_MODELVIEW);
glLoadIdentity();
gluLookAt(eye.x, eye.y, eye.z, center.x, center.y, center.z, up.x, up.y, up.z);
glMatrixMode(GL_PROJECTION);
glLoadIdentity();
glFrustum(left, right, bottom, top, near, far);
```



HMD

FRAME BUFFER FORMATS (+ RENDERING)

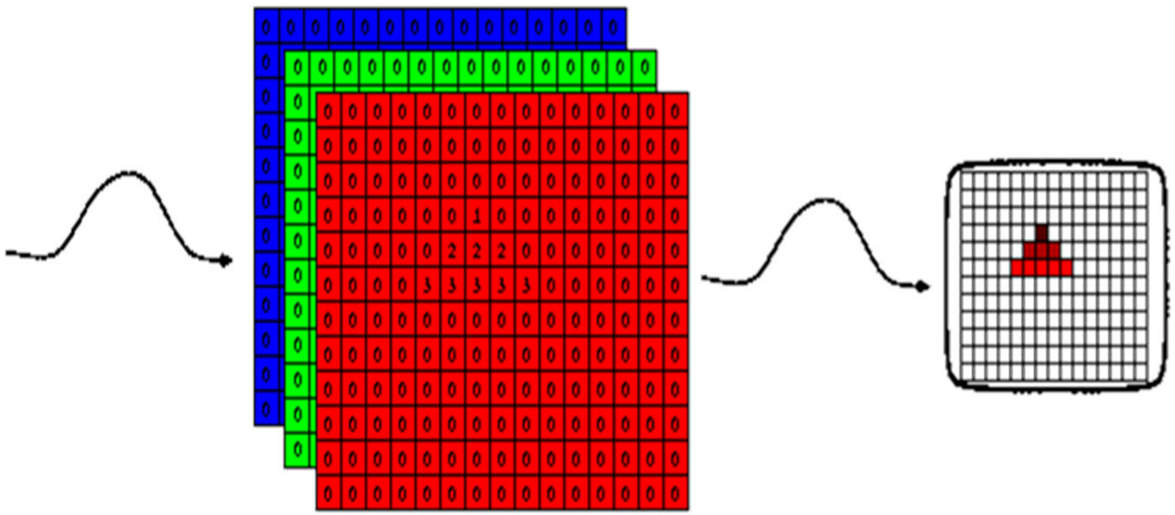
Synthesis of stereo images



Frame buffer: review

Frame buffer:

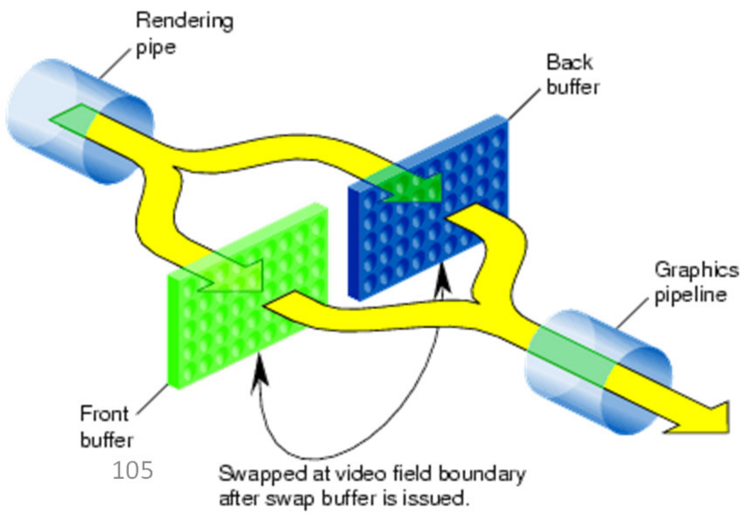
- Color buffer(s)
- Depth buffer
- Stencil buffer



Frame buffer: review

Doble buffering

- Two color buffers: GL_FRONT, GL_BACK
- Depth buffer
- Stencil buffer



Frame buffer formats

- Encoding a stereo pair in the color buffer (required when a single FB has to hold both L/R images)
- Not required e.g. when each image is generated by a different PC.
- Linked with *image separation*

Frame buffer formats

Classification:

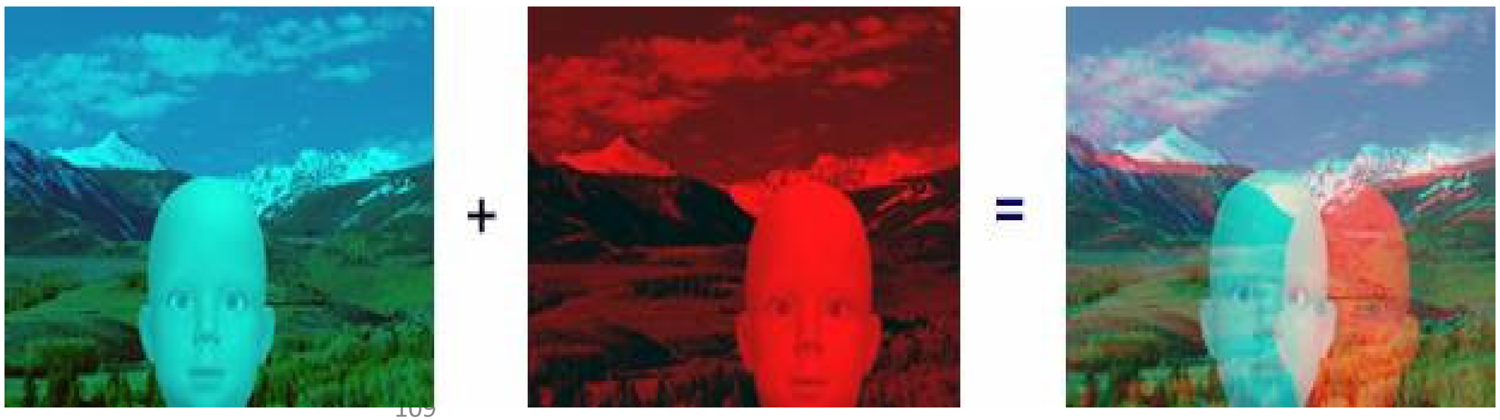
- Color based
  - Anaglyph stereo
- Frame-based:
  - Above-and-below, or Side-by-side
  - Quad buffering
- Line-based and column-based
  - Line-interlived (line sequential), column-interlived

Frame buffer formats

Classification:
- Color based
  - **Anaglyph stereo**
- Frame-based:
  - Above-and-below, or Side-by-side
  - Quad buffering
- Line-based and column-based
  - Line-interlived (line sequential), column-interlived

Anaglyph stereo

```
glColorMask(GL_TRUE, GL_FALSE, GL_FALSE, GL_TRUE);
setupCamera(left);
drawScene();
glColorMask(GL_FALSE, GL_TRUE, GL_TRUE, GL_TRUE);
setupCamera(right);
drawScene();
```

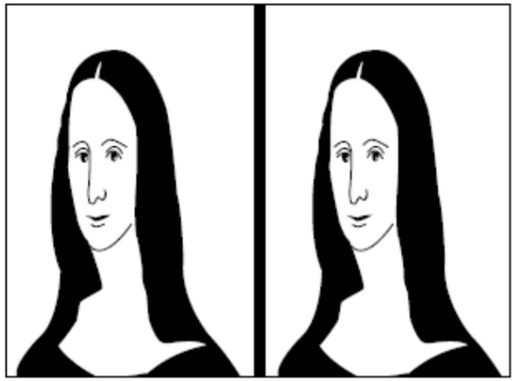


Frame buffer formats

Classification:

- Color based
  - Anaglyph stereo
- Frame-based:
  - Above-and-below, or **Side-by-side**
  - Quad buffering
- Line-based and column-based
  - Line-interlived (line sequential), column-interlived

Side-by-side



- Left half → left image
- Right half → right image
- Example, assuming a 2048x768 desktop:

```
glViewport(0, 0, 1024, 768);
setupCamera(left);
drawScene();
glViewport(1024,0,1024,768);
setupCamera(right);
drawScene();
```

Frame buffer formats

Classification:

- Color based
  - Anaglyph stereo
- Frame-based:
  - Above-and-below, or Side-by-side
  - **Quad buffering**
- Line-based and column-based
  - Line-interlived (line sequential), column-interlived

Quad buffering

Frame buffer

- Color buffers: GL_FRONT_LEFT, GL_FRONT_RIGHT, GL_BACK_LEFT, GL_BACK_RIGHT
- Depth buffer
- Stencil buffer

Quad buffering is used in *active stereo*

- *Why two front buffers?* We need to provide both images during the time the application takes to draw the next frame.
- *Why two back buffers?* Both L/R buffers are swapped simultaneously!
- Supported by *stereo-ready* graphics cards
- Common format when using a single monitor/projector.

Quad buffering: sample code

Setup a stereo format (Qt sample code):

```
QGLFormat f;
f.setStereo(true);
QGLFormat::setDefaultFormat(f);
```

Rendering:

```
glDrawBuffer(GL_BACK_LEFT);
setupCamera(left);
drawScene();
glDrawBuffer(GL_BACK_RIGHT);
setupCamera(right);
drawScene();
```

Frame buffer formats

Classification:

- Color based
  - Anaglyph stereo
- Frame-based:
  - Above-and-below, or Side-by-side
  - Quad buffering
- **Line-based and column-based**
  - Line-interlived (line sequential), column-interlived

Line/column-interlived

- Example:
  - Even lines encode the left image
  - Odd lines encode the right image
- Limitation: each eye sees half the number of lines (or columns).

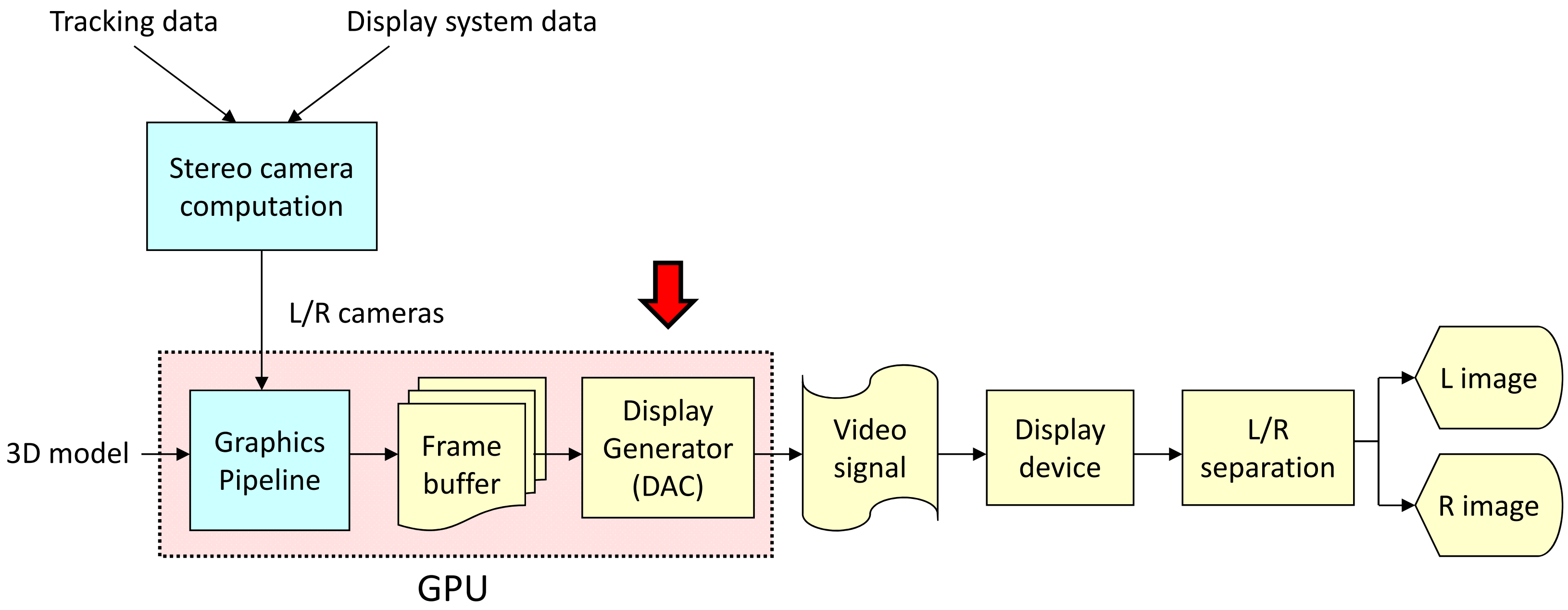
Rendering:

```
shaders.setUniformValue(eye, 0);
setupCamera(left);
drawScene();
shaders.setUniformValue(eye, 1);
setupCamera(right);
drawScene();
```

Fragment Shader:

```
uniform int eye; //   0 = left eye; 1 = right eye
void main()
{
  int parity = int(gl_FragCoord.y) % 2;
  if (parity == eye )  discard;
  …
}
```

Synthesis of stereo images

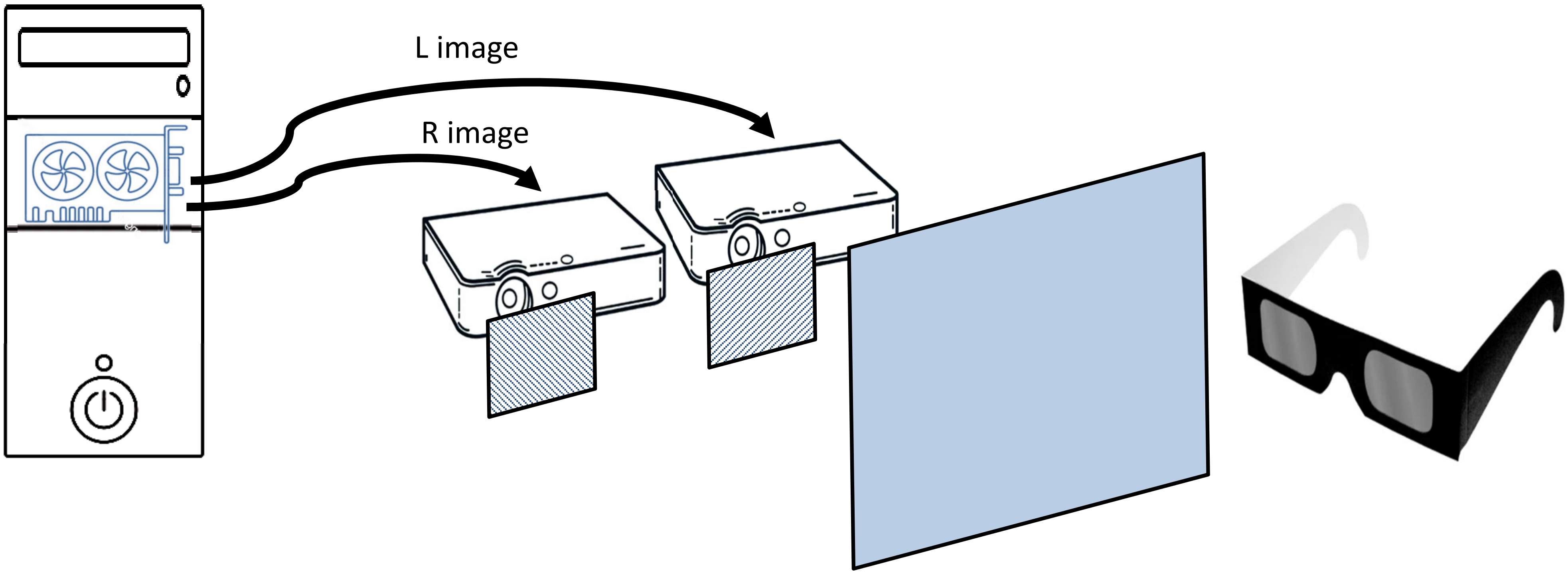


Display generator

- Converts the contents of the (FRONT) color buffer(s) into a video signal.
- Multiple configurations:
  – **Stereo-ready cards**: **one video signal**, encoding sequentially left and right images.
  – **Dual-head PCs** (most current graphics cards): two independent video signals (often DVI+VGA or DVI+DVI connectors)
  – **Quad-head PCs** (high-end graphics cards)

SAMPLE CONFIGURATIONS

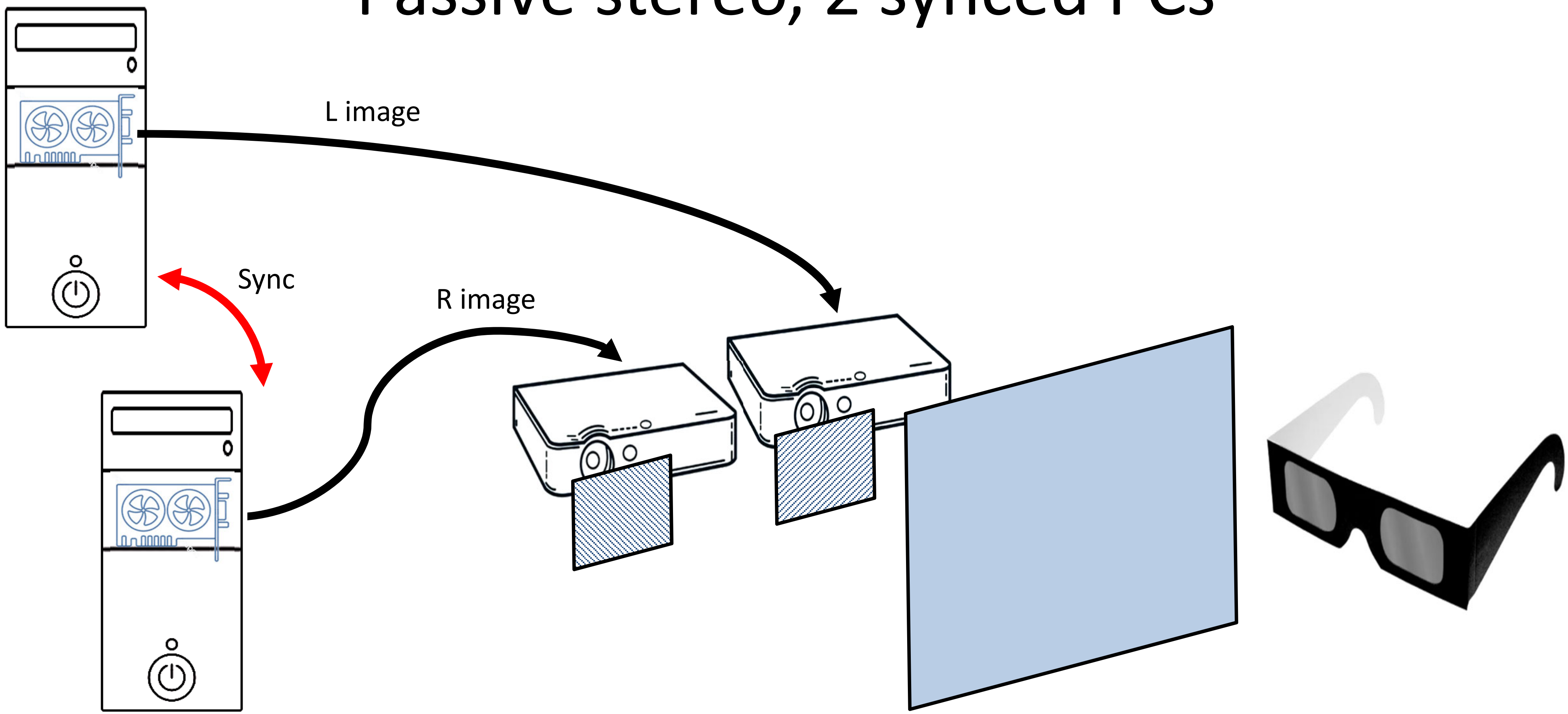
Passive stereo, 1 PC



Sample configurations

# PC	# Gfx cards	FB format	# Projectors	Filter	# Screens	Glasses	Comment
1	**1 (dual-head)**	**Side-by-side**	**2 (passive, 60 Hz)**	**Polarizing filter**	**1**	**Polarizing glasses**	**Passive stereo**
2	2	Normal	2 (passive, 60 Hz)	Polarizing filter	1	Polarizing glasses	Passive stereo; Needs PC sync
1	1	Quad buffering	1 (active, 120Hz)	-	1	Shutter glasses	Active stereo
4	4	Quad buffering	4 (active, 120 Hz)	-	4	Shutter glasses	Active CAVE Needs gfx sync!

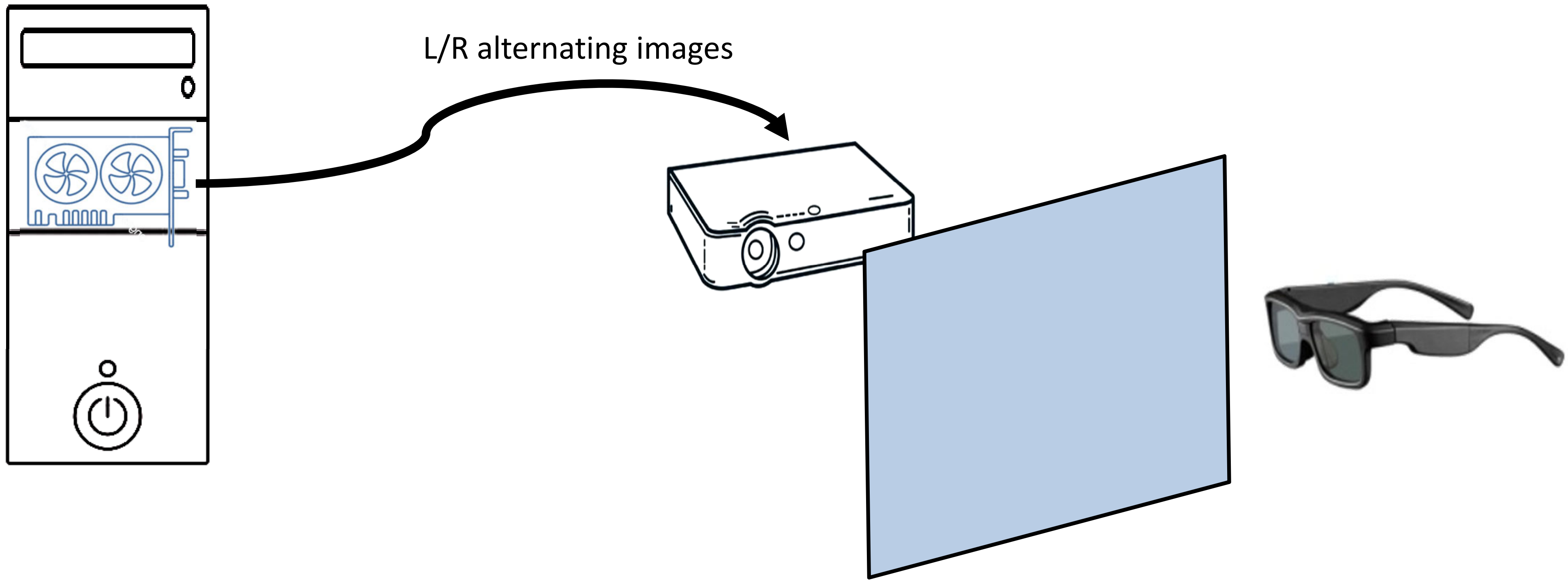
Passive stereo, 2 synced PCs



Sample configurations

# PC	# Gfx cards	FB format	# Projectors	Filter	# Screens	Glasses	Comment
1	1 (dual-head)	Side-by-side	2 (passive, 60 Hz)	Polarizing filter	1	Polarizing glasses	Passive stereo
2	**2**	**Normal**	**2 (passive, 60 Hz)**	**Polarizing filter**	**1**	**Polarizing glasses**	**Passive stereo; Needs PC sync**
1	1	Quad buffering	1 (active, 120Hz)	-	1	Shutter glasses	Active stereo
4	4	Quad buffering	4 (active, 120 Hz)	-	4	Shutter glasses	Active CAVE Needs gfx sync!

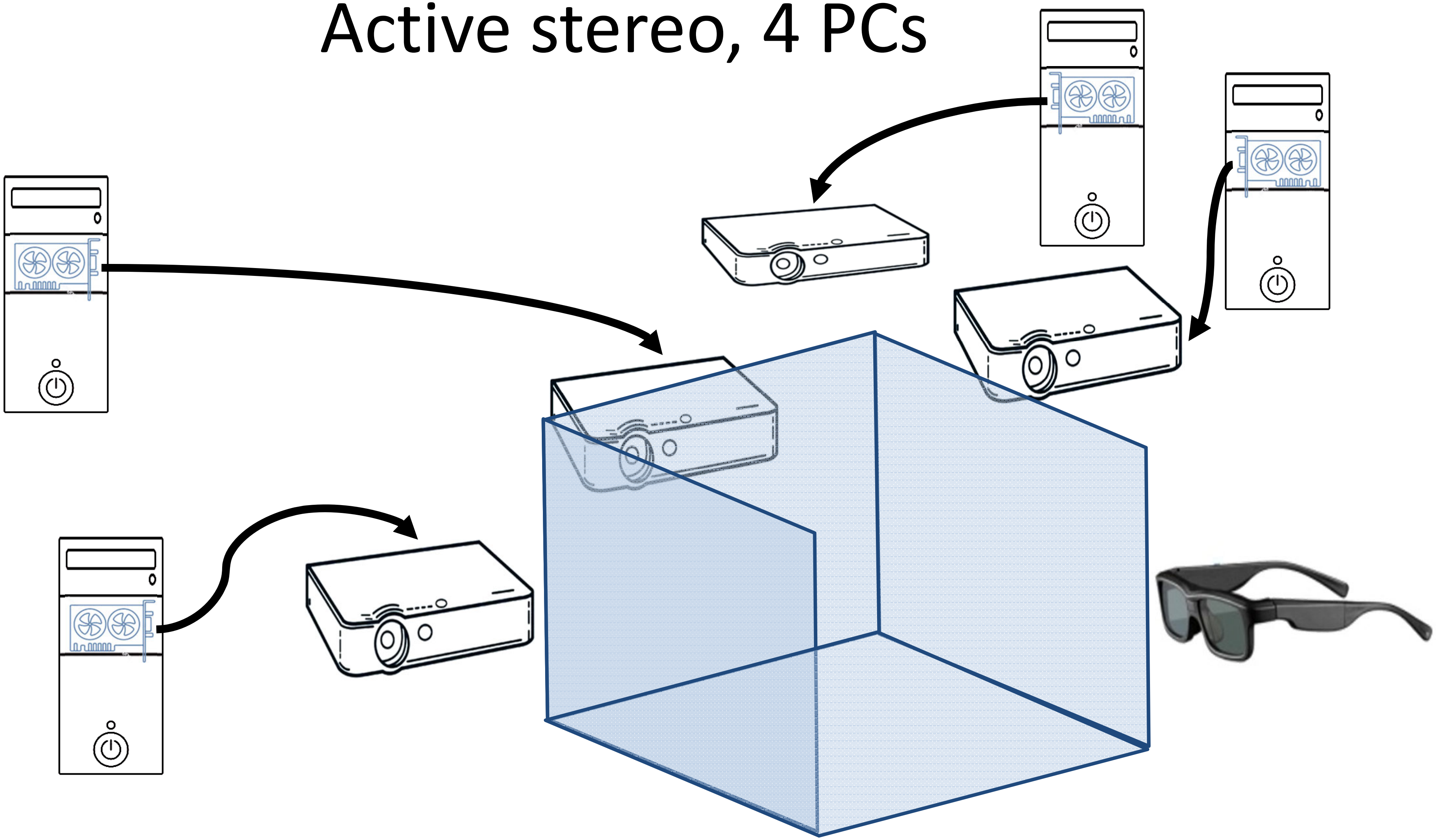
Active stereo, 1 PC



Sample configurations

# PC	# Gfx cards	FB format	# Projectors	Filter	# Screens	Glasses	Comment
1	1 (dual-head)	Side-by-side	2 (passive, 60 Hz)	Polarizing filter	1	Polarizing glasses	Passive stereo
2	2	Normal	2 (passive, 60 Hz)	Polarizing filter	1	Polarizing glasses	Passive stereo; Needs PC sync
1	**1**	**Quad buffering**	**1 (active, 120Hz)**	**-**	**1**	**Shutter glasses**	**Active stereo**
4	4	Quad buffering	4 (active, 120 Hz)	-	4	Shutter glasses	Active CAVE Needs gfx sync!

Active stereo, 4 PCs



Sample configurations

# PC	# Gfx cards	FB format	# Projectors	Filter	# Screens	Glasses	Comment
1	1 (dual-head)	Side-by-side	2 (passive, 60 Hz)	Polarizing filter	1	Polarizing glasses	Passive stereo
2	2	Normal	2 (passive, 60 Hz)	Polarizing filter	1	Polarizing glasses	Passive stereo; Needs PC sync
1	1	Quad buffering	1 (active, 120Hz)	-	1	Shutter glasses	Active stereo
4	**4**	**Quad buffering**	**4 (active, 120 Hz)**	**-**	**4**	**Shutter glasses**	**Active CAVE Needs gfx sync!**