ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Multimedia Technology

Section # 24: Immersive Video

Instructor: George Xylomenos

Department: Informatics

Contents

- 3D video
- 360-degree video
- Volumetric video

3D video

Class: Multimedia Technology, **Section # 24:** Immersive Video

Instructor: George Xylomenos, **Department:** Informatics

Stereopsis (1 of 2)

- Humans see the world in 3D
  - Each eye is slightly offset from the other
  - The brain combines the two images
  - The result is a sense of depth
- The sense of depth depends on distance
  - The views become more similar with distance
    - We only experience depth for nearby objects

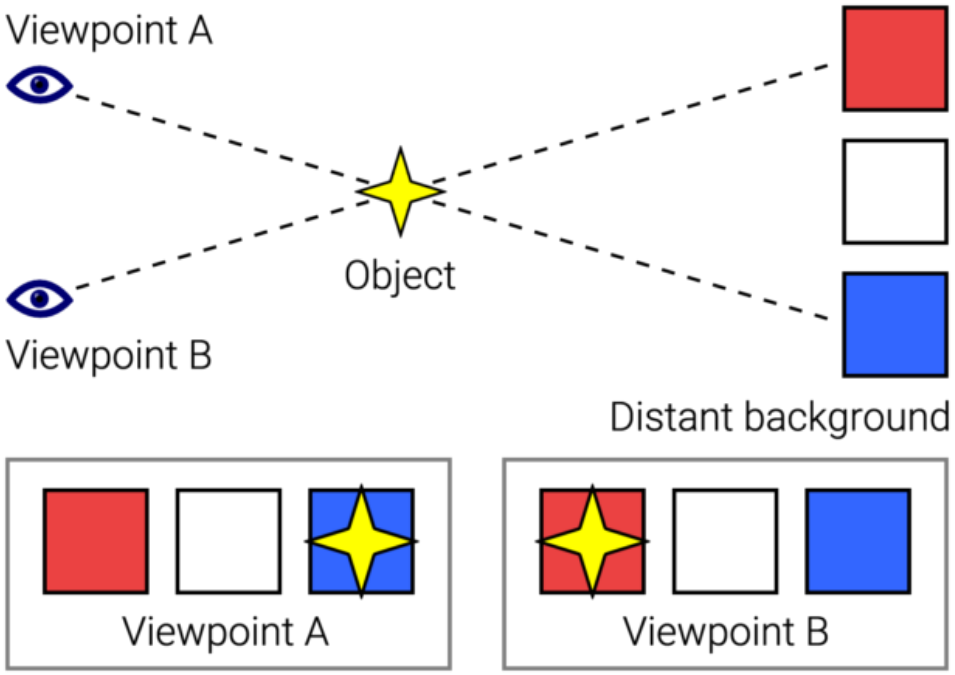
Stereopsis (2 of 2)

- Left and right eye views
  - The left eye shows more detail on the left side
  - The right eye shows more detail on the right side
  - The two images are also slightly offset

Parallax (1 of 2)

- Change in relative object position
  - Say that we look at a nearby vase
  - And the background is a window
  - When we move, we change our viewpoint
    - The nearby object seems to move more
    - The remote object seems to move less
    - Far away objects do not seem to move at all
  - So, viewing position is important!

Parallax (2 of 2)

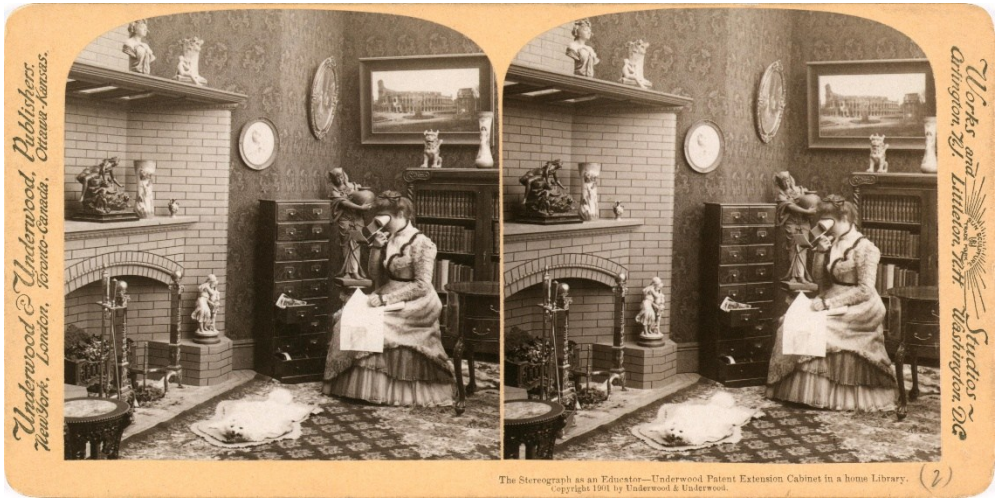


- Parallax contributes to the sense of depth
  - It does not depend on the eyes, though

Anaglyph 3D (1 of 3)

- 3D video capture
  - Requires two cameras at eye-like distances
    - Each camera captures the view for one eye
  - Each view is processed with a filter
  - Initially, two projectors were used
  - Later on, both views printed together
  - Without filters, the image seems doubled
    - With a strange discoloring

Anaglyph 3D (2 of 3)



- Example monochrome image
  - We start with two images (left/right eye)
  - Each one is colored for the filter
    - This is for red-cyan filtering

Anaglyph 3D (3 of 3)

- 3D video viewing
  - Color filter glasses are used
    - Red-Cyan is most common
    - Simple paper glasses with plastic filters
  - Each filter removes one eye view
    - Each eye sees the appropriate view
    - But the views are not perfect
    - Overprinting/projecting loses some detail

Polarized 3D (1 of 2)

- Same basic idea as anaglyph
  - We record two views with two cameras
  - We print them as separate films
  - Each projector uses a different polarizer
    - A filter removes horizontal or vertical rays
  - Glasses with appropriate polarizers used
    - One per eye
  - More expensive than filter, but better quality

Polarized 3D (2 of 2)

- Polarizations must be preserved
  - A special projection screen is needed
    - Regular screens destroy the polarization
- Newer systems use a single film
  - Over-Under: half a frame for each eye
    - Loses some quality, as frames are half sized
  - Alternate frame: each frame is for one eye
    - Needs double the frame rate

Active shutter 3D (1 of 2)

- We start with the same two cameras
- We project frames alternately
  - First for one eye, then for the other
- Active glasses use LCD lenses as shutters
  - When left image shown, only left lens is open
  - When right image shown, only right lens is open
  - Tight synchronization is required!
  - Only reasonable option for TVs

Active shutter 3D (2 of 2)

- Various ways to synchronize
  - Wired or wireless signals
- Costlier, but better than previous techniques
  - You need active glasses
    - With power and sync processors
  - But you can get full light and resolution
  - Provided frame rate is doubled
    - Otherwise, we experience flicker

Head-mounted displays (1 of 2)

- HMDs show one image per eye
  - Each eye sees the output of one camera
  - No need to synchronize
  - Requires two miniature displays
- AR glasses
  - The user sees display plus real world
    - Small screens inside the glasses
  - The system shows regular frames
    - 2D or 3D
  - Goal: augment reality (AR)

Head-mounted displays (2 of 2)

- VR headsets
  - The user sees what is projected
  - The system shows large frames
    - They must cover the entire Field of View (FOV)
  - Goal: create a virtual reality (VR) world
  - Usually, also requires 360-degree views
    - When you move your head, the view must change
    - This is not always the case with AR glasses

3D video coding (1 of 2)

- 2D+Delta coding
  - The two camera views are obviously similar
  - We encode one of the views normally
    - With intra and inter-frame prediction
  - The other view is encoded based on that
    - This is the “+Delta” component
    - The two frames are nearly the same
    - The right eye (say) is predicted from the left

3D video coding (2 of 2)

- H.264 supports 2D+Delta
  - 2D displays can ignore the second view
  - 3D displays can render it in different ways
    - Synchronized shutter or HMD with two displays
- The extension is called H.264/MVC
  - MVC: Multi-view coding
    - It allows more than two cameras
    - Always using the 2D+Delta approach

360-degree video

Class: Multimedia Technology, **Section # 24:** Immersive Video

Instructor: George Xylomenos, **Department:** Informatics

360-degree vs 3D video (1 of 2)

- 3D video provides a specific viewpoint
  - The view of the two capturing cameras
  - More immersive than 2D, but fixed
- 360-degree allows multiple viewpoints
  - It uses an array of cameras
    - Or omnidirectional cameras
  - Captures video in a full 360-degree radius
    - With the user in the center

360-degree vs 3D video (2 of 2)

- Spherical video is 360 in all directions
  - The user is in the middle of the sphere
  - And can change viewpoint in any axis
  - But the viewpoint is always from the center
- 360-degree video may be 2D or 3D
  - Depending on the type of camera used
  - 2D is more common for practical reasons
    - Bandwidth and quality issues

Capturing 360-degree (1 of 2)

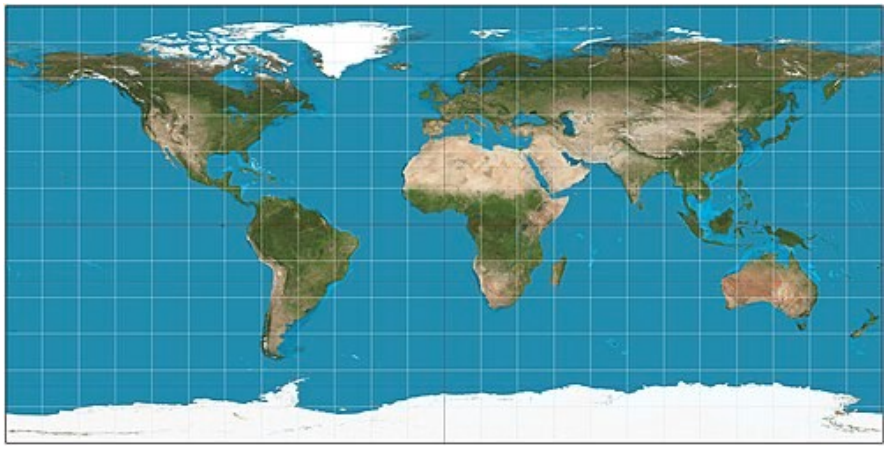
- Multiple camera rigs
  - All cameras are placed in the center
  - Each camera covers a different area
    - E.g., one camera per 60 degrees
  - The images are then stitched together
    - Neighboring cameras have different edge distortions
    - Stitching smooths the image between them
    - Hides the discontinuities between the viewpoints

Capturing 360-degree (2 of 2)

- Omnidirectional cameras
  - Cameras with fisheye lenses
    - Can capture a very wide field of view
  - For 360-degree, at least two lenses required
    - Each covers around ~200 degrees
    - The two viewpoints must be are stitched
  - Lower quality than multiple camera rigs
    - But lower costs, also

360-degree video coding (1 of 3)

- Coding based on existing techniques
  - The stitched image is basically a very large frame
  - It can be projected to a plane
    - Example: equirectangular projection
    - Imagine you are in the center of the globe



360-degree video coding (2 of 3)

- Transmitting 360-degree video is hard
  - The full frame is huge!
- But, do we need the entire frame?
  - Say that you are using a VR headset
  - You can only see a part of the frame
    - Which you know from the headset sensors
  - Why not send just the current FoV?
    - Is the FoV fixed?

360-degree video coding (3 of 3)

- Tile-based transmission schemes
  - Breaks down the video into tiles
    - Each tile is a rectangular frame
    - Various numbers and placements of tiles
  - The tiles in the FoV are transmitted normally
  - Adjacent tiles are transmitted with lower quality
    - When you move your head you can see something
    - Many systems try to predict head movements

Volumetric video

Class: Multimedia Technology, **Section # 24:** Immersive Video

Instructor: George Xylomenos, **Department:** Informatics

Volumetric vs. 3D vs. 360 degree

- 3D video offers a single viewpoint with depth
- 360-degree video offers many viewpoints
  - With or without depth (2D or 3D)
  - But all viewpoints are from the same point
  - You can rotate or move your head up/down
- Volumetric video allows changing position
  - Offers six degrees of freedom (6DoF)
    - Allows full movement of the viewer

Volumetric video capture (1 of 5)

- Volumetric video is a very general term
  - It requires recording the “volume” of objects
  - So, you need some kind of spatial information
- Depth cameras: Microsoft Kinect 360
  - Projects a pattern in the infrared spectrum
  - An IR camera captures the deformed grid

Volumetric video capture (2 of 5)

- Depth is inferred by deformation
  - On flat surface the pattern is aligned
  - On uneven the pattern is deformed
  - A processor calculates depth

Volumetric video capture (3 of 5)

- LIDAR cameras
  - LIDAR sends and detects laser pulses
    - Often used for distance sensors
  - LIDAR cameras can scan an entire scene
    - Similar to Kinect but with laser rather than IR
    - Works over larger distances
  - Used in smart cars to detect surroundings
    - Only interested in depth, not color

Volumetric video capture (4 of 5)

- RGB-D cameras: Intel Real-Sense D415
  - Uses two IR sensors and an IR projector
    - The IR sensors re-create the depth info
    - Higher accuracy but for a short range
      - Compared to LIDAR
  - An additional RGB sensor detects color

Volumetric video capture (5 of 5)

- Multicamera rigs
  - Multiple depth cameras can be combined
    - For example, 6 cameras in 60 degree offsets
  - Scene is captured from all directions
    - From a perimeter, not from a single point
  - Allows user to see scene from many angles
    - With a single camera, viewing angles are limited
    - At other angles, the image is partially reconstructed

Volumetric video presentation

- Holographic projection
  - Image is projected into space
  - Usually based on laser beams
    - Beams cross each other to form image
- AR glasses or VR headsets
  - Video is projected to user's viewpoint
    - In 3D or 2D, depending on application
  - Head movement means viewpoint change
    - User can see the scene from any angle

Volumetric representations (1 of 8)

- Point Cloud (PC)
  - A set of points with (x,y,z) co-ordinates
  - Native output of depth cameras
    - (x,y) implicit from frame, z is distance
  - Points may have additional info
    - RGB-D cameras add color to each point
  - Multiple cameras can provide a combined PC
    - They need to be synced and aligned

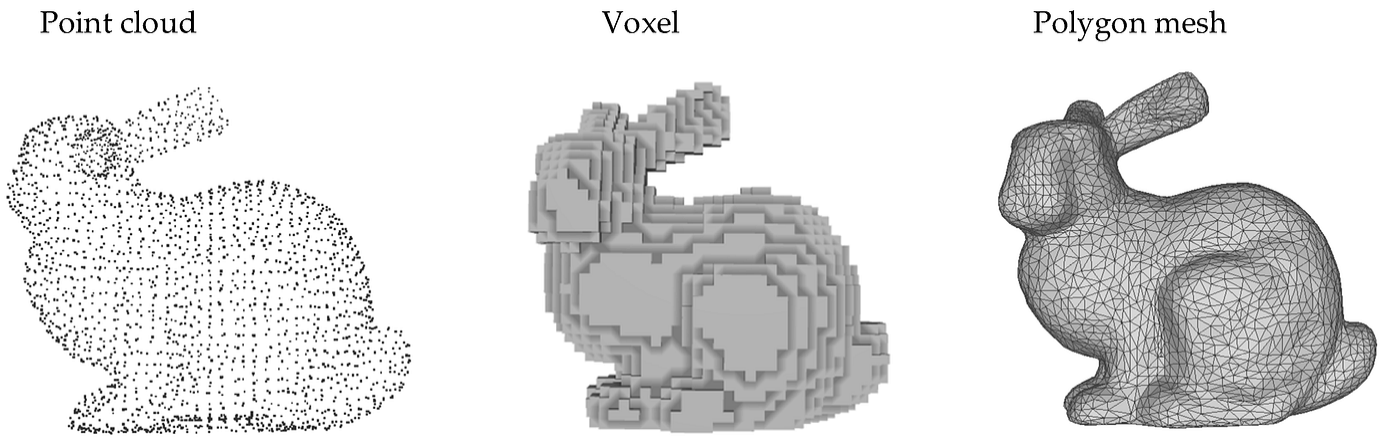
Volumetric representations (2 of 8)

- Voxelization: mapping a PC to voxels
  - Voxel: volume element
    - Analogous to a pixel (picture element)
    - A cube in 3D space
  - Voxelization transforms points to voxels
    - We break down space into voxels
    - Each point in the cloud is quantized to a voxel
      - Quantization depends on the voxel resolution

Volumetric representations (3 of 8)

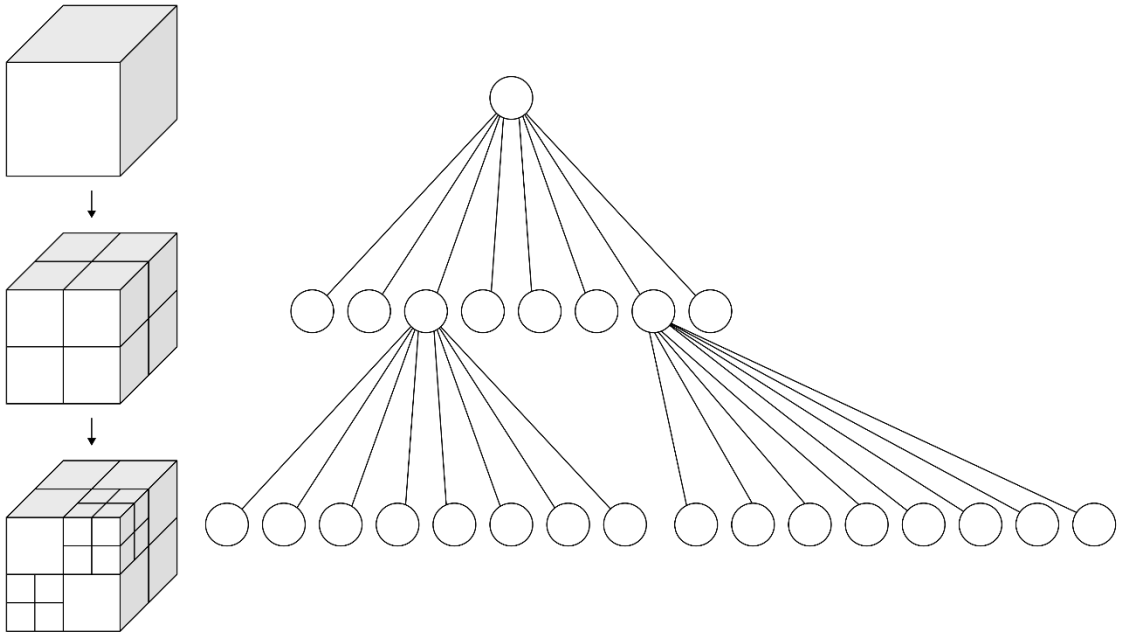
- 3D Mesh
  - A set of polygons (usually, triangles)
  - The vertices have (x,y,z) co-ordinates
  - May be calculated from PCs
    - A surface reconstruction algorithm is used
    - Each surface covers a number of points
  - Each surface may have additional info
    - An RGB color or a texture

Volumetric representations (4 of 8)



- PC vs. Voxel vs. Mesh
  - The PC can come from a depth camera
  - The PC is easily mapped to voxels
  - Translation to a mesh is harder

Volumetric representations (5 of 8)



- Octree: Octal trees
  - Each node has (up to) eight children
  - Each child is a smaller “cube”

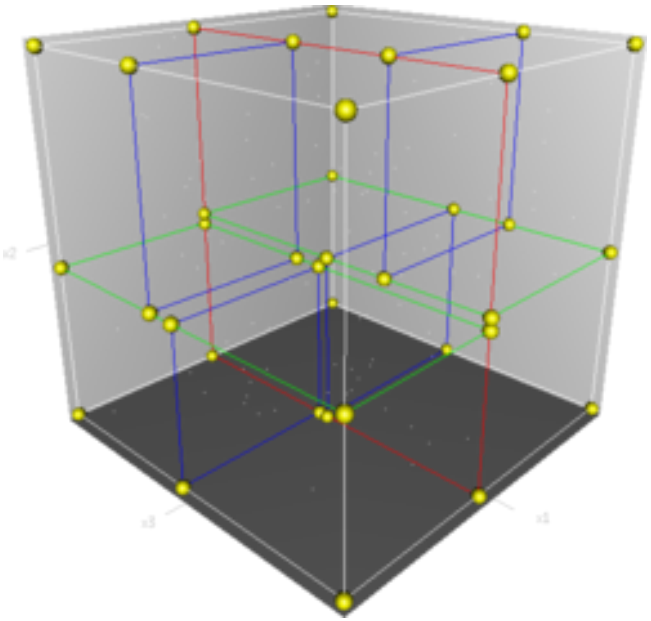
Volumetric representations (6 of 8)

- Octrees can represent voxelized spaces
  - The root is the entire space
  - Its children are its eight constituent cubes
  - Their children are smaller cubes, and so on
- Each leaf is a real voxel
  - Empty areas are null pointers
- We can traverse the tree to render a scene

Volumetric representations (7 of 8)

- k-d tree: a binary three for k dimensions
  - Each level splits space in one dimension
    - For 3D, we have three levels per split
  - Recursively, we reach the actual voxels
- k-d trees are simpler than octrees
  - Binary at each level
  - But triple the levels for the same space
  - So, why use them?

Volumetric representations (8 of 8)



- Can be more efficient than octrees
  - The split is usually at the median
  - More detail where there are actual points

PC compression (1 of 5)

- G-PCC: Geometry PC compression
  - One of two proposals by MPEG
  - Two different use cases
    - Dynamically acquired LIDAR images
    - Static PCs of objects from 3D scanning
  - PC consists of three types of data
    - 3D co-ordinates
    - Reflectance
    - RGB color

PC compression (2 of 5)

- G-PCC: Geometry PC compression
  - Directly encodes the 3D space
    - PC geometry is encoded in an octree
    - Arithmetic coding is used on the octree
    - A triangular mesh structure is also an option
  - Attributes are encoded separately
    - Transform or predictive coding
    - Arithmetic coding in the end

PC compression (3 of 5)

- V-PCC: Video PC compression
  - Used for dynamic volumetric video
  - Maps everything to a 2D space
    - Projection from 3D to 2D
  - The PC is split into patches
    - Each patch is a smooth surface
  - Patches are projected to a 2D space
    - Each patch has a position in a 2D grid

PC compression (4 of 5)

- V-PCC: Video PC compression
  - Two patch maps are created
    - For near and far patches
    - This simplifies the projection
  - The patches are projected to the grid
    - Separately for geometry and attributes
  - The grid is padded into a smooth image
    - Values are copied to empty spaces

PC compression (5 of 5)

- V-PCC: Video PC compression
  - Standard 2D coding is used on each grid
  - An occupancy map is also coded
    - It shows what is occupied in the grid
  - Patch metadata are also coded
    - They show where each patch comes from
  - Decompression reverses this procedure
    - Patches are decoded in 2D and rendered in 3D

PC rendering

- Volumetric video allows any viewpoint
- But it is rendered for a specific viewpoint!
  - The PC or mesh is decoded
  - The scene is translated for a viewpoint
  - The voxels are rendered to an image
    - In the best way possible for the viewpoint
  - Projection to AR glasses or VR headset
    - Or even a 2D screen

End of Section # 24

Class: Multimedia Technology, **Section # 24:** Immersive Video

Instructor: George Xylomenos, **Department:** Informatics