Image-Based Rendering and View Synthesis summary

A. Kouzelis, Prof. Ioannis Pitas
Aristotle University of Thessaloniki
pitas@csd.auth.gr
www.aiia.csd.auth.gr

Version 1.0.1

Artificial Intelligence & Information Analysis Lab

Image-Based Rendering and View Synthesis

- Introduction
- Rendering with no geometry
  - Plenoptic function
  - Light field and Lumigraph
  - Concentric mosaic rendering
  - Panoramic mosaic rendering
- Rendering with implicit geometry
  - View interpolation
  - View morphing
  - Transfer methods
- Rendering with explicit geometry
  - 3D warping
  - Layered Depth Images
  - View-dependent texture mapping
  - Surface rendering
  - Volume rendering
- Learning-based view synthesis
  - Free View Synthesis
  - NERF

Introduction

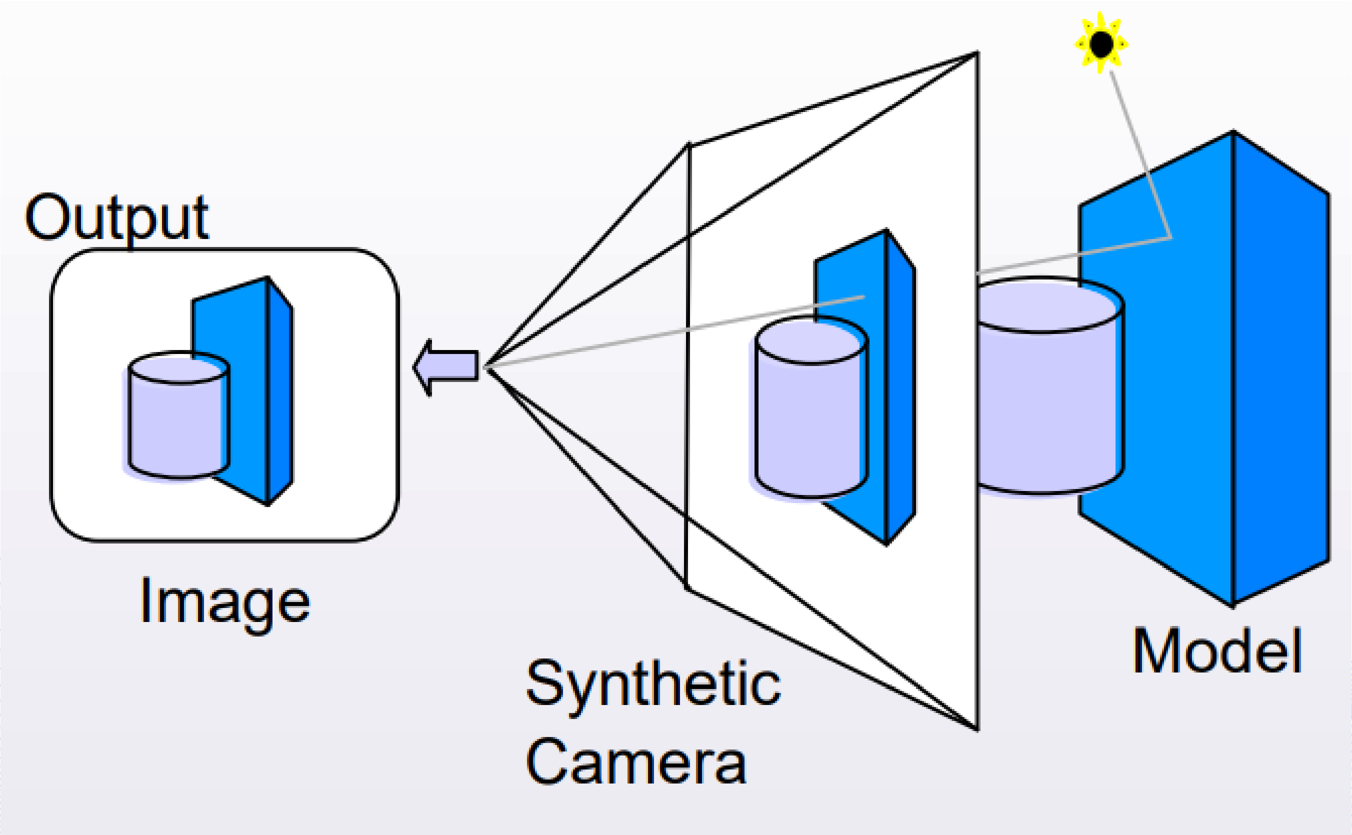
- ***Image rendering*** is the process of generating 2D images (2D views) from 3D object or scene representations.

- ***View synthesis*** is the process of generating novel views of a 3D object or scene from one or multiple other views of the same scene, with or without full 3D model reconstruction.

Introduction

- ***Geometry based rendering***: creation of synthetic images from a 3D scene model by simulating the interaction of light with objects in the scene.
- Besides a geometric model of the scene, material properties of the objects, description of light sources and viewing parameters are also needed.
- Used in traditional computer graphics.
- Examples: rasterization, ray tracing, radiosity.

Introduction



Geometry-based image rendering used in Computer graphics [Cohen1999].

Introduction

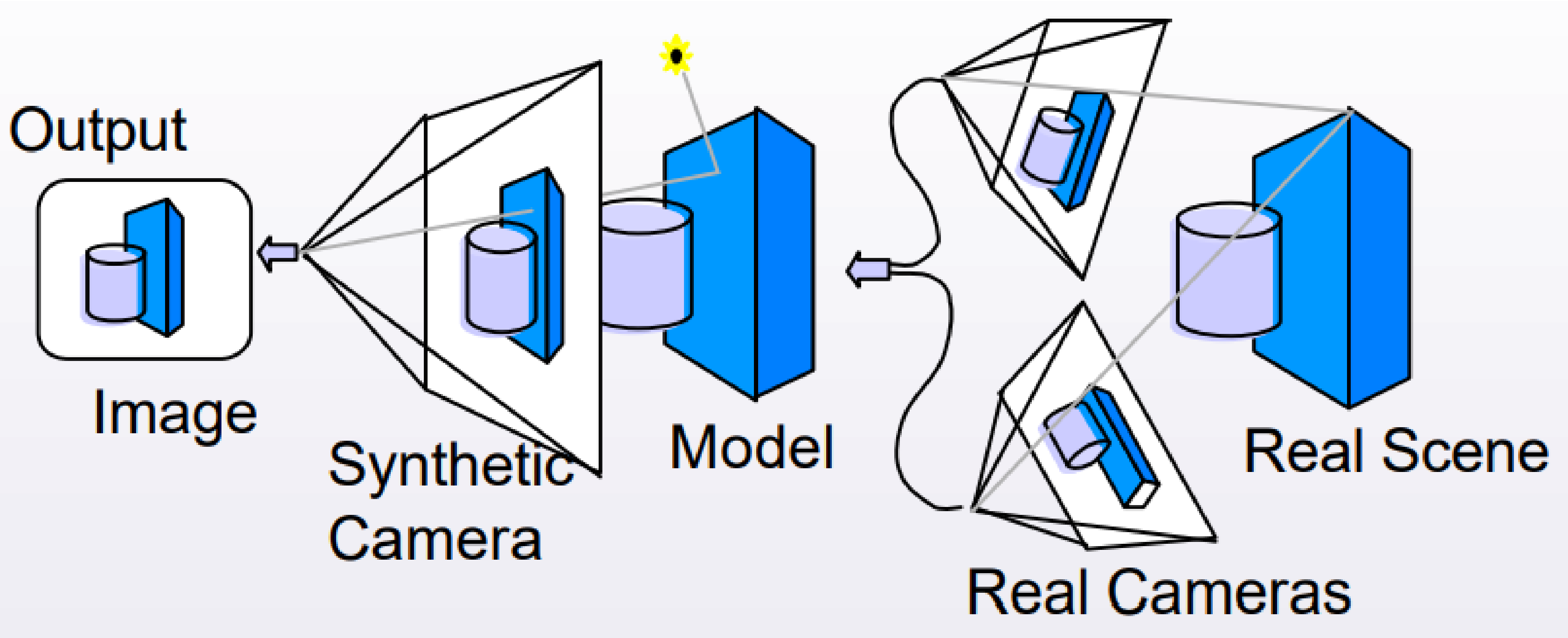


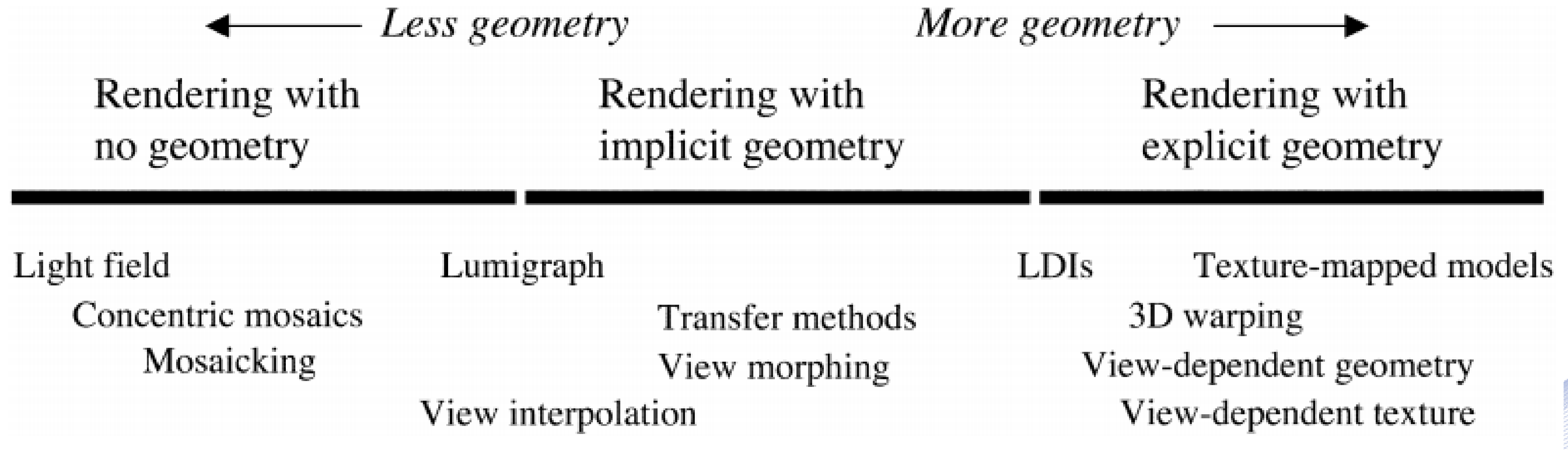
Image-based rendering with model reconstruction [Cohen1999].

Introduction

Image-based rendering can be categorized by the amount of geometry information it uses [Shum00]:

- ***Rendering with no geometry***
- ***Rendering with explicit geometry***
- ***Rendering with implicit geometry***

Introduction



Rendering categories based on the amount of geometry information used [Shum00].

Rendering with no geometry

- In rendering with no geometry, or ***image-based rendering (IBR)***, new views are generated using existing images of the 3D scene from different viewpoints.
- Explicit 3D model of the scene is not used.
- IBR can be viewed as a set of techniques to reconstruct a continuous representation of the ***plenoptic function*** [Adelson91] from observed discrete samples.

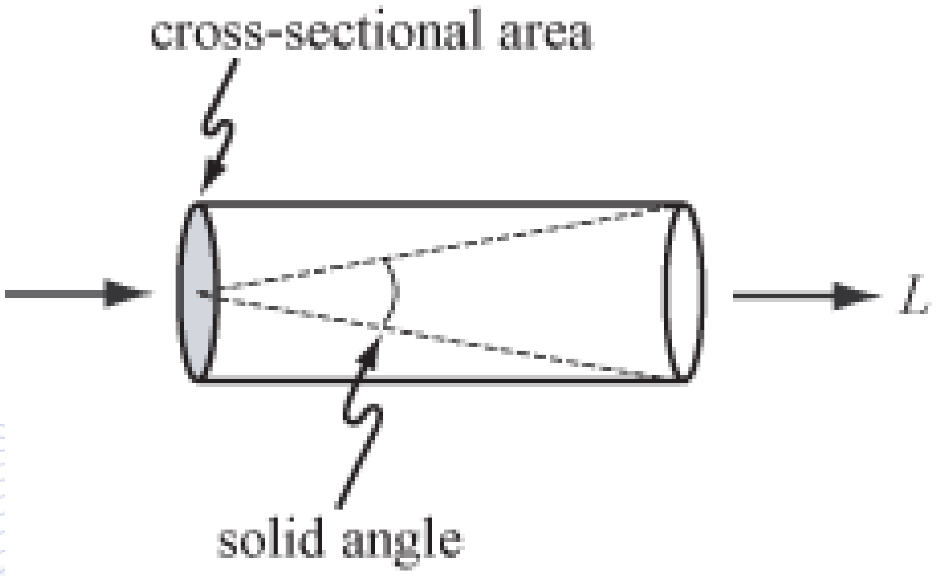
Plenoptic function

- The original 7D ***plenoptic function*** is defined as the radiance of light rays passing through every location in space (V_x, V_y, V_z) where the camera center is located, at every possible angle (θ, ϕ), for every wavelength λ, and for every point in time t:

$$P_7 = P(V_x, V_y, V_z, \theta, \phi, \lambda, t)$$

- The radiance is measured in Watts per meter squared per steradian, $W/m^2/sr$.

Plenoptic function



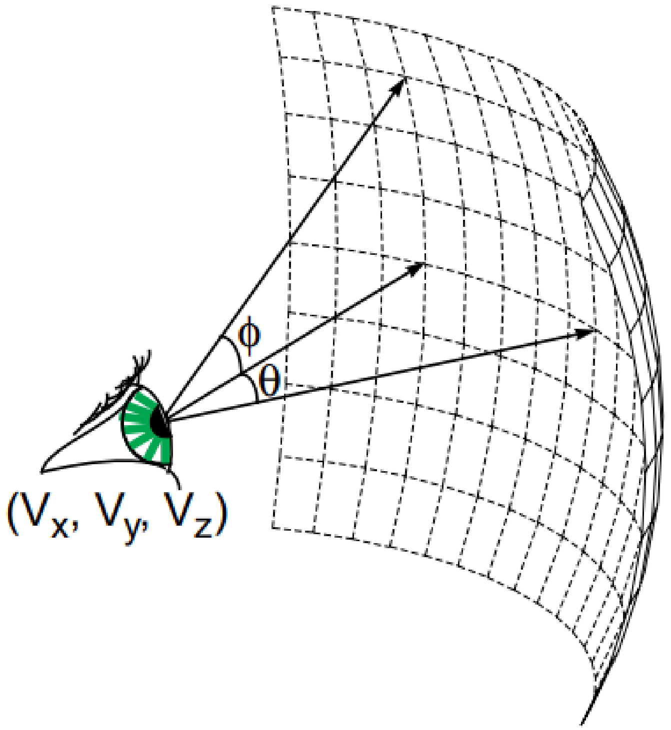
Radiance L along a ray [WIKI1].

Plenoptic function

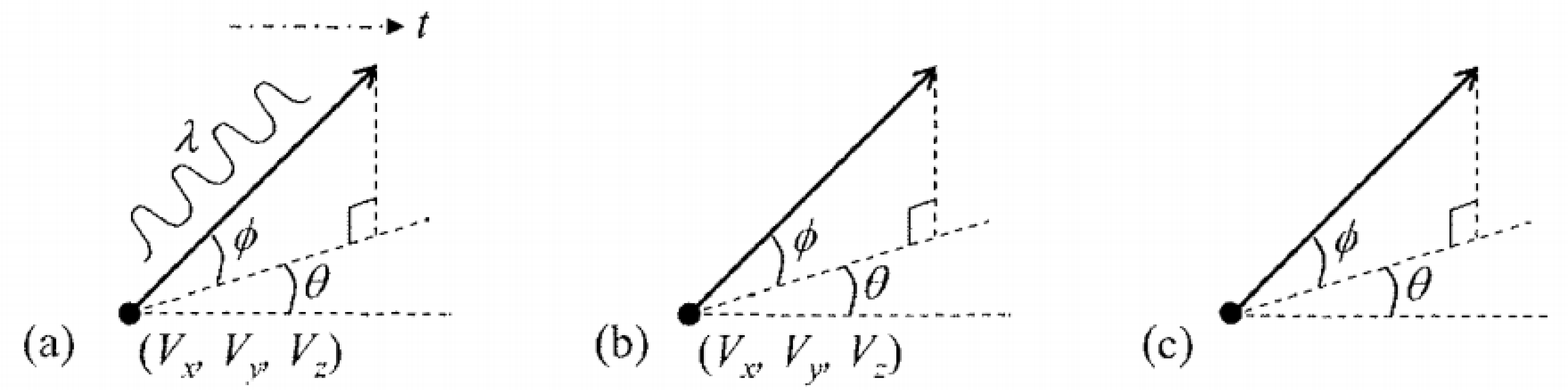
- Considering a static environment and fixed wavelength, the plenoptic function reduces to a 5D function:

$$P_5 = P\left(V_x, V_y, V_z, \theta, \phi\right)$$

- This approach to image-based rendering is known as ***plenoptic modeling*** [McMillan95].



Plenoptic function

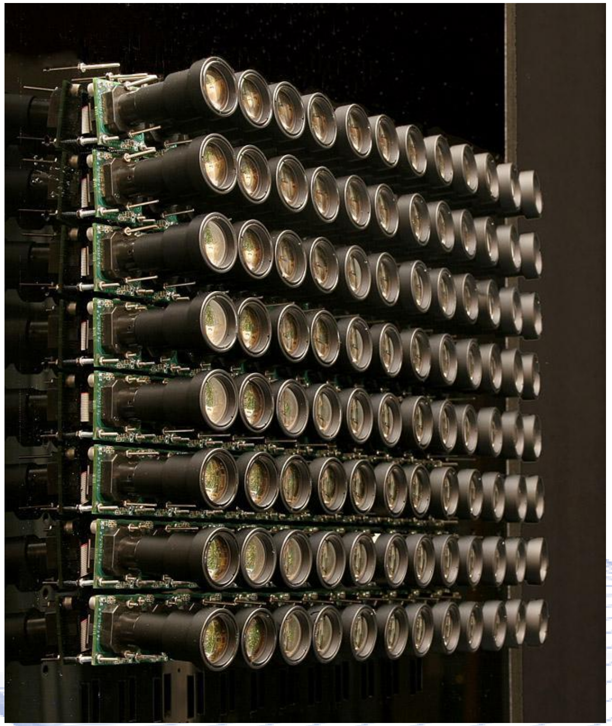


Plenoptic functions: (a) P_7, (b) P_5 and (c) P_2 [Shum07].

Light field and lumigraph

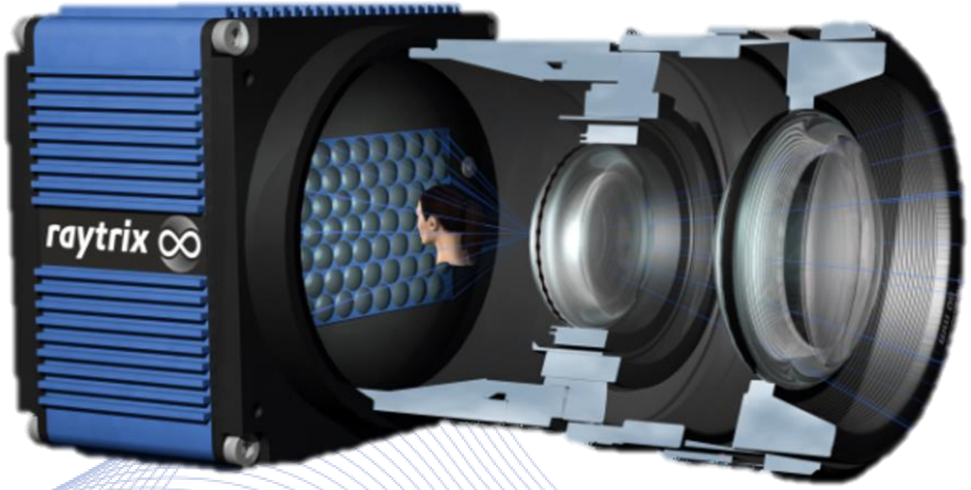
- ***Light field rendering*** [Levoy96] does not use any form of 3D scene geometry, but instead uses many images.
- It generates a new view of a scene by appropriately filtering and interpolating an existing set of samples.
- ***Lumigraph*** [Gortler96] is similar to light field rendering, but it uses approximate 3D scene geometry to counterbalance any lack of uniformity in image sampling.
- The difference lies in uniform vs. non-uniform image sampling.

Light field and lumigraph



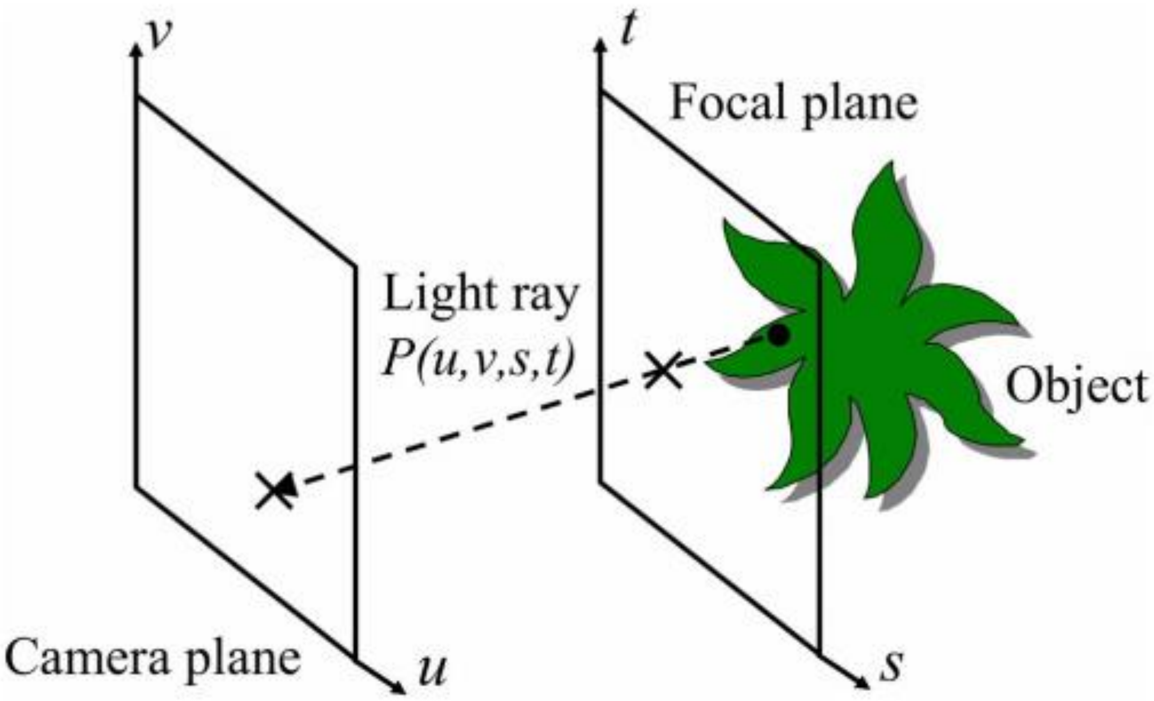
(left) Stanford's multi-camera array, (right) Adobe's "magic lens".

Light field and lumigraph



Commercial light field cameras: (left) Lytro, (right) Raypix.

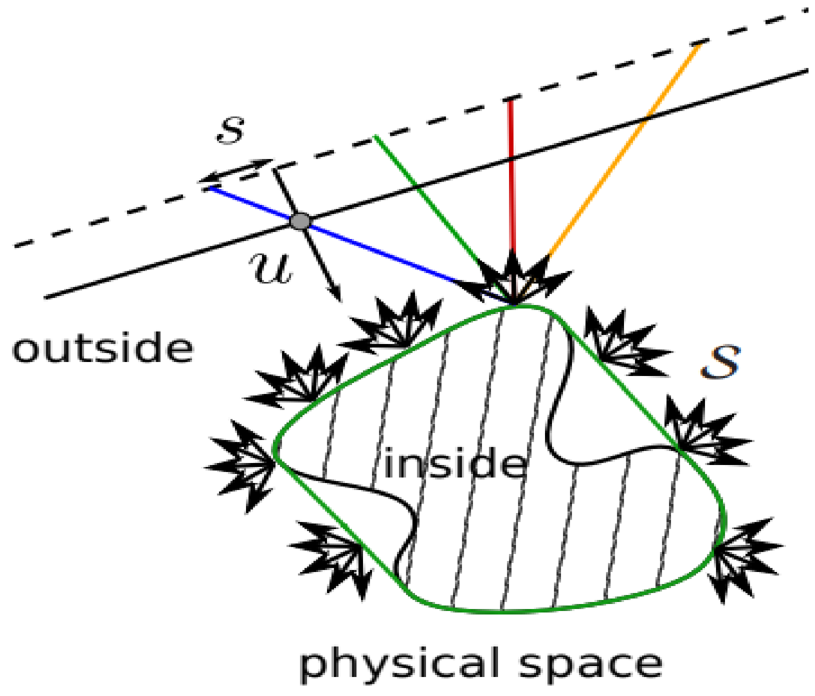
Light field and lumigraph



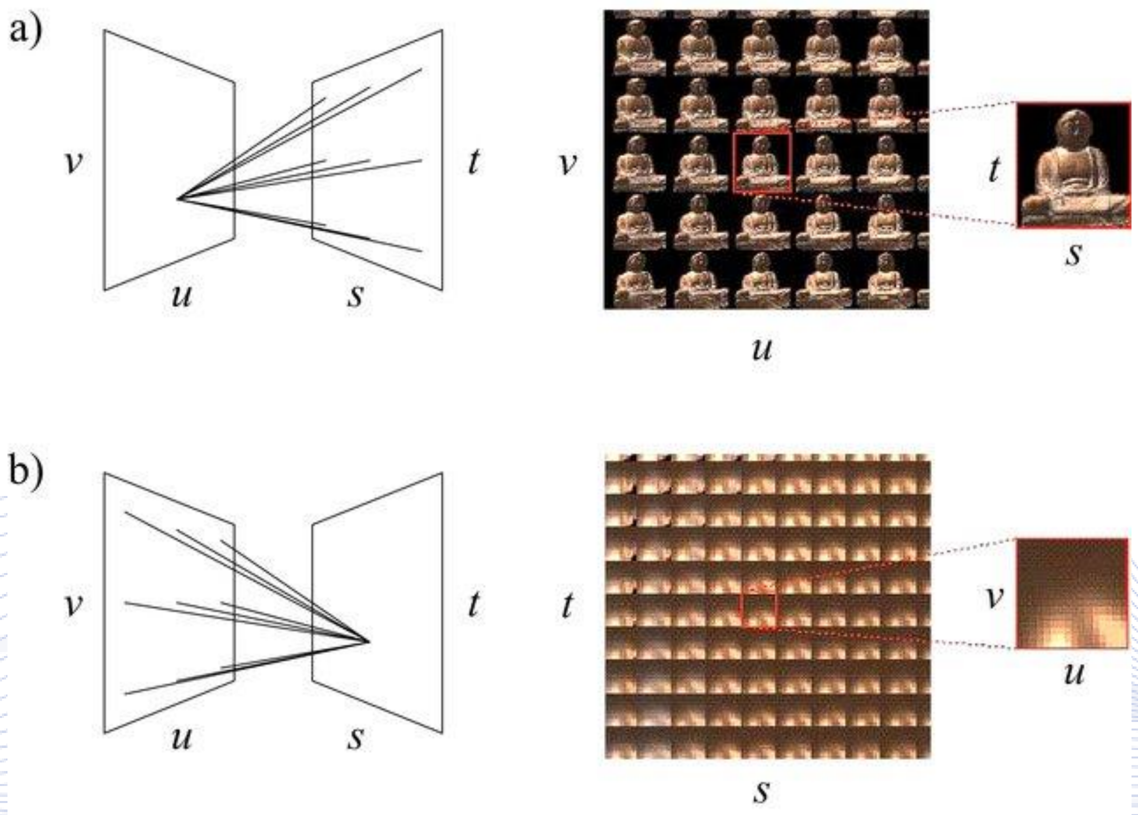
Representation of a light field (light slab) [Shum03]

Light field and lumigraph

The convex boundary S defines the "inside" region which contains the scene of interest and the "outside" empty space where the cameras are located. The radiance of light rays exiting through the boundary S defines the light field. The figure shows only one spatial dimension u and one directional dimension s [Ihrke16].

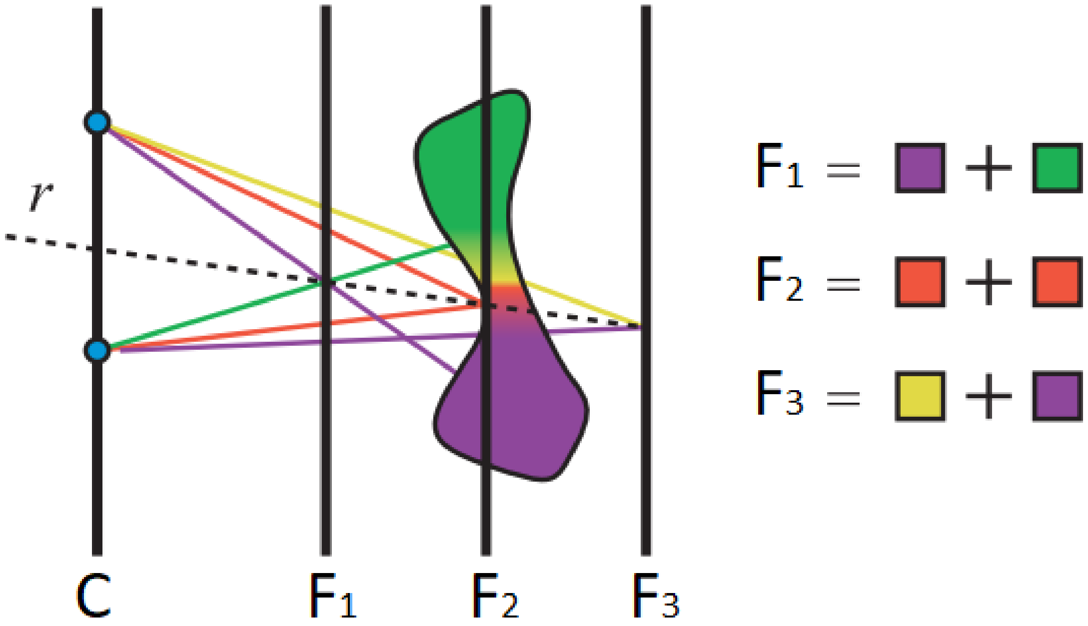


Light field and lumigraph



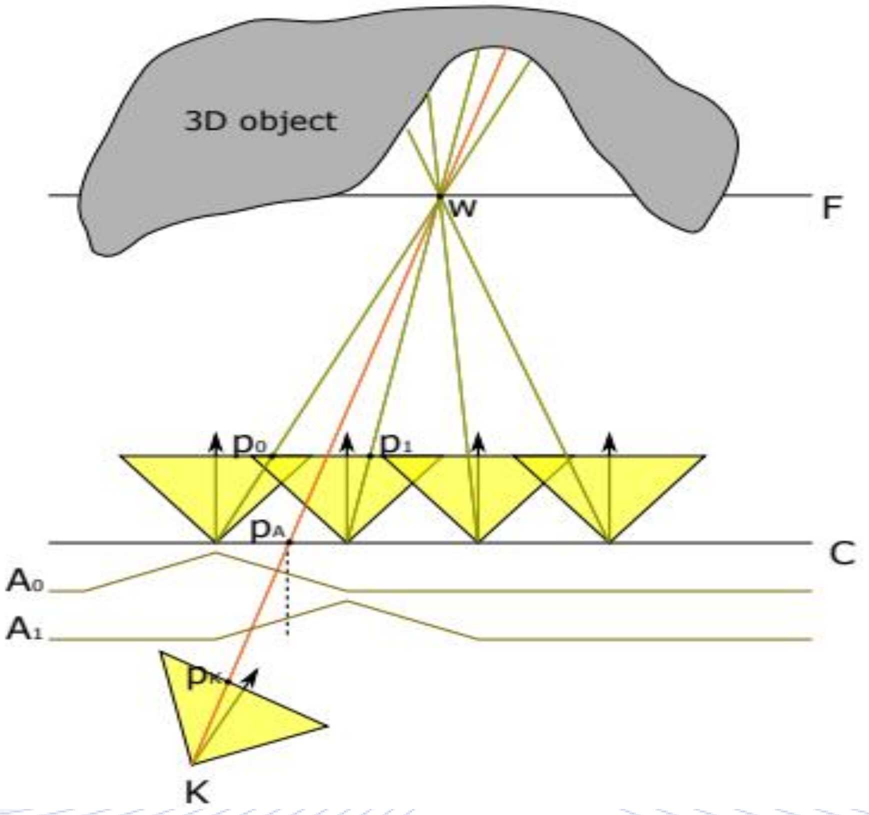
Light field can be viewed from the camera plane (a) or the focal plane (b) [Levoy96].

Light field and lumigraph



Demonstration of how the choice of focal plane affects the reconstruction of a desired ray r. Image taken from [Isaksen00] with slight modifications.

Light field and lumigraph



Light field rendering for the novel view represented by virtual camera K [Mantiuk18].

Concentric mosaic rendering

- ***Concentric mosaics*** (CMs) is a 3D parametrization of the plenoptic function proposed by Shum and He [Shum99].

- The camera is constrained to move only along concentric circles on a plan: it rotates at constant angular speed and captures images at regular intervals.

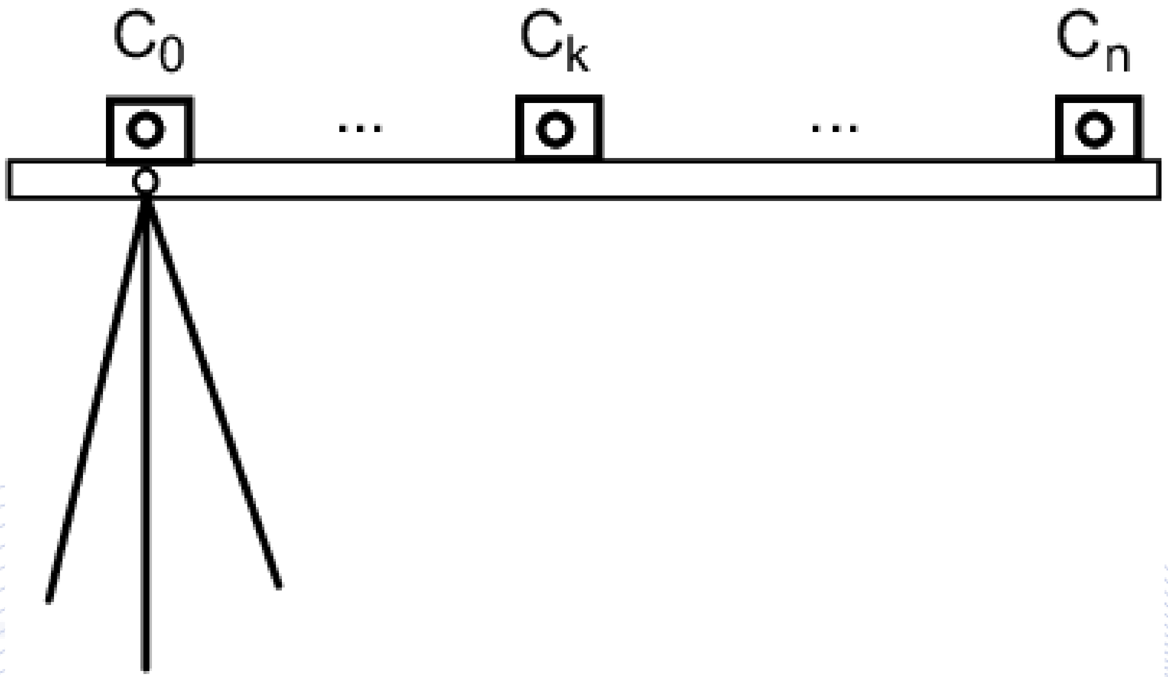
Concentric mosaic rendering

CMs define a ***3D plenoptic function***:

$$P_3 = P(\theta, r, Z)$$

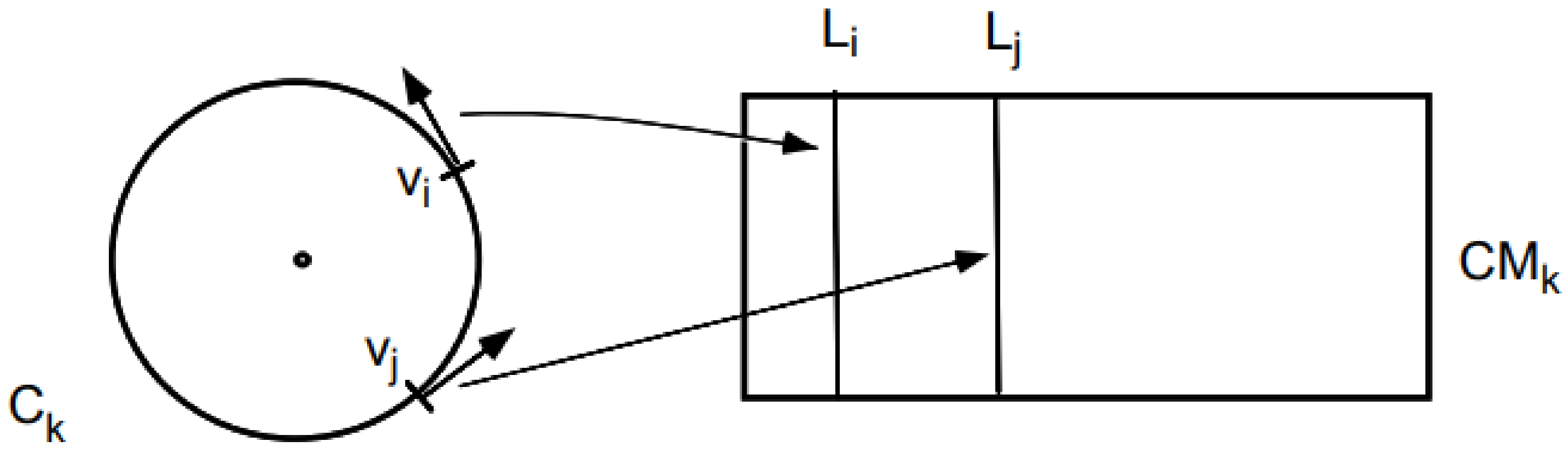
where θ is the rotation angle, r is the radius and Z is the vertical elevation.

Concentric mosaic rendering



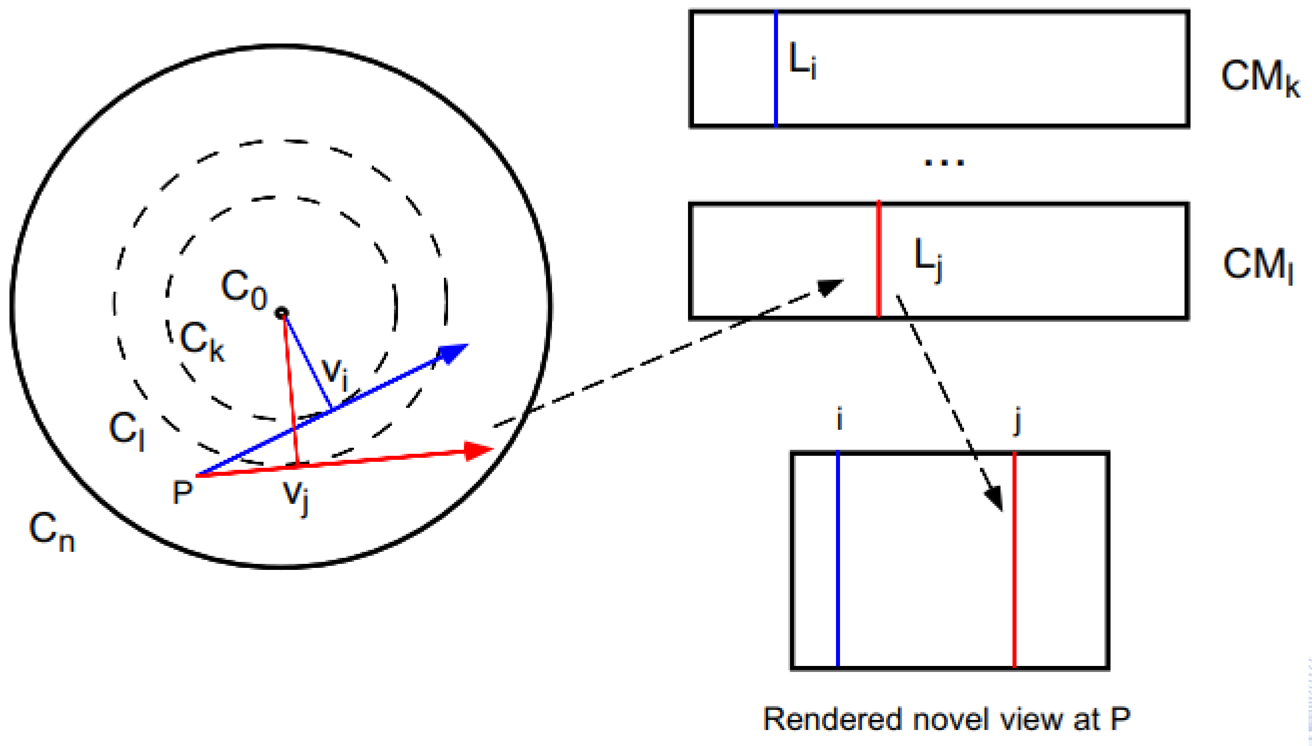
A system setup for obtaining concentric mosaics [Shum99].

Concentric mosaic rendering



Construction of a concentric mosaic [Shum99].

Concentric mosaic rendering



Rendering a new view with CMs [Shum99].

Panoramic mosaic rendering

- ***Image mosaicing*** is the process of aligning and compositing multiple images into a larger image that represent portion of a 3D scene.

- The output mosaic is a ***2D plenoptic function***.

- Usually mosaics are composed to increase the field of view of the camera.

Panoramic mosaic rendering

Example of a cylindrical panorama [WIKI2].

Rendering with implicit geometry

- In ***rendering with implicit geometry*** the geometric information is not directly available, but is deduces after appropriate calculations on the images.
- It relies on information extracted from images, such as feature correspondences between images, depth maps etc.
- The most characteristic techniques of this class are: view interpolation, view morphing and transfer methods.

View interpolation

- ***View interpolation*** uses existing images to create new in-between views by interpolation.

- The method proposed by Chen and Williams [Chen93] uses ***dense optical flow*** between two input images.

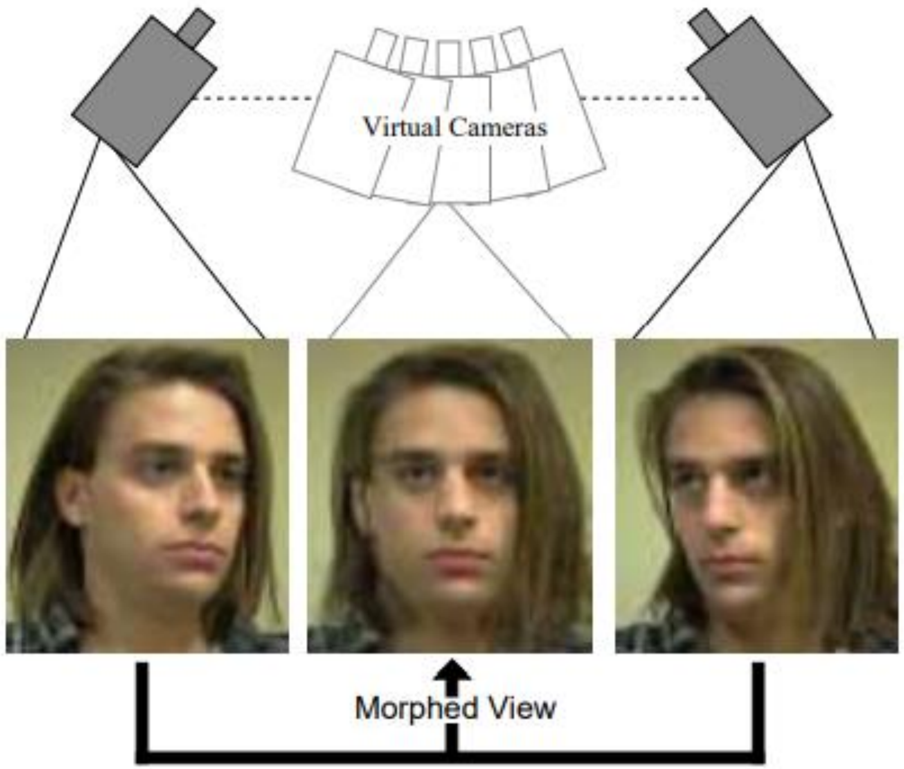
- Optical flow is the apparent motion of objects, surfaces and edges in a visual scene caused by the relative motion between the camera and the scene [Pitas00].

View morphing

- ***Image morphing*** is a technique used to generate transitions between two images, without ensuring the preservation of shapes of the objects.

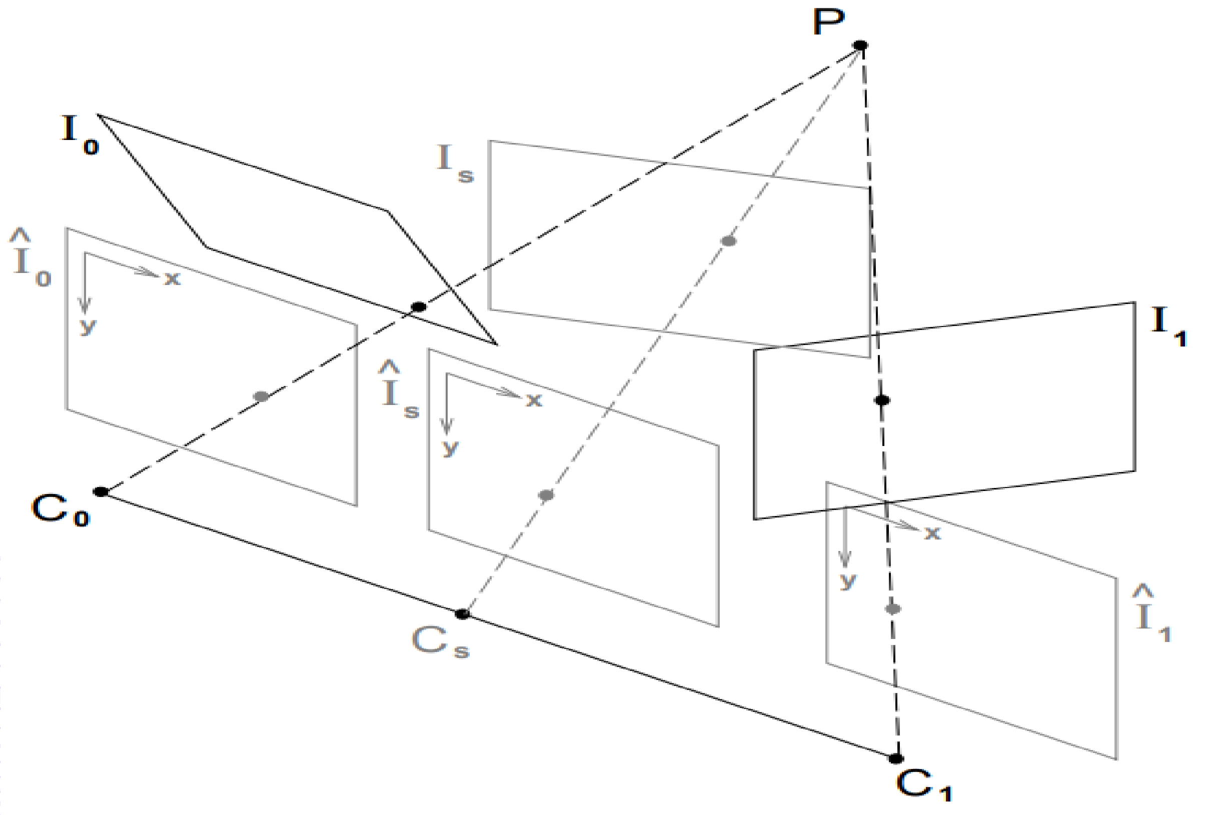
- ***View morphing*** [Seitz96] is an extension of image morphing, which ensures object shape preservation in the new views and results in very realistic transitions.

View morphing



View morphing between two different views of an object [Seitz96].

View morphing



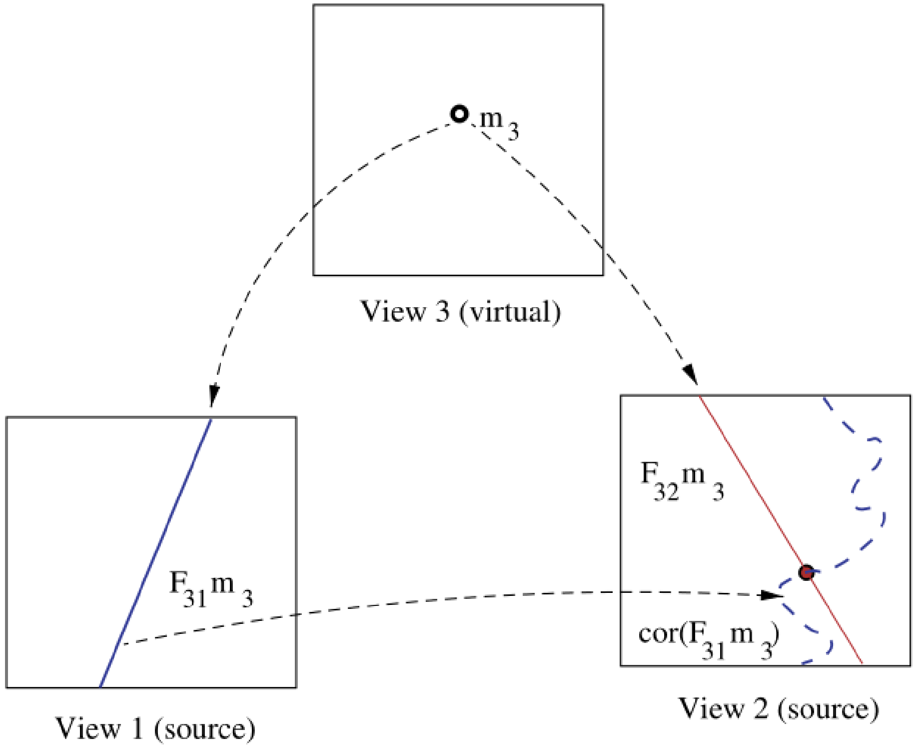
View morphing [Seitz96]: Original images I_0 and I_1 are pre-warped to form parallel views $\hat{I}_0$ and $\hat{I}_1$. The image $\hat{I}_s$ is formed by interpolating the $\hat{I}_0$ and $\hat{I}_1$. Finally, I_s is formed by post-warping $\hat{I}_s$.

Transfer methods

- ***Transfer methods*** use a relatively small number of images and apply geometric constraints to appropriately reproject image pixels at a virtual camera viewpoint.

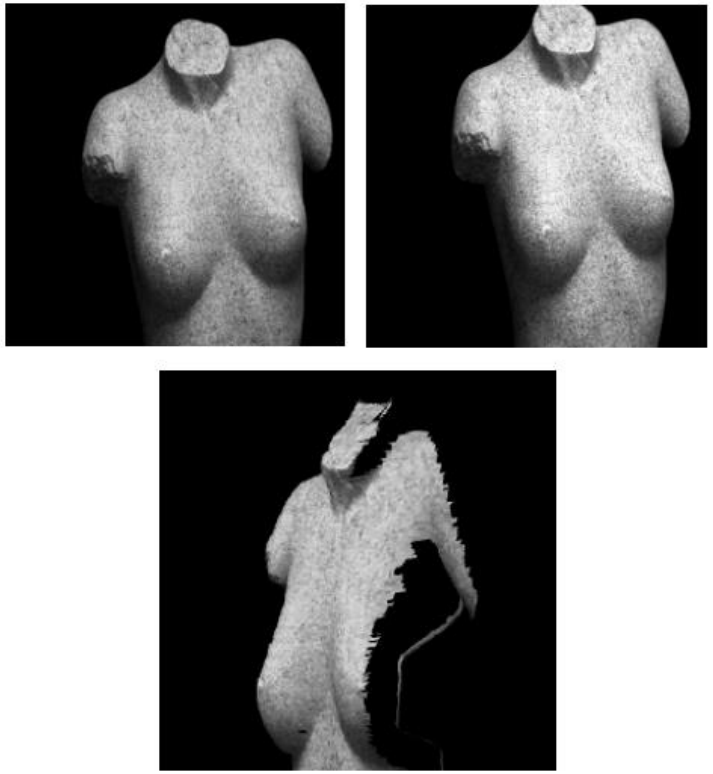
- The geometric constraints can be of the form of pixel depth values, epipolar constraints between pair of images, or trifocal tensors that link correspondences between triplets of images [Shum07].

Transfer methods



Corresponding points in two source views that are projections of a point $\mathbf{m}_3$ in the virtual view 3 [Shum07].

Transfer methods



Top: front images of a mannequin. Bottom: predicted side view of the mannequin [Laveau94].
The unseen areas in the reference views such as the right breast are not visible in the transferred.

Rendering with explicit geometry

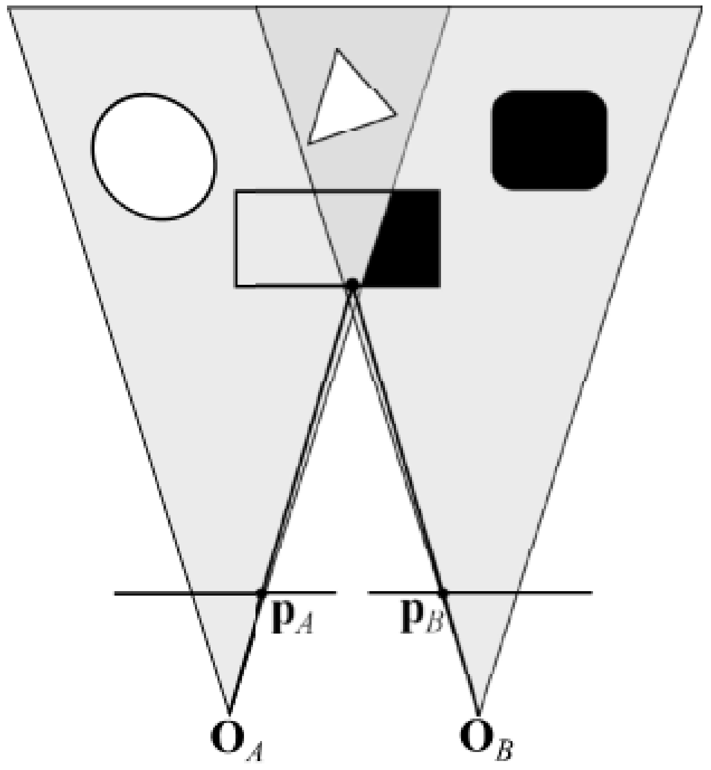
VML

- In ***rendering with explicit geometry***, the representations of a scene have direct 3D information encoded in them, in form of depth along known lines of sight, or 3D coordinates.
- Traditional rendering with a 3D model and single texture map is a special case in this category used in conventional computer graphics.
- Here we are going to review rendering methods that use depth maps, such as 3D warping and layered-depth images, and the view-dependent texture mapping which uses explicit 3D models of objects or scenes.

3D warping

- In ***3D image warping***, a given 2D reference image and a depth map (for every point in this image) can be used to render *nearby* views of the 3D scene [McMillan97].
- The pixels of the reference image are projected to their proper 3D locations.
- Then the 3D points are re-projected onto a new image plane pixels.

3D warping

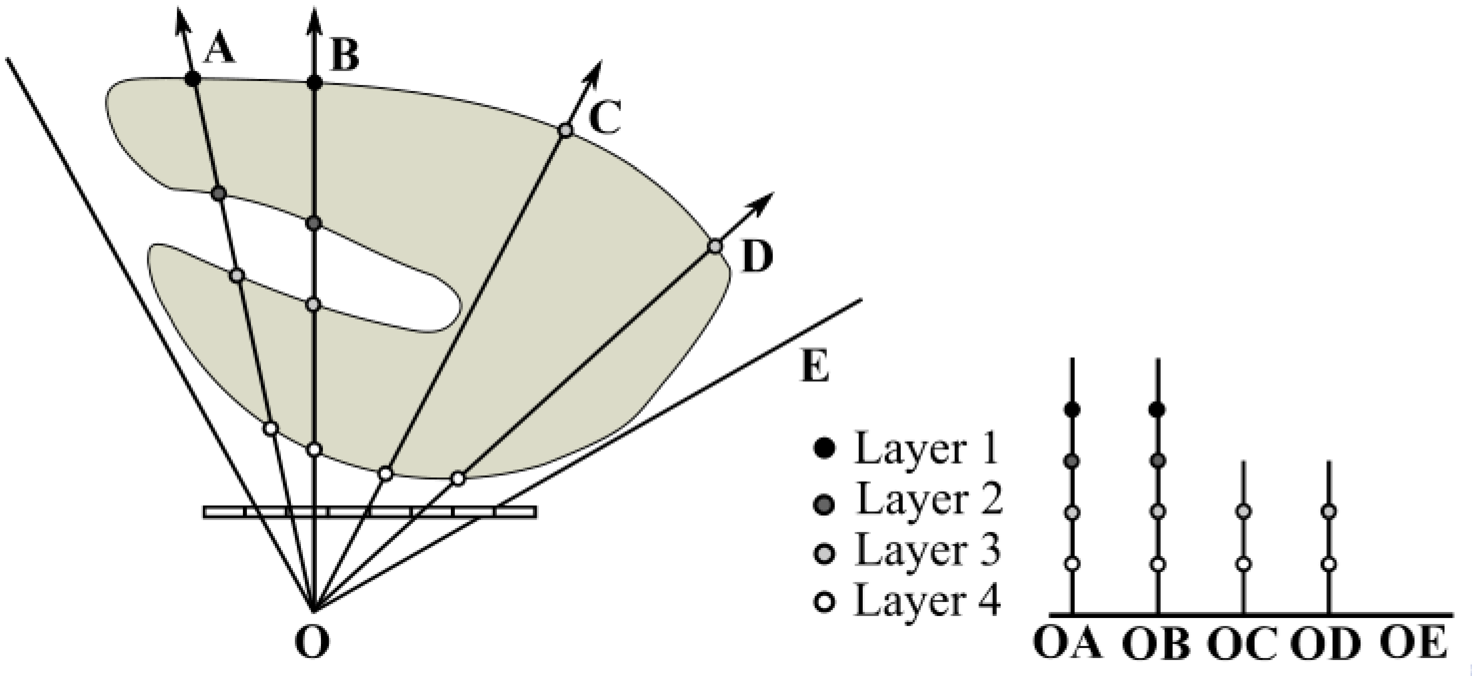


Visibility gap [Pitas].

Layered Depth Images

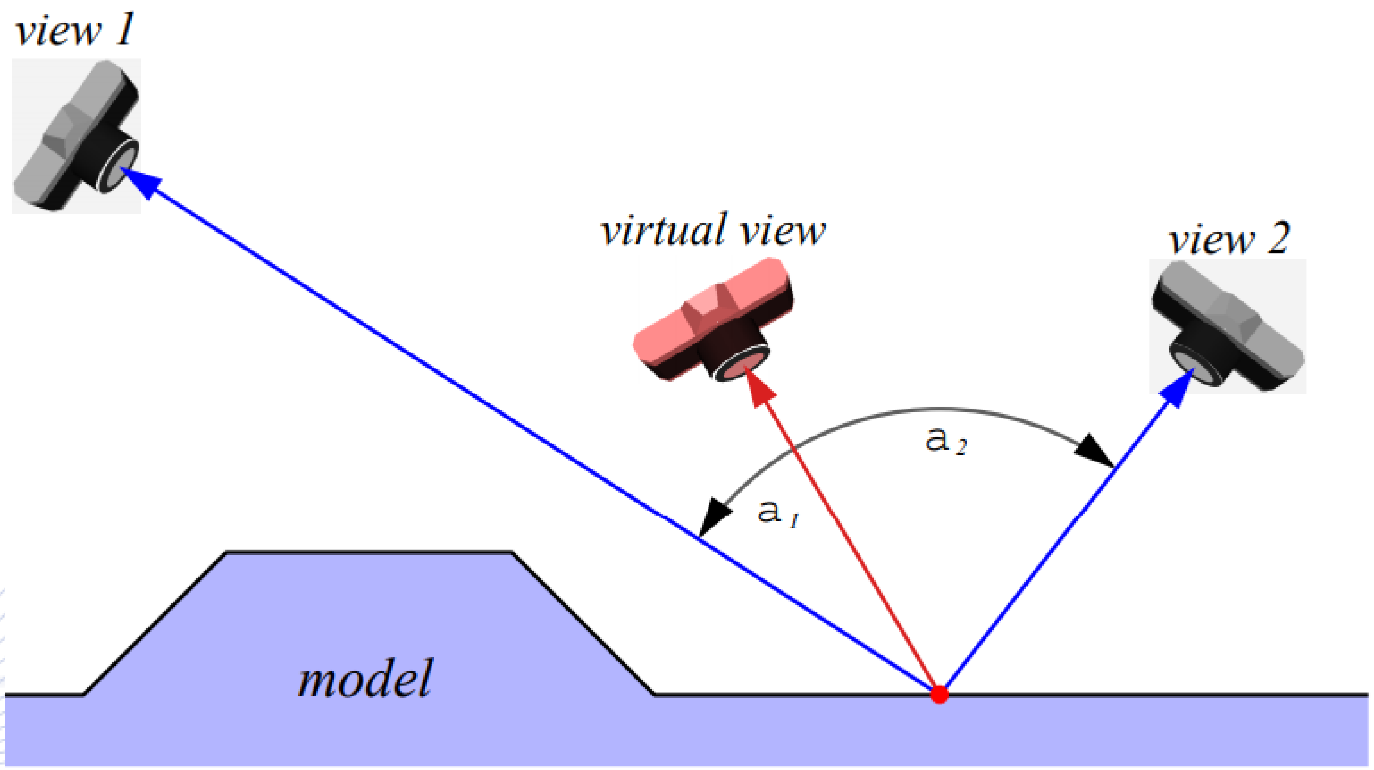
- To deal with the disocclusion problem in 3D warping, Shade et al. proposed ***Layered Depth Images (LDI)*** [Shade98] to store not only what is visible in the reference image, but also what is behind the visible surface.
- LDI stores multiple pixels and their depths along each line of sight, forming the ***LDI multiple layers***.
- LDIs can be constructed either by using multiple images for which depth information is available at each pixel or directly from synthetic environments with known geometries [Baker98].

Layered Depth Images



Layered Depth Image [Pitas]

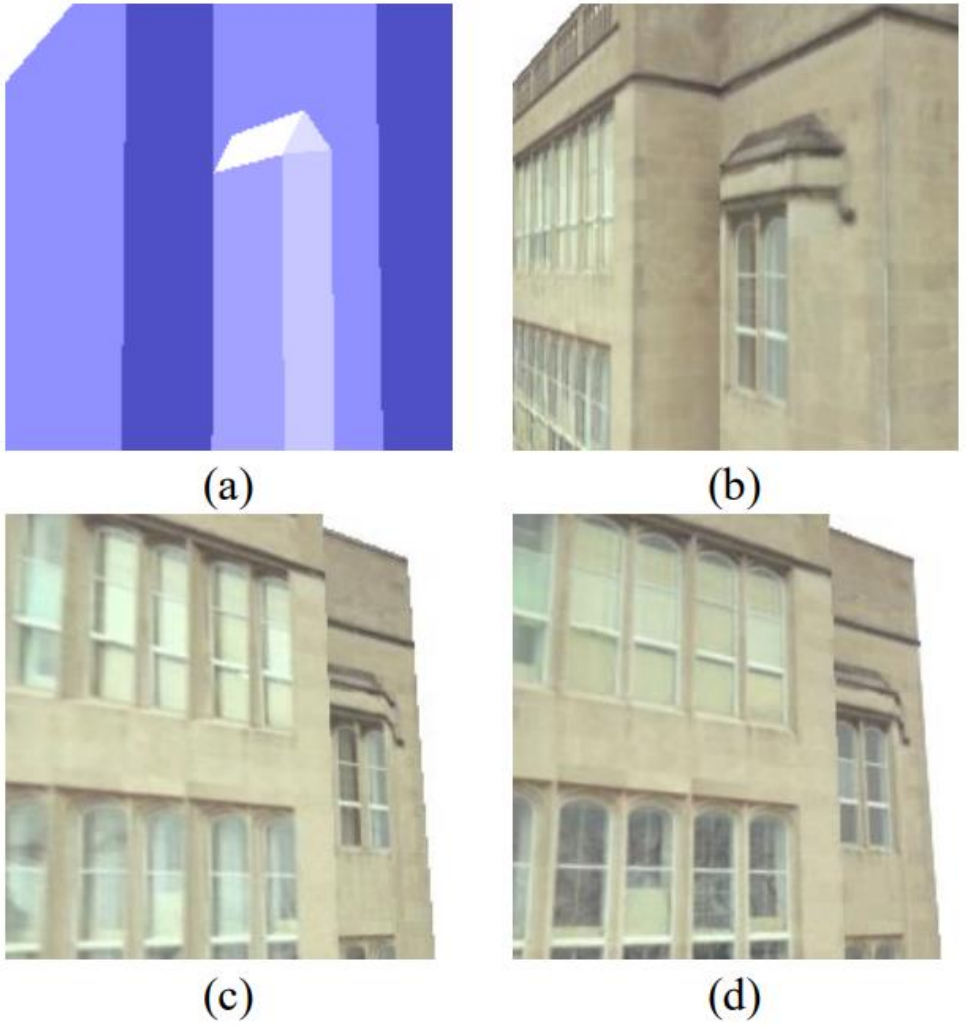
View-dependent texture mapping



Weighting used in view-dependent texture mapping [Debevec96]. The pixel in the virtual view is a weighted sum of pixels in sampled viewpoints. The weights are inversely proportional to the magnitude of angles a_1 and a_2.

View-dependent texture mapping

Results of view-dependent texture mapping (VDTM) [Debevec96]. **(a)** 3D model of a building, **(b)** A rendering of the model using VDTM, **(c)** Rendering from a other view using a single texture map. Because the texture is from a different viewpoint, the windows look not natural, **(d)** A better result obtained by using VDTM.



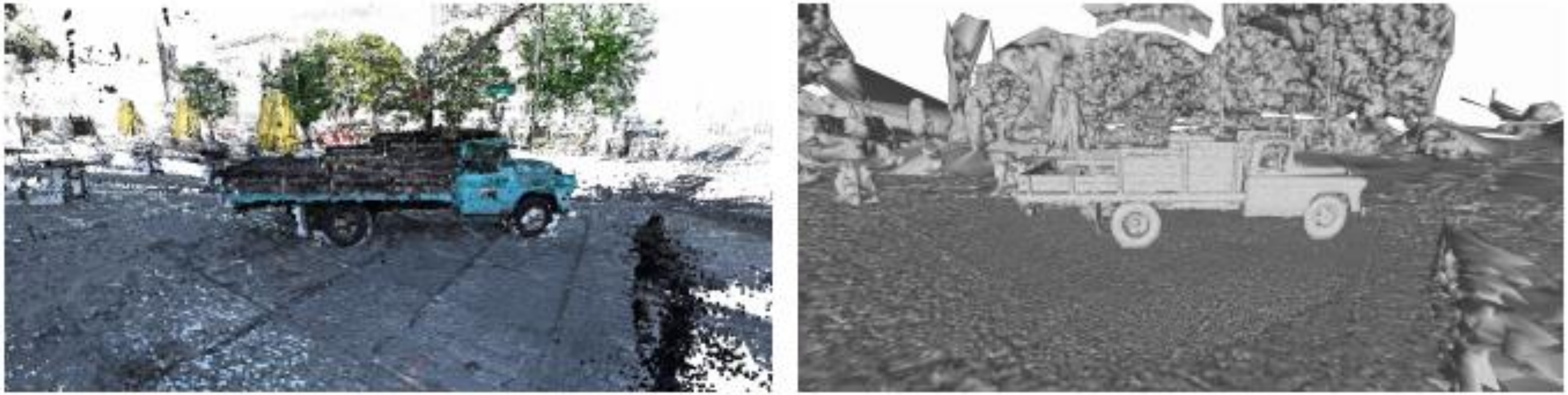
Learning-based view synthesis

- In the recent years, there have been many successful attempts in applying deep learning techniques to view synthesis, exploring different deep learning architectures, 3D scene representations and application scenarios.
- View synthesis can be naturally formulated as a learning problem: Given a large set of images of a 3D scene from different viewpoints, keep some of them as ground truth, train a model to predict the missing views and compare them to the ground truth images as the objective that the model seeks to optimize.

Free View Synthesis

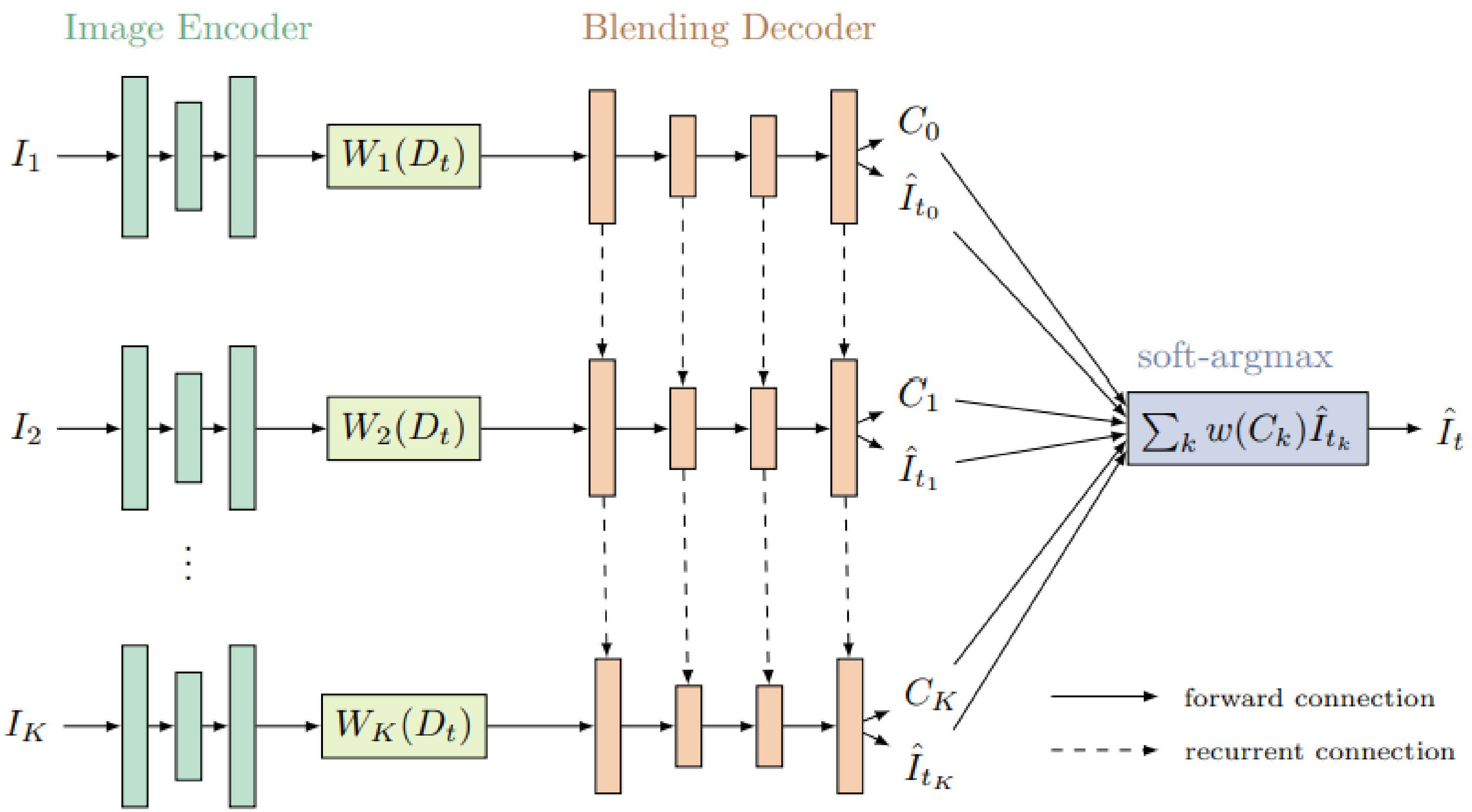
- In the paper “Free View Synthesis” [Riegler20] authors propose a deep learning based method for novel view synthesis from unstructured input images, i.e. images that are freely distributed around a scene.
- Their method works for general scenes with unconstrained geometric layouts.
- After training on a dataset, the method works on previously unseen environments without the need for fine-tuning or per-scene optimization.

Free View Synthesis


(a) Point cloud (b) Mesh

Proxy geometry used for view synthesis [Riegler20].

Free View Synthesis



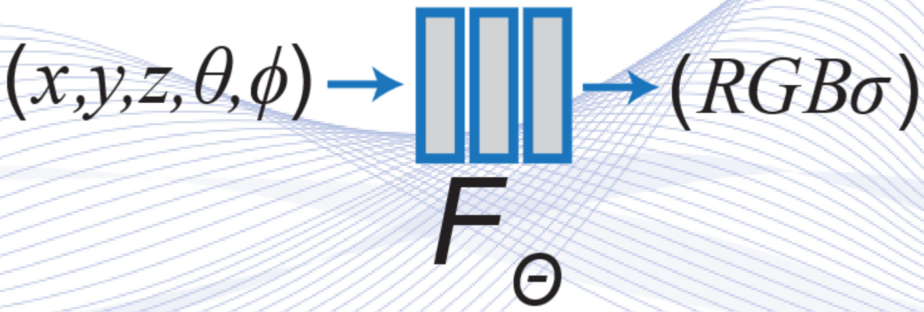
Structure of the recurrent mapping and blending network [Riegler20].

NeRF

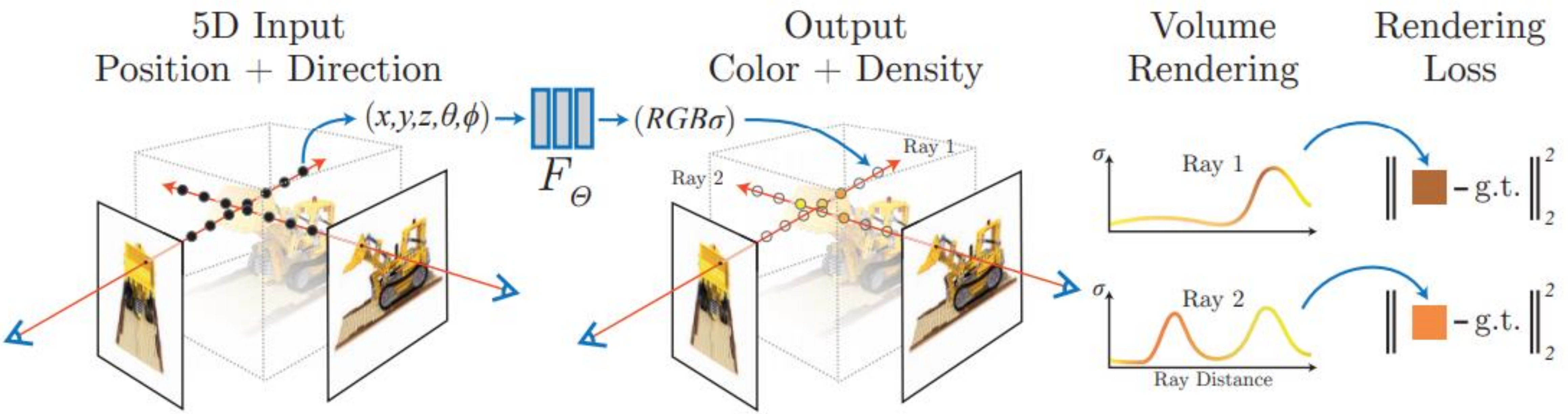
- ***Neural Radiance Field (NeRF)*** is a state-of-the-art method for generating novel views of complex scenes [MIL2020].
- A static scene is represent as a continuous 5D function that outputs:
  - the emitted ***color*** $\mathbf{c} = (r, g, b)$ at each point $\mathbf{x} = (x, y, z)$ in space in each direction (θ, ϕ) (represented as a 3D unit vector $\mathbf{d}$),
  - a ***volume density*** σ at each point $\mathbf{x}$ which acts like a differentiable opacity controlling how much radiance is

NeRF

- This continuous 5D function is approximated with a Multilayer Perceptron (MLP):
$$F_\Theta : (\mathbf{x}, \mathbf{d}) \to (\mathbf{c}, \sigma)$$
by optimizing its weights Θ.



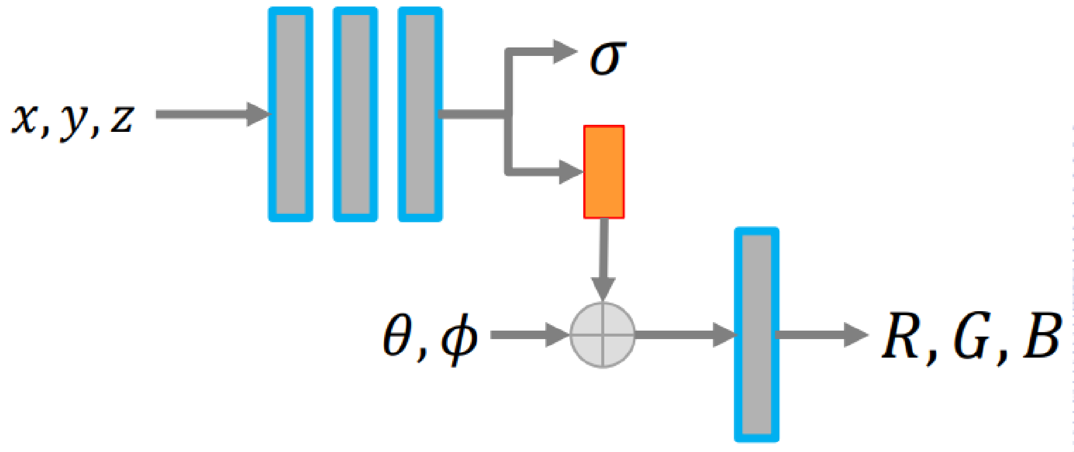
NeRF



NeRF scene representation and differentiable rendering procedure [MIL2020].

NeRF

- The feature vector is then concatenated with camera's viewing direction and passed to one additional FC layer (with ReLU activation and 128 channel) that outputs the view-dependent RGB color.



NeRF

Results on a synthetic dataset [MIL2020]..

NeRF

Results on a real world photo dataset [MIL2020].
(Press on each video to play)

Artificial Intelligence & Information Analysis Lab

Bibliography

[PIT2021] I. Pitas, “Computer vision”, Createspace/Amazon, in press.

[PIT2017] I. Pitas, “Digital video processing and analysis” , China Machine Press, 2017 (in Chinese).

[PIT2013] I. Pitas, “Digital Video and Television” , Createspace/Amazon, 2013.

[NIK2000] N. Nikolaidis and I. Pitas, “3D Image Processing Algorithms”, J. Wiley, 2000.

[PIT2000] I. Pitas, “Digital Image Processing Algorithms and Applications”, J. Wiley, 2000.

Bibliography

[Adelson91] Adelson E. H., Bergen J., **“The plenoptic function and the elements of early vision”**, In Computational Models of Visual Processing, MIT Press, Cambridge, MA, (1991).

[Chen93] Chen S.E., Williams L., “**View Interpolation for Image Synthesis**”, ACM Transactions on Graphics (SIGGRAPH) (1993).

[McMillan95] McMillan L., Bishop G. **“Plenoptic modeling: an image-based rendering system”**, SIGGRAPH (1995).

[Debevec96] Debevec P.E., Taylor C.J., Malik J., “**Modelling and rendering architecture from photographs: A hybrid geometry-and image-based approach**”, SIGGRAPH (1996).

[Gortler96] Gortler S.J., Grzeszczuk R., Szeliski R., Cohen M.F., “**The Lumigraph**”, SIGGRAPH (1996).

Bibliography

[Levoy96] Levoy M., Hanrahan P., “**Light field rendering**”, SIGGRAPH (1996)

[Cohen99] Cohen M., **"Course on Image-based, Modeling, Rendering, and Lighting"**, Siggraph (1999)

[Scharstein99] Scharstein D., “**View Synthesis Using Stereo Vision**”, Lecture Notes in Computer Science (1999).

[Shum00] Shum H., Kang S. B., **“Review of image-based rendering techniques”**, Visual Communications and Image Processing (2000).

[Shum03] Shum H., Chan S. C., Kang S. B., **“Survey of image-based representations and compression techniques”,** IEEE Transactions on Circuits and Systems for Video Technology (2003).

Bibliography

[Shum07] Shum H., Chan S. C., Kang S. B., **“Image-Based Rendering**”, Springer US (2007).

[Pitas00] Pitas I., Nikolaidis N., “**3-D image processing algorithms**”, Wiley (2000).

[McMillan97] McMillan L., **“An Image-Based Approach to Three Dimensional Computer Graphics”**, Ph.D. Thesis, Department of Computer Science, University of North Carolina at Chapel Hill, (1997).

[Ihrke16] Ihrke I., Restrepo J., Mignard-Debise L., **“Principles of Light Field Imaging: Briefly revisiting 25 years of research”**, IEEE Signal Processing Magazine (2016)

[Isaksen00] Isaksen A., McMillan L., Gortler S. J., **“Dynamically reparameterized light fields”**, SIGGRAPH (2000).

Bibliography

[Shum99] Shum H. Y.,He L. W., **“Rendering with Concentric Mosaics”**, SIGGRAPH (1999).

[Mantiuk18] Mantiuk R. I., **“Advanced Graphics and Image Processing - Lecture notes”**.

[Neto02] Neto M., **“Image-Based Modeling and Rendering Techniques: A Survey”**, RITA (2002).

[Zhang04] Zhang C., Chen T. **“A survey on image-based rendering—representation, sampling and compression”**, Signal Processing: Image Communication (2004).

[Szeliski97] Szeliski R., Shum H. Y., **“Creating full view panoramic image mosaics and environment maps”**, SIGGRAPH (1997).

[Seitz96] Seitz S. M., Dyer, C. R., **“View morphing”**, SIGGRAPH (1996).

Bibliography

[Laveau94] Laveau S., Faugeras O. D., **“3-D scene representation as a collection of images”**, Proceedings of 12th International Conference on Pattern Recognition (1994).

[Shade98] Shade J., Gortler S., He L. W., Szeliski R., **“Layered depth images”**, SIGGRAPH (1998).

[Baker98] Baker S., Szeliski R., Anandan P., **“A Layered Approach to Stereo Reconstruction”**, CVPR (1998).

[Gortler97] Gortler S., He L., Cohen, M., **“Rendering Layered Depth Images”** (1997).

[Porter84] Porter T., Duff T, **“Compositing digital images”**, SIGGRAPH Computer Graphics (1984).

Bibliography

[McMillan95] McMillan L., **“A list-priority rendering algorithm for redisplaying projected surfaces”**, UNC Technical Report 95-005, University of North Carolina (1995).

[Debevec96] Debevec P. E., Taylor C. J., Malik J., **“Modeling and Rendering Architecture from photographs: A hybrid geometry- and image-based approach”**, SIGGRAPH (1996).

[Riegler20] Riegler G., Koltun V., **“Free View Synthesis”**, ECCV (2020).

[Mildenhall2020] Mildenhall B., Pratul P. Srinivasan, Tancik M., Barron J. T., Ramamoorthi R., Ng R., **“NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis”**, ECCV (2020)

[WIKI1] https://en.wikipedia.org/wiki/Light_field
[WIKI2] https://en.wikipedia.org/wiki/QuickTime_VR

Q & A

Thank you very much for your attention!

More material in
http://icarus.csd.auth.gr/cvml-web-lecture-series/

Contact: Prof. I. Pitas
pitas@csd.auth.gr