ICC DevCon 2020

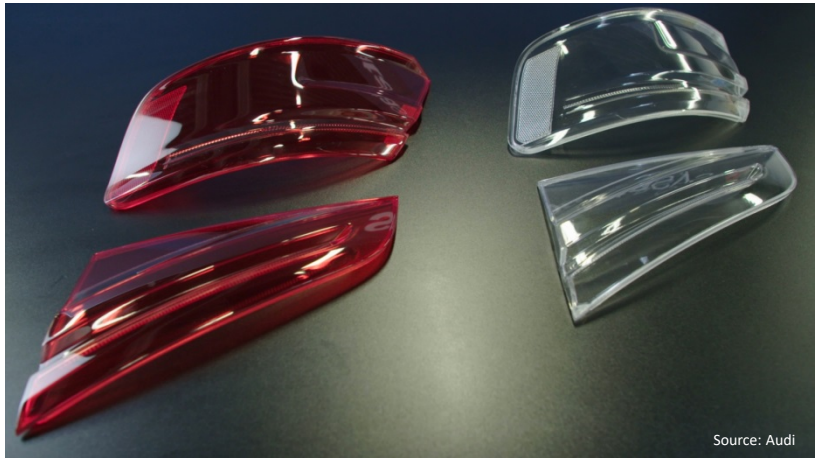
3D Printing spatially varying color and translucency

Philipp Urban

Fraunhofer IGD

NTNU

Application areas



Design prototypes

Replacements parts fitting into existing visual environment

Cultural heritage

Entertainment

3D printers available

Stratasys J750

Mimaki 3DUJ-553

Materials:

Cyan

Magenta

Yellow

Black

White

Transparent

Hi-Resolution Multimaterial 3D Printers (Polyjet)

Related work

Color 3D Printing (examples)

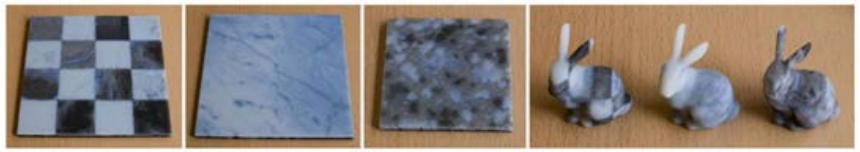
Joint

color + translucency

Translucency 3D Printing (examples)

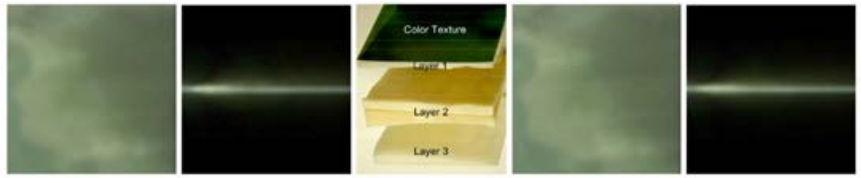
Brunton et al. 2015

Hašan et al. 2010



Babaei et al. 2017

Dong et al. 2010



Reproducing perceptual attributes

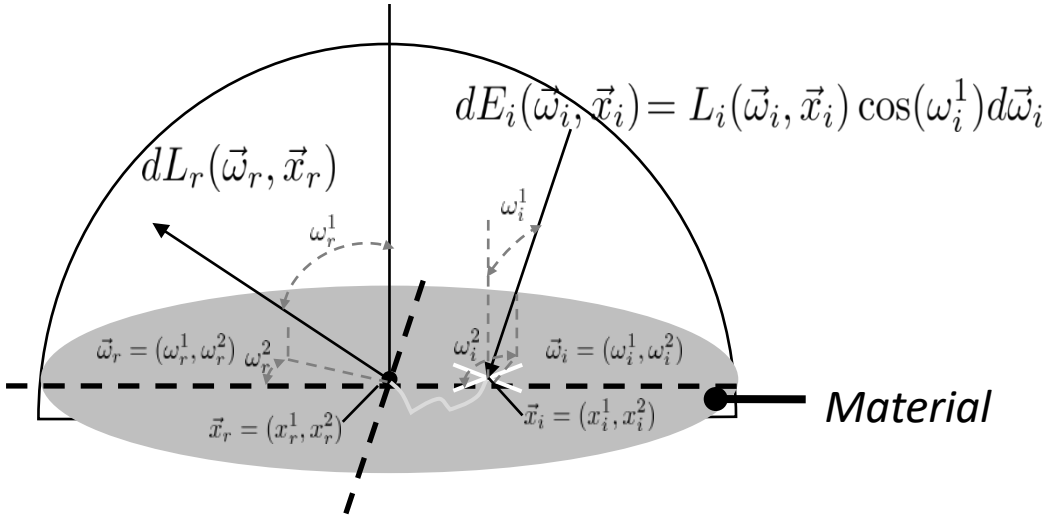
?

Reproducing physical quantities

Reproducing Physical Quantities?

Why not using BSSRDFs?

- High dimensionality
  - High memory consumption
- Measurement effort
  - Instruments not available
  - Time consuming
- No perceptual metric



Can we reproduce given BSSRDFs?

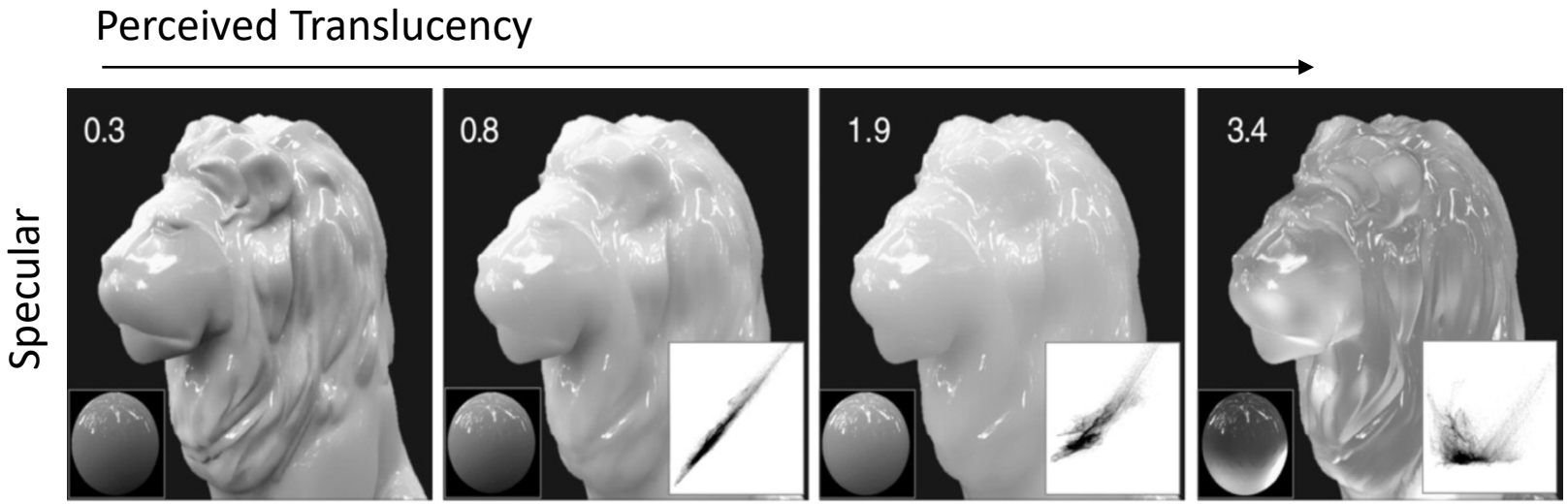
Mostly impossible due to printing systems limits:

- Absorption (factor)
- Scattering (factor / function)
- Refractive index
- Resolution (surface roughness)

$$S(\vec{\omega}_i, \vec{x}_i, \vec{\omega}_r, \vec{x}_r) = \frac{dL_r(\vec{\omega}_r, \vec{x}_r)}{dE_i(\vec{\omega}_i, \vec{x}_i)}$$

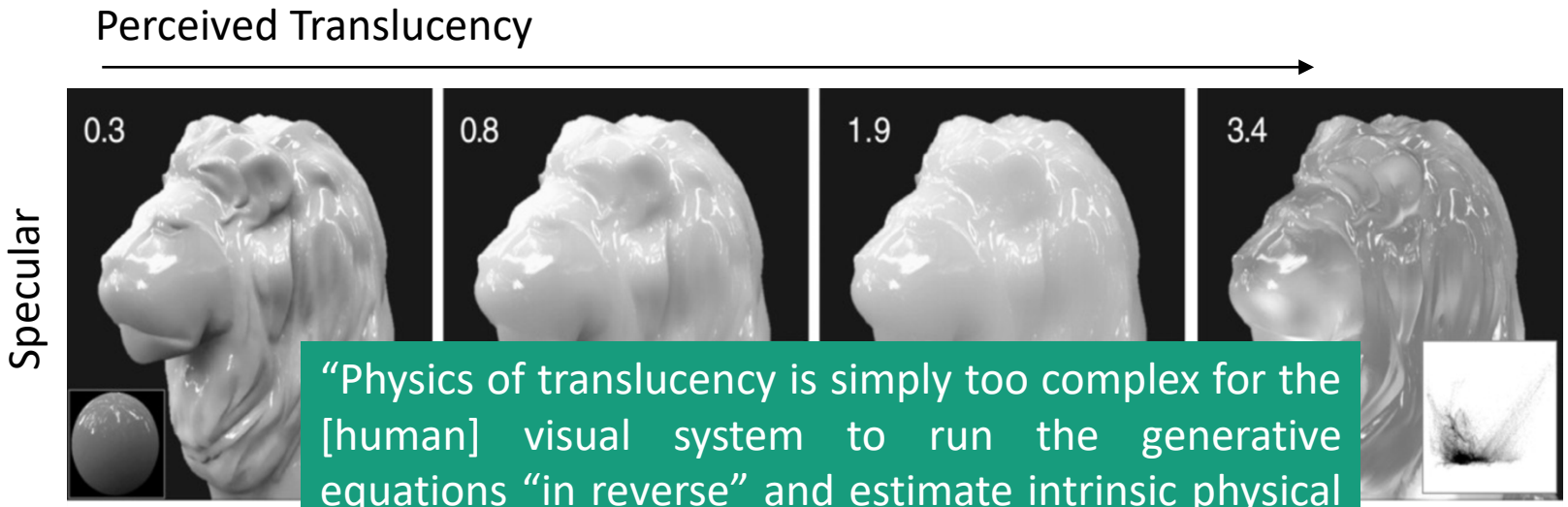
Bidirectional Surface Scattering
Reflectance Distribution Function (BSSRDF)

Perceptual attributes



Motoyoshi, 2010

Perceptual attributes



"Physics of translucency is simply too complex for the [human] visual system to run the generative equations "in reverse" and estimate intrinsic physical parameters via inverse optics.

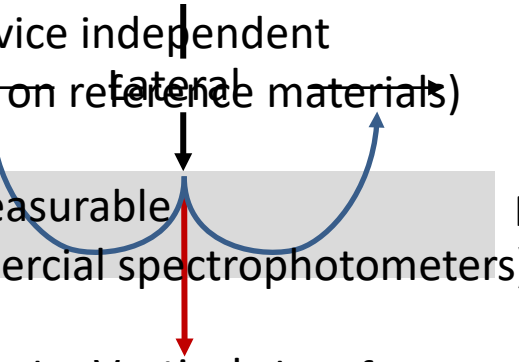
… instead the visual system relies on simple image heuristics [cues] that correlate with translucency."

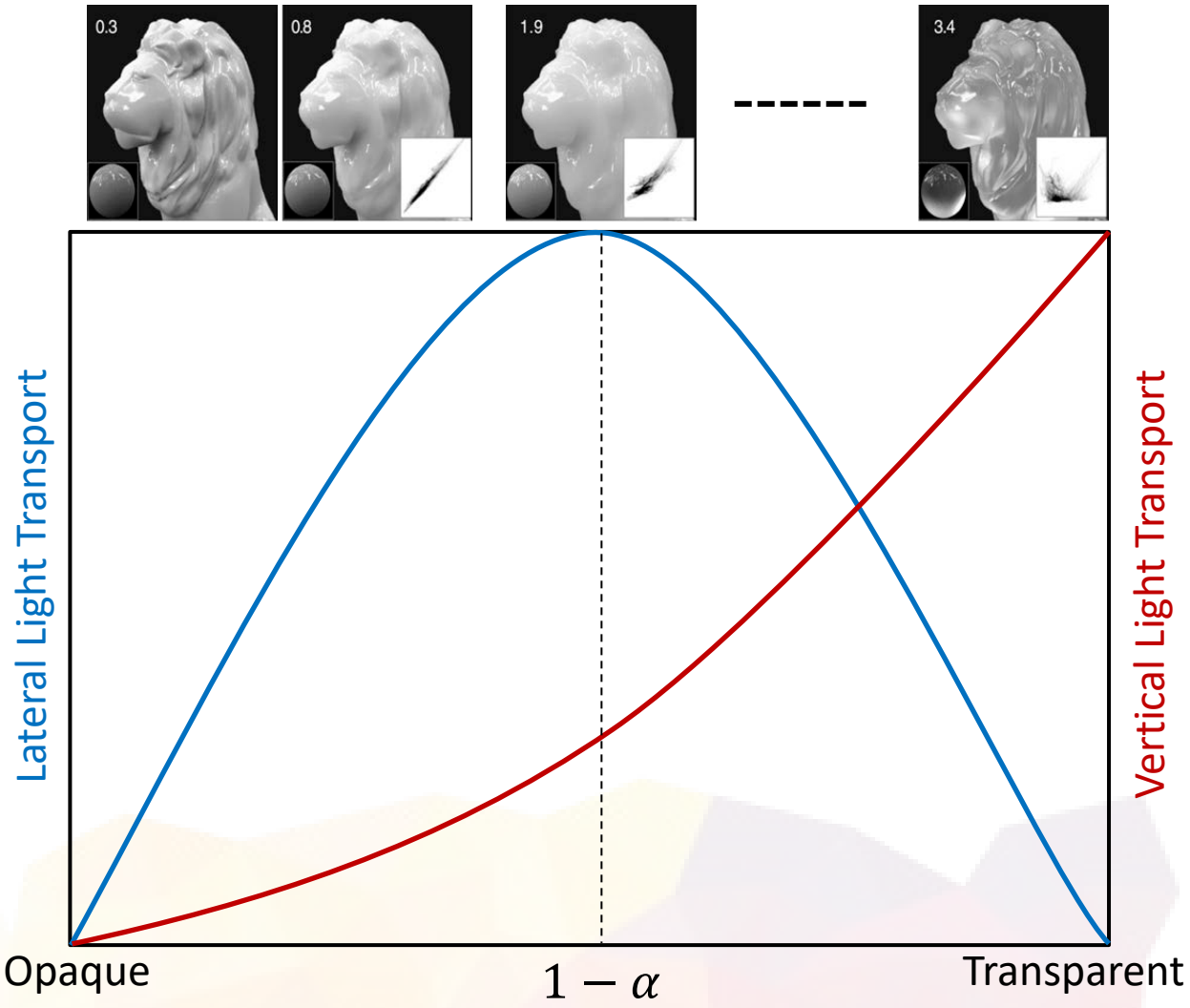
Fleming and Bülthoff, 2005

Motoyoshi, 2010

Visual cues induced by light transport

- is device independent
(based on reference materials)
- is measurable Material
(commercial spectrophotometers)
- is nearly perceptual uniform
(for reference materials)
- Add to RGB reproduce RGBA





Urban et al., 2017

Benefits of reproducing RGBA

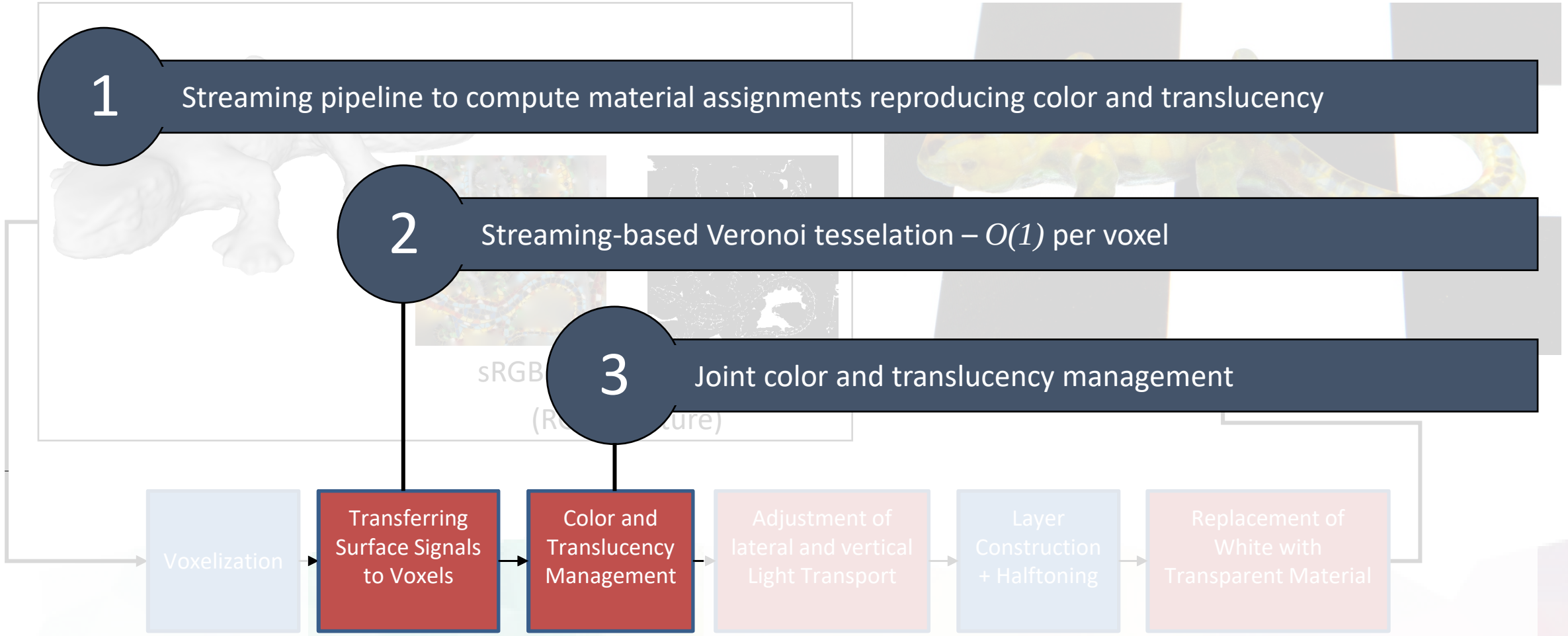
RGBA textured 3D models are

- supported by various 3D file formats

- can be created and modified by various 3D modeling tools

- RGBA textures can be modified by image processing tools

Streaming voxel-based pipeline



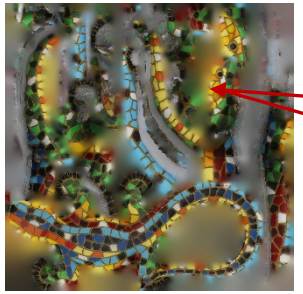
Approach: Minimize translucency error while preserving relative color

Precomputation: Conversion to voxels

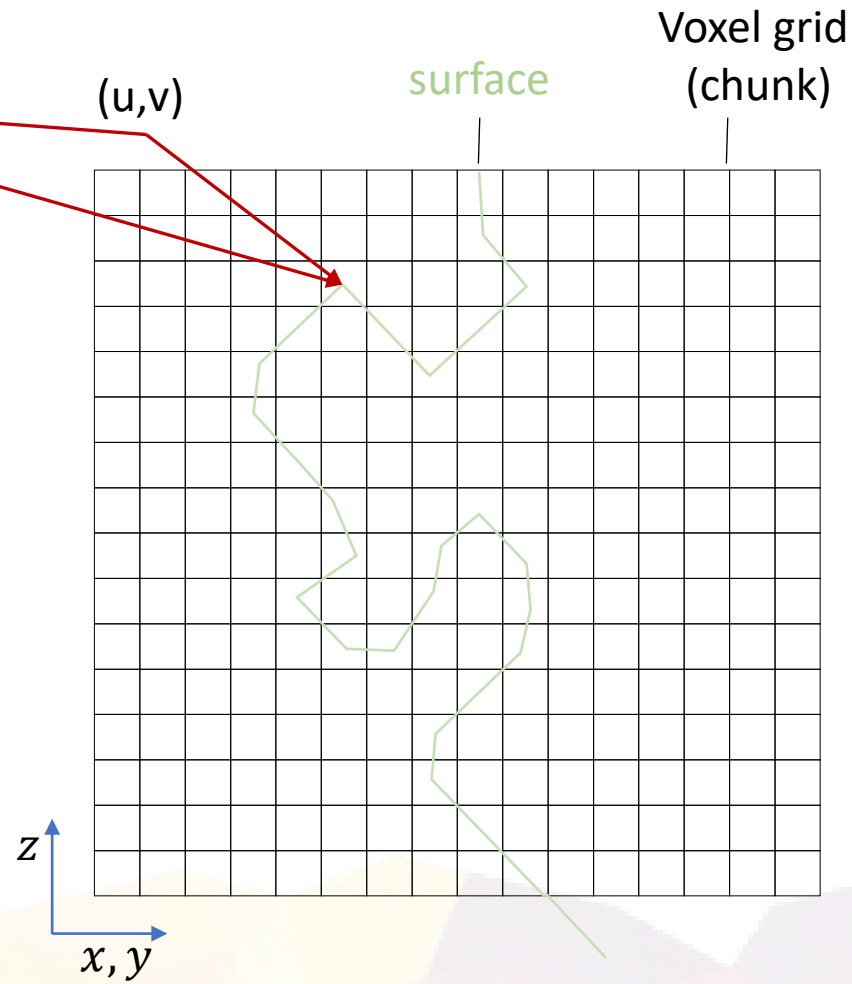
Per chunk (const. # slices):

6-separating surface voxelization + rasterization of RGBA values:

- Interpolate texture coordinates
- Sample texture
- Write RGBA to surface voxel



sRGBA

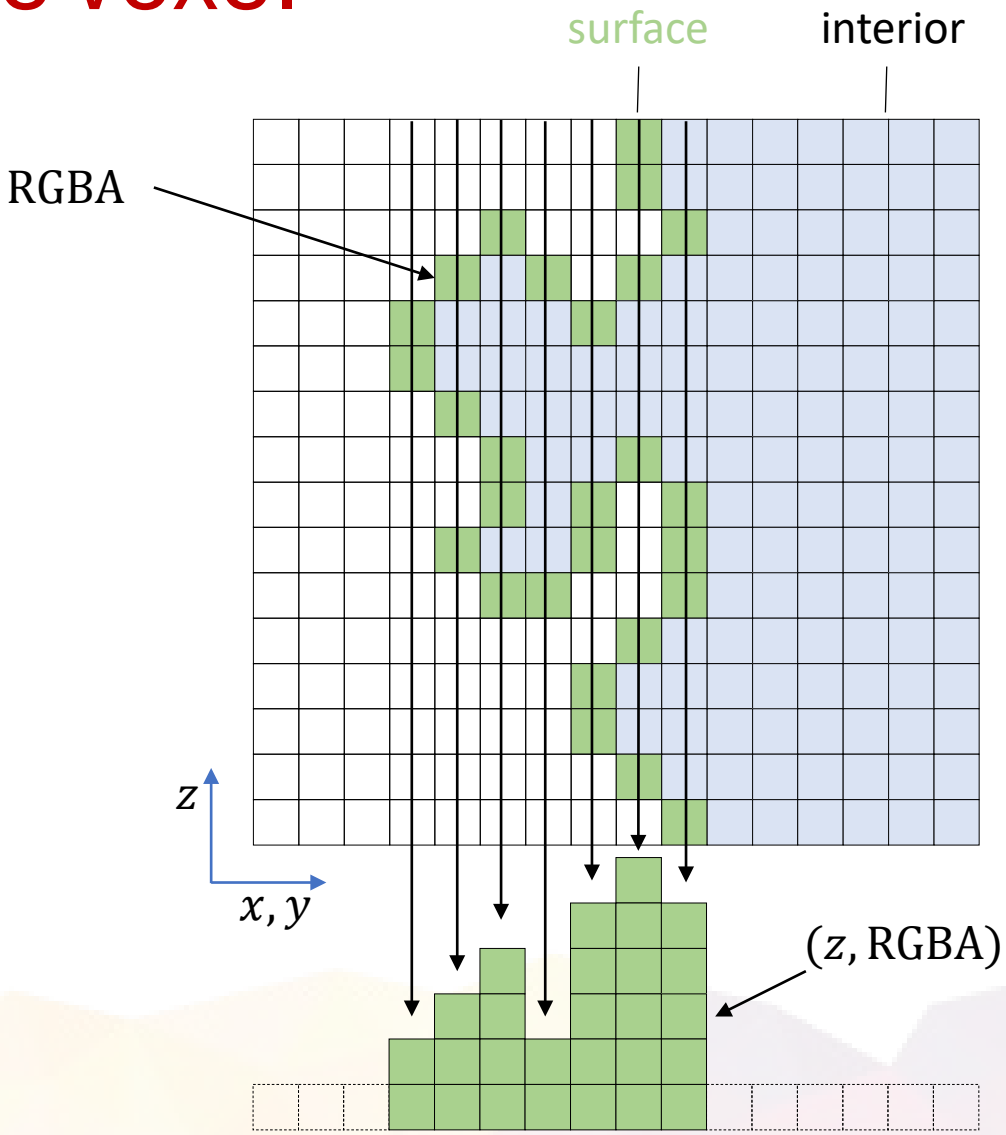


Precomputation: Sparse surface voxel representation

Per chunk:

From surface voxels, construct ordered lists of (z, RGBA) pairs

Construction in linear time w.r.t. # surface voxels



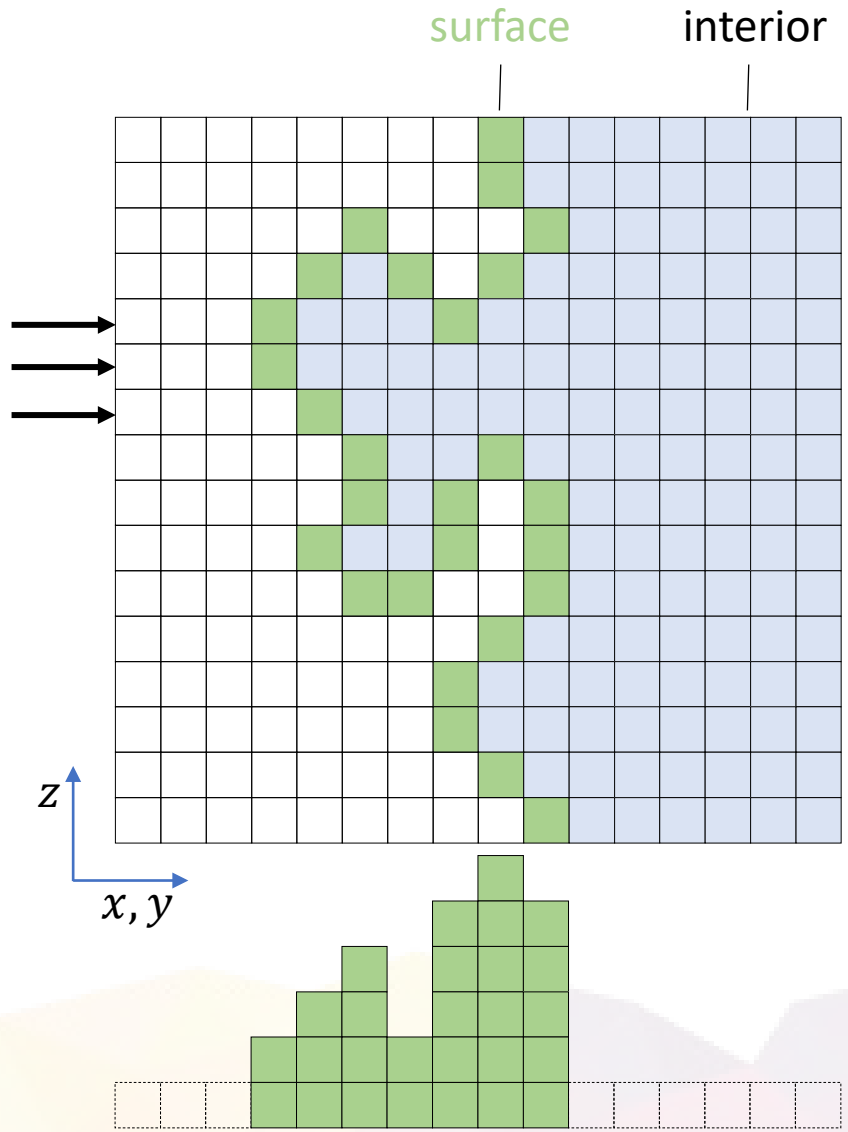
Transfer RGBA to all voxels

We need the following at **all** voxels:

- RGBA
- Distance to nearest surface voxel

Discrete Voronoi tesselation

- slice-wise computation in $O(1)$ time per voxel
- Incremental look-up of nearest voxel + distance in z
- 2D distance transform in (x, y)
  Felzenzwalb and Huttenlocher, 2012

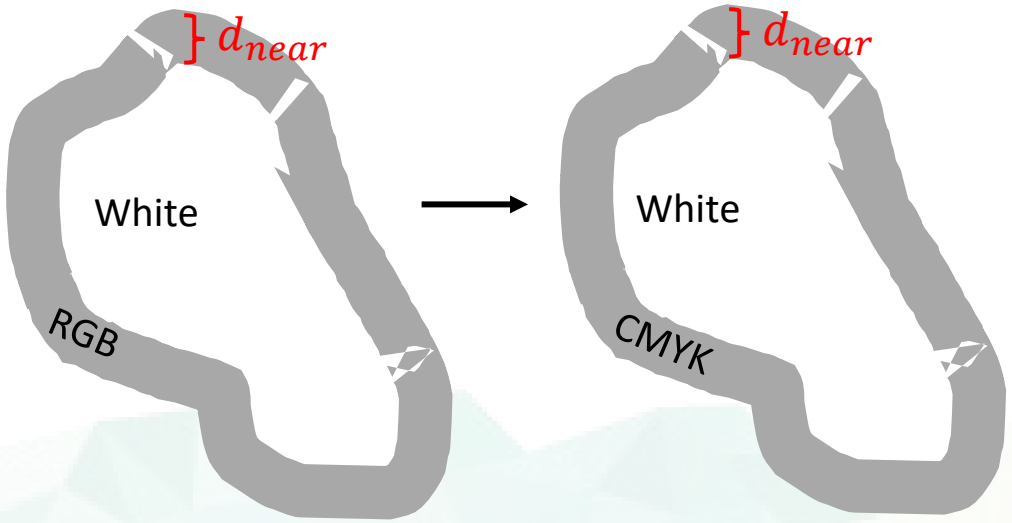


$$DT(\Delta z^2)(x,y) = \min_{x',y'} \left(\Delta z(x',y')^2 + (x'-x)^2 + (y'-y)^2\right)$$

RGBA → material assignments

Transferring Surface Signals to Voxels

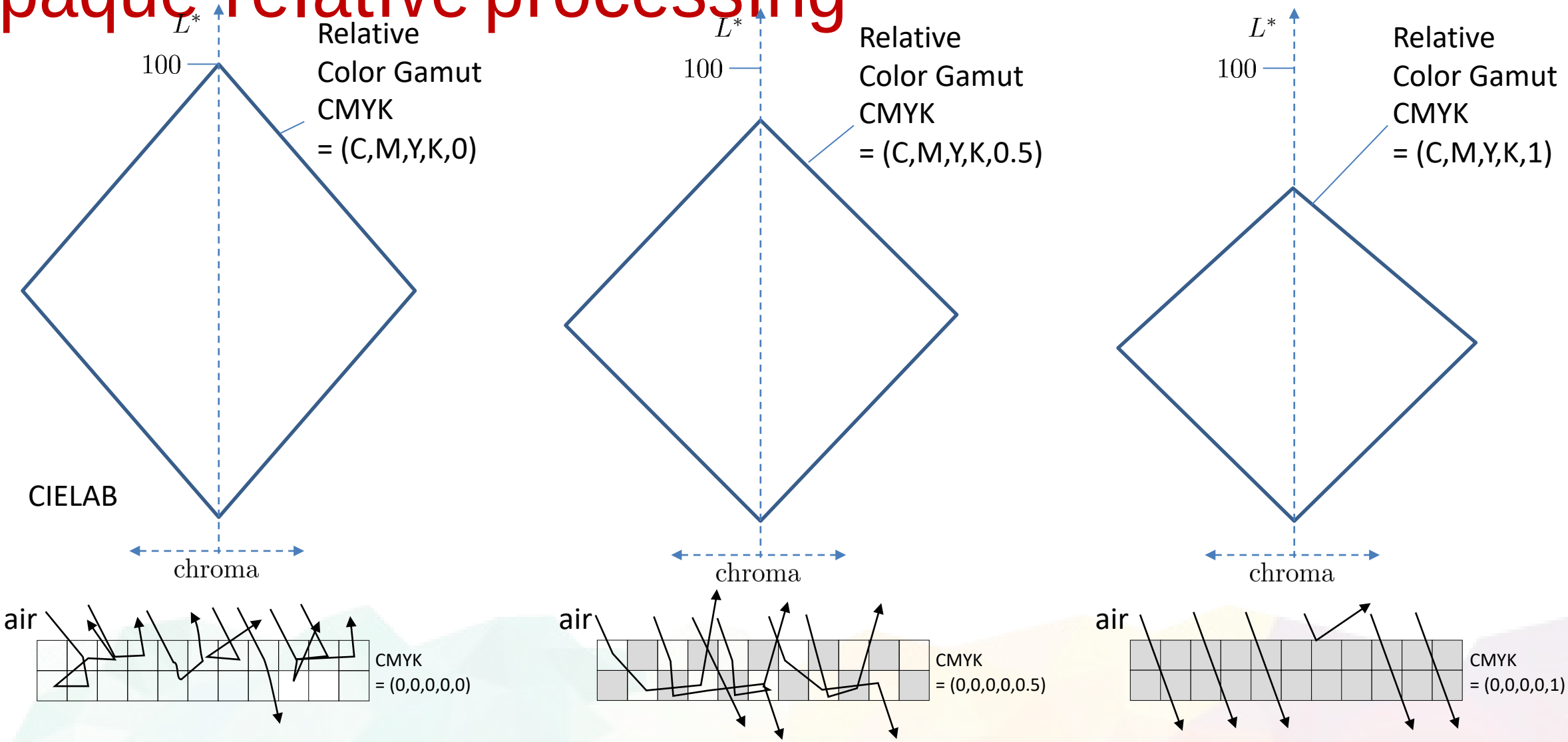
Beyond distance d_{near} look-up tables to convert RGBA to CMYK with (1,1,1,)



ratio of white and clear material:

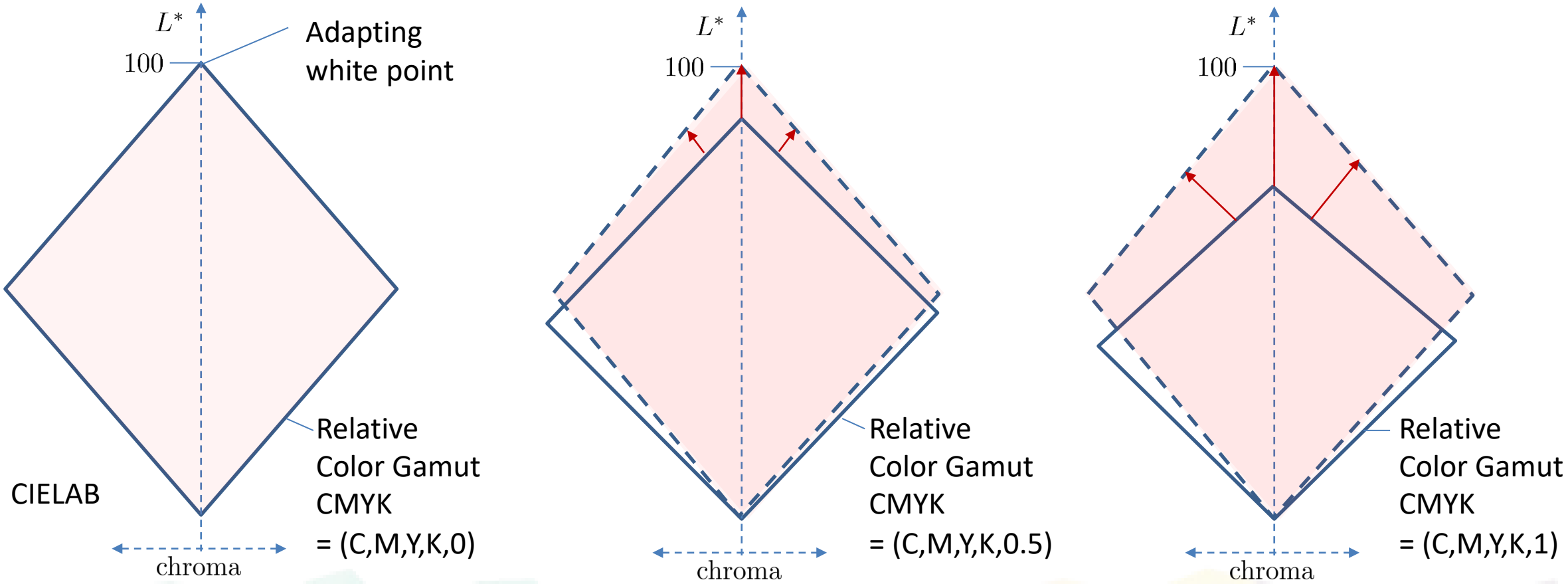
= 0 only white material
= 0.5 50% white / 50% clear material
= 1 only clear material

Opaque-relative processing



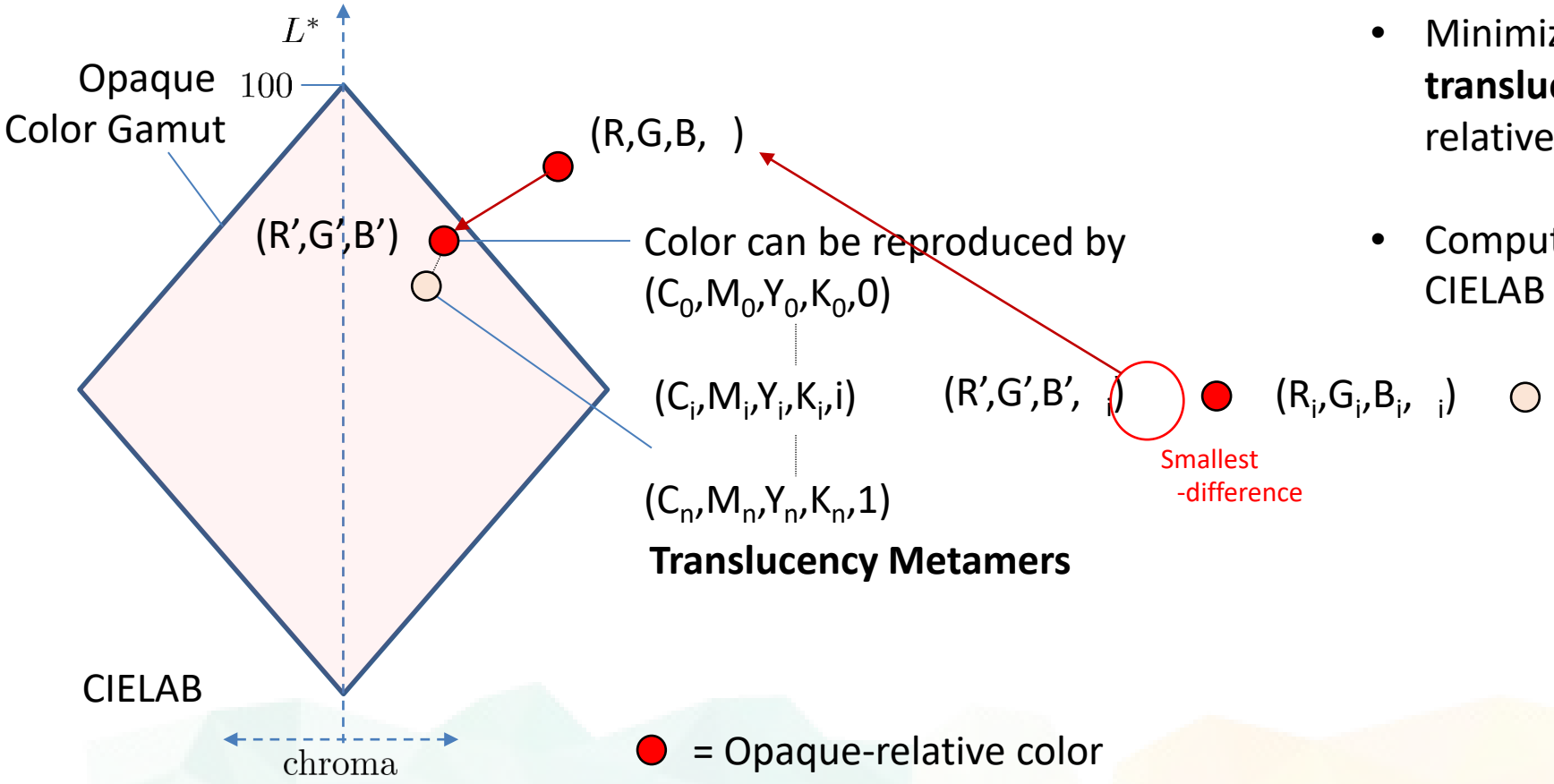
Voxelization → Transferring Surface Signals to Voxels → **Color and Translucency Management** → Adjustment of lateral and vertical Light Transport → Layer Construction + Halftoning → Replacement of White with Transparent Material

Opaque-relative processing



- Perform hue-preserving gamut expansion towards the opaque gamut [CMYK = (C,M,Y,K,0)]

Strategy for joint color and translucency reproduction



- Gamut mapping (hue-preserving)
- Minimize translucency difference within all **translucency metamers** for opaque-relative (R',G',B')
- Compute multi-dimensional look-up table CIELAB CMYK

= Opaque-relative color

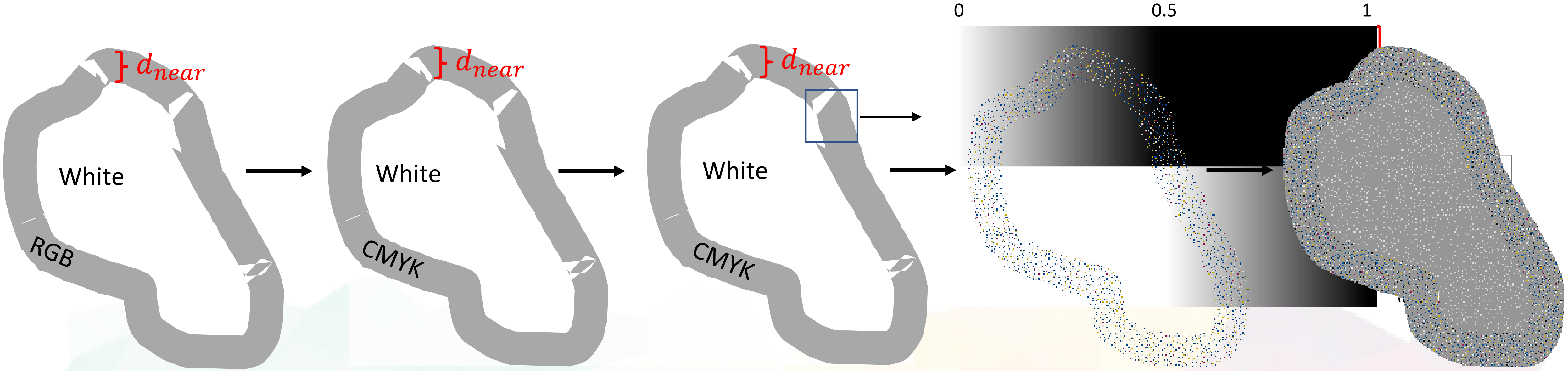
= Absolute color (same hue)

RGBA → material assignments

Layer Construction + Halftoning

…t of …h …aterial

Within d_{near} of the surface, … transport gamut (several overlapping lines printed on top of one another: "Replace white voxels …", "… d_{near} …", "… halftoning … CMYK … light transport gamut")

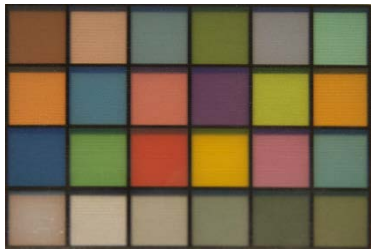
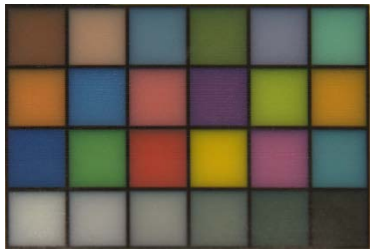
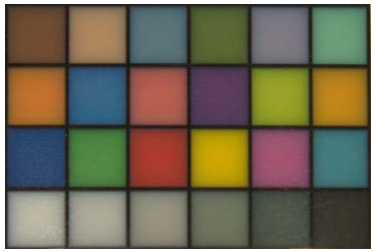
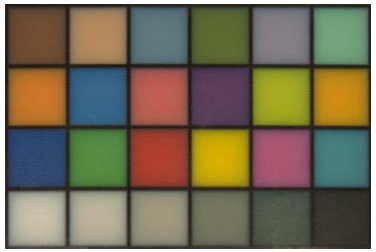
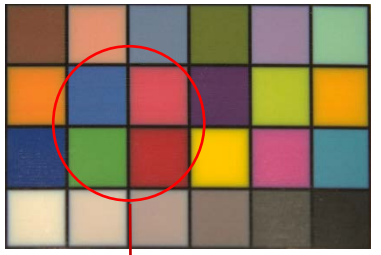


Adjustment of lateral and vertical Light Transport

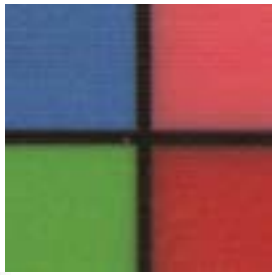
Layer Construction + Halftoning

Replacement of White with Transparent Material

Evaluation (Color Checker – Frontlit)

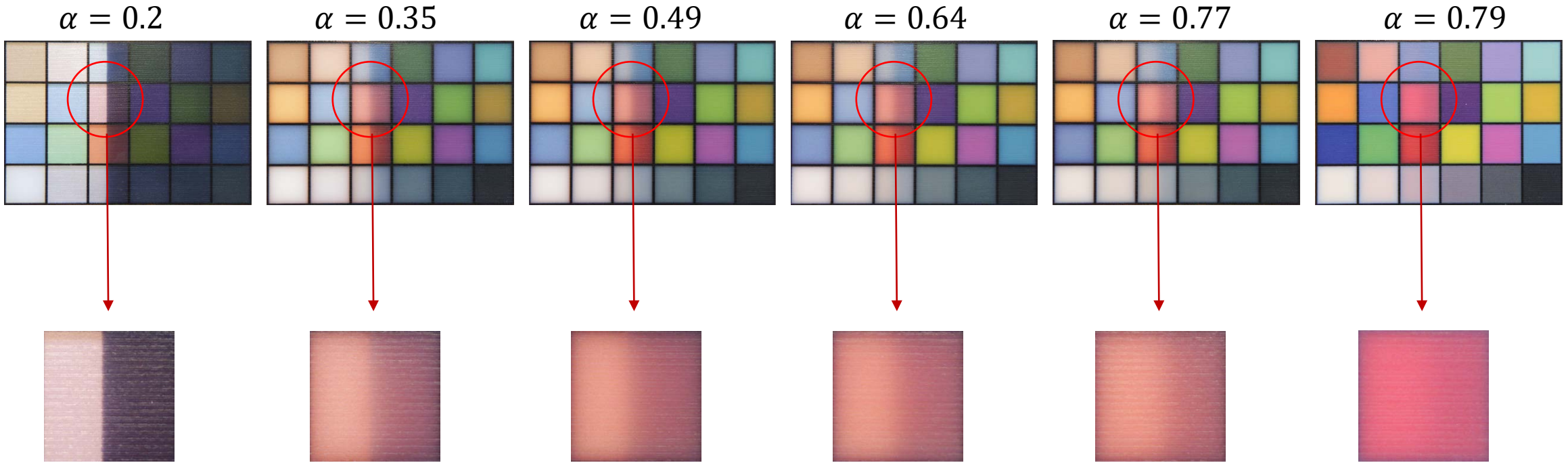
	$\alpha = 0.2$	$\alpha = 0.35$	$\alpha = 0.49$	$\alpha = 0.64$	$\alpha = 0.77$	$\alpha = 0.79$
						
E_{00}	5.5	3.7	3.7	3.0	3.2	2.3
a	0.7	0.23	0.15	0.06	0.02	0.06

Median errors: input vs. print



- Printer: Stratasys J750
- Optical Printer Model: Broadband cellular Neugebauer model
- Performance: Computation faster than printing

Evaluation (Color Checker – Backlit)



Evaluation (Color + Transparency)

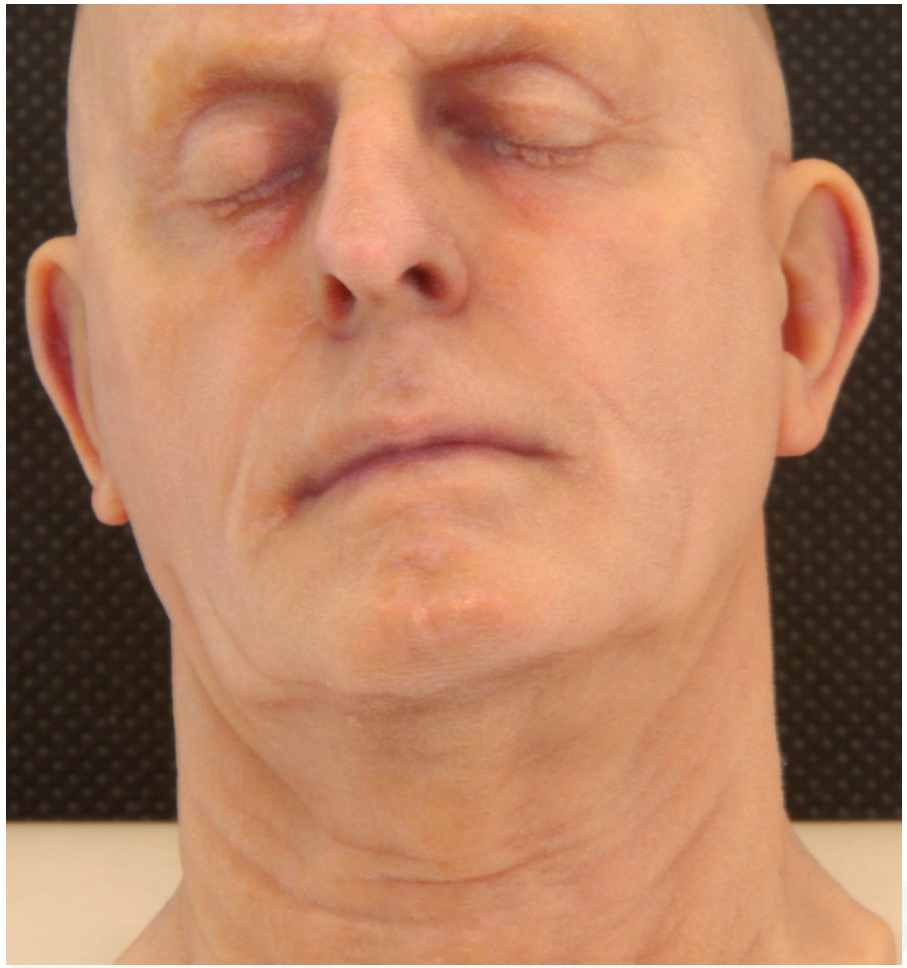
This and next slides:
Pictures of 3D prints,
not renderings

0.7mm (polished)

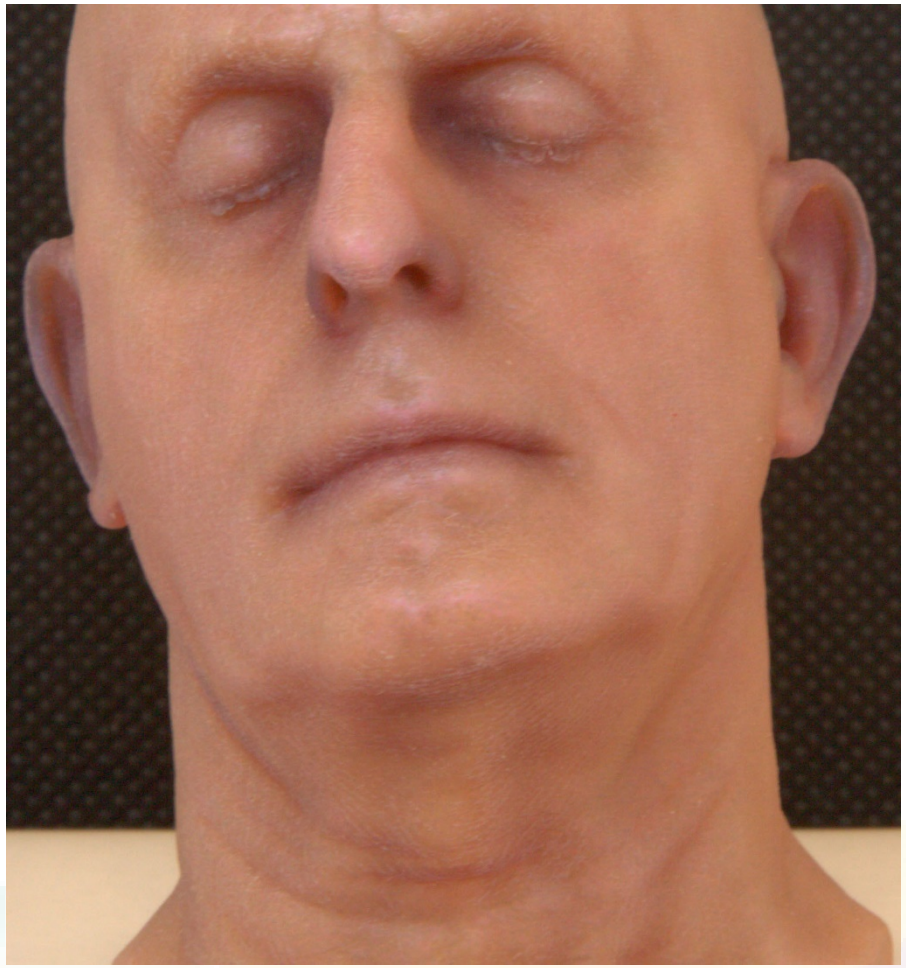
Revealing internal structures of complex models (28 sub-part). [skin: RGBA = (1,1,1,0)]

Translucency transitions

Control over translucent appearance



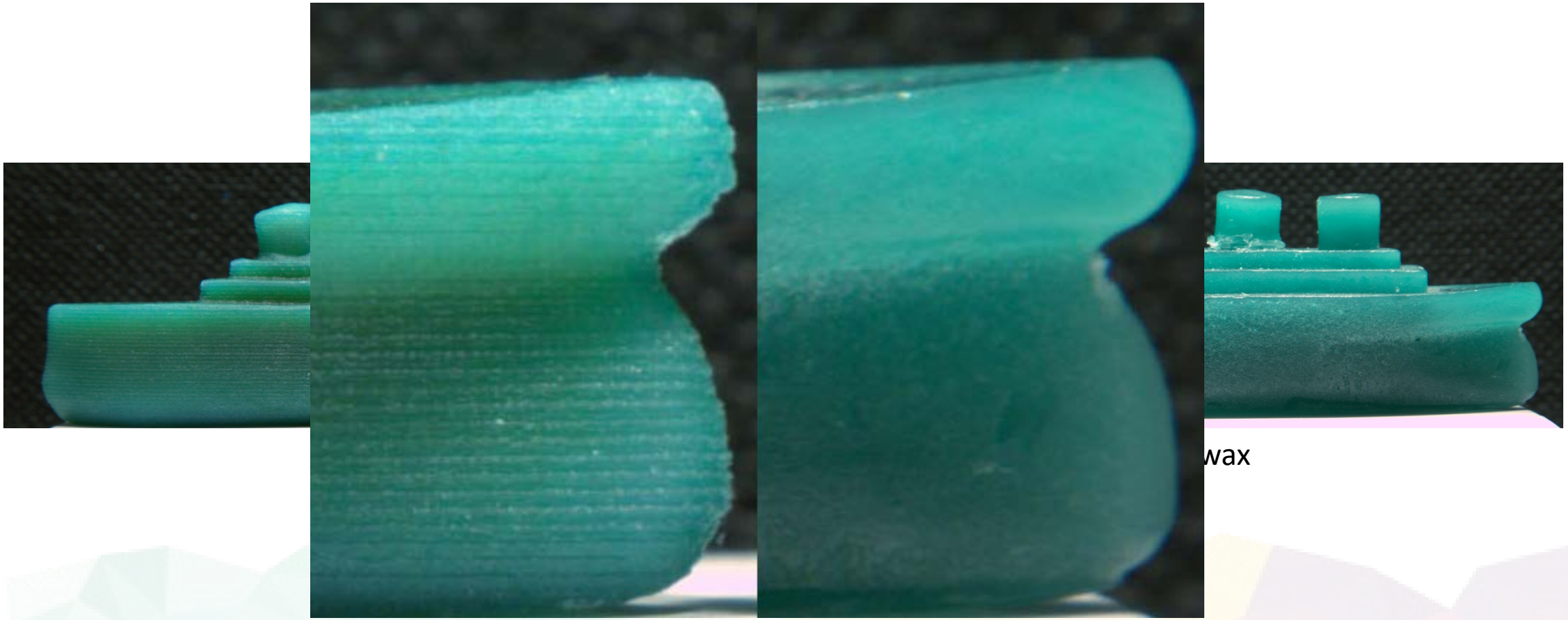
relative $\alpha = 1$



relative $\alpha = 0.54$

10 cm

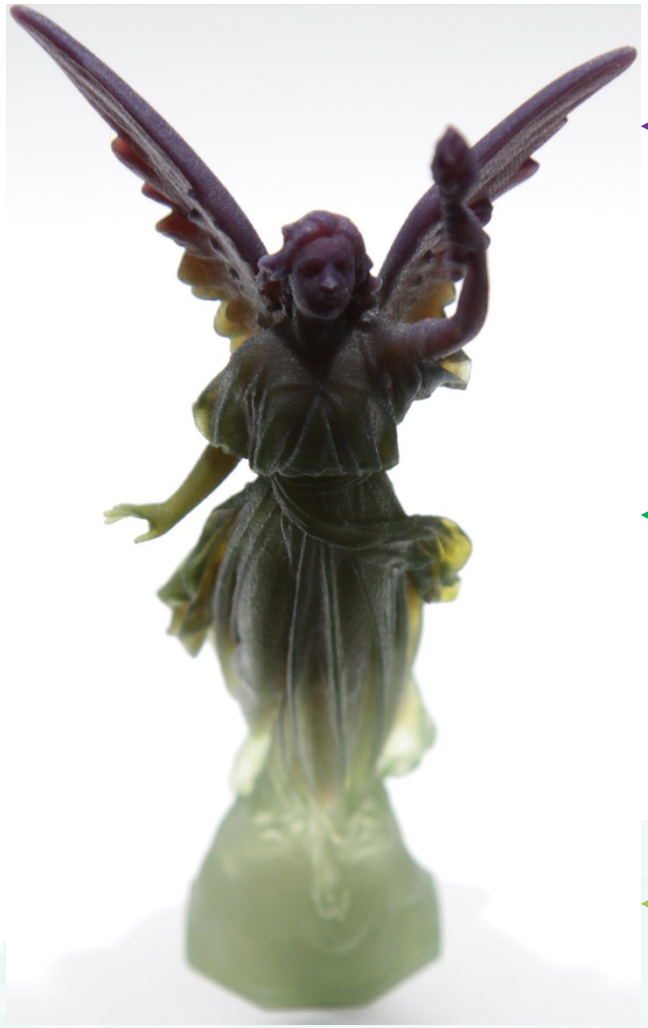
Reproducing translucent appearance

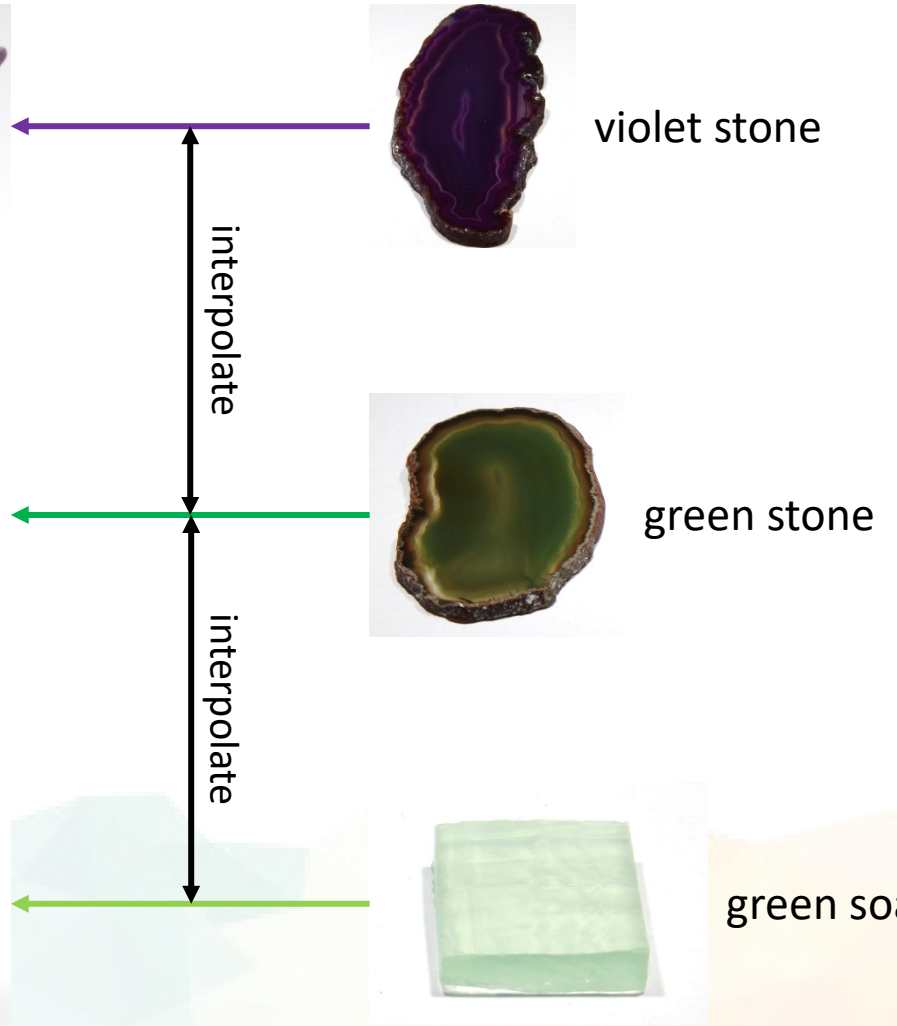


wax

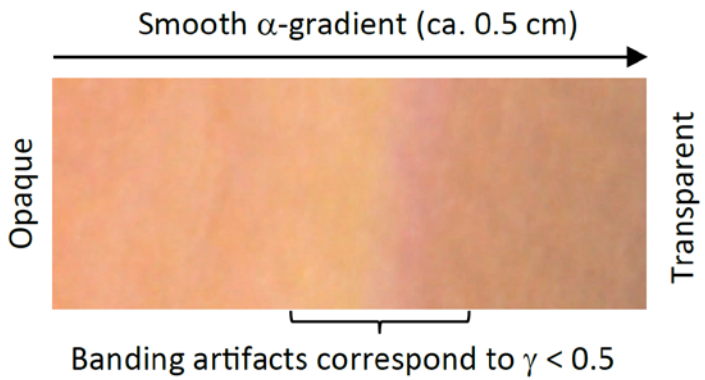
Increasing design space

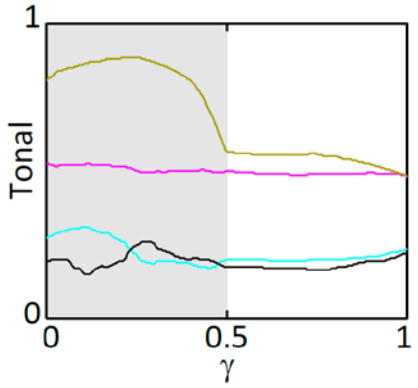
Reproducing translucent appearance





Limitations



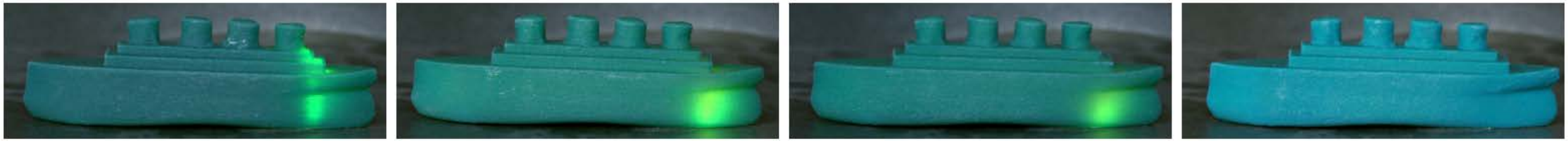


- Banding artifacts for gradients cover full range of α
- 1D translucency space: no directional dependencies
- Translucency as surface attribute: reduced degrees of freedom

Conclusions

- Fabrication of joint color and translucency
  - Efficient streaming algorithm to compute material assignments
  - Efficient streaming algorithm to compute Veronoi tesselations in *O(1)* time per voxel
  - Combined color and translucency management via translucency metamers (Opaque-relative Processing)

More information:
www.cuttlefish.de



Acknowledgements

Marco Dennstaedt and Andre Schulz (Fraunhofer IGD)

CultLab3D, Fraunhofer IGD
Hannes Urban

DIG:ED GmbH
Alphacam GmbH
Laika Studios
Stratasys

Anonymous reviewers

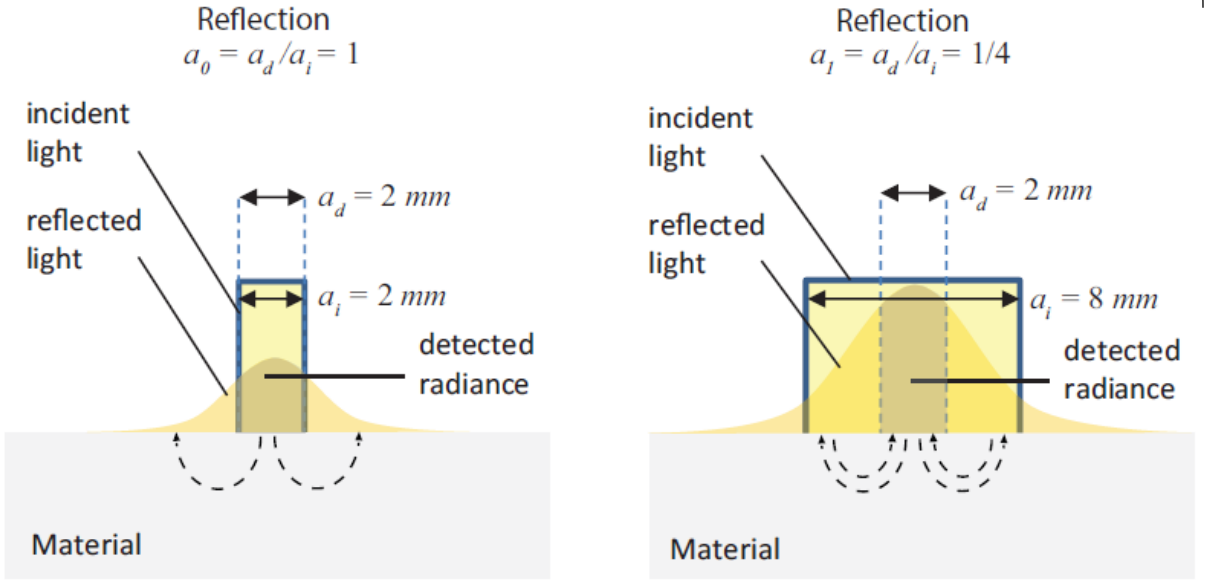
Funding:
Fraunhofer Grant Attract 008-600075,
Scan4Reco project funded by EU Horizon 2020 Framework Programme under Grant No. 665091

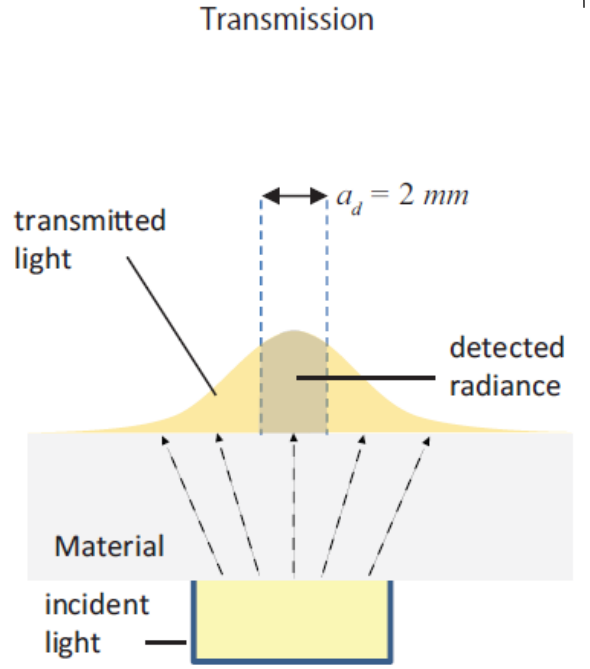
Backup Slides

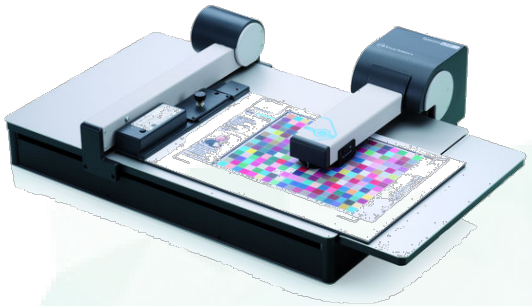
Measuring Lateral and Vertical Light Transport

Lateral light transport

Vertical light transport



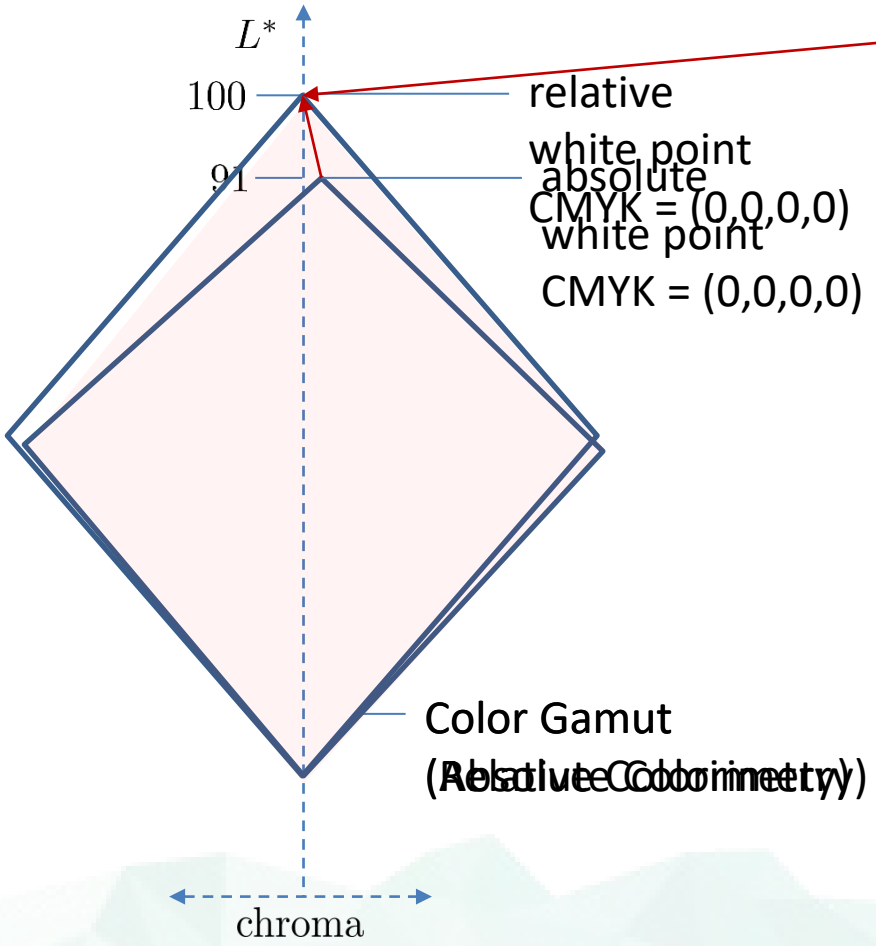




Barbieri Spectro LFP:
Commercial reflection/transmission spectrophotometer
Reflectance 45/0, Transmittance d/0

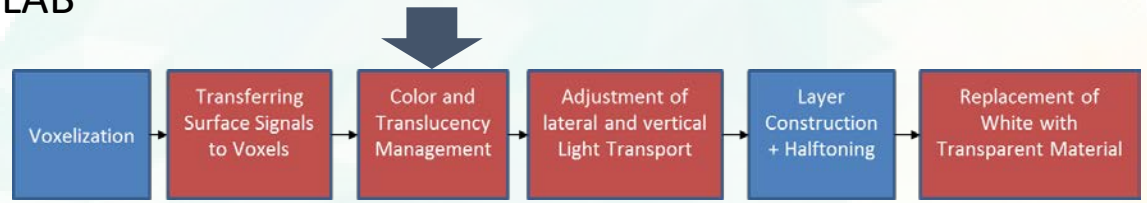
Measuring Lateral and Vertical Light Transport

Media Relative Processing



- sRGB = (1,1,1) CIELAB = (100,0,0) is not printable
- Scale colors by mapping white point to CIELAB = (100,0,0)
- Ensures that sRGB = (1,1,1) maps to CMYK = (0,0,0,0)
- Accounts for white point adaptation of the human visual system

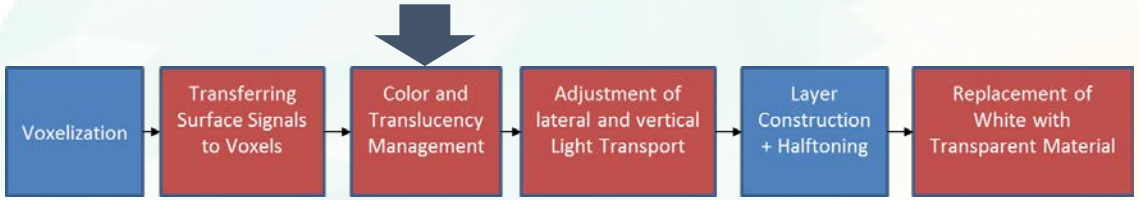
CIELAB



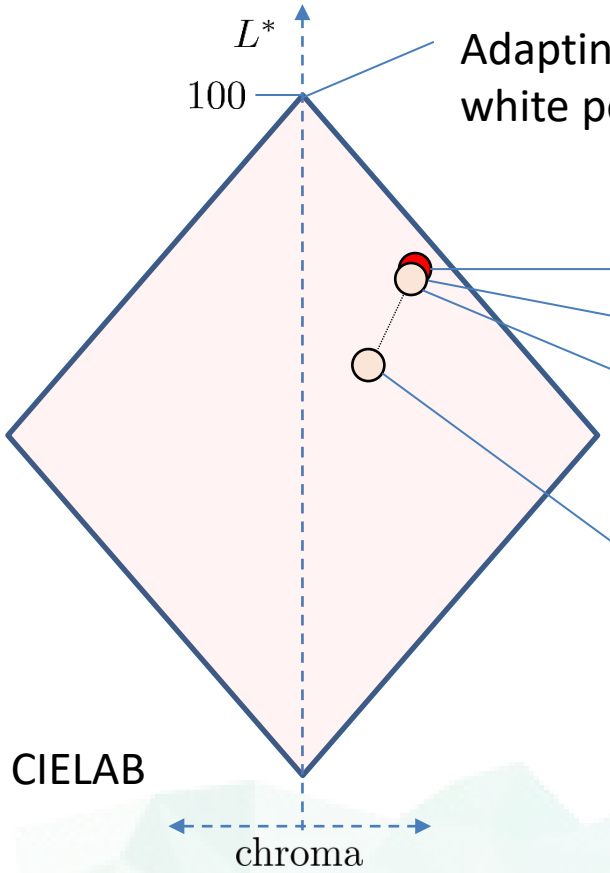
Media Relative Processing

- = 0 (fully transparent) and = 1 (fully opaque) are usually not printable for CMYK = (0,0,0,0,)

- Scale linearly to ensure that = 0 for CMYK = (0,0,0,0,1) and = 1 for CMYK = (0,0,0,0,0)



In-Gamut Colors in Opaque-Relative Representation



CIELAB

Adapting white point

Color can be reproduced by

$(C_0,M_0,Y_0,K_0,0)$ (R_0,G_0,B_0,A_0)

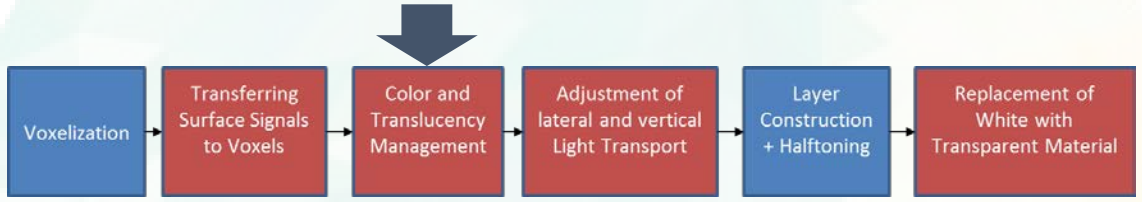
$(C_1,M_1,Y_1,K_1,1)$ (R_1,G_1,B_1,A_1)

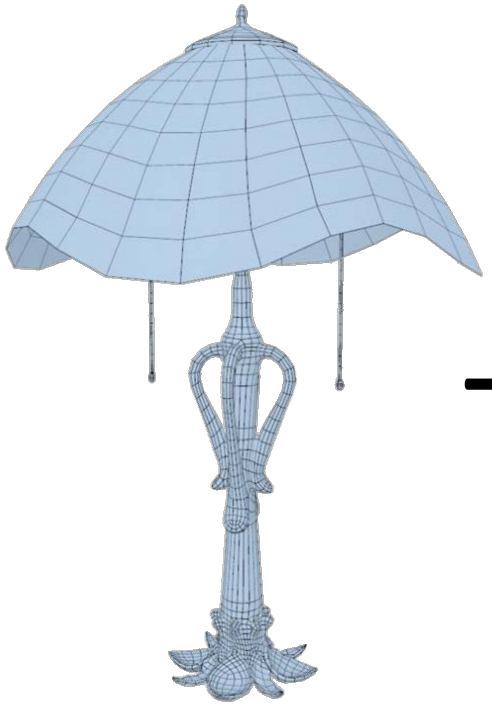
$(C_{255},M_{255},Y_{255},K_{255},255)$ $(R_{255},G_{255},B_{255},A_{255})$

Translucency metamers w.r.t. opaque-relative colors

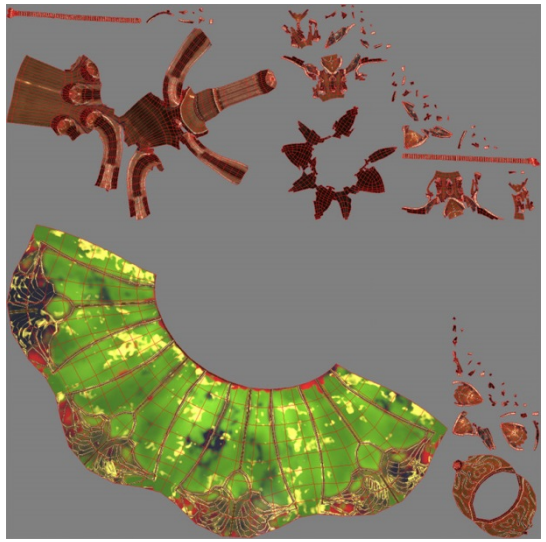
= Opaque-relative color

= Absolute color (same hue)





+



Design / modify
A in RGBA-texture

Source: turbosquid.com