3D Appearance Management using iccMax

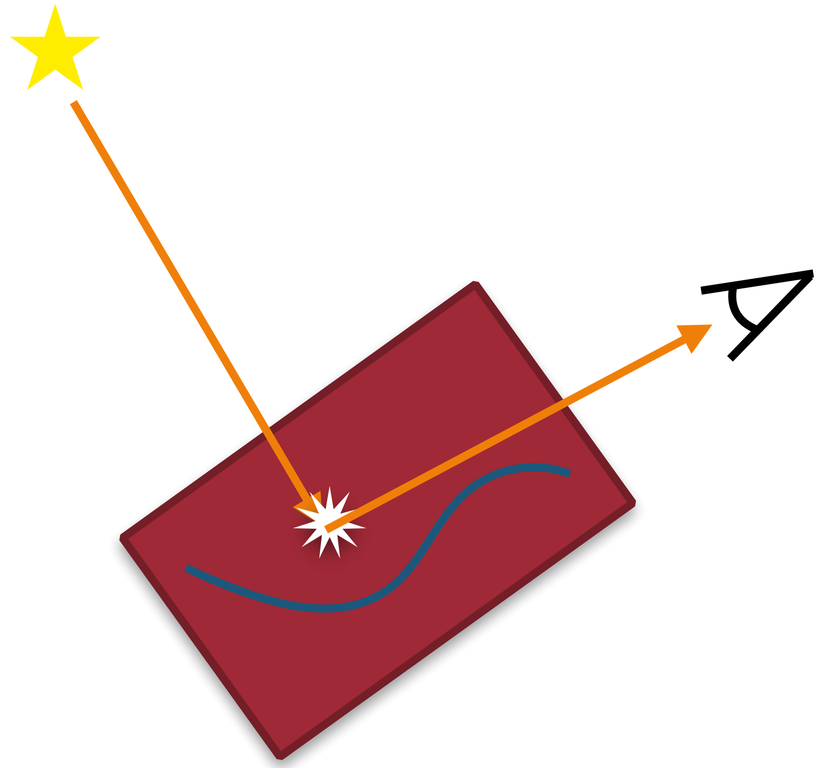
James Vogh

X-Rite

May 6, 2016
Taipei

Surface Appearance

Surface appearance is controlled by both illumination and viewing angles



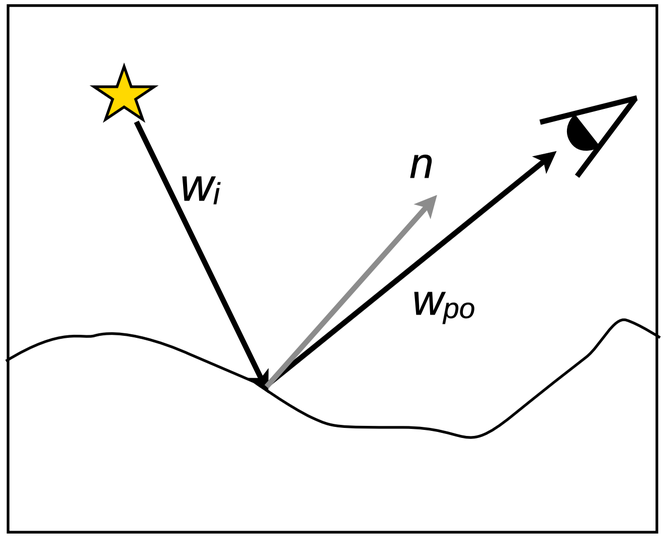
Example & Demo

Why Add Surface Appearance to iccMAX?

- To include information in the profile that can describe the appearance for arbitrary lighting and viewing conditions
- Allows soft proofing under arbitrary conditions
- Allows the use of different measurement geometries when connecting profiles

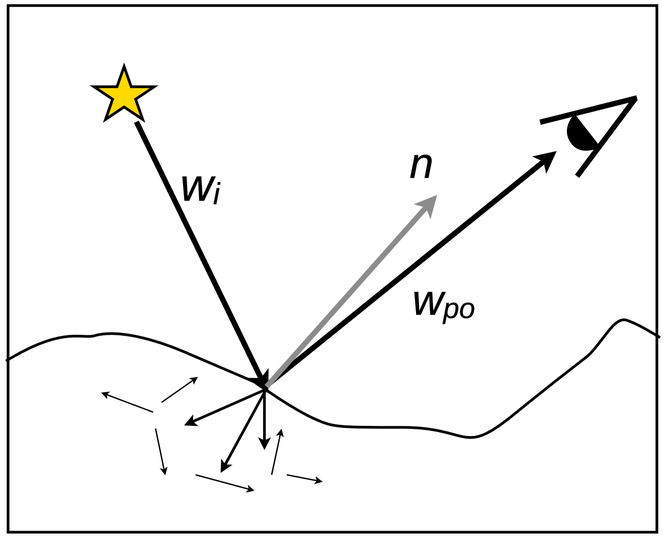
Bidirectional Reflectance Distribution Function

- A Bidirectional Reflectance Distribution Function (BRDF) is a function that specifies the reflectance of a surface for a particular light (position & color) and a viewer (position)

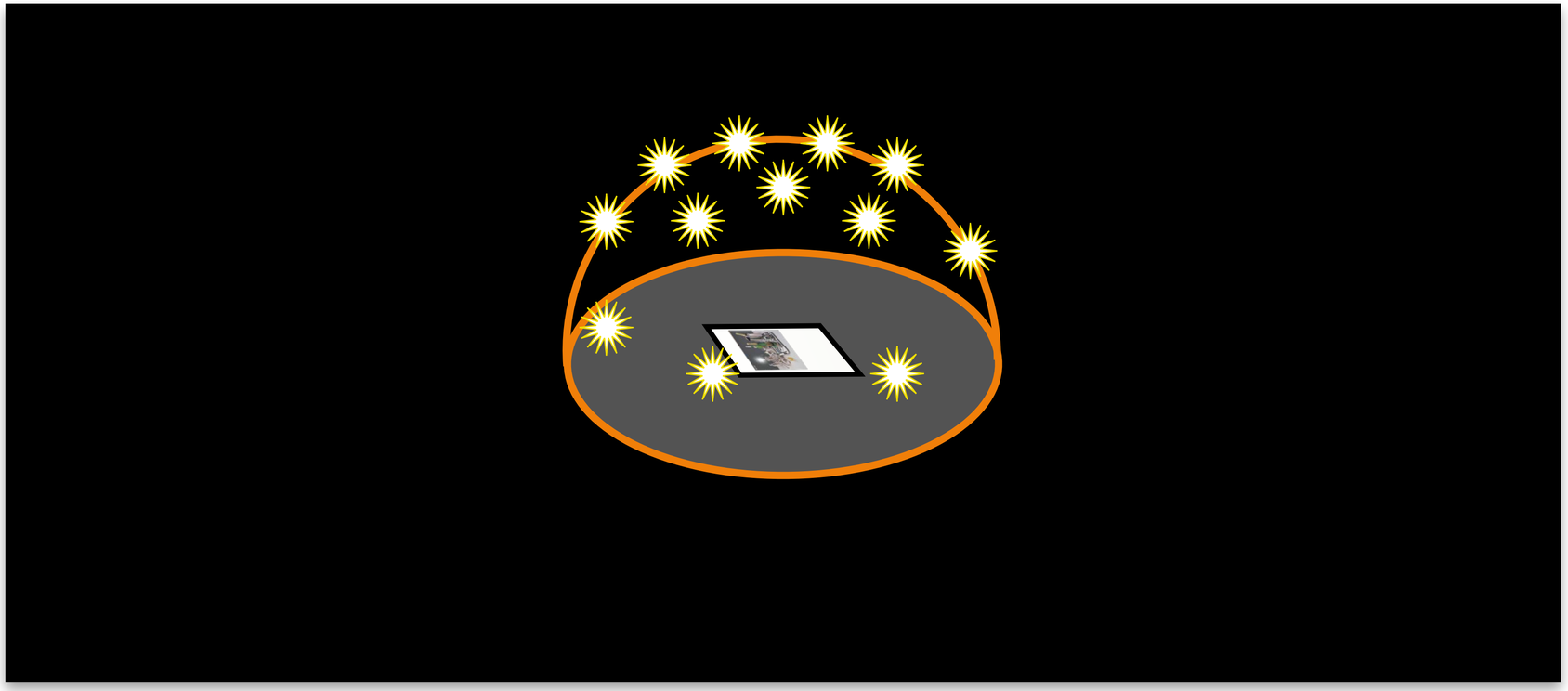


Related Functions

- Can describe 2D texture
- Can describe scattering of light once it enters the surface



Acquiring Surface Appearance Measurements



- One method is to use a dome with multiple light sources and multiple cameras

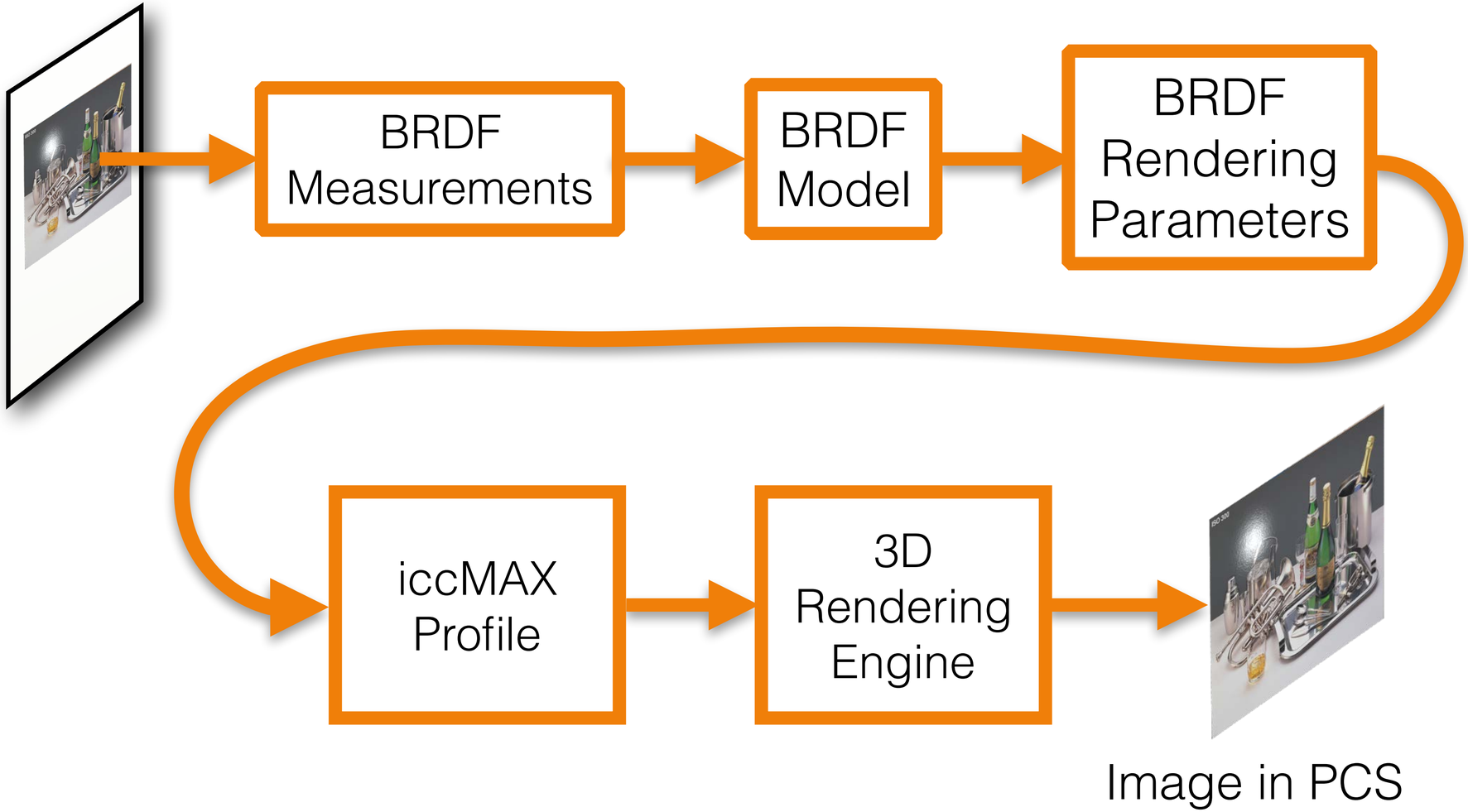
BRDFs in iccMAX

- BRDF information is optional in a profile
- In first version of iccMAX, BRDF information is allowed for output class profiles and named color profiles
  - Directional Tags support similar capabilities for displays
- The BRDF information is in two forms:
  - BRDF parameters for various BRDF models
    - Suitable for use with 3D rendering applications
  - Direct implementation using multiProcessElementType tag
    - Transforms return reflectance when given illumination angle and viewing angle
- Normal map or height map is used to specify surface texture

BRDF Parameters & Direct BRDF Implementation

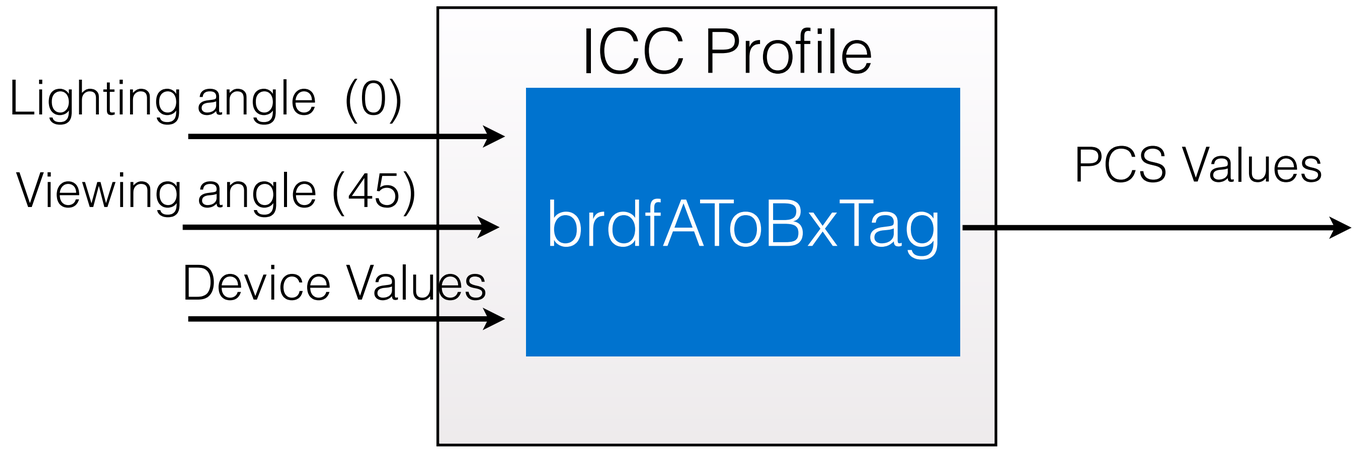
- A profile can have one type or both types of BRDF representation
- Parameterized BRDF
  - Accuracy is only as good as the fit of the selected model to the data
  - Can easily be used for 3D rendering
  - Can be implemented with a very small tag
- Direct BRDF Implementation
  - Uses a multiProcessElementType tag to directly implement the BRDF model
  - Accuracy can be very high
  - Processing speed of the tag might by poor and might not be suitable for use with 3D rendering

Example Usage of Parameterized BRDF with iccMAX



Example Usage of Direct Implementation of BRDF with iccMAX

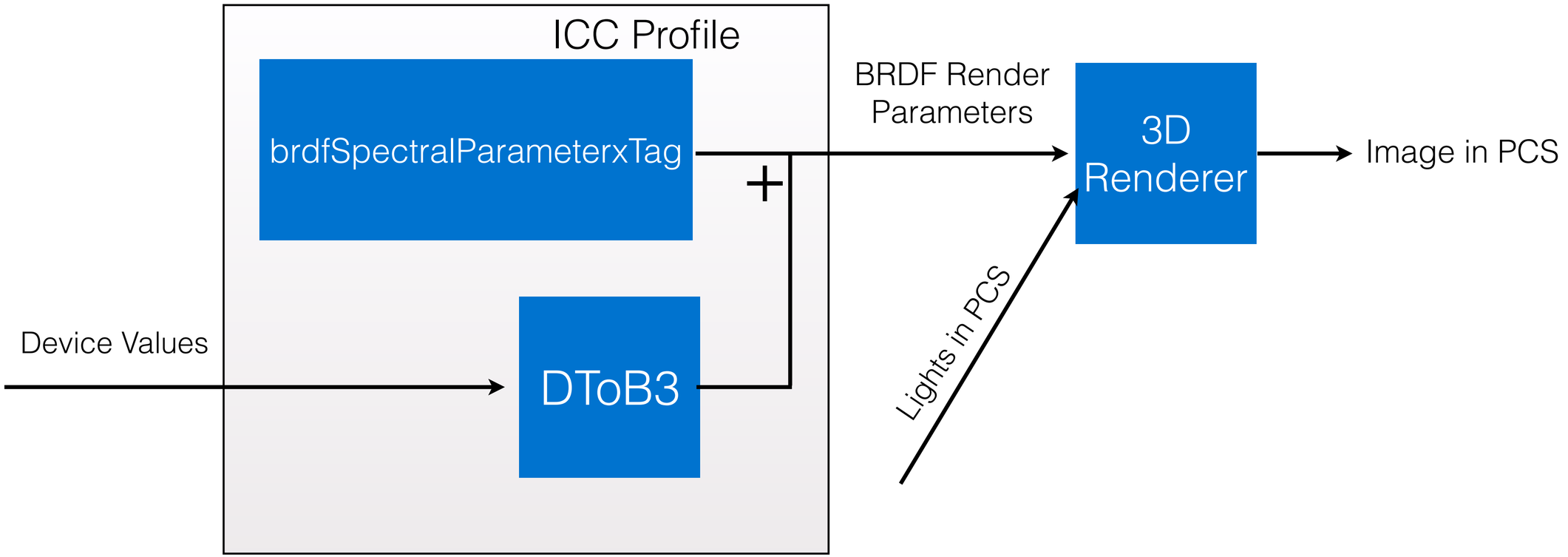
- Want to get measurements for 0/45 from profile that uses spherical geometry



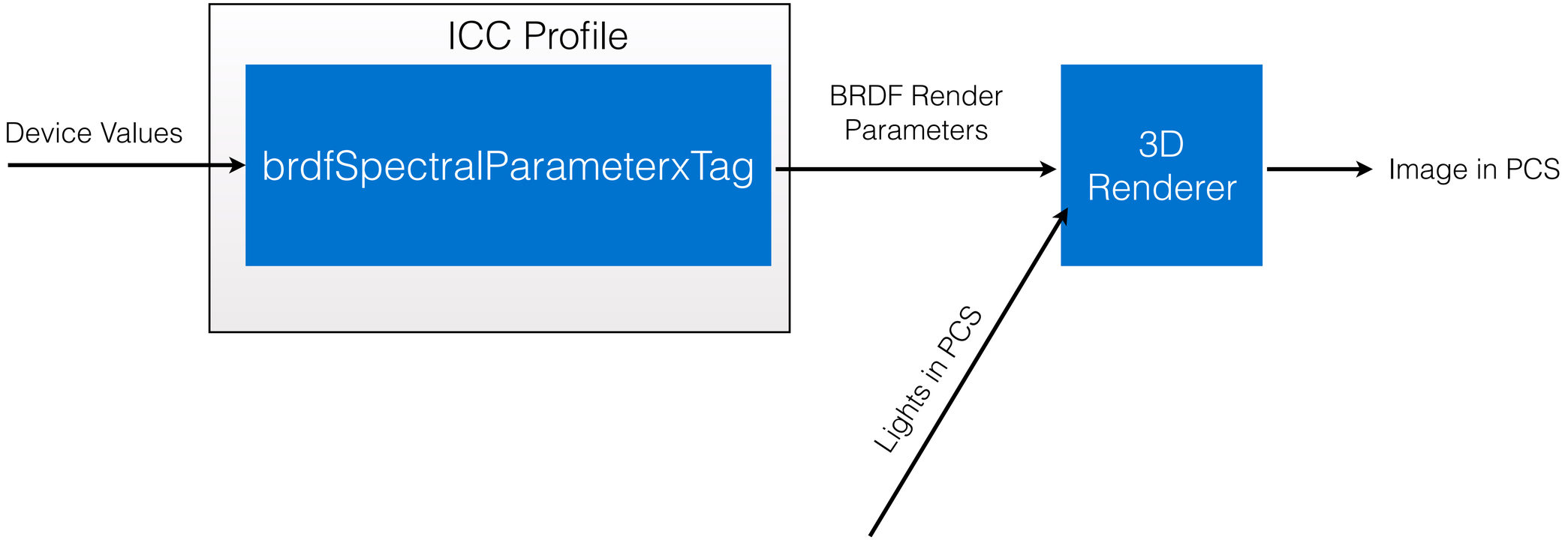
Parameterized BRDF

- iccMAX supports the following BRDF models
  - Blinn-Phong, Ward, Cook-Torrance, and Lafortune
- Two forms of Parameterized BRDF
  - Monochrome
    - One set of BRDF parameters for all device values
    - Allows for very simple inclusion of BRDF information in the profile when one set of parameters is sufficiently accurate
  - Chromatic
    - Parameters are a function of device values
      - Characterizing printing with a metallic ink would be a good use case for this form
  - Can create spectral or colorimetric parameters depending on the type of tag

Monochrome Parameterized BRDF

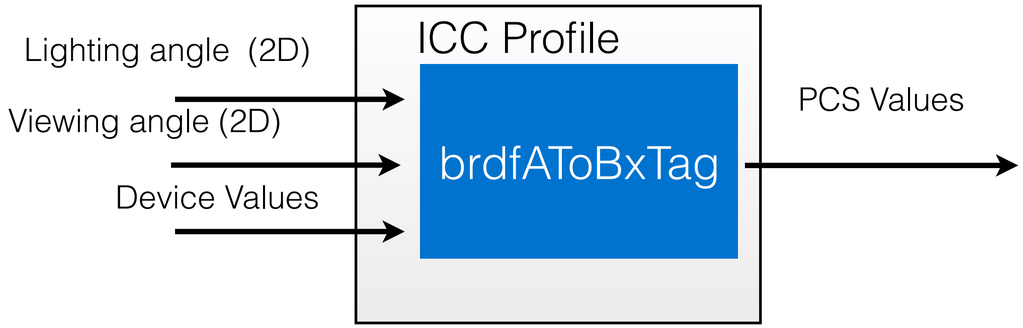


Chromatic Parameterized BRDF



Direct BRDF Calculation

- Transform is implemented as multiProcessingElement
- Input
  - 2D lighting angle (azimuth & zenith)
  - 2D viewing angle (azimuth & zenith)
  - Device values
- Output can be colorimetric or spectral depending on the type of tag
  - A profile may contain both spectral and colorimetric tags

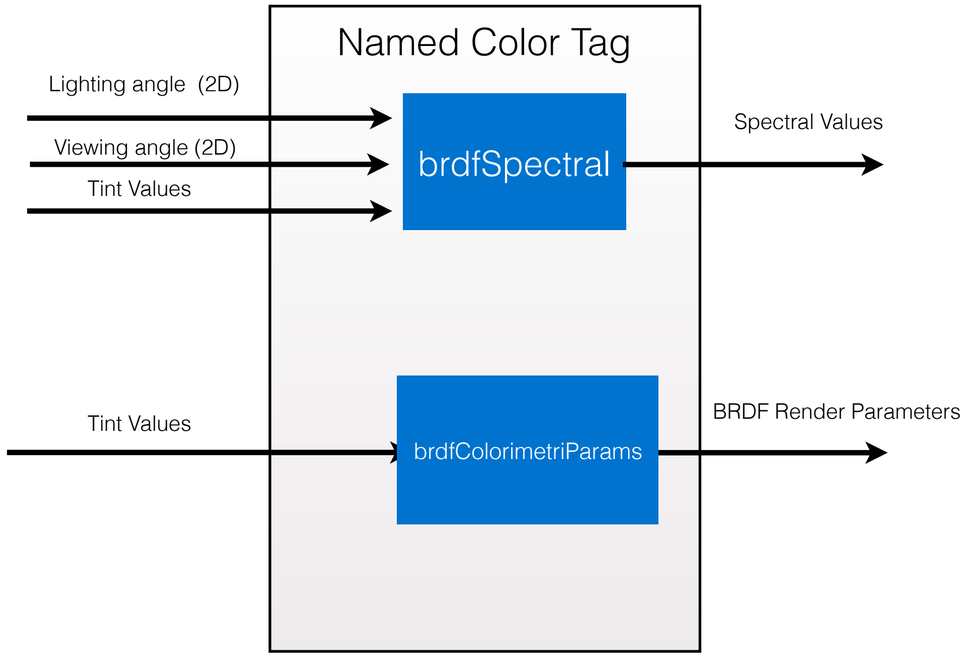


Compute Device Values

- Transform is implemented as multiProcessingElement
- Input
  - 2D lighting angle (azimuth & zenith)
  - 2D viewing angle (azimuth & zenith)
  - PCS values
- Output is device value
  - A profile may contain both spectral and colorimetric tags

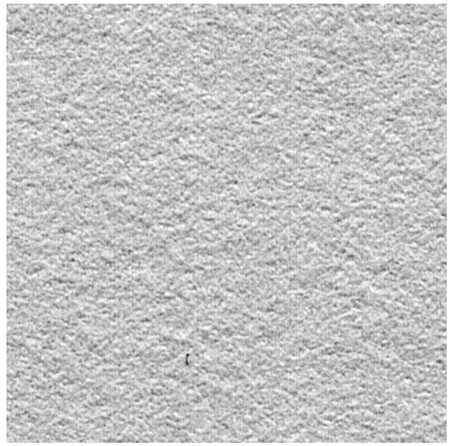
Named Color Profile

- A namedColorStructure can optionally contain BRDF Parameter tags and/or tags that directly implement the BRDF
  - Can contain colorimetric and/or spectral tags
  - BRDF for different tint values is supported
    - Monochrome and chromatic are supported for BRDF Parameter type

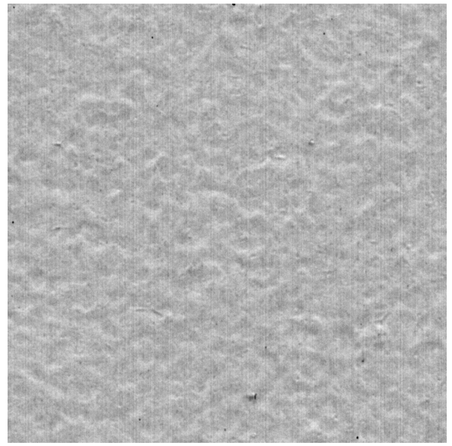


Texture

- BRDF doesn't provide information about the texture of a substrate.
- Example substrate textures (enhanced contrast)



Matte Paper

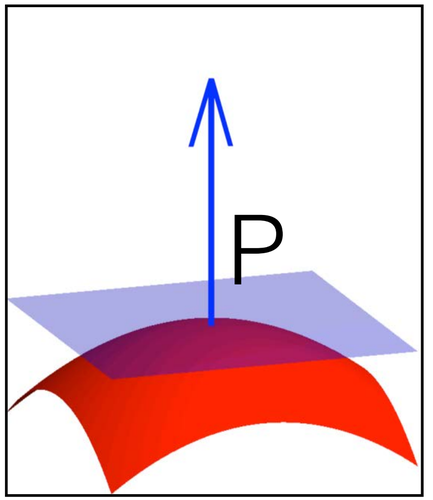


Premium Luster Paper

- Can be represented with a Normal Map or a Height Map

Normal Map

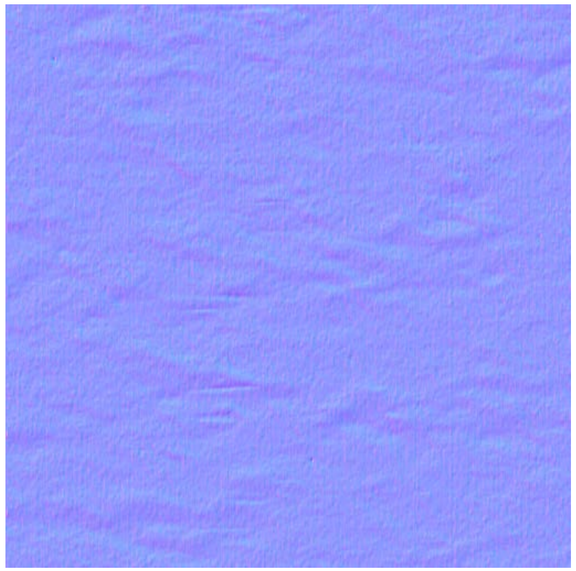
- The surface texture of a substrate can be represented with a normal map
- A surface normal is the vector that is perpendicular to the tangent plane of a surface at point P



- A normal map is a set of surface normals across a surface
- Normal map represents how the normal varies across the surface

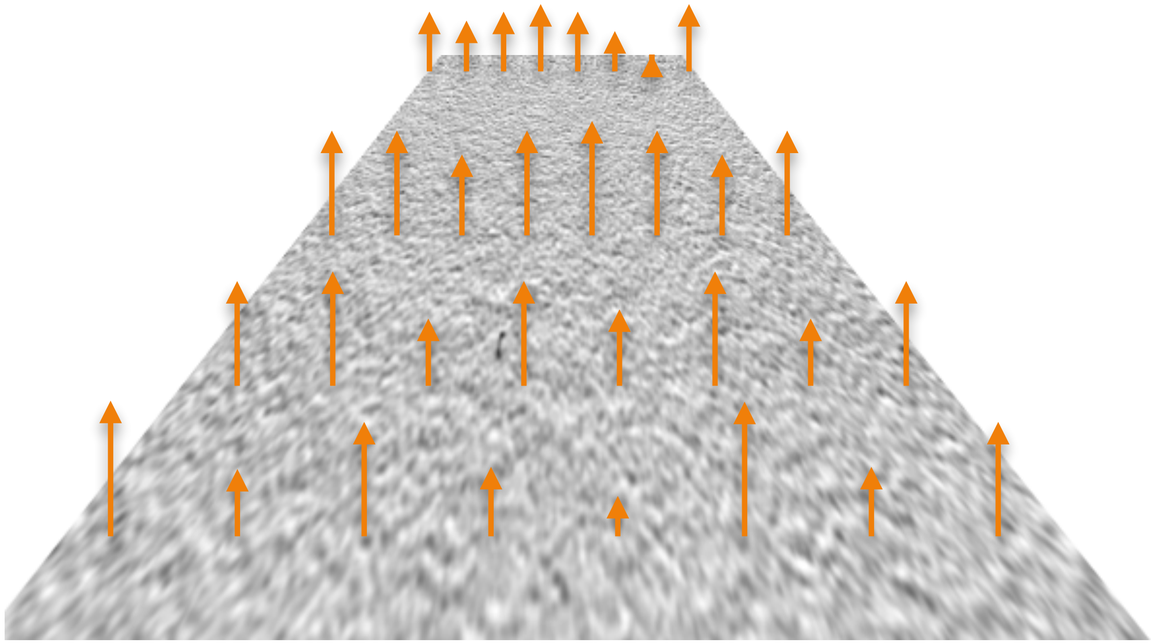
Normal Map

- In ICC labs a normal map is optional, even if the profile contains BRDF information.
- Contains spatial dimensions of the map for correct rendering
- Stored as PNG or TIFF



Height Map

- Represents the texture of a 2D surface with height information
- Each location in 2D map is a height value
- Can be used with displacement mapping to alter geometry and produce correct outlines and shadows
- Requires more work by renderer when render is performing bump mapping
- Stored as PNG or TIFF in the profile



Height Map & Normal Map

- Shall include spatial dimensions in the image header
- Should be seamless
  - Don’t want visible borders when textures are tiled
- Should be power of 2 size image size for use in 3D rendering applications

Normal Map Consideration

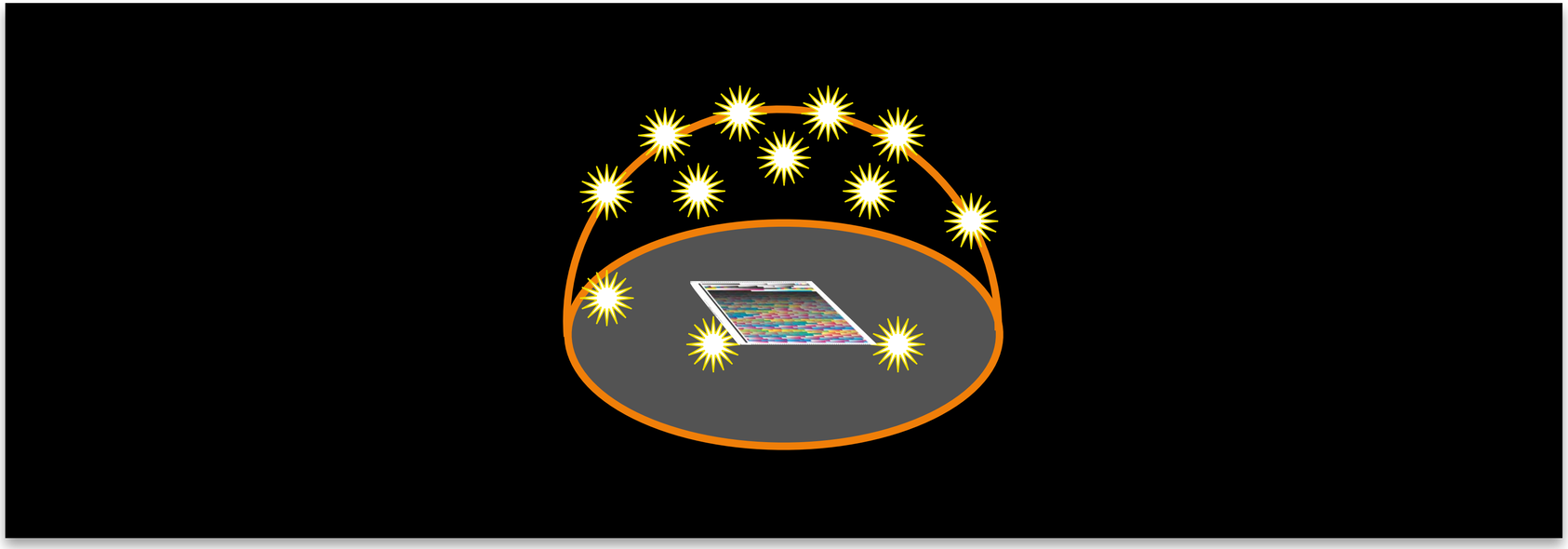
- The normal map and the BRDF aren’t independent from each other.
- Notice how the specular lobe is enlarged by the normal map.
  - The BRDF parameters and normal map should be calculated together to get the correct results.

Putting it Together

- Construct CMYK profile with BRDF and Texture information
  - Create output profile with chromatic BRDF model
- Use profile to soft-proof an image by using a 3D renderer

iccMAX Profile with Chromatic Ward BRDF Model

- Measure color chart to obtain surface appearance measurements



- Fit the measured data to the Ward BRDF model

Construct brdfColorimetricParameter3Tag

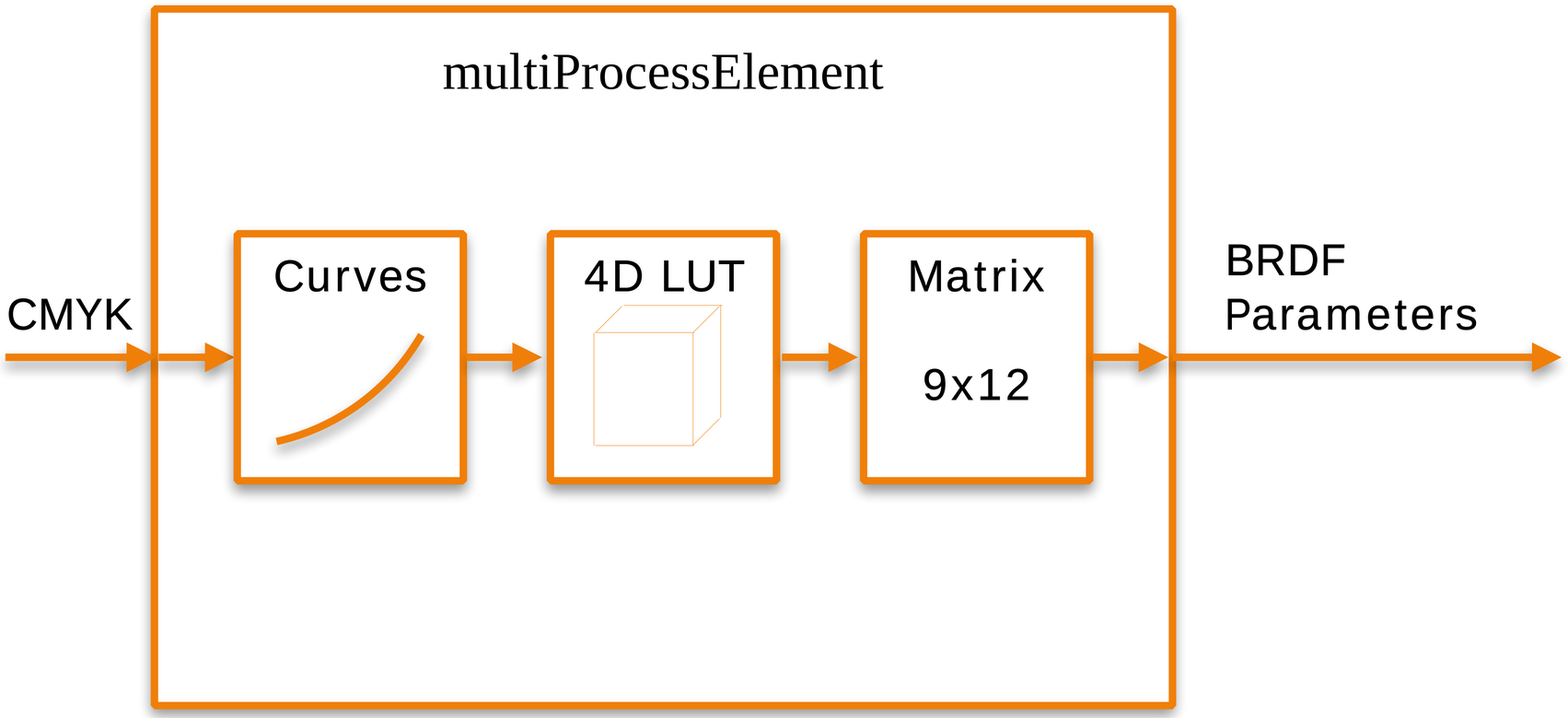
- Tag is tagStructureType tag of type brdfTransformStructure
  - BRDF model type is Chromatic Ward
    - Has four parameters per output channel resulting in a total of 12 output channels
    - Transform from CMYK to BRDF parameters is achieved with a multiProcessElement sub-tag

brdfTransformStructure

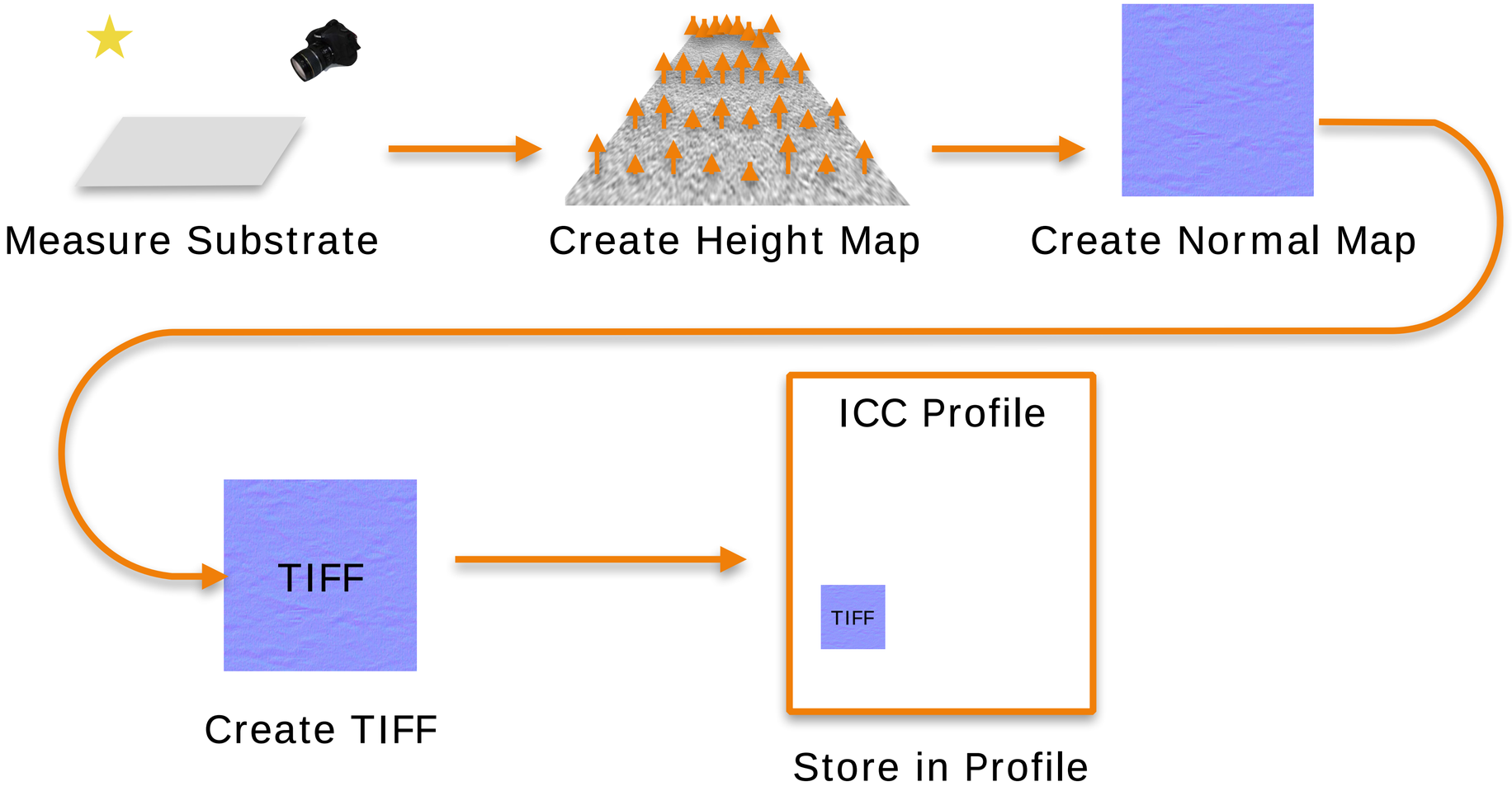
- Type: Chromatic
- Function: Ward
- Number of Parameters per output channel: 4
- Transform:

multiProcessElement
CMYK ->BRDF Parameters

Transform CMYK to BRDF parameters

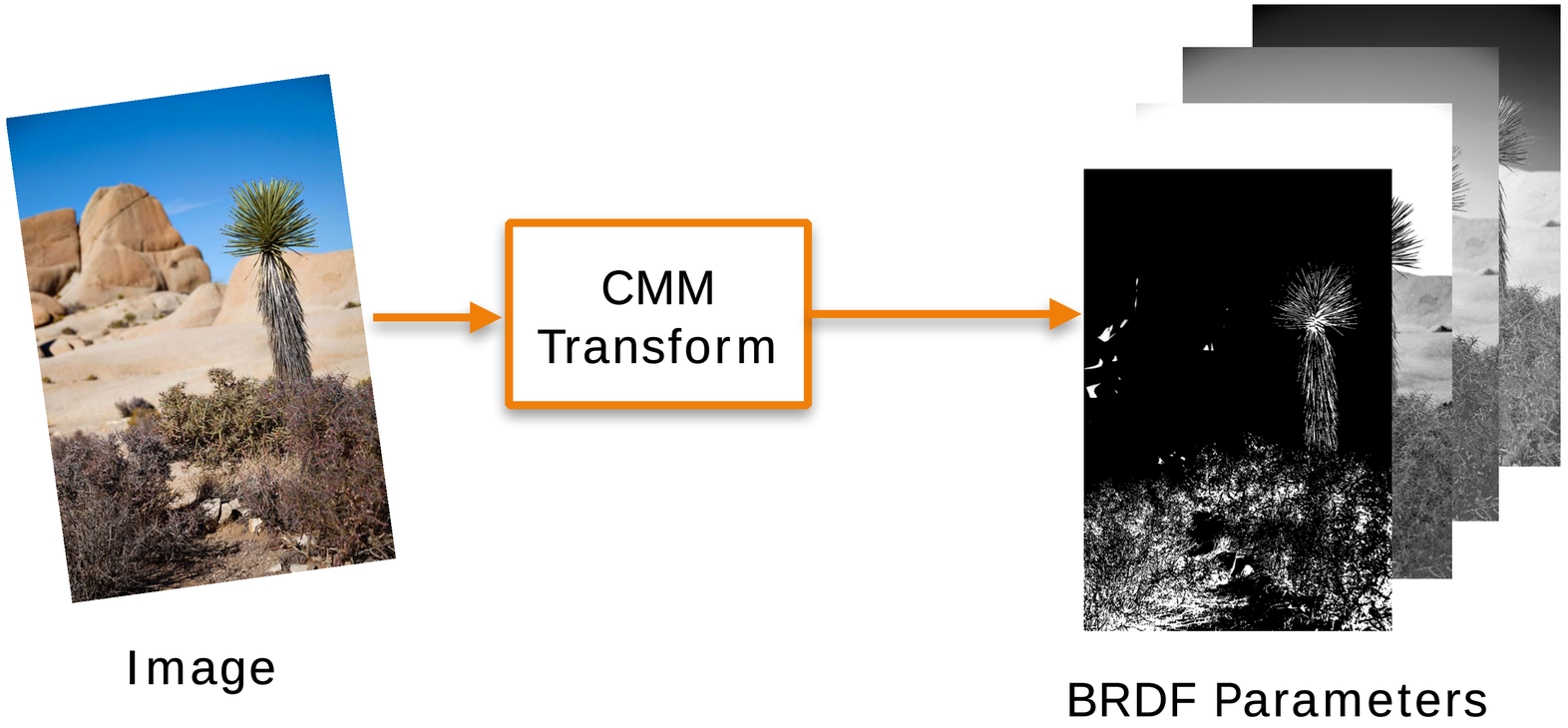


Create Normal Map

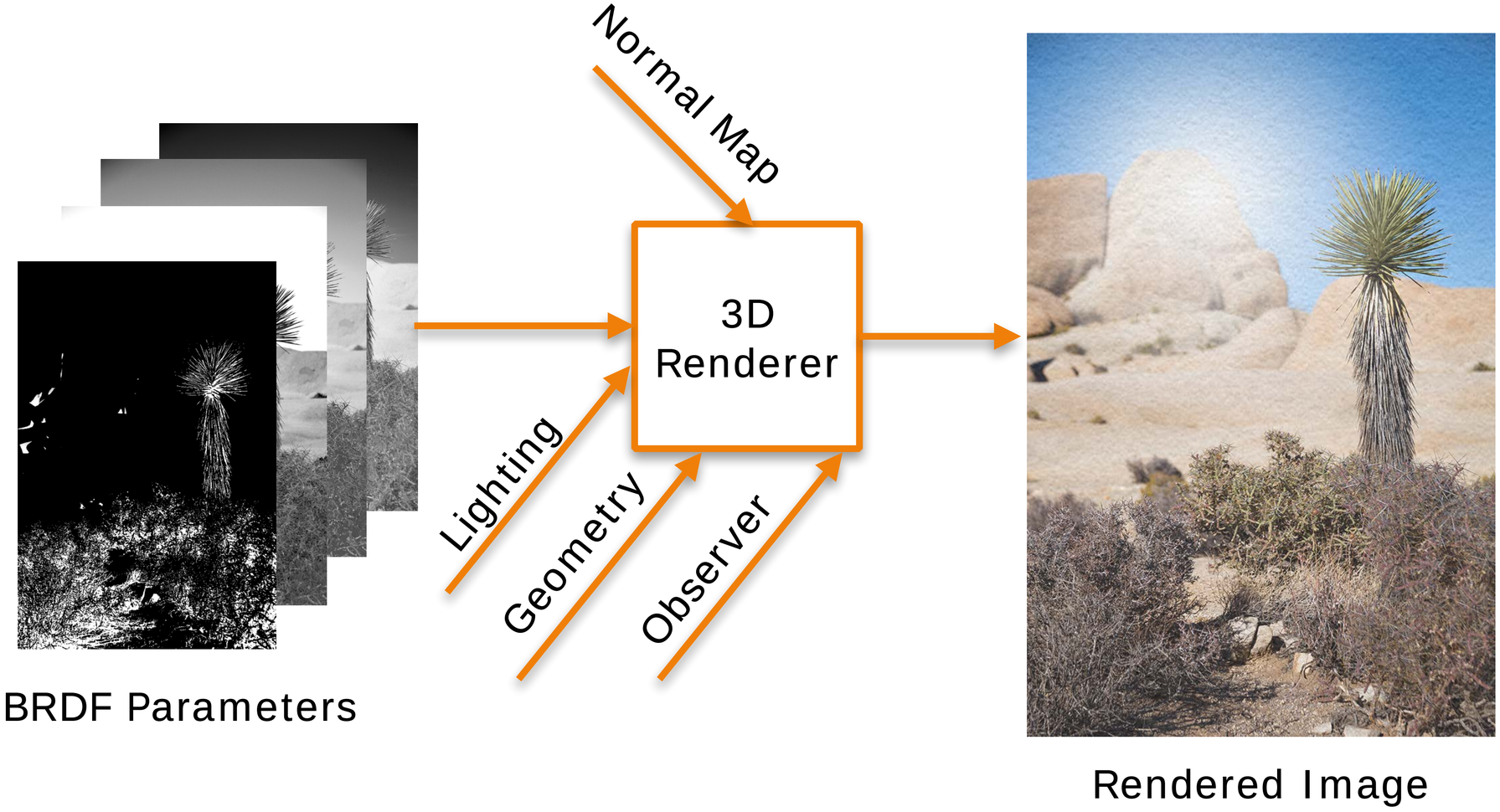


Get BRDF Parameters from Profile

- Load profile and create color transform
- Transform image to BRDF parameters



Render the Image



Demo

Thank You!

Questions?