Slide 1

Spectral Rendering with the Bounded MESE and sRGB Data

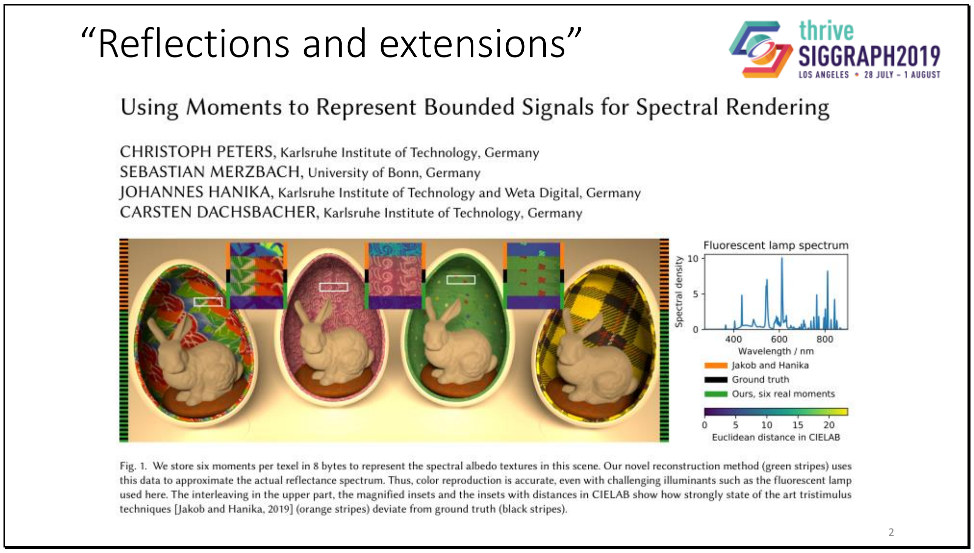
Christoph Peters[1], Sebastian Merzbach[2], Johannes Hanika[1,3], Carsten Dachsbacher[1]

[1]Karlsruhe Institute of Technology
[2]University of Bonn
[3]Weta Digital

Workshop on Material Appearance Modeling
2019-07-09

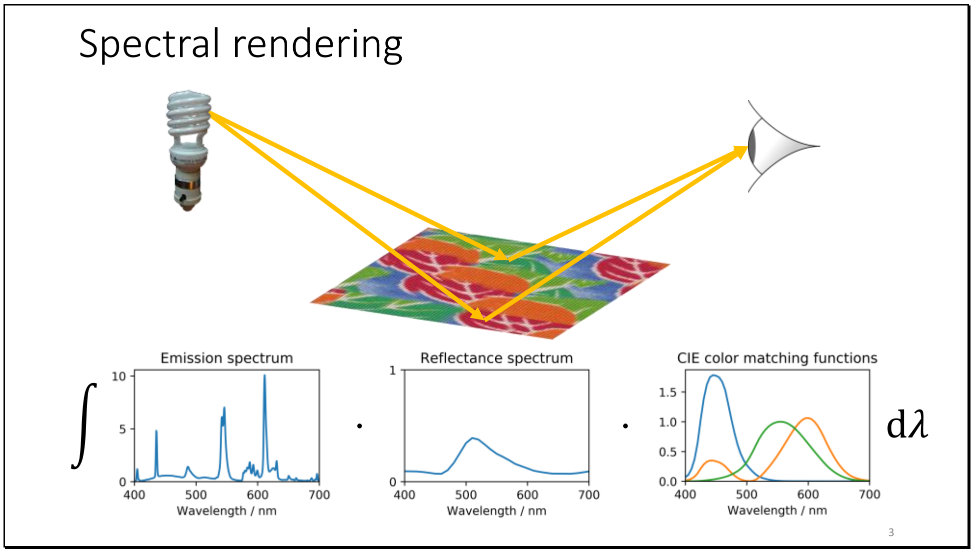
Welcome.

Slide 2



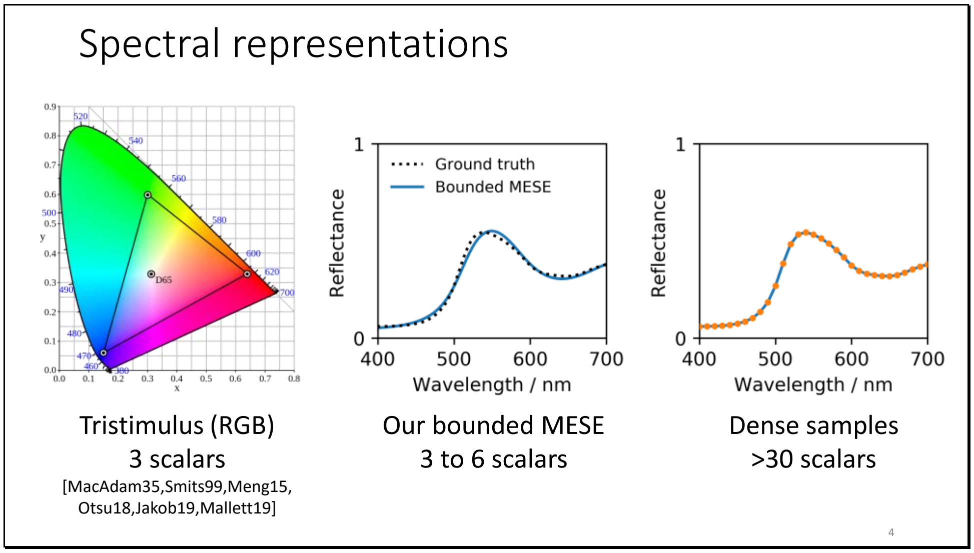
- This paper is a followup to an upcoming SIGGRAPH paper,
- Most of the talk summarizes this SIGGRAPH paper,
- Towards the end we introduce a new color space called Fourier sRGB, which lets you compress spectral reflectance textures in much the same way as sRGB textures.

Slide 3



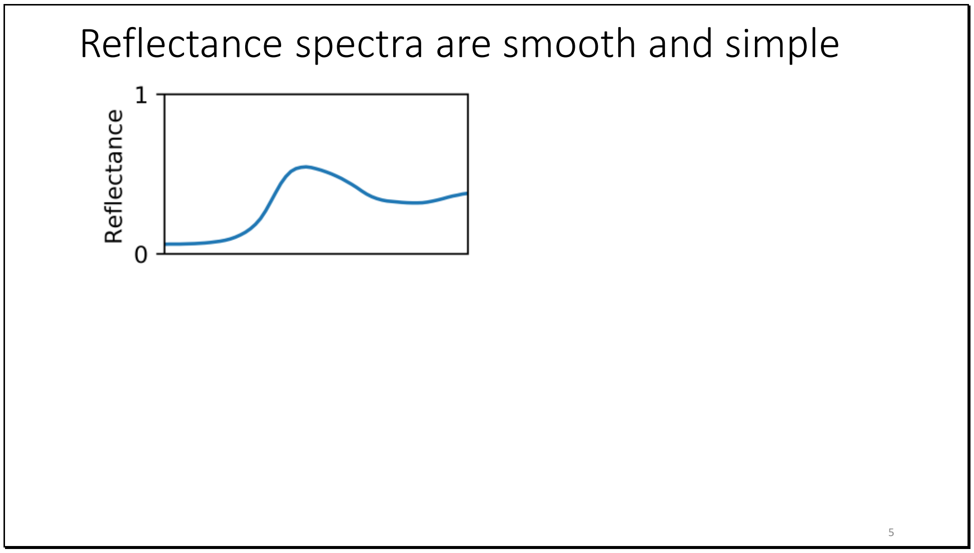
- Lets start with some basics of spectral rendering, CLICK
- We consider light transport from a light source via a surface to the eye, CLICK
- In a spectral renderer, each path vertex is associated with a spectral density,
- The emission spectrum of the light source can be very complex with sharp peaks as well as continuous components,
- Reflectance spectra are far more well-behaved,
- Human perception is characterized by the three CIE color matching functions for the standard observer,
- We multiply these spectral functions and integrate over the range of visible wavelengths to obtain our tristimulus color,
- Storing the three color matching functions is no problem and there are not too many emission spectra either,
- For reflectance however we need a more compact representation because if we look at a different pixel of our texture, CLICK
- We have a completely different reflectance spectrum.

Slide 4



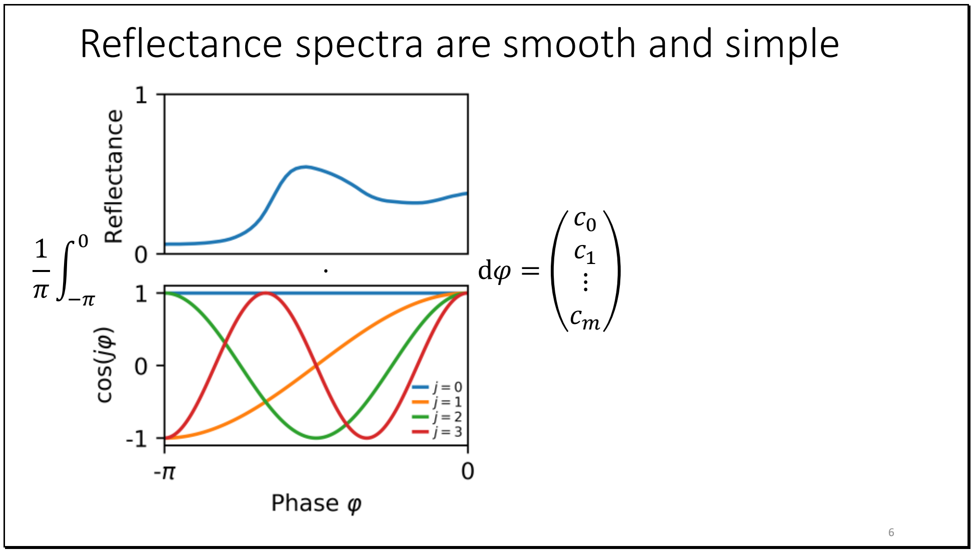
- There are two established representations for reflectance spectra,
- Most commonly a tristimulus color space is used, e.g. sRGB,
- In this case it is not trivial to turn this triple into a continuous spectrum, but there are quite a few prior works addressing this problem,
- On the other hand, we may sample the reflectance spectrum densely,
- With dense sampling we need around 30 samples for good color reproduction,
- Thus, we either use three scalars or 30 scalars per pixel and there is no sensible middle ground, CLICK
- Our work fills this gap,
- We present a representation of reflectance spectra that works well using anything from three to six scalars per pixel,
- With six scalars (shown here) results are close to ground truth even in challenging situations.

Slide 5



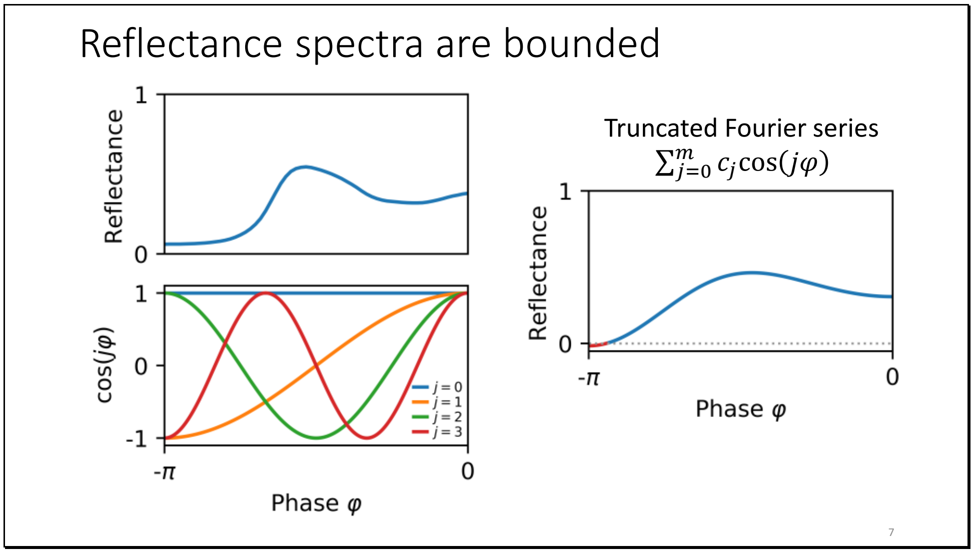
- So lets see how that works,
- Our representation can be so compact because reflectance spectra are well-behaved for physical reasons,
- As you can see in this example, they do not vary rapidly and do not exhibit a lot of complexity,
- There are databases with thousands of measured reflectance spectra but none of them are substantially more complicated than what you see here.

Slide 6



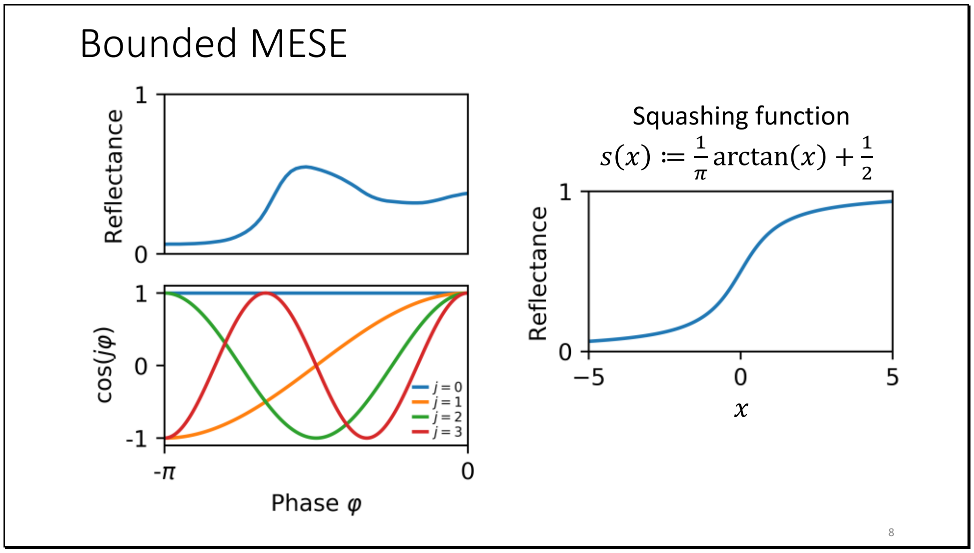
- Since reflectance spectra are so simple, Fourier coefficients are an obvious choice for their representation,
- However, they are aperiodic and therefore a cosine transform is more suited, i.e. our Fourier basis consists of cosine functions only,
- We associate wavelengths with the phase domain from –pi to 0, CLICK
- Then we take the product integral of our reflectance and the basis functions to obtain a small number of Fourier coefficients.

Slide 7



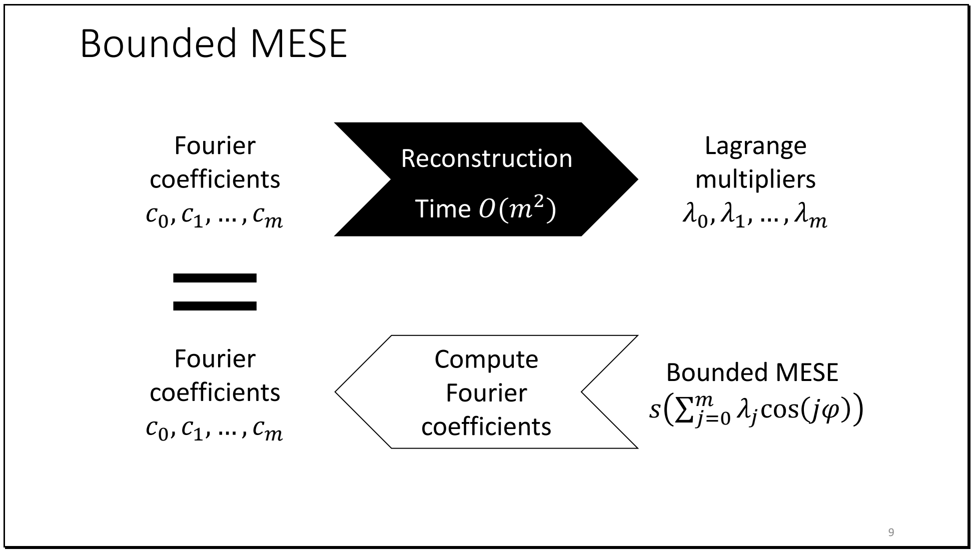
- For the reconstruction of a continuous spectrum from Fourier coefficients, the most obvious choice would be a truncated Fourier series,
- We simply take a linear combination of our basis functions using the Fourier coefficients as weights,
- However, the resulting function is not a physically meaningful reflectance spectrum,
- Real reflectance spectra obey energy conservation and thus they are bounded between zero and one,
- The truncated Fourier series suffers from ringing that leads to negative values in our reconstruction (shown in red),
- The negative values have small magnitude but original spectrum still has moderately large reflectance in this wavelength range,
- The reconstruction is inadequate.

Slide 8



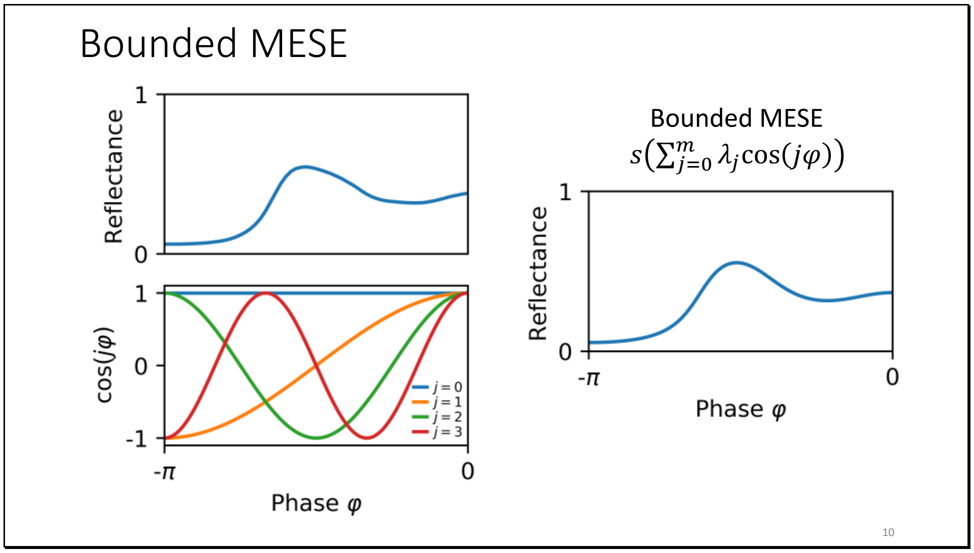
- This is why we introduced the bounded maximum entropy spectral estimate (bounded MESE for short),
- The bounded MESE enforces reflectance values between zero and one using a squashing function,
- As you can see, it is a monotonic function mapping all real numbers to numbers between zero and one,
- Our squashing function is a scaled and shifted arctangent function.

Slide 9



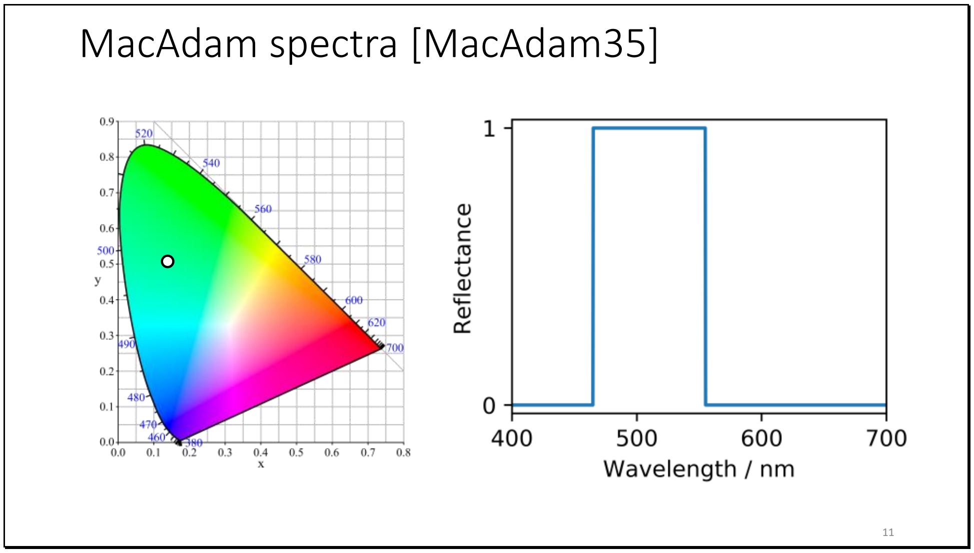
- With this particular squashing function, we define our bounded MESE,
- It is a Fourier series that is forced into the allowable range by the squashing function,
- The coefficients in this Fourier series are *not* the original Fourier coefficients, CLICK
- Instead they are so-called Lagrange multipliers that we compute from the original Fourier coefficients, CLICK
- The way in which we do this, ensures that the bounded MESE has exactly the same Fourier coefficients as the original signal,
- There is a lot of theory behind the algorithm computing the Lagrange multipliers and the bulk of the SIGGRAPH paper is concerned with this problem,
- For now it should suffice to know that the computation is efficient,
- The run time is quadratic in the number of Fourier coefficients.

Slide 10



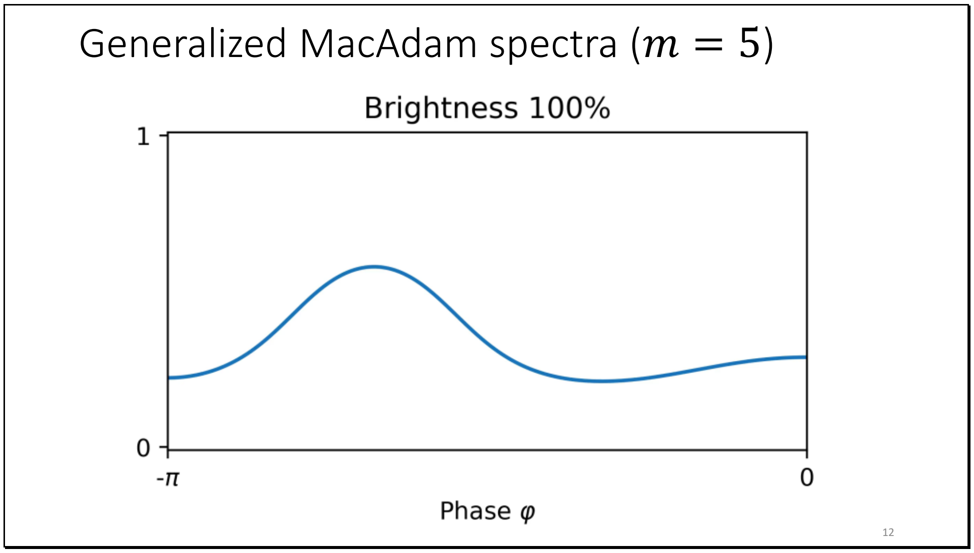
- If we apply the bounded MESE to our previous example, we obtain the reconstruction shown here,
- As expected, it is bounded between zero and one and behaves smoothly,
- Overall it resembles the ground truth signal very well,
- By incorporating the knowledge that reconstructions should be bounded between zero and one into our reconstruction, we have obtained a better reconstruction from the same Fourier coefficients.

Slide 11



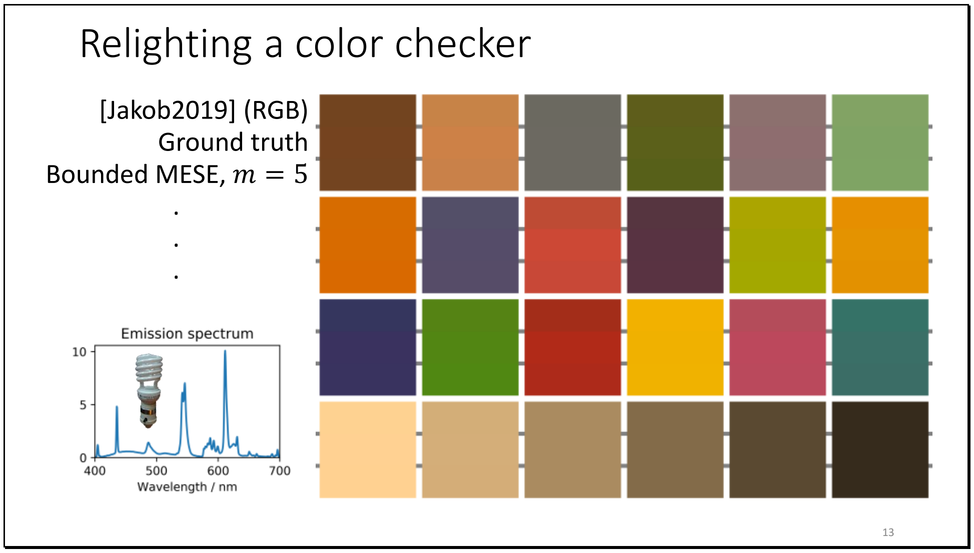
- It may seem that we have invested a lot of effort only to avoid some seldom negative values,
- But the capability to guarantee bounded results is more valuable than it may seem from the previous example,
- To understand why, lets take a look at a classic work in color science by MacAdam,
- For any fixed chromaticity (left), MacAdam sought after the reflectance spectrum of maximal brightness matching this chromaticity,
- He found that it is always a reflectance spectrum with binary values zero and one only two discontinuities (right),
- So in essence, it reflects all light within an interval of wavelengths and no light outside of that interval,
- While these MacAdam spectra are too extreme to be found in nature, bright and saturated colors have artistic value and dye chemists have tried to come close to these idealized spectra.

Slide 12



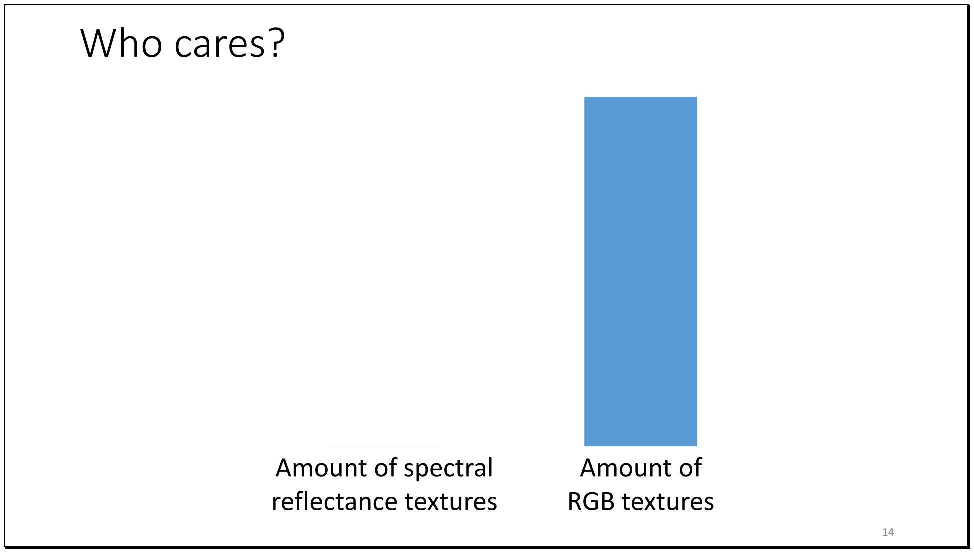
- So lets see what our bounded MESE does if we ask for a spectrum of maximal brightness,
- We start from an unmodified reconstruction,
- Now we scale up all the Fourier coefficients to make the spectrum brighter, CLICK
- Initially, the spectrum changes in a fairly linear way,
- But as it gets closer to the bounds, the only way in which a bounded reconstruction is still possible is to use a spectrum reflecting mostly in a small number of intervals,
- Thus, our higher-dimensional representation of reflectance leads to generalized MacAdam spectra,
- The ability to represent such extreme spectra implies that our color spaces are complete,
- Even the most extreme physically plausible reflectance spectra can be represented and reconstructed.

Slide 13



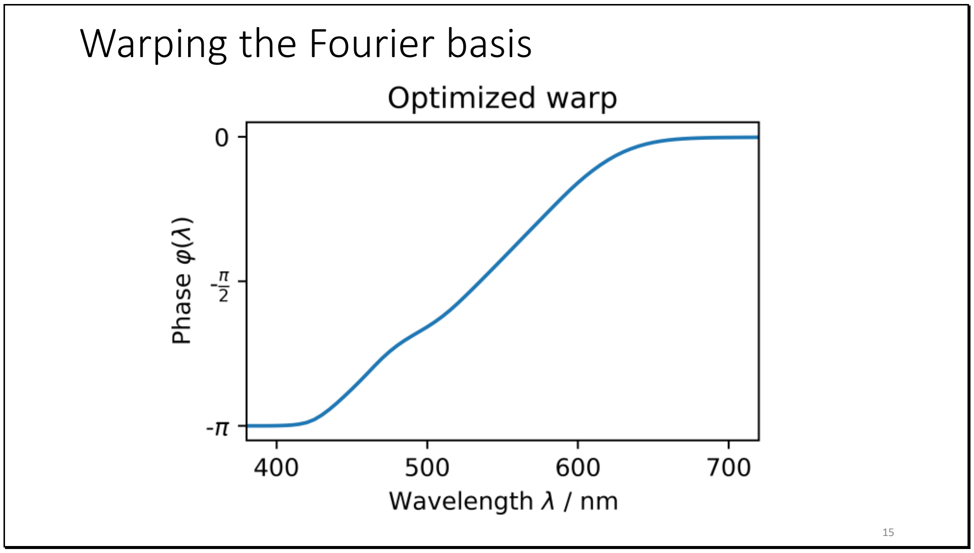
- To see how our method helps in rendering, we run a simple experiment,
- We take the X-Rite color checker and represent its spectra using our bounded MESE with six coefficients,
- Then we illuminate it using a fluorescent lamp and show the resulting colors, CLICK
- Using the ground truth, we get nearly identical colors, CLICK
- If we instead use a state-of-the-art technique for computing reflectance spectra from RGB triples, there are some fairly strong errors, especially on red patches,
- This example shows that the additional information that we can use with our approach truly improves the quality of renderings.

Slide 14



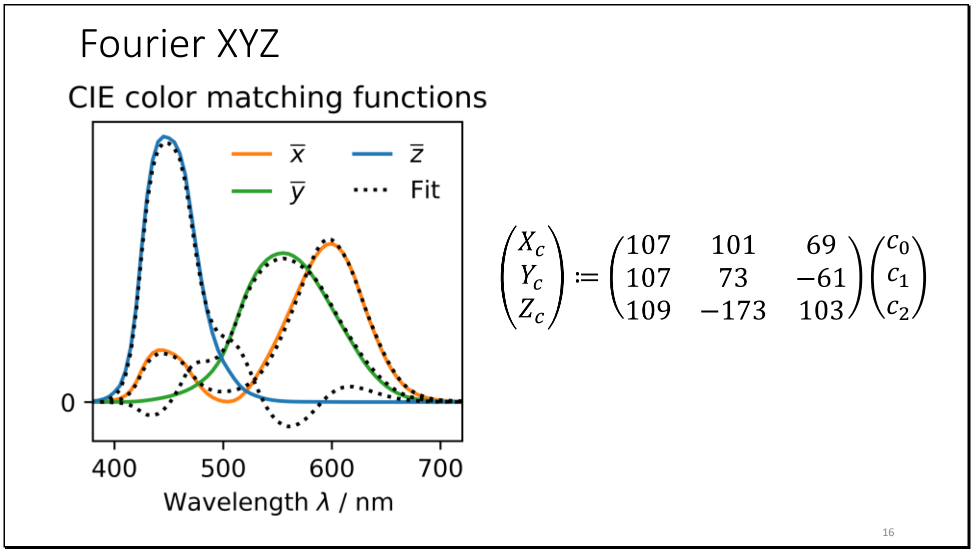
- Sadly, the spectral reflectance data needed to benefit from this approach is hardly ever available, CLICK
- In most production environments, there are huge amounts of RGB textures but hardly any spectral reflectance textures, CLICK
- To make our approach more widely applicable, we want to convert RGB textures to our representation,
- To account for bandwidth and memory limitations, we will introduce an encoding of these converted textures that can be compressed as well as sRGB.

Slide 15



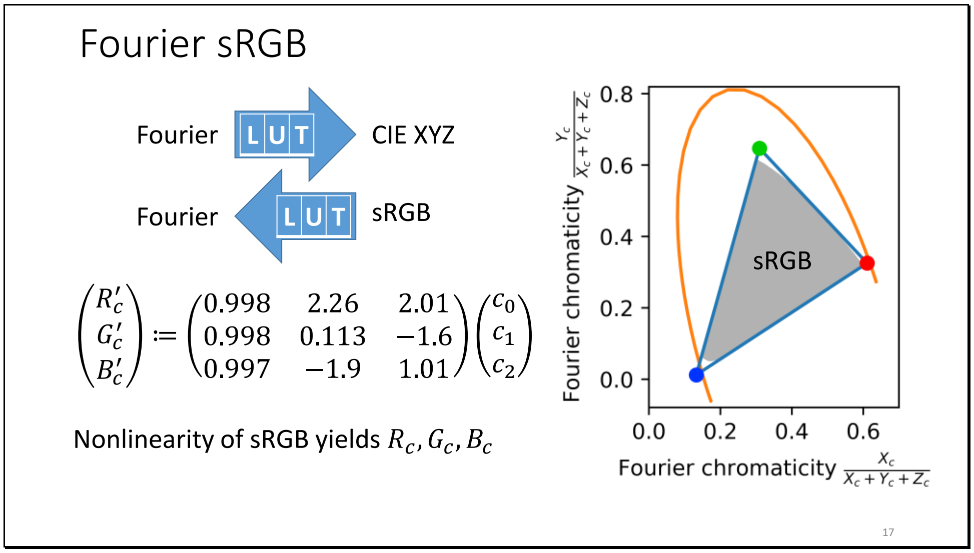
- sRGB is defined in terms of the CIE color matching functions,
- If we want to use the bounded MESE, we cannot use those because the algorithm only works with Fourier coefficients,
- However, we do get to choose how wavelengths in the visible range are mapped to phases between –pi and 0,
- Effectively, this changes the Fourier basis in the wavelength domain through warping and weighting,
- We optimize this warp such that the changed basis is as close to XYZ as possible.

Slide 16



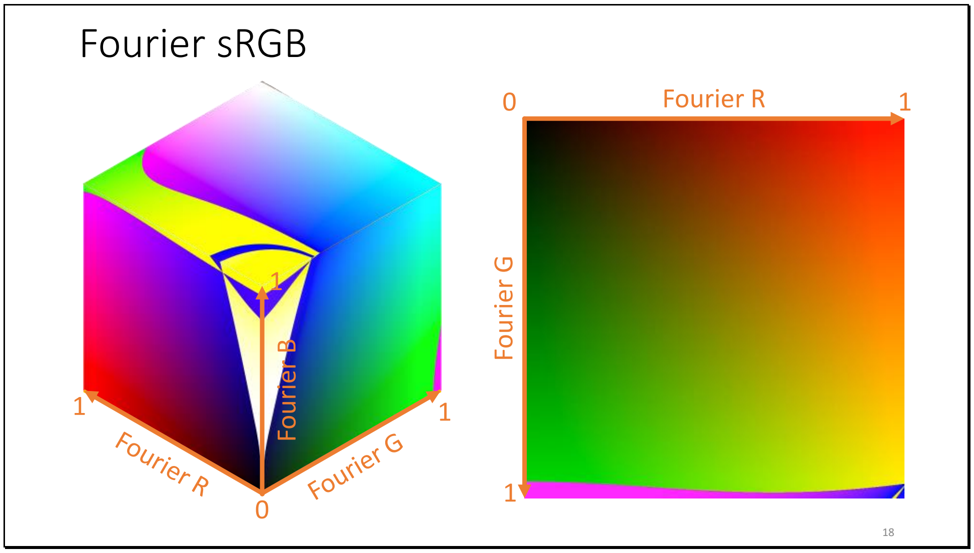
- With this warp, we get the fits to the color matching functions shown here,
- Using only three Fourier basis functions, we obtain the three least-squares fits shown as dotted lines,
- Based on these fits we can define the Fourier XYZ color space, which is designed to behave similar to XYZ,
- We simply take linear combinations of the three Fourier coefficients in accordance with the three least-squares fits shown on the left,
- The three resulting coefficients are not identical to the CIE XYZ coefficients of a spectrum but their overall behavior is similar because the fits are reasonably close.

Slide 17



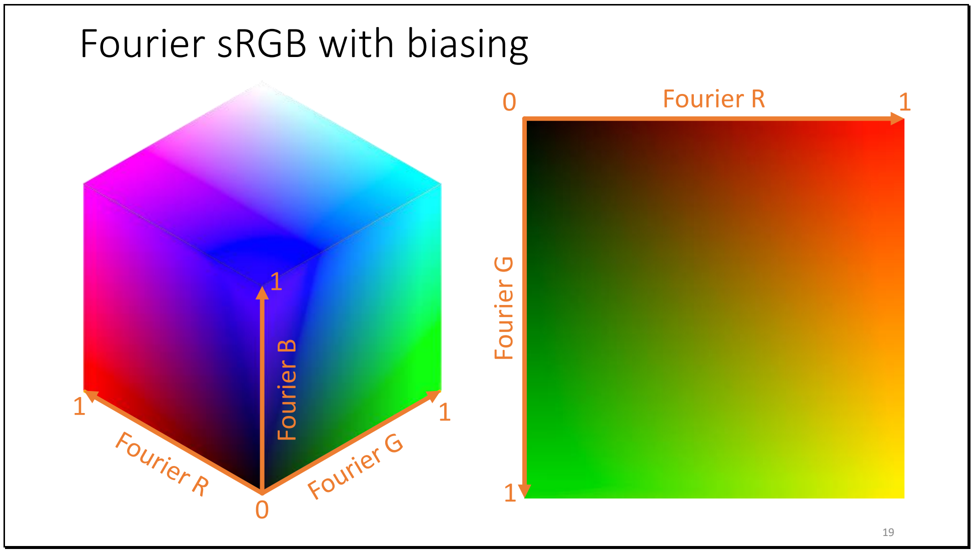
- Though, what we really care about are RGB color spaces and especially the sRGB color space,
- We want to define a color space called Fourier sRGB in terms of Fourier coefficients that behaves similar to sRGB,
- Thus, we begin by constructing two lookup tables,
- The first maps a dense sampling of Fourier coefficients to the CIE XYZ coefficients of the reconstructed spectra,
- Using this lookup table and a nearest-neighbor index, we can construct a lookup table going the other way,
- It maps a dense sampling of sRGB triples to matching triples of Fourier coefficients,
- To the right, you see a chromaticity diagram defined in terms of Fourier XYZ,
- The grey area shows the range of Fourier coefficients needed to cover all of sRGB,
- As you can see, the shape is a somewhat rounded triangle because the mapping from Fourier coefficients to reflectance spectra through the bounded MESE is non-linear, CLICK
- To define a color space covering this range, we fit an enclosing triangle of minimal area, CLICK
- The three vertices of this triangle can be thought of as primaries and we use them to define a linear transform mapping Fourier coefficients to linear Fourier sRGB,
- Then to define our final Fourier sRGB color space, we apply the same non-linearity as in sRGB to each individual component,
- This way, all existing implementations of the dequantization and linearization for sRGB are also applicable to Fourier sRGB.

Slide 18



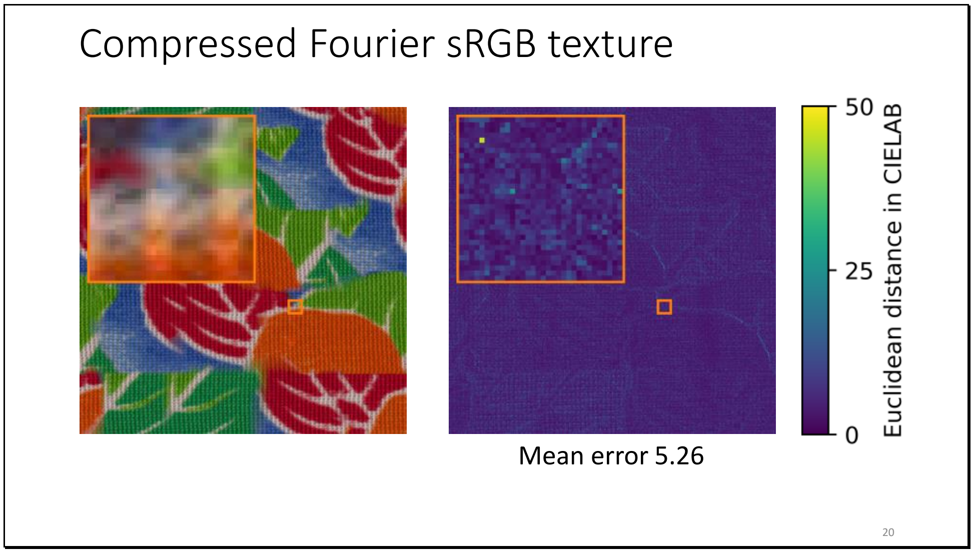
- Lets take a closer look at our Fourier sRGB color space,
- To the left you see the whole color cube,
- To the right you can see a slice for a constant blue value,
- Over time, we now look at all possible blue values, CLICK
- As you can see, colors mostly change smoothly throughout our color cube but there are some regions where results are completely different,
- No spectrum bounded between zero and one can produce the Fourier coefficients in these regions and thus the reconstruction can impossibly provide such a spectrum,
- In this sense, the behavior is expected,
- The good news is that we do not need these regions to cover sRGB,
- Nonetheless, compression errors may turn valid Fourier sRGB triples into triples in these invalid regions,
- In this case, the color reproduction would be completely wrong.

Slide 19



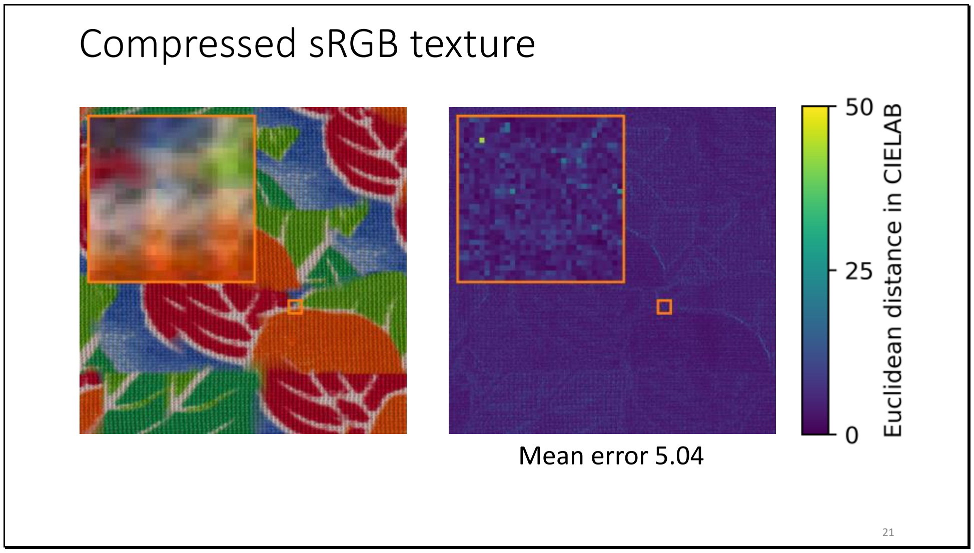
- To avoid this potential problem, we introduce a biasing strategy,
- It is integrated into our reconstruction algorithm and only takes a few additional instructions,
- Invalid Fourier coefficients are identified and corrected on the fly,
- With this modified reconstruction, we now get a smooth change of colors throughout the entire Fourier sRGB cube,
- Unexpected amplifications of compression errors are no longer an issue.

Slide 20



- With the lookup table described before, we can convert any sRGB texture to a Fourier sRGB texture,
- Lets see what happens when we compress such a texture,
- The texture to the left has been converted to Fourier sRGB, JPEG compressed and then converted back to sRGB,
- There are some typical JPEG artifacts but nothing out of the ordinary,
- If we look at the average perceptual error, defined using CIELAB, it is 5.26.

Slide 21



- If we had instead compressed the sRGB texture directly, the average error would be 5.04,
- Thus applying the compression in Fourier sRGB gives a slightly increased error but the texture can be used for spectral rendering directly.

Slide 22

Conclusion

Fourier sRGB can be compressed like sRGB

but enables fast computation of smooth spectra.

- With that we have reached our goal,
- Fourier sRGB textures can be compressed as efficiently as sRGB textures using all the same methods,
- But the Fourier coefficients let us reconstruct smooth spectra efficiently using the bounded MESE.

Slide 23

- Thank you for your attention.

Slide 24

References

Wenzel Jakob and Johannes Hanika. 2019. A Low-Dimensional Function Space for Efficient Spectral Upsampling. Computer Graphics Forum 38, 2 (2019). https://doi.org/10.1111/cgf.13626

David L. MacAdam. 1935. The Theory of the Maximum Visual Efficiency of Colored Materials. J. Opt. Soc. Am. 25, 8 (1935). https://doi.org/10.1364/JOSA.25.000249

Johannes Meng, Florian Simon, Johannes Hanika, and Carsten Dachsbacher. 2015. Physically Meaningful Rendering using Tristimulus Colours. Computer Graphics Forum 34, 4 (2015). https://doi.org/10.1111/cgf.12676

Slide 25

References (continued)

Ian Mallett, and Cem Yuksel. 2019. Spectral Primary Decomposition for Rendering with sRGB Reflectance. Proceedings of the 30th Eurographics Symposium on Rendering. https://doi.org/10.2312/sr.20191216

Hisanari Otsu, Masafumi Yamamoto, and Toshiya Hachisuka. 2018. Reproducing Spectral Reflectances From Tristimulus Colours. Computer Graphics Forum 37, 6 (2018), 370–381. https://doi.org/10.1111/cgf.13332Smits99

Brian Smits. 1999. An RGB-to-spectrum Conversion for Reflectances. Journal of Graphics Tools 4, 4 (1999), 11–22. https://doi.org/10.1080/10867651.1999.10487511