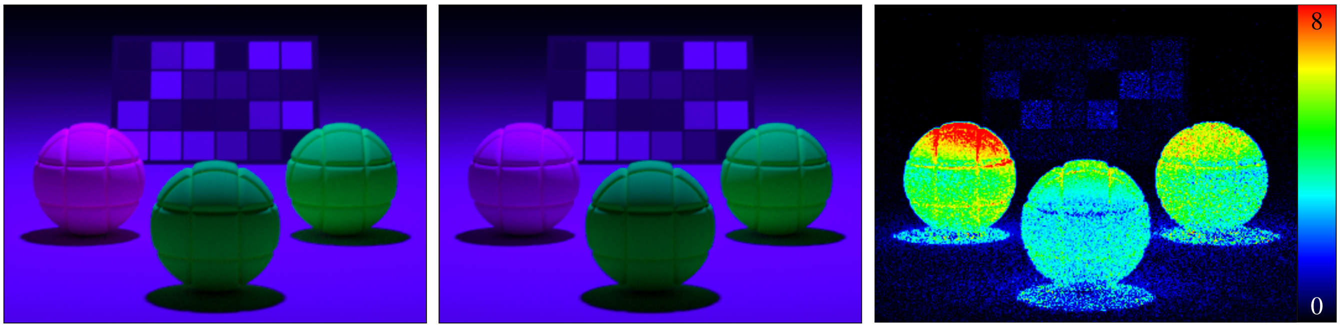
Eurographics Symposium on Rendering (DL-only Track) (2021)
A. Bousseau and M. McGuire (Editors)

A Compact Representation for Fluorescent Spectral Data

Q. Hua, A. Fichet, A. Wilkie

Charles University



(a) *Rendering of fluorescent materials using a re-radiation matrix (27 552 B for the 3 pure re-radiation spectra).*

(b) *Rendering of fluorescent materials using the proposed Gaussian Mixture Model representation (529 B for all 3 pure re-radiation spectra).*

(c) ΔE^*_{00} *between (a) and (b).*

Figure 1: *In this scene, we use a near monochromatic light for illumination. While non-fluorescent materials only reflect shades of the light colour, fluorescent surfaces also reflect other wavelengths leading to the colourful appearance of the three fluorescent spheres in this scene. In daylight environment, fluorescent effects are generally responsible for brighter and more vivid colours: weakly visible radiation are re-emitted to wavelength our eyes are more sensitive to. To handle such spectra, we commonly use a re-radiation matrix (a). These triangular matrices have an important memory overhead. We propose to represent this data more efficiently with a Gaussian Mixture Model (b). While being compact, this representation is also precise with limited difference compared with using the full re-radiation matrix (c).*

Abstract

We propose a technique to efficiently importance sample and store fluorescent spectral data. Fluorescence behaviour is properly represented as a re-radiation matrix: for a given input wavelength, this matrix indicates how much energy is re-emitted at all other wavelengths. However, such a 2D representation has a significant memory footprint, especially when a scene contains a high number of fluorescent objects, or fluorescent textures. We propose to use Gaussian Mixture Domain to model re-radiation, which allows us to significantly reduce the memory footprint. Instead of storing the full matrix, we work with a set of Gaussian parameters that also allow direct importance sampling. When accuracy is a concern, one can still use the re-radiation matrix data, and just benefit from importance sampling provided by the Gaussian Mixture. Our method is useful when numerous fluorescent materials are present in a scene, an in particular for textures with fluorescent components.

CCS Concepts

• ***Computing methodologies*** → ***Reflectance modeling;***

1. Introduction

Fluorescence is a common phenomenon both in natural elements and in chemical compounds. It has an important impact on how we perceive a scene, both in terms of hue, saturation, and luminance: a part of non-visible or near visible light is transferred into a different part of the visible range, so that the effect is widely used to enhance material brightness. In addition to this, it can also have the effect of making colour appear more vivid than reflective colour can normally look. While the importance of this effect for appearance modelling in computer graphics has been known for years, it has not been used in practice much yet, as adding it to graphics workflows – both on the side of user interfaces, as well as within rendering software proper – is a demanding problem.

One of the issues standing in the way of widespread adoption is that representing fluorescent effects consumes significant memory, and causes non-negligible computational overhead. To fully describe fluorescent spectra, one normally uses *re-radiation matrices* [Don54]: these represent emission and reemission efficiency for a given set of wavelengths. This requires storing a densely sampled 2-dimensional dataset for each fluorophore used in a scene. While explicitly storing this data is manageable for few fluorescent elements, this does not scale for fine-grained material definitions such as fluorescent textures.

In recent years, there has been an increasing interest in the effect in the research community. In one recent example, it was used to enlarge the colour gamut during spectral uplifting [JWH*19], in a fashion analogous to the use of optical brighteners in real materials. However, for lack of better alternatives, the method presented in that paper had to store a full re-radiation matrix for all pixels in a texture: which in turn made it incapable of working with more than a single fairly low-resolution input image.

In this paper, we propose to efficiently represent re-radiation matrix by using a Gaussian Mixture Model (GMM) to represent the re-radiation mechanism. This allows not only a low memory footprint, but also enables efficient importance sampling of wavelength shifting events.

2. Previous work

Spectral rendering Many rendering pipelines still rely on RGB or tristimulus values to compute light transport. This allows a compact representation for reflectance and illuminants, and fast computation. However, the resulting colours are not accurate and cannot be used for predictive rendering applications. Also, this type of rendering cannot simulate wavelength dependent effects such as dispersion or fluorescence. With the increasing computational power and the efficient multiplexing approach proposed by [WND*14], spectral rendering is becoming increasingly popular.

Bi-spectral rendering While not widely used, the concept of fluorescence in computer graphics is not new. Glassner [Gla95] introduced fluorescence and phosphorescence to the computer graphics community and proposed a method to support those effects in a Whitted ray tracer. Wilkie *et al.* [WTP01] proposed a combination of rendering fluorescent and polarisation effects. Despite using a path tracer, they relied on a defined number of spectral bands, the wavelength not being a part of the Monte-Carlo integration. Mojzík *et al.* [MFW18] introduced a method for both handling fluorescent medium in a Monte Carlo path tracer and making the wavelength domain a part of the integrand, and adapting Hero Wavelength Spectral Sampling [WND*14] to support fluorescence.

Color gamut enlargement One of the challenges when using a spectral renderer is the need to have spectrally defined assets. Assets are commonly defined in a tristimulus space (RGB, XYZ, etc.) and cannot be used directly as the input of a spectral renderer. To allow usage of RGB assets in a spectral renderer, one needs to "uplift" the original RGB data to a spectral form. There is previous work dealing with non-fluorescent spectral uplifting [OYH; JH19; MSHD15]. Recently Jung *et al.* [JWH*19] proposed an uplifting pipeline that additionally also uses fluorescent spectra to enhance the colour gamut. An important drawback of this bi-spectral uplifting method is its memory requirement: when dealing with texture uplifting, each pixel potentially needed to store a full re-radiation matrix. However, this was not an intrinsic drawback of their method, but an indicator that this part of fluorescent rendering technology needed improvement.

Bi-spectral material models Wilkie *et al.* [WWLP06] proposed one of the early bi-spectral models adapted to current rendering techniques. This model uses a layered BRDF with a diffuse fluorescent component. Later, Hullin *et al.* [HHA*10] proposed an efficient acquisition setup to capture Bi-spectral Bidirectional Reflectance and Re-radiation Distribution Functions (BRRDF). They guided their acquisition by a Principle Component Analysis to lower the acquisition time.

Gaussian representations in rendering The general idea of using Gaussian distributions in rendering is very powerful, as they can approximate the distribution of samples in any shape, given a large amount of sample numbers. A number of papers have experimented with Gaussian representations in the area of physically-based rendering. Jakob *et al.* [JRJ11] introduced how to represent radiance using Gaussian mixture and apply for volumetric media. Herholz *et al.* [HEV*] and Vorba *et al.* [VKŠ*14] showed how Gaussian mixture can help with importance sampling in the path guiding framework. Herholz *et al.* [HES*18] used the Gaussian Mixture to sample analytic BRDF models and measured BRDF data.

3. Background

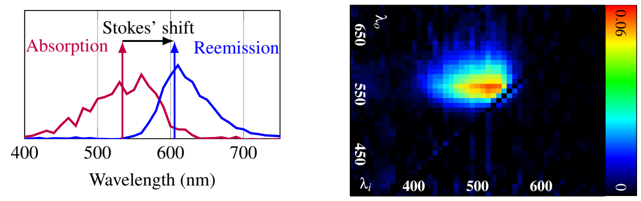
In this section, we introduce the core mathematical and physical context used in the remainder of this article. We present the fluorescent effect, its necessity in physically-based rendering, and the typical representation of fluorescent data in a fluorescent capable renderer. We also review the basic notation of Gaussian mixture model and the formal framework of expectation maximisation, which is one of the most effective methods to train GMM in the machine learning field.

3.1. Fluorescence

Fluorescence is the effect where a "fluorophore" absorbs photons and re-emits those at a lower energy state. In the context of rendering, this means a colour shift when light is interacting with such a molecule. In contrast to phosphorescence, fluorescent effects occur within a very short period of time, and can be considered as instantaneous in the context of computer graphics.

Fluorescence produces interesting visual effects: as the human vision system is only sensitive to a comparatively narrow range of wavelengths, fluorescent materials often are used to increase visibility and colourfulness. Energy from the ultra-violet region, which is at best barely visible for the human eyes, is transferred to a bandwidth with much higher sensitivity. Such fluorescent elements appear brighter than the environment despite the decrease of energy between the absorption and the reemission that still happens.

A fluorescent element is sometimes characterised by an absorption and a reemission spectrum (see Fig. 2a). The first defines which wavelenghts lead to a reemission event and the amplitude define



(a) *A fluorescent element is often characterised by its absorption spectrum and its reemission spectrum. The offset between the two spectra is called the Stokes shift.*

(b) *Example of a re-radiation matrix. This matrix gives the reemission spectra for every radiating λ_i wavelength. We removed the diagonal ($\lambda_i = \lambda_o$) to emphasise on the fluorescent effect.*

Figure 2: *(**a**) The synthetic representation of the average absorption and reemission spectra is often used to characterise a fluorophore. It assumes the shape of the reemission is similar regardless of the radiating wavelenght but scaled in intensity. However, Kasha's rule does not hold for may fluorophore hence the use of re-radiation matrices (**b**). Such matrix provides precise information about the reemitted spectra given a radiating wavelength but have a significant memory footprint.*

the intensity of the reemission. The reemission spectrum defines the average reemission accross all radiating wavelengths. The shift between the two central frequencies of those spectra is the Stokes' shift. These spectra then only scale in intensity depending on the radiating wavelength according to Kasha's rule.

When following the Kasha's rule, this reemission spectrum is considered of the same shape regardless of the radiating wavelength but scaled in intensity. Kasha's rule does not always strictly hold, however. This leads to the need for re-radiation matrices, which characterise, for each radiating wavelength, a distribution of reemission wavelengths (see Fig. 2a). While more flexible and suited for compute graphics application, this representation has an important memory overhead.

In the context of path tracing, a renormalised reemission spectra can be seen as the PDF of wavelength shifting for a given incoming radiation λ_i. In reverse, a renormalised absorption can be seen as the PDF of absorbing a wavelength λ_i for a given outgoing radiation λ_o.

3.2. Gaussian Mixture Model

A Gaussian Mixture Model (GMM) is a parametric probabilistic model representing a dataset by a linear superposition of Gaussian functions. This linear superposition is represented by:

$$p(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x} \mid \mu_k, \Sigma_k) \tag{1}$$

where:

$p(\mathbf{x})$	is the density of mixture of Gaussians,
$\mathcal{N}(\mathbf{x} \mid \mu_k, \Sigma_k)$	is a single Gaussian density,
π_k	is the mixing coefficient (weight) of each Gaussian,
K	is the number of Gaussians.

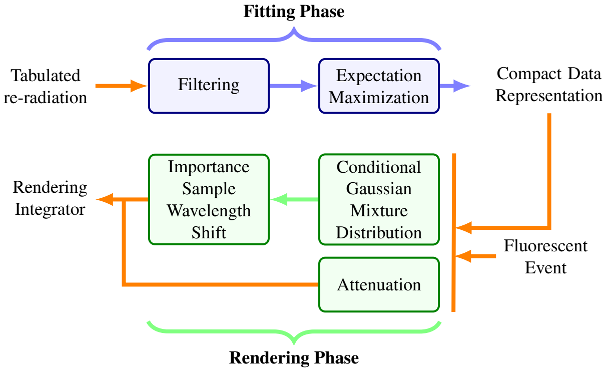


Figure 3: *Representation of our pipeline. We model the input re-radiation with a GMM in the fitting phase and then use it in the renderer. In the rendering phase, we importance sample the new wavelength according to the GMM.*

In this article, we use a 2-dimensional multivariate Gaussian distributions for each component:

$$\mathcal{N}(\mathbf{x} \mid \mu, \Sigma) = \frac{1}{2\pi\sqrt{|\Sigma|}} \mathrm{e}^{-\frac{1}{2}(\mathbf{x}-\mu)^T \Sigma^{-1} (\mathbf{x}-\mu)} \tag{2}$$

where:

μ	is the mean,
Σ	is the covariance matrix.

We model our fluorescent dataset as a mixture of Gaussian and determine their individual parameters to most accurately represent the input data.

Expectation Maximisation (EM) is a two-stage iterative algorithm to estimate the mixture model parameters. It tries to maximise likelihoods for parametric models. The E-step (expectation) performs estimation of the expected log-likelihood for complete-data set. M-step (maximisation) maximses the expected complete log-likelihood. The E-step and M-step are run iteratively until the expectation converges. EM and its variants are widely used to fit parametric models. As detailed introduction of EM is beyond the scope of our work, we only review briefly the sketch of EM, for detailed study on EM please refer to [Bis06]. Vorba *et al.* [VKŠ*14] also show a detailed version of how to adapt EM to the rendering context.

4. Method overview

Our pipeline consists of two main stages (see Fig. 3):

A fitting phase were we use the Expectation Maximisation algorithm to represent the input re-radiation dataset as a Gaussian Mixture Model. We describe this part in Sec. 5.

A rendering phase were we reconstruct the re-radiation on the fly and perform importance sampling based on the fitted model. We describe this part in Sec. 6.

The pipeline starts from a 2-dimensional fluorescent measurement dataset. To get a clearer representation for fitting, we first filter this dataset. Then we perform weighted Expectation Maximisiation to get a compacted representation of the distribution. We then load the fitted GMM representation in the renderer. For each fluorescent sampling event, we use conditional GMM sampling on the given wavelength. Depending on the path direction, we either sample an excitation wavelength or reemission wavelength. The sampled wavelength will then shift to reproduce the fluorescent effect.

Using a GMM to represent the original dataset drastically reduces the memory footprint. Instead of relying on a tabulated triangular matrix, we only need to store few parameters representing parameters of the mixture, the re-radiation is then evaluated analytically. This reduction of memory footprint is even more prominent when the rendering technique rely on random sampling, as this also removes the need for additional tabulated CDFs used to sample the re-radiation matrix.

5. Fitting phase

In the fitting phase, we apply parametric density estimation to our fluorescence data. Later, we show that we can reconstruct the re-radiation matrix from the GMM on the fly with high accuracy, instead of using the raw tabulated data.

5.1. Filtering

Very often, measured re-radiation data contains high frequency noise. In our pipeline, we remove all data under a certain threshold to pre-filter the data prior to the fitting phase. We found this simple approach appropriate for all the datasets we have analysed. For more challenging cases, e.g., lower signal-to-noise with less pronounced fluorescence or less precise measurement device, a more advanced method such as Fourier analysis may be used to analyse and remove the noise without impacting the other steps of the pipeline.

5.2. Weighted Expectation Maximisation

The re-radiation matrix is a 2-dimensional dataset where each entry corresponds to an attenuation. When each value is divided by the integral of the matrix, it represents a tabulated probability distribution function, each entry is a weight of this distribution.

In the standard EM-algorithm, each element corresponds to an observation. To emulate this, we could repeat each entry according to its weight in the re-radiation matrix. From our experiments, this strategy works well for fitting the data with different weights. However, this technique requires multiplying the weight by a large number to accurately represent the original data. This implies dealing with huge observation sets and consequently, a high memory footprint.

A more suitable approach is to use the weighted EM-algorithm [GAFH16; VKŠ*14]. All non-null elements of the matrix are used as observation outcomes and weighted according to their respective values.

We apply the weighted EM algorithm to fit Gaussian Mixture Model with our datasets using the pomegranate Python library [Sch]. Each data entry is associated with the normalized re-radiation weights. The details of the derivation on weighted EM can be found in appendix.

5.3. Re-radiation reconstruction

We can reconstruct the re-radiation matrix from the Gaussian Mixture Model found previously. However, the Gaussian Mixture gives a probability density function, hence its integral is unity. The re-radiation integral does not hold this property. So, we scale the Gaussian mixture to reconstruct the re-radiation later on:

$$\Phi\left(\lambda_i, \lambda_o\right) = S \cdot p\left(\lambda_i, \lambda_o\right) \tag{3}$$

where:

S	is the scaling factor to apply,
$\Phi\left(\lambda_i, \lambda_o\right)$	is the re-radiation function represented by a matrix in its tabulated form,
$p\left(\lambda_i, \lambda_o\right)$	is the probability density function of the GMM defined in Eqn. 1,
λ_i, λ_o	are the wavelengths at a given radiating and re-emitting event.

There are different valid strategies to retrieve the scaling factor S. In this paper, we propose two solutions.

Error minimisation The first strategy consists in a minimisation process. S is set as a parameter to minimise the error between the reference dataset and the reconstructed mixture.

$$S = \underset{S}{\arg\min} \sum_{\Lambda_i} \sum_{\Lambda_o} \left\| \Phi_{\text{measured}}\left(\lambda_i, \lambda_o\right) - S \cdot p\left(\lambda_i, \lambda_o\right) \right\| \tag{4}$$

Such minimisation process involves a choice of a norm function. But, this choice has little influence when dealing with fluorescence: the data have low dynamic range. Therefore, we use a L^2 norm in the result section i.e. the mean square error between the reference dataset and the reconstruction defined in Eqn. 3.

By definition, such strategy reduces the average reconstruction error for a given value but, we show in Sec.7 that is provides less accurate rendering.

Integral ratio The second strategy consist in defining S as the ratio between the integral of the original dataset and the GMM.

$$S = \frac{\iint_\Lambda \Phi_{\text{measured}}(\lambda_i, \lambda_o) \mathrm{d}\lambda_i \mathrm{d}\lambda_o}{\iint_\Lambda p(\lambda_i, \lambda_o) \mathrm{d}\lambda_i \mathrm{d}\lambda_o} \tag{5}$$

This method ensures the reconstructed dataset has the same fluorescent albedo as the original dataset. While this technique does not reduce error for each entry, it provides better rendering results: our eyes are more sensitive to brightness variation than to slight color shift as we show in Sec.7.

6. Rendering Phase

During the rendering phase, when a fluorescent event occurs, we first need to sample an in-shifting or an out-shifting wavelength depending on the ray direction. Then, we need to evaluate the attenuation given a set of λ_i and λ_o wavelengths. The sampling phase relies on conditional probability: either λ_i or λ_o is known, and we have to sample a λ_o or a λ_i according to the ray direction, i.e., an in-shifting or out-shifting event.

6.1. Conditional GMM

To sample a random wavelength shift after a fluorescence event, we need to “sample a slice” on the multivariate Gaussian mixture. With a fixed input wavelength λ_i, we construct the conditional parameters for the resulting Gaussian mixture and generate the random variable as the output wavelength λ_o.

In our 2-dimensional dataset, we have:

$$\lambda = \begin{pmatrix} \lambda_i \\ \lambda_o \end{pmatrix}, \quad \mu_k = \begin{pmatrix} \mu_{k_a} \\ \mu_{k_b} \end{pmatrix}, \quad \Sigma_k = \begin{pmatrix} \sigma_{k_{aa}} & \sigma_{k_{ab}} \\ \sigma_{k_{ba}} & \sigma_{k_{bb}} \end{pmatrix} \tag{6}$$

where λ is a sample point, μ_k and Σ_k are the GMM parameters for the k^{th} Gaussian.

We define T, the inversion of k^{th} covariance matrix as:

$$\mathrm{T}_k = \Sigma_k^{-1} = \begin{pmatrix} \tau_{k_{aa}} & \tau_{k_{ab}} \\ \tau_{k_{ba}} & \tau_{k_{bb}} \end{pmatrix} \tag{7}$$

The conditional probability that given λ_i, we shift to λ_o in each single component k is:

$$p_k(\lambda_i \to \lambda_o) = \mathcal{N}\left(\lambda_o \mid \mu_{k_{b|a}}, \tau_{k_{bb}}^{-1}\right) \tag{8}$$

$$\text{with :} \quad \mu_{k_{b|a}} = \mu_{k_b} - \tau_{k_{bb}}^{-1} \tau_{k_{ba}} \left(\lambda_i - \mu_{k_a}\right) \tag{9}$$

Then, we get the probability for the entire conditional GMM from a given λ_i to λ_o:

$$p(\lambda_i \to \lambda_o) = \sum_{k=1}^{K} \pi_k^{(\lambda_i)} p_k\left(\lambda_i \to \lambda_o\right) \tag{10}$$

$$\text{with :} \quad \pi_k^{(\lambda_i)} = \frac{\pi_k \mathcal{N}\left(\lambda_o \mid \mu_{k_a}, \sigma_{k_{aa}}\right)}{\sum_k \mathcal{N}\left(\lambda_o \mid \mu_{k_a}, \sigma_{k_{aa}}\right)}. \tag{11}$$

Finally, we generate the random variable based on the new conditional GMM model, we first choose which Gaussian component we will sample by the conditional mixing coefficients $\pi_k^{(\lambda_i)}$, then we generate each sample following normal distribution using the chosen k^{th} Gaussian:

$$\mu_k^{(\lambda_i)} = \mu_{k_{b|a}}, \quad \sigma_k^{(\lambda_i)} = \tau_{k_{bb}}^{-1}. \tag{12}$$

We have only shown the model for sampling the light path, but it is straightforward to symmetrically apply the model for sampling on an eye path.

6.2. Importance Sampling

When a ray hits a fluorescent element, two events can occur:

- the ray is reflected with the same wavelength as the radiating ray, there is no fluorescence event,
- the ray is re-radiated to a different wavelength, there is a fluorescent event happening.

We first determine if a fluorescent event occurs based on the ratio between the main diagonal and the total reflectance for the ray radiating wavelength. If a fluorescent event occurs, we importance sample the re-radiation. We then use the Gaussian mixture to importance sample this fluorescent event (see Alg. 1).

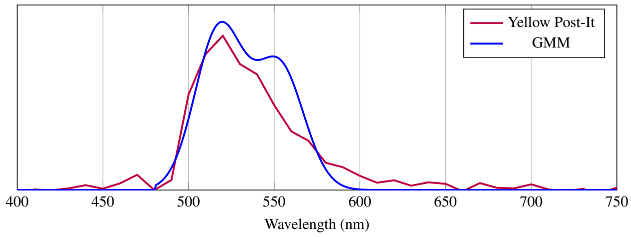


Figure 4: *Example of a conditional PDF derived from the GMM model compared to the unfiltered yellow fluorescent dataset. The figure shows the expected distribution of the reemission wavelength λ_o when the material is radiated at $\lambda_i = 460\,\text{nm}$. When a fluorescent event occur, the outgoing wavelength λ_o is sampled according to this conditional PDF. Notice the measurement noise present in the original data lowers the highest value of the PDF after normalisation when compared to the analytic conditional PDF derived from the GMM.*

Figure 4 show an example of the conditional GMM for a specific radiating wavelength λ_i. Given a radiating wavelength, the model can generate a distribution of reemitted wavelengths. We use this distribution to importance sample the reemitted wavelength λ_o.

The probability for the transition between the set of radiating wavelength λ_i and re-emitting wavelength λ_o is:

$$p_{\text{shift}}(\lambda_i \to \lambda_o) = \begin{cases} \frac{\Phi(\lambda_i,\lambda_i)}{r_t(\lambda_i)} & \text{if } \lambda_i = \lambda_o \\ \left(1 - \frac{\Phi(\lambda_i,\lambda_i)}{r_t(\lambda_i)}\right) \cdot \left(\sum_{k=1}^{K} \pi_k^{(\lambda_i)} \mathcal{N}(\lambda_o \mid \mu_k^{(\lambda_i)}, \sigma_k^{(\lambda_i)}\right) \end{cases}$$

$$r_t(\lambda_i) = \int_\Lambda \Phi(\lambda_i, \lambda_r) \mathrm{d}\lambda_r \tag{13}$$

where:
$\Phi(\lambda_i, \lambda_i)$ is the non-fluorescent reflectance,
$r_t(\lambda_i)$ is the total reflected energy given λ_i.

In our renderer, we use Hero Wavelength Spectral Sampling (HWSS) [WND*14]. The HWSS technique multiplexes multiple wavelengths in a Monte-Carlo (MC) sample. It differs from a simple multiplexing approach by using a “Hero” component which is used to take all decisions on the directional component. This technique drastically reduces colour noise in contexts where there is a wavelength dependency on the directional decisions such as participating medium. In the case of fluorescence, we allow each of the wavelength in the HWSS vector to be independently importance sampled. If done directly, we can find non-zero samples with a zero probability which is incorrect for a MC integrator. Mojzík *et al.* [MFW18] introduced a balance term to avoid this artefact. We use the same term in our implementation:

$$p_j(\lambda_{i_j} \to \lambda_{o_j}) = \prod_{k=1, k \neq j}^{N} p_{\text{shift}}(\lambda_{i_k} \to \lambda_{o_k}) \tag{14}$$

where:
$p_j(\lambda_{i_j} \to \lambda_{o_j})$ is the probability of j^{th} HWSS sample,
$p_{\text{shift}}(\lambda_i \to \lambda_o)$ is the probability of shifting from λ_i to λ_o,
N is the size of the HWSS vector.

Algorithm 1: Random wavelength shifting. Given a λ_i, we either have no fluorescent event, i.e. $\lambda_i = \lambda_o$ or, we sample a λ_o after a fluorescent event.

```
input : $\lambda_i$
output: $\lambda_o$ and pdf $:= p_{\text{shift}}(\lambda_i \to \lambda_o)$
$\xi_1 \leftarrow \mathcal{U}(0,1)$;
$p_{\text{diag}} \leftarrow \frac{\Phi(\lambda_i,\lambda_i)}{r_t(\lambda_i)}$;
if $\xi_1 \le p_{\text{diag}}$ then
    // Sample the main diagonal
    $\lambda_o \leftarrow \lambda_i$;
    pdf $\leftarrow p_{\text{diag}}$;
else
    // Wavelength shifting event
    /* Select one Gaussian from the
       mixture                                  */
    $\xi_2 \leftarrow \mathcal{U}(0,1)$;
    $s \leftarrow 0$;
    for $k \leftarrow 1$ to $K$ do
        $s \leftarrow s + \pi_k^{(\lambda_i)}$;
        if $s \le \xi_2$ then
            break;
        end
    end
    /* Conditional sampling with
       rejection method                         */
    do
        $\lambda_o \leftarrow \mathcal{N}\left(\mu_k^{(\lambda_i)}, \sigma_k^{(\lambda_i)}\right)$;
    while $\lambda_o \le \lambda_i$;
    /* Eqn. 13                                  */
    pdf $\leftarrow (1 - \text{p}_{\text{diag}}) \cdot \left(\sum_{k=1}^{K} \pi_k^{(\lambda_i)} \mathcal{N}(\lambda_o \mid \mu_k^{(\lambda_i)}, \sigma_k^{(\lambda_i)})\right)$;
end
```

6.3. Implementation details

To evaluate the re-radiation, we store for each Gaussian of the mixture, its mean, covariance matrix and weight. We also need to store the scaling factor S. We precompute the determinant and the inverse of the covariance matrix before rendering to speed-up evaluation of the mixture. We store the diagonal in its original tabulated form without any alteration.

To efficiently perform the conditional wavelength shifting we also precompute two additional tabulated values to retrieve $\pi_k^{(\lambda_i)}$ or $\pi_k^{(\lambda_o)}$: the sum of rows and the sum of columns of the Gaussian Mixture. In our implementation, we use a 1 nm sampling rate.

While additional values are precomputed and represent an additional memory footprint, this is still far less than the memory required when using a tabulated CDF which are needed when using only tabulated measurements.

We provide as supplemental material a repository containing the source code for fitting the measured data with Gaussian Mixture Model. The repository also contains a patch for ART renderer [18] to use the computed GMM as input to reproduce the rendering showcased in this paper. It is accessible here: https://gitlab.com/huaq_/compact-representation-fluorescence.

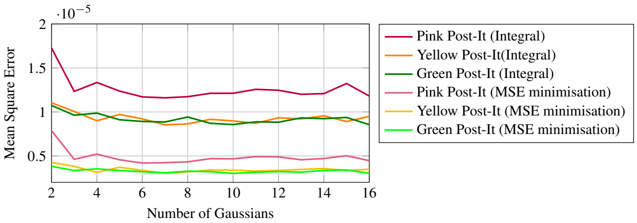


Figure 5: *Mean squared error between the measured datasets and the reconstruction with different number of Gaussians and different weighting factors. We use the same mixture for both techniques to compute the scaling factor. From 3 Gaussians, we observe a significant drop in the reconstruction error using the MSE metric. Adding additional Gaussians does not significantly reduce MSE.*

7. Results and Discussions

Looking at the entirety of our pipeline, we have:

- A reflectance spectrum corresponding to diagonal, the original non-fluorescent part.
- A set of parameters representing the Gaussian mixture, which are:
    - a weighting factor (1 element),
    - a mean vector (2 elements),
    - a covariance matrix (2x2 elements).

To evaluate the accuracy of our reconstruction, we propose to compare the results of the reconstructed re-radiation with the original tabulated data. We use the perceptual comparison on both rendering and reconstruction, with ΔE_{00}^* on rendering using a Monte-Carlo path tracer and the Mean Square Error between the original and reconstructed re-radiation matrices.

In this section, we review the results and discuss the strengths and weaknesses of our model, we especially emphasising on the reconstruction accuracy.

7.1. Re-radiation reconstruction accuracy

We use Mean Square Error (MSE) to evaluate the accuracy of the reconstructed matrices on a per-wavelength basis (see Fig. 5). For all datasets, the MSE significantly drops at 3 Gaussians. Adding additional Gaussians then have little effect on improving the reconstruction accuracy. The MSE shows Pink Post-It dataset is being more challenging to reconstruct: the re-radiation spectrum behaves less like a Gaussian distribution than the two other datasets. Using a different set of basis functions, which can account for skewness would probably improve the accuracy for this specific dataset.

Overall, our method provides good accuracy. Even with a small number of Gaussians, we fit the original datasets well, with a maximum MSE at $1.73 \cdot 10^{-5}$ when using the integral ratio for computing the weighting factor.

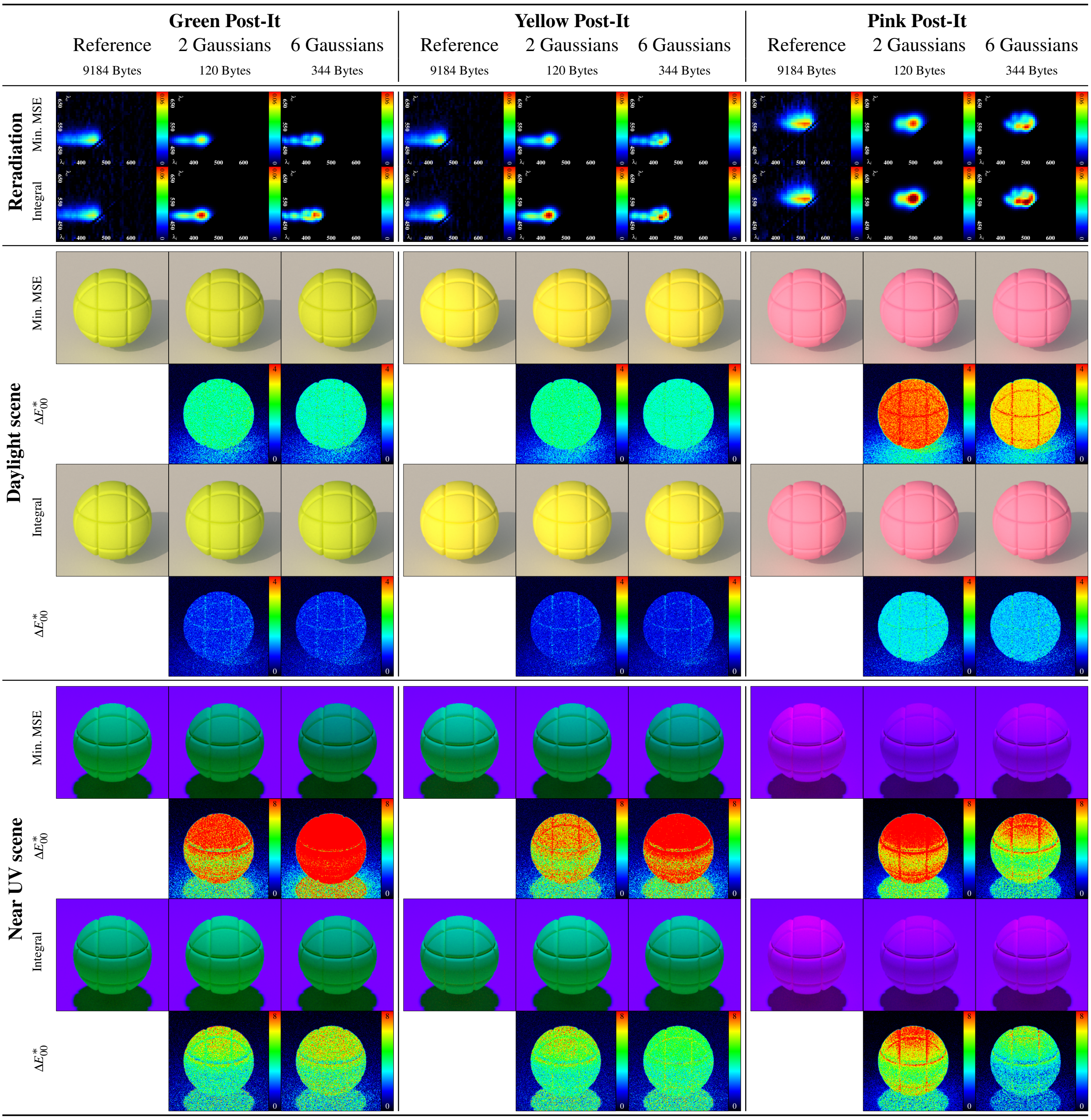


Figure 6: *Comparison of reference datasets (tabulated re-radiation) with the proposed GMM representation using a different number of Gaussians in two different lighting conditions. We compare with two different strategies to scale the GMM for re-radiation reconstruction: minimizing the MSE and integral ratio. The original datasets are measured with the excitation wavelength ranging from* 300 *to* 700 nm *and reemission ranging from* 380 *to* 700 nm *sampled each at* 10 nm*. The original dataset are stored in* 49 *by* 41 *matrices. To store the pure re-radiation spectrum, each of the original dataset needs* 1148 *elements. The GMM representation requires* 7 *elements per Gaussian and a weighting factor to represent the pure-re-radiation. Even with a low number of Gaussian, we get an accurate reconstruction with a minimal memory footprint. Using the integral ratio for the weighting factor gives better results despite the higher MSE (see Fig. 5): our vision system better notice a variation of brightness than a slight wavelength difference. Using the integral ratio better conserves the overall light intensity coming from re-radiation while MSE minimisation better conserves the accuracy on a per-wavelength basis.*

7.2. Rendering accuracy

To evaluate the rendering accuracy, we use CIE-ΔE^*_{00} difference between the reference datasets (tabulated values) and the reconstructed re-radiations using our proposed method with a varying number of Gaussians on renderings using a path tracer. We use three different datasets: measured re-radiation for pink, yellow and green Post-It. These datasets come from the ART open source renderer [18] and originate from Labsphere Inc. according to the metadata included in the measurement archive[†].

The lighting conditions have a significant impact on the error. The fluorescent nature of the dataset makes an objective evaluation of the rendering accuracy difficult. We choose to use two drastically different lighting environments: a daylight scene using a skydome from Hosek and Wilkie [HW12] (this model is appropriate to use here as it contains an ultra-violet component) and a near monochromatic light radiating at 450 nm.

Our technique performs well even with a limited number of Gaussians (see Fig. 6). ΔE^*_{00} are lower than 4 in a daylight environment. Even for challenging cases where most of the energy falls into the re-radiation, with near monochromatic 450 nm illumination, the difference remains low for green and yellow Post-It dataset. For the pink Post-It dataset, the difference is more visible in the near monochromatic lighting. This is expected as this dataset is more challenging to fit with Gaussians.

7.3. Weighting factor

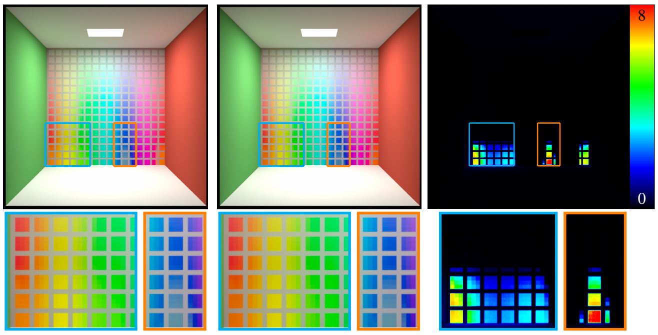
We evaluated two strategies for computing the weighting factor utilised to scale the GMM to reconstruct the re-radiation matrices: minimisation process on the MSE between the original dataset and the scaled GMM and integral ratio (see Sec. 5.3).

As expected, with the minimisation process, we obtain a lower MSE than using the integral ratio. Nevertheless, a lower MSE does not imply better rendering accuracy. In fact, we observed the opposite. At first glance, this might seem surprising. However, the ΔE^*_{00} evaluates visible difference for the human vision system. Our eyes are more sensitive to variation of intensity than to a slight shift in the spectral domain. Using the integral ratio ensures having a similar luminous quantity falling into fluorescence. Note that we did not use any perceptual weighting for computing this ratio. This gives us the insight that the weighting factor could also be optimised to minimise the ΔE^*_{00} given a specific illuminant.

The strategy for computing the scaling factor depends on the target application. The integral ratio is better suited for perceptual applications like rendering with tone mapping. If the accuracy of the spectral reconstruction of re-radiation is needed, then the minimisation of MSE is a better choice.

7.4. Memory footprint analysis

Our method is memory efficient and works with only a fraction of the memory required when using raw tabulated data.



(a) *Bi-spectral tabulated texture.* **(b)** *The same bi-spectral using GMM for re-radiation.* **(c)** ΔE^*_{00} *difference between (a) and (b)*

Figure 7: *With our method, we can drastically reduce the memory footprint of bi-spectral textures. We use the bi-spectral uplifting technique from Jung et al. [JWH*19] to generate a bi-spectral texture from an RGB image. Notice that the bottom part of both images has visible fluorescence. The reference image and our method are rendered with 60k SPP in the same base renderer except for the fluorescent sampling process.*

The measured re-radiation dataset we have used are sampled at a 10 nm rate, for radiating wavelength ranging from 300 nm to 800 nm and reflected wavelength ranging from 380 nm to 800 nm. They are stored in 41 × 49 matrices. In the most favourable case, we only have to store the upper triangular part of the re-radiation due to the nature of fluorescence shifting wavelengths to a lower energy state. If extreme high accuracy is desired, each matrix can be interpolated to a 2-dimensional matrix sampled at a 1 nm rate; this matrix uses 98000 Bytes.

Apart from doing away with having to use the raw re-radiation matrix data, in practice, effective sampling of wavelength shifting is desirable. This requires additional memory space: additional tabulated CDF are computed at the initialisation stage. This triples the memory footprint when exclusively using tabulated data.

With our method, we only need a small set of Gaussian parameters to both evaluate on the fly the re-radiation and to importance sample wavelength shifting. If rendering accuracy is desired, our method can be used solely to importance sample while sparing the storage of tabulated CDF.

We have shown in Fig. 6 our fitting provides a good reconstruction of the tabulated re-radiation matrix with only a few Gaussians. Assume we use 4 Gaussian for fitting, while the tabulated data are sampled at 1 nm rate, the memory difference of one fluorescent material between the tabulated method and our method is approximate 2985776 Bytes.

7.5. Fluorescent textures

We can use our method along with fluorescent texture. In [JWH*19] they use such kind of textures to uplift RGB textures. This technique uses fluorescence on some pixels to enlarge the colour gamut. The generated textures can be very large in mem-

[†] The ART measured bispectral data are available from `ART_Resources/SpectralData/FluorescentPostIt.ark`

ory due to the potential large number of re-radiation matrices. For a 256 × 256 texture, where every pixel is fluorescent, it could require up to 3 × 256 × 256 re-radiation matrices (one used for the attenuation and the two others for precomputed CDFs).

We applied our method to compress spectral uplifted textures from [JWH*19]. Our method accurately renders the fluorescent texture with a $\Delta E^*_{00} \leq 8$ between the reference image and our method (see Fig. 7). The texture is 32 × 32 pixel with 255 fluorescent pixels after the uplifting. The reradiation matrices sizes are 500 by 500 elements. The memory consumption for the diagonal is 4.10 MB and an additional 254.49 MB for the re-radiation. In this rendering, we used 2 Gaussians per fluorescent pixel lowering the re-radiation memory footprint down to 30.60 KB. The total memory footprint of the fully tabulated bi-spectral 32 × 32 texture is 258.59 MB compared to the 4.13 MB needed by when using GMM for the re-radiation. This makes it 62.67 times more memory efficient for this specific dataset. This does not even account for the additional memory overhead of tabulated CDFs desirable for efficient importance sampling of the tabulated dataset.

This ratio is even higher when increasing the resolution of the texture implying a larger number of fluorescent pixels.

8. Conclusion and Future Work

We have presented a compact and efficient method for compressing re-radiation matrices. This allows to drastically lower the memory footprint needed to store a fluorescent spectrum. The memory reduction is useful in the case of fluorescent texture is present, as the large fluorescent texture has a high memory requirement. Our proposed method also enables efficient importance sampling.

In future work, we want to develop a parametric re-radiation model suited for edition. The presented method, while being memory efficient and accurate, lacks the ability for an artist to intuitively edit fluorescence. We also want to evaluate additional metrics for computing the weighting factor used in re-radiation reconstruction depending on the targeted application: visual accuracy or absolute re-radiation difference.

Although we show that Gaussian Mixture can be used to compact fluorescence, due to the limitation of public fluorescent dataset in the graphics community, it is hard to say which parametric model is the best to suit the fluorescence. Add some skewness distribution can be an interesting work to the specific pink fluorescent dataset. The access to more fluorescent data and make them usable in computer graphics will be beneficial to the future work in fluorescent representation.

There is also possibility to try out other fitting methods for the mixture, such as using MAP-EM or non-linear fitting framework with different loss functions. We tried variational bayesian inference for the fitting of the mixture, however, the rendering results can not give us consistent observation. We need more research on that to have a decisive answer on which fitting method is best to the fluorescence data. For the particular problem in color shift in Fig. 6, we think different fitting method might help us with the color shift, but another more potential way to improve that will be to design a human perception metric instead of any distance based metrics.

Acknowledgements

We acknowledge funding by the Czech Science Foundation grant 19-07626S, the EU H2020-MSCA-ITN-2020 grant number 956585 (PRIME), and the Charles University grant number SVV-260588.

References

[18] *The Advanced Rendering Toolkit*. 2018. URL: https://cgg.mff.cuni.cz/ART/ 6, 8.

[Bis06] Bishop, Christopher M. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag, 2006. ISBN: 0387310738 3.

[Don54] Donaldson, R. "Spectrophotometry of fluorescent pigments". *British Journal of Applied Physics* 5.6 (June 1954), 210–214. DOI: 10.1088/0508-3443/5/6/303. URL: https://doi.org/10.1088/0508-3443/5/6/303 2.

[GAFH16] Gebru, Israel Dejene, Alameda-Pineda, Xavier, Forbes, Florence, and Horaud, Radu. "EM algorithms for weighted-data clustering with application to audio-visual scene analysis". *IEEE transactions on pattern analysis and machine intelligence* 38.12 (2016), 2402–2415 4.

[Gla95] Glassner, Andrew S. "A Model for Fluorescence and Phosphorescence". *Photorealistic Rendering Techniques*. Ed. by Sakas, Georgios, Müller, Stefan, and Shirley, Peter. Berlin, Heidelberg: Springer Berlin Heidelberg, 1995, 60–70. ISBN: 978-3-642-87825-1 2.

[HES*18] Herholz, Sebastian, Elek, Oskar, Schindel, Jens, et al. "A Unified Manifold Framework for Efficient BRDF Sampling based on Parametric Mixture Model". *Proceedings of the Eurographics Symposium on Rendering: Experimental Ideas & Implementations*. EGSR '18. Eurographics Association, 2018. ISBN: 978-3-03868-019-2 2.

[HEV*] Herholz, Sebastian, Elek, Oskar, Vorba, Jiří, et al. "Product Importance Sampling for Light Transport Path Guiding". *Computer Graphics Forum* 35.4 (), 67–77. DOI: 10.1111/cgf.12950. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.12950. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.12950 2.

[HHA*10] Hullin, Matthias B., Hanika, Johannes, Ajdin, Boris, et al. "Acquisition and Analysis of Bispectral Bidirectional Reflectance and Reradiation Distribution Functions". *ACM Trans. Graph.* 29.4 (July 2010). ISSN: 0730-0301. DOI: 10.1145/1778765.1778834. URL: https://doi.org/10.1145/1778765.1778834 2.

[HW12] Hosek, Lukas and Wilkie, Alexander. "An Analytic Model for Full Spectral Sky-Dome Radiance". *ACM Trans. Graph.* 31.4 (July 2012). ISSN: 0730-0301. DOI: 10.1145/2185520.2185591. URL: https://doi.org/10.1145/2185520.2185591 8.

[JH19] Jakob, Wenzel and Hanika, Johannes. "A Low-Dimensional Function Space for Efficient Spectral Upsampling". en. *Computer Graphics Forum* 38.2 (May 2019), 147–155. ISSN: 0167-7055, 1467-8659. DOI: 10.1111/cgf.13626. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.13626 (visited on 03/30/2021) 2.

[JRJ11] Jakob, Wenzel, Regg, Christian, and Jarosz, Wojciech. "Progressive Expectation–Maximization for Hierarchical Volumetric Photon Mapping". *Computer Graphics Forum (Proceedings of EGSR)* 30.4 (June 2011). DOI: 10/dtwcjj 2.

[JWH*19] Jung, A., Wilkie, A., Hanika, J., et al. "Wide Gamut Spectral Upsampling with Fluorescence". *Computer Graphics Forum* 38.4 (2019), 87–96. DOI: https://doi.org/10.1111/cgf.13773. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.13773. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.13773 2, 8, 9.

[MFW18] MOJZÍK, MICHAL, FICHET, ALBAN, and WILKIE, ALEXANDER. "Handling Fluorescence in a Uni-directional Spectral Path Tracer". *Computer Graphics Forum* 37.4 (July 2018), 77–94. DOI: 10.1111/cgf.13477. URL: https://hal.inria.fr/hal-01818826 2, 5.

[MSHD15] MENG, JOHANNES, SIMON, FLORIAN, HANIKA, JOHANNES, and DACHSBACHER, CARSTEN. "Physically Meaningful Rendering using Tristimulus Colours". en. *Computer Graphics Forum* 34.4 (July 2015), 31–40. ISSN: 0167-7055, 1467-8659. DOI: 10.1111/cgf.12676. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.12676 (visited on 03/30/2021) 2.

[OYH] OTSU, H, YAMAMOTO, M, and HACHISUKA, T. "Reproducing Spectral Reflectances from Tristimulus Colors". en. (), 11 2.

[Sch] SCHREIBER, JACOB. *pomegranate*. https://github.com/jmschrei/pomegranate 4.

[VKŠ*14] VORBA, JIŘÍ, KARLÍK, ONDŘEJ, ŠIK, MARTIN, et al. "On-line Learning of Parametric Mixture Models for Light Transport Simulation". *ACM Transactions on Graphics (Proceedings of SIGGRAPH 2014)* 33.4 (Aug. 2014) 2–4.

[WND*14] WILKIE, A., NAWAZ, S., DROSKE, M., et al. "Hero Wavelength Spectral Sampling". *Proceedings of the 25th Eurographics Symposium on Rendering*. EGSR '14. Lyon, France: Eurographics Association, 2014, 123–131. DOI: 10.1111/cgf.12419. URL: https://doi.org/10.1111/cgf.12419 2, 5.

[WTP01] WILKIE, ALEXANDER, TOBLER, ROBERT F., and PURGATHOFER, WERNER. "Combined Rendering of Polarization and Fluorescence Effects". *Rendering Techniques 2001*. Ed. by GORTLER, STEVEN J. and MYSZKOWSKI, KAROL. Vienna: Springer Vienna, 2001, 197–204. ISBN: 978-3-7091-6242-2 2.

[WWLP06] WILKIE, ALEXANDER, WEIDLICH, ANDREA, LARBOULETTE, CAROLINE, and PURGATHOFER, WERNER. "A Reflectance Model for Diffuse Fluorescent Surfaces". *Proceedings of the 4th International Conference on Computer Graphics and Interactive Techniques in Australasia and Southeast Asia*. GRAPHITE '06. Kuala Lumpur, Malaysia: Association for Computing Machinery, 2006, 321–331. ISBN: 1595935649. DOI: 10.1145/1174429.1174484. URL: https://doi.org/10.1145/1174429.1174484 2.

Appendix A:

The deviation of weighted EM used in the paper.

Given an observed data set $X = \{x_1, x_2, ..., x_n\}$, for each observation in X, there is a corresponding latent variable in the set $Z = \{z_1, z_2, ..., z_n\}$ which helps to model dependencies between variables. The data set X, Z is seen as a complete data set.

When using EM to fit a mixture of K Gaussians, the latent variable set Z is a set of N-dimensional vector. Each component z_n is a K-dimensional binary random variable, where z_{nk} denotes the k^{th} element of z_n. $z_{nk} \in \{0,1\}$, $\sum_k z_{nk} = 1$. This latent variable vector represents which Gaussian component generated our sample x_n with probability $p\left(z_{nk} = 1\right) = \pi_k$.

Assume we have a dataset $X = \{x_1, ..., x_n\}$, the corresponding weight vector for each sample in X is $W = \{w_1, ..., w_n\}$. The mixture of Gaussian is:

$$p(X|\theta, W) = \sum_{k=1}^{K} \pi_k \mathcal{N}\left(X|\mu_k, \frac{1}{W}\Sigma_k\right). \tag{15}$$

The responsibility function is thus:

$$\gamma(z_{nk})^{(\text{weighted})} = \frac{\pi_k \mathcal{N}\left(\mathbf{x}_n \mid \mu_k, \frac{1}{w_n}\Sigma_k\right)}{\sum_{j=1}^{K} \pi_j \mathcal{N}\left(x_n \mid \mu_j, \frac{1}{w_n}\Sigma_j\right)}. \tag{16}$$

The means, covariances and mixture coefficients used to maximise the expected complete-data log-likelihood are:

$$\mu_k^{(\text{weighted})} = \frac{1}{N_k} \sum_{n=1}^{N} w_n \gamma(z_{nk})^{(\text{weighted})} \mathbf{x}_n, \tag{17}$$

$$\Sigma_k^{(\text{weighted})} = \frac{1}{N_k} \sum_{n=1}^{N} w_n \gamma(z_{nk})^{(\text{weighted})} \left(\mathbf{x}_n - \mu_k\right)\left(\mathbf{x}_n - \mu_k\right)^{\mathrm{T}}, \tag{18}$$

$$\pi_k^{(\text{weighted})} = \frac{N_k}{N}, \tag{19}$$

$$N_k^{(\text{weighted})} = \sum_{n=1}^{N} \gamma(z_{nk})^{(\text{weighted})}. \tag{20}$$