ColorLab: Visualizing Color from Absorbance Spectra

Spencer M. Yeager,a Michael A. Anderson,b Priscilla Babiak, Bryon W. Larson,c Erin L. Ratcliffabd*
a. Department of Chemistry and Biochemistry, University of Arizona, Tucson, AZ 85721
b. Department of Materials Science and Engineering, University of Arizona, Tucson, AZ 85721
c. Chemistry and Nanoscience Center, National Renewable Energy Laboratory, Golden, CO 80401
d. Department of Chemical and Environmental Engineering, University of Arizona, Tucson, AZ 85721
* Corresponding Author Contact: ratcliff@arizona.edu

Keywords: ColorLab, Degradation, Data Visualization, Absorbance, Color

We present here the first public release of ColorLab, a Python-based program that can convert absorbance spectra into color images. It was designed for use with organic photovoltaic (OPV) materials and blends, which represent a myriad of colors based on molecular design and material blending that can exhibit persistent color or evolve over time via degradation or morphology changes. However, ColorLab is not limited to this application, and can generate color images from a single spectrum or an evolving color bar on a time axis from multiple time-stamped spectra. Using internationally defined illuminants, ColorLab can display colors that are representative of a variety of lighting situations, from indoor to outdoor. The development of this program aims to aid with the visualization of semitransparent materials and to connect researchers with designers, through conversion of spectra to color.

1. Introduction

Visible light is the region of the electromagnetic spectrum that spans the wavelengths of 380 nm to 780 nm.1 This wavelength range gives rise to the vibrant variety of colors that are seen every day. Beyond the aesthetic beauty that color can provide, color is a visual language that conveys information throughout society. The most common examples are traffic signals, which use red to signal stop, yellow to signal caution, and green to signal go. The association between aesthetics and information makes color a powerful universal form of communication, something that designers consider when utilizing colors in a project or artists consider when conveying emotion in their work. In science, color is often used differently. Rather than using the qualitative description of color, scientists employ quantitative techniques such as absorbance spectroscopy. In absorbance spectroscopy, a sample is exposed to a range of wavelengths and the spectral attenuation of those wavelengths are measured by a detector as
transmission values. The transmission of light at each wavelength is then converted to an absorbance value, indicating how strongly a particular sample attenuates, by absorbing/transmitting, a given spectrum of incident light. In visible absorbance spectroscopy, the wavelengths used correspond to the visible color range, thereby linking a particular sample’s absorbance profile to its visible color, which by default is a metric referenced to the human eye. This fundamental technique is taught to scientists-in-training as early as high school science classes, and used in a variety of disciplines including materials science, chemistry, biochemistry, and physics. An absorbance plot can provide a variety of material-dependent information that takes a trained individual in their respective field to identify. The need for a trained individual to interpret this data means that the universal approachability of color is lost. To translate absorbance data for interpretation by every day users, an additional tool is needed to readily convert between absorbance and RGB color. Herein, we present the first public release of ColorLab, a Python-based program that can convert absorbance spectra into an image of the corresponding representative RGB color. ColorLab utilizes internationally-defined illuminant values to determine how an absorbance spectrum will appear in color under a variety of lighting conditions. The ability to convert absorbance spectra to a color bridges a gap between science and a more general, design-oriented audience. The resultant images can be used as supplementary information for experimental researchers when investigating material properties, or allow ab initio computational researchers to translate simulated spectra into colors.

While Version 1.1.0 of ColorLab is showcased here, the most up to date version can always be found in the following GitHub repository: https://github.com/LISPEM/ColorLab.

2. Use and Implementation

2.1 Initiating ColorLab

ColorLab is written in Python and utilizes GLP-2.0 license. To use ColorLab on Windows, we highly recommend installing Anaconda as it includes all required packages. The “clgui.py” file must be run to start ColorLab, with specific operating system instructions provided in the README file. Running “clgui.py” will open the ColorLab GUI as shown in Figure 1. Navigate to the folder containing absorbance data, select the desired illuminant, color saturation scalar, and aspect ratio for the figure, and optionally include a title. “Load Files” will generate the image, saving it in an “images” subdirectory within the ColorLab working directory.
Figure 1. ColorLab’s interface.

2.2 Input Data

ColorLab is able to read files that are comma separated (either in .csv or .txt file formats), or saved in the Microsoft Excel .xls file format. Absorbance or transmission data can be used through selection of the appropriate data type radio button. If absorbance data is used, two column headers must be present in the selected data file: “Wavelength” and “Absorbance”. If transmission data is used, it must be fractional transmission, not percent transmission, and contain “Wavelength” and “Transmission” as column headers. The “AIPS” radio button is used for the specific format in which data is exported from the AIPS system at the National Renewable Energy Laboratory.8 A single spectrum or time-dependent spectra can be used as input data. If there is only a single file in the target directory, ColorLab will take the single spectrum and generated an image of the corresponding color. If there are multiple files in the directory for time-dependent spectra, they must have the corresponding time in seconds present at the end of the filename, separated by an underscore (“_”) from the rest of the filename characters. A sample dataset has been provided that can be used as a template for formatting data and filenames. When multiple spectra are used, ColorLab will determine the color at each time point and will generate an image of the chronological color evolution with time on the x-axis. Each spectrum will be represented by a color bar that begins at its associated time, and span the x-axis until the start of another spectrum. For example, four spectra will result in four bars of color on the image, in chronological order. To improve the color gradient effect, decrease the time interval between recorded spectra.
2.3 Illuminants

ColorLab works by utilizing a set of CIE illuminants to convert absorbance data into specific color representations. Each illuminant included in Version 1.0.0 is corresponds to a list of wavelength-dependent spectra power densities defined by the International Commission on Illumination (CIE). Each illuminant will be described to provide insight into the lighting condition it represents and how each color is generated.

**CIE Standard Illuminant A.** CIE Standard Illuminant A is “intended to represent typical, domestic, tungsten filament lighting”, and is best used when simulating color in a residential lighting situation. It is based on a black body radiator emitting at a temperature of approximately 2,848 K. Equation 1 defines the spectral distribution of Standard Illuminant A:

\[
S_A(\lambda) = 100 \left( \frac{560}{\lambda} \right)^5 \times \frac{\exp \left( \frac{1.435 \times 10^7}{2848 \times 560} - 1 \right)}{\exp \left( \frac{1.435 \times 10^7}{2848\lambda} - 1 \right)}
\]  

(1)

Where \( \lambda \) is the measured wavelength in nm, 2848 represents the temperature of the blackbody radiator in K, and \( 1.435 \times 10^7 \) is a constant.\(^9\)

**CIE Standard Illuminant DX.** CIE Standard Illuminants DX are representations of different daylight colors. The number following D is indicative of the associated black body radiator temperature divided by 100. Table 1 refers to each of the daylight illuminants included in ColorLab, along with a description of its intended color representation.\(^10\)
Table 1. Daylight illuminants included in ColorLab. Descriptions of each color are adapted from the Encyclopedia of Color Science and Technology.\textsuperscript{10}

<table>
<thead>
<tr>
<th>Illuminant</th>
<th>Temperature (K)</th>
<th>Intended Color Representation</th>
</tr>
</thead>
<tbody>
<tr>
<td>D50</td>
<td>5,000</td>
<td>Daylight when the sun is rising or setting, alternatively early morning or late afternoon. Colors will appear “warmer”</td>
</tr>
<tr>
<td>D55</td>
<td>5,500</td>
<td>Average daylight at the start of the afternoon, with special note that it is more akin to a summer afternoon.</td>
</tr>
<tr>
<td>D65</td>
<td>6,500</td>
<td>Average daylight at the start of the afternoon. Colors will appear “warmer”</td>
</tr>
<tr>
<td>D75</td>
<td>7,500</td>
<td>Daylight when the blue sky is dominant, i.e. there are no clouds in the sky. Colors will appear “cooler”</td>
</tr>
</tbody>
</table>

**Illuminant C.** Illuminant C is another representation of average daylight.\textsuperscript{9} The spectral distribution of this illuminant is determined such that contributions from the UV and IR are filtered out. CIE no longer recommends the use of illuminant.

**FX Illuminants.** Three fluorescent bulb spectral power distributions are included to provide indoor lighting options. The three included fluorescent bulbs included are FL2, FL7, and FL11. FL2 is classified as normal, FL7 is classified as broad-band, and FL11 is classified as multi-band. For more information about what each of these classifications mean, refer to Appendix A5.3 in *Measuring Colour*.\textsuperscript{11}

**2.4 Aspect Ratio**

The aspect ratio changes the height/width ratio of the image. For example, an aspect ratio of 0.5 will result in an image twice as wide as it is tall.

**2.5 Generating RGB value from Wavelengths**

ColorLab utilizes a set of calculations to convert contributions from both absorbance and wavelength to a corresponding RGB color value.\textsuperscript{9} The main steps are be outlined here and the calculations for this conversion can be found in *dataManager>CIE_XYZ.py*.

To begin, wavelengths must be converted to XYZ tristimulus values, which act as intermediary values between spectral radiance and color value. The X, Y, and Z tristimulus
values can be determined using the fractional transmission of a particular sample. If absorbance data is used, it must be converted to fractional transmission values. Equations 2-5 can be used:

\[ X = k \int_{380 \text{ nm}}^{780 \text{ nm}} T(\lambda)S(\lambda)\bar{x}(\lambda)d\lambda \]  
\[ Y = k \int_{380 \text{ nm}}^{780 \text{ nm}} T(\lambda)S(\lambda)\bar{y}(\lambda)d\lambda \]  
\[ Z = k \int_{380 \text{ nm}}^{780 \text{ nm}} T(\lambda)S(\lambda)\bar{z}(\lambda)d\lambda \]  
\[ k = \frac{1}{\sum_{\lambda} S(\lambda)\bar{y}(\lambda)d\lambda} \]  

Where \( T(\lambda) \) is the experimentally determined transmission value, \( S(\lambda) \) is the spectral power distribution of the chosen CIE illuminant\(^{12} \), \( \bar{x}(\lambda) / \bar{y}(\lambda) / \bar{z}(\lambda) \) are wavelength-dependent values associated with CIE color matching functions\(^{13} \), and \( k \) is a normalization factor based on the Y stimulus value. These equations must be integrated over the entire visible light range, hence a range of 380 nm to 780 nm. ColorLab utilizes a summation approximation to determine the integrals of the tristimulus values, shown in Equations 6-8:

\[ X = k \sum_{\lambda} T(\lambda)S(\lambda)\bar{x}(\lambda)\Delta\lambda \]  
\[ Y = k \sum_{\lambda} T(\lambda)S(\lambda)\bar{y}(\lambda)\Delta\lambda \]  
\[ Z = k \sum_{\lambda} T(\lambda)S(\lambda)\bar{z}(\lambda)\Delta\lambda \]  

Where \( \Delta\lambda \) is the wavelength interval. Standard computer screens, when employing sRGB color encoding functions, represent colors using Standard Illuminant D65. To account for differences in the white point between each of the different illuminants, a chromatic adaptation step much be taken to convert the XYZ tristimulus values from one illuminant into the representative XYZ tristimulus values under Standard Illuminant D65. The linearized form of the Bradford Transform is used and described generally in Equations X-Y. The corresponding Bradford RGB values of the white point for the chosen illuminant and the white point of Standard Illuminant D65 must be calculated. These values can be calculated using Equation 9, where the XYZ white point for the chosen illuminant and D65 are used as the \( X_{wp} Y_{wp} Z_{wp} \) values.
\[
\begin{bmatrix}
R_{wp} \\
G_{wp} \\
B_{wp}
\end{bmatrix} =
\begin{bmatrix}
0.8951 & 0.2664 & -0.1614 \\
-0.7502 & 1.7135 & 0.0367 \\
0.0389 & -0.0685 & 1.0296
\end{bmatrix}
\begin{bmatrix}
X_{wp} \\
Y_{wp} \\
Z_{wp}
\end{bmatrix}
\]

(9)

With the Bradford RGB white points for both the chosen illuminant and Standard Illuminant D65 calculated, Equation 10 can be used to calculate an intermediary matrix used in the final expression:

\[
[I] =
\begin{bmatrix}
\frac{Rd}{Rs} & 0 & 0 \\
0 & \frac{Gd}{Gs} & 0 \\
0 & 0 & \frac{Bd}{Bs}
\end{bmatrix}
\]

(10)

Where \(X_d\) is the D65 R, G, or B white point, and \(X_s\) is the chosen illuminant R, G, or B white point. The adapted XYZ values can then be calculated using Equation 11:

\[
\begin{bmatrix}
Xa \\
Ya \\
Za
\end{bmatrix} = [B] * [I] * [B]^{-1} *
\begin{bmatrix}
X \\
Y \\
Z
\end{bmatrix}
\]

(11)

Where the input XYZ values are calculated from Equations 6-8. The resulting XaYaZa values can then be used for the RGB tristimulus value calculation. Specifically, the XaYaZa values are encoded using sRGB.¹⁴ This is performed using Equation 12:

\[
\begin{bmatrix}
R \\
G \\
B
\end{bmatrix} =
\begin{bmatrix}
3.2410 & -1.5374 & -0.4986 \\
-0.9692 & 1.8760 & 0.0416 \\
0.0556 & -0.2040 & 1.0570
\end{bmatrix}
\begin{bmatrix}
Xa \\
Ya \\
Za
\end{bmatrix}
\]

(12)

The calculated RGB values must then be transformed into nonlinear sR’G’B’, known as a gamma correction. The input RGB values must lie between 0 and 1. Any value above 1 is simply clipped so that it is equal to 1. After any necessary clipping, Equation 13 can be used for the gamma correction, where \(C\) represents R, G, or B:

\[
\text{If } C \leq 0.00304:\n\quad C' = 12.92 \times C
\]

\[
\text{else if } C > 0.00304:\n\quad C' = 1.055 \times C^{\frac{1}{2.4}} - 0.055
\]

(13)

\(C'\) represents the resulting nonlinearized sR’G’B’ tristimulus values. These components are then encoded as shown in Equation 13 to result in a final, usable rgb value to represent color:
\[ r = 255R' \]
\[ g = 255G' \]
\[ b = 255B' \]

For each analyzed spectra, the corresponding rgb value is stored in an array. To display this color as an image, Matplotlib’s `imshow` function is used.

### 3. Demonstration of ColorLab: Organic Photovoltaic Materials

To demonstrate the capabilities of ColorLab, a sample data set of organic photovoltaic (OPV) materials is highlighted. OPVs are promising thin-film solar technologies with a wide range of applications\(^{15-17}\). These materials have highly tunable optical properties through synthetic modification, which results in a wide variety of colors. Left exposed to harsh conditions, these materials can be susceptible to chemical degradation, and therefore color change. To visualize the color change of these films over time, as opposed to looking at an absorbance waterfall plot, ColorLab can be used to turn a series of spectra into a color gradient image. This visualization of transient absorptance data is useful to researchers as it gives a rapid qualitative assessment of material stability, aids in the understanding of degradation behavior, and it adds time-dependent aesthetic information which is critical for market adoption in real-world applications. We selected a common OPV blend consisting of a polymer donor, poly[1,3,4-thiadiazole-2,5-diyl(3-octyl-2,5-thiophenediyl)][4,8-bis[(2-butyloctyl)thio]benzo[1,2-b:4,5-b']dithiophene-2,6-diyl][4-octyl-2,5-thiophenediyl)] (PBDTS-TDZ, CAS: n.a.), and a non-fullerene acceptor, 2,2'-(2Z,2'Z)-(((4,4,9,9-tetrakis(4-hexylphenyl)-4,9-dihydro-sindaceno[1,2-b:5,6-b']dithiophene-2,7-diyl)bis(4-((2-ethylhexyl)oxy)thiophene-5,2- diyl))bis(methanylylidene))bis(5,6-difluoro-3-oxo-2,3-dihydro-1H-indene-2,1-diylidene))dimalononitrile (IEICO-4F, CAS: 2089044-02-8), to showcase the use of ColorLab for OPVs. The data set used to generate the figures can be found here (https://doi.org/10.5281/zenodo.6332946).\(^{18}\) The initial color of the pristine film is shown in Figure 2 as generated by ColorLab using only the first recorded (time zero) absorption spectrum.
In Figure 2, the RGB color fills the entire image area. This is useful for highlighting the color from a single absorbance spectrum. Figure 3 shows the color evolution of the OPV blend as generated by ColorLab using multiple files of time-dependent absorption data during 140 hours of intentional ambient photodegradation.

In Figure 3, the time-dependent spectra make up a gradient of colors that progresses from a pale pink to light green color over the course of approximately 144 hours. From 0 hours to approximately 60 hours, there is a smooth transition between the colors, indicating that the time between data collection at early times was short. From 60 hours and onwards, there are
more visible color bars, since the time between data acquisition increased at longer times. Hence, the “smoothness” of the gradient in the image is determined by the data collection interval over the duration of the time change measured. More spectra between a specific time interval results in a smoother gradient, whereas less spectra results in more visible bars.

4. Conclusion

ColorLab is a Python-based tool that can be used to visualize absorbance spectra as their corresponding color. It can display multiple spectra for visualizing material color changes, regardless of whether these changes are by design such as in the case of switchable materials or as a function of degradation over time. Additionally, ColorLab can process a single spectrum to show a single image of its corresponding color. ColorLab is a valuable program to help researchers across multiple disciplines understand visual properties of their materials. Additionally, materials with commercial applications can be easily translated to a color, extending the impact of work to include the design community.

Acknowledgements

This work was supported by the National Science Foundation under grant award DMR-2003631. This was authored in part by the National Renewable Energy Laboratory, operated by Alliance for Sustainable Energy, LLC, for the U.S. Department of Energy (DOE) under Contract No. DE-AC36-08GO28308 with writing support for BWL by ARPA-E DIFFERENTIATE program under grant no. DE-AR0001215. The views expressed in the article do not necessarily represent the views of the DOE or the U.S. Government.

References

(5) Han, Y.; Tian, Z.; Zhang, S.; Liu, X.; Li, J.; Li, Y.; Liu, Y.; Gao, M.; Liu, Z. Half-Sandwich IridiumIII N-Heterocyclic Carbene Antitumor Complexes and Biological


