Journal of Engineering, Management and Information Technology

ASPUR PUBLISHING

Vol. 04, No. 04 (2026) 351-360, DOI:10.61552/JEMIT.2026.04.001 - http://jemit.aspur.rs

PYTHON-BASED GAUSSIAN MODELING AND VALIDATION OF YELLOW PHOSPHOR EMISSION SPECTRUM USING NORMALIZED CROSS-CORRELATION

Huynh-Tuan-Anh Nguyen
Nguyen-Quang-Huy Tran
Anh-Thu Nguyen
Nguyet-Thuan Phan
Quang-Khoi Nguyen[1]

Received 27.04.2026.
Revised 14.06.2026.
Accepted 06.07.2026.

Keywords:

PYTHON Spectrum Modeling, PYTHON Yellow Phosphor, PYTHON Normalized Cross Correlation.

Original research

A B S T R A C T

The spectrum modeling of emission spectrum of yellow phosphor YAG:Ce3+ is an important role in investigation the optical properties of white light. We proposed and demonstrated a novel solution for modeling the emission of yellow phosphor YAG:Ce3+ by using the PYTHON software. The emission curve is mathematically described by using the Gaussian function. This mathematical model is simulated by programming in PYTHON. The similarity between the simulation and experimental results are compared by normalized cross correlation (NCC) algorithm. All NCCs show high values larger than 99.5% which indicates the accuracy and reliability of the model. The proposed model is interesting and meaningful for emission spectrum modeling using the PYTHON software programming. Its applications offer a blend of cost-efficiency, speed, and comprehensive insight while allowing for extensive control and customization. The combination of these advantages positions this method as a powerful tool in materials science, lighting technology, and display.

© 2026 Journal of Engineering, Management and Information Technolog

1. INTRODUCTION

Light Emitting Diodes (LED) base solid state lighting has become the dominant light source due to its advantages of energy saving, color vividness, compact size, high luminous efficiency, and environment friendly due to free mercury (Schubert & Kim, 2005; Narendran & Gu, 2005; Zhu & Humphreys, 2016). The most popular method in industrial lighting to generate white light is using a blue LED die covered with yellow phosphor YAG:Ce^{3+} (Berends et al., 2020). The type of yellow phosphor has a broad emission band (e.g. wavelength range from 500 nm to 750 nm) which is a high overlapping level to the human eye's sensitivity curve (Xie et. al., 2011). These unique characteristics help generate a high luminous efficiency while saving energy. Figure 1 illustrates the white light mixing process with the yellow and blue light combination (Chen et al., 2010). The blue light from blue LED excitation sources pump to the yellow phosphor will be absorbed by the yellow phosphor (Jang & Jeon, 2007). The amount of absorbed blue light will be converted to yellow light. The unabsorbed blue light is transmitted through the phosphor layer and mixed with converted yellow light.

[1] Corresponding author: Quang-Khoi Nguyen
Email: nqkhoi@hcmus.edu.vn

The mixture of light causes a white feeling in the perceiving of humans' eyes. Under the excitation of blue light, electrons in the $(5s^25p^6)4f^1$ ground state are moved up to the $(5s^25p^6)5d^1$ excited state. Through a down-conversion process, the down-conversion of electrons between the $(5s^25p^6)5d^1$ excited state and $(5s^25p^6)4f^1$ ground state releases the energy as a broad emission band in the yellow light region of the visible region. As a result, an asymmetry band of the emission spectrum of yellow light is obtained as shown in Figure 2.

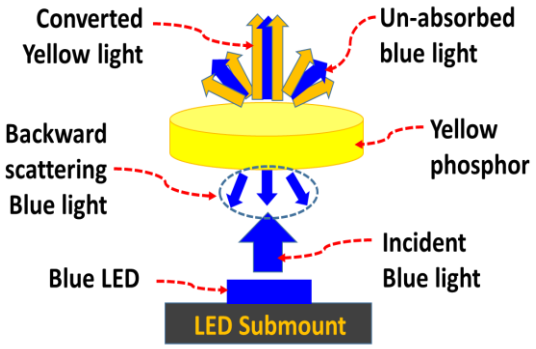

Figure 1. Illustrates the principle of white light generation of yellow phosphor under excitation of blue LED sources.

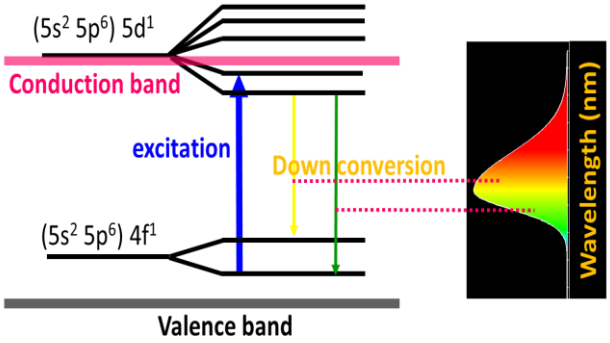

Figure 2. The electronics transition during long wavelength conversion process of yellow phosphor.

The simulation of the emission spectrum has attracted much attention. De Bonfils et al. (2008) reported that a simplified inactive rare-earths doped nuclear waste glasses have been obtained by molecular dynamics (MD) simulation in order to investigate the local structure around the rare-earth by luminescence studies. MD calculations were performed with modified Born–Mayer–Huggins potentials and three body angular terms representing Coulomb and covalent interactions. The simulated emission spectra are obtained by convolution of all the Eu^{3+} ions contributions. A comparison with the experimental data issue from fluorescence line narrowing and micro luminescence spectroscopies allowed us not only to validate the simulation of luminescence spectra from simulated environments, but also to confirm the presence and the identification of two major Eu^{3+} sites distribution in the nuclear glasses thanks to spectra-structure correlations (Peres, 1999). Koryukina & Koryukin (2017) simulated the emission spectra of Ne and Ar atoms in an inductive discharge. The influence of the alternating electric field generated by this discharge on the spectral characteristics of light sources is studied (Nagi et al., 2020). Based on the results of calculations, the regularities in the behavior of spectra characteristics of Ne and Ar atoms in the electric field have been revealed. Practical applications of the simulation results are considered (Koryukina & Koryukin, 2017). Avtaeva (2023) developed a modeling of the Sulfur emission in the asymmetric pulse dielectric barrier discharge. The asymmetric pulse-periodic dielectric barrier discharge (DBD) in a mixture of argon with sulfur vapor exhibits the emission spectrum of the discharge with intense bands of S_2 molecules in the visible region. The modeling of the discharge characteristics in Ar–S_2 mixture within the 1D fluid model is presented showing that during a voltage pulse of 150 ns, an electropositive plasma is formed in the entire discharge gap with the electron number density reaching values above 1,5.1018 m^{-3} at the maximum under the mean electron energy of about 3 eV. The high electron number density during the current pulse leads to efficient excitation of sulfur dimers, as a result, their radiative efficiency during the voltage pulse reaches approximately 4%. The discharge is considered as promising one for creating radiation source in visible region of spectrum (Avtaeva, 2023). Xie et al. (2011) investigated a secondary electron emission model for photo-emission from metals in the vacuum ultraviolet. The two secondary electron emission models for photoelectric energy distribution curves, absolute quantum efficiency, and the mean escape depth of photo-emitted electrons of metals have been developed. The proposed models are developed based on the density of states and the theories of photo-emission in the vacuum ultraviolet. The mean probability that an internal photo-emitted electron escapes into vacuum upon reaching the emission surface of the metal, and the mean energy of photo-emitted electrons measured from vacuum are utilized in this emission modeling (Xie et al., 2011). Suryan Sivadas et. al. reported that the atomic emission spectroscopy is a potential tool for qualitative and quantitative elemental analysis of multi-element materials. The emission spectrum is composed of the characteristic lines of the constituent species, and their intensities are proportional to the population of their energy states and transition probability. The authors have simulated the emission spectrum of copper-tin-zinc alloys under typical laser-induced plasma conditions. Spectra are generated for varying temperatures, electron density, and compositions and show variations based on these parameters. This simulation provides a simple and convenient way to perform laser-induced breakdown spectroscopy applications on any alloy sample (Pasquini et al., 2007). Nguyen (2025) have proposed and demonstrated a precision optical model for red light-emitting diodes and applied this model to improve the color rendering index of phosphor-converted white light-emitting diodes. The inherent asymmetry properties of red light emission bands have been conquered and

effectively modeled. The accuracy of the red light optical model is confirmed with the experimental spectrum and verified by normalized cross-correlation. In an application, the Gaussian functions-based mathematical model is applied to simulate the emission spectrum of white light with and without adding red light. The supplement of red light is optimized to improve the CRI index of output white light. As a result, the color performance of white LEDs is improved when the CRI value significantly increases from 79 to 93 by adding an optimized amount of red light. The effect of added red light on white-LEDs properties has been investigated including correlated color temperature, output lumen, CIE-chromaticity coordinates, and luminous efficiency of radiation. In addition, the effect of adding red light on color rendering quality of output white light is studied with the IES TM 30–18 application (Nguyen, 2025). Nguyen has reported a mathematical model for emission spectrum modeling that is simple, easy to use, understandable, and applicable. The model can be applied to design an emission spectrum using LED, studying the optical and color performance of the emitted white light spectrum. This efficient method is helpful for researchers working in the solid-state lighting (SSL) field, spectrum design for light sources, and optimizing the optical and electrical performance of any specific desired light source (Nguyen, 2025). Nguyen (2025) developed an efficient optical model for simulating the white light spectrum design with high color performance. The optical model began with experimental data collection, simulation with the mathematical model using Gaussian functions, and verifying the similarity between simulation and experiment. In application, the model is applied to study the effect of red light on the optical properties of the generated white light spectrum. The obtained results indicate that the established model is helpful for the solid-state lighting application to fabricate a light source that can emit white light with high optical performance (Nguyen, 2025). Nguyen (2025) developed an optical model for green phosphor pumped by ultraviolet (UV) laser. The model began by defining an efficient mathematical model to describe the emission of the UV band and green light. Next, the data of peak emission wavelength, the width of the emission band, and the emission wavelength range are extracted from the measurement of the emission spectrum of green phosphor pumped by a UV laser. These parameters are put in the mathematical model to simulate the emission spectrum of green phosphor pumped by a UV laser. Finally, the optical model is verified with experimental measurements. The similarity of simulation and experiment is quantified using a normalized cross-correlation algorithm.

Spectrum modeling can help to predict the optical and color properties of output light (Fan et al., 2020; Jacquemin et al., 2009; Lokesh et al., 2025). Traditional methods use Excel to calculate the power spectrum distribution of white light (Islam et al., 2013; Houser et al., 2013). This method is simple and does not require advanced software, however, it is time-consuming. The other method to simulate the SPD is using Matlab. This method is fast and convenient for those who are familiar with Matlab. Along with the development of information technology, many programming languages are launched such as C, C++, and Python. This leads to a requirement to apply these new programming languages in scientific and technology-related works. Python has become preferred and widely used due to it being an open-source programming language (Srinath, 2017). The application of Python in the solid-state lighting field is still in the beginning stages. Thus, it motivated us to conduct this research to promote the development of this field by using the PYTHON programing language.

In this paper, we studied and verified a novel method to simulate the yellow phosphor's emission spectrum. The emission is described by the Gaussian function which is simulated effectively by programming in PYTHON. Emission spectra with different bandwidths are simulated easily. The similarities between the simulation and experimental results are compared by the normalized cross-correlation (NCC) algorithm. The proposed model is clear evidence to prove that PYTHON software programming is an interesting and meaningful tool in the solid-state lighting field.

2. METHODOLOGY

Figure 3 (a) shows the asymmetric band of the emission spectrum of yellow light.

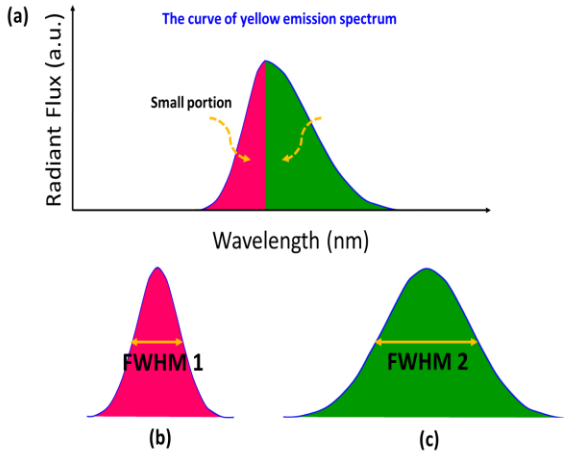

Figure 3. Illustration of analysis of the curve of the yellow emission spectrum to simplify the simulation process

To simplify the simulation, the detailed analysis of the emission curve of the yellow emission spectrum is described as follows. At the peak emission wavelength position of the yellow band, the emission band is divided into two parts by a normal to the wavelength axis. These two halves of a bell shape as shown in Figures 3(b) and (c). These two bell shapes have the same peak emission wavelength but have different full widths at haft maximum (FWHM). Based on these characteristics, the

simulation of the asymmetric band of the emission spectrum of yellow light as shown in Figure 3(a) is described as follows. To simulate the halves of the curve of the bell shape with FWHM 1, the wavelength range is from the value of wavelength 1 to peak wavelength. Half of the curve of the bell shapes with FWHM 2, the wavelength range is from the value of peak wavelength to wavelength 2.

In terms of mathematical representation, the emission curve with the bell shape is described by Gaussian functions.

$$P(\lambda) = P\ exp\left[-\beta\left(\frac{\lambda - \lambda_{peak}}{\Delta E}\right)^2\right] \quad (1)$$

Where β is the corrected coefficient, ΔE is the FWHM in nm, $\boldsymbol{\lambda}_{peak}$ is the peak wavelength of the yellow emission band, $\boldsymbol{\lambda}$ is the wavelength range for each bell shapes emission region and P is the optical power of emission band (W).

In validation the model, the beta value which has the highest level of overlapping of the simulated and experimental spectrum should be defined. The precise comparison is conducted through quantitative evaluation using the normalized cross correlation (NCC) algorithm**.** The similarity between simulation and experiment spectrum curves is quantitatively evaluated based on the value of NCC (e.g. the value of 100 for NCC means the complete similarity between the simulated and experimental spectrum). The mathematical expression of the NCC algorithm is described as follows.

$$NCC = \frac{\sum_n (S_{\lambda n} - \bar{S})(E_{\lambda n} - \bar{E})}{\sqrt{(\sum_n (S_{\lambda n} - \bar{S})^2)(\sum_n (E_{\lambda n} - \bar{E})^2)}} \quad (2)$$

Where $\bar{S}$, $\bar{E}$ are the mean values of the radiant power per unit of wavelength for simulation and experiment spectra, respectively. $S_{\lambda n}$, $E_{\lambda n}$ are the values of the light power per unit of wavelength having a wavelength at the wavelength of the ordering of n for simulation and experiment spectra that are under comparison, respectively.

3. SIMULATION SPECTRUM OF YELLOW EMISSION USING PYTHON

To simulate and visualize the emission spectrum of yellow light using Python, we will utilize libraries such as NumPy and Matplotlib. The emission spectrum of yellow light can be approximated with a Gaussian distribution centered around the wavelength of yellow light, which typically ranges from about 400 nm to 800 nm in visible light. Based on the mathematical equation (1) and Python programming, the simulation spectrum of yellow emission is obtained as shown in Figure 4. The codes for the simulation spectrum of yellow emission using PYTHON are shown as follows

```
# Import the libraries used in this script
import numpy as np
import matplotlib.pyplot as plt
import matplotlib.ticker as ticker
import pandas as pd
# File path
file_path = '/kaggle/input/yellow-phosphor-
spectrum/normalized-experimental-spectrum-of-yellow-
phosphor-PLP1279.xlsx'
# Read data
data = pd.read_excel(file_path)
# Extract data from the file
lambda_experimental_values =
data['Wavelength'].values
experimental_data = data['normalized Spectral
flux'].values
#Check the data in file
try:
    lambda_experimental_values =
data['Wavelength'].dropna().values
    experimental_data = data['normalized Spectral
flux'].dropna().values
except KeyError as e:
    print(f"Error: Missing expected column in data: {e}")
# Main parameters
P = 1  # Emission band power
lambda_peak = 564  # Peak emission wavelength
lambda_mid = 564  # Wavelength dividing the emission
band into two parts
# Beta values for simulation
betas = [2.6, 2.7, 2.8, 2.9, 3.0, 3.1]
beta_step_1 = [1, 2, 3, 4, 5, 6, 7]
beta_step_05 = [1.5, 2, 2.5, 2.8, 2.85, 2.9, 3, 3.5, 4, 4.5,
5, 5.5, 6, 6.5, 7]
# Simulate and plot the phosphor emission spectrum
using a Gaussian function
plt.figure(figsize=(10, 6))
for beta in beta_step_1:
    P_lambda_left = P * np.exp(-beta *
((lambda_experimental_values[lambda_experimental_v
alues <= lambda_mid] - lambda_peak) / 52) ** 2)
    P_lambda_right = P * np.exp(-beta *
((lambda_experimental_values[lambda_experimental_v
alues > lambda_mid] - lambda_peak) / 174) ** 2)
        P_lambda = np.concatenate([P_lambda_left,
P_lambda_right])
        plt.plot(lambda_experimental_values, P_lambda,
label=f'Simulation with Beta={beta}', linewidth=5)
# Plot the phosphor emission spectrum
plt.xlabel('Wavelength (nm)', fontsize=12,
fontweight='bold')
plt.ylabel('Normalized Flux (a.u.)', fontsize=12,
fontweight='bold')
plt.title('Simulated Spectrum (Beta Step=1)',
fontsize=14, fontweight='bold')
plt.legend()
plt.grid(alpha=0.5)
plt.tight_layout()
plt.show()
```

 # There are seven beta values (in beta_step_1 list) used to plot the simulation spectrum

Running the code will generate a graph displaying a Gaussian curve peaking at 580 nm, representing the simulated emission spectrum of yellow light. Adjust the values of "lambda_peak" and "Beta" to simulate different shapes of yellow emission. Figure 4 shows the simulated emission spectrum under different values of beta constant. Different values of beta including 1, 2,3, 4, 5, 6, and 7, respectively were used in the simulation. The result indicates that the width of the emission band is proportional to the used value of beta.

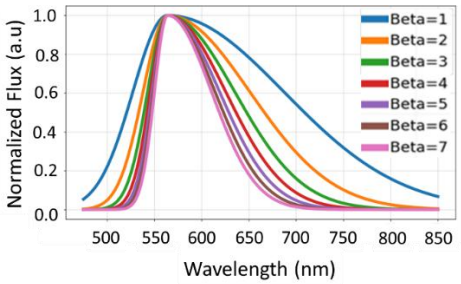

Figure 4. Simulation spectrum of yellow emission using PYTHON

Verifying the emission spectrum modeling involves comparing the simulated data generated by the model with experimental data, assessing the accuracy of the model, and ensuring it behaves as expected under various conditions. A basic approach to verify the emission spectrum modeling using Python can include these steps for verification. Firstly, it is necessary to ensure that the model accurately represents the physical phenomena of the yellow phosphor emission. This includes checking the parameters of the Gaussian functions or other mathematical representations used in the model. Secondly, Compare with Experimental Data. By obtaining experimental data for the yellow phosphor's emission spectrum with a spectrometer or from the literature. The simulated emission spectrum is plotted against the experimental data for visual comparison (Figure 5). Next, statistical analysis should be conducted using measures like R-squared, Root Mean Square Error (RMSE), or other statistical techniques to quantify the model's accuracy in predicting the experimental data. It is important to find out the value of the beta constant to have the best match between simulation and experiment results. For that purpose, different values of the beta constant were used in the simulation to simulate the corresponding emission spectrum. Then this simulated spectrum is compared to the experimental spectrum by overlapping them in the same figure. The codes for conducting these purposes in PYTHON are shown as follows.

```
# Function calculates the value of NCC
def calculate_ncc(simulated, experimental):
    S_mean = np.mean(simulated)
    E_mean = np.mean(experimental)
    numerator = np.sum((simulated - S_mean) * (experimental - E_mean))
    denominator = np.sqrt(np.sum((simulated - S_mean) ** 2) * np.sum((experimental - E_mean) ** 2))
    return (numerator / denominator) * 100  # Return NCC as a percentage
# Plot six phosphor emission spectra corresponding to different beta values
fig, axes = plt.subplots(2, 3, figsize=(14, 7)) #Create figure with two rows and three columns
x_ticks = np.linspace(450, 900, 10)
y_ticks = np.linspace(0, 1, 6)
ncc_values = [] # Initial an empty list for ncc_values
for i, beta in enumerate(betas):
    # Compute P(lambda) for two regions
    P_lambda_1 = P * np.exp(-beta * ((lambda_experimental_values[lambda_experimental_values <= lambda_mid] - lambda_peak) / 52) ** 2)
    P_lambda_2 = P * np.exp(-beta * ((lambda_experimental_values[lambda_experimental_values > lambda_mid] - lambda_peak) / 174) ** 2)
    P_lambda = np.concatenate([P_lambda_1, P_lambda_2])
    # Compute NCC for each beta (in betas list)
    ncc = calculate_ncc(P_lambda, experimental_data)
    ncc_values.append(ncc) #Append value to ncc_values
    print(f'NCC for Beta={beta}: {ncc:.3f}%')
    # Select subplot and plot
    ax = axes[i // 3, i % 3]
    ax.plot(lambda_experimental_values, experimental_data, 'k-', label='Normalized Experimental Data', linewidth=3)
    ax.plot(lambda_experimental_values, P_lambda, 'r-', label=f'Simulation with Beta={beta} (NCC={ncc:.3f}%)', linewidth=3)
    # Configure axes
    ax.set_xticks(x_ticks)
    ax.set_yticks(y_ticks)
    ax.set_xticklabels(x_ticks, rotation=45, ha='right')
ax.xaxis.set_major_formatter(ticker.StrMethodFormatter('{x:.0f}'))
    ax.set_title(f'Beta = {beta}')
    ax.set_xlabel('Wavelength (nm)')
    ax.set_ylabel('Normalized P(λ) (a.u.)')
    ax.legend()
plt.tight_layout()
plt.show()
# There are six different plots and corresponding NCC values for six different beta values. Each plot includes one simulated spectrum and one experimental spectrum.
```

The similarity between simulation and experiment spectrum curves is quantitively evaluated based on the value of NCC (e.g. the value of 100 for NCC means the complete similarity between the simulated and experimental spectrum). The code for optimizing beta value to obtain the highest NCC is shown below, the obtained result is shown in Figure 6 and Table 1.

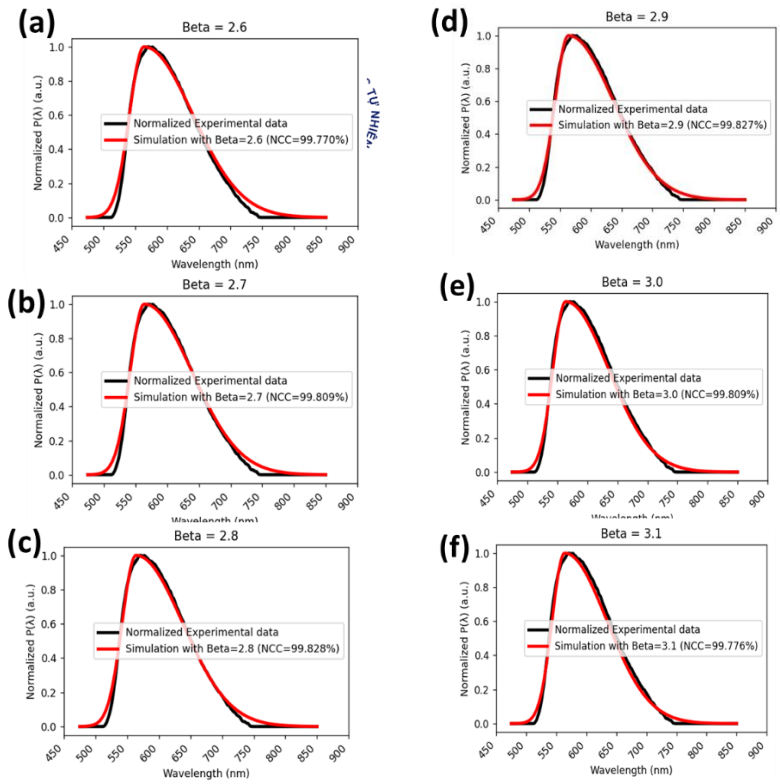

Figure 5. Comparison between simulation at different values of beta and experiment spectrum

```
# Calculate NCC value for each beta in beta_step_05 in percentage
ncc_values = []
for beta in beta_step_05:
    P_lambda_1 = P * np.exp(-beta * ((lambda_experimental_values[lambda_experimental_values <= lambda_mid] - lambda_peak) / 52) ** 2)
    P_lambda_2 = P * np.exp(-beta * ((lambda_experimental_values[lambda_experimental_values > lambda_mid] - lambda_peak) / 174) ** 2)
    P_lambda = np.concatenate([P_lambda_1, P_lambda_2])
# Compute NCC
    ncc = calculate_ncc(P_lambda, experimental_data)
    ncc_values.append(ncc)
    print(f'NCC for Beta = {beta:4.2f} : {ncc:6.3f}%')
# Find the optimal beta value
max_ncc = max(ncc_values)
optimal_beta = beta_step_05[ncc_values.index(max_ncc)]
print(f'The maximum NCC value: {max_ncc:.3f}% at Beta = {optimal_beta}')
# Plot the NCC values
plt.figure(figsize=(8, 5))
plt.plot(beta_step_05, ncc_values, 'o-', color='red', linewidth=2, markersize=8, label='NCC (%)')
plt.xlabel('Value of Beta (β)', fontsize=12, fontweight='bold')
plt.ylabel('NCC (%)', fontsize=12, fontweight='bold')
plt.title('NCC vs Beta (Step=0.5)', fontsize=14, fontweight='bold')
plt.grid(alpha=0.5)
plt.legend()
plt.tight_layout()
plt.show()
```

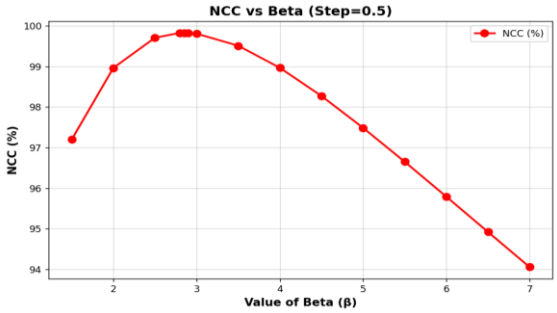

Figure 6. Optimize beta value to obtaine the highest NCC

Table 1. Changing of NCC versus Beta values

Beta values	NCC (%)	Beta values	NCC (%)
1.5	97.198	**4.0**	98.969
2.0	98.965	**4.5**	98.274
2.5	99.708	**5.0**	97.488
2.8	99.828	**5.5**	96.652
2.85	99.830	**6.0**	95.791
2.9	99.827	**6.5**	94.923
3.0	99.809	**7.0**	94.057
3.5	99.512		

4. DISCUSSIONS

The method for simulating the emission spectrum of yellow phosphor materials using Python has several applications across various fields in science and

engineering. Below are some notable applications, along with examples of how the simulation could be applied in each context:

First, in material science research, emission spectrum modeling is helpful for the characterization and development of new phosphors. Researchers can use the simulation to predict the emission spectra of new phosphor materials. By adjusting the parameters of the Gaussian functions, they can model and visualize how changes in material composition or doping concentration affect the emission spectrum. In developing a new yellow phosphor, the expected emission spectrum is simulated accurately based on theoretical calculations or prior experience with similar materials.

Second, in lighting technology, the emission modeling is applied in the design of LED and lighting Systems. LED manufacturers can apply this method to optimize phosphor blends used in white light LEDs. Simulating the light emitted from yellow phosphors helps engineers select appropriate materials to achieve desired light quality and color rendering (e.g. an engineer designing a white LED might simulate different ratios of yellow phosphors and see how the resulting emission spectrum influences the overall light output and color temperature of the LED).

Third, in display technology, emission modeling can be helpful in the improvement of display systems. Simulation of yellow phosphor emission can aid in the technological development of displays by enhancing color accuracy and brightness in screens (e.g., TVs, monitors). As an application of emission modeling in quantum dot or OLED display technology, understanding how yellow phosphors interact with other colors can improve color balance and intensity. By simulating the emission spectrum, designers can adjust the material properties of quantum dots or phosphors to achieve a better visual experience.

Simulating the emission spectrum of yellow phosphor materials using Python offers several advantages across various disciplines, including research, development, and education. Some of the key advantages are as follows:

First, in term of reducing experimental costs, by simulating the emission spectrum, researchers can explore multiple material compositions and configurations without the need to synthesize every variation. This reduces the cost associated with material procurement and laboratory experiments.

Second, in term of resources, simulations require less physical material, thereby saving on raw materials and minimizing waste.

Third, in term of time efficiency, researchers can quickly modify parameters (like Gaussian parameters) in the simulation to test different hypotheses, enabling faster iteration cycles than traditional experimental methods. It allows for the exploration of a larger number of combinations of material compositions and excitation conditions in a fraction of the time it would take to conduct actual experiments.

Fourth, visualization capabilities are powerful in graphical representation. PYTHON 's powerful libraries (like Matplotlib) allow for effective visualization of the emission spectra. Simulations can easily plot multiple emission spectra on the same graph, facilitating comparative analysis of different material configurations or conditions.

Fifth, the customization and flexibility of the emission model are effective. Researchers can build customized models that reflect specific phosphor materials or excitations relevant to their studies, enabling tailored simulations to meet unique project needs. Beyond simple Gaussian functions, more complex models can be integrated as needed, such as Lorentzian peaks, a range of curves, or empirical data fitting.

Sixth, the emission model is important for facilitation of innovation in exploring novel materials. The method encourages the exploration of new yellow phosphors and blends, pushing the boundaries of current materials science and leading to innovations in lighting technology and displays. By simulating the properties of hypothetical materials or configurations, researchers might identify promising candidates for further study, saving time in the exploratory phase of new material development.

5. CONCLUSIONS

We proposed and demonstrated a novel solution for modeling the emission of yellow phosphor YAG:Ce3+ by using the PYTHON software. The emission curve is mathematically described by using the Gaussian function. This mathematical model is simulated by programming in PYTHON. The similarities between the simulation and experimental results are compared by the normalized cross-correlation (NCC) algorithm. All NCC values show high values larger than 99.5%. This indicates the high accuracy and trustable degrees of the spectrum model. The proposed model is interesting and meaningful in spectrum modeling for luminescent material with the PYTHON software programming. In addition, the method for simulating the emission spectrum of yellow phosphor materials using Python significantly enhances research and engineering processes. It offers a blend of cost-efficiency, speed, and comprehensive insight while allowing for extensive control and customization. The combination of these advantages positions this method as a powerful tool in materials science, lighting technology, and display.

References:

Avtaeva, S. (2023). Modeling of the sulfur emission in the asymmetric pulse dielectric barrier discharge. *Plasma Chemistry and Plasma Processing*, *43*(4), 867-877.

Berends, A. C., Van De Haar, M. A., & Krames, M. R. (2020). YAG: Ce3+ phosphor: from micron-sized workhorse for general lighting to a bright future on the nanoscale. *Chemical reviews*, *120*(24), 13461-13479.

De Bonfils, J., Panczer, G., De Ligny, D., Peuget, S., Delaye, J. M., Chaussedent, S., ... & Champagnon, B. (2008). Simulation of Eu3+ luminescence spectra of borosilicate glasses by molecular dynamics calculations. *Optical Materials*, *30*(11), 1689-1693.

Chen, L., Lin, C. C., Yeh, C. W., & Liu, R. S. (2010). Light converting inorganic phosphors for white light-emitting diodes. *Materials*, *3*(3), 2172-2195.

Fan, J., Chen, W., Yuan, W., Fan, X., & Zhang, G. (2020). Dynamic prediction of optical and chromatic performances for a light-emitting diode array based on a thermal-electrical-spectral model. *Optics express*, *28*(9), 13921-13937.

Houser, K. W., Wei, M., David, A., Krames, M. R., & Shen, X. S. (2013). Review of measures for light-source color rendition and considerations for a two-measure system for characterizing color rendition. *Optics Express*, *21*(8), 10393-10411.

Islam, M. S., Dangol, R., Hyvärinen, M., Bhusal, P., Puolakka, M., & Halonen, L. (2013). User preferences for LED lighting in terms of light spectrum. *Lighting Research & Technology*, *45*(6), 641-665.

Jacquemin, D., Perpete, E. A., Ciofini, I., & Adamo, C. (2009). Accurate simulation of optical properties in dyes. *Accounts of chemical Research*, *42*(2), 326-334.

Jang, H. S., & Jeon, D. Y. (2007). Yellow-emitting Sr3SiO5: Ce3+, Li+ phosphor for white-light-emitting diodes and yellow-light-emitting diodes. *Applied Physics Letters*, *90*(4), 041906. DOI: 10.1063/1.2432947

Koryukina, E. V., & Koryukin, V. I. (2017). Simulation of emission spectra of light sources based on an inductive discharge. *Russian Physics Journal*, *59*(12), 1996-2003.

Lokesh, J., Kini, S. G., Mahesha, M. G., & Padmasali, A. N. (2025). Color-based lifetime estimation of LEDs using spectral power distribution prediction through analytical and machine learning models. *IEEE Access*, *13*, 61665-61674.

Nagi, Ł., Kozioł, M., & Zygarlicki, J. (2020). Comparative analysis of optical radiation emitted by electric arc generated at AC and DC voltage. *Energies*, *13*(19), 5137.

Narendran, N., & Gu, Y. (2005). Life of LED-based white light sources. *Journal of display technology*, *1*(1), 167.

Nguyen, Q. K. (2025). Precision optical model of asymmetry emission band of red light emitting diodes and its application in improving the color rendering index of white LEDs. *Results in Optics*, *21*, 100825.

Parauha, Y. R., & Dhoble, S. J. (2021). Photoluminescence and electron-vibrational interaction in 5d state of Eu2+ ion in Ca3Al2O6 down-conversion phosphor. *Optics & Laser Technology*, *142*, 107191.

Pasquini, C., Cortez, J., Silva, L. M., & Gonzaga, F. B. (2007). Laser induced breakdown spectroscopy. *Journal of the Brazilian Chemical Society*, *18*(3), 463-512.

Peres, T. (1999). *Applications of a three-body potential model and crystal field calculations to the simulation of EU^{3+}, Er^{3+} and Yb^{3+} environments in PbO. SiO_2 glass* (Doctoral dissertation, Concordia University).

Satpute, N. S., Mehare, C. M., Tiwari, A., Swart, H. C., & Dhoble, S. J. (2022). Synthesis and luminescence characterization of downconversion and downshifting phosphor for efficiency enhancement of solar cells: Perspectives and challenges. *ACS Applied Electronic Materials*, *4*(7), 3354-3391.

Schubert, E. F., & Kim, J. K. (2005). Solid-state light sources getting smart. *Science*, *308*(5726), 1274-1278.

Srinath, K. R. (2017). Python–the fastest growing programming language. *International Research Journal of Engineering and Technology*, *4*(12), 354-357.

Xie, R. J., Li, Y. Q., Hirosaki, N., & Yamamoto, H. (2011). *Nitride phosphors and solid-state lighting*. Boca Raton, FL: Taylor & Francis.

Zhu, D., & Humphreys, C. J. (2016). Solid-state lighting based on light emitting diode technology. In *Optics in our time* (pp. 87-118). Cham: Springer International Publishing.

Huynh-Tuan-Anh Nguyen
Faculty of Physics and Engineering Physics, VNUHCM-University of Science, Hochiminh City, Vietnam, Vietnam National University Hochiminh City, Hochiminh City, Vietnam.
nhtanh@hcmus.edu.vn
ORCID: 0009-0001-3705-2790

Nguyen-Quang-Huy Tran
Faculty of Physics and Engineering Physics, VNUHCM-University of Science, Hochiminh City, Vietnam, Vietnam National University Hochiminh City, Hochiminh City, Vietnam.
21130046@student.hcmus.edu.vn
ORCID: 0009-0000-0659-1700

Anh-Thu Nguyen
Faculty of Physics and Engineering Physics, VNUHCM-University of Science, Hochiminh City, Vietnam, Vietnam National University Hochiminh City, Hochiminh City, Vietnam.
nathu@hcmus.edu.vn
ORCID: 0009-0004-7518-8658

Nguyet-Thuan Phan

Quang-Khoi Nguyen

1Faculty of Physics and Engineering Physics, VNUHCM-University of Science, Hochiminh City, Vietnam,
Vietnam National University Hochiminh City, Hochiminh City, Vietnam.
pnthuan@hcmus.edu.vn
ORCID: 0000-0001-9478-734X

Faculty of Physics and Engineering Physics, VNUHCM-University of Science, Hochiminh City, Vietnam,
Vietnam National University Hochiminh City, Hochiminh City, Vietnam.
nqkhoi@hcmus.edu.vn
ORCID: 0000-0002-6456-365X