



**Politecnico
di Torino**

POLITECNICO DI TORINO

Master's degree in Electronics Engineering for
Industrial Applications

Master's degree thesis
Academic year 2023-2024

**Fluorescence Spectroscopy
empowered by Data-Driven
algorithms.**

**A fast and cost-effective
approach to Environmental
Monitoring and Analysis.**

Advisor: Prof. Guido Perrone

Candidate: Lorenzo Riggi

Tutor: Chiara Bellezza Prinsi

Abstract.....	6
Introduction.....	8
Chapter 1: Fluorescence	11
What is Fluorescence	11
What is absorption and how it works.....	18
Principles and characteristics of fluorescence emission.....	20
Stokes shift	20
Mirror image rule and independence of the emission wavelength from the excitation wavelength.....	22
Franck Condon principle	23
Fluorescence lifetime and quantum yield.....	24
Some factors that affect fluorescence	26
Quenching.....	26
Reabsorption.....	27
Temperature	29
Solvent effects.....	30
pH	30
Chapter 2: Laboratory Measurements	35
Lab setup.....	36
Avantes spectrometer	38
How does the lab setup work	40
Experiment 1: Chlorophyll.....	40

Goals	40
What is Chlorophyll.....	41
Preparation of the solutions	44
Setup	44
Experimental results	45
Experiment 2: Curcumin.....	48
Goals	48
What is Curcumin	49
Preparation of the solutions	50
Setup	51
Experimental results	52
Experiment 3: Fluorescein	55
Goals	55
What is Fluorescein	55
Preparation of the solutions	57
Setup	59
Experimental results	59
Experiment 4: All together: fluorescein, chlorophyll and curcumin .	65
Goals	65
Preparation of the solutions	65
Setup	66
Experimental results	67
Experiment 5: pH experiments	69
Goals	69
Preparation of the solutions	69
Setup	70

Experimental results for fluorescein	71
Experimental results for curcumin	76
Experiment 6: Tyrosine and Tryptophan from dairy milk solution..	79
Goals	79
What are Tyrosine and Tryptophan	80
Preparation of the solutions	82
Setup	82
Experimental results	85
Final conclusions on all the experimental results obtained.....	87
Chapter 3: FluoDetect	90
Introduction.....	90
Creating a database	93
Adding a spectrum to a DB	95
Searching a spectrum	100
Comparing spectrums	103
Compatible file formats for inserting spectrums	110
Software classes and methods	111
Introduction: Flask, databases, and most important files/folders ..	111
DB Class	115
FluoRec Class	118
PeaksSpot Module	120
AddSpectra Class	121
CompareSpectra Class	124
App Module	129

Chapter 4: Experimental results with FluoDetect and future developments	133
Chapter 5: Possible implementation of an EWS on the field.....	139
Chapter 6: Conclusions.....	142
Bibliography	144

Abstract

In recent years the concern for accidental or deliberate contamination of water and soil has increased, especially in parts of the world where scarcity of drinking water, arable land, and geopolitical power plays are in place. Because of that nowadays there is a need for technologies that can perform an online monitoring of the contamination as cheaply and fast as possible. These kinds of systems are called **EWS** (**E**arly **W**arning **S**ystems) and serve the purpose of warning the user/public authority in time of a possible contamination taking place.

Fluorescence spectroscopy is a possible solution for cheap and fast online monitoring, its main advantages are on time and cost: Spectrofluorometers give a result (in the form of an emission spectrum) in a scale from milliseconds to seconds, moreover, their cost does not go far beyond the thousands of euros.

This kind of instrument can be connected to a PC or embedded system, which can acquire the spectrum in the form of a file, analyze the data, determine if a contaminant is present, and inform the authorities.

Fluorescence spectroscopy allows the detection of a contaminant if the contaminant shows a fluorescent behavior when illuminated by an excitation source (such as an LED or Laser). It is a known and studied fact that the most hazardous bacteria (for human beings and animals) contain proteins or other compounds that show visible fluorescence on the spectrofluorometer, and this is one of the key reasons why fluorescence spectroscopy is gaining popularity as an EWS system.

When a compound exhibits fluorescent behaviour, it emits light on a different wavelength (colour) than the excitation source's wavelength. The emission wavelength of the compound indicates the nature of the compound itself, along with the shape of the emitted light's spectrum, which constitutes a unique fingerprint for the compound to be detected.

This thesis work is a study of fluorescence, meaning every possible environmental or compound's chemical factor that can influence the emitted light spectrum's shape and wavelength.

In this work, the fluorescence process and its applications in detecting pollutants/bacteria in water have been studied both in literature and in lab experiments carried on at the “PhotoNext” lab in PoliTo premises, using a lab setup made of a spectrofluorometer as optical instrument, PC to acquire the spectrums trough the spectrofluorometer, cuvette and cuvette holders, and different LEDs to excite the different fluorophores studied. The collected fluorophores’ emission spectrums have been analysed and studied with MATLAB.

The goal behind the said study has been to create a software application that can detect the presence of possibly dangerous compounds via spectrum analysis algorithms empowered by data collected by the user of the application or other scientists in the field.

The goal which the software has achieved is that of providing a platform for fast detection of fluorescent compounds customisable to the need of the application the user intends to use it for. Moreover, the software application is written in Python and intended to run on a web server, to be compatible with whatever hardware the user is using it for in his application.

The software which has been created on purpose for this thesis work is presented along with a possible implementation of it in an EWS system.

Keywords: *Contaminant, Water contamination, Soil contamination, Bacteria, EWS, Fluorescence, Fluorescence spectroscopy, Spectrum analysis, Data, Algorithms, Python.*

Introduction

Water/soil contamination and why fluorescence spectroscopy

Water and soil contamination in European countries is a matter of serious concern. Europe faces numerous challenges regarding its water resources, including industrialization, agricultural intensification, and population changes. These factors can affect both the quality and quantity of water available for various uses. Ensuring universal access to safe drinking water and sanitation is crucial for public health and national prosperity. However, unsustainable water use and management practices could jeopardize this fundamental principle. Over 2.5 million cases of gastrointestinal and other waterborne diseases were reported over 11 years in 18 European countries, with 2% linked to drinking water [1]. Agriculture contributes to water pollution through the application of fertilizers, including inorganic nitrate fertilizers. The intensification of agriculture has led to increased use of these fertilizers, which can result in the leaching of nitrates and nitrites into water sources. Elevated levels of nitrites in drinking water can pose health risks, particularly to vulnerable populations such as infants and pregnant women. Nitrites can interfere with the blood's ability to carry oxygen, leading to methemoglobinemia (blue baby syndrome) in infants.

Also, contaminated water from healthcare settings poses a serious risk to human health, and outbreaks of *Legionella*, a bacteria that thrives in warm, stagnant water, commonly found in hospital plumbing systems pose a serious risk for immunocompromised patients in hospitals. Not only *Legionella* but also Various Gram-negative bacteria, including *Pseudomonas* and *Acinetobacter*, can contaminate hospital water sources, leading to sporadic infections and outbreaks [2].

Most of these pollutants/hazardous bacteria have something in common: they exhibit fluorescence when illuminated by an excitation source such as an LED or laser. When illuminated they emit a spectrum of light, called “emission spectrum” which for every compound has its emission

peak(s) wavelength(s), and shape. This constitutes a fingerprint unique to every fluorescent compound and allows the detection of a pollutant via its emission spectrum characteristics.

Fluorescence spectroscopy has gained popularity in recent years due to its ease of use, fastness in obtaining results, and low costs. Fluorescence spectroscopy has been already used successfully in the detection of nitrite and nitrate in environmental samples [3], in detecting bacteria [4], and in analyzing wastewater composition [5].

The author of this work deemed interesting the fluorescence process and its applications in detecting pollutants/bacteria in water studied the topic both in literature and in lab experiments carried on at the “PhotoNext” lab in PoliTo premises. Toward the end of this thesis experience, a software has been developed, called “FluoDetect” capable of detecting fluorescent compounds on the fly via a data-driven algorithm. Moreover, the software accessible to everyone who wants to use it, edit the code, or simply wants to use it as a groundwork for another project in this field.

The work is presented as:

- An overview of what fluorescence is and how it works and is influenced by chemical and external parameters is presented in **Chapter 1**.
- In **Chapter 2** a study of different fluorescent compounds done in the “PhotoNext” lab in PoliTO premises is presented to provide laboratory evidence of the theory presented in Chapter 1.
- In **Chapter 3** a software created on purpose for this thesis work, called “FluoDetect” is presented. The software serves the purpose of detecting fluorescent compounds on the fly via a data-driven algorithm. **Github link:** <https://github.com/LorenzoRiggi/fluoProject>
- In **Chapter 4** some experimental results obtained with FluoDetect software are presented.
- In **Chapter 5** a prototype of a real implementation of an EWS (Early Warning System) with an embedded system that uses FluoDetect software is presented.

- **Chapter 6** presents the final conclusions on the thesis' work.

Chapter 1: Fluorescence

What is Fluorescence

Fluorescence is a phenomenon where a substance absorbs light at one wavelength and emits light at a longer wavelength. When certain molecules, called **fluorophores**, absorb photons of light, they become excited and then release energy in the form of light. This emitted light typically has a longer wavelength than the absorbed light [6]. Fluorescence is part of a broader phenomenon called **luminescence**.

Luminescence refers to the emission of light from various substances, which happens when they are in electronically excited states. It can be categorized into two main types: **fluorescence** and **phosphorescence**, depending on the specific characteristics of the excited state. **Fluorescence occurs in singlet states where the excited electron is paired with its counterpart in the ground state, allowing for a rapid return to the ground state through photon emission** (Fig. 1.3). The emission rates for fluorescence are typically around 10^8 per second, resulting in lifetimes of approximately 10 nanoseconds. This short timescale necessitates sophisticated optics and electronics for accurate time-resolved measurements. Despite the technical challenges, time-resolved fluorescence is widely utilized due to the wealth of information it provides compared to steady-state measurements, and advancements in technology have facilitated its implementation, even in microscopic settings.

Phosphorescence, on the other hand, involves the emission of light from triplet excited states where the electron's spin orientation matches that of the ground-state electron (Fig. 1.3). Transitions to the ground state are forbidden, leading to slower emission rates and longer lifetimes, typically lasting milliseconds to seconds. Some materials exhibit even longer phosphorescence lifetimes, as seen in "glow-in-the-dark" toys, where the emission persists for several minutes after exposure to light. Phosphorescence is generally not observed in fluid solutions at room temperature due to competing deactivation processes such as non-radiative decay and quenching. However, certain complexes, like

transition metal–ligand complexes (MLCs), can exhibit mixed singlet–triplet states with intermediate lifetimes ranging from hundreds of nanoseconds to several microseconds.

Fluorescence predominantly occurs in aromatic molecules, with quinine being a common example found in tonic water (Fig 1.1). When tonic water is exposed to sunlight, a faint blue glow is often visible at the surface.

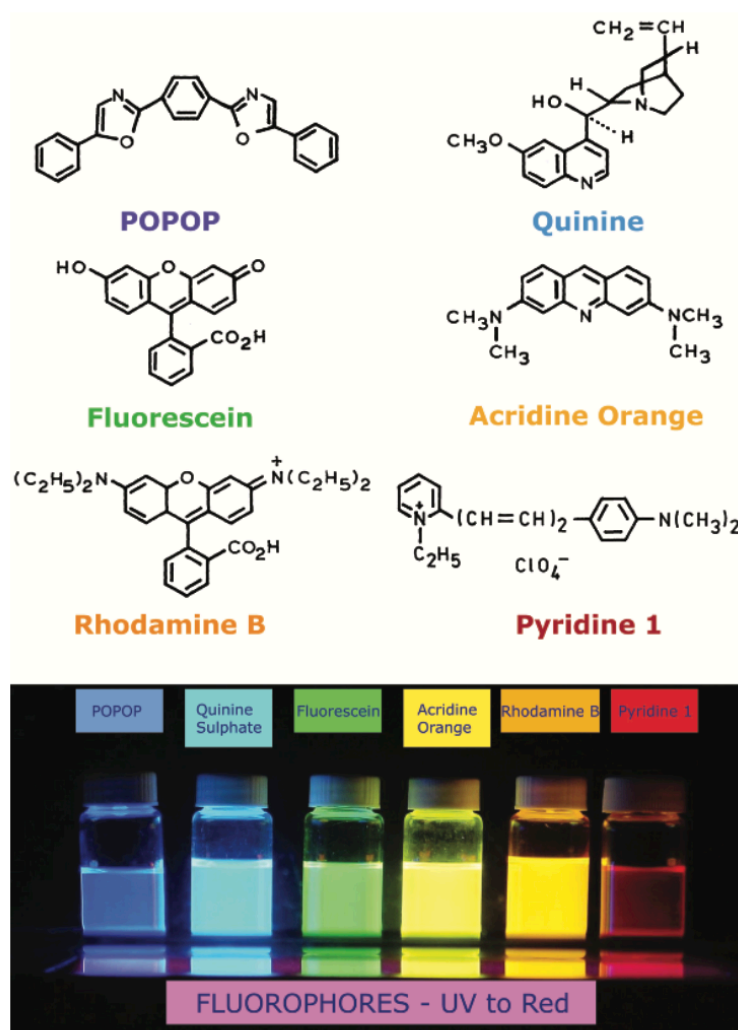


Fig 1.1: Chemical structures of some known fluorophores [6].

Fluorescence spectral data are commonly represented through emission spectra. An emission spectrum of fluorescence illustrates the fluorescence intensity plotted against wavelength (measured in

nanometers) or wavenumber (measured in cm^{-1}). Fig. 1.2 displays typical fluorescence emission spectra. **These spectra exhibit considerable variation, influenced by the chemical makeup of the fluorophore and the properties of the solvent in which it is dissolved.** For certain compounds like perylene, the spectra exhibit notable structural features attributed to the distinct vibrational energy levels of both ground and excited states. Conversely, compounds like quinine display spectra that lack discernible vibrational structures.

The “**Stokes-shift**” displayed in Fig. 1.2 is the distance in wavelength between the peak of emission and that of absorption. Its value depends on the vibrational structure of the molecule examined, it will be discussed later on in this work.

The transitions between light absorption and emission are commonly depicted using **Jablonski diagrams** (Fig 1.3). These diagrams serve as a foundational tool for explaining the processes of light absorption and emission, showcasing various molecular events that take place in excited states.

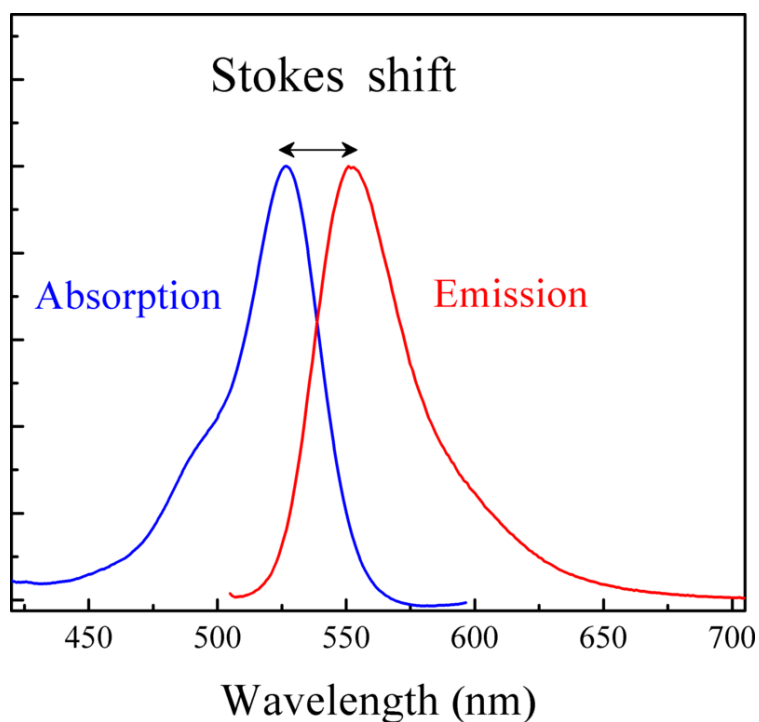


Fig 1.2: Absorption + emission spectrum of Rhodamine 6G, a fluorescent dye. The fluorophore finds widespread application in various fields including highlighter pens, dye lasers, and automotive leak detection. Its absorption profile mirrors its emission profile [27].

Fig. 1.3 illustrates a standard Jablonski diagram, where the singlet ground, first, and second electronic states are denoted by S_0 , S_1 , and S_2 , respectively. Each of these electronic energy levels accommodates various vibrational energy levels, indicated by 0, 1, 2, and so forth. In this depiction, several interactions such as quenching, energy transfer, and solvent interactions have been omitted for discussion later in this work. The transitions between these states are represented by vertical lines, emphasizing the instantaneous nature of light absorption. These transitions occur within approximately 10^{-15} seconds, a timeframe too brief for significant displacement of nuclei. This principle is known as the **Franck-Condon principle**.

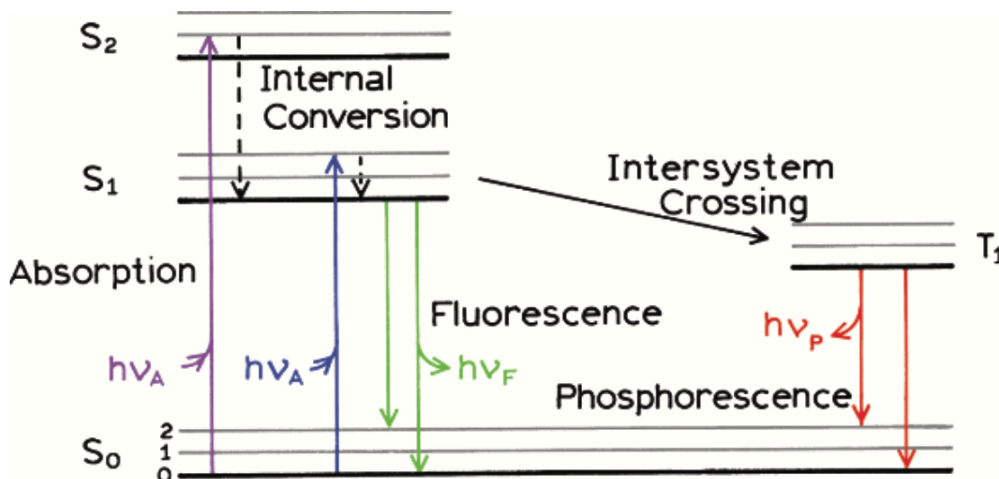


Fig 1.3: Example of Jablonski diagram for the fluorescence and phosphorescence processes [6].

The emission spectrum of perylene, as depicted in Fig. 1.4, demonstrates the energy spacing between different vibrational energy levels. Each emission maximum, corresponding to different vibrational energy levels, is approximately 1500 cm^{-1} apart. At room temperature, thermal energy is insufficient to significantly populate the excited vibrational states. Consequently, absorption and emission predominantly occur from molecules in the lowest vibrational energy state. The considerable energy difference between the S_0 and S_1 excited states precludes the thermal population of S_1 , necessitating the use of light rather than heat to induce fluorescence.

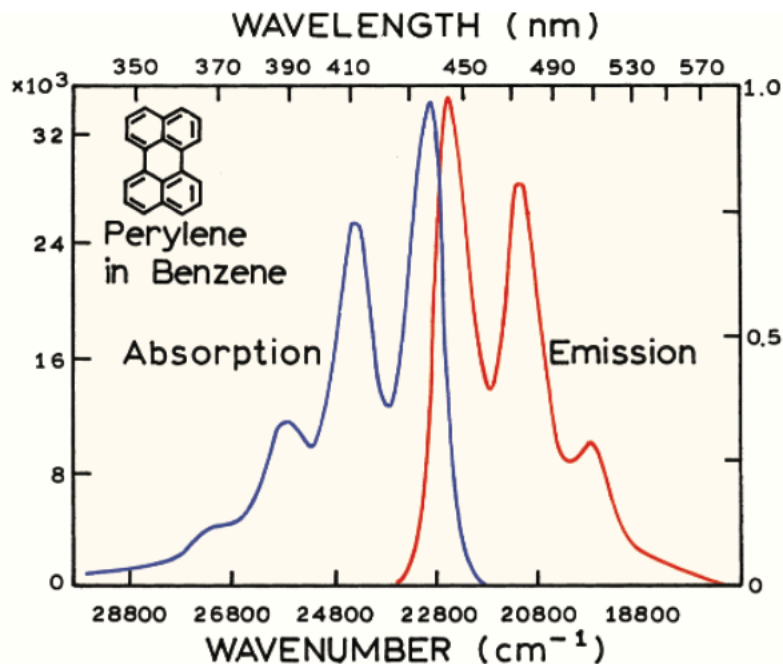


Fig 1.4: Emission spectrum of Perylene [6].

Upon light absorption, various processes typically ensue. A fluorophore is usually excited to a higher vibrational level of either S_1 or S_2 . With a few exceptions, molecules in condensed phases promptly relax to the lowest vibrational level of S_1 , a phenomenon known as internal conversion, which generally completes within approximately 10^{-12} seconds or less. Since fluorescence lifetimes are typically around 10^{-8} seconds, internal conversion is typically finished prior to emission. Thus, fluorescence emission generally arises from a thermally equilibrated excited state, specifically the lowest energy vibrational state of S_1 .

Return to the ground state typically occurs at a higher excited vibrational ground state level, which subsequently reaches thermal equilibrium swiftly (within 10^{-12} seconds), as shown in Fig. 1.5. The return to an excited vibrational state at the level of the S_0 state explains the vibrational structure observed in the emission spectrum of perylene. An intriguing consequence of emission to higher vibrational ground states is that the emission spectrum usually mirrors the absorption spectrum of the $S_0 \rightarrow S_1$ transition. This similarity arises because electronic excitation minimally alters the nuclear geometry, resulting in the similar spacing of vibrational energy levels in the excited and ground states. Consequently, the vibrational structures observed in the absorption and emission spectra are akin.

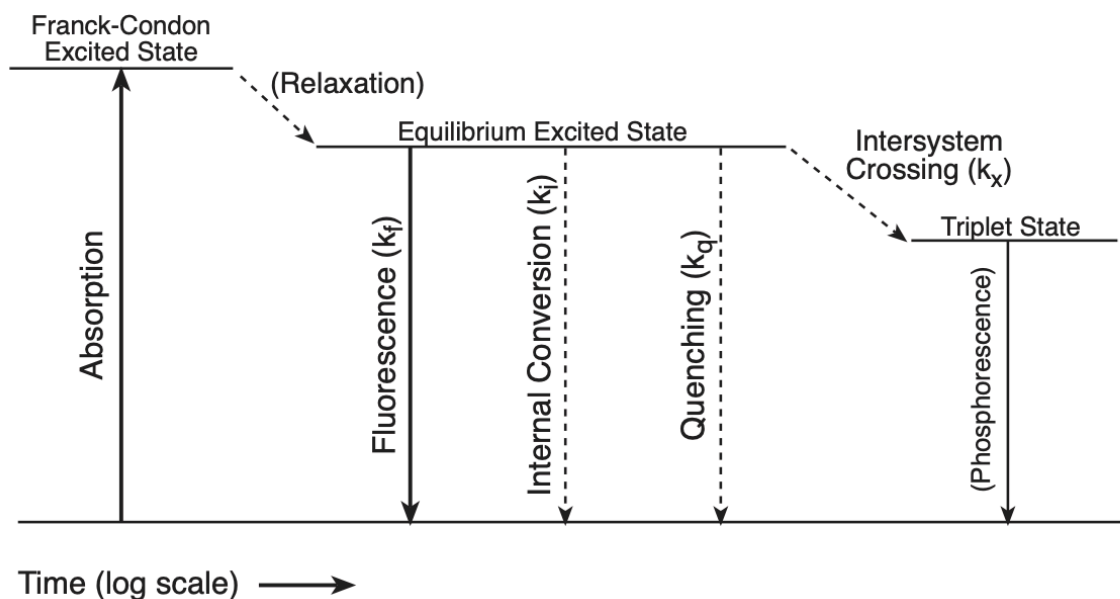


Table 1: Rates of Absorption and Emission comparison.

Process	Transition	Timescale (sec)
Light Absorption (Excitation)	$S_0 \rightarrow S_n$	ca. 10^{-15} (instantaneous)
Internal Conversion	$S_n \rightarrow S_1$	10^{-14} to 10^{-11}
Vibrational Relaxation	$S_n^* \rightarrow S_n$	10^{-12} to 10^{-10}
Intersystem Crossing	$S_1 \rightarrow T_1$	10^{-11} to 10^{-6}
Fluorescence	$S_1 \rightarrow S_0$	10^{-9} to 10^{-6}
Phosphorescence	$T_1 \rightarrow S_0$	10^{-3} to 100
Non-Radiative Decay	$S_1 \rightarrow S_0$ $T_1 \rightarrow S_0$	10^{-7} to 10^{-5} 10^{-3} to 100

Fig 1.5: summary of luminescence processes with the related lifetimes of each process [23].

Molecules in the S_1 state may also undergo a spin conversion to the first triplet state, T_1 . Emission from T_1 , termed phosphorescence, typically shifts to longer wavelengths (lower energy) compared to fluorescence. The conversion of S_1 to T_1 is known as intersystem crossing. The transition from T_1 to the singlet ground state is prohibited, leading to rate constants for triplet emission several orders of magnitude smaller than those for fluorescence. Molecules containing heavy atoms like bromine and iodine often exhibit phosphorescence, facilitated by these heavy atoms that enhance intersystem crossing and consequently phosphorescence quantum yields.

What is absorption and how it works

“In physics, absorption of electromagnetic radiation is how matter (typically electrons bound in atoms) takes up a photon's energy — and so transforms electromagnetic energy into internal energy of the absorber” [7]. To understand how the fluorescence process works is of utmost importance to understand absorption, the process that absorbs light which in turn brings the molecule to an excited state that when decaying can produce fluorophores if the process is radiative.

Each molecule contains the energy orbitals depicted in Fig. 1.6.1 and 1.6.2 [8]:

- **Sigma bonds:** these contain the highest electron density
- **Pi bonds.**
- **N bonds** (nonbonding); Lone pairs of electrons (not bounded), these are the most energetic states.
- The respective anti-bonding states.

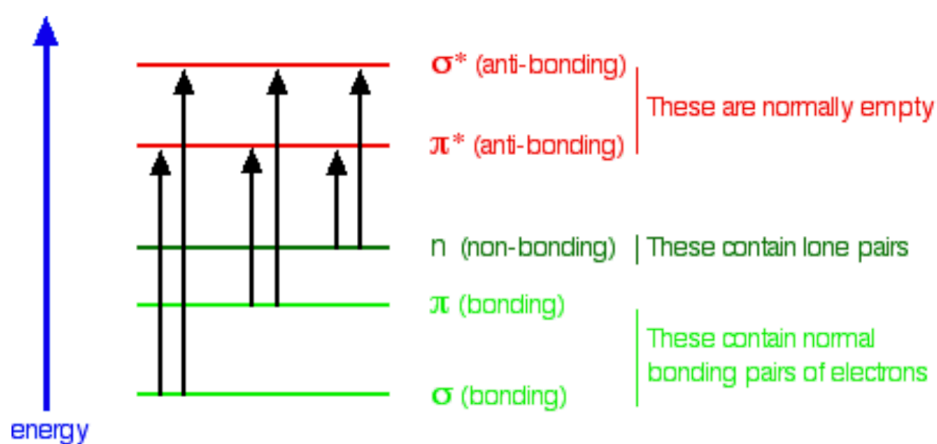


Fig 1.6.1: Molecules general electronic structure [23].

It requires energy for an electron to be promoted to the highest (and mostly void of electrons) anti-bonding electronic states. The electrons can obtain the energy to make the jump through energy from a light source (e.g. compound illuminated by UV light).

Not all the ground electronic states partake in the jump:

- It requires way more energy to break an electron from a sigma bond.
- **Electrons that partake in this process are mostly in Pi and n orbitals** because they can make the jump more easily (less energy gap from anti-bonding orbitals).

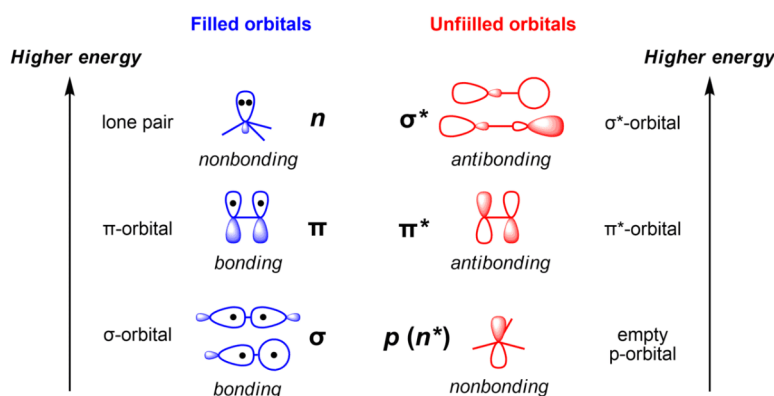


Fig 1.6.2: 3D scheme of bonding and anti-bonding orbitals [23].

From this analysis follows that absorption is augmented in compounds with more densely populated orbitals.

An example is provided in Fig. 1.6.3:

CH ₂ =CH ₂	CH ₂ =CH-CH=CH ₂	CH ₂ =CH-CH=CH-CH=CH ₂
ethene	buta-1,3-diene	hexa-1,3,5-triene
molecule	wavelength of maximum absorption (nm)	
ethene	171	
buta-1,3-diene	217	
hexa-1,3,5-triene	258	

Fig 1.6.3: example on how much energy absorption requires for different compounds [23].

Ethene has a single double bound (sigma), the wavelength at maximum absorption is 171 nm, lower wavelength -> Higher frequency -> Higher Energy. This means that it takes a lot of energy to absorb an electron.

In buta-1,3-diene there is a conjugated double bound, meaning there is a delocalization of the bound over the whole molecule. The wavelength of

maximum absorption is higher, meaning it takes less energy to promote an electron to its anti-bond orbital.

In light of this:

- The maximum absorption is moving to longer wavelengths as the amount of delocalization increases.
- Therefore maximum absorption is moving to shorter frequencies as the amount of delocalization increases.
- Therefore absorption needs less energy as the amount of delocalization increases.
- Therefore there must be less energy gap between the bonding and anti-bonding orbitals as the amount of delocalization increases.

Principles and characteristics of fluorescence emission

Stokes shift, mirror image rule, Franck Condon principle and quantum yield

At this point it is useful to discuss more in detail some general characteristics of fluorescence emission, useful for the subsequent analysis in the lab of some common fluorophores in chapter 2.

Stokes shift

The Stokes shift (Fig. 1.2 shows an example) refers to the difference (measured in energy, wavenumber, or frequency units) between the peaks of the absorption and emission spectra (such as fluorescence and Raman) corresponding to the identical electronic transition.

When a system, whether it's a molecule or an atom, absorbs a photon, it acquires energy and transitions to an excited state. One mechanism for the system to return to its ground state is by releasing a photon,

consequently dissipating its energy. Another possibility involves the dissipation of energy as translational mode energy through vibrational-translational or electronic-translational collisions with other atoms or molecules. The discrepancy in energy between the absorbed and emitted photons defines the Stokes shift.

The Stokes shift predominantly arises from two factors: vibrational relaxation or dissipation and solvent reorganization.

A fluorophore, acting as a dipole, is encompassed by solvent molecules. Upon entering an excited state, the dipole moment of the fluorophore changes, yet the surrounding solvent molecules cannot promptly readjust. **It's only after vibrational relaxation that their dipole moments realign.**

The solvent reorganization will be discussed more in detail when **solvent effects** will be presented. **For now, it is sufficient to say that when an atom/molecule is in an excited state, the processes described above contribute to the falling of the molecule to a lesser excited state (with less energy). The emitted photon has the same energy as the difference between this excited state and the first available ground state (the most energetic one).**

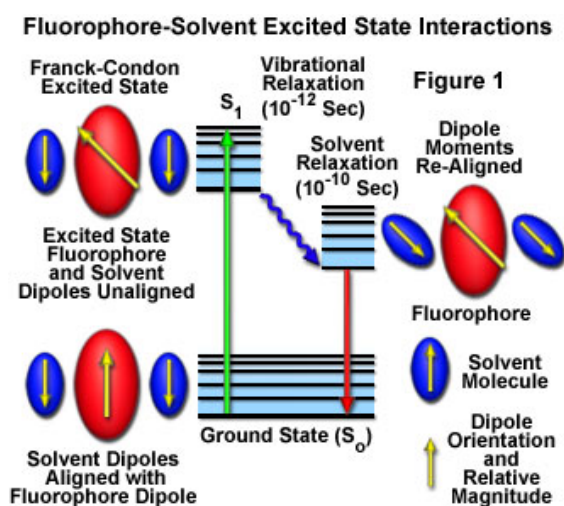


Fig 1.7: Fluorophore-solvent excited state interactions [28].

Mirror image rule and independence of the emission wavelength from the excitation wavelength

Another common characteristic of fluorescence is that the fluorescence emission spectrum typically remains independent regardless of the excitation wavelength used.

After being excited to higher electronic and vibrational states, the surplus energy dissipates rapidly, leaving the fluorophore in the lowest vibrational state of S_1 . This relaxation typically occurs within about 10–12 seconds, likely due to significant overlap among multiple states with nearly identical energies. As a result of this swift relaxation process, emission spectra generally remain unaffected by the excitation wavelength. However, there are exceptions, such as fluorophores existing in two ionization states, each displaying distinct absorption and emission spectra. Additionally, some molecules may emit from the S_2 level, although such emissions are rare and typically not observed in biological molecules.

Typically, the emission spectrum mirrors the absorption spectrum during the transition from the ground state (S_0) to the first excited state (S_1). This correspondence in transitions between the ground state and the first excited state underscores the similarity in vibrational levels between these two states.

Deviation from the mirror image pattern occurs when transitioning to excited states beyond the immediate next state, such as from S_0 to S_2 or S_3 . The absorption of high-energy photons prompts electrons to transition from the ground state (S_0) to higher electronic and vibrational levels like S_2 and S_3 . Subsequently, the excess energy in these higher energy states dissipates rapidly, leading to relaxation to the lowest vibrational level within that specific excited state, followed by a transition to the first excited state (S_1), and ultimately returning to the ground state (S_0). As a result, the emission spectra differ from the excitation spectra, and the higher excitation states do not adhere to the mirror image principle.

Franck Condon principle

The Franck–Condon principle, attributed to James Franck and Edward Condon, is a concept in spectroscopy and quantum chemistry explaining the intensity of vibronic transitions. These transitions entail simultaneous alterations in the electronic and vibrational energy levels of a molecule upon absorbing or emitting a photon with the appropriate energy. According to this principle, during an electronic transition, a transition between two vibrational energy levels is more likely to happen when the overlap between their vibrational wave functions is more pronounced.

- i) The nucleus has a much greater mass than the electrons so that during the electronic transition the nucleus remains stationary but the electronic configuration changes.
- ii) Therefore, the inter-nuclear separation remains the same after absorption.
- iii) The most intense electronic transition occurs between the vibration states that are arranged vertically in the different electronic states (Fig. 1.8).
- iv) Classically the Frank-Condon principle says that the transition of electrons from the ground to the excited state occurs without any change in the position of the nucleus. These promote the ground state electrons to the relative excited states through vertical transitions and are plotted as vertical lines in Fig. 1.8.

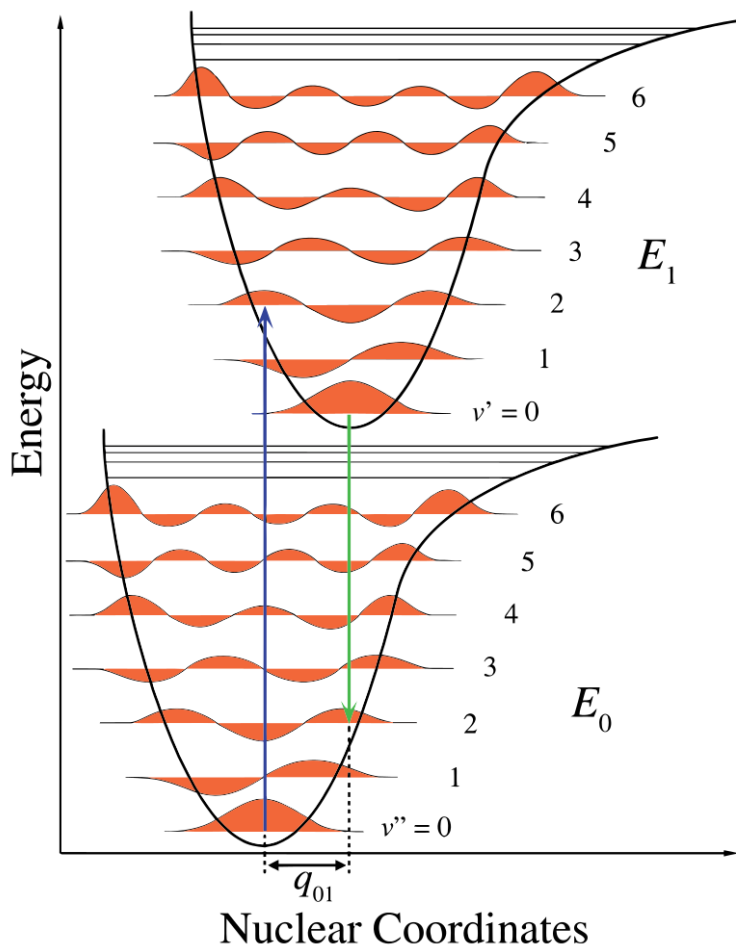


Fig. 1.8: transitions between the different electronic states. Since electronic transitions are very fast compared with nuclear motions, vibrational levels are favored when they correspond to a minimal change in the nuclear coordinates. The potential wells are shown favoring transitions between $v=0$ and $v'=2$ [26].

Fluorescence lifetime and quantum yield

Fluorescence lifetimes and quantum yields, along with spectral distribution, stand out as crucial characteristics of a fluorophore. **The fluorescence quantum yield (Φ_F) represents the ratio of emitted photons (per unit time and volume) relative to absorbed photons ($\Phi_F \leq 1$).** Therefore, Φ_F is the ratio of the intrinsic radiative

rate constant (k_{rad}) to the sum of the rate constants for all processes contributing to the depopulation of the first excited singlet state. These processes include internal conversion (k_{IC}), intersystem crossing (k_{ISC}), and various types of quenching or photophysical/photochemical reactions (k_{react}). Consequently, the fluorescence quantum yield approaches unity only when the sum of radiationless decay rates significantly surpasses the rate of radiative decay, that is, $k_{\text{nr}} \gg k_{\text{rad}}$.

$$\Phi_{\text{F}} = \frac{k_{\text{rad}}}{k_{\text{rad}} + k_{\text{IC}} + k_{\text{ISC}} + k_{\text{react}}} = \frac{k_{\text{rad}}}{k_{\text{rad}} + k_{\text{nr}}} \quad [1]$$

It's important to note that due to numerous nonradiative processes, the lifetime of the first singlet excited state (referred to as the observed fluorescence lifetime τ_{F}) is generally shorter than the radiative lifetime (the reciprocal of the radiative rate). Conversely, the time-dependent fluorescence intensity is directly proportional to the population of the excited state. The kinetic equation governing the decay of the singlet excited-state population S_1 after the excitation ceases is formed by summing up all parallel de-excitation pathways.

$$\frac{d[S_1]}{dt} = -(k_{\text{rad}} + k_{\text{nr}}) [S_1] \quad [2]$$

$$[S_1](t) = [S_1](0)e^{-t/\tau_{\text{F}}} \quad [3]$$

The solution to this equation can be expressed as $S_1 = S_1 * e^{-(t/\tau_{\text{F}})}$, where S_1 represents the excited-state population at time zero (defined at any moment after the end of excitation), and τ_{F} is the observed fluorescence lifetime.

$$\tau_{\text{F}} = (k_{\text{rad}} + k_{\text{IC}} + k_{\text{ISC}} + k_{\text{react}})^{-1} = (k_{\text{rad}} + k_{\text{nr}})^{-1} \quad [4]$$

In practical terms, when using this equation to describe the observed decay of fluorescence intensity, the time origin is typically set at the end of excitation, namely, at the peak of the measured fluorescence signal.

By combining equations [1] and [4], we derive the relationship:

$$\Phi_F = k_{\text{rad}} / (k_{\text{IC}} + k_{\text{ISC}} + k_{\text{react}} + k_{\text{nr}})$$

Fluorescence quantum yields and lifetimes, as well as fluorescence emission spectra, typically remain unaffected by the excitation wavelength. In an ideal scenario, the fluorescence excitation spectrum of a molecule (defined as the fluorescence intensity measured at a fixed wavelength as a function of excitation wavelength) perfectly overlaps with the absorption spectrum. Any discrepancy between absorption and excitation spectra indicates excitation-wavelength-dependent photoinduced reactions [9].

Some factors that affect fluorescence

Quenching, reabsorption, temperature, solvent effects, and pH

Fluorescence is affected by different phenomena that quench its intensity in a range of the emission wavelengths (Quenching phenomena) or change the shape of the emission spectrum (reabsorption, solvent effects, and pH). Here it will be presented a short overview of the factors that affect fluorescence. In chapter 2 these factors will be tested in the lab with some experiments.

Quenching

Fluorescence quenching refers to the reduction in fluorescence intensity caused by various mechanisms, including excited state reactions, energy transfer (between energy levels), complex formation (e.g. number of lone pairs of ions), and collisional quenching.

The mechanisms for quenching can be divided into **static** ones and **dynamic** ones [6]:

- **Dynamic quenching** involves a collisional process between the fluorescent molecule and the quencher (usually the solvent) in the

excited state. **The collision leads to non-radiative energy transfer, reducing the fluorescence intensity.**

- **Static quenching** occurs when a non-fluorescent complex is formed between the fluorophore and the quencher without the need for molecular collisions. In the case for example of fluorescein (polar compound) and ethanol (polar compound), the formation of a ground-state complex may contribute to fluorescence quenching (See the section about solvent effects).

There are also self-quenching effects related to the concentration of the solution:

- **Self-Quenching Effects:** In highly concentrated solutions, the probability of fluorophores being in close proximity to each other is increased. This can lead to self-quenching effects, where the excited state energy of one fluorophore is transferred to another nearby fluorophore, effectively reducing the overall fluorescence intensity observed.

Reabsorption

In fluorescence, **reabsorption refers to the phenomenon where emitted light (fluorescence) from a substance is absorbed again by the same substance**, leading to a decrease in the observed fluorescence intensity and possible shift of peak wavelength (increase of stokes-shift). The reason behind reabsorption is **RET** (resonance energy transfer) [6].

Reabsorption is a phenomenon more evident **when the excitation and emission spectra of a substance overlap the most.**

In other words, **the smaller the stokes shift** (distance between excitation and emission wavelength) **the more the reabsorption.**

When a solution is more concentrated there are more excited phosphors that are available to be reabsorbed, for this reason for a very concentrated solution, the effect is very visible. As a consequence of

reabsorption, the lower part of the spectrum is reabsorbed (meaning the excited states do not decay in a radiative fashion emitting fluorophores but are reabsorbed), **and the peak emission wavelength shifts to longer wavelengths** (those states that are not reabsorbed and decay radiatively).

Fig 1.9 illustrates this phenomenon well, in the example Rhodamine 6G's absorption and emission spectrum overlap by a great margin, **the stokes shift is only 23 nm**.

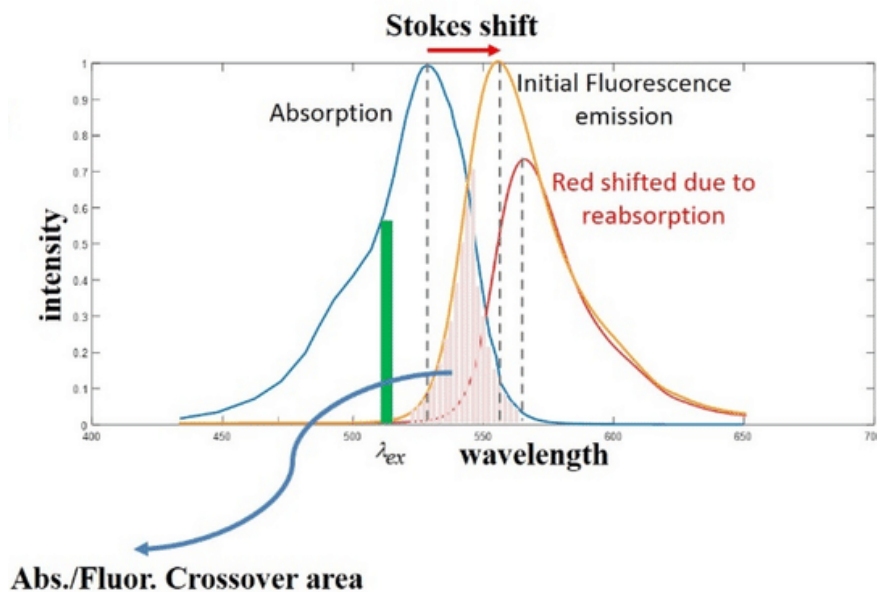


Fig 1.9: Illustration of the reabsorption phenomena, fluorophores emitted at the overlap wavelengths can be reabsorbed [23].

Reabsorption happens also for other reasons **related to the concentration of the solution**:

- **Increased Path Length:** In a very concentrated solution, the path length traversed by the emitted fluorescence is longer due to the higher density of fluorophores (the molecules responsible for fluorescence). Consequently, **there is a higher probability of reabsorption occurring along this longer path.**

- **Increased Chance of Collision:** In very concentrated solutions, the density of molecules is higher, leading to more frequent collisions between fluorophores. These collisions can result in energy transfer

processes such as Förster resonance energy transfer (FRET) or Dexter energy transfer, **which can contribute to reabsorption events.**

Temperature

Though it was observed for fluorescein in this work that temperature does have no dramatic effect on the emission spectrum's characteristics (see the lab experience in Chapter 3 regarding fluorescein) some research on the topic has highlighted possible effects on some fluorophores [10]. The most important effect the research highlights is that of **quenching** beyond a certain temperature. Citing [8]: *“The absorption and fluorescence spectra of peroxidase solutions is independent of temperature in the range from 10 to 45 degrees C. Above 45 degrees C, the absorption decreases in the visible range and increases in the ultraviolet. The intensity of fluorescence decreases with the increase in temperature. The temperature quenching of the fluorescence is accompanied by an energy transfer from tyrosine to tryptophan. The optimum enzyme activity is observed at about 40 degrees C, at 75 degrees C the activity ceases.”*

Other effects highlighted by research are:

- **The shift in Emission Spectra:** Temperature changes can alter the conformation and dynamics of molecules, affecting their electronic states and thus the fluorescence emission spectrum. In some cases, temperature variations can lead to shifts in the peak emission wavelength (Stokes shift) due to changes in the energy levels of excited states.
- **Rate of Photobleaching:** Photobleaching, the irreversible loss of fluorescence due to photochemical reactions, can be accelerated at higher temperatures. Increased temperatures can enhance the rate of molecular degradation and photochemical reactions, leading to faster photobleaching of fluorophores.
- **Viscosity Effects:** Temperature changes can also influence the viscosity of the solvent, affecting the molecular mobility and diffusion rates of fluorophores. Changes in viscosity can alter the rates of energy transfer processes and molecular interactions, impacting fluorescence lifetimes and intensity.

- **Quantum Yield:** Quenching induces a decrease of quantum yield.

Solvent effects

The solvent choice affects not only the solubility of the fluorophore in the solution but also its emission peak wavelength and intensity (quenching).

The main cause of solvent effects **is the difference in polarity** between solute and solvent. A polar solvent reduces drastically the fluorescence of a non-polar solute and enhances that of a polar solute [11, 12].

After the fluorophore has been excited to higher vibrational levels of the first excited singlet state (S_1), excess vibrational energy is rapidly lost to surrounding solvent molecules as the fluorophore slowly **relaxes** to the lowest vibrational energy level (occurring in the picosecond time scale) as depicted in **Fig. 1.7**. Solvent molecules assist in stabilizing and further lowering the energy level of the excited state by re-orienting (termed **solvent relaxation**) around the excited fluorophore in a slower process that requires between 10 and 100 picoseconds. This has the effect of reducing the energy separation between the ground and excited states, which results in a redshift (to longer wavelengths) of the fluorescence emission.

The more the difference in polarity between solute and solvent, the more the fluorophore emission peak is red-shifted (loses energy due to solvent relaxation process).

pH

The pH is another important factor that affects fluorescence, the ionization state of the molecules (both that of the solute and solvent) determines the intensity of fluorescence and also the peak wavelength, since the absorption spectrum depends on the ionisation state as explained in the “What is absorption and how does it work” section of this chapter.

How does the ionisation state of the molecule change the absorption spectrum? In most cases, the electronic states that are responsible for fluorescence are specific to ionisation state of a molecule. A change in pH may result in **the molecule bonding or dissociating an H⁺** (or even breaking the whole molecule altogether), therefore changing the density of electronic states [13]. Then the fluorescence (wavelength, but also quantum yield, lifetime, etc...) changes.

Let's analyze as an example the case of **fluorescein** (Fig. 1.10):

- Fluorescein exists in different species.
- In strongly acidic solutions, the cation is the predominate species
- With higher pH (towards more alkaline solutions) fluorescein ionises from its neutral species to its dianionic one.
- At pH values from 2 to 4, fluorescein exists in its neutral species, and as the pH values become more basic, within the range of 4.3 to 6.4, the monoanionic form is present.
- At pH above 6.4, the dianion is the most prevalent.

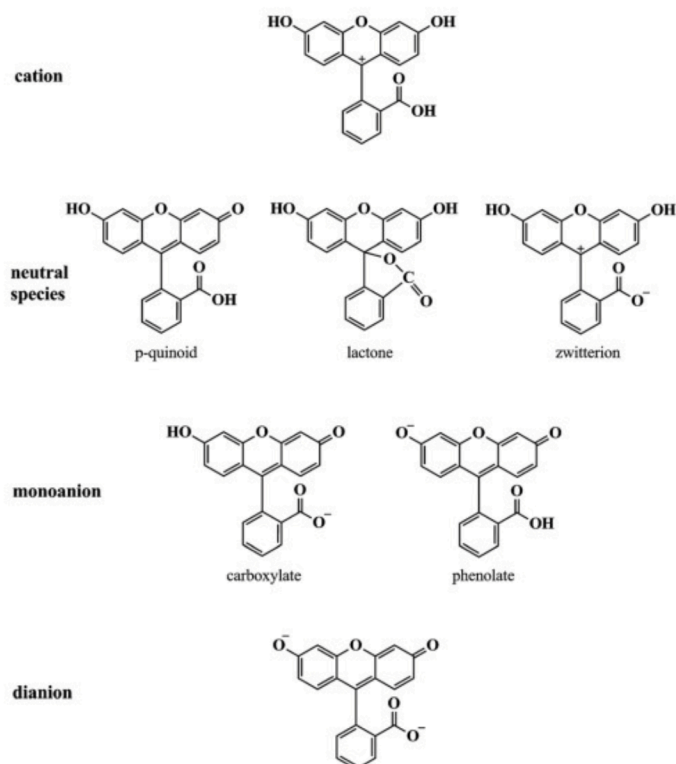


Fig 1.10: Fluorescein's different ionisation states [13].

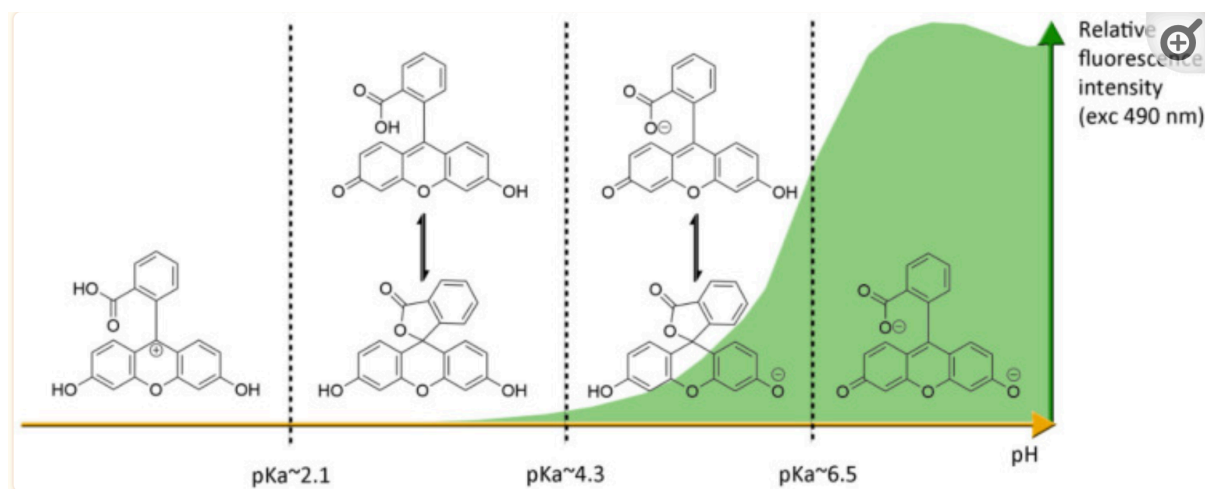


Fig 1.11: How fluorescein ionisation states evolve with pH [13].

The absorption and emission spectra are directly correlated to the ionisation state of the molecule, the ionisation state of the molecule indeed modifies the electronic structure of it.

The more a compound becomes ionised, the more it will fluoresce, that is because more donor groups (OH-) are available.

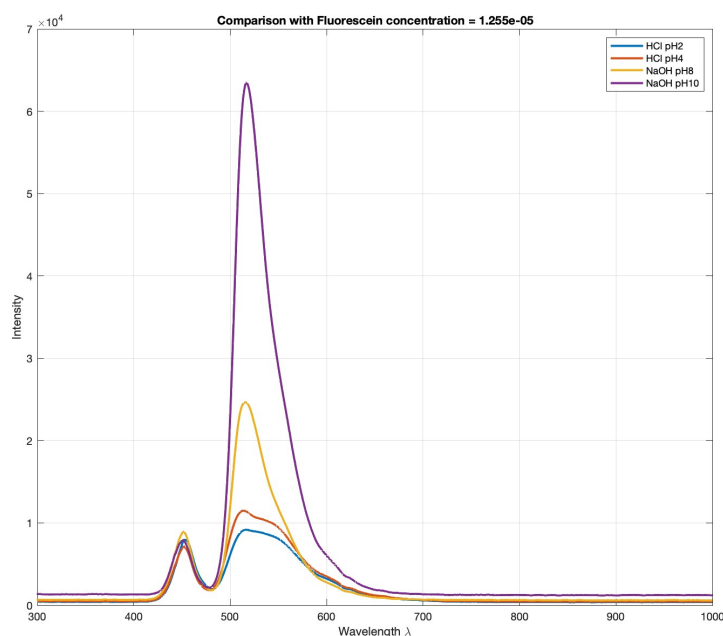


Fig 1.12: Fluorescein at different pHs, fluorescein's peak is the one on the right, the left peak is the LED's used in the experiment, the peak is approximately at 520 nm.

Indeed, recalling the discussion at the beginning of this chapter about how aromatic rings and donor groups contribute the most to fluorescence the picture is more clear of why pH in certain instances enhances fluorescence by making available more donor groups, more electrons are now easily available to make the jump to higher energy states.

This is why towards higher pH fluorescein has a higher quantum yield and less broad spectrum (Fig. 1.12). The main contributors of absorption (and so emission) are the energy states related to the new OH- donor groups.

It is important to note that each compound can behave very differently with pH changes, and so does the fluorescence emission behaviour, since pH modifies the electronic and chemical structure of the compound. Also, very high values of pH (basic) usually destroy the molecule structure, as can very low levels of pH (acidic). Chapter 2 will also delve into the

examination of how curcumin behaves in response to pH alterations. The observed behaviour is anticipated to differ from that of fluorescein.

Chapter 2: Laboratory

Measurements

Lab setup, measurements, and experiments

In this chapter, will be presented the experimental work carried on in the *PhotoNext* LINKS lab at PoliTo premises on the subject of fluorescence.

All the laboratory experiments presented in this chapter have the goal of verifying the theory explained in the previous chapter, quantifying phenomena (such as quenching caused by the different polarities between solvent and solute), and more, in general, to allow the author of this thesis to familiarise with the instruments and become competent in measuring and analysing fluorescence phenomena.

First will be presented the laboratory setup common to most experiences, both hardware and software. Then in the “Experiment #” sections will be presented the goal of the single experiences and the experimental results.

The list of experiments (chronologically) carried on at the lab is as follows:

- **Experiment 1: Chlorophyll.**
- **Experiment 2: Curcumin.**
- **Experiment 3: Fluorescein.**
- **Experiment 4: Fluorescein + Chlorophyll + Curcumin.**
- **Experiment 5: pH experiments.**
- **Experiment 6: Dairy milk.**

Lab setup

The laboratory setup common to most experiments is composed of (Fig. 2.1 and 2.2):

- **M395L4 and M450LP2 mounted LEDs** (nominal wavelength 395 nm and 450 nm respectively) with LED driver LEDD1B (Thorlabs).
- **Avantes spectrometer** (AvaSpec-3648) connected to a PC with Avantes acquisition software installed.
- Standard avantes fiber optic cable (multimodal).
- Plastic and quartz cuvettes.
- Cuvette holder.
- Micrometric pipette and millimetric pipette.
- Beaker or equivalent glass for mixing the solution and quantifying the volume.
- Mini scale (0.01 to 100 grams).
- Water (tap water) and isopropyl alcohol are used in most experiments as solvents, in experiment 5 (“pH experiments”) different solvents have been used to quantify how pH impacts fluorescence emission.

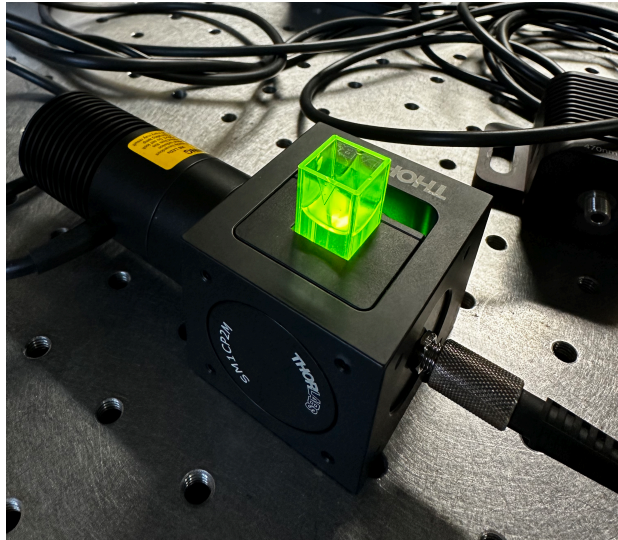


Fig 2.1: M450LP2 LED illuminating fluorescein, plastic cuvette, cuvette holder and fiber optical cable on the front side to the LED (in most experiences it will be positioned in one of the two other sides when the LED intensity is too strong).

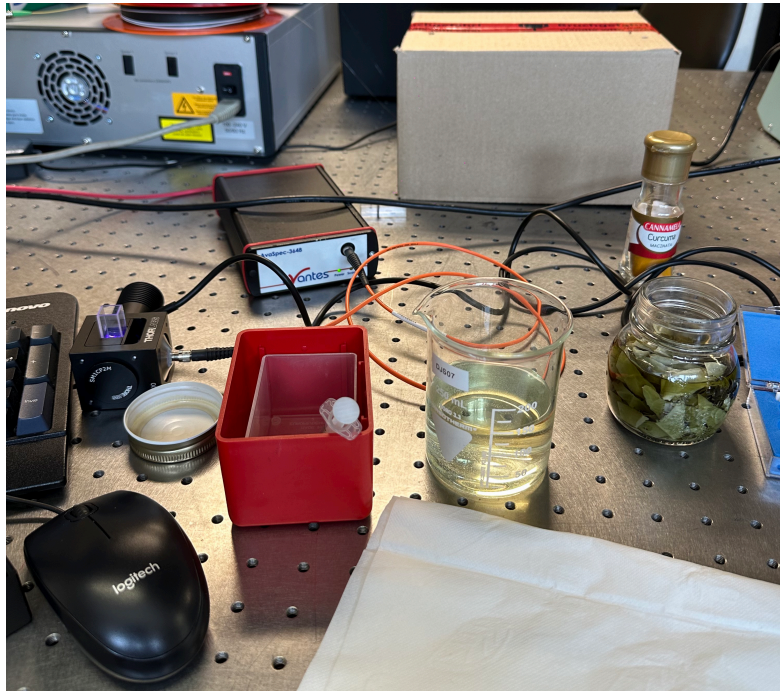


Fig 2.2: Some of the above described lab equipment.

Avantes spectrometer

The spectrometer is the most important piece of hardware in the following experiments, it captures the spectrum (between 200 nm and 1100 nm) of the light at its input (through a fiber optic cable usually) and connects to the PC where a program (acquisition software) is installed which displays the spectrum on screen and allows saving it on the PC.

Citing the documentation about the Avantes AvaSpec-3648[14]:

“The AvaSpec-3648 Fiber Optic Spectrometers is based on the AvaBench-75 symmetrical Czerny-Turner design with a 3648 pixel CCD Detector Array. The spectrometer has a fiber optic entrance connector (standard SMA), collimating and focusing mirror and diffractive grating. A choice of 16 different gratings with different dispersion and blaze angles enable applications in the 200-1100nm range. The AvaSpec-3648 comes with a 16 bit AD converter, and USB2.0 high speed interface. The AvaSpec- 3648 is especially suitable for measuring high resolution applications and thanks to the 10µsec electronic shutter has a wide dynamic range. An optional detector coating can enhance the CCD performance for the UV range and a detector collection lens offers high sensitivity.

Digital IO ports enable external triggering and control of shutter and pulsed light sources from the Avantes line of instruments.

The AvaSpec-3648 has a USB2 interface with fast data sampling of 270 spectra per second and data transfer in 3.7 msec and supports analog in- and outputs as well. Optional Bluetooth® (-BT) communication and an SD card for on-board saving of spectra can be added. The AvaSpec-3648-USB2 runs on USB power and comes with AvaSoft-basic, a complete manual and USB interface cable. Multiple (up to 127) USB2 spectrometers with different detector types can be externally coupled’.



Fig 2.3: Avantes spectrometer used (AvaSpec-3648) [14].

This spectrometer is compact, lightweight, and has more than the necessary resolution for the measuring applications of this thesis work.

Using Avantes' acquisition software (Fig. 2.4) it is possible to save on a PC the spectrums obtained through the spectrometer for later analysis with MATLAB or other applications.

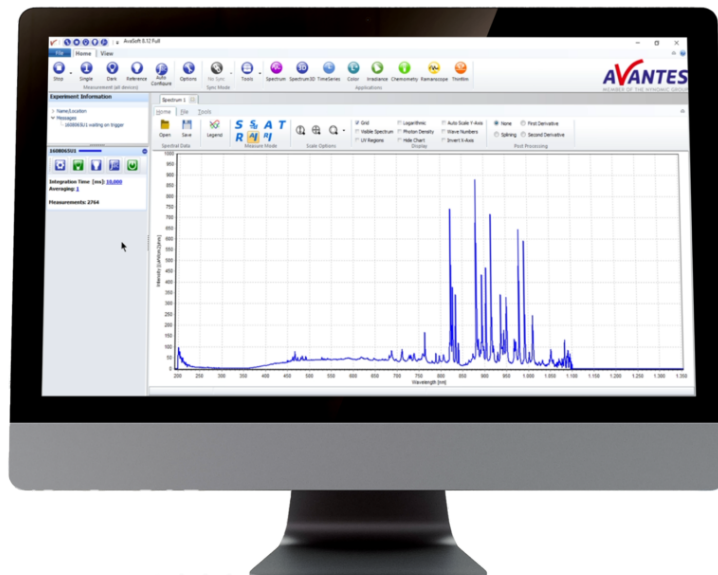


Fig 2.4: Avantes' spectrum acquisition software (AvaSoft) [14].

How does the lab setup work

M395L4 and M450LP2 are LEDs designed to be mounted on the cuvette holder in Fig. 2.1. They have a relatively broadband spectrum (around 100 nm for the M450LP2 and around 60 nm for the M395L4). They are mounted on the cuvette holder with no optical filter for their spectrum, so the LEDs' spectrum has to be removed manually later in the analysis of the spectrum (via Matlab as an example).

The optical fiber mounted on one of the ends of the cuvette holder captures light inside the cuvette holder. The choice of where the optical fiber has to be mounted depends on both the intensity of the LED mounted and the cuvette used. If the LED light is too intense (as in most cases in the experiments in this work) it needs to be mounted on one of the sides not facing directly the LED. Also if the cuvette is a quartz cuvette or just a cuvette with each side having a different glass thickness the proper side for collecting the light needs to be chosen.

The sides of the cuvette holder (including the top side) that are not used need to be closed with a cap to avoid any other light interference coming from the sides.

The light inside the cuvette holder by the optical fiber is transmitted to the Avantes spectrometer and displayed on the AvaSoft acquisition software on the PC connected via a USB cable.

Experiment 1: Chlorophyll

Goals

The first experiment carried out in this experience is the characterisation of chlorophyll fluorescence. The choice of chlorophyll is purely a result of how easily obtainable it is, the process to make it is indeed the following:

- Fresh spinach leaves.
- Food alcohol (95%) from a regular grocery store.

- Leave the spinach leaves in alcohol inside a closed container for about one night.

From this process a dark-green solution (Fig. 2.5) of chlorophyll is created.



Fig. 2.5: Chlorophyll solution extracted from spinach.

As a first experiment the goal of this experience was to get acquainted with how all the lab setup works and how to process the resulting spectrum on Matlab, the unknown concentration of chlorophyll resulting from this experiment doesn't allow to quantify any fluorescence quantity (quantum yield, LOD of the instrument).

What is Chlorophyll

Citing Britannica [15]: “Chlorophyll, any member of the most important class of pigments involved in photosynthesis, the process by which light energy is converted to chemical energy through the synthesis of organic compounds. Chlorophyll is found in virtually all photosynthetic organisms, including green plants, cyanobacteria, and algae. It absorbs energy from light; this energy is then used to convert carbon dioxide to carbohydrates.”

Chlorophyll displays fluorescent behavior when illuminated by light in the 400 - 700 nm range. Different kinds of Chlorophyll exist in nature (mainly “Chlorophyll A” and “Chlorophyll B”). Below in Figures 2.5.1 and 2.5.2 are displayed the emission and absorption spectrums for Chlorophyll A. Quantum yield for Chlorophyll A is approximately 0.32 [16] and the emission peak is around 670 nm.

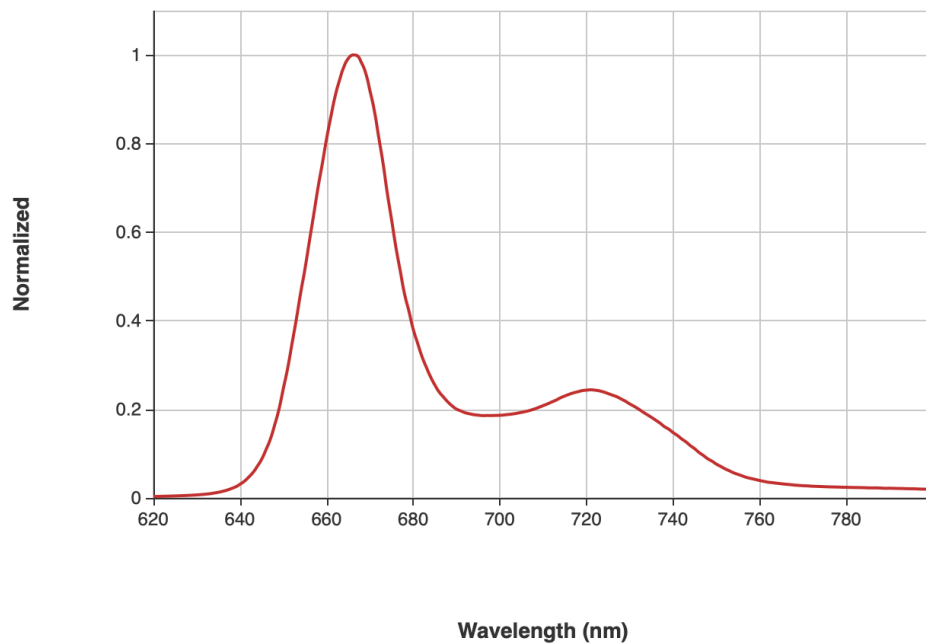


Fig 2.5.1: “Chlorophyll A” emission spectra, courtesy of PhotoChemCad [16].

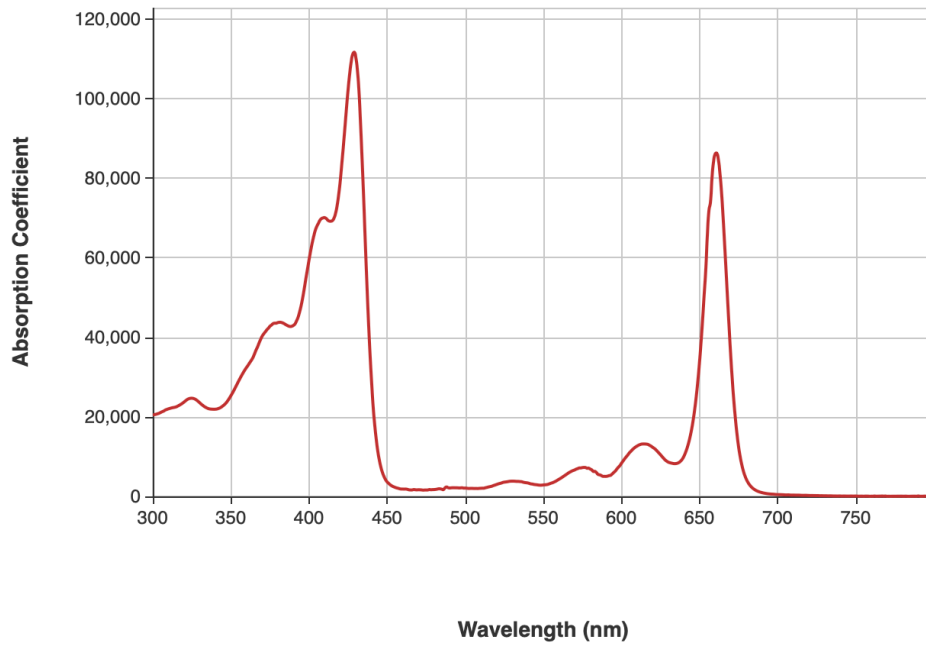


Fig 2.5.2: “Chlorophyll B” absorption spectra, courtesy of PhotoChemCad [16].

Preparation of the solutions

The solutions inside the plastic cuvettes are prepared in the following way:

1. Fill a beaker with tap water (150 ml).
2. Extraction using a millimetric pipette of a known quantity of the solution (between 0.5 ml and 3 ml depending on the single measurement to be carried).
3. Inserting the content of the millimetric pipette inside the beaker and mixing of the two solutions (chlorophyll in ethanol and tap water).
4. Acquiring the spectrum from the spectrum analyser and saving it for later analysis on MATLAB.
5. And repeating from 2 to 4 as many times as needed for the experiment, increasing in this way the concentration of chlorophyll inside the beaker.

The first part of this experiment was to observe how the intensity of fluorescence increased increasing the concentration of the solution in tap water and the results are available in the “Experimental results” section of this paragraph.

The second part was trying to find the lowest concentration of chlorophyll that can be detected by the spectrometer. A micrometric pipette and a beaker filled with 200 ml of water were used to try to find it.

The micrometric pipette was filled with a volume of the chlorophyll in ethanol solution variable between 2.5 μl and 0.1 μl to find the lowest concentration detectable (LOD). Results are in the “Experimental results” section of this paragraph.

Setup

- **M395L4** mounted LED (nominal wavelength 395 nm) with LED driver LEDD1B (Thorlabs) set at 0.5 A

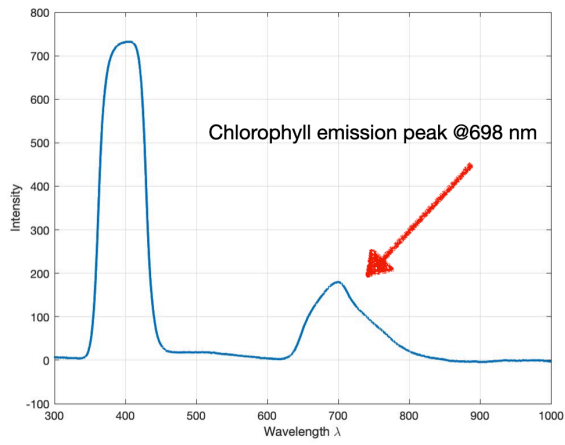
- Avantes spectrometer (AvaSpec-3648) connected to a PC. Integration time for every measurement = 50 ms, Averages = 100. Smoothing window = 100.
- Standard avantes fiber optic cable mounted on one of the lateral sides of the cuvette holder (not facing directly the LED).
- Plastic cuvette (3 ml).
- Cuvette holder.
- Micrometric and millimetric pipette.
- Beaker (200 ml max volume).

Experimental results

From this experiment, the following spectrums and plots have been obtained:

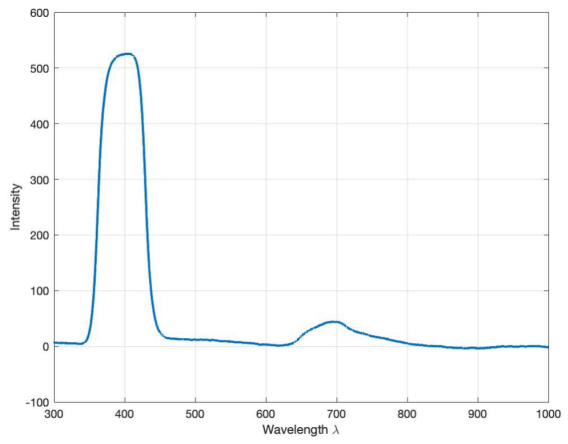
- Fig 2.6, 2.7, 2.8: Spectrums at different concentrations of chlorophyll in ethanol/tap water, the latter spectrum (Fig. 2.8) represents the LOD for the detection of the solution's fluorescence for this experiment. The final concentration that has been reached is **0.07% Solution/Tap water**. Note that the result is given as a percentage since the molar concentration of the chlorophyll obtained with the process explained in the first part of the paragraph is not known (could not be obtained through the instruments at disposal in the lab).
- Fig 2.9: Concentration of Solution/Tap water [%] vs the intensity of fluorescence. It is important to note that the intensity value is a value relative to the digital counts of the spectrometer.

Fig 2.6



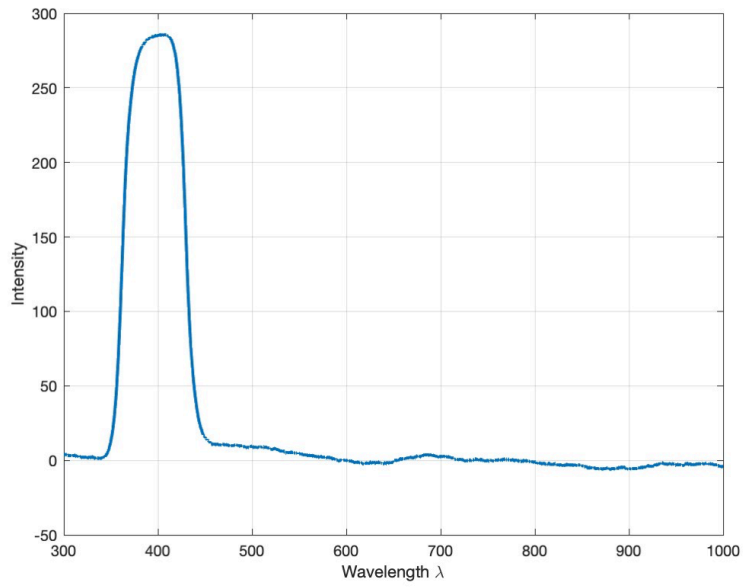
**@3.43% Solution/
Tap water**

Fig 2.7



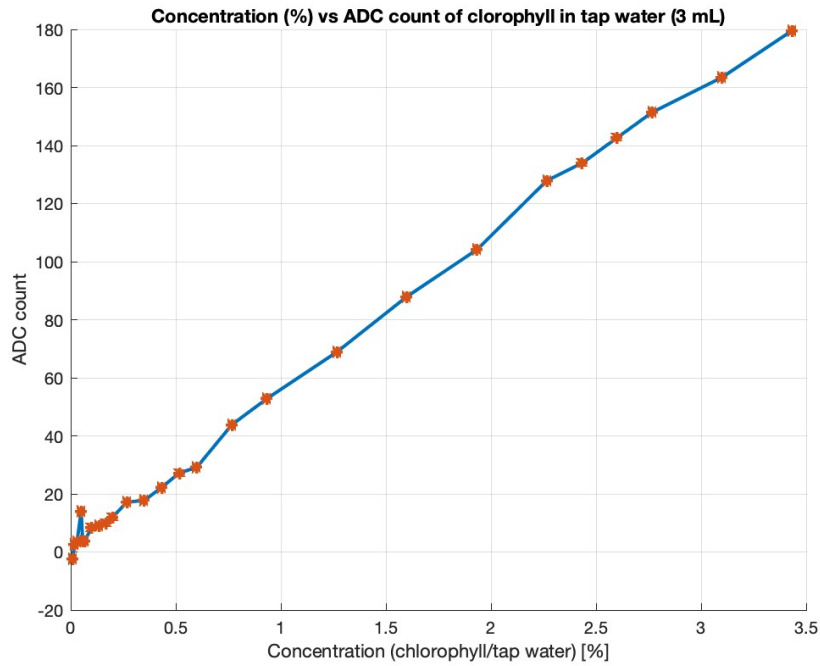
**@0.77% Solution/
Tap water**

Fig 2.8: LOD detected (little peak visible at 698 nm).



**@0.07% Solution/Tap
water**

Fig 2.9: Fluorescence intensity vs concentration plot



If compared to the emission spectrum provided by Photochemcad (Fig. 2.5.1) the overlap of the found emission spectrum in the lab experience shows some difference both in the shape of the spectrum and the emission peak probably due to the many kinds of chlorophyll found in nature (Fig. 2.10 shows the overlap).

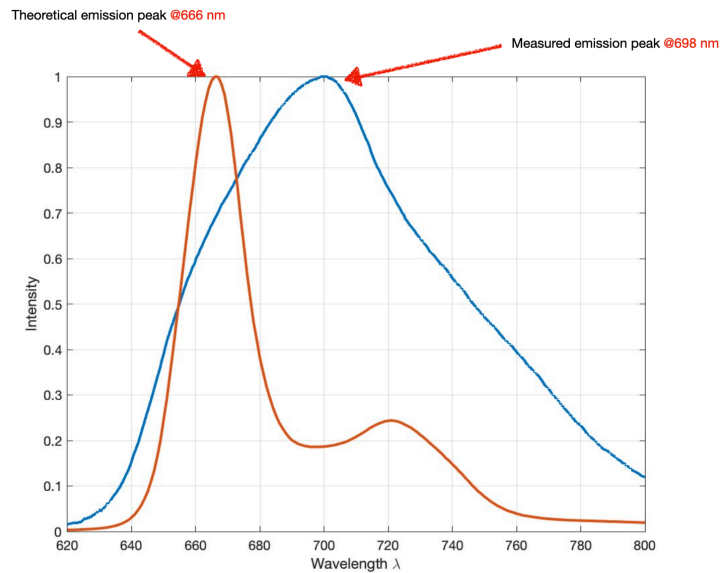


Fig. 2.10: overlap of the spectrum found in PhotoChemCad and that acquired in this lab measurements.

Experiment 2: Curcumin

Goals

The goals of this lab experience were more or less the same as the previous one with Chlorophyll: characterising the fluorescence of an easily obtainable compound such as curcumin. This time though it was possible to characterise the fluorescence more precisely because the concentration of curcumin in the compound was known, so it was possible to obtain a more precise LOD, though it should be disclaimed that curcumin being a very apolar substance is not easily soluble in water, ethanol was this time used as a solvent. **Ethanol is less polar than water but still a polar compound, so to make curcumin more soluble in ethanol very small quantities have been diluted in big volumes of ethanol, providing the best possible result with the combination of these two substances differing greatly in polarity.**

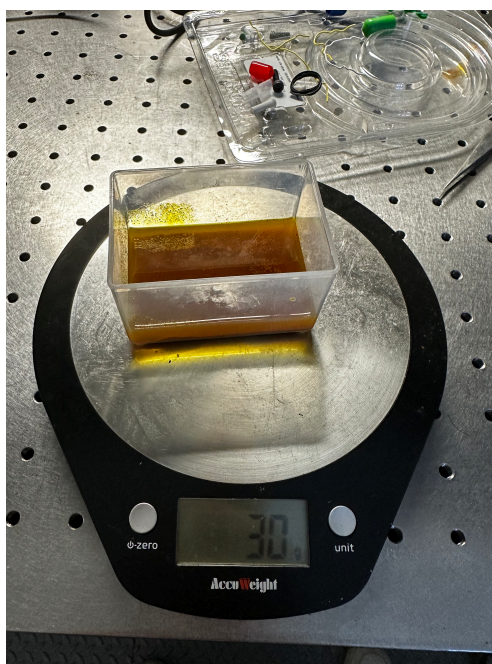


Fig. 2.11: Curcumin and ethanol mixed.

What is Curcumin

Curcumin is a chemical compound found mainly in turmeric, a plant that is native to India and Indonesia. Curcumin appears bright yellow to the human eye. It fluoresces in the right conditions, if turmeric is sprinkled into alcohol while illuminating it with UV light, a bright green-yellow fluorescence is seen. Alcohol is used as curcumin is soluble in alcohol but not in water and this helps make the fluorescence visible [17]. Curcumin's chemical structure differs subtly in acidic and alkaline solutions. This allows it to be used as an indicator. When added to acids it remains yellow. However, when added to an alkaline solution above pH 8, the shift of a hydrogen atom causes the compound to change colour, giving a red hue. The molar concentration of curcumin is 368.38

Curcumin absorption and emission spectrums are reported below in Figs 2.12 and 2.13. **The emission peak of fluorescein is around 540 nm and its quantum yield is very low (0.063).**

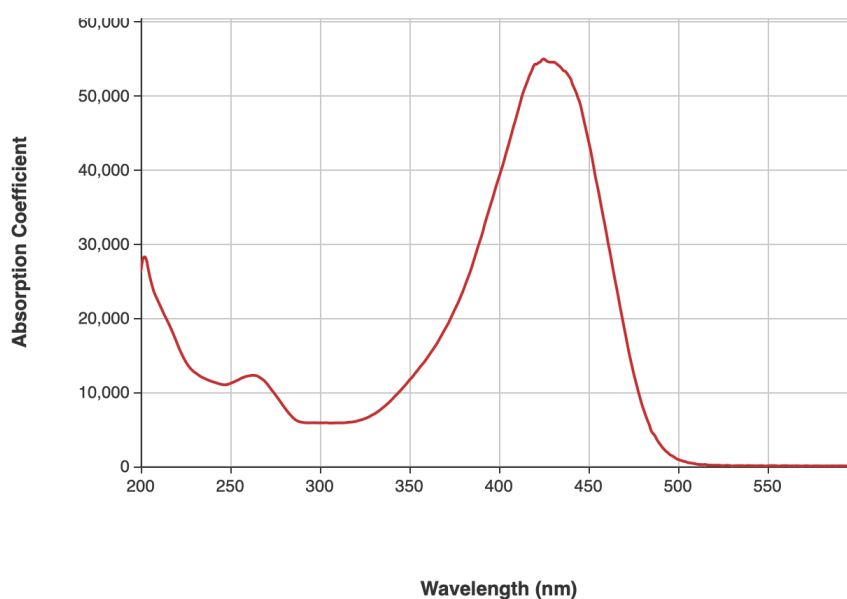


Fig 2.12: Curcumin's absorption spectrum, source: PhotoChemCad [30].

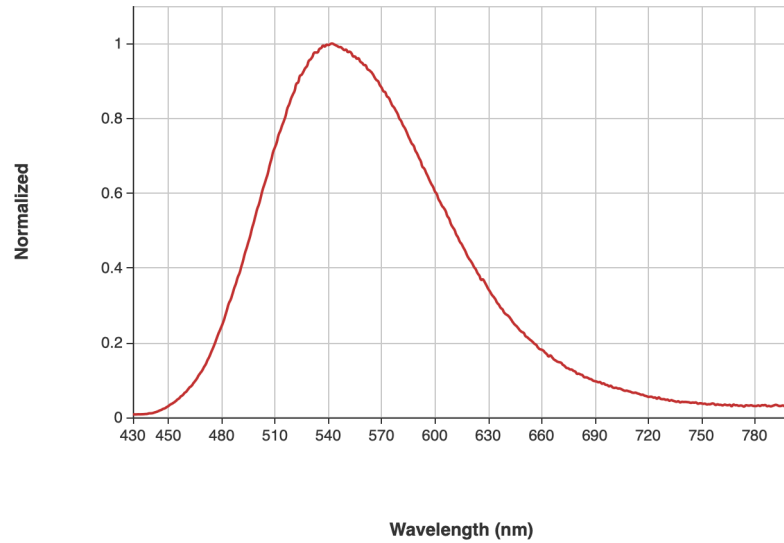


Fig 2.13: Curcumin's emission spectrum, source: PhotoChemCad [30].

Preparation of the solutions

Curcumin was extracted from a little bottle containing curcuma bought from a grocery store and diluted in ethanol. The following are the passages done to prepare the solutions of ethanol-curcumin to be analysed:

- 5 grams of curcuma bought from a regular grocery store (in which curcumin is contained) have been taken.
- The said curcuma has been diluted in 25 grams of ethanol (95% concentration), obtaining the solution in Fig 2.11.
- The solution was then diluted in even more ethanol with the help of the millimetric pipette inside different plastic cuvettes (each containing a different volume of ethanol) to measure it in different concentrations.
- Then the fluorescence spectrums of the different solutions in the pipettes have been acquired from the spectrum analyser and saved in the PC for later analysis on MATLAB.

Setup

- **M450LP2 mounted LED (nominal wavelength 450 nm) with LED driver LEDD1B (Thorlabs) set at 1.2 A.**
- Avantes spectrometer (AvaSpec-3648) connected to a PC. Integration time and averages vary for each measurement, generally the number of averages has been set low and the integration time high to capture very low concentrations. The smoothing window also varies but is generally set to a minimum (1-10). The spectrums have been smoothed later with MATLAB.
- Standard Avantes fiber optic cable mounted on one of the lateral sides of the cuvette holder (not facing directly the LED).
- Plastic cuvette (3 ml).
- Cuvette holder.
- Millimetric pipette.
- A commercial scale (sensitivity: 1 gram).

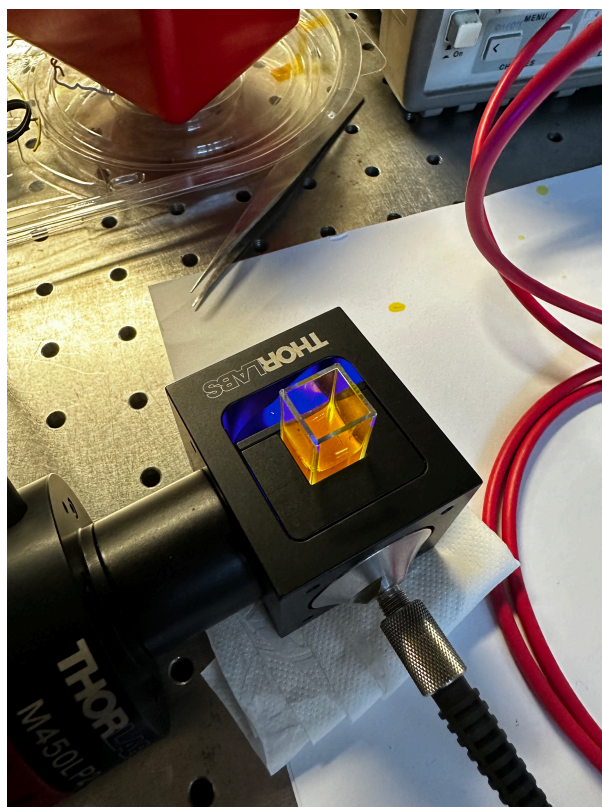


Fig. 2.14: The M450LP2 LED illuminating one of the solutions prepared.

Experimental results

From this experiment the following spectrums and plots have been obtained:

- Fig 2.15 and Fig 2.16: These plots show a series of spectrums for curcumin obtained at different concentrations, with the plot at the rightmost of Fig. 2.16 showing the LOD obtained for this compound ($4.2769 \cdot 10^{-4}$ mol/L). The plots also show an interesting fact: The emission peak wavelength changes with concentration, this is due to reabsorption, as explained in the dedicated section in Chapter 1. The more a solution is concentrated the more the effect of reabsorption red-shifts its peak emission spectrum.
- Fig 2.17: This plot shows the effect of reabsorption vs concentration, from low concentrations to high ones, showing a pattern that reaches a maximum for high concentrations.

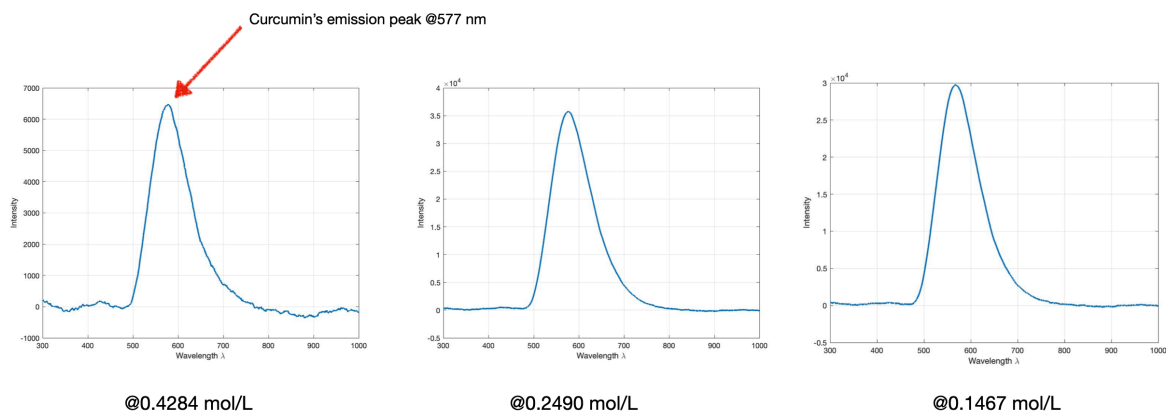


Fig 2.15: Solution of curcumin-ethanol at decreasing concentrations (left to right).

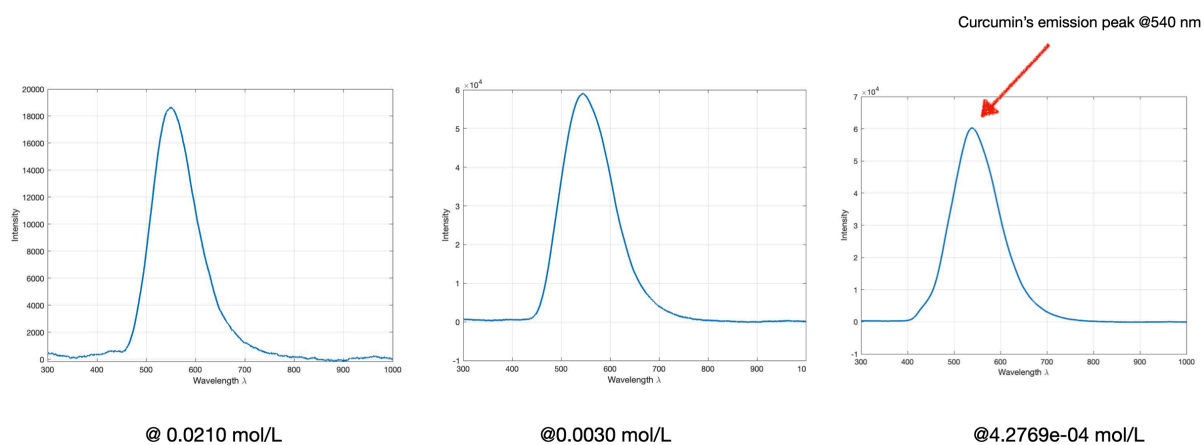


Fig 2.16: Solution of curcumin-ethanol at decreasing concentrations (left to right),
 LOD= $4.2769 \cdot 10^{-4}$ mol/L.

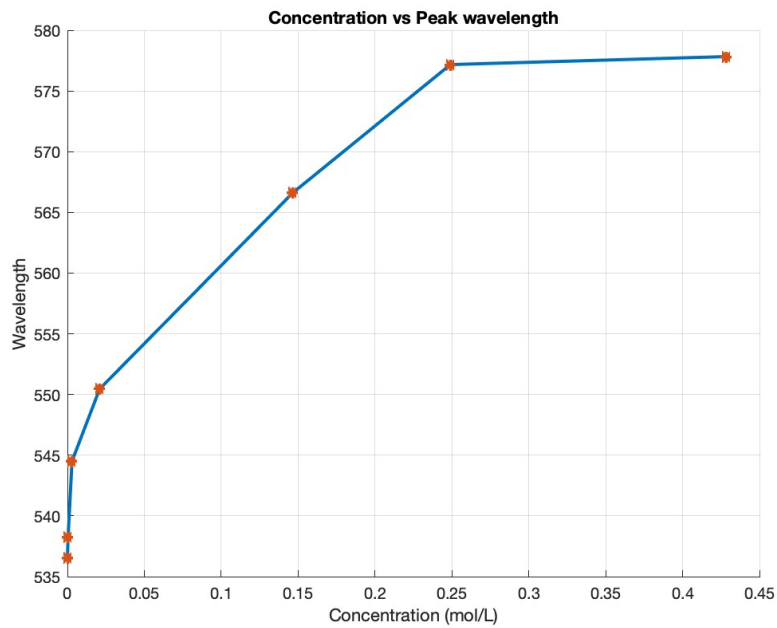


Fig 2.17: Plot of peak emission wavelength of the curcumin-ethanol solution vs concentration, the effect of reabsorption is clearly visible (a red shift at higher concentrations of almost 40 nm).

The red-shift due to reabsorption is clearly visible also to the naked eye, Fig. 2.18 shows two different solutions in the cuvettes. The solution on the left is the most concentrated one while the one on the right is the least concentrated.

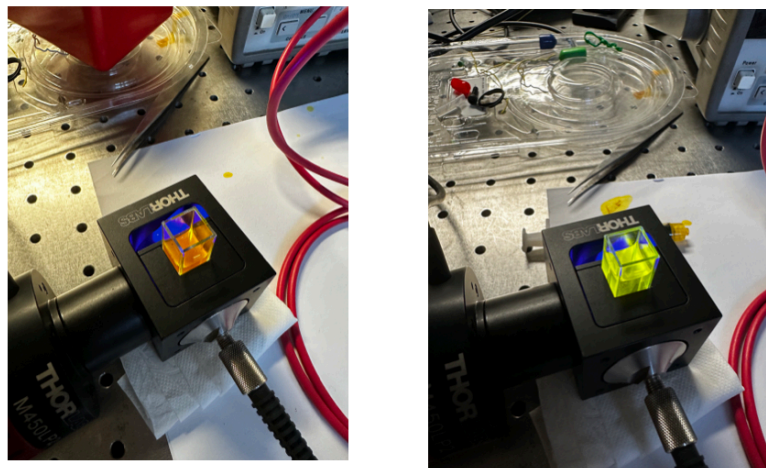


Fig 2.18: Pictures of the two cuvettes with different concentrations of curcumin in ethanol illuminated with LED light.

Experiment 3: Fluorescein

Goals

The goal of this lab experience was characterising the fluorescence of Fluorescein. In this lab experience this time the quantity of fluorescein and concentration was well established since the fluorescein used was bought from a seller of chemical compounds that well characterised its quantity and characteristics. Moreover, fluorescein is a polar compound, so can easily be mixed with water or ethanol, making the accurate measurement of the concentration that has been mixed in water/ethanol possible. The LOD in this case was established with more precision than in the previous experiments (chlorophyll and curcumin).

Fluorescein has a Molar mass of 332 g/mol.

What is Fluorescein

Fluorescein, as the name hints, is a chemical agent with remarkable fluorescence (quantum yield in water around 0.97). It is used in different fields, such as medicine, as a biomarker for the identification of non-diseased tissues, tumor-affected tissues, or histological markers [18].

Peak excitation occurs at 495 nm and peak emission at 520 nm, specifically observed in the deprotonated form within a basic solution. Fluorescein, with a pKa of 6.4, undergoes ionization equilibrium, resulting in pH-dependent absorption and emission spanning the pH range of 5 to 9. Moreover, the fluorescence lifetimes of the protonated and deprotonated forms of fluorescein are approximately 3 and 4 ns, respectively, facilitating pH determination through methods not reliant on intensity measurements [19].

The absorption and emission spectrums from PhotoChemCad are displayed in Fig.s 2.19 and 2.20.

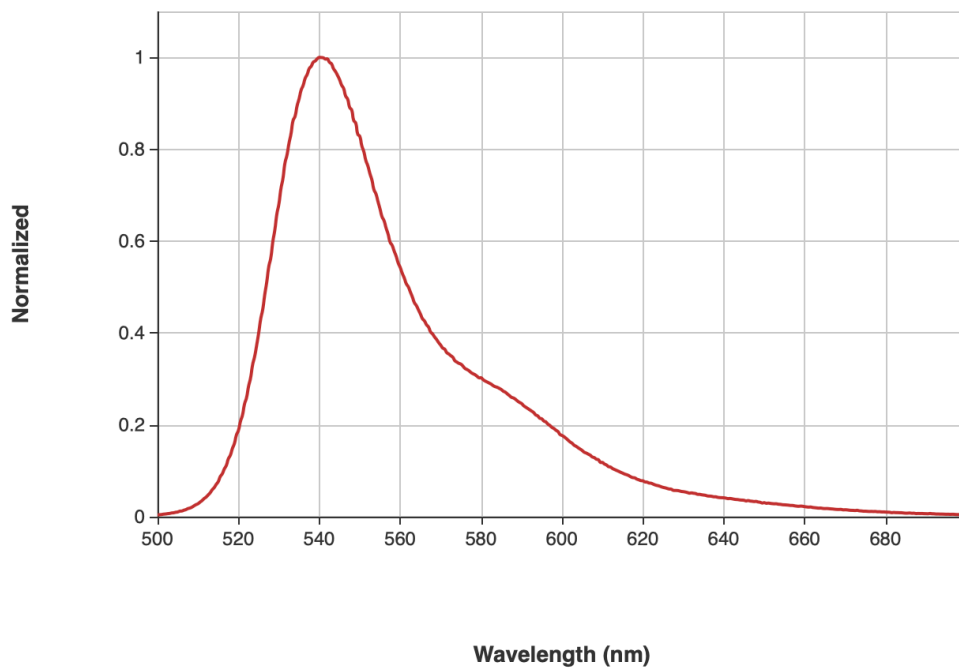


Fig 2.19: emission spectrum of fluorescein, courtesy of PhotoChemCad [31].

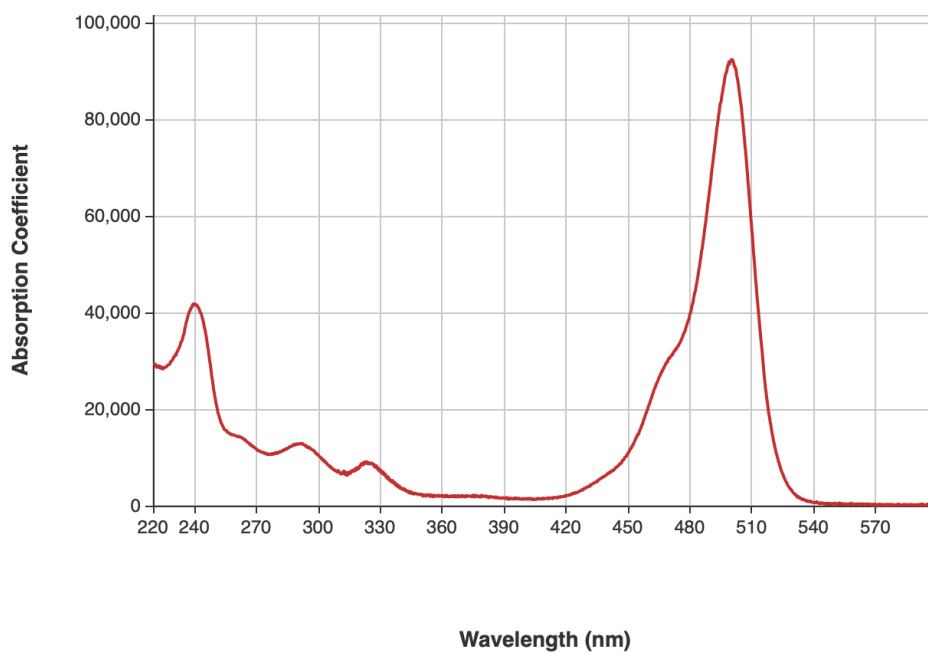


Fig 2.20: absorption spectrum of fluorescein, courtesy of PhotoChemCad [31].

Preparation of the solutions

Fluorescein was acquired from a seller online that sells chemical compounds (Hanse Pro Uranina, 100 g). The passages to prepare the solutions have been the following:

1. Weighted and acquired 1.1 grams of the fluorescein powder in a cuvette with the help of a precision scale (Fig. 2.21)
2. Using a 30 cl glass fluorescein has been mixed and diluted 11 times using tap water in the following way:
 1. First dilution: filled the glass with 20 cl of water and mix with the fluorescein in the cuvette of Fig 2.21
 2. 3 ml of the content of the glass has been inserted in a cuvette inside the cuvette holder and illuminated by the LED for measurement with the spectrometer and spectrum acquisition with the PC.
 3. The 3 ml of the cuvette have been reinserted in the glass.
 4. 15 cl out of the 20 cl of content in the glass have been removed.
 5. The solution has been diluted again with 15 cl of water (new tap water not previously mixed with fluorescein) (Fig. 2.22).
 6. The process from 2 to 6 has been repeated 11 times.

Final concentration reached with the dilutions: $2.7398 \cdot 10^{-10}$ mol/L (**possible LOD**).

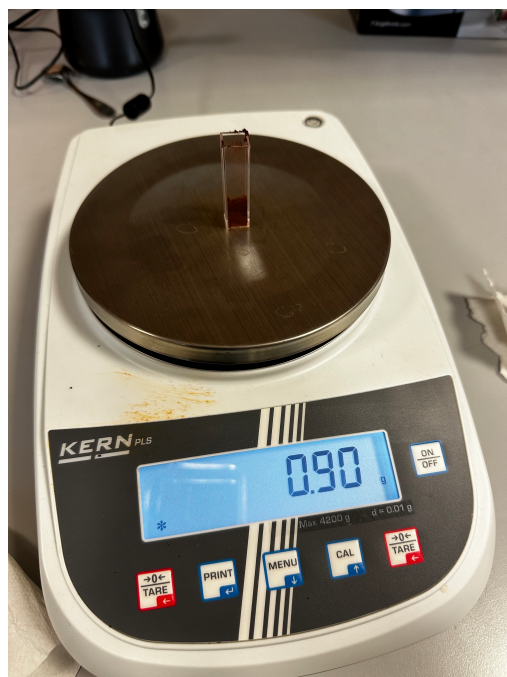


Fig 2.21: sample of fluorescein being weighted with the precision scale, 0.2 grams of fluorescein have been added after this pictures has been taken.



Fig 2.22: fluorescein/water solution on the left in the 30 cl glass, tap water in the plastic container on the right.

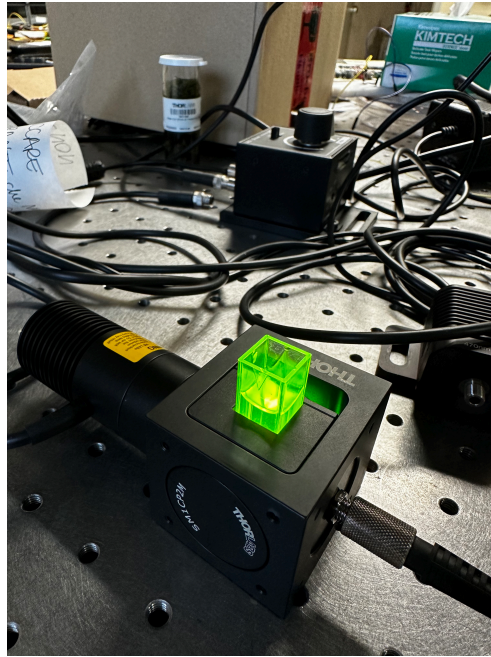


Fig 2.23: fluorescein/water solution illuminated by LED.

Setup

- **M450LP2 mounted LED (nominal wavelength 450 nm) with LED driver LEDD1B (Thorlabs) set at 1.2 A.**
- Avantes spectrometer (AvaSpec-3648) connected to a PC. Integration time = 6 s, averages = 3 for each spectrum acquired.
- Standard Avantes fiber optic cable mounted on one of the lateral sides of the cuvette holder (not facing directly the LED).
- Plastic cuvette (3 ml).
- Cuvette holder.
- Millimetric pipette.
- A precision scale (KERN PLS).

Experimental results

From this experiment, the following spectrums and plots have been obtained:

- Fig 2.24: the plots show the fluorescence of fluorescein in different solutions with different concentrations (from left to right the concentration is decreasing). The saturated peak on the left side of each plot is the LED's light spectrum. Has can be seen from these measurements fluorescein fluorescence is clearly visible even at very low concentrations.
- Fig 2.25: plot showing the intensity of the fluorescence of fluorescein vs concentration.

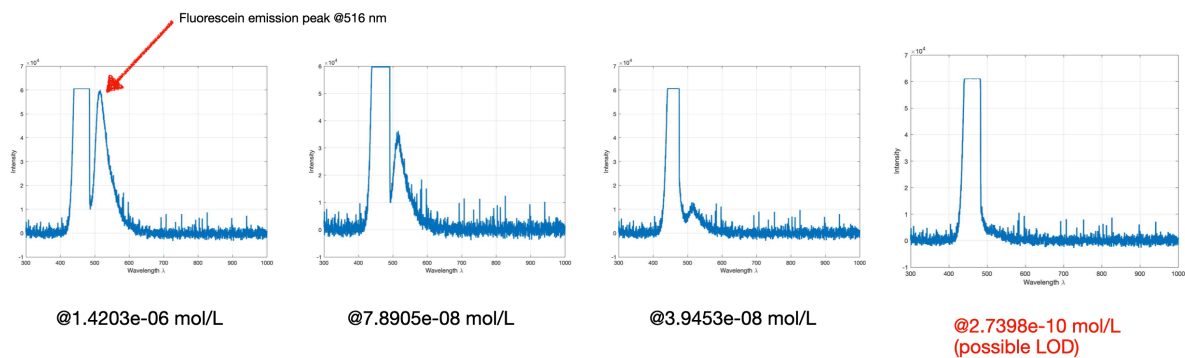


Fig 2.24: Fluorescein's fluorescence at different concentrations (left to right is the direction of decreasing concentration).

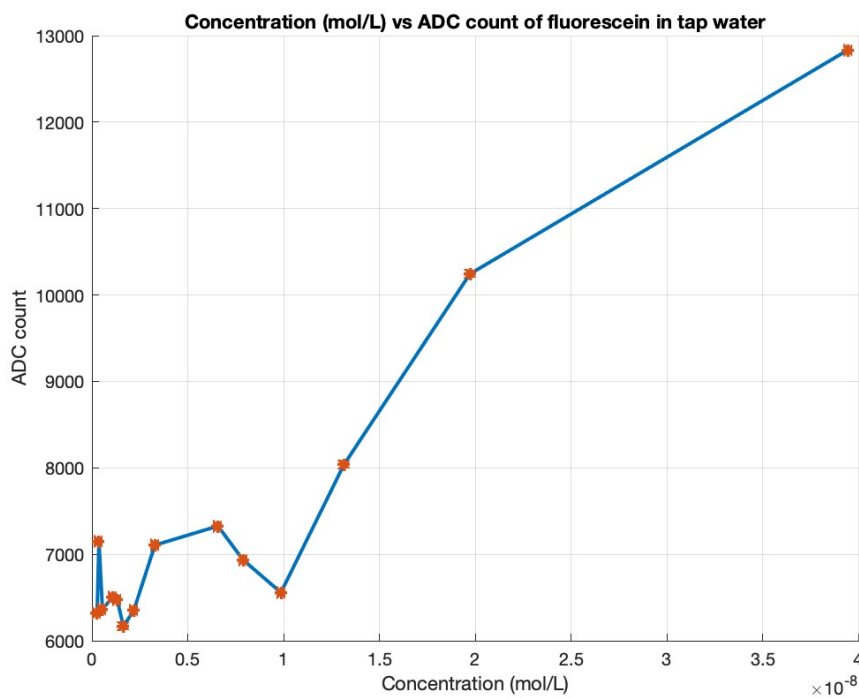


Fig 2.25: Fluorescein's fluorescence intensity vs concentration.

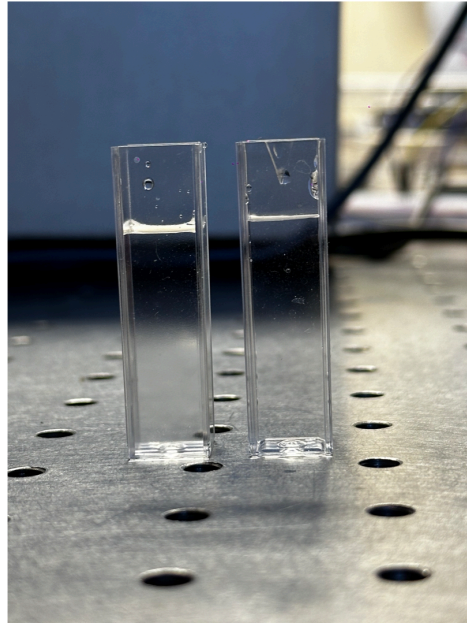


Fig 2.26: Picture showing the unnoticeable different (with the naked eye) of the color between tap water (left cuvette) and tap water + fluorescein (right cuvette) at $2.7 \cdot 10^{-10}$ mol/L of concentration.

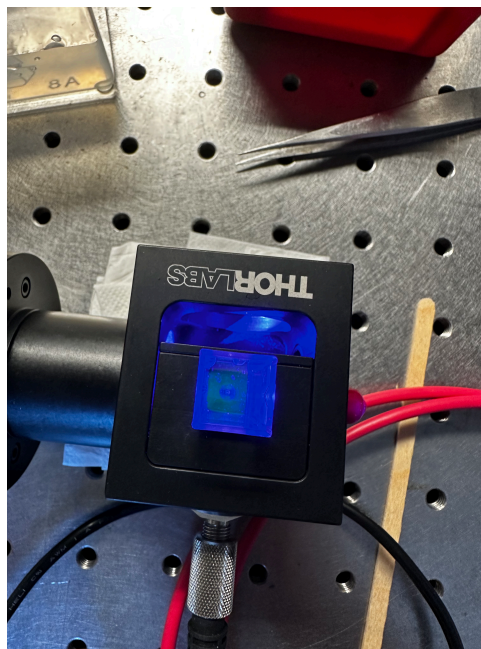


Fig 2.27: The above solution of fluorescein ($2.7 \cdot 10^{-10}$ mol/L) illuminated.

Similarly to the experiment with curcumin, also fluorescein shows a very pronounced stokes shift caused by reabsorption at high concentrations.

Fig 2.28 shows the difference in colour noticeable with only the naked eye. Fig 2.29 shows a plot representing the emission peak wavelength vs concentration, it shows the same trend seen in curcumin, **at higher concentrations the spectrum is red-shifted**, and the shift in this case (from the less concentrated solution to the most concentrated) is approximately 40 nm.

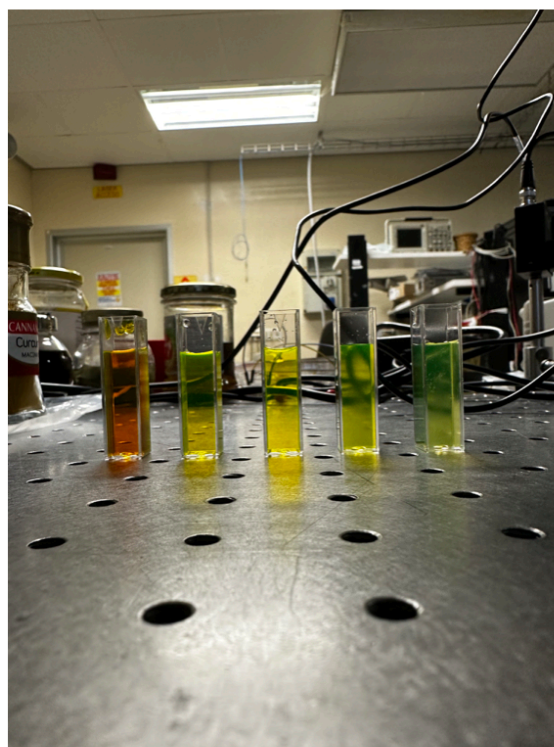
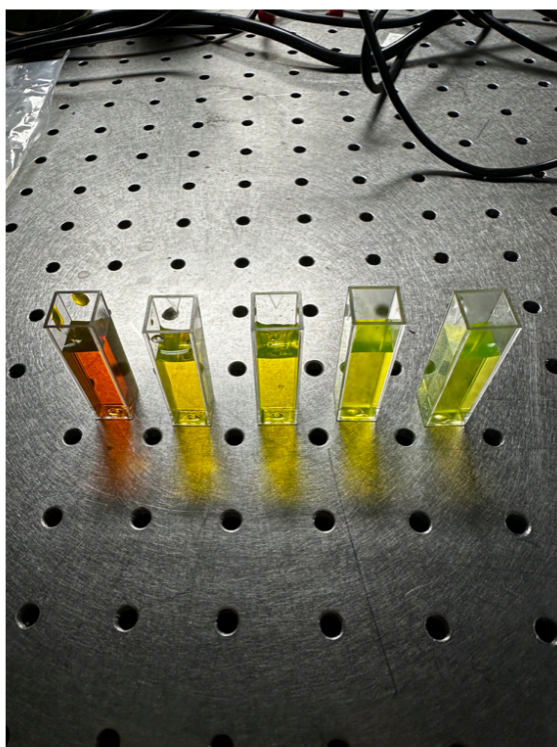


Fig 2.28: solutions of fluorescein+tap water at different concentrations (decreasing from left to right).

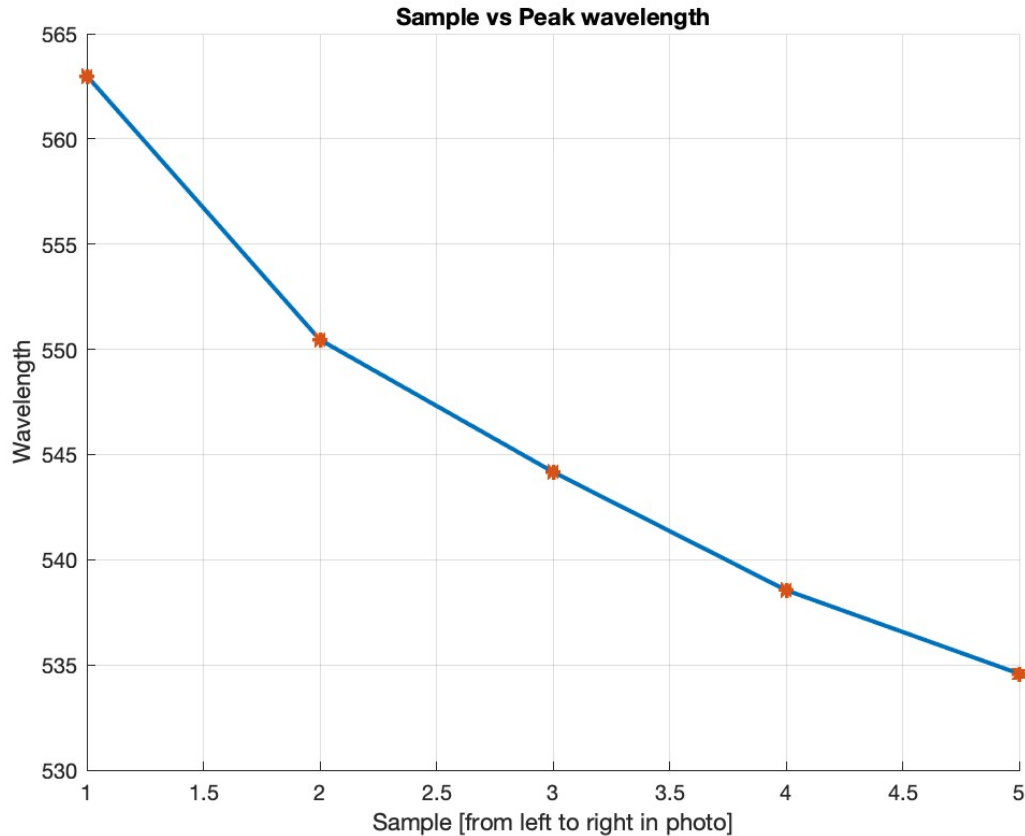


Fig 2.29: emission peak wavelength vs concentration for the fluorescein+tap water solutions. The samples are those in Fig. 2.28. It is a red shift of approximately 40 nm from the less concentrated solution to the most concentrated.

During the experiments with fluorescein also its dependence on temperature was tested. A **FBG (Fiber Bragg Grating)** was immersed in a solution with 0.0054 mol/L of concentration of fluorescein (Fig. 2.30), then it was illuminated with LED light like in the setup described above. It was then observed within 30 minutes, acquiring a spectrum every 3.3 minutes of observation. **In 30 minutes the the FBG detected a temperature shift of the solution of 10 degrees Kelvin. However, no noticeable shift in emission peak wavelength or quenching of intensity nor a change in the shape of the emission spectrum was detected (Fig. 2.31).**

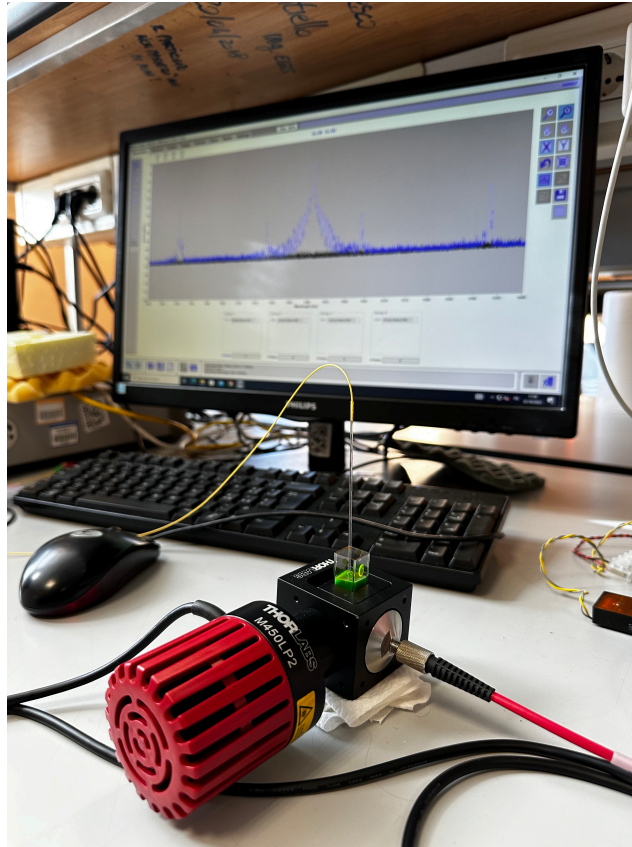


Fig 2.30: Setup with the FBG immersed in the solution.

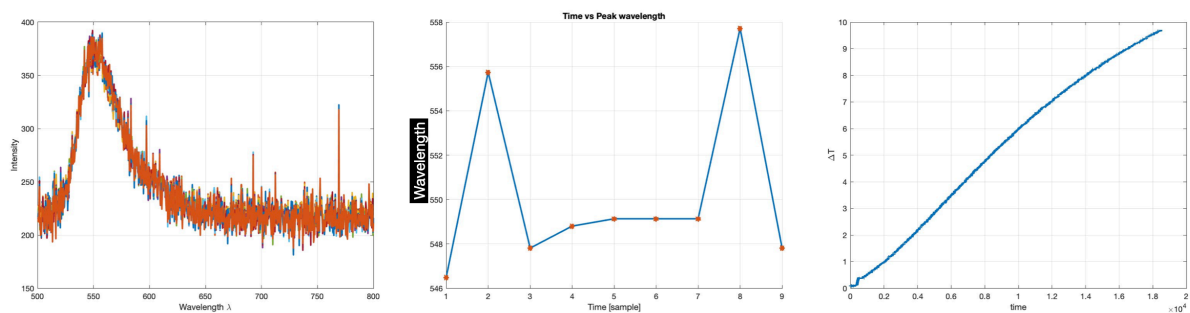


Fig 2.31: Leftmost plot shows the plots acquired overtime overlapped, the central Fig. the emission peak wavelength of said plots, the rightmost plot shows the change in temperature detected by the FBG.

Experiment 4: All together: fluorescein, chlorophyll and curcumin

Goals

The goal of these experiments was primarily to observe all the spectrums of all the analysed compounds together (mixed in a solution) and see what happens and to find a possible detection algorithm for detecting every one of the compounds involved.

Preparation of the solutions

A first solution with all three compounds has been prepared, the solvent utilised has been ethanol, since curcumin as already explained does not mix well with water. The solution has been prepared in the following way:

- 0.5 ml of chlorophyll+ethanol (not diluted, it is the original solution in which chlorophyll has been extracted from spinach in Fig. 2.5) of unknown concentration (impossible to estimate it as already explained in the dedicated section).
- 0.5 ml of curcumin+ethanol (0.1467 mol/L)
- 2.5 μ l (extracted with the micrometric pipette) of fluorescein+water (0.0054 mol/L)
- Diluted in + 3 more ml of ethanol (95%)

From this solution more have been created, diluting or adding more of a certain compound to detect the differences with other solutions. The solution in Fig 2.32 has been obtained.

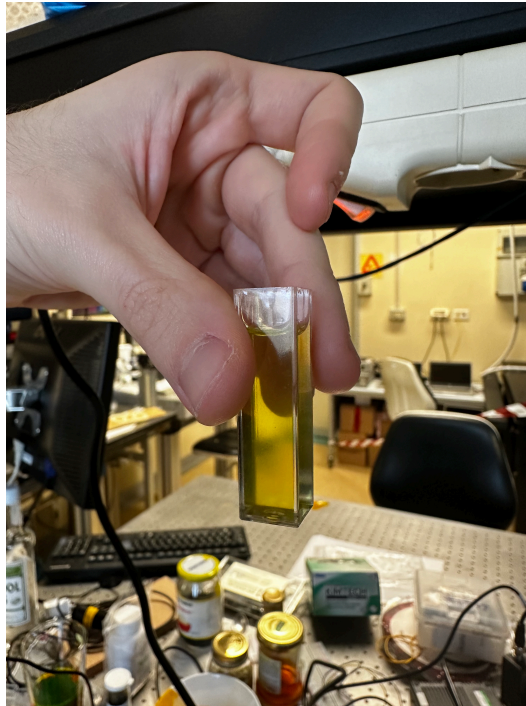


Fig 2.32: the solution obtained with the above recipe.

Setup

- **M450LP2 mounted LED (nominal wavelength 450 nm) with LED driver LEDD1B (Thorlabs) set at 1.2 A.**
- Avantes spectrometer (AvaSpec-3648) connected to a PC. Integration time and averages vary for each measurement, generally the number of averages has been set low and the integration time high to capture very low concentrations. The smoothing window also varies but is generally set to a minimum (1-10). The spectrums have been smoothed later with MATLAB.
- Standard Avantes fiber optic cable mounted on one of the lateral sides of the cuvette holder (not facing directly the LED).
- Plastic cuvette (3 ml).
- Cuvette holder.
- Millimetric and micrometric pipette.

Experimental results

The plots in Figs 2.33 and 2.34 show the visible fluorescence of curcumin and chlorophyll. Even at higher concentrations of fluorescein though it is not possible to detect the presence of fluorescein since the spectrum of curcumin has almost the same emission peak wavelength of fluorescein and has a larger FWHM (almost double that of fluorescein). In Fig. 2.35 a plot of fluorescein and curcumin overlapped clearly shows this. **What one can learn from this experiment is that it is possible to distinguish curcumin and fluorescein by shape (they differ a little bit, fluorescein seems to have a very softly pronounced secondary peak around 590 nm) and especially by FWHM. It is not possible to detect fluorescein if curcumin is present at a considerable concentration (taking into account that fluorescein's quantum yield is 0.97 and curcumin's is 0.067, so curcumin as in this experiment needs to have a very high concentration in the solution compared to fluorescein).**

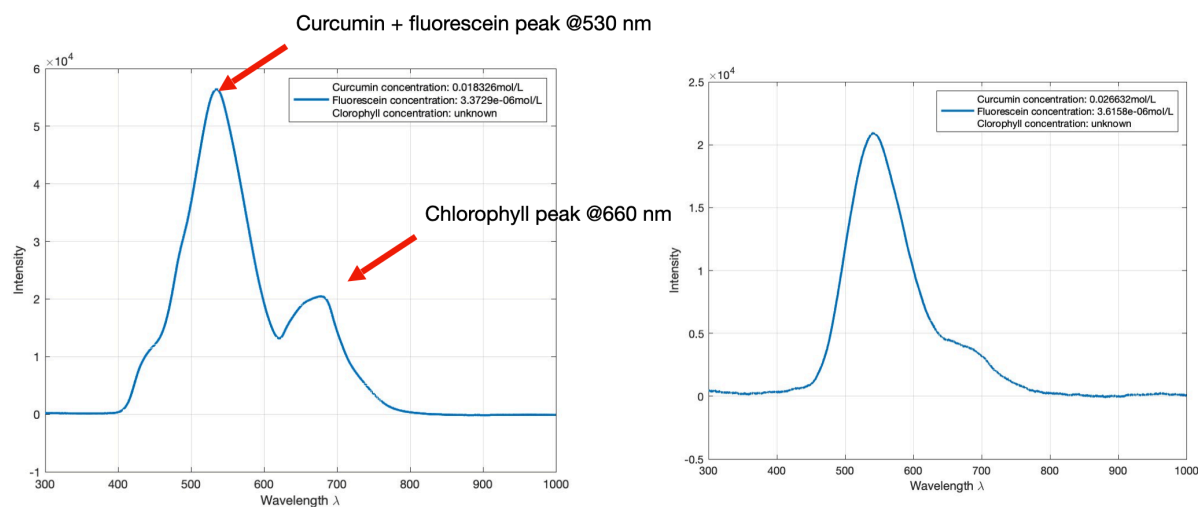


Fig 2.33: Curcumin + fluorescein + chlorophyll solution at different concentrations (Left plot: Curcumin concentration: 0.018 mol/L, Fluorescein concentration: $3.337 \cdot 10^{-6}$ mol/L, Chlorophyll concentration: unknown. Right plot: Curcumin concentration: 0.027 mol/L, Fluorescein concentration: $3.616 \cdot 10^{-6}$ mol/L, Chlorophyll concentration: unknown).

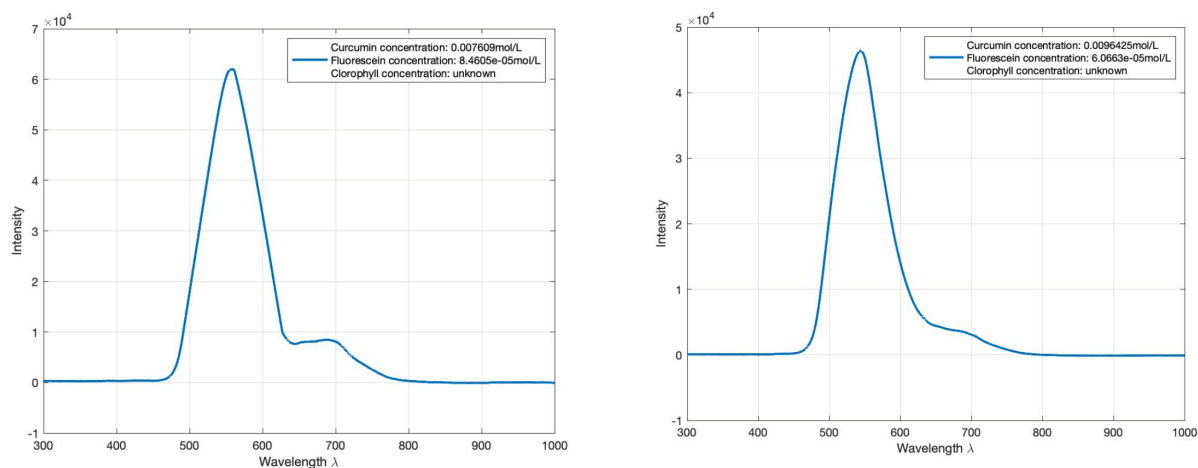


Fig 2.34: Curcumin + fluorescein + chlorophyll solution at different concentrations (Left plot: Curcumin concentration: 0.008 mol/L, Fluorescein concentration: $8.4605 \cdot 10^{-5}$ mol/L, Chlorophyll concentration: unknown. Right plot: Curcumin concentration: 0.010 mol/L, Fluorescein concentration: $6.066 \cdot 10^{-5}$ mol/L, Chlorophyll concentration: unknown).

Common Compounds Spectra Viewer

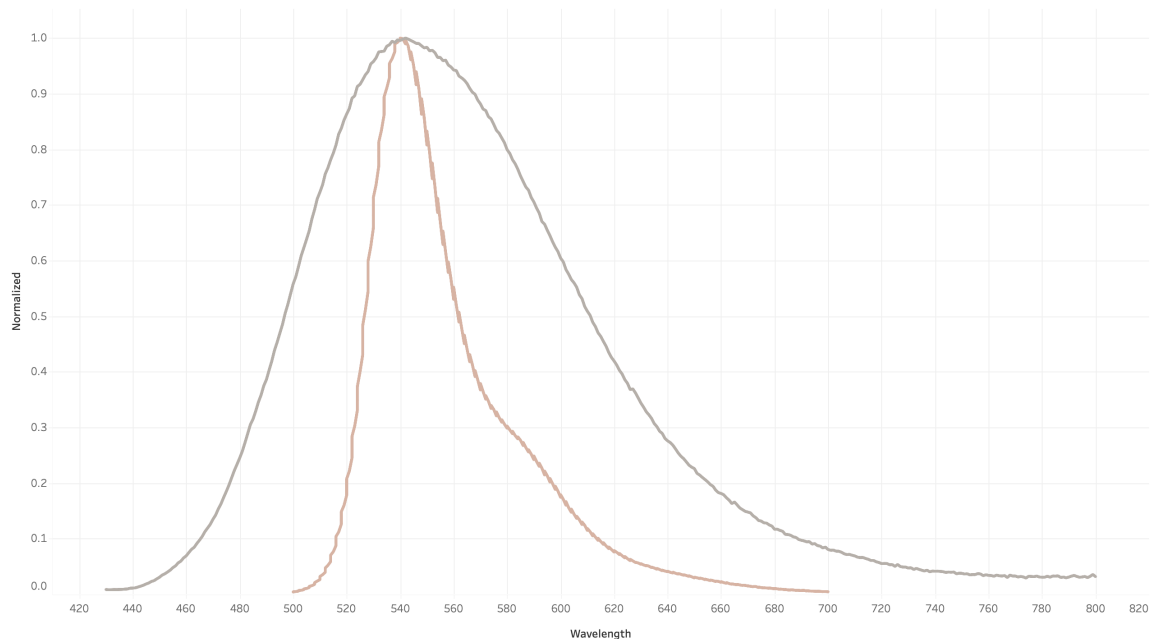


Fig 2.35: Curcumin emission spectrum (dark brown) and fluorescein's (light brown) overlapped. It is clearly visible that curcumin's spectrum overlaps by large (almost double) in width that of fluorescein. Courtesy of PhotoChemCad.

Experiment 5: pH experiments

Goals

The goal of the experiments in this section is to quantify the influence of solvents with different pH (ranging from 2 to 12) on the fluorescence properties of fluorescein and curcumin (analysed so far).

Preparation of the solutions

To test the effect of pH on fluorescence the following fluorophores with different solvents have been employed:

Fluorescein with solvents:

- HCl pH 2, HCl pH 4, NaOH pH 8, NaOH pH 10

Curcumin with solvents:

- HCl pH 2, HCl pH 4, NaOH pH 8, NaOH pH 10

Both HCl and NaOH are polar solvents, they are not as polar as water so they can mix with curcumin well enough.

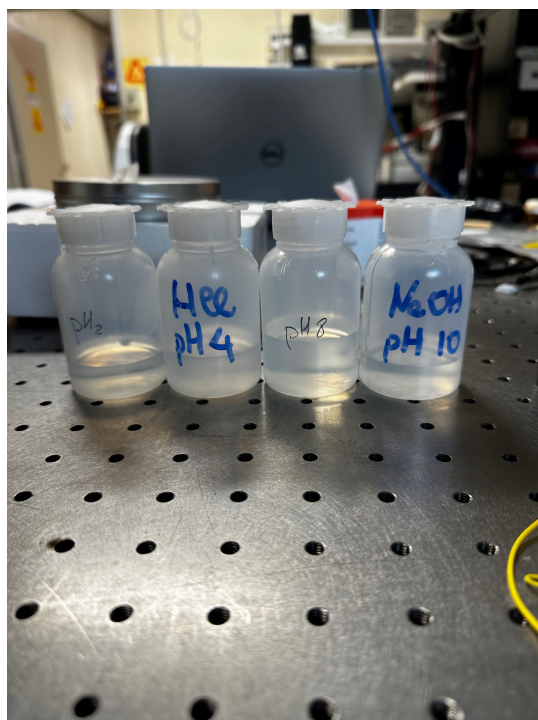


Fig 2.35: The different solvents.

The solvents have been provided by PoliTo's Department of Applied Science and Technology (DISAT).

The solutions have been prepared with the same procedures as the previous experiments, small samples of fluorescein and curcumin (separately) have been collected with the aid of millimetric and micrometric pipettes from solutions of curcumin+ethanol and fluorescein+water already prepared in the previous experiments and mixed with the solvents above described in plastic cuvettes.

For each solvent (pH 2, pH 4, pH 8, pH 10) have been prepared 3 solutions of fluorescein with different concentrations ($1.255 \cdot 10^{-5}$ mol/L, $1.509 \cdot 10^{-6}$ mol/L, $8.2104 \cdot 10^{-7}$ mol/L), it has been done so to compare solutions of different pH taking into account also possible shaping/quenching effects of concentration.

For curcumin the same has been done but with concentrations: 0.06 mol/L, 0.0087 mol/L, 0.00124 mol/L. Larger concentrations than fluorescein have been used since curcumin has a way lower quantum yield (100 times lower) than fluorescein.

Setup

- **M450LP2 mounted LED (nominal wavelength 450 nm) with LED driver LEDD1B (Thorlabs) set at 1.2 A.**
- Avantes spectrometer (AvaSpec-3648) connected to a PC. Integration time and averages vary for each measurement, generally the number of averages has been set low and the integration time high to capture very low concentrations. The smoothing window also varies but is generally set to a minimum (1-10). The spectrums have been smoothed later with MATLAB.
- Standard Avantes fiber optic cable mounted on one of the lateral sides of the cuvette holder (not facing directly the LED).
- Plastic cuvette (3 ml).
- Cuvette holder.
- Millimetric and micrometric pipette.

- The solvents already described (Fig. 2.35)

Experimental results for fluorescein

In Figs 2.36, 2.37, 2.38, and 2.39 are reported the plots obtained from the fluorescent solutions of fluorescein and the solvents used, The concentrations of the solutions are reported on the plots in the Figures.

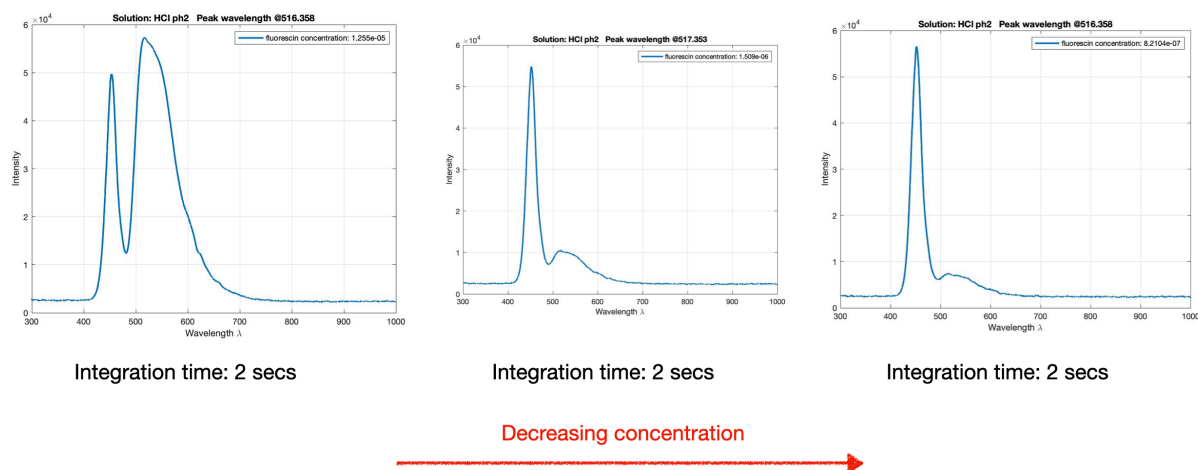


Fig 2.36: Fluorescein's measurements with HCl pH 2 solvent and changing concentration, peaks' wavelengths are indicated on the top of each plot.

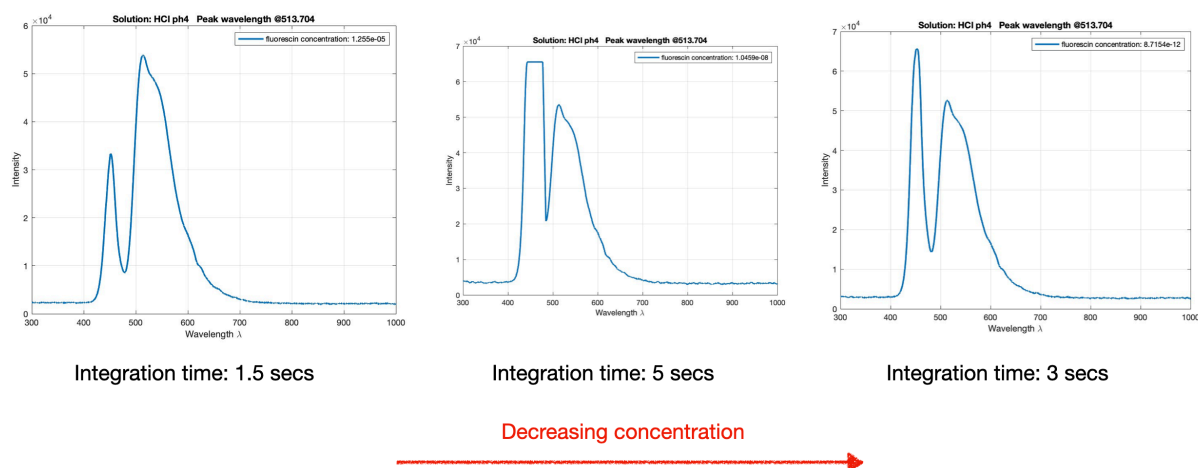


Fig 2.37: Fluorescein's measurements with HCl pH 4 solvent and changing concentration, peaks' wavelengths are indicated on the top of each plot.

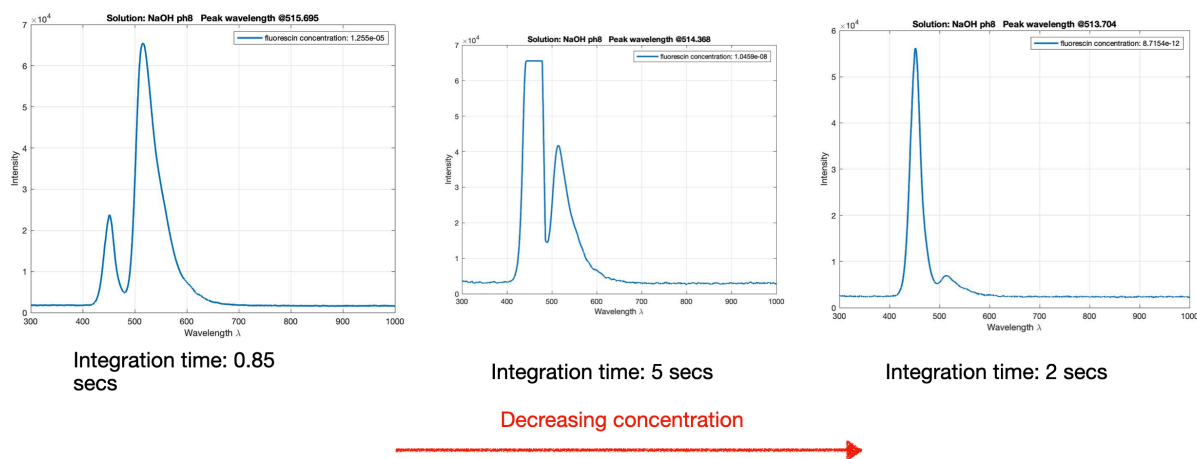


Fig 2.38: Fluorescein's measurements with NaOH pH 8 solvent and changing concentration, peaks' wavelengths are indicated on the top of each plot.

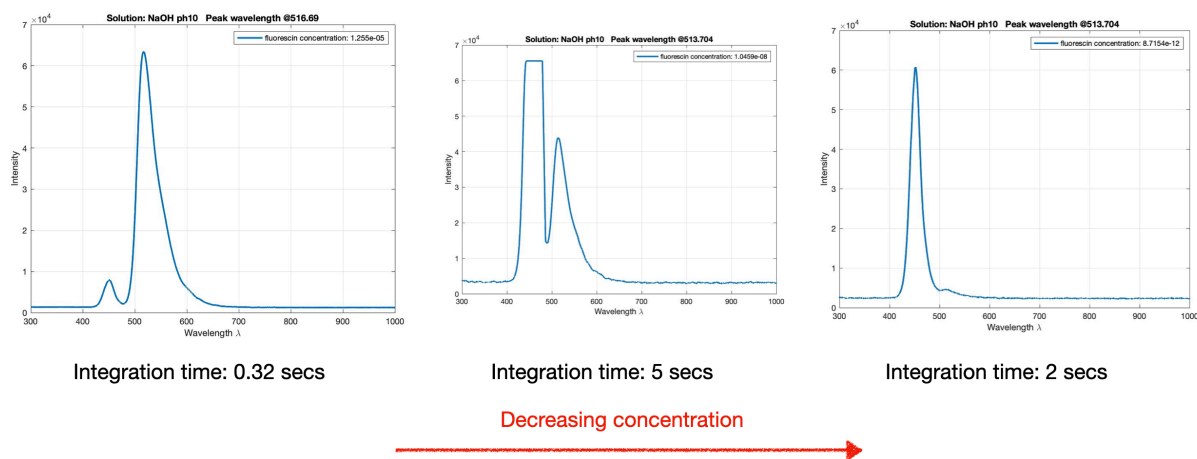


Fig 2.39: Fluorescein's measurements with NaOH pH 10 solvent and changing concentration.

Looking at the effects of solutions with the same concentration two phenomena can be observed (Fig. 2.40):

- Quenching of fluorescence at lower pH
- Broadening of the peak's shape (for the most acid solutions) and also a different shape of the spectrum.

The more acid the solvent (pH 2) the less it seems to react with the solvent. In other words, acid solvents seem to quench fluorescein's fluorescence. Also, the choice of the solvent used (HCl or NaOH) doesn't seem to be an influencing factor, since the solution shows a broadening

and quenching of the spectrum in the same way regardless of the solvent used (Fig. 2.41).

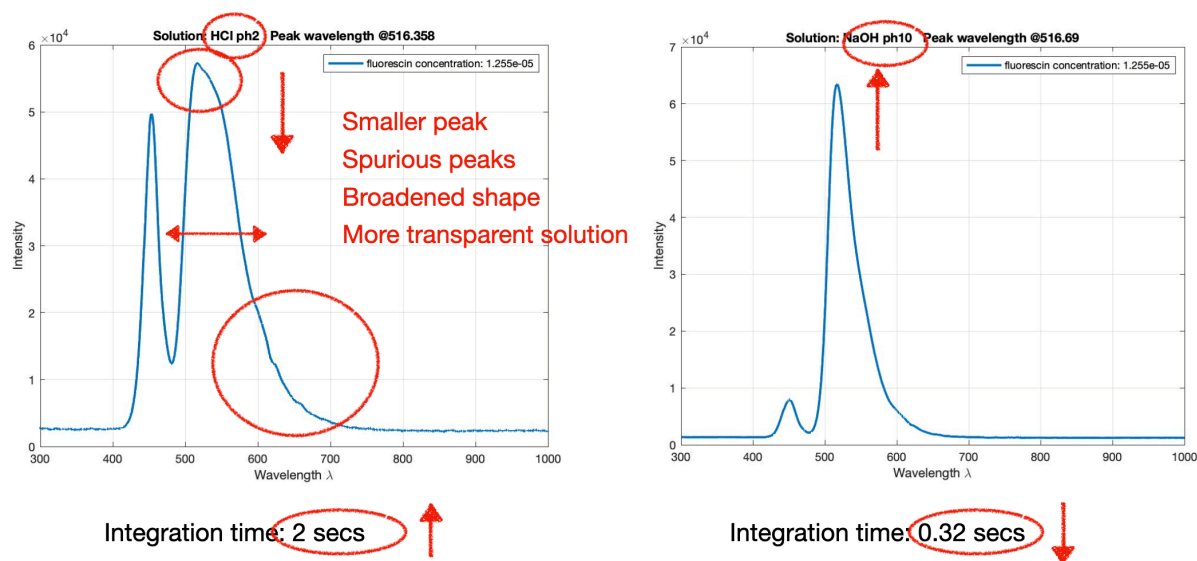


Fig 2.40: Fluorescein at different pH (the two ends of the scale: pH 2 and pH 10 solvents) at the same concentration, on the picture the effects of pH are highlighted. **Note that different integration times have been used**, so the intensity of two different spectrums acquired with different integration time should be normalised with respect of the integration time like in picture 2.41.

Why does fluorescein react this way? The reason is that fluorescein modifies its ionisation state at different pHs (Fig. 1.11). The absorption and emission spectra are directly correlated to the ionisation state of the molecule and the ionisation state of the molecule indeed modifies the electronic structure of it. This means that the more a compound becomes ionised, the more it will fluoresce because more donor groups (OH-) are available. **Aromatic rings and donor groups in a molecule are the elements that contribute the most to fluorescence.**

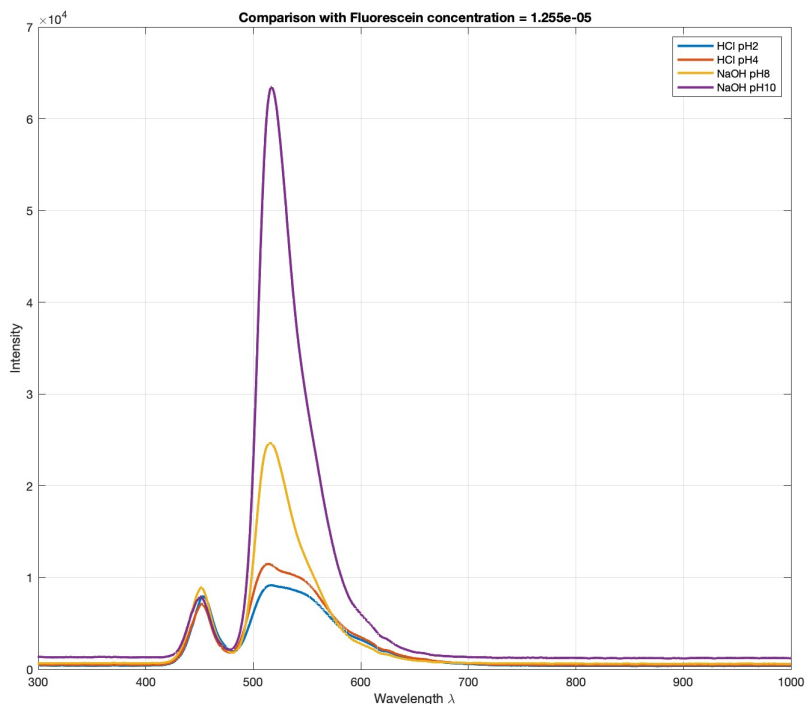


Fig 2.41: Comparison of the different spectrums obtained for the different pHs, here each spectrum has been normalised with respect to the integration time used for the acquisition of the relative spectrum.

So a change in pH in solutions with fluorescein quenches the intensity (at lower pHs, so at more acidic solutions), and as a consequence reduces its quantum yield (Fig. 2.42 and 2.43). It also changes the spectrum of fluorescein, making the peak FWHM more narrow for basic solutions and broader for acidic solutions. What pH does not influence the fluorescein's fluorescence is the peak wavelength (pH 2.44) as opposed to curcumin, for which the effects will be seen in the "Experimental results for curcumin" section next in this chapter.

- pH 10: $6.34 \cdot 10^4$ (relative intensity)
- pH 8: $2.4651e4 \cdot 10^4$
- pH 4: $1.1492e4 \cdot 10^4$
- pH 2: $0.9172e4 \cdot 10^4$

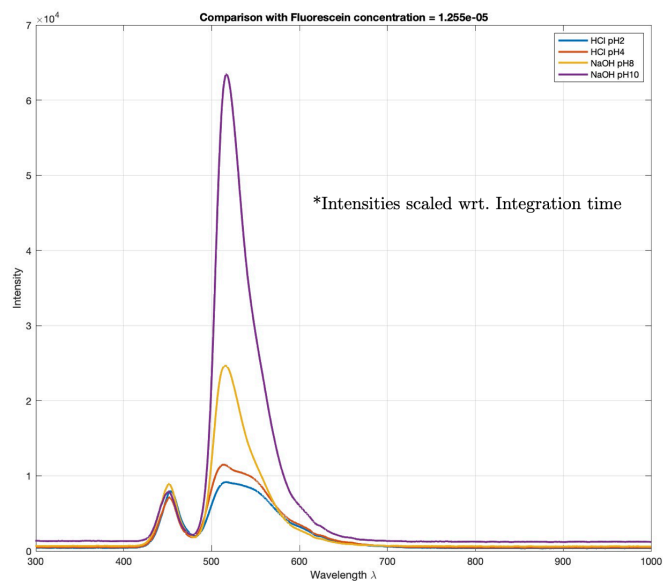
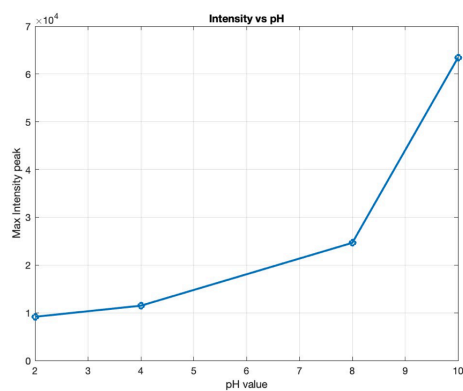


Fig 2.42: pH vs intensity at the same concentration of fluorescein in the solutions.

- The dianion has the most intense fluorescence with a quantum yield of **0.93**.
- Anion: 0.37
- Neutral: 0.30
- Cation: 0.18

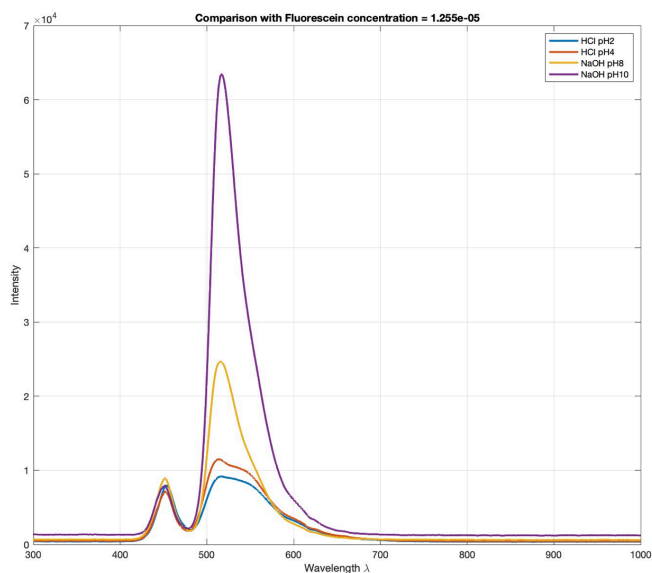
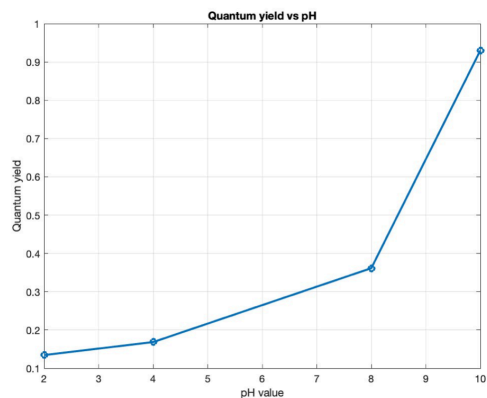


Fig 2.43: pH vs quantum yield (of fluorescein's fluorescence) at the same concentration of fluorescein in the solutions.

- pH changes do not influence the peak wavelength of the compound

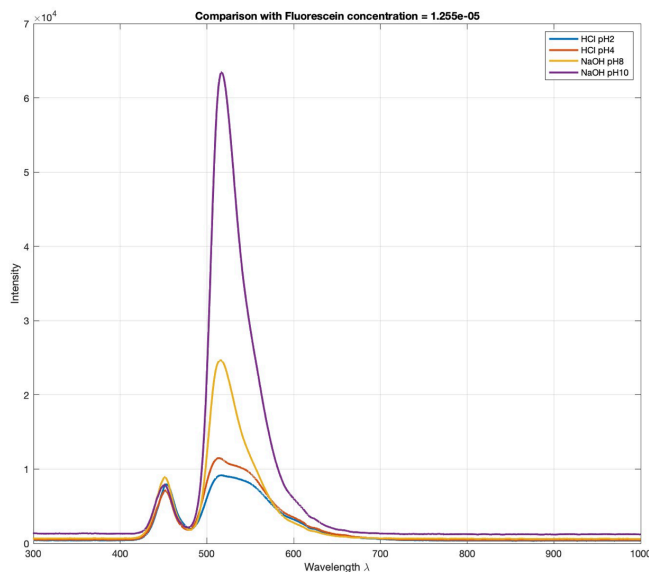
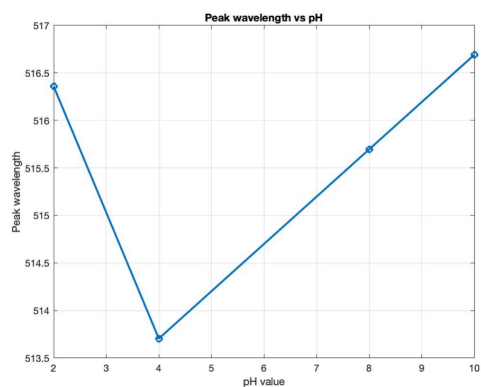
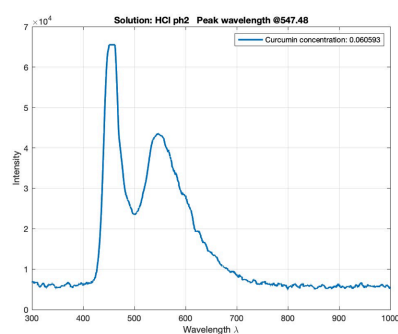


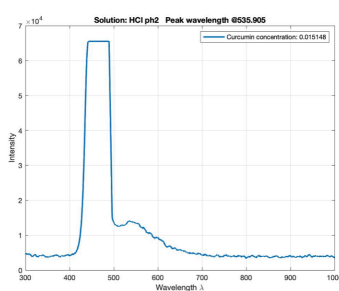
Fig 2.44: pH vs peak wavelength at the same concentration of fluorescein in the solutions.

Experimental results for curcumin

In Figures 2.45, 2.46, 2.47, and 2.48 are reported the plots obtained from the fluorescent solutions of fluorescein and the solvents used. The concentrations of the solutions are reported on the plots in the Figures.



Integration time: 11 secs



Integration time: 6 secs

Decreasing concentration



Fig 2.45: Curcumin's measurements with HCl pH2 solvent and changing concentration, the last concentration on the right (0.0012366 mol/L) is missing because its intensity was so quenched for pH 2 that it was not visible with the spectrometer used, not even increasing the integration time.

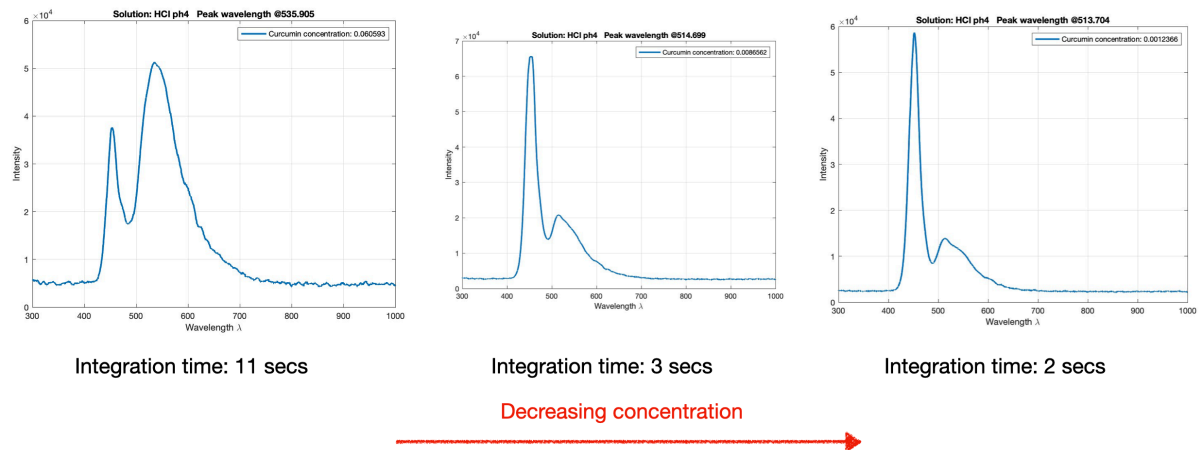


Fig 2.46: Curcumin's measurements with HCl pH4 solvent and changing concentration.

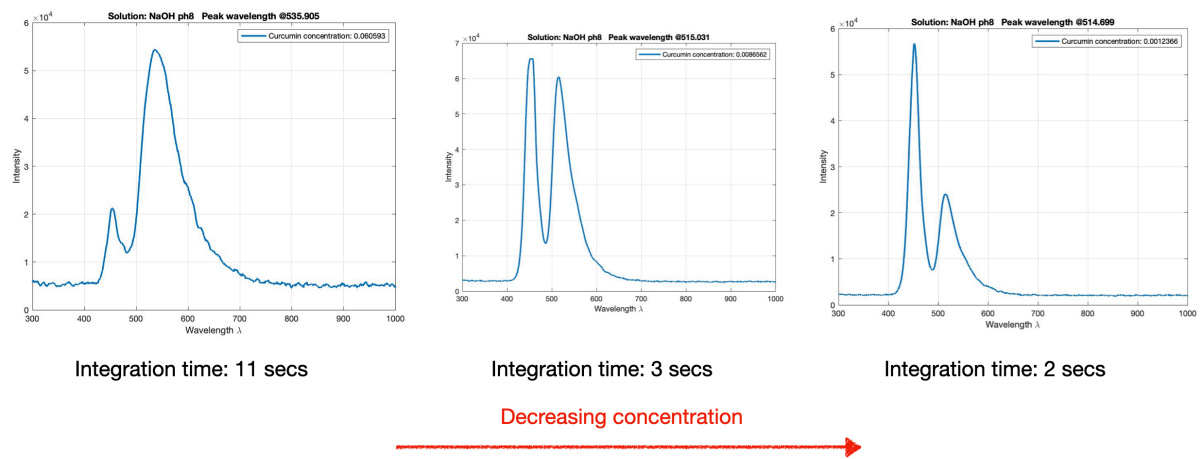


Fig 2.47: Curcumin's measurements with NaOH pH8 solvent and changing concentration.

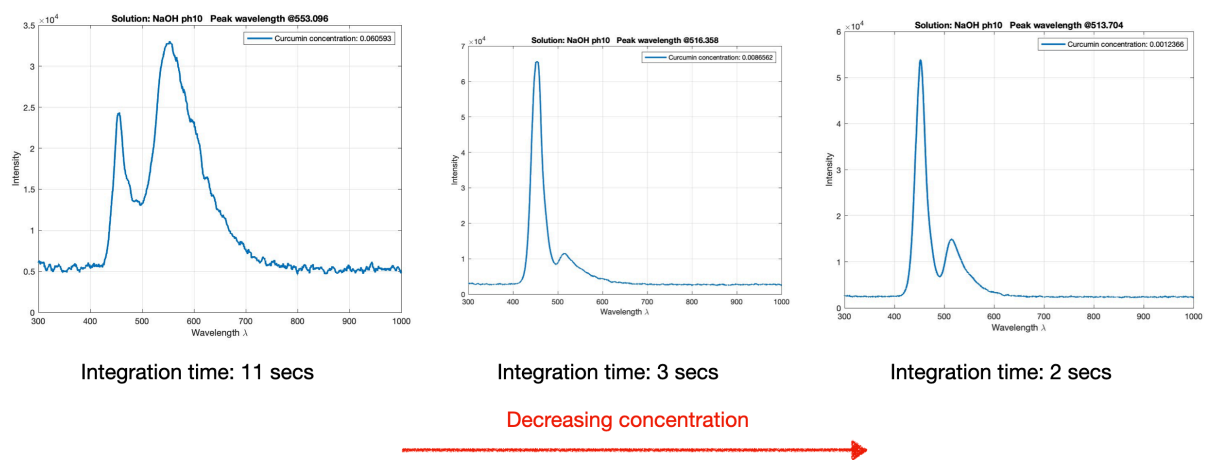


Fig 2.48: Curcumin's measurements with NaOH pH10 solvent and changing concentration.

Observing the spectrums at the same concentration of curcumin one can observe two effects of pH (Fig. 2.49).

- Like in fluorescein's experiments, pH (lower pH, i.e. more acidic solutions) quenches the fluorescence of curcumin also.
- With curcumin pH does not seem to change the spectrum's shape like with fluorescein.

A third effect observed, which is a red-shift of the peak's emission wavelength well-documented in other scientific papers [20] was not observed in this experiment, the causes could be many (small variations of concentration from the expected during the preparation of the solutions due to human error, the solvent and curcumin in some solutions could have not mixed so well). The fact that curcumin's emission spectrum's shape does not change with pH and that the emission peak red-shifts (though not observed) compared with the different effects seen on fluorescein serve as another proof of the fact that pH influences the fluorescent properties of different compounds in a different way.

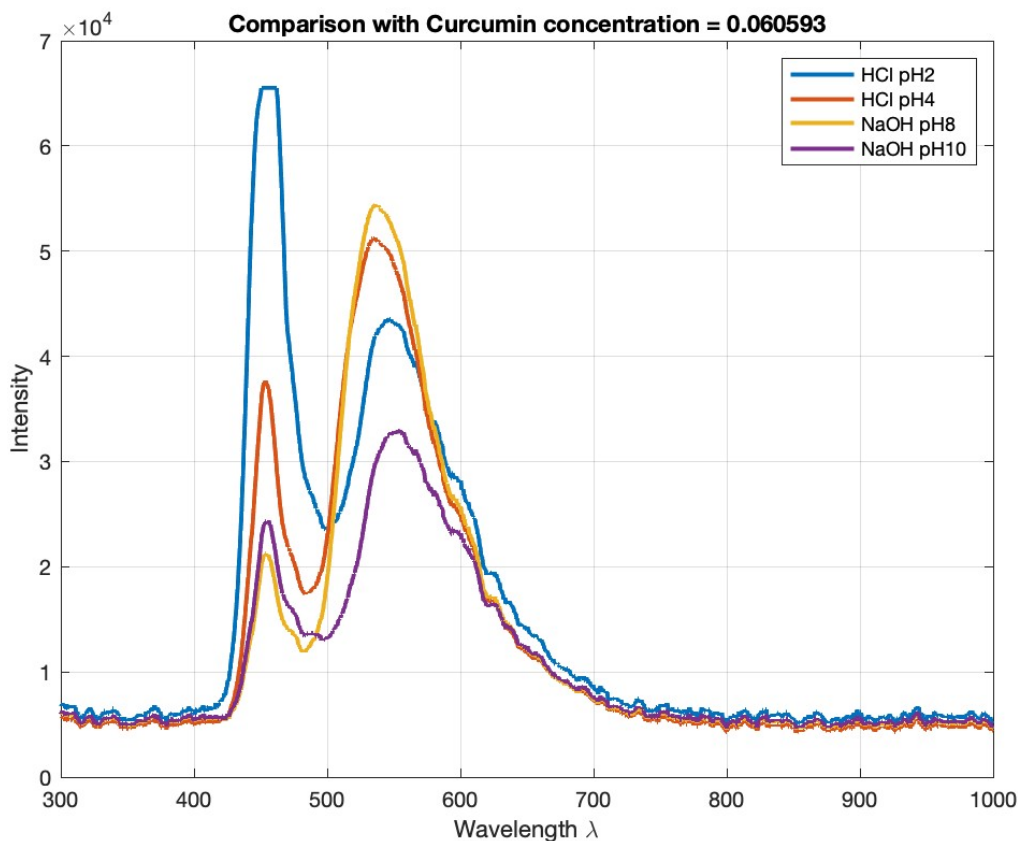


Fig 2.49: Comparison of the different spectrums obtained for the different pHs at the same concentration of curcumin. The red-shift at higher pH and quenching of fluorescence at lower pH is clearly visible. pH contrary to fluorescein does not modify curcumin's emission spectrum.

Experiment 6: Tyrosine and Tryptophan from dairy milk solution

Goals

The main goal of this lab experiment was to characterise tyrosine and tryptophan, two fluorescent proteins that can be found in a lot of different bacteria (and in this case from milk bacteria) to demonstrate of it is possible to detect with spectrofluorometry a possible contamination of bacteria of the substance investigated.

What are Tyrosine and Tryptophan

Tyrosine, tryptophan, and phenylalanine are the three aromatic amino-acids responsible for protein synthesis in most living beings, including bacteria. While these amino acids are part of the same family, they exhibit variations in polarity, hydrophobicity, and internal structures. All spectrofluorometry methods to detect bacteria have been so far carried looking for these proteins as biomarkers [20]. Tyrosine and tryptophan are more interesting than phenylalanine because the two have a way higher quantum yield than the former (0.13 and 0.12 vs 0.022, source: PhotoChemCad) making the detection of tyrosine and tryptophan a more interesting option in this application.

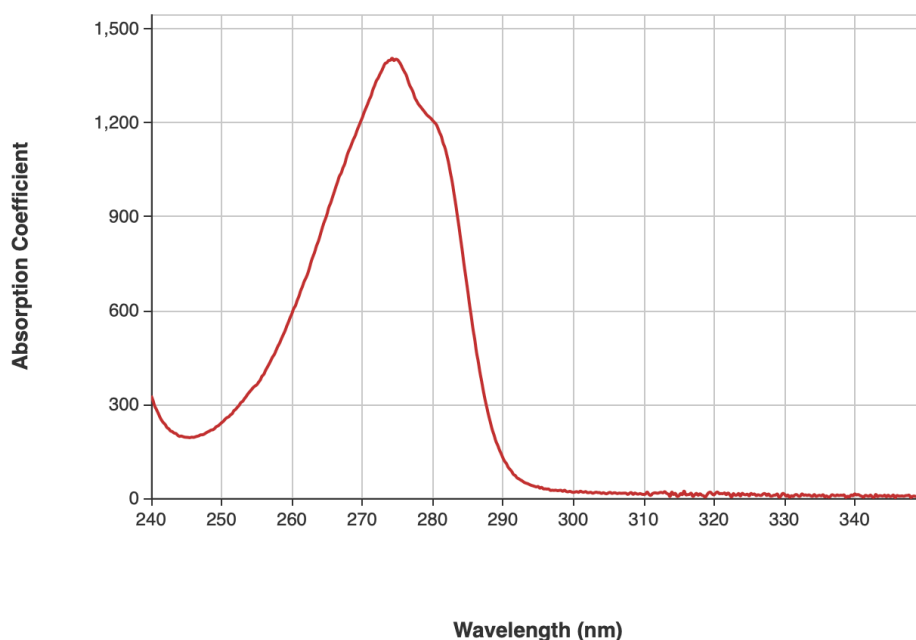


Fig 2.50: absorption spectrum of **Tyrosine**, source: PhotoChemCad [29].

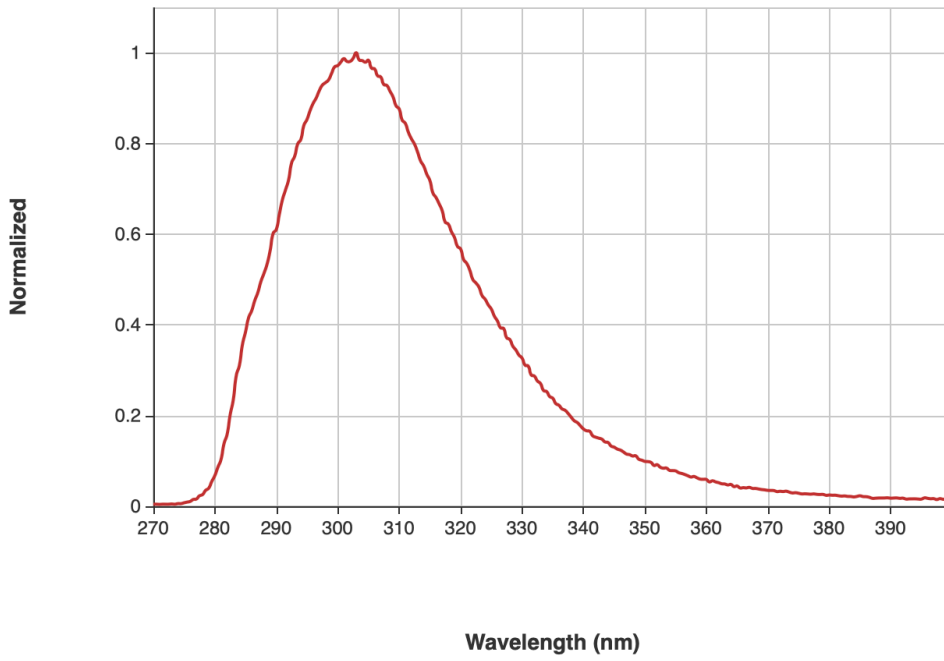


Fig 2.51: emission spectrum of **Tyrosine**, source: PhotoChemCad [29].

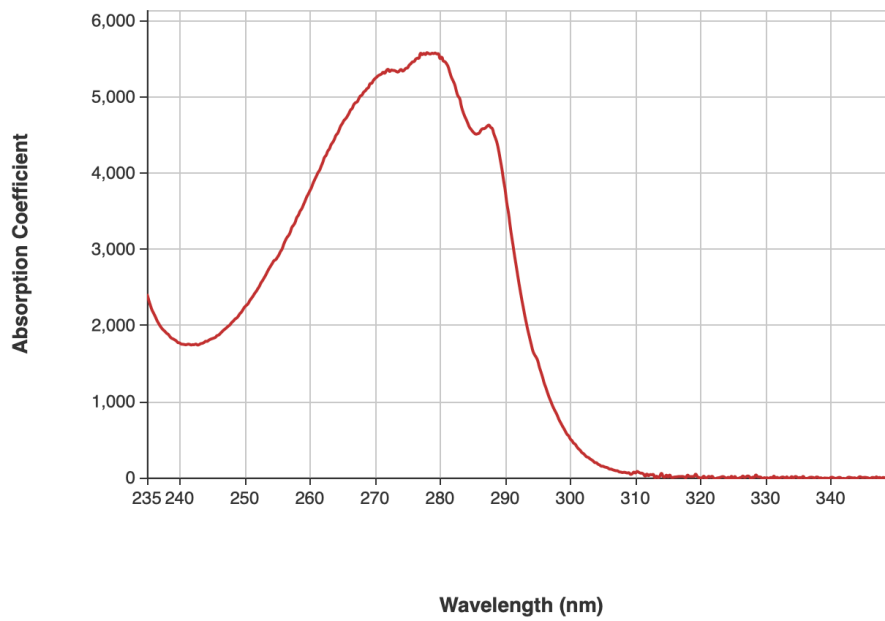


Fig 2.52: absorption spectrum of **Tryptophan**, source: PhotoChemCad [28].

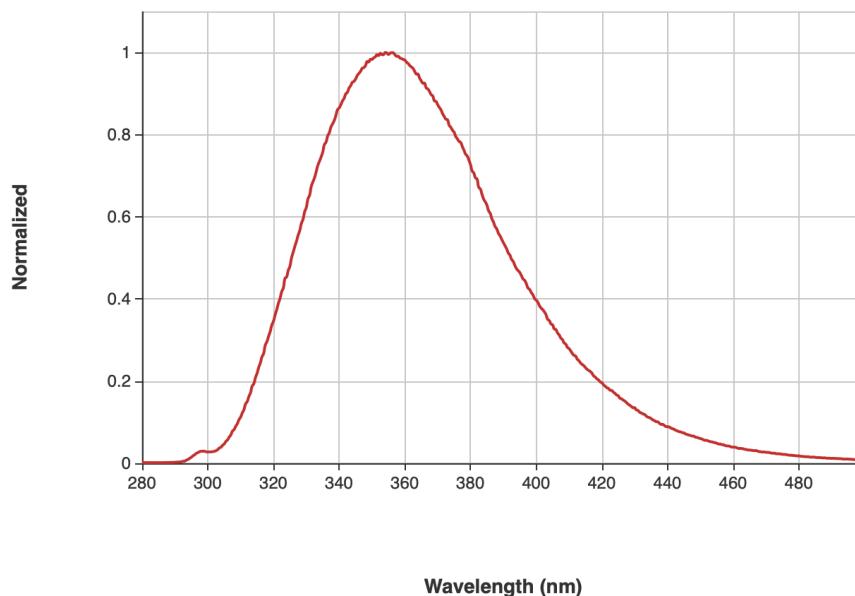


Fig 2.53: emission spectrum of **Tryptophan**, source: PhotoChemCad [28].

Preparation of the solutions

The solutions have been prepared with the same procedures of the previous experiments, small samples of dairy milk have been extracted with the aid of millimetric and micrometric pipettes and mixed with water. As has been said previously in milk among other fluorescent substances tyrosine and tryptophan are present, and their detection is the goal of this experiment.

There has been an attempt to quantify the concentrations of milk in this experiment but milk does not mix well with water or ethanol and it is highly opaque to light so this attempt has been abandoned in favor of simply observing the fluorescence of milk in a small concentration diluted in water.

Setup

Since tyrosine's and tryptophan's absorption spectrums stop at 290 nm (Fig. 2.50 and 2.52) the M450LP2 LED could not be used (because its nominal wavelength is 450 and its spectrum is relatively narrow compared to this application) another LED has been employed.

The LED employed is the LA KH120U8F, which has a nominal wavelength of 275 nm and its optical power is 100 mW. The LED is embedded in a custom integrated circuit (Fig. 2.54) that was used for sanitization purposes, which was the topic of another thesis work in PoliTO's PhotoNext lab.

- **LA KH120U8F LED, nominal wavelength: 275 nm, optical power: 100 mW.**
- Avantes spectrometer (AvaSpec-3648) connected to a PC. Integration time and averages vary for each measurement, generally the number of averages has been set low and the integration time high to capture very low concentrations. The smoothing window also varies but is generally set to a minimum (1-10). The spectrums have been smoothed later with MATLAB.
- Standard Avantes fiber optic cable mounted on one of the lateral sides of the cuvette holder (not facing directly the LED).
- Plastic cuvette (3 ml).
- Cuvette holder.
- Millimetric and micrometric pipette.
- Black box.
- Scale (sensitivity 0.1 g), was used in the initial effort to quantify the milk being measured.

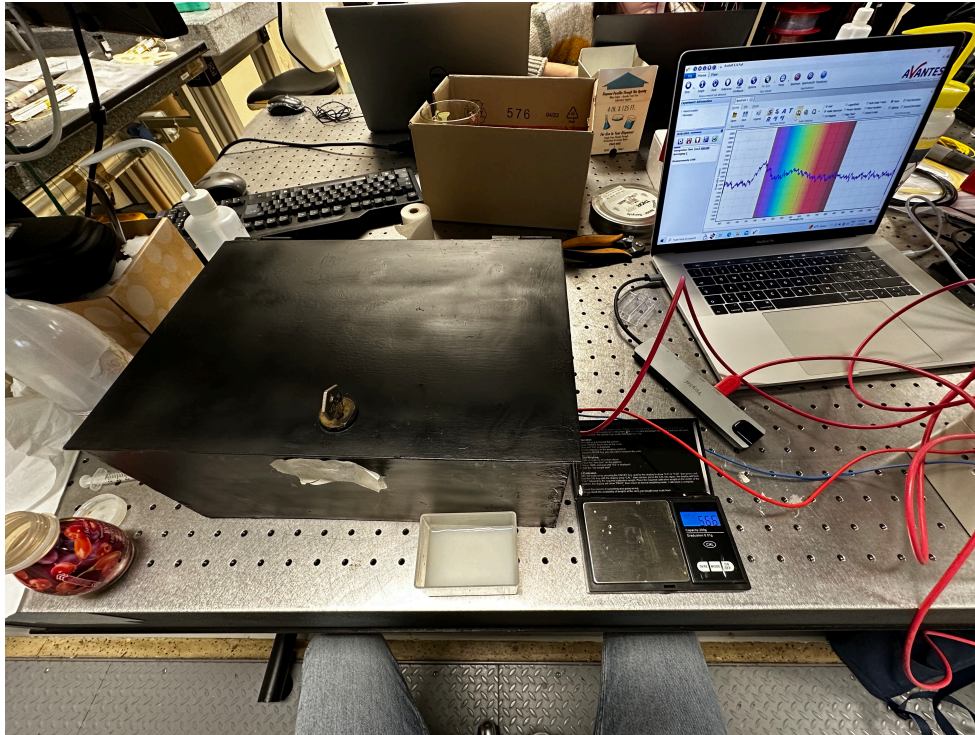


Fig 2.54: the setup, most of it is inside the black box on the left.

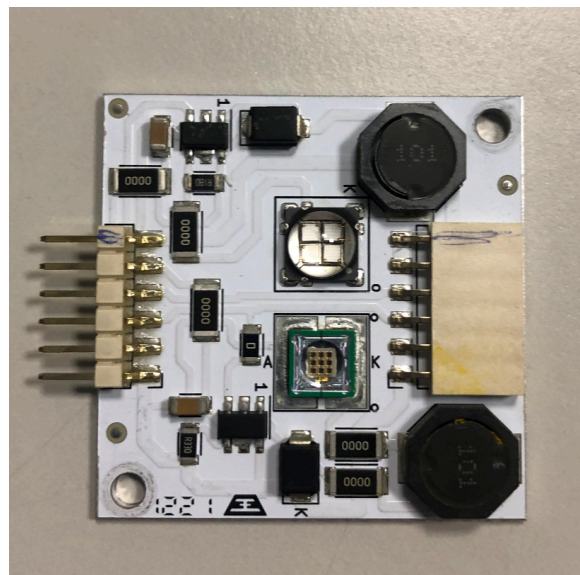


Fig 2.55: LA KH120U8F LED embedded in a custom integrated circuit.

Experimental results

In Figures 2.56, 2.57, and 2.57 are reported the plots obtained from the fluorescent solutions of dairy milk mixed with water and the solvents used, The concentrations of the solutions are not reported since it was not possible to estimate them nor in the interest of this experiment estimating them.

It is possible to observe on the plots both tyrosine's and tryptophan's emission peaks as one (the two's emission peak's distance is only approximately 50 nm so the two appear together as only one peak on the spectrometer) and riboflavin's emission peak at approximately 540 nm. Riboflavin is a vitamin that is fluorescent like tyrosine and tryptophan, it is sometimes used as a biomarker [21] and has a relatively high quantum yield (0.3 in ethanol) compared to tyrosine's and tryptophan's (0.12 and 0.13).

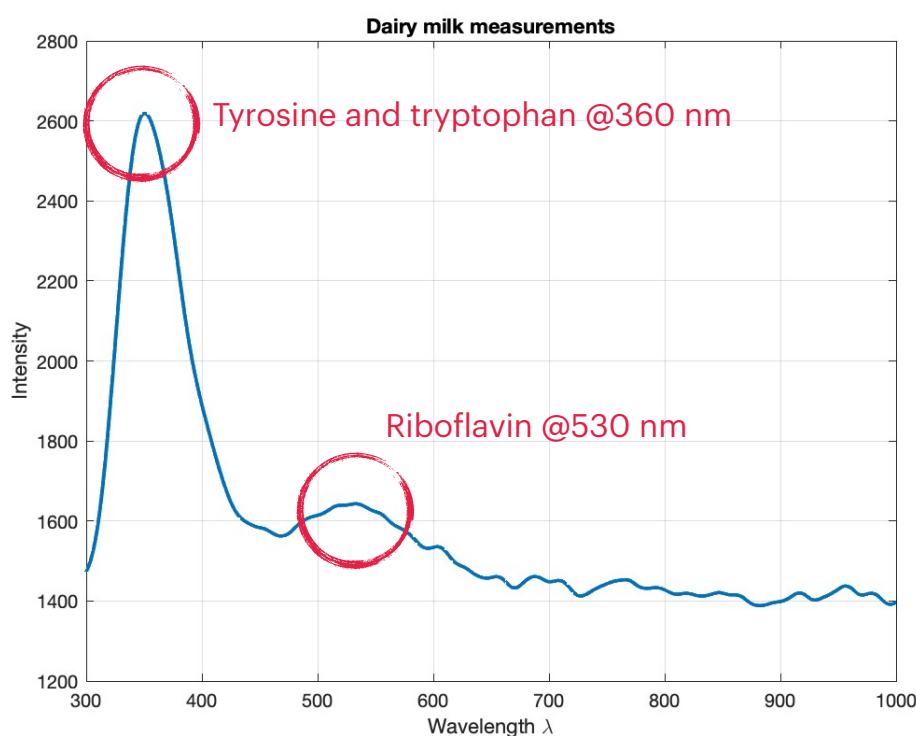


Fig 2.56: Fluorescence of dairy milk, the peak on the left represents both tryptophan and tyrosine (their peaks are very close). The first peak on the right of the previous is riboflavin, which is also a fluorescent compound.

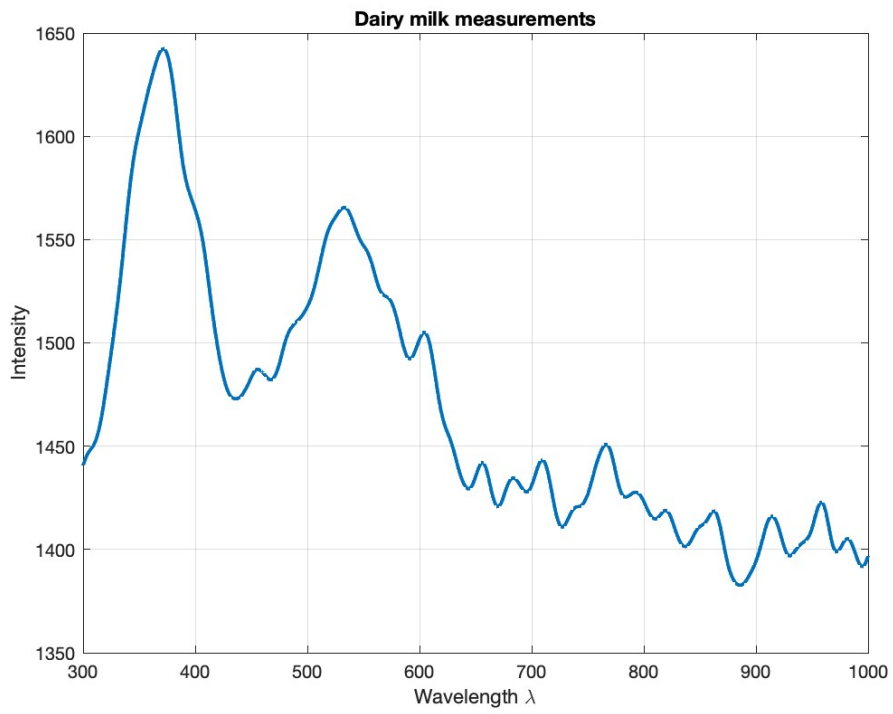


Fig 2.57: Fluorescence of dairy milk, tyrosine, tryptophan and riboflavin in one plot.

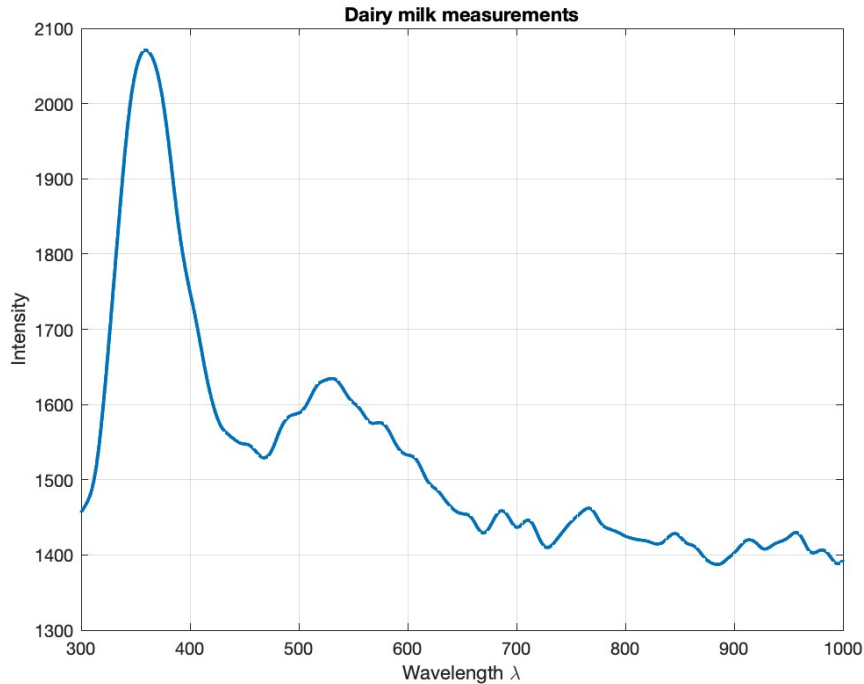


Fig 2.58: Fluorescence of dairy milk Fluorescence of dairy milk, tyrosine, tryptophan and riboflavin in one plot.

Final conclusions on all the experimental results obtained

All the experimental work done in the laboratory was to investigate what influences fluorescence, and to characterise how much each factor influences it (temperature, pH, solvent choice, excitation source). The goal in mind behind this experiment has always been from the beginning of this thesis work to create a tool for recognizing a compound (possibly a hazardous one for the environment and/or humans, such as bacteria like *Escherichia Coli*) from the possible fluorescence of some components of the compound. Creating such a tool is of uttermost importance to understanding how fluorescence works and what influences it.

The conclusion on this experiments is that many factors influence fluorescence, and each compound's fluorescence is unique in the sense that each compound's fluorescence is influenced by different factors than

others. Some compounds are more temperature influenced, and some are more pH influenced depending on their ionisation state (see curcumin and fluorescein) and react differently to changes of pH (curcumin red-shifted its spectrum to higher pHs, fluorescein broadened it to lower pHs); Each solution's fluorescence changes dramatically based on the solvent used, the polarity of each component has to be taken into account. Moreover, reabsorption effects manifest themselves when the solution has a high concentration of the solute, red-shifting the peak emission of the fluorophore to 50 nm or more with respect to less concentrated solutions.

All of this means that a possible tool has to take into account all of these factors and possibly more if the goal of the said fluorescence recognizing tool is to recognize hundreds of possible compounds that could react very differently from one another to the said factors.

For this reason, the best approach to solve this problem might be to use a data-driven approach, meaning that the scientist/user that wants to analyse compounds based on their fluorescence can exploit a possibly huge database of known fluorescent compounds, of which their spectrums have been acquired in different conditions:

- By concentration
- By pH
- By temperature
- By solvent
- And possibly more factors that influence it

And this tool to recognise a compound from its fluorescence should recognise key elements of the spectrums such as:

- Peak(s) wavelength(s)
- FWHM(s) of the peak(s)
- Shape of the spectrum (each normalised (from 0 to 1) intensity at k wavelength)

With a comparison algorithm, the tool should assign a compatibility score between the unknown spectrum inputted by the user/scientist and those present in the said database.

In this way, the problem of taking into account all the different factors investigated before and taking into account and also how they affect each compound differently is shifted to simply collecting emission spectrums in the said different conditions and letting the algorithm do the job of recognising the compound.

The tool briefly mentioned earlier is thoroughly introduced and explained in the following chapter (“**Chapter 3: FluoDetect**”).

Chapter 3: FluoDetect

Introduction

A short overview of the motivations, technology, and capabilities of FluoDetect

FluoDetect is one of the results of the thesis work carried out by the author of this thesis. From the experiments a need for a more comprehensive and easy-to-use application for studies in fluorescence spectroscopy emerged. Most of the applications available to this day for fluorescence analysis have huge databases filled with fluorescent elements of any kind, from those present in organic compounds such as bacteria (such as tyrosine or tryptophan) to inorganic compounds. These applications mainly allow the user to search for fluorescent elements by name and to compare those by means of graphic plots that overlap the spectra of said fluorescent elements.

What these software **lack** though are some important tools/characteristics:

- Fluorescent **compounds**: Many fluorescent elements when mixed in a compound change dramatically their spectrum's shape, peak wavelength, and FWHM to the point where it is difficult to distinguish the single fluorescent elements spectrums even for an AI algorithm.
- Search for a fluorescent compound/element by peak wavelength and FWHM, in today's software since the user can only look up fluorescent records in the database by name (such as "tyrosine" or "fluorescein") the user has to know what to look for.
- **Compare an unknown spectrum obtained by the user**, for example by means of a spectrum analyser, to those present in the database via a comparison algorithm. Again the user/scientist has to know what to look for.

A scientist in the field of fluorescence spectroscopy knows that a fluorescence compound's spectrum can change dramatically based on environmental conditions such as solvent nature, pH of the solution, temperature, concentration of the solution, excitation wavelength of the light source, and presence of fluorescent elements that interact with each other (both chemically and optically by overlapping of the spectrums).

FluoDetect allows the user/scientist to create databases of fluorescent compounds/elements based on the user's needs, to look for records in the database by means of peak wavelength and FWHM, and to compare an unknown obtained spectrum with those present in the database by a customisable (by search parameters) comparison algorithm and displaying the results in an intuitive UI by means of graphic plots and tables.

The **primary goal** was to create this software for online in-situ detection of dangerous bacteria in drinkable water by means of fluorescence spectroscopy, but the software is customisable to the point where it can be used for more general-purpose applications also.

The **secondary goal** was (and still is) to create this software for a **data-driven approach** to the problem of fluorescence analysis, meaning that the more data (spectrums) in the database, the more reliable the successful recognition of an unknown fluorescent compound by means of spectrum comparison with those already in the database. **Too many factors (most of which have been already cited) influence the emission spectrum of a fluorescent compound, so an AI cannot still with satisfactory results recognise the fluorescent elements in a compound without human intervention still. But a human with enough tools and data can certainly (or more easily) recognise the fluorescent elements in a compound and aid (train) a computer in doing so.**

FluoDetect is a web application coded in Python, with Flask framework used for full-stack development. The application handles data by means of a SQLite database containing all the registered fluorescent records' main characteristics (name, peak wavelength, FWHM) and uses .txt files for their spectrum. The text file data (containing the emission wavelengths and their intensities of the fluorescent record) are handled using Pandas data structures and NumPy arrays. Plots are

displayed with the help of the Plotly library for Python. OpenAI's GPT-4 engine is used to generate descriptions of the fluorescent compound records' when a description is not provided by the user when inserting the record in a database. GPT-4 generates those descriptions based on the name given by the user to the record and the characteristics of the spectrum (peak wavelengths and FWHMs). It is to be noted that GPT-4 descriptions are not always reliable, especially if the user didn't provide a descriptive name for the fluorescent compound. This is also proof that an only AI approach to fluorescence spectroscopy is still not a reliable approach to recognise compounds in a spectrum.

Disclaimer (!)

The current version of FluoDetect is intended as a beta, but many minor bugs are still to be fixed, moreover, the applications still lack an account system (authenticator) to allow only certain users to add databases, and fluorescent records and delete them. To future good-willed and interested software developers are left with the task of carrying on the development of this application.

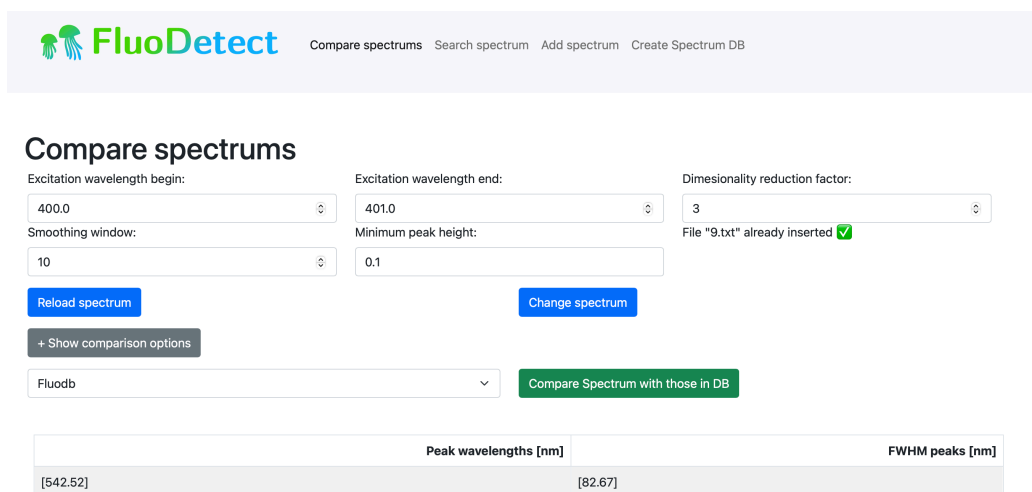


Fig. 3.1: Compare spectrums page.

Comparison Matches

Name	Peak wavelengths [nm]	FWHM peaks [nm]	Score [%]
Fluorescin	[515.03]	[40.44]	35.305564
Inserted spectra	[542.52]	[82.67]	35.305564

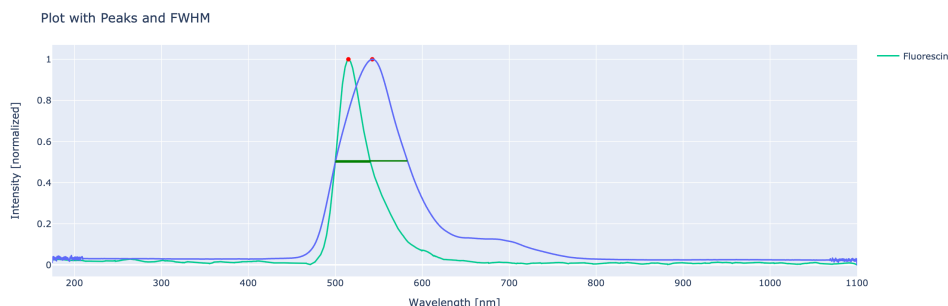


Fig 3.1.1: Compare spectrum results.

Creating a database

How to add a database

Creating databases is important in order to organise fluorescent records per the context of acquisition. As an example, a user could want to create a database for all the spectrum acquisitions of a fluorescent compound in different pH conditions of the solvent employed and does not want to mix these acquisitions with others for example of a fluorescent compound acquired with different temperatures but same pH. **Separating acquisitions in this way is important since when the user wants to compare an unknown acquired spectrum he will have to specify a database in which to do the comparison.**

Creating a database is a straightforward process:

1. Click on the navigation bar element **“Create Spectrum DB”**. A page with all the databases shown on a table (if there are any) (Fig. 2)

2. Insert a name for it in the “Database name” field. **The name must be unique, meaning no other databases with the same name exist.**
3. Click on “Save DB”, after the successful operation of adding a database the user will be redirected to the same page with the database added to the list.

FluoDetect Compare spectrums Search spectrum Add spectrum Create Spectrum DB

Create Database

Database name: Save DB

Create db form

Database name	Nr. of records
Fluodb	6
prova	2
Ph_db	4

List of all databases

Fig. 3.2: Create database page.

This page is also useful for inspecting the elements of every database on the list, if the user clicks on a database name he will be redirected to a page (Search Spectrums) with all the fluorescent records added so far in the selected database (Fig. 3.1 and Fig. 3.2).

Search Spectrums

Choose a DB: Fluodb
Select search criteria: Choose criteria
Search criteria:
Search spectrum

id	Spectra name	Nr. Peaks	Peak wavelengths	Peaks FWHM	Chosen min_peak_height
1	Fluorescin curcumin and chlorophyll	2	[534.58, 677.0]	[99.68, 54.7]	0.05
2	Tyrosine Tryptophan and Riboflavin	2	[361.84, 521.0]	[77.69, 65.67]	0.10
3	Chlorophyll	1	[679.27]	[80.41]	0.10
4	Curcumin	1	[538.55]	[118.07]	0.10
5	Fluorescin	1	[515.03]	[40.44]	0.10
6	fluorescina curcuma clorofilla 2	2	[534.58, 677.65]	[101.01, 55.39]	0.10

Fig 3.3.1: “Fluodb” database list of records.

Search Spectrums

Choose a DB: Ph_db
Select search criteria: Choose criteria
Search criteria:
Search spectrum

id	Spectra name	Nr. Peaks	Peak wavelengths	Peaks FWHM	Chosen min_peak_height
1	curcuma hcl4	1	[538.22]	[87.62]	0.1
2	curcuma ph2	1	[548.47]	[96.41]	0.1
3	curcuma ph8	1	[516.03]	[47.3]	0.1
4	curcuma ph10	1	[522.99]	[51.37]	0.1

Fig 3.3.1: “Ph_db” database list of records, this database for example is specifically made for samples of curcumin acquired with different pHs.

Adding a spectrum to a DB

How to add spectrums to a database

Once a database is created is time to populate it with fluorescent records. To do this the user can go to the “**Add spectrum**” page. On

this page (Fig. 4.1) several parameters for adding a new spectrum are present:

- **Excitation Wavelength begin (and end):** Two parameters describing the begin and end of the excitation wavelength source (in nm) if present in the spectrum (for example an LED or laser source).
- **Dimensionality reduction factor:** More often than not a spectrum file contains way more points (wavelengths and their respective intensities) than is necessary for describing the spectrum. Removing unnecessary points is useful more for the reason of comparison performance, both in CPU and memory used for the successive analysis of the record, but also for a well-known data analysis principle known as generalisation, which improves comparison results with unknown spectrums. **This parameter deletes $1/n$ points from the spectrum, where n is the dimensionality reduction factor.**
- **Smoothing window:** parameter describing the smoothing window size ($N \times N$) to use to smooth the spectrum inserted (for example in case of unwanted small peaks that could be detected instead by the application algorithm).
- **Minimum peak height:** parameter describing the minimum peak prominence (with respect to nearby points) for which a peak should be recognised as such.
- **Spectrum file:** the spectrum file the user wants to insert in the database, must be a .csv or .txt file—*warning (!!): the file must be in a specific format, which is described [here](#).*

Add Spectra

Excitation wavelength begin: Excitation wavelength end: Dimensionality reduction factor:

Smoothing window: Minimum peak height: Spectrum file:

Fig 3.4.1: Add spectrum page.

Once a spectrum file has been inserted (Fig. 4.2) it is possible to observe the summary of all the peaks with their respective FWHMs detected.

In the spectrum in Fig. 4.2 (Fluorescein), it is possible to observe that the first peak is that of an LED excitation source around 450 nm.

Peak wavelengths [nm]	FWHM peaks [nm]
453.118	19.595822
514.699	38.869493

Elaborated Spectra

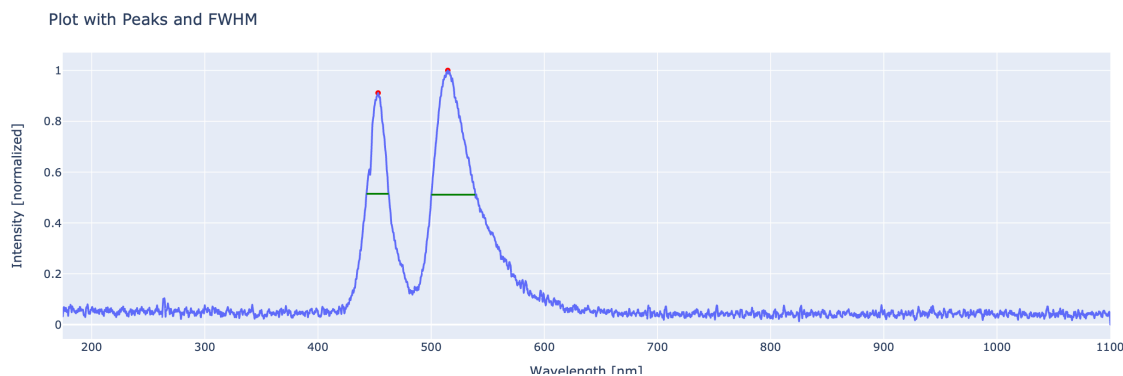


Fig. 3.4.2: lots of small peaks and excitation source present still.

Moreover, there are lots of small peaks (the spectrum has not been smoothed before). To fix this let's change the parameters and click on "Reload spectrum" to apply the changes (Fig. 3.4.3 and Fig. 3.4.4):

Add Spectra

Excitation wavelength begin: Excitation wavelength end: Dimensionality reduction factor:

Smoothing window: Minimum peak height:

File "fluorescina.txt" already inserted

Choose a DB: Spectrum name:

Fig. 3.4.3: new parameters.

Peak wavelengths [nm]	FWHM peaks [nm]
515.031	40.39814

Elaborated Spectra

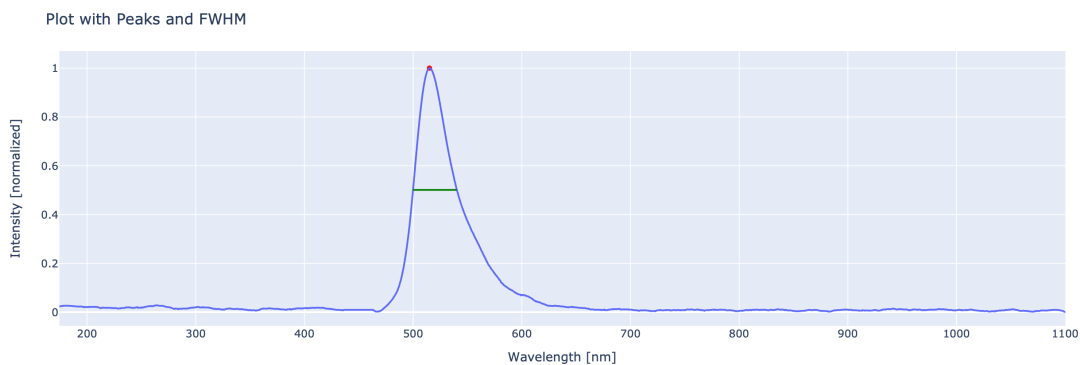


Fig. 3.4.4: spectrum with the applied changes.

To save the spectrum is necessary to choose a DB in which to save it and specify a name (preferably a descriptive name such as in this case “Fluorescein”) (Fig. 3.4.4).

After having saved successfully the new spectrum the user will be redirected to a page showing the summary of the added spectrum (Fig. 3.4.5).

The spectrum's most important information (table in Fig 3.4.5) is saved on a SQLite record in the selected DB.

The spectrum's wavelengths and respective intensities are saved in a .txt file in the following path: *high_res_db->[db_name]->[id of the saved spectrum]*.

The spectrum intensities in the .txt file are all saved normalised in the range [0,1].

For compatibility reasons with other fluorescent records that could use a different dimensionality reduction factor, the spectrums are all saved in the .txt at the highest resolution (which is fixed and equal for all records in all of the databases). Setting a dimensionality factor in this phase of adding a record is only useful to see what it could look like if the inserted dimensionality factor were used. This parameter will only be useful at the moment of comparing the spectrums in the “Compare spectrum page”.

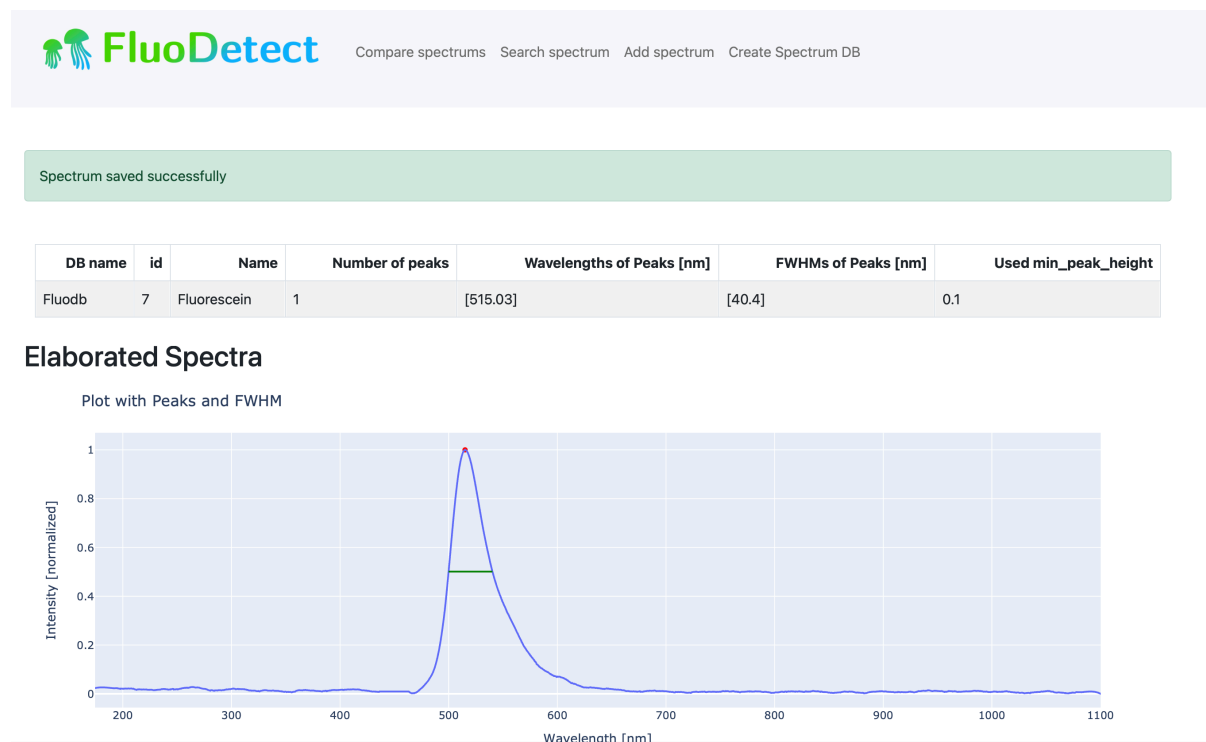


Fig. 3.4.5: spectrum saved successfully page.

Searching a spectrum

Different ways to look up records in the databases

There are three possible ways to search for a spectrum in the databases:

1. **By peak wavelength and FWHM** of the peaks present in the spectrum, the user wants to look for.
2. **By the name** given by the user to the fluorescent record.
3. **By id** in the SQLite database.

The 1. is the most “interesting” way, since looking up for a name or (especially) id implies knowing what to look for.

To search for a spectrum:

- The user has to click on “**Search Spectrum**” in the navigation toolbar to open the respective page (Fig. 5.1).
- Select a database in which to look for.
- Select the search criteria (between those already listed).

If for example the user were to use the “Peak wavelength and FWHM” criteria:

- It is necessary to insert a peak wavelength and its respective FWHM in the appropriate fields (Fig. 5.2).
- It is also necessary to **specify tolerances** in the appropriate fields (Fig. 5.2). These tolerances describe the range in which it is acceptable to find a fluorescent record wrt. to the previous parameters inputted. As an example if the user were to look for fluorescein:
 - Peak wavelength: 520 nm
 - FWHM: 40 nm
 - Peak wavelength tolerance: 10 nm
 - FWHM tolerance: 20 nm

The results of the previous example are shown in Fig. 5.3

FluoDetect Compare spectrums Search spectrum Add spectrum Create Spectrum DB

Search Spectrums

Choose a DB: Choose DB Select search criteria: Choose criteria Search criteria: Search spectrum

Choose a DB!

Fig. 3.5.1: Search spectrum page.

FluoDetect Compare spectrums Search spectrum Add spectrum Create Spectrum DB

Search Spectrums

Choose a DB: Fluodb Select search criteria: Peak wavelengt Peak wavelength: 520 FWHM: 40 Peak wavelength tolerance: 10 FWHM tolerance: 20 Search spectrum

Choose a DB!

Fig. 3.5.2: Example of the parameters inserted.

FluoDetect Compare spectrums Search spectrum Add spectrum Create Spectrum DB

Search Spectrums


Choose a DB: Fluodb Select search criteria: Peak wavelengt Peak wavelength: 520.0 FWHM: 40.0 Peak wavelength tolerance: 10.0 FWHM tolerance: 20.0 Search spectrum

id	Spectra name	Nr. Peaks	Peak wavelengths	Peaks FWHM	Chosen min_peak_height
5	Fluorescein	1	[515.03]	[40.44]	0.1
7	Fluorescein	1	[515.03]	[40.4]	0.1

Fig 3.5.3: Results of research displayed.

It is possible to inspect the spectrum plot and other information by clicking on the name of the compound (Fig. 5.4).

On the compound's information page, there is a section for the description of the compound, if no description is manually provided it is AI generated by OpenAI's GPT-4 engine based on the name given to the compound and its characteristics (peak wavelengths, FWHMs).

 [Compare spectra](#) [Search spectrum](#) [Add spectrum](#) [Create Spectrum DB](#)

View Spectra

id	Spectra name	Nr. Peaks	Peak wavelengths	Peaks FWHM	Chosen min_peak_height
7	Fluorescein	1	[515.03]	[40.4]	0.1

Spectrum

Plot with Peaks and FWHM

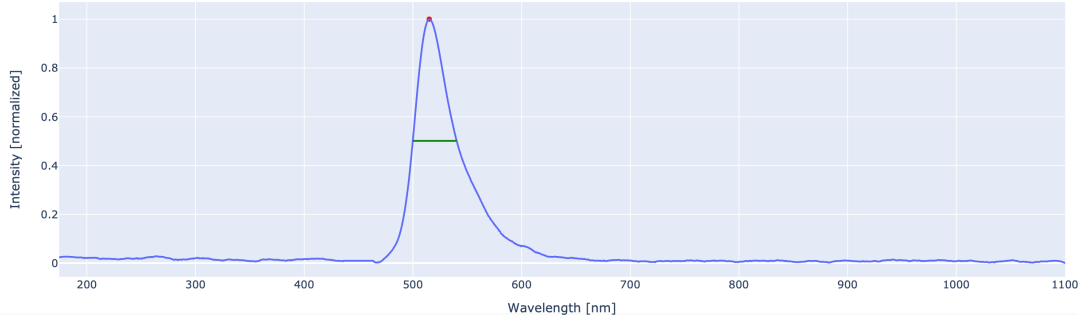


Fig. 3.5.4: Spectrum information page.

Description of the spectrum

[Save description](#)

[Use this as description](#) AI description of the spectrum

Fluorescein is indeed a fluorescent compound that can correspond to peak wavelength of 515.031 nm and full width at half maximum (fwhm) of 40.39813992 nm. Fluorescein is a synthetic organic compound available as a dark orange/red powder soluble in water and alcohol. It is widely used as a fluorescent tracer for many applications.

Fluorescein absorbs light of the wavelength about 494 nm and emits light at about 512 nm. Thus, its fluorescence can appear yellowish-green. The slight difference in the observed peak value could be a result of variation in experimental conditions or measurement precision. The fwhm value gives an estimate of the spectral width of the fluorescence peak, which can vary depending on several factors like medium, concentration, etc. Fluorescein is used in many applications, from biological staining and angiography to oil leak detection and much more.

[Delete spectrum](#)

Fig. 3.5.5: The same page as Fig. 5.4 scrolling down there is the description section.

If the user wills to do so it is possible to write manually a description or save that generated by the AI by simply pressing the button “**Use this as the description**” and then pressing “**Save description**” (Fig. 5.5).

It is also possible to delete the fluorescent record by pressing the “**Delete Spectrum**” button.

Comparing spectrums

How to compare an unknown spectrum with those present in the databases

The most useful feature of this application is to compare an unknown spectrum obtained by the user with those present in the database that most resemble the inserted spectrum by a comparison algorithm.

To do so is possible to click on the “**Compare spectrums**” element in the navigation bar.

On this page (Fig. 7.1) several parameters for adding a new spectrum are present:

- **Excitation Wavelength begin (and end):** Two parameters describing the begin and end of the excitation wavelength source (in nm) if present in the spectrum (for example an LED or laser source).
- **Dimensionality reduction factor:** More often than not a spectrum file contains way more points (wavelengths and their respective intensities) than is necessary for describing the spectrum. Removing unnecessary points is useful more for the reason of comparison performance, both in CPU and memory used for the successive analysis of the record, but also for a well-known data analysis principle known as generalisation, which improves comparison results with unknown spectrums [citation needed]. **This parameter deletes $1/n$ points from the spectrum, where n is the dimensionality reduction factor.**

- **Smoothing window:** parameter describing the smoothing window size ($N \times N$) to use to smooth the spectrum inserted (for example in case of unwanted small peaks that could be detected instead by the application algorithm).
- **Minimum peak height: parameter describing the minimum peak prominence (with respect to nearby points)** for which a peak should be recognised as such.
- **Spectrum file:** the spectrum file the user wants to insert in the database, must be a .csv or .txt file—*warning (!!): the file must be in a specific format, which is described [here](#).*

The comparison algorithm of this software works by comparing the peaks' wavelengths, number of peaks, FWHMs of the peaks, and the shape to those of the spectrums in the database.

The algorithm assigns “likelihood” scores to each comparison between the inserted spectrum and those in the database based on the four parameters described above. Each score is a result of the following parameters (that can be edited by the user clicking on the “**Show comparison options**” button):

- **pk_num_tol:** tolerance factor for the difference of peak number between the spectrums.
- **pk_tol:** tolerance factor for the difference of peak wavelength between the spectrums, each peak wavelength is confronted with the nearest in terms of the wavelength of the respective spectrum to compare.
- **fwhm_tol:** this tolerance factor is for the FWHMs of the respective peaks to be confronted.
- **weight_peak:** weight parameter describing how much do the difference of peak number and difference of peak wavelength weigh in the final comparison score.
- **weight_fwhm:** the same as the above weight parameter but for the difference of FWHMs of the respective compared peaks.

- **weight_shape**: weight parameter describing of much the difference in the shape of the two spectrums compared to weighs.
- **num_records_to_show**: number of records to show the user, records are ordered by score, so for example, if this parameter is “5” the application will show the 5 records that have scored the highest by the comparison algorithm.
- **shape_pt_tol**: tolerance for the difference in intensity between the points in the spectrums to be compared (**part of the shape comparison algorithm**).
- **baseline_pts**: defines a baseline for the shape comparison algorithm, a baseline is defined in intensity value. As an example, a value of “0.1” for this parameter will only compare in the comparison algorithm points in the spectrums to be compared between 0.1 and 1 values of intensity (Fig. 3.5.6).

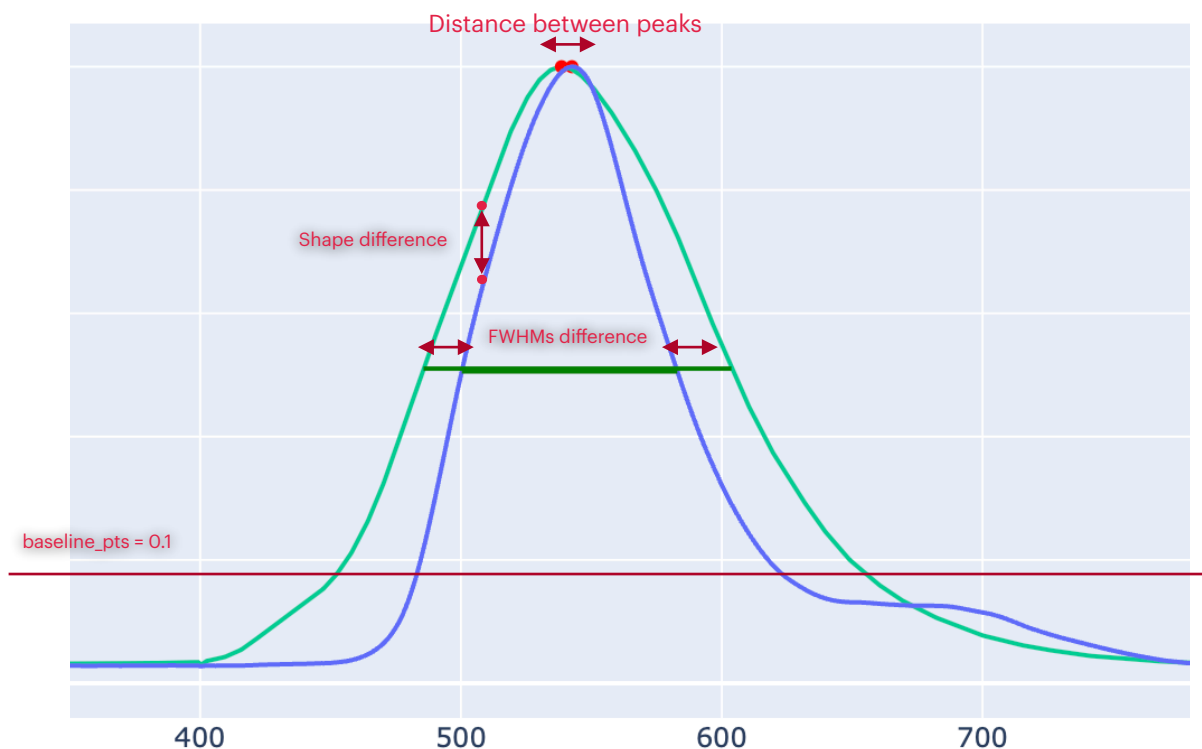


Fig 3.5.6: Illustrative example of some parameters.

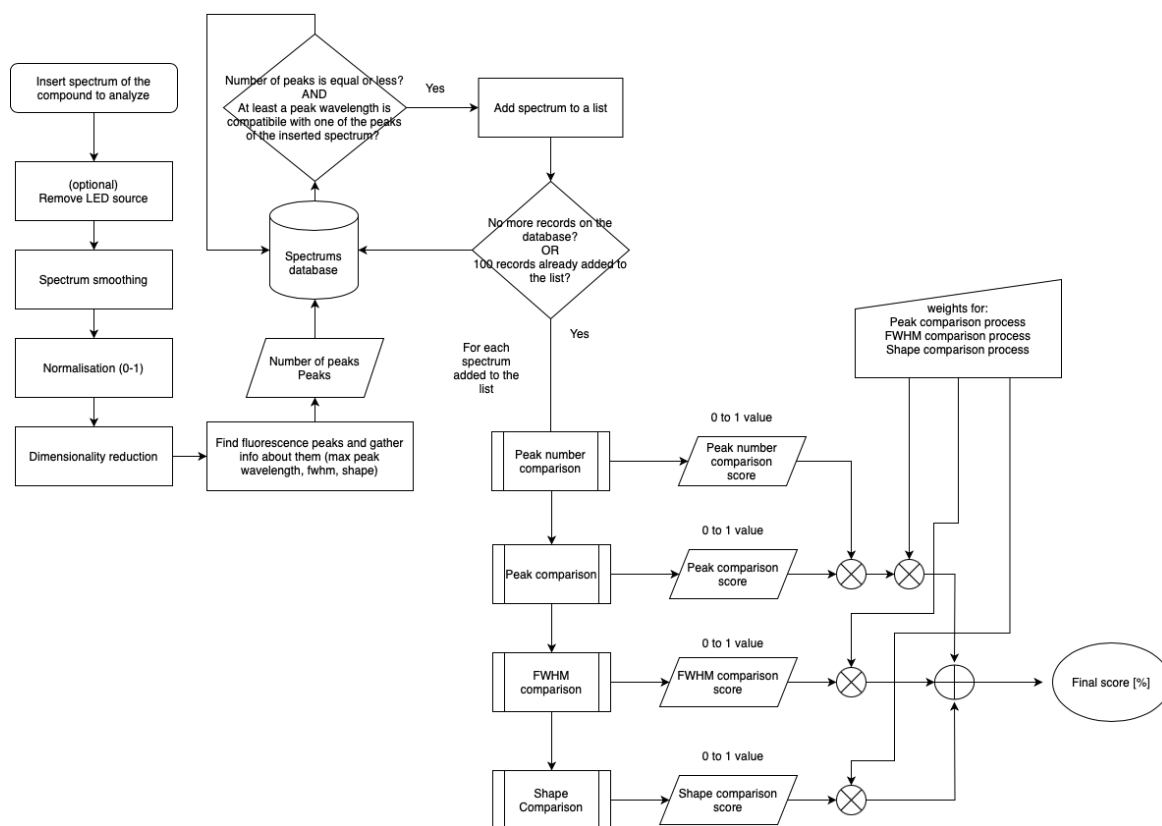


Fig 3.6.1: General process for the comparison algorithm.

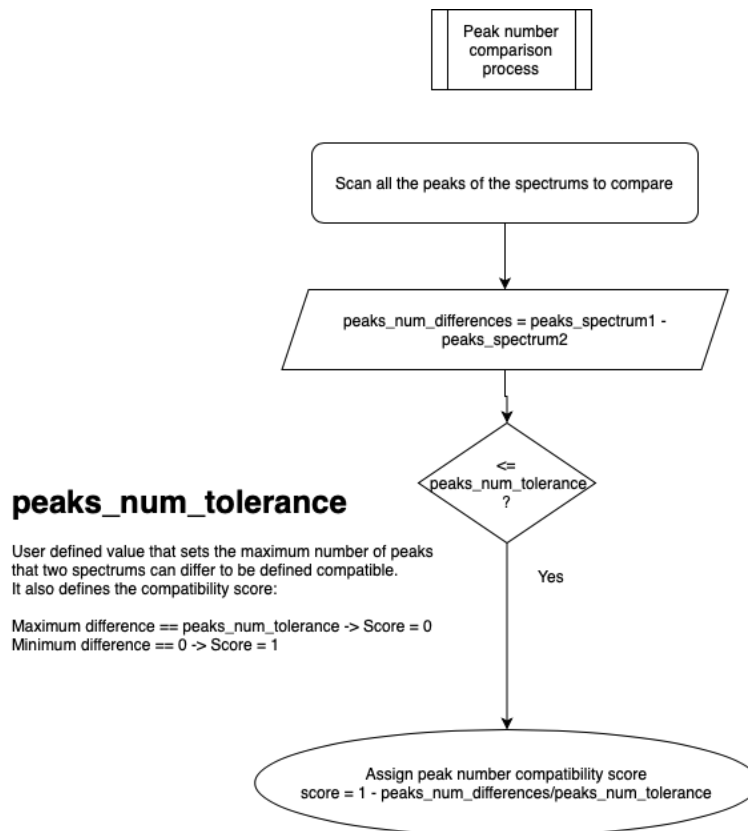


Fig 3.6.2: peak number comparison process.

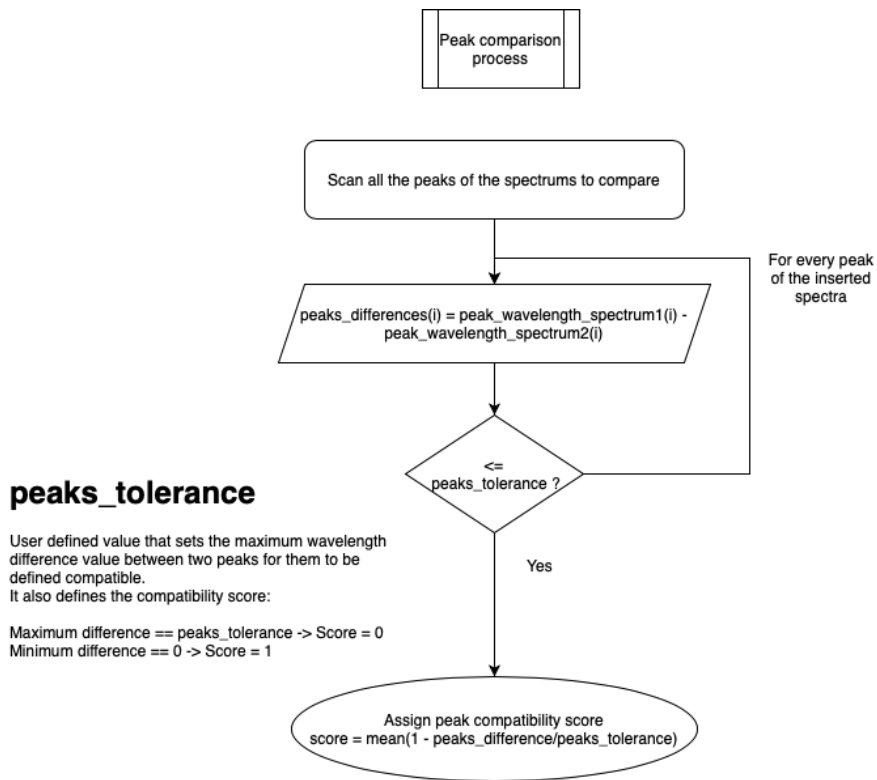


Fig 3.6.3: peak comparison process.

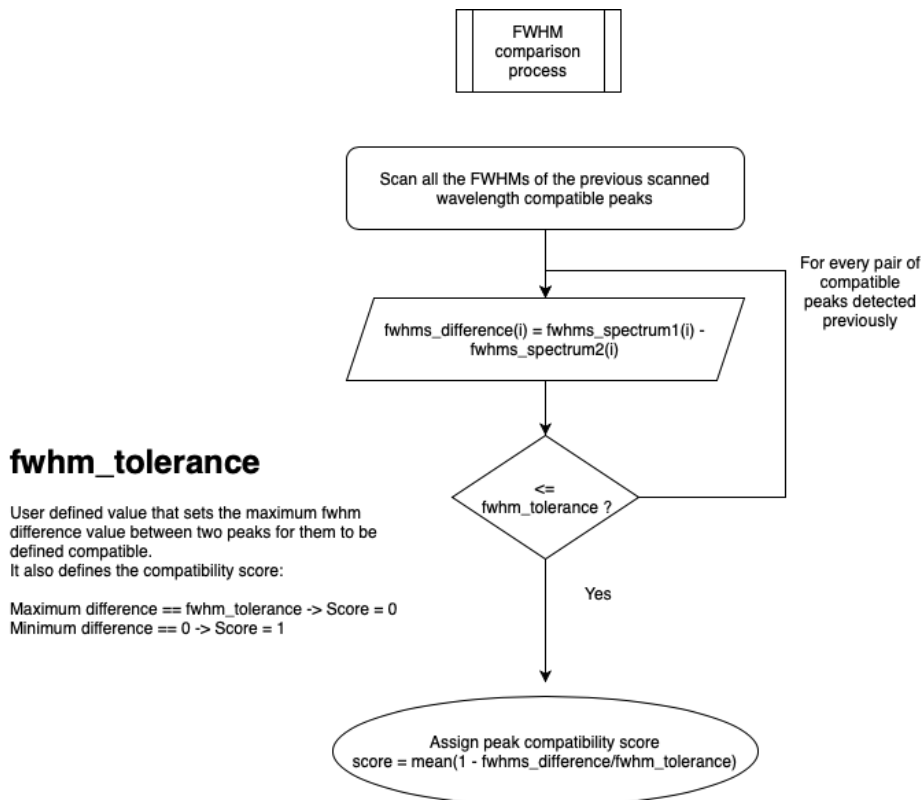


Fig 3.6.4: FWHM comparison process.

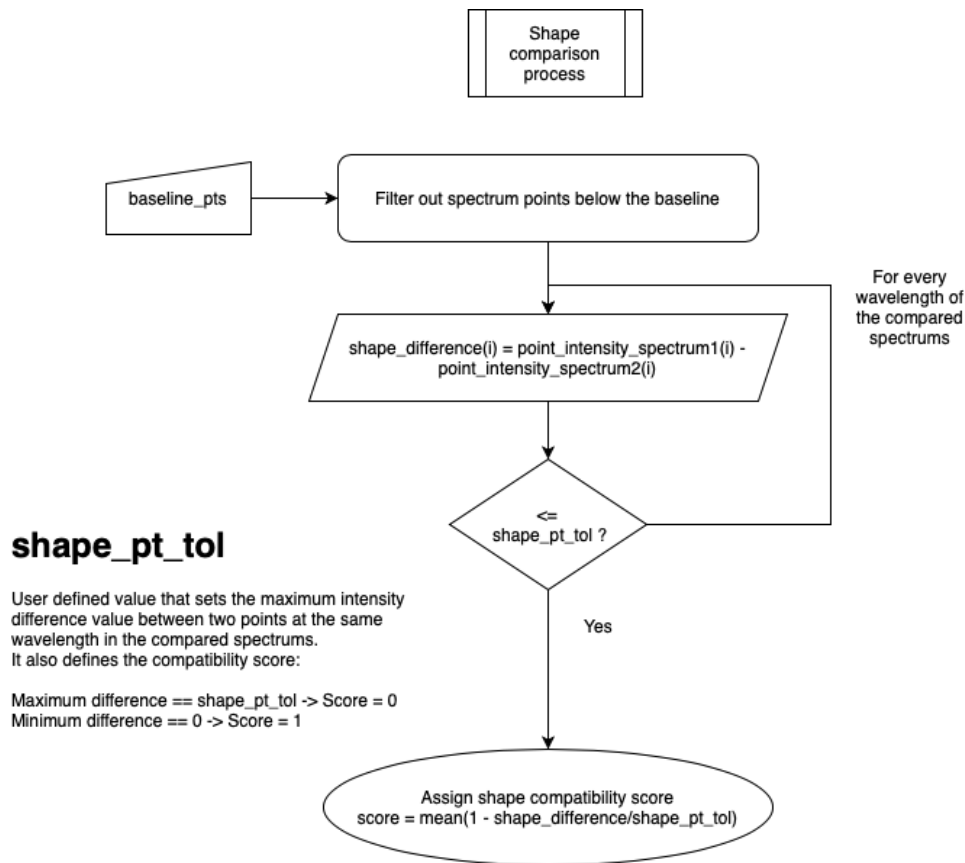


Fig 3.6.5: Shape comparison process.

To compare the spectrum the user has to upload it (“**Choose file**”), and then press “**Load spectrum**” button (Fig. 7.1 and 7.2).

Compare spectrums

Excitation wavelength begin:

Excitation wavelength end:

Dimensionality reduction factor:

Smoothing window:

Minimum peak height:

Spectrum file: no file selected

Upload it

Fig 3.7.1 Compare spectrums page.

Compare spectrums

Excitation wavelength begin: Excitation wavelength end: Dimensionality reduction factor:

Smoothing window: Minimum peak height:

pk_num_tol: pk_tol: fwhm_tol: weight_peak: weight_fwhm: weight_shape: num_records_to_show: shape_pt_tol: baseline_pts:

Peak wavelengths [nm]	FWHM peaks [nm]
[542.52]	[82.67]

Elaborated inserted Spectrum

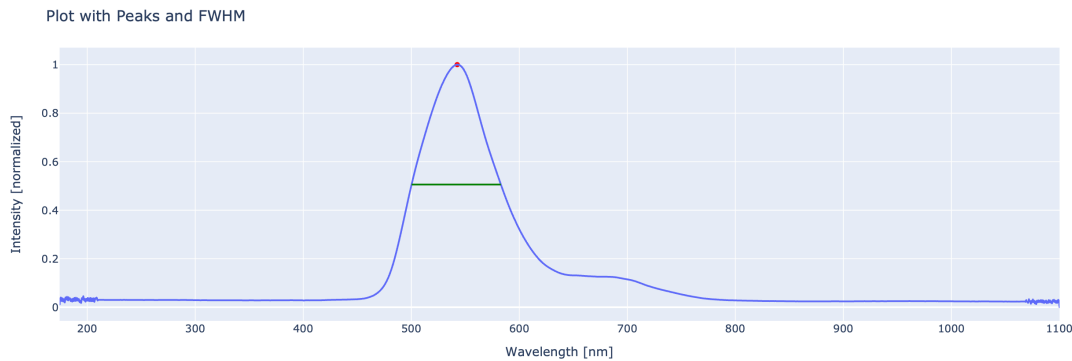


Fig 3.7.2: spectrum inserted.

Then the user **can choose a db** to compare the spectrum after having modified the comparison and conditioning parameters according to his necessities. The next step is pressing the “**Compare spectrum with those in DB**” button to start the comparison algorithm.

According to the “num_records_to_show” parameter a number of records of the highest likelihood will be shown (Fig. 7.3).

Every record is shown both graphically via a plot (overlapped with the conditioned inserted spectrum) and with a table that displays both the inserted spectrum and that in database which has been compared with most important characteristics (Fig. 7.3 shows an example).

Name	Peak wavelengths [nm]	FWHM peaks [nm]	Score [%]
Curcumin	[538.55]	[118.07]	77.631886
Inserted spectra	[542.52]	[82.67]	77.631886

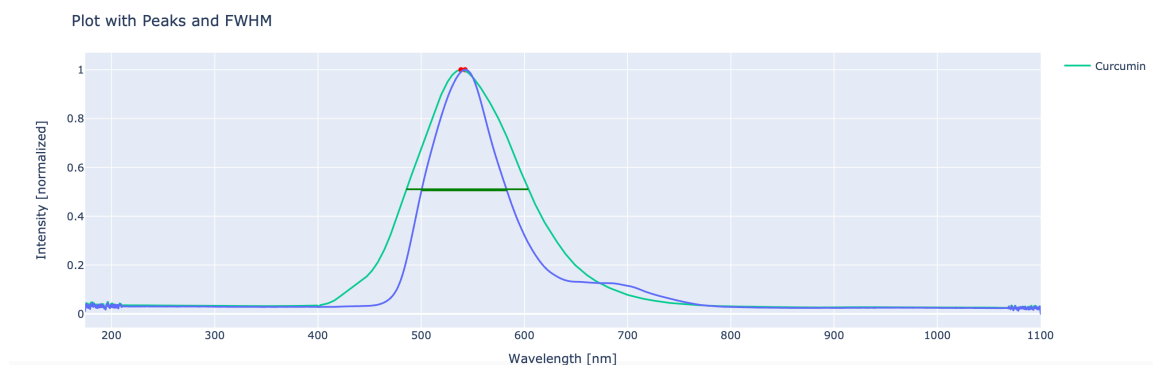


Fig 3.7.3: Example result of comparison algorithm.

The score displayed in the table is an indication of the similarity between the two compared spectrums based on the algorithm explained before in this section.

Compatible file formats for inserting spectrums

This software can only read and interpret spectrums coming from certain formatted .txt files. In Fig. 8 there is an example:

```

Integration time [ms]: 2000.000
Averaging Nr. [scans]: 1
Smoothing Nr. [pixels]: 100
Data measured with spectrometer [name]: 1010129U1
Wave ;Sample ;Dark ;Reference;Scope
[nm] ;[counts] ;[counts] ;[counts]

174.456; -674.000; 0.000; 0.000
174.799; -801.000; 0.000; 0.000
175.142; 1893.000; 0.000; 0.000
175.485; 838.000; 0.000; 0.000
175.828; -796.000; 0.000; 0.000
176.170; -136.000; 0.000; 0.000
176.513; 804.000; 0.000; 0.000
176.856; 606.000; 0.000; 0.000
177.199; -459.000; 0.000; 0.000
177.542; -159.000; 0.000; 0.000
177.884; 778.000; 0.000; 0.000
178.227; 681.000; 0.000; 0.000
178.570; -893.000; 0.000; 0.000
178.913; -674.000; 0.000; 0.000
179.255; 729.000; 0.000; 0.000

```

Fig 3.8: example of file FluoDetect can read.

The file has to follow the following specifications:

- **At least two columns: the first one containing the wavelength of a point, the second one containing its intensity at that wavelength.**
- The columns must be separated by a “;” semi-column sign.
- Rows must be separated by a new line.
- **The spectrum data must begin after 6 lines of written header.** It cannot be blank.
- **The spectrum data must contain 2846 points.**
- **The spectrum points must begin from 174.456 [nm] and end at 1100.148 [nm], every subsequent point must be separated from the other by 0.343 [nm].**

Software classes and methods

Introduction: Flask, databases, and most important files/folders

A brief overview

The project code is shared under conditions of the **GNU General Public License v3.0**. Permissions of this strong copyleft license are conditioned on making available the complete source code of licensed works and modifications, which include larger works using a licensed work, under the same license. Copyright and license notices must be preserved. Contributors provide an express grant of patent rights.

Github link: <https://github.com/LorenzoRiggi/fluoProject>

The web server has been created using Flask. Flask is a web framework that allows developers to build lightweight web applications quickly and easily with Flask Libraries.

There are 3 important files/folders for Flask to maintain to view the website correctly:

- **app.py**: contains the app routes, i.e. URLs that trigger Python functions. These functions are useful to handle REST API requests (GET, POST) and call other functions in the code useful for webpage visualisation. **This is the most important code file and the one that's called for every webpage loading first.**
- **templates folder**: contains the webpages front-end code (HTML, CSS, Javascript). Code that is under brackets “{% code %}” is **Jinja2 template code** and is executed before loading the page (at the back end) and invisible to the user that wants to inspect the code. Jinja is a very useful templating engine that makes the job of passing variables between the back end and front end way easier and quicker. The code looks similar to Python but it isn't in many instances.
- **requirements.txt**: file that contains all the module requirements of the application, makes the job easier to import and maintain updated the list of modules used by the Python code.

Then there is the database where all the information about spectrums are stored:

- **fluodb.sqlite**: is the SQLite database where all the spectrums' most important info are stored:
 - Tables inside this database represent the different spectrum databases. Each row in the tables (representing a spectrum) contains these columns:
 - **id**: unique integer identifier for the spectrum (main key).
 - **Name**: spectrum name given by the user when the spectrum is inserted in the database [TEXT type].
 - **pks_num**: number of peaks in the spectrum [INT type].
 - **pks_wav**: wavelengths of peaks [REAL type].

- **fwhm**: FWHMs of the peaks [REAL type].
 - **min_peak_height**: prominence height used to detect peaks at the moment of inserting the spectrum in the database (See “Adding a spectrum to a DB” section). [REAL type]
 - **Dangerous**: optional boolean parameter to be used in the future to classify hazardous compounds for humans in the database.
- **high_res_spectra_db folder**: the folder containing all the spectrums .txt files containing the wavelengths and their relative intensities (Fig. 8.2). **Every spectrum file is named after its id and is in a folder named the parent database in which the spectrum is inserted.** Example: *high_res_spectra_db/Fluodb/7.txt*.
 - **descriptions folder**: the folder containing all the descriptions for the spectrums (if they are provided by the user) in .txt files (Fig. 8.2). **Every spectrum file is named after its id and is in a folder named the parent database in which the spectrum is inserted.** Example: *descriptions/Fluodb/5.txt*.

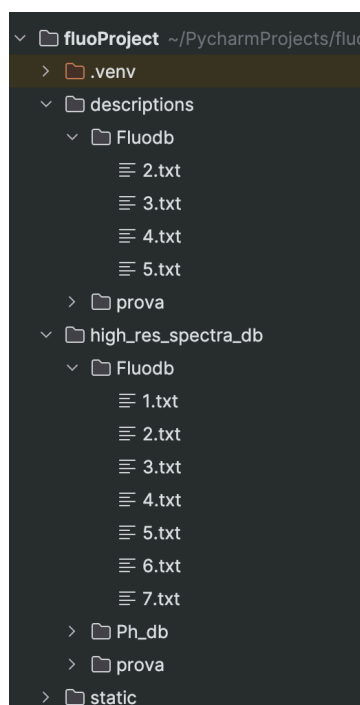


Fig 3.9.2: .txt files.

id	Name	pks_num	pks_wav	fwhm	min_peak_height	dangerous
1	Fluorescin curcumin and chlorophyll	2	534.58	99.68	0.85	0
2	Fluorescin curcumin and chlorophyll	2	677	54.7	0.85	0
3	Tyrosine Tryptophan and Riboflavin	2	361.84	77.69	0.1	0
4	Tyrosine Tryptophan and Riboflavin	2	521	65.67	0.1	0
5	Chlorophyll	1	679.27	80.41	0.1	0
6	Curcumin	1	538.55	118.89	0.1	0
7	Fluorescin	1	515.83	48.44	0.1	0
8	Fluorescina curcuma clonofilla 2	2	534.58	181.01	0.1	0
9	Fluorescina curcuma clonofilla 2	2	677.45	55.39	0.1	0
10	Fluorescin	1	515.83	48.4	0.1	0

Fig 3.9.1: example of fluodb.sqlite database.

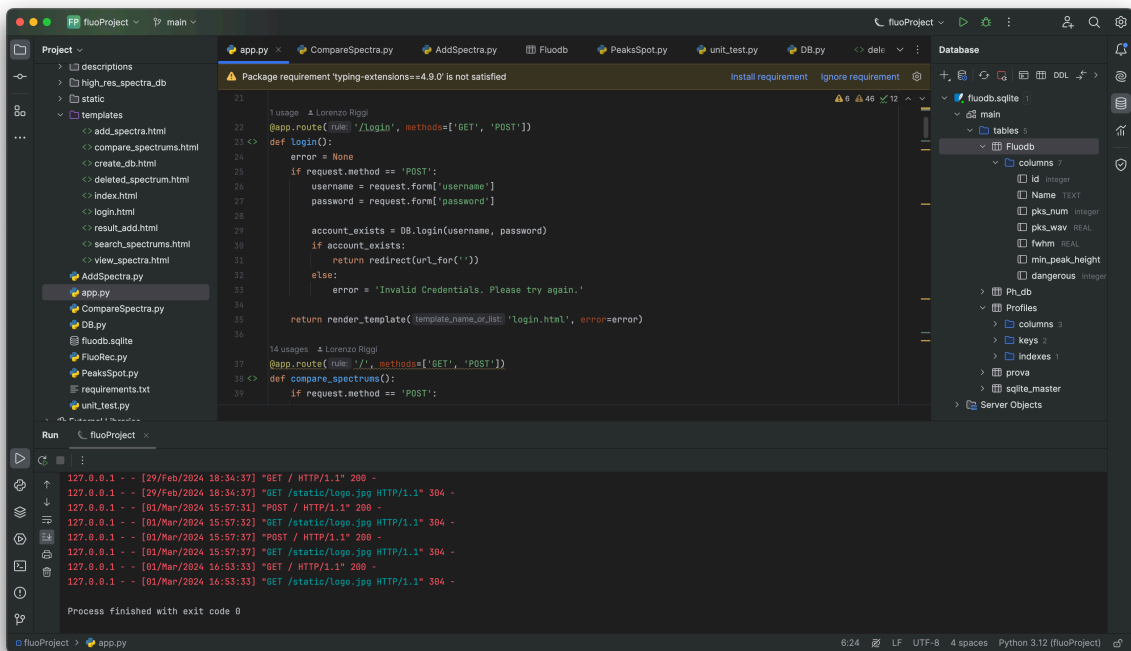


Fig 3.9.3: Example view of the FluoDetect project on PyCharm.

DB Class

An overview of the class to handle database requests

The DB class, contained in the module named “*DB.py*”, handles all the requests to the SQLite database in the application.

Its only initialisation parameter is **db_name**, the name of the DB to work on in the current session: `def __init__(self, db_name)`.

As an example, if the user wants to work with curcumin (Fig. 9.1) the app should initialise an instance of the DB class with the “Fluodb” table in the database (Fig. 9.1):

```
db = DB("Fluodb")
```

The most important methods for handling database requests are the following:

- `def add_record(self, name, pks_wavs, fwhm, min_peak_height)`
 - Adds a record to the SQLite database in the database (table) in which the DB class has been initialised. Returns the id given to the newly inserted record. **The id number is the larger id number in the database + 1**. The input arguments are the following:
 - **name**: spectrum name given by the user when the spectrum is inserted in the database [String type].
 - **pks_num**: number of peaks in the spectrum [integer type].
 - **pks_wav**: wavelengths of peaks [float type].
 - **fwhm**: FWHMs of the peaks [float type].
 - **min_peak_height**: prominence height used to detect peaks at the moment of inserting the spectrum in the database (See “Adding a spectrum to a DB” section). [float type]

- **Important disclaimer:** if a fluorescent record contains more than 1 peak **more rows in the table are added with the same id, name, pks_num and min_peak_height but with different pks_wav and fwhm.** As an example a fluorescent record with 2 peaks will be inserted in the database as 2 different rows, each having its own peak wavelength and fwhm. When records are retrieved from the table using the methods in this class the “clone” records are put together using Pandas data-frames (see `def get_db(self)` as an example of how they are assembled together).

- `def delete_record(self, id_):`
 - Deletes a record from the SQLite database in the database (table) in which the DB class has been initialised and deletes also its related files in the *high_res_spectra_db* and *descriptions* folders. **Takes as argument the id of the record in the selected database.**

- `@staticmethod def create_db(table_name):`
 - Creates a database with the name specified in the argument.

- `@staticmethod def get_databases_list()`
 - Returns the list of database in the general SQLite database as a python list of strings and the number of records in each database as a python list of integers. Example: ["Fluodb", "Phdb"], [7, 4]

- `def get_db(self)`
 - Returns the list of records in the selected database for which the DB class has been initialised for. **The list is returned as a Pandas data-frame** in the following format (data inserted in the Pandas columns in the following is provided as an example):

```
{'DB Name': "Fluodb", 'id': 7, 'Name': "Fluorescein and curcumin", 'pks_num': 2, 'pks_wav': [520.0, 640.0], 'fwhm': [40.0, 50.0], 'min_peak_height': 0.1}
```

- `def look_up_by_id(self, id_)`
 - Searches in the selected database for which the DB class has been initialised for a record by its id. Returns the record as a Pandas data-frame (formatted in the same way as the `def get_db(self)` method returned data-frame) and as a FluoRec class (see this section for details).

- `def look_up_by_name(self, name)`
 - Searches in the selected database for which the DB class has been initialised for all the records with the name specified in the argument (records with the same name are allowed). Returns the records as a Pandas data-frame (formatted in the same way as the `def get_db(self)` method returned data-frame **but with more than one row if more than one record has the same name**).

- `def look_up_records_pks_wav_fwhm(self, pks_wav, fwhm, tolerance_pks, tolerance_fwhm)`
 - Searches in the selected database for which the DB class has been initialised for all the records that satisfy the following conditions:
 - The record as one or more peak wavelengths that are within the boundaries specified by :
 - $pks_wav - tolerance_pks \leq peak_wavelength \leq pks_wav + tolerance_pks$
 - The records peak wavelengths' FWHMs are within the boundaries specified by :
 - $fwhm - tolerance_fwhm \leq FWHM \leq fwhm + tolerance_fwhm$
 - Returns the records as a Pandas data-frame (formatted in the same way as the `def get_db(self)` method returned data-frame **but with**

more than one row if more records that satisfy the conditions specified are present).

- `@staticmethod def get_record_description(self, db_name, id_)`
 - Return the record description for the spectrum with the id and in the database specified in the method arguments. The record description is returned as a string.
- `@staticmethod def update_record_description(self, db_name, id_, description)`
 - Updates the record description for the spectrum with the id and in the database specified in the method arguments. The description argument must be a string.
- `def close(self)`
 - Closes the current connection to the database. It is important to close it manually by calling this function every time a database manipulation has ended so other connections can be opened.

FluoRec Class

FluoRec is a class that is mostly used as a struct more than a class. There are some instances in which this class is used instead of a Pandas data-frame for reasons of readability of the code (in the opinion of the author of this software).

The constructor of this class (`__init__`) defines the class variables:

```
def __init__(self, db_name, id_, name, pks_num, pks_wav,
             fwhm, min_peak_height, score=None):
    self.db_name = db_name
    self.id_ = id_
```

```

self.name = name
self.pks_num = pks_num
self.pks_wav = pks_wav
self.fwhm = fwhm
self.min_peak_height = min_peak_height
self.score = score

```

- **db_name**: name of the database in which this record is contained [String type].
- **id_**: id of the record in the database [int type]
- **name**: spectrum name given by the user when the spectrum is inserted in the database [String type].
- **pks_num**: number of peaks in the spectrum [integer type].
- **pks_wav**: wavelengths of peaks [float type].
- **fwhm**: FWHMs of the peaks [float type].
- **min_peak_height**: prominence height used to detect peaks at the moment of inserting the spectrum in the database (See “Adding a spectrum to a DB” section). [float type]
- **score**: this optional class variable is used when comparing different spectrums, it is the likelihood score of the instance of FluoRec scores with the spectrum to be compared with [int type, expressed as a percentage from 0 to 100].

This class has to useful methods that exploit the knowledge of the data inserted in the constructor:

- `def get_spectra_high_res(self)`

- Return the spectrum data in high resolution (no dimensionality reduction factor). The spectrum data contained in the file (read more about the file formatting [here](#)) **is returned in two NumPy arrays; One for the wavelengths, and one for the intensities relative to the wavelengths.**
- `def get_spectra_dim_reduction(self, alpha)`
 - Return the spectrum data with the resolution provided by the dimensionality reduction factor (**alpha**). As an example if $\alpha = 3$ then 1/3 of the points are discarded (both wavelengths and intensities). The manipulated spectrum data contained in the file (read more about the file formatting [here](#)) **is returned in two NumPy arrays; One for the wavelengths, and one for the intensities relative to the wavelengths.**

PeaksSpot Module

This module is used to analyse a spectrum's most important parameters (peaks, FWHMs of the peaks).

- `def peaks_spot(wavelengths, intensities, min_peak_height)`
 - Analyses the before-mentioned characteristics of a spectrum through `scipy.signal`'s functions (`find_peaks` for finding the peak wavelengths, `peak_widths` to find out their FWHMs).
 - It accepts as arguments the spectrum's wavelengths and intensities **NumPy arrays**. An eight prominence (`min_peak_height`) must also be specified in order for the algorithm to distinguish what is a peak and what is not a peak.
 - Returns:
 - **locations:** NumPy array containing the wavelengths (locations) of the peaks detected.

- **width_heights**: the heights (y coordinate) of the peaks' FWHMs (height of FWHMs) (useful for graphic reasons) of the inserted spectrum, it is a Python list.
- **left_ips, right_ips**: x coordinates (python lists) of the FWHMs of the peaks (beginning and end of the width of the peak, useful for graphic reasons).
- **fwhm**: NumPy array containing the FWHMs of the peaks detected.
- `def index_to_xdata(xdata, indices)`
 - Interpolate the values from signal.peak_widths indices to the xdata space (in this case the wavelengths space). Used only inside the `def peaks_spot` function.

AddSpectra Class

This class contains the methods useful for adding a spectrum to the database. Its class variables are also used to this goal:

```
def __init__(self, led_range, smoothing_wind,
min_peak_height, alpha, spectrum_to_read):
    self.spectrum_to_read = spectrum_to_read
    self.led_range = led_range
    self.smoothing_wind = smoothing_wind
    self.min_peak_height = min_peak_height
    self.alpha = alpha
```

- **spectrum_to_read**: Pandas data-frame containing the wavelengths and intensities (NumPy arrays) of the spectrum to be analysed.

Wavelengths must be the first column and intensities the second.

- **led_range**: python list of 2 elements (example: [400, 450]) indicating the range of wavelengths of the excitation source (if present in the spectrum). These wavelengths are zeroed in intensity and then smoothed (`def analyze_spectra(self)` method). It is a Python list.
- **smoothing_wind**: Smoothing window, it is a square window of NxN.
- **min_peak_height**: minimum height prominence that a peak has to have in order to be recognised as such. **It can be a float from 0 to 1.**
- **alpha**: dimensionality reduction factor, used to reduce the resolution (number of points) in the spectrum. Example alpha = 3 discards 1/3 of the points of the spectrum. **Important disclaimer:** the spectrum's .txt records file is always saved at the highest possible resolution (the resolution of the .txt file inserted).

The following are the methods declared in this class:

- `def analyze_spectra(self)`
 - This method serves the purpose of conditioning the spectrum (with the parameters above explained) and finding the spectrum's most important parameters (peaks, FWHMs) through the methods described in the PeaksSpot Module.
 - It has no arguments but the instance of the class itself, so an instance of the class with the desired conditioning parameters should be initialised in order for it to work.
 - Returns: *wavelengths_reduced, intensities_reduced, wavelengths, intensities, locations, fwhm, width_heights, left_ips, right_ips.*
 - **wavelengths_reduced**: conditioned wavelengths (dimensionality reduction), it is a NumPy array.

- **intensities_reduced**: conditioned intensities (gone through the removal of excitation source wavelengths, smoothing and normalization, and dimensionality reduction), it is a NumPy array.
- **wavelengths**: original wavelengths, it is a NumPy array.
- **intensities**: intensities gone through the removal of excitation source wavelengths, smoothing, and normalization (0 -> 1 scale), it is a NumPy array.
- **locations**: location of detected peaks in nm, NumPy array.
- **fwhm**: FWHMs of detected peaks in nm, it is a NumPy array.
- **width_heights**: the heights (y coordinate) of the peaks' FWHMs (height of FWHMs) (useful for graphic reasons) of the inserted spectrum, it is a Python list.
- **left_ips, right_ips**: x coordinates (python lists) of the FWHMs of the peaks (beginning and end of the width of the peak, useful for graphic reasons).
- `def save_spectra(self, db_name, spectra_name, wavelengths, intensities, locations, fwhm, min_peak_height)`
 - This method serves the purpose of saving the conditioned spectrum (the version that is not dimensionality reduced) in a .txt file and saving its most important parameters in the SQLite database.
 - The arguments of this method are respectively:
 - **db_name**: String representing the database name in which the user wants to save the spectrum (example: "Fluodb").
 - **spectra_name**: String representing the name for which the user wants to save the spectrum in the DB.
 - **wavelengths**: original wavelengths, it is a NumPy array.
 - **intensities**: intensities gone through the removal of excitation source wavelengths, smoothing, and normalization (0 -> 1 scale), it is a NumPy array.
 - **locations**: location of detected peaks in nm, NumPy array.

- **fwhm**: FWHMs of detected peaks in nm, it is a NumPy array.
- **min_peak_height**: peak height used to recognise peaks as such. It can be a float from 0 to 1.

CompareSpectra Class

This class contains methods to compare spectrums. The algorithm for comparing spectrums is explained [in this section](#).

This class conditions the inserted spectrum (removal of the excitation source, smoothing, normalization, dimensionality reduction) the user wants to compare with other spectra in a db of his choice and then uses comparison algorithms to assign scores to the comparisons.

The declaration and constructor of this class are the following:

```
def __init__(self, db_name, led_range, smoothing_wind,
min_peak_height, alpha, spectrum_to_read, comp_set):

    self.db_name = db_name

    self.spectrum_to_read = spectrum_to_read

    self.led_range = led_range

    self.smoothing_wind = smoothing_wind

    self.min_peak_height = min_peak_height

    self.alpha = alpha

    self.comp_set = comp_set
```

- **db_name**: String representing the database name in which the user wants to compare the spectrum(example: “Fluodb”).
- **spectrum_to_read**: Pandas data-frame containing the wavelengths and intensities (NumPy arrays) of the spectrum the user has inserted and wants to compare with other spectrums in the db. **Wavelengths must be the first column and intensities the second.**

- **led_range**: python list of 2 elements (example: [400, 450]) indicating the range of wavelengths of the excitation source (if present in the spectrum). These wavelengths are zeroed in intensity and then smoothed (`def analyze_spectra(self)` method). It is a Python list.
- **smoothing_wind**: Smoothing window, it is a square window of NxN.
- **min_peak_height**: minimum height prominence that a peak has to have in order to be recognised as such. **It can be a float from 0 to 1.**
- **alpha**: dimensionality reduction factor, used to reduce the resolution (number of points) in the spectrum. Example alpha = 3 discards 1/3 of the points of the spectrum.
- **comp_set**: dictionary representing the comparison parameters:
 - **pk_num_tolerance**: tolerance factor for the difference of peak number between the spectrums.
 - **pk_tolerance**: tolerance factor for the difference of peak wavelength between the spectrums, each peak wavelength is confronted with the nearest in terms of the wavelength of the respective spectrum to compare.
 - **fwhm_tolerance**: this tolerance factor is for the FWHMs of the respective peaks to be confronted.
 - **weight_peak**: weight parameter describing how much the difference of peak number and difference of peak wavelength weigh in the final comparison score.
 - **weight_fwhm**: the same as the above weight parameter but for the difference of FWHMs of the respective compared peaks.
 - **weight_shape**: weight parameter describing of much the difference in the shape of the two spectrums compared weighs the final comparison score.
 - **num_records_to_show**: number of records to show the user, records are ordered by score, so for example, if this parameter is “5” the application will show the 5 records that have scored the highest by the comparison algorithm.

- **shape_pt_tol**: tolerance for the difference in intensity between the points in the spectrums to be compared (**part of the shape comparison algorithm**).
- **baseline_pts**: defines a baseline for the shape comparison algorithm, a baseline is defined in intensity value. As an example, a value of “0.1” for this parameter will only compare in the comparison algorithm only points in the spectrums to be compared between 0.1 and 1 values of intensity

The declared methods are the following:

- `def compare_fluorescence_recs(self, spectra_compare, fluo_rec)`
 - Compares the fluorescent records’ number of peaks, peak wavelengths and FWHMs.
 - **spectra_compare**: Pandas data-frame containing the 2 NumPy arrays with the wavelengths and intensities of the inserted spectrum to be compared with **fluo_rec**.
 - **fluo_rec**: FluoRec instance representing the spectrum to be compared with spectra_compare.
 - Returns the score of the comparison of the records’ number of peaks, peak wavelengths and FWHMs, it is a float value between 0 and 1 that will be added with the score scored by the shapes’ comparison.
- `def compare_fluo_recs_shapes(self, spectra_compare_ints, spectra_compare_wavs, fluo_rec)`
 - Compares the fluorescent records’ shapes.
 - **spectra_compare_ints**: NumPy array containing the intensities values of the spectrum to be compared.
 - **spectra_compare_wavs**: NumPy array containing the wavelengths’ values of the spectrum to be compared.

- **fluo_rec**: FluoRec instance representing the spectrum to be compared with `spectra_compare_ints` and `spectra_compare_wavs`.
 - Returns the score of the comparison of the records' shapes, it is a float value between 0 and 1 that will be added with the score scored by the number of peaks, peak wavelengths, and FWHMs comparison.
- `def compare_spectra(self)`
 - This method starts the comparison algorithm. It chooses the spectrums the inserted spectrum has to be compared with based on the number of peaks and similarity in wavelength value of the peaks and calls the functions `compare_fluorescence_recs` and `compare_fluo_recs_shapes` to assign scores.
 - Returns:
 - **records**: FluoRec instances representing the spectrums that have scored the highest score in comparison with the inserted spectrum.
 - **wavelengths_reduced**: conditioned wavelengths (dimensionality reduction) of the inserted spectrum, it is a NumPy array.
 - **intensities_reduced**: conditioned intensities (gone through the removal of excitation source wavelengths, smoothing and normalization, and dimensionality reduction) of the inserted spectrum, it is a NumPy array.
 - **wavelengths**: original wavelengths of the inserted spectrum, it is a NumPy array.
 - **intensities**: intensities gone through the removal of excitation source wavelengths, smoothing, and normalization (0 -> 1 scale) of the inserted spectrum, it is a NumPy array.
 - **locations**: location of detected peaks in nm of the inserted spectrum, NumPy array.
 - **fwhm**: FWHMs of detected peaks in nm of the inserted spectrum, it is a NumPy array.

- **width_heights**: the heights (y coordinate) of the peaks' FWHMs (height of FWHMs) (useful for graphic reasons) of the inserted spectrum, it is a python list.
- **left_ips, right_ips**: x coordinates (python lists) of the FWHMs of the peaks (beginning and end of the width of the peak, useful for graphic reasons) of the inserted spectrum.

App Module

An overview of the module to handle the Flask web server URLs

The app module serves the purpose for the Flask framework to trigger specific python functions in the module when a webpage is opened by the user.

As an example, when the “Search spectrums” webpage is opened it triggers the `search_spectrums()` function:

```
@app.route('/search_spectrums', methods=[ 'GET' ])
def search_spectrums():
```

And in doing so it specifies that in the function are passed parameters by GET (REST API) in this example.

The web pages in the web application are all present in the *template* folder.

No further detail about the specific implementation of each function in this module will be provided since very little processing is done in each function. Each function serves the goal of collecting parameters passed by GET or POST request, calling the functions described in the previous sections, and preparing the data for visualisation (plotting, and displaying tables).

An overview of what each function does is listed in the following:

- `def compare_spectrums()`
 - Collects data for comparing a spectrum inserted by the user with spectrums in the database and prepares the data for visualisation in the relative page.

- `def deleted_spectrum_page()`
 - Collects data for the spectrum to be deleted and deletes it, calls the webpage for the successful deletion of a fluorescent record, or in case of failure calls the relative webpage for an error.

- `def view_spectra()`
 - Collects data for viewing a specific spectrum and prepares it for visualisation in the relative webpage.

- `def search_spectrums()`
 - Collects data for searching spectrums with the parameters inputted in the relative page and prepares the data to be visualised on a webpage with a table representing all the fluorescent records fitting the criteria inputted by the user.

- `def create_db()`
 - Presents the webpage with all the databases created and allows the creation of a new DB.

- `def add_spectra_page()`
 - Calls the webpage related to the creation of a new fluorescent record in the database the user wants to add the record to and once the user has manipulated the spectrum to his will allows the adding of such spectrum in the selected database.

- `def result_add_successful()`
 - Shows the webpage for the successful creation of a new fluorescent record in the selected database and case of failure displays an error.

- `def create_plot(df, plot_type, locations=None, width_heights=None, left_ips=None, right_ips=None, df_comp=None, df_comp_name=None, df_comp_params=None)`
- This function is called often in the app module. Its purpose is to create a Plotly plot representing the spectrums that are being shown on the webpages
- The arguments of the function are:
 - **df**: Pandas data-frame representing the wavelengths and relative intensities of every point in the spectrum file. The first column is reserved for wavelengths, the second for intensities. The values in the columns must be NumPy arrays.
 - **plot_type**:
 - plot type 1: simple spectrum without annotations (peaks, FWHMs).
 - plot type 2: with annotations (mark peaks and FWHMs)
 - plot type 3: overlap of two spectrums.
 - **locations (optional, plot_type = 2, 3)**: location of detected peaks in nm of the inserted spectrum, NumPy array.
 - **width_heights (optional, plot_type = 2, 3)**: the heights (y coordinate) of the peaks' FWHMs (height of FWHMs) (useful for graphic reasons) of the inserted spectrum, it is a Python list.
 - **left_ips, right_ips (optional, plot_type = 2, 3)**: x coordinates (python lists) of the FWHMs of the peaks (beginning and end of the width of the peak, useful for graphic reasons) of the inserted spectrum.
 - **df_comp (optional, plot_type = 3)**: Pandas data-frame representing the wavelengths and relative intensities of every point in the second spectrum to be overlapped with the first. The first column is reserved for wavelengths, the second for intensities. The values in the columns must be NumPy arrays.

- **df_comp_name** (optional, **plot_type = 3**): name of the spectrum to be overlapped with the first.
- **df_comp_params** (optional, **plot_type = 3**): dictionary contains the locations of the peaks (wavelengths), the height of the FWHMs, and FWHMs coordinates. The dictionary keys are:
 - **locations**: location of detected peaks in nm of the inserted spectrum, NumPy array.
 - **width_heights**: the heights (y coordinate) of the peaks' FWHMs (height of FWHMs) (useful for graphic reasons) of the inserted spectrum, it is a Python list.
 - **left_ips, right_ips**: x coordinates (python lists) of the FWHMs of the peaks (beginning and end of the width of the peak, useful for graphic reasons) of the inserted spectrum.

Chapter 4: Experimental results with FluoDetect and future developments

All the compounds' spectrums' of interest obtained in Chapter 2 were uploaded to FluoDetect and tested against the same compounds obtained in slightly different conditions (concentration, smoothing, presence in the solution of more fluorescent compounds than the one tested against in the database) in order to test the accuracy of the algorithm.

The results have been generally very positive, with the highest score in each simulation always assigned to the proper fluorophore. Sometimes the score could seem to the user/scientist a little low considering the apparent resemblance of two spectrums, like in Fig 3.7.2 and 3.7.3 comparison of two samples of curcumin obtained in slightly different conditions (the inserted spectrum has also a little bit of chlorophyll in the mix in the solution, visible at approximately 700 nm).

Compare spectrums

Excitation wavelength begin: Excitation wavelength end: Dimensionality reduction factor:

Smoothing window: Minimum peak height: File "9.txt" already inserted

pk_num_tol: pk_tol: fwhm_tol: weight_peak: weight_fwhm: weight_shape: num_records_to_show: shape_pt_tol: baseline_pts:

Fluodb

Peak wavelengths [nm]	FWHM peaks [nm]
[542.52]	[82.67]

Elaborated inserted Spectrum

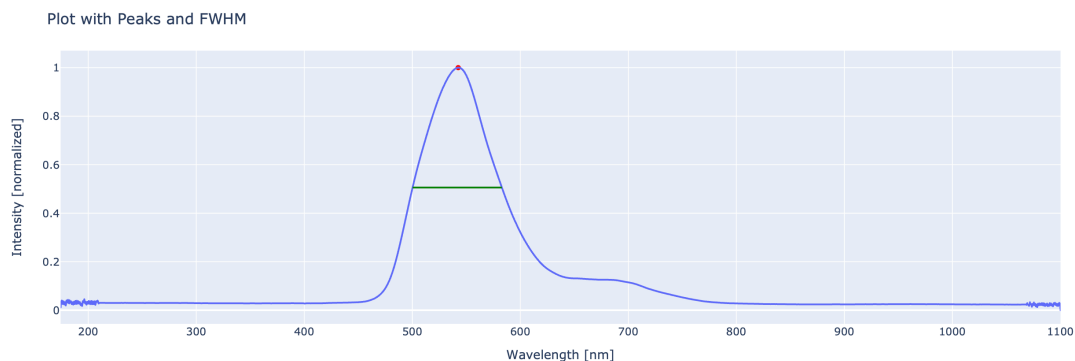


Fig 3.7.2: spectrum inserted.

Name	Peak wavelengths [nm]	FWHM peaks [nm]	Score [%]
Curcumin	[538.55]	[118.07]	77.631886
Inserted spectra	[542.52]	[82.67]	77.631886

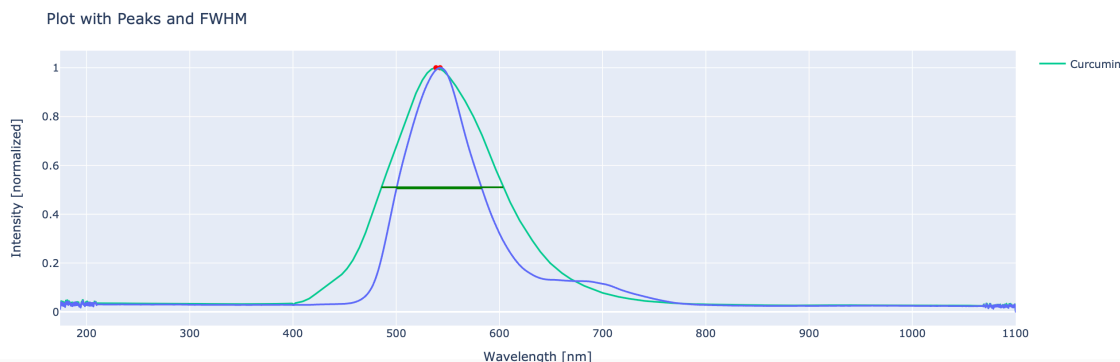


Fig 3.7.3: Example result of comparison algorithm.

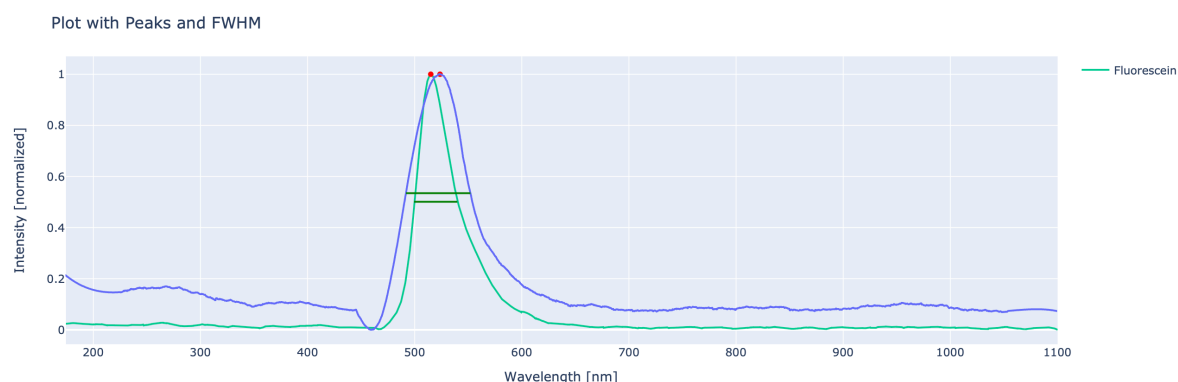
Sometimes (more often than not) this has to do with the comparison parameters inputted (Fig. 3.7.2). Choosing the appropriate comparison parameters improves a lot the comparison

likelihood score, and that requires at the beginning a little bit of tinkering with the software application.

Similar results have been obtained also with fluorescein and chlorophyll. As can be seen in Fig. 4.1 below the comparison with a highly diluted solution of fluorescein+water inputted by the user with other compounds in the database shows that the maximum comparison score has been assigned to fluorescein in the database (70%), meanwhile also a score of 50% has been assigned to curcumin, since the distance between the emission peaks of the two is narrow but they are not similar in shape nor FWHM, hence the lower score.

Comparison Matches

Name	Peak wavelengths [nm]	FWHM peaks [nm]	Score [%]
Fluorescein	[515.03]	[40.4]	70.432284
Inserted spectra	[523.65]	[60.19]	70.432284



Name	Peak wavelengths [nm]	FWHM peaks [nm]	Score [%]
Curcumin	[538.55]	[118.07]	49.435161
Inserted spectra	[523.65]	[60.19]	49.435161

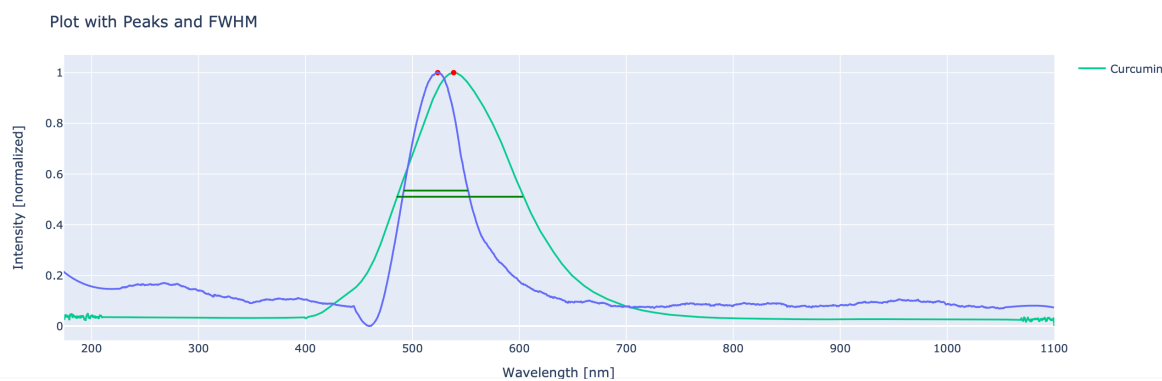


Fig 4.1: screenshot of the comparison's page results.

As it has already been said this result could be even more precise if the comparison options were changed accordingly: as an example in this case more weight could be assigned to fwhm and shape and less to the peak's distance, if $\text{weight_peak}=0.2$, $\text{weight_fwhm}=0.4$ and $\text{weight_shape}=0.4$ then the comparison shows a 68.71% match with fluorescein and 42.42% match with curcumin, making the result more accurate.

Also, the comparison between two samples of chlorophyll+water from different batches of measurements with very different dilutions shows a match of 95% (Fig. 4.2).

Comparison Matches

Name	Peak wavelengths [nm]	FWHM peaks [nm]	Score [%]
Chlorophyll	[679.27]	[80.41]	95.350703
Inserted spectra	[677.65]	[79.48]	95.350703

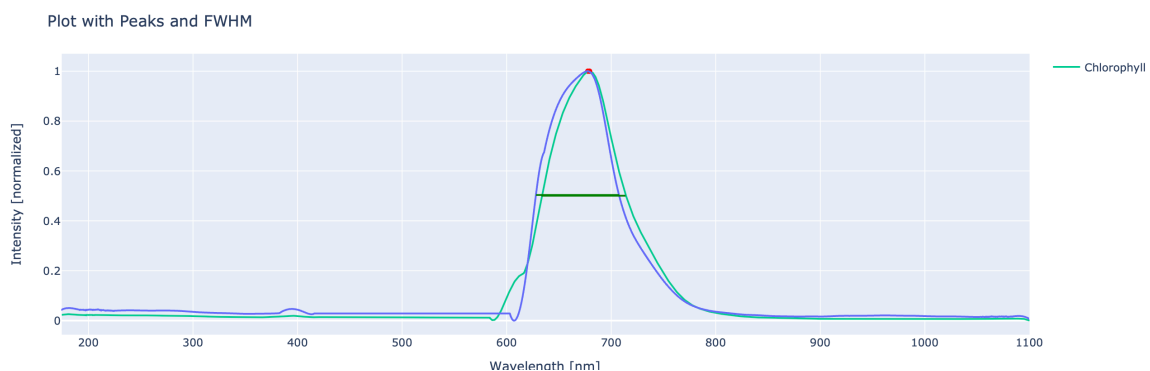


Fig 4.2: Comparison result with a sample of chlorophyll.

Also, the comparison of the two different samples of dairy milk diluted in water in very different concentrations showed a very good comparison result (78.62%, Fig. 4.3). The comparison showed some similarity (51%) with another sample containing curcumin, fluorescein, and chlorophyll together (Fig. 4.4) since there is a similarity in peak wavelength between the two of them (at approx. 532 nm) and there is some similarity in the shape of the two also.

Comparison Matches

	Name	Peak wavelengths [nm]	FWHM peaks [nm]	Score [%]
	Tyrosine Tryptophan and Riboflavin	[361.84, 521.0]	[77.69, 65.67]	78.619739
Inserted spectra		[352.73, 532.93]	[67.70, 68.10]	78.619739

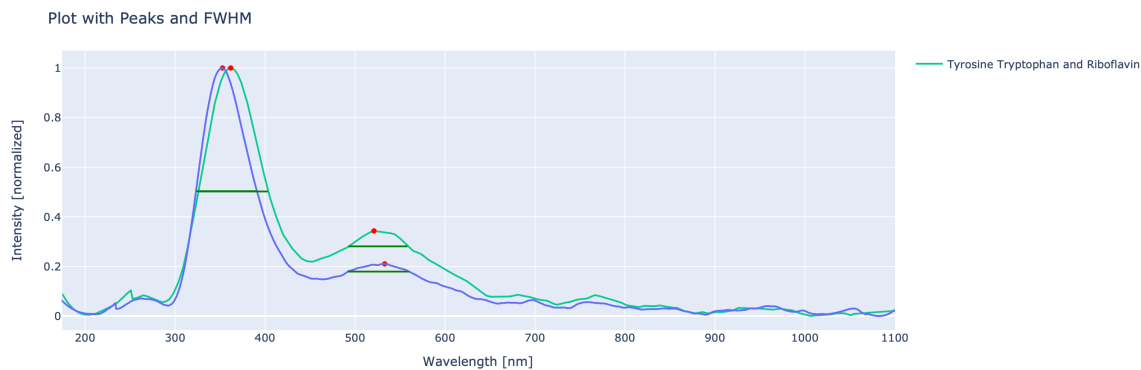


Fig 4.3: comparison result with a sample of dairy milk inserted.

	Name	Peak wavelengths [nm]	FWHM peaks [nm]	Score [%]
	Fluorescin curcumin and chlorophyll	[534.58, 677.0]	[99.68, 54.7]	51.487894
Inserted spectra		[352.73, 532.93]	[67.70, 68.10]	51.487894

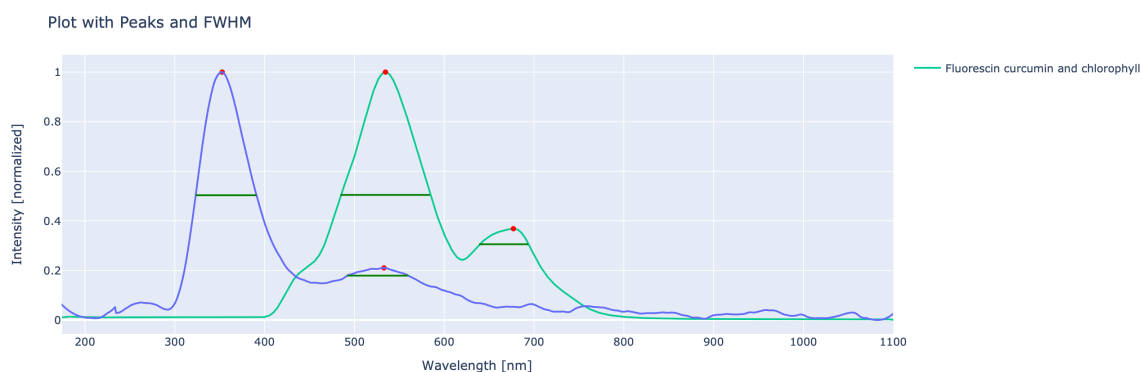


Fig 4.4: Another comparison result with a sample of dairy milk inserted.

Clearly, the correct compound fluorescence composition is the first one (Fig. 4.3), though it is interesting to see how the software can also suggest other compounds based on similarities in other regards that were not noticed at first in the lab.

FluoDetect has not been uploaded on the web, all the results shown in this work have been obtained by a simulation of a web server inside a personal computer. A possible development is uploading a version of it on the web allowing certain users/scientists accredited to upload verified spectrums of fluorescent compounds on the database, in this way

creating a potentially huge database that all users/scientists of the world (even those not accredited) can consult and compare their acquired spectrums with those in the database, de-facto creating an open-source and open-data tool for the community of spectrofluorometry scientists.

Another improvement would be creating a tool to convert each spectrum file to the format required by FluoDetect (see the relative section).

As it has been written before in this section, this software is intended as a beta in the current state of the art (which refers to the date this thesis was published), so there are still many bugs to fix in this application that are left to fix to good-willed and interested people of this software application to fix.

Chapter 5: Possible implementation of an EWS on the field

A possible EWS system (Early Warning System) for the detection of contamination happening in water/soil using spectrofluorometry could have the following basic setup (Fig. 5.1):

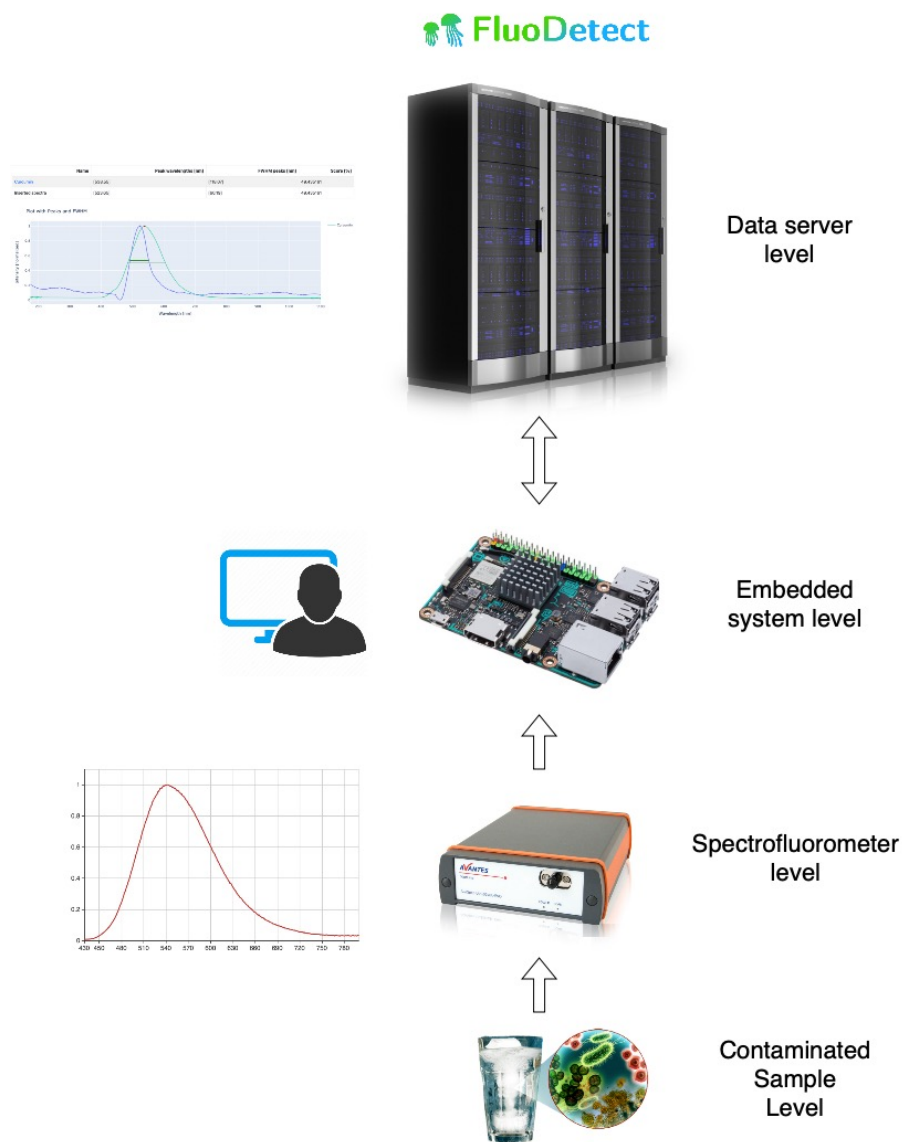


Fig 5.1: the EWS setup schematised.

- A spectrofluorometer to detect fluorescence and acquire the spectrum of fluorescence, this can be a portable and light instrument like

Avantes, described before in Chapter 2, or even a smaller device if possible by the current state of the art.

- An optical sensor, like an optical fiber, to allow the spectrofluorometer to collect the light from which the instrument will draw the spectrum.
- An excitation source, like an LED, to excite the object's (liquid or solid) fluorophores and in doing so permit to detect fluorescence.
- A personal computer or embedded system capable of receiving the spectrums from the spectrofluorometer. The device requirements on computational power, memory, and storage are very low, but the device needs to be connected to the internet or intranet (networking capabilities) to send the spectrum acquired to a data server running the FluoDetect software. In this way, all the computational, memory, and storage requirements are on the data server instead of the embedded system on the field.
- A data server with enough computational power, memory, and storage capabilities to run FluoDetect with all the data the user/scientist plans to store.

The whole setup should work like this:

- The embedded system acquires periodically every n seconds/minutes spectrums from the fluorometer.
- Sends to the data server running FluoDetect the acquired spectrum.
- The server running FluoDetect compares the received spectrum with those in the database and returns the results.
- If the server finds a fluorophore or a mix of fluorophores that indicates a possible contamination (like a bacterium) and the compatibility score with the contaminant exceeds a certain threshold it sends back to the embedded device a warning for a possible contamination happening.
- The user/scientist can then analyze the results if connected to the embedded system, the embedded system can either have a screen or allows to connect a third device to it (like a smartphone or tablet) which has a screen to visualize the comparison results. At this point,

the EWS system can either act alone and prevent the contamination by ways of for example using a powerful LED to disinfect the object if it is programmed to do so or ask for human intervention.

Chapter 6: Conclusions

Fluorescence is an interesting and complex topic, many factors influence it, to cite a few: concentration of the solution, pH, temperature, nature of the solvent, and nature of the solute (fluorophore). Moreover, each compound reacts differently to the environmental factors cited, since fluorescence is strictly correlated to the chemical nature of the compound and how it physically/chemically interacts with its environment. The goal behind this research was to create a tool that could recognize a compound based on its fluorescence, which like it has been said depends on many factors. After recognising the complexity behind each fluorophore fluorescence in the laboratory experiments it was concluded that the only way for which such a tool could work is by creating a data-driven software tool that uses previously collected data on the spectrums of the fluorophores stored in a database by the scientist conducting the research or other scientists of the field. The spectrums and data stored already in the server running the software tool are then compared via a comparison algorithm that uses as comparison parameters:

- The peak(s) wavelength(s) of the spectrums
- The FWHM(s) of the peak(s) of the spectrums
- The shape (geometrically how the two 2D plots are similar) of the spectrums

To detect if the compound's spectrum inserted by the scientist/user is a match with one in the database, this way possibly allowing the detection of any fluorescent compound.

In Chapters 3 and 4 this software tool (FluoDetect) was presented, in Chapter 5 a possible hardware and software setup which could allow the fast detection on the spot of contamination by bacteria happening in water/soil was also presented. The realization of an EWS (Early warning system) like that presented would be a game-changer in preventing the proliferation of bacteria in critical environments such as hospitals since the detection of the contamination would be almost instantaneous and preemptive measures could be taken both by humans or other electronic/

optical devices, like UVC sanitization LEDs that could remove the bacteria using optical power.

Bibliography

1. Lack, T., *Water and health in Europe: an overview*. BMJ, 1999. **318**(7199): p. 1678-82.
2. Decker, B.K. and T.N. Palmore, *The role of water in healthcare-associated infections*. Curr Opin Infect Dis, 2013. **26**(4): p. 345-51.
3. Singh, P., et al., *A review on spectroscopic methods for determination of nitrite and nitrate in environmental samples*. Talanta, 2019. **191**: p. 364-381.
4. Ou, F., et al., *Rapid and cost-effective evaluation of bacterial viability using fluorescence spectroscopy*. Analytical and Bioanalytical Chemistry, 2019. **411**(16): p. 3653-3663.
5. Carstea, E.M., et al., *Fluorescence spectroscopy for wastewater monitoring: A review*. Water Research, 2016. **95**: p. 205-219.
6. Lakowicz, J.R., *Principles of Fluorescence Spectroscopy*. 2006, Boston, MA: Springer US.
7. Baird, C.S.A., McGraw Hill, Sept. 2019., *Absorption of electromagnetic radiation*.
8. Jim Clark, T.S.i.C., *What causes molecules to absorb UV and visible light*.
9. Bassani, G.F., G.L. Liedl, and P. Wyder, *Encyclopedia of condensed matter physics*. 1st ed. Gale virtual reference library. 2005, Amsterdam: Elsevier.
10. Varkonyi, Z. and K. Kabok, *Effect of temperature on light-absorption and fluorescence of the peroxidase*. Acta Biochim Biophys Acad Sci Hung, 1975. **10**(1-2): p. 129-37.
11. <https://micro.magnet.fsu.edu/primer/java/jablonski/solventeffects/index.html>.
12. Lakowicz, J.R., J.R. Lakowicz, and C. Ebook Central Academic, *Principles of Fluorescence Spectroscopy*. 2nd 1999. ed. 1999, New York, NY: Springer US : Imprint: Springer.

13. Le Guern, F., et al., *Fluorescein Derivatives as Fluorescent Probes for pH Monitoring along Recent Biological Applications*. Int J Mol Sci, 2020. **21**(23).
14. <https://www.avantes.ru/en/spectrometer/avaspec3648.php>.
15. <https://www.britannica.com/science/chlorophyll>.
16. <https://www.photochemcad.com/databases/common-compounds/chlorins-bacteriochlorins/chlorophyll-a>.
17. <https://www.compoundchem.com/2016/11/30/turmeric/>.
18. <https://www.ncbi.nlm.nih.gov/books/NBK555957/#:~:text=Fluorescein%20is%20a%20diagnostic%20contrast,c.c.i.i.s>.
19. Robert Sjöback 1, J.N., Mikael Kubista, *Absorption and fluorescence properties of fluorescein*. 18 April 2000.
20. Shlosberg, Y.e.a.F.l.-f.i.o.b.b.s.f.o.a.a.-a.a.b.c., SpringerLink. Available at: <https://link.springer.com/article/10.1007/s00216-021-03642-8> (Accessed: 19 March 2024).
21. Ramanujam VS, N.F., Anderson KE, Kuo YF, Chen NW, Ju H, Lu LW. Riboflavin as an independent and accurate biomarker for adherence in a randomized double-blind and placebo-controlled clinical trial. Biomarkers. 2017 Sep;22(6):508-516. doi: 10.1080/1354750X.2016.1269201. Epub 2016 Dec 23. PMID: 27918196; PMCID: PMC5613754.
22. Amani, M.B., Ali & Parvin, Parviz. (2022). Optical characterization of the liver tissue affected by fibrolamellar hepatocellular carcinoma based on internal filters of laser-induced fluorescence. Scientific Reports. 12. 6116. 10.1038/s41598-022-10146-7.
23. Wong, D. *Fluorescence and Phosphorescence*, [https://chem.libretexts.org/Bookshelves/Physical_and_Theoretical_Chemistry_Textbook_Maps/Supplemental_Modules_\(Physical_and_Theoretical_Chemistry\)/Spectroscopy/Electronic_Spectroscopy/Fluorescence_and_Phosphorescence](https://chem.libretexts.org/Bookshelves/Physical_and_Theoretical_Chemistry_Textbook_Maps/Supplemental_Modules_(Physical_and_Theoretical_Chemistry)/Spectroscopy/Electronic_Spectroscopy/Fluorescence_and_Phosphorescence).

24. Ferrari, L.R., Luigi & Fabbri, Paola & Pilati, Francesco. (2012). Disposable Fluorescence Optical pH Sensor for Near Neutral Solutions. *Sensors* (Basel, Switzerland). 13. 484-499. 10.3390/s130100484.
25. Franck-Condon-diagram.png: Mark M. Somozaderivative work: Jacopo Werther, C.B.-S.h.c.o.l.b.-s., via Wikimedia Commons.
26. Image by Sobarwiki - Own work, P.D., <https://commons.wikimedia.org/w/index.php?curid=29474504>.
27. <https://www.olympus-lifescience.com/en/microscope-resource/primer/java/jablonski/solventeffects>.
28. <https://www.photochemcad.com/databases/common-compounds/biomolecules/l-tryptophan>.
29. <https://www.photochemcad.com/databases/common-compounds/biomolecules/l-tyrosine>.
30. <https://www.photochemcad.com/databases/common-compounds/polyenes-polyynes/curcumin>, *Curcumin spectrums by PhotoChemCad*.
31. <https://www.photochemcad.com/databases/common-compounds/xanthenes/fluorescein>, *Fluorescein spectrums by PhotoChemCad*.

Acknowledgments

I would like to express my gratitude to my thesis advisor, Prof. Guido Perrone, and my tutor Chiara Bellezza Prinsi for their guidance throughout this experience. They provided me with laboratory equipment and invaluable expertise in its use.

I would also like to extend my thanks to my family, to my father Maurizio, who has always given me invaluable life and university advice, and to my mom Ornella who has always supported and cherished me at every step of this journey. I am who I am today thanks to them.

I also would like to thank my dearest friends with whom I have shared this journey: Daniele, Matteo, Giovanni, Raffaele, Benjamin, and Enrico. These years in Turin have been so amazing thanks to you, the time we have spent together will be forever in my memory as one of the best times of my life.

I could not conclude this section without thanking my sweet love, Amanda, with whom during this past year I have found true love and companionship. I am thrilled about our new life in Finland together and for all the new adventures that await us in the future. *Rakastan sinua kulta.*