EAC Internship - ML - Caves Pangaeas

Nicolas DAMAGEUX

February - July 2025

Contents

Acknowledgements

First of all, I would like to thank the entire CAVES&PANGAEA team at the European Space Agency (ESA), led by Loredana Bessone and Samuel Payler, who gave me a warm welcome and supported me throughout my internship. I would especially like to thank Igor Drozdovskiy, who placed his trust in me and gave me the freedom to work on a wide range of topics. I am very grateful for your support and all your advice. Finally, thank you to the team at the European Astronaut Center (EAC), with whom I shared many moments that made my experience unforgettable.

Chapter 1

Introduction

This internship took place during my gap year and led me to work as a intern in Artificial Intelligence at the European Astronaut Center (EAC). Its objective was to continue the research and development work on the CAVES&PANGAEA team's tools, specifically in the area of Machine Learning. I therefore participated in numerous tasks, from maintaining the work environment to putting models into production. Most precisely, I worked on two main subjects: the research and implementation of new Machine Learning models for planetary minerals recognition using spectroscopic methods, and the development and deployment of such models in the context of the PANGAEA 2025 astronaut trainig session.

In this report, I will first present the organization that welcomed me, their missions, and specifically the missions of the team I joined, CAVES&PANGAEA, notably their use of Machine Learning to train astronauts for space exploration. I will then present my work on the two topics mentioned above, which represents the core of my contribution to this team.

1.1 Context

1.1.1 The company

The European Space Agency (ESA) is Europe's gateway to space. Its mission is to shape the development of European space capabilities and ensure that European citizens continue to benefit from investments in space. The ESA is an international organization with 23 member states. By coordinating the financial and intellectual resources of its members, the ESA can undertake programs and activities that go far beyond what any of these countries could achieve on their own.

ESA's mission is to develop and implement the European space program. The Agency's projects are designed to learn more about Earth, its immediate space environment, the Solar System, and the Universe, as well as to develop satellite technologies and services and promote European industries.

The European Astronaut Center (EAC) is the ESA center located in Cologne, Germany, and is the European center of excellence for astronaut training, operations, and space medicine. It welcomes and equips all European astronauts for their flights to the International Space Station (ISS) while preparing for the future of space exploration beyond Earth's orbit. The EAC's main missions are as follows:

- select and train the ESA astronauts;
- keep the astronauts in shape and ready to fly;

- train the astronauts from international partners (NASA, Rocosmos, CSA, JAXA) to use the ISS tools;
- supporting and communicating with astronauts throughout their missions, from departure to return, including their time in orbit;
- exploring new concepts and technologies for the future of space exploration at their research center with their interns and researchers;

My internship therefore fell under this last theme.

1.1.2 The team

During my internship, I was part of the CAVES&PANGAEA team, a training team dedicated to preparing astronauts for geology and outdoor exploration tasks (PANGAEA) and for working in multicultural teams in hazardous environments such as caves (CAVES).

The PANGAEA program allows European scientists and geologists to teach astronauts the basics of geology and astrobiology, thereby providing them with the skills they need to identify and document relevant samples in the field and communicate their findings to scientists on Earth using effective and geologically accurate language. The CAVES program teaches astronauts to explore underground systems as a team. They travel deep underground to conduct scientific experiments, map the environment, and document their activities. This forces them to adapt to living and working in a unique environment as a team. It therefore simulates space travel and allows seasoned astronauts to share their knowledge with trainee astronauts on Earth.

I was personally involved in the research side of the department, specifically in developing the tools used in the field by Machine Learning research. This part of the team was created in 2016, when the agency's geologists decided to create tools for in-situ mineral recognition using portable spectrometers. The work carried out is divided into three parts: the machine learning aspect, the mineralogical database aspect, and the embedded tool aspect (Electronic Field Book - EFB).

1.1.3 The intenship

Since 2016, the research team integrated into CAVES PANGEA has been carrying out important work for the development of astronaut training processes. Machine learning tools have developed considerably since the start of the project. Given the different types of data collected or obtained through self-service, various tasks have been studied over the years by the project's various interns and researchers.The first and most important task is the classification of minerals using portable spectrometers, using the following types of spectra:

- Visible and Near-Infrared (VNIR) spectrum: spectrum obtained by measuring the reflectance of minerals at visible and near-infrared wavelengths.
- Raman spectrum: spectrum obtained by the inelastic scattering of a laser beam interacting with molecular vibrations. Each energy shift (Raman shifts) measured corresponds to a characteristic vibrational signature of the mineral.

Using spectra from open-source datasets and measurements taken by the ESA, numerous avenues were explored with regard to classification, ranging from simple models (Decision Tree, Random Forest, SVM) to neural models (MLP, CNN) and architectures such as Iterative Extreme Learning Machines and Two-level classifiers. Some types of models also used high- and low-level data fusion. The main objectives of this research project were to find high-performance, compact models that could be imported into

the embedded tool developed in parallel, but also to understand which types of data provide the most information about minerals and why.

The second task is then to detect chemical elements and their abundances using the following types of spectra:

- Laser-induced breakdown spectroscopy (LIBS): spectrum obtained by exciting a plasma created on the surface of the sample by a pulsed laser. The emission lines observed correspond to the chemical elements present in the material.
- X-ray fluorescence (XRF) spectrum: spectrum obtained by exciting the atoms in the sample under the effect of incident X-ray radiation. The X-rays emitted reveal the characteristic lines of the chemical elements present.

As this data is rarer, less work has been done on these types of spectra, but a study of different models has already been carried out to determine the presence of elements in minerals. For LIBS spectra, the use of real data has made it possible to determine that by using simple models (SVM, Random Forest) it was possible to detect elements, and that it was also possible to use the results of this task to improve the results of mineral classification. With regard to XRF spectra, it has been established that this type of data allows for the detection of minerals as well as the abundance of these minerals; however, the lack of data has prevented further study. The scarcity of this type of data has also led to the generation of synthetic data for both LIBS and XRF.

1.2 Completed work

As previously stated, this internship gave me the opportunity to participate in two main tasks. First, as I arrived just before the PANGAEA 2025 training exercise in Norway, I was tasked with preparing a Convolutional Neural Network model imported onto the EFB for mineral identification. This was my introduction to the machine learning infrastructure that had been developed and its use.

I then turned my attention to a research project using data fusion (RAMAN + LIBS). My main objective was to highlight solutions that would improve the performance of previous data fusion models based on innovative practices found in the literature. As part of this study, which I will detail later, I had to look at two-level classification models, models such as Extreme Learning Machines, and intermediate fusion via Multimodel Transfer Modules.

Chapter 2

The PANGAEA 2025 astronaut training

2.1 Introduction

To understand the technical challenges of this project, it is first necessary to describe what a traverse is in the context of astronaut training at the EAC. Three astronauts and reserve astronauts from both ESA and NASA have been selected in order to participate to the PANGAEA 2025 training, located in the Lofoten Islands (Norway). The Lofoten Islands are one of the few places in the world with geological features similar to those found in bright, heavily cratered lunar regions, the lunar highlands. Thus, this place offers a wide variety of advantages when it comes to manipulating moon-like geological samples.

Figure 2.1: NASA astronaut Jessica Wittner and astronaut reserve member Arnaud Prost analyze a rock using a portable spectrometer.

The EFB mentionned earlier is brought to such locations, and is linked to a portable spectrometer by a bluetooth connection. The spectrometer is able to scan VNIR spectrum (see 1.1.3) between $\approx 950\ nm$ and $\approx 1650\ nm$, and the bluetooth connections allows the user to get the scanned spectra on the EFB for further processing (display, zoom, annotation, and of course Machine Learning classification). Eventually, the main goal is to provide the user (astronaut/instructor) with insightful data like mineral composition, or at least the endmember name. Stakes like computing time, portability, and performance were at the heart of this project.

2.2 Scientific approach

2.2.1 Development

Although the EFB is running in `Javascript`, ML-V3 (which is the Machine Learning project) offers Python scripts that allow the user to export the ML model and all relevant data using modules such as `TensorFlow.js`. Currently, the EFB is using version `4.17` of `TensorFlow.js` (listed as `tensorflowjs` in the ML-V3 requirements for Python 3.10, updated in the EFB for the Lofoten model). For confidentiality reasons, the architectures of ML-V3 and the EFB will not be detailed in this report. However, I will detail the various tasks I had the opportunity to work on from a development perspective. These tasks not only took up a large part of my time, but also taught me a lot in terms of technical skills and personal development.

2.2.2 Exporting the Machine Learning models

When buliding a Machine Learning model, whether it is a simple perceptron or a Convolutional Neural Network, the architecture (layers, weights, activation functions, etc.) is not sufficient to deploy and use the algorithm on new data. In fact, a lot of additional data is needed, which will henceforth be referred to as *metadata*. This includes:

- the preprocessing steps ;
- the mineral identifiers ;
- the dataset filters ;

Moreover, in the context of astronaut training or scientific expeditions, data such as the model performance are relevant for instructors, and it might be useful to share them while exporting the Machine Learning model.

The development and export pipeline proposed when I arrived (ML-V3) was relatively easy to use but was obsolete in terms of deploying these models. In particular, it was impossible to transform the metadata so that it would be accepted by the EFB (inconsistent JavaScript versions, incompatible label translations, etc.). In addition, a lot of information was lost, such as filters, certain preprocessing steps, and performance data. So, in addition to my research, I had to spend many hours debugging the entire pipeline to implement the steps necessary for it to function properly before I could even begin to develop my own ideas. It took me weeks to understand the whole architecture of the pipeline, but I eventually managed to update it, upgrade to few components that needed it, and deploy it on the shared GitHub:

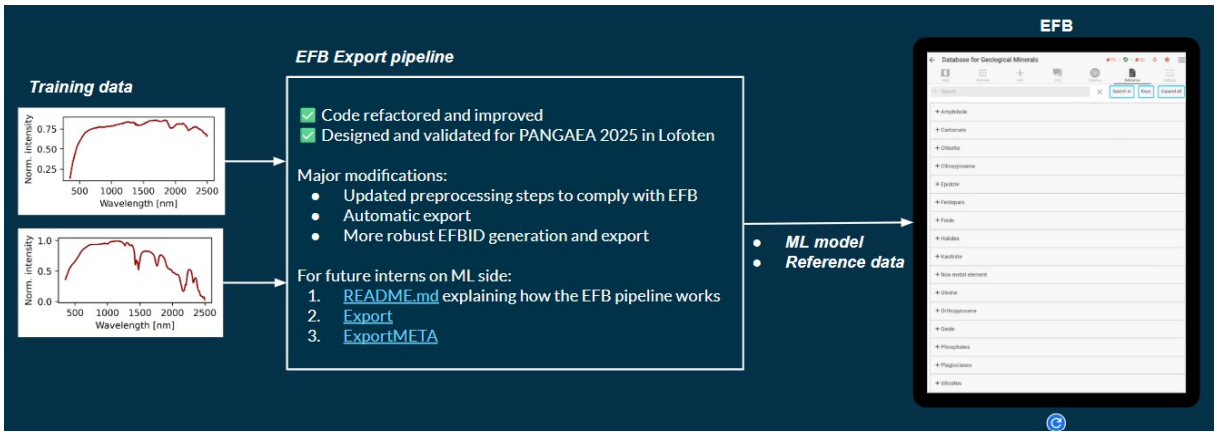


Figure 2.2: Export pipeline architecture

In the end, no human intervention was needed in order to put the Machine Learning models into the EFB, as the export was fully automatic. The models were refactored and translated from `Python` to `Javascript`. As for the endmembers (minerals to classify), they were listed as identifiers, called EFBIDs, which are basically the official geological symbols related to these endmembers translated in hexadecimal (example: Aegirine = 416567). I also took this opportunity to create several reports showcasing the many updates, as well as `Python` scripts that could be used as tutorials for future interns.

2.2.3 Support

Processing the dataset

When creating the dataset, some endmembers are eventually removed from the training set as they may belong to blacklisted minerals or they do not have enough spectra to be trained on. For the Lofoten ML model, the considered minerals are the following:

Table 2.1: Lofoten endmember dataset

Name	blacklisted	kept after filtering	merged as
Aegirine		Yes	Diopside
Albite		Yes	Anorthite
Allanite		No	
Anorthite		Yes	
Augite		Yes	Diopside
Calcite		Yes	
Chalcopyrite		Yes	
Cubanite		No	
Diopside		Yes	
Edenite		No	
Fayalite		Yes	Forsterite
Ferro-pargasite		No	
Ferrosilite		Yes	
Forsterite		Yes	
Grossular		Yes	
Halite	Yes	Yes	
Hematite		Yes	

End on the next page

Name	blacklisted	kept after filtering	merged as
Hornblende		Yes	
Ilmenite		Yes	
Jadite		Yes	
Kaersutite		Yes	Magnesio-hornblende
Kyanite		Yes	
Magnetite		Yes	
Marcasite		Yes	
Marialite		Yes	
Microcline		Yes	
Molubdenite		Yes	
Muscovite		Yes	
Phlogopite		Yes	
Pyrite		Yes	Pyrrhotite
Pyrrhotite		Yes	
Quartz	Yes	Yes	
Rutile		Yes	
Spodumene		Yes	
Sylvite		Yes	
Titanite		Yes	
Valleriite		No	
Zircon		Yes	

As far as the filters/preprocessing steps are concerned, please have a look at this report. Classes (minerals/endmembers) can be merged when their features (VNIR spectra) are close enough on the $[950, 1650]$ nm spectral range, or when they belong to the same mineral families (see this database [1]). For example, **Augite** and **Diopside** have been merged together and had the same target/output. The merged classes were chosen accordingly to the minerals specifically present in Lofoten, using the instructors' insights. Other filters such as `NaN filtering` and removing undersampled classes can be mentionned. Finally, the preprocessing are fairly important to export as they will be applied to scanned spectra in the EFB. Thus, the used preprocessing steps during the development had to be reproducible on the field: **interpolation** (same number of features), **normalization** (local, min-max) and imputation for NaN values.

Training the models

The models have been chosen among several Convolutional Neural Network architectures previously developed in CAVES&PANGAEA, and will thus not be described in this paper (also for confidentiality purposes). Cross-validation has been used to observe the behavior of the model before the final training: **3-fold Stratified K-Fold**. The final confusion matrix and f-scores are shown as part of the report in appendice. No early stopping has been used during the cross-validation, with a batch size of 32 and 50 epochs. As for the final training, the same training parameters have been used, with early stopping (training stopped in 27 epochs):

[1] `https://www.mindat.org/min-50468.html`

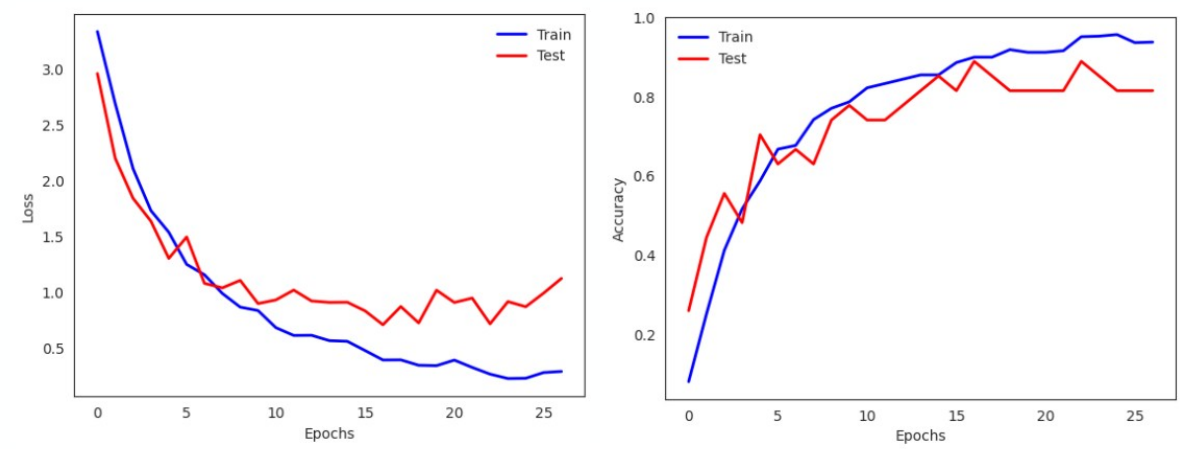


Figure 2.3: Lofoten ML model training

2.2.4 Performance

Even though the model shows an accuracy of 88.1%, both the EFB and the ML model have been tested on real samples: the **validation samples**. For each one of the minerals, 20 scans have been made on random spots in order to increase the variance of the scans. As the EFB does not currently provide an easy access to the computation metrics, we decided to propose a "user guide" graph that sums up the performance of the whole classification pipeline:

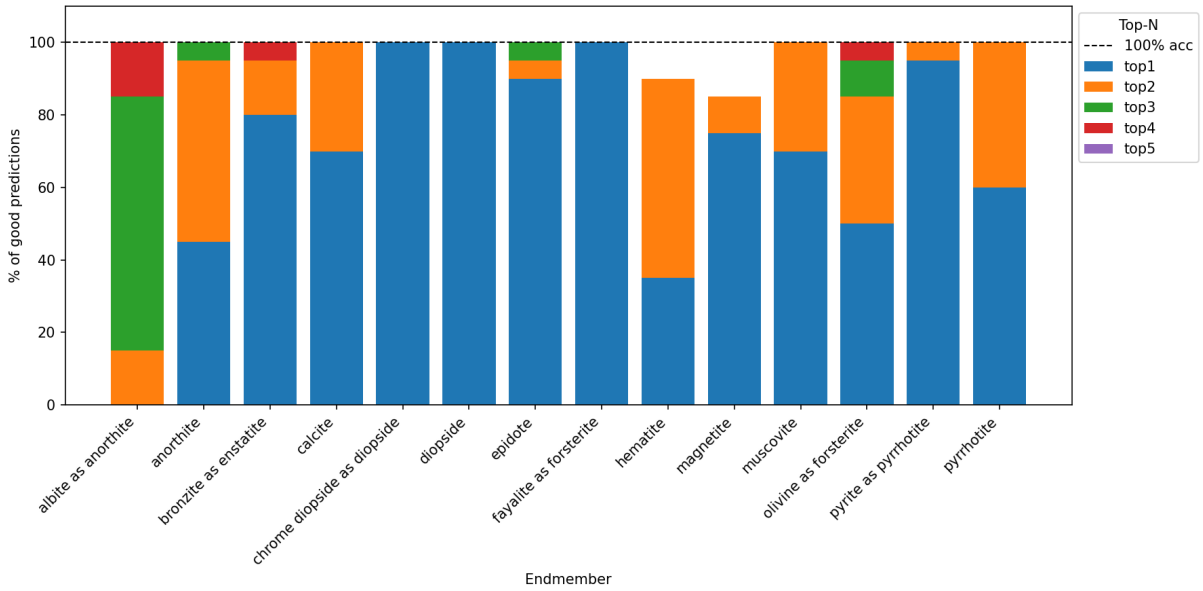


Figure 2.4: Lofoten ML model validation

The figure illustrates the Top-N prediction accuracy of the machine learning model that outputs the five most probable classes (Top-5) for each input. For each endmember (x-axis), the colored bars represent the proportion of correct predictions ranked within the top-1, top-2, up to top-5 predicted classes. The blue segment shows the percentage of samples correctly predicted as the top-1 class, while the orange, green, red, and purple segments correspond to correct predictions at the top-2, top-3, top-4, and top-5 positions, respectively. A dashed line at 100% indicates the maximum achievable accuracy. This

visualization helps assess the confidence and ranking quality of the model's predictions across different minerals or mineral groupings. In some cases (e.g., "albite as anorthite"), the correct label is not ranked first but still appears within the top-3 or top-5, highlighting the importance of evaluating beyond top-1 accuracy when working with closely related classes.

2.3 Conclusion

The final model has been exported to the EFB and successfully used in the PANGAEA 2025 astronaut training (see [2] for furhter information on the training). This experience has been really rewarding as it led me to take a lot of decisions on my own, thus giving me a great sense of responsibilities. Moreover, I had to explain my choices and actions not only to my supervisors and colleagues, but also to the instructors who would later train the astronauts using my work. They gave me a huge amount of useful insights, and it truly boosted my confidence.

[2] https://www.esa.int/Space_in_Member_States/France/PANGAEA_2025_les_astronautes_etudient_la_geologie_lunaire_dans_un_fjord

Chapter 3

Advancing classifiers for PHOENIX spectrometers

3.1 Introducing data fusion for the PHOENIX instrument

The PHOENIX instrument is a new tool developped specifically for planetary missions in order to scan samples using both RAMAN and LIBS spectroscopy [8]. Using data fusion allows the developers to improve performance by adding independant data to the input features. Indeed, both RAMAN and LIBS spectroscopy operate in different scales (the first on the molecular scale, the second at the atomic level), thus multiplying the number of relevant data. In this context, data fusion has already been used at CAVES&PANGAEA [3][2][4], yet the goal of this research project was to advance some of the current classifiers specifically for the PHOENIX spectrometers.

3.2 MTMM: Multimodal Transfer Module for CNN Fusion

The Multimodal Transfer Model for CNN Fusion (MTMM) is a deep learning approach designed to integrate and leverage information from multiple data modalities using convolutional neural networks (CNNs). In multimodal settings, each modality provides unique and complementary features, but effectively combining them requires careful alignment and fusion strategies [1]. MTMM addresses this challenge by learning shared and modality-specific representations, allowing the model to transfer knowledge between modalities and improve overall performance [6].

The MTMM architecture typically involves separate CNN branches for each modality to extract initial features, followed by fusion layers where the information is combined—either through concatenation, attention mechanisms, or more advanced transfer modules. These transfer modules are key components that allow the model to share useful patterns across modalities while preserving the distinct characteristics of each input. This structured fusion not only enhances feature richness but also improves generalization, particularly in scenarios with limited data or unbalanced modality quality. In this case, we concatenate each feature the following way:

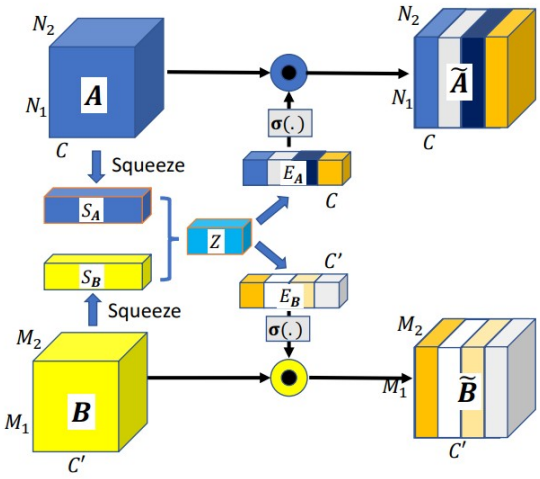


Figure 3.1: MTMM feature fusion pipeline

The inputs to the MMTM layer are two feature tensors, A and B, each with dimensions corresponding to batch size, sequence length, and number of channels. The objective is to compute scaled versions of these tensors, denoted as $\tilde{A}$ and $\tilde{B}$, by leveraging a learned cross-modal attention mechanism that captures interdependencies between the two inputs.

First, a global average pooling operation is applied independently to A and B along their sequence dimension. This operation reduces each tensor from shape (batch_size, sequence_length, C_1) and (batch_size, sequence_length, C_2) to vectors of size (batch_size, C_1) and (batch_size, C_2), respectively. The resulting vectors, denoted S_A and S_B, summarize the global channel-wise information from A and B.

Next, the vectors S_A and S_B are concatenated along the channel dimension to form a joint feature vector $Z \in R^{batch\_size \times (C_1+C_2)}$, which is then passed through a fully connected layer with ReLU activation to produce a lower-dimensional embedding Z'. From this embedding, two separate fully connected layers generate attention vectors $E_A \in R^{C_1}$ and $E_B \in R^{C_2}$, corresponding to the original channel dimensions of A and B. These vectors are passed through a sigmoid activation function scaled by a factor of 2 to obtain scaling factors:

$$scale_A = 2 \cdot \sigma(E_A), \quad scale_B = 2 \cdot \sigma(E_B)$$

where $\sigma(\cdot)$ denotes the sigmoid function. Finally, these scaling factors are expanded along the sequence dimension and applied element-wise to the original inputs A and B, resulting in the recalibrated feature tensors:

$$\tilde{A} = scale_A \odot A, \quad \tilde{B} = scale_B \odot B$$

where $\odot$ denotes element-wise multiplication broadcasted along the sequence length. This operation adaptively recalibrates the feature maps, enhancing their representational power by incorporating complementary information from the other modality. First results tend to show that for classic CNNs (same architecture for both RAMAN and LIBS), adding MMTM dicreases the overall accuracy. However, it is very likely that such layers might induce some improvements on more complex models (image segmentation + RAMAN, etc. as suggested in [6].

3.3 Incremental Extreme Learning Machines

3.3.1 Introduction

Extreme Learning Machines (ELMs) are a type of feedforward neural network specifically designed for fast and efficient learning, particularly in classification tasks [5][7]. Unlike traditional deep learning models such as Convolutional Neural Networks (CNNs), which rely on iterative backpropagation and complex layer structure, ELMs use a single hidden layer where input weights are randomly assigned and remain fixed. The training process focuses solely on determining the output weights, which can be computed analytically, resulting in signficantly faster training times.

Incremental Extreme Learning Machines (I-ELMs) design a subtype of ELMs that are built iteratively in order to tune the number of hidden layer nodes [7]. The number of hidden layer nodes keeps increasing from 1 to L_{max} until the expected error is less than the expected ϵ, and the output weight is given by:

$$\beta_n = \frac{E.H^T}{H.H^T} \tag{3.1}$$

where E is the residual vector and $H = J(x, a, b)$ is the output of the hidden layer x (following the use of an activaion function J on the random weights a and bias b).

3.3.2 I-ELM for mineral endmember classification using RAMAN and LIBS

It has been shown that Kernel Extreme Learning Machines may offer good results when classifying endmembers using both RAMAN et LIBS spectra [2]. The described approach suggests that using **attention mechanisms** to select features/wavelengths offers enough insights on the related endmember so that the used ELM can accurately classify incoming spectra into labels. After a few "classic" preprocessing steps (baseline removal, normalization), a peak/feature identification is done using Fisher score, and the N most important wavelenghts are kept, thus reducing to input size from $N_{spectra}$ (intitial resolution) to N 3.2.

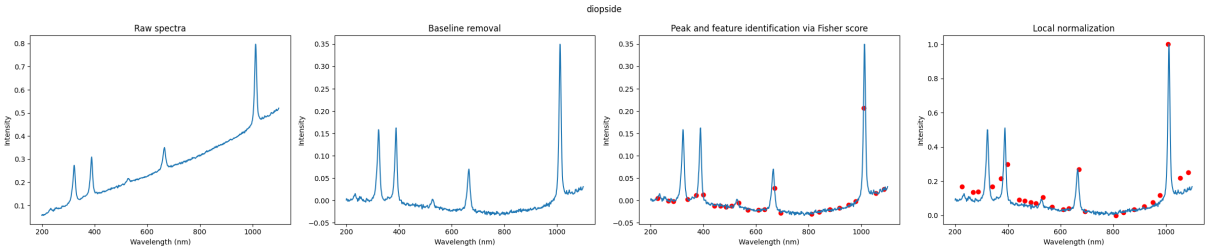


Figure 3.2: Peak and feature extraction using Fisher Score (applied to a RAMAN spectra of diopside)

This offers a significant advantage when it comes to model scaling, as the little number of hidden layer nodes (maximum L_{max}) and the very few features N reduce the size of the whole ELM when compared to classic CNNs. Following this first step, the chosen inputs can be the N intensities $[I_i = S(\lambda_i)]_{i=1}^{N}$, the average spectra over a specific window of width $d\lambda_i$, $[\overline{I_i} = \overline{S(\lambda_i)}]_{i=1}^{N}$, or the integral of the spectrum centered at λ_i over a window of width $d\lambda_i$, $[A_i]_{i=1}^{N}$:

$$A_i = \int_{\lambda_i - d\lambda_i/2}^{\lambda_i + d\lambda_i/2} S(\lambda) d\lambda \tag{3.2}$$

The use of a controllable window $d\lambda_i$ introduces an additional degree of freedom to the algorithm, acting as a tunable hyperparameter. This parameter defines the width of the spectral region considered around each wavelength λ_i, directly influencing how the spectral information is aggregated—either

through averaging or integration. By adjusting $d\lambda_i$, it becomes possible to control the level of spectral smoothing and reduce local noise, which can be particularly beneficial in cases where the raw spectra are noisy or contain high-frequency fluctuations. Moreover, this flexibility allows the algorithm to adapt to different types of spectra: narrower windows preserve fine spectral features that might be relevant for certain classes, while wider windows emphasize broader patterns or peak envelopes. Ultimately, incorporating $d\lambda_i$ as a control parameter enhances the adaptability of the model and opens the door to performance optimization through data-driven tuning.

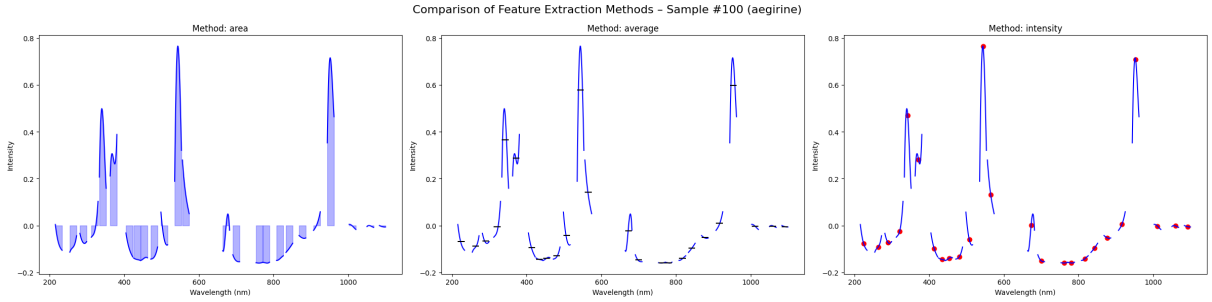


Figure 3.3: Feature preprocessing

3.3.3 Results

In this experiment, the following endmembers are considered: **actinolite**, **aegirine**, **albite**, **almandine**, **andradite**, **annite**, **anorthite**, **augite**, **calcite**, **celsian**, **diopside**, **edenite**, **enstatite**, **fluorophlogopite**, **grossular**, **grunerite**, **hedenbergite**, **magnesite**, **microcline**, **muscovite**, **orthoclase**, **phlogopite**, **polylithionite**, **pyrope**, **rhodochrosite**, **riebeckite**, **siderite**, **smithsonite**, **tremolite**, and **uvarovite**. The dataset is composed of 1800 spectra (60 per class).

To create a specific I-ELM model for RAMAN+LIBS classification, the following parameters have to be tuned/optimized:

- L_{max} the maximum number of hidden layer nodes in the I-ELM, or similarly ϵ the target error
- $d\lambda_{FS,r}$ and $d\lambda_{FS,l}$ the windows used in `scipy.signal.find_peak`[1] to apply the **attention mechanisms**, respectively for the RAMAN and LIBS spectra
- $d\lambda_r$ and $d\lambda_l$ the windows used to compute the area/average intensity (if using one of these methods), fixed for all wavelengths, respectively for the RAMAN and LIBS spectra
- N_{RAMAN} and N_{LIBS}, the number of inputs (see 3.2), respectively for the RAMAN and LIBS spectra

The 6 parameters are chosen in the following ranges ($1 \times 5 \times 5 \times 5 \times 5 \times 4 \times 4 = 10000$ calculations, 5 days to compute):

- $L_{max} \in \{150\}$
- $d\lambda_{FS,r} \in \{20, 40, 60, 80, 100\}$
- $d\lambda_{FS,l} \in \{20, 40, 60, 80, 100\}$
- $d\lambda_r \in \{20, 40, 60, 80, 100\}$

[1] `https://docs.scipy.org/doc/scipy/reference/generated/scipy.signal.find_peaks.html`

- $d\lambda_l \in \{20, 40, 60, 80, 100\}$
- $N_{RAMAN} \in \{10, 25, 50, 100\}$
- $N_{LIBS} \in \{10, 25, 50, 100\}$

Initial intuition suggests that low values of $d\lambda_{FS,r}$ and $d\lambda_{FS,l}$ will produce better results, given that detecting single peaks (rather than clusters) would facilitate attention mechanisms and thus classification. By observing the average accuracy of the model once these two parameters have been set, we can see this confirmed in the following graphs:

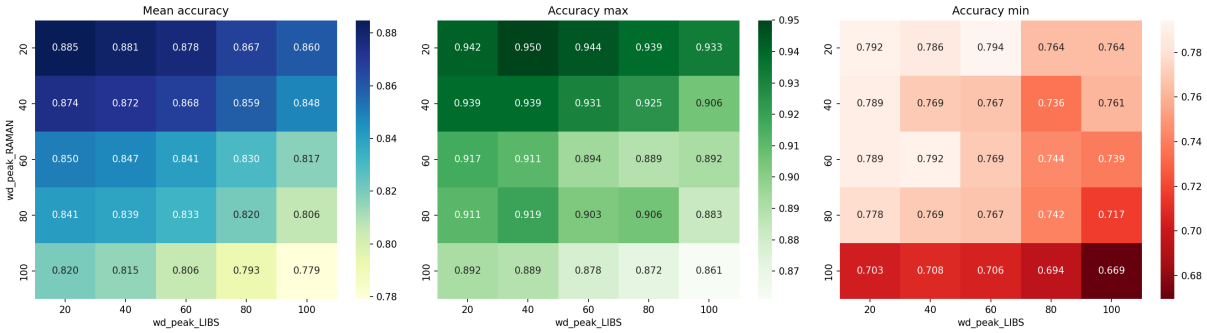


Figure 3.4: Peak window tuning

Once the first two parameters have been fixed, we can have a look at the behaviour of the I-ELM with respect to the last four parameters: $d\lambda_r$, $d\lambda_l$, N_{RAMAN} and N_{LIBS}. In the followuing graph, each heatmap is related to a (N_{RAMAN}, N_{LIBS}) pair, and present the accuracy of the considered ELM as a function of both $d\lambda_r$ and $d\lambda_l$. Considering the random behaviour of ELMs, we choose not to take the maximum accuracy as a decisive parameter for optimization. Eventually, the best averaged accuracy is achieved with the lowest $(d\lambda_{FS,r}, d\lambda_{FS,l})$:

Figure 3.5: ELM optimization results

3.4 Two-level classification

In this section, we use the ELM as an intermediate classifier for a two-level classification algorithm. The objective is to use the small size of the ELM to compute an additionnal information to the RAMAN and LIBS spectra (in that case the **mineral group**) as a third input for early fusion. The ELM parameters are the ones selected in the previous subsection:

- $L_{max} = 150$
- $d\lambda_{FS,r} = 20$
- $d\lambda_{FS,l} = 20$
- $d\lambda_r = 20$
- $d\lambda_l = 60$

- $N_{RAMAN} = 100$
- $N_{LIBS} = 10$

Each one of the 30 minerals is assigned its mineral group:

Mineral group	Mineral endmember
Feldspar	Microcline, Orthoclase, Albite, Anorthite, Celsian
Mica	Fluorophlogopite, Muscovite, Polylithionite, Phlogopite, Trilithionite
Garnet	Almandine, Grossular, Pyrope, Uvarovite, Andradite
Calcite	Magnesite, Siderite, Calcite, Rhodochrosite, Smithsonite
Pyroxene	Aegirine, Augite, Diopside, Enstatite, Hedenbergite
Amphibole	Actinolite, Riebeckite, Tremolite, Grunerite, Edenite

Table 3.1: Mineral groups

In this two-level classification framework, an important design choice lies in the source of labels used to train the CNN at the second stage. Specifically, one can either use the predicted outputs from the intermediate ELM classifier or the actual ground-truth labels. Using the true labels ensures that the CNN is trained on accurate and clean data, which can facilitate faster convergence and potentially higher performance under ideal conditions. However, relying exclusively on the ground-truth labels ignores the possibility of errors introduced by the ELM in real-world scenarios. Alternatively, training the CNN using the predicted labels from the ELM introduces the inherent noise and misclassifications of the intermediate classifier into the training process. While this may initially seem detrimental, this approach can increase the overall robustness of the combined system by allowing the CNN to learn to compensate for the ELM's errors, thus improving generalization and resilience to imperfect intermediate predictions. This trade-off between accuracy and robustness is crucial when designing multi-stage classifiers, especially in practical applications where prediction errors are unavoidable. Our experiments investigate both strategies to assess their impact on the final classification performance.

Table 3.2: Performance metrics for different training methods

Endmember	**True label as CNN input**			**ELM prediction as input**			**Support**
	Precision	**Recall**	**F1-score**	**Precision**	**Recall**	**F1-score**	
0	0.973	1.000	0.986	1.000	1.000	1.000	36
1	1.000	0.972	0.986	1.000	1.000	1.000	36
2	0.973	1.000	0.986	1.000	1.000	1.000	36
3	1.000	0.972	0.986	1.000	1.000	1.000	36
4	1.000	0.944	0.971	1.000	0.917	0.957	36
5	0.968	0.833	0.896	0.795	0.972	0.875	36
6	1.000	0.972	0.986	1.000	0.944	0.971	36
7	1.000	0.944	0.971	1.000	0.944	0.971	36
8	1.000	0.917	0.957	1.000	0.917	0.957	36
9	1.000	1.000	1.000	1.000	1.000	1.000	36
10	1.000	0.972	0.986	1.000	1.000	1.000	36
11	1.000	0.889	0.941	1.000	0.972	0.986	36
12	1.000	0.972	0.986	1.000	1.000	1.000	36
13	1.000	0.944	0.971	1.000	0.944	0.971	36
14	0.947	1.000	0.973	0.923	1.000	0.960	36
15	0.900	0.750	0.818	1.000	0.778	0.875	36
16	0.973	1.000	0.986	0.947	1.000	0.973	36

(end next page)

Endmember	True label as CNN input			ELM prediction as input			Support
	Precision	Recall	F1-score	Precision	Recall	F1-score	
17	1.000	1.000	1.000	1.000	1.000	1.000	36
18	0.931	0.750	0.831	0.854	0.972	0.909	36
19	0.972	0.972	0.972	1.000	0.917	0.957	36
20	0.791	0.944	0.861	0.939	0.861	0.899	36
21	1.000	1.000	1.000	1.000	1.000	1.000	36
22	0.973	1.000	0.986	0.923	1.000	0.960	36
23	1.000	1.000	1.000	1.000	1.000	1.000	36
24	0.947	1.000	0.973	0.946	0.972	0.959	36
25	0.673	0.917	0.776	0.921	0.972	0.946	36
26	0.947	1.000	0.973	0.923	1.000	0.960	36
27	1.000	1.000	1.000	1.000	0.972	0.986	36
28	0.857	1.000	0.923	0.973	1.000	0.986	36
29	1.000	1.000	1.000	1.000	1.000	1.000	36
accuracy			0.956			0.969	1080
macro avg	0.961	0.956	0.956	0.972	0.969	0.969	1080
weighted avg	0.961	0.956	0.956	0.972	0.969	0.969	1080

Conclusion

During my internship at the European Space Agency, I was able to put to good use all the lessons I had learned during my academic training over the past few years. This experience allowed me to consolidate my theoretical and technical skills while enriching them thanks to the professional environment and the entire team I had the opportunity to work with.

The two projects I was assigned were particularly interesting and allowed me to explore and understand different methods, comparing not only the results obtained, but also their physical meaning. I also had the opportunity to work under a constraint that was new to me: the need for a compact model that was inexpensive in terms of resources. This prompted me to think differently and deepen my scientific approach. Working within a team that was so diverse (researchers in many fields, interns, etc.) and dynamic, I was able to learn a lot from others and develop my own scientific approach. I also had the opportunity to work on a project that was particularly interesting and challenging: the development of a new type of sensor that could be used in the field of biomedicine. This pushed me to think differently and deepen my scientific approach. Within a team composed of such diversity (researchers in many fields, interns, engineers, etc.) and with many nationalities, I was able to learn to share my opinions, listen, and understand cultural and scientific differences, both in method and interpretation.

In conclusion, by allowing me to join their team, ESA gave me the opportunity to participate in innovative projects, marking an important milestone in my academic career. It enabled me to strengthen my knowledge, broaden my perspectives in scientific research, and confirm my desire to pursue a career in this field. In conclusion, by allowing me to join their team, ESA gave me the opportunity to participate in innovative projects, marking an important milestone in my academic career. It enabled me to strengthen my knowledge, broaden my perspectives in scientific research, and confirm my desire to pursue a career in this field.

Bibliography

[1] Said Boulahia, Abdenour Amamra, and Mohamed Madi. *Early, intermediate and late fusion strategies for robust deep learning-based multimodal action recognition*. consulted in 2025. 2021. URL: https://www.researchgate.net/publication/354984828_Early_intermediate_and_late_fusion_strategies_for_robust_deep_learning-based_multimodal_action_recognition.

[2] Yujia Dai, Ziyuan Liu, and Shangyong Zhao. *Fusion of Laser-Induced Breakdown Spectroscopy and Raman Spectroscopy for Mineral Identification Based on Machine Learning*. consulted in 2025. 2024. URL: https://www.mdpi.com/1420-3049/29/14/3317.

[3] Igor Drozdovskyi and Samuel Payler. *IDENTIFYING PLANETARY MATERIALS BY COMBINING A CUSTOM MINERALOGICAL DATABASE WITH MACHINE LEARNING BASED MULTISPECTRAL CLASSIFICATION*. consulted in 2025. 2022. URL: https://www.hou.usra.edu/meetings/lpsc2022/pdf/1585.pdf.

[4] Konrad Gadzicki, Razieh Khamsehashari, and Christophe Zetzsche. *Early vs Late Fusion in Multimodal Convolutional Neural Networks*. consulted in 2025. 2025. URL: https://ieeexplore.ieee.org/document/9190246.

[5] Guang-Bin Huang, Hongming Zhou, Xiaojian Ding, and Rui Zhang. *Extreme Learning Machine for Regression and Multiclass Classification*. consulted in 2025. 2012. URL: https://ieeexplore.ieee.org/document/6035797.

[6] Hamid Joze, Amirreza Shaban, Michael Iuzzolino, and Kazuhito Koishida. *MMTM: Multimodal Transfer Module for CNN Fusion*. Computer Vision Foundation, consulted in 2025. 2020. URL: https://openaccess.thecvf.com/content_CVPR_2020/papers/Joze_MMTM_Multimodal_Transfer_Module_for_CNN_Fusion_CVPR_2020_paper.pdf.

[7] Mengcan Min et al. *A Novel Kernel-based Extreme Learning Machine with Incremental Hidden Layer Nodes*. consulted in 2025. 2020. URL: https://ifatwww.et.uni-magdeburg.de/ifac2020/media/pdfs/0261.pdf.

[8] Andone Moral and al. *Novel Performances of a Combined Raman-LIBS Instrument for Future Lunar Astronaut Exploration Program: The PHOENIX for PANGAEA Project*. consulted in 2025. 2025. URL: https://analyticalsciencejournals.onlinelibrary.wiley.com/doi/10.1002/jrs.6840.

1 Model Information

Input shape: (None, 450, 1)
Output shape: (None, 21)

Parameters summary:
Total params: 162485 (634.71 KB)
Trainable params: 162485 (634.71 KB)
Non-trainable params: 0 (0.00 Byte)

2 Filtering

- Merging classes

Merged class	New class
augite	diopside
aegirine	diopside
oligoclase	anorthite
albite	anorthite
fayalite	forsterite
kaersutite	magnesiohornblende
pyrite	pyrrhotite

- NaN filtering
  - 50.0%
- Minimum samples per class
  - n: 10
  - strategy: ge

3 Preprocessing

- Interpolation
  - {'lower_bound': 950, 'upper_bound': 1650, 'num': 450}
  - fill_value: nan
  - method: linear
- Normalization
  - strategy: local
- Imputation
  - strategy: constant
  - fill_value: 0.0

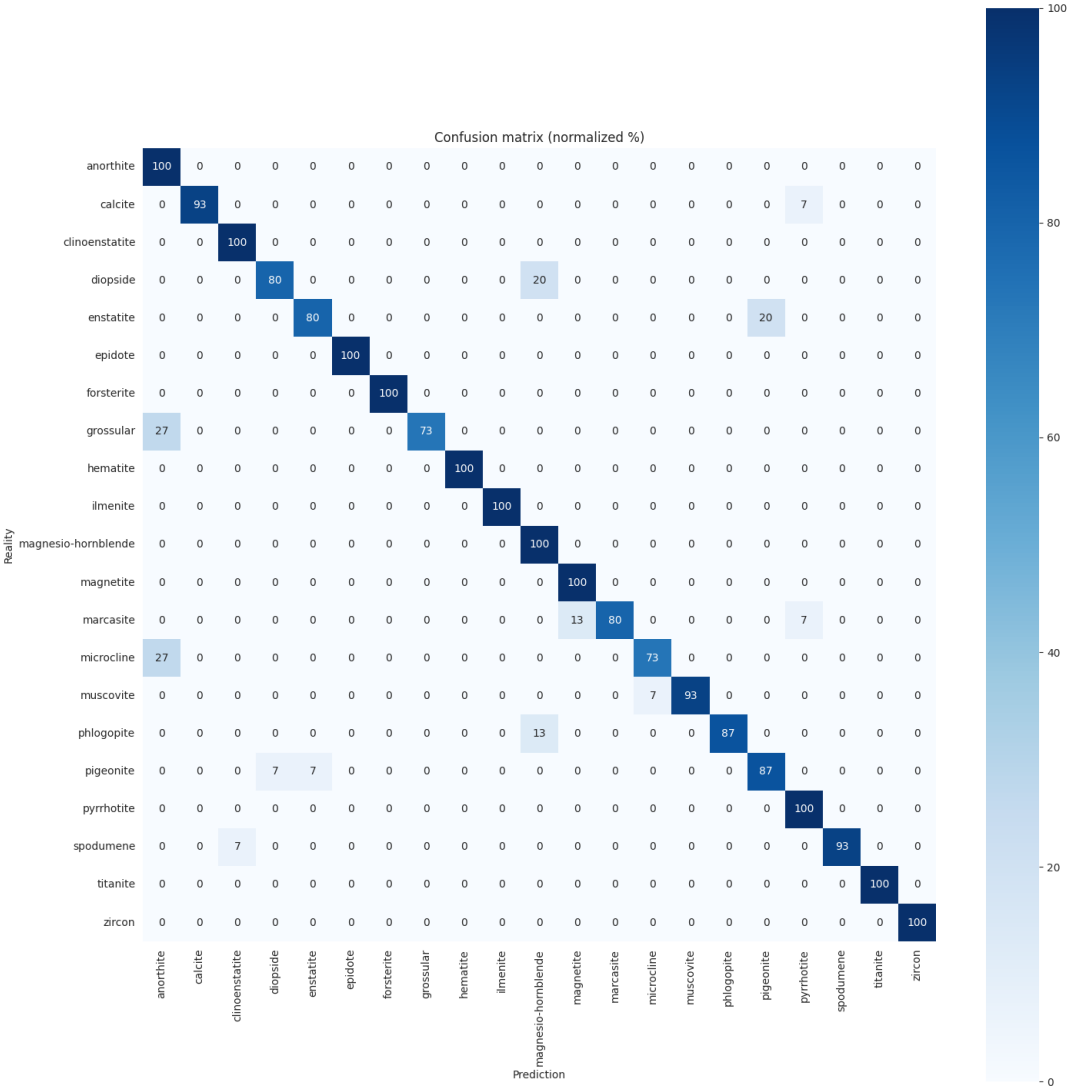
4 Training Metrics

Metric	Value
loss	0.13
accuracy	97.61%
val_loss	0.41
val_accuracy	95.24%

5 Considered minerals

anorthite	calcite
clinoenstatite	diopside
enstatite	epidote
forsterite	grossular
hematite	ilmenite
magnesio-hornblende	magnetite
marcasite	microcline
muscovite	phlogopite
pigeonite	pyrrhotite
spodumene	titanite
zircon	

6 Cross-validation results



7 F-score (mean ± std)

anorthite: 0.79 ± 0.06 — Plagioclase, Rare
calcite: 0.96 ± 0.09 — Calcite, Major
clinoenstatite: 0.97 ± 0.06 — Clinopyroxene, Rare
diopside: 0.85 ± 0.09 — Clinopyroxene, Rare
enstatite: 0.83 ± 0.21 — Clinopyroxene, Rare
epidote: 1.00 ± 0.00 — Epidote, Major
forsterite: 1.00 ± 0.00 — Olivine, Rare
grossular: 0.84 ± 0.09 — Garnet, Major
hematite: 1.00 ± 0.00 — Hematite, Minor
ilmenite: 1.00 ± 0.00 — Ilmenite, Minor
magnesio-hornblende: 0.86 ± 0.09 — Unknown subgroup, Unknown occurrence
magnetite: 0.94 ± 0.08 — Spinel, Major
marcasite: 0.88 ± 0.11 — Marcasite, Rare
microcline: 0.81 ± 0.12 — K Feldspar, Major
muscovite: 0.96 ± 0.09 — Dioctahedral mica, Major
phlogopite: 0.92 ± 0.11 — Unknown subgroup, Unknown occurrence
pigeonite: 0.84 ± 0.10 — Clinopyroxene, Rare
pyrrhotite: 0.94 ± 0.08 — Pyrrhotite, Rare
spodumene: 0.96 ± 0.09 — Clinopyroxene, Rare
titanite: 1.00 ± 0.00 — Titanite, Minor
zircon: 1.00 ± 0.00 — Zircon, Minor