UNIVERSITAT DE BARCELONA

Final Degree Project

Biomedical Engineering Degree

Image processing software for seizure onset zone localization in refractory epilepsy

Barcelona, 14 de juny de 2021

Author: Laura Latorre Moreno

Director/s: Aida Niñerola Baizán

Tutors: Aida Niñerola and Raúl Tudela

Acknowledgements

En primer lloc, agrair al Departament de Medicina Nuclear de l'Hospital Clínic per oferir-me la possibilitat de realitzar el meu treball de fi de grau. En especial, a l'Aida Niñerola per comptar i confiar en mi per al desenvolupament d'aquest projecte, així com per brindar-me les eines, el suport i ajuda necessàries, i per ensenyar-me i guiar-me en una branca fascinant de l'enginyeria biomèdica, a la qual vull dirigir el meu futur. També vull agrair la incorporació del Raúl Tudela al projecte, per aportar els seus coneixements de programació en Python i per ajudar-me a solucionar els problemes als quals m'anava enfrontant.

Per altra banda, donar les gràcies al Dr. Xavier Setoain i al Dr. Andrés Perissinoti pel seu temps dedicat en avaluar els resultats del projecte i per la bona resposta davant la eina proposada.

A tots els meus amics que han compartit la il·lusió per al meu TFG encara que no acabessin d'entendre el que estava desenvolupant.

Finalment vull agrair a la meva família, als meus pares i a la meva germana. Gràcies per fer el camí més fàcil i creure sempre en mi i les meves capacitats.

Abstract

Epilepsy is one of the most common serious neurological disorders in the world and a 30-40% of the affected population is resistant to the pharmacological treatment (refractory epilepsy). A possible treatment for them is the surgical resection of the epileptogenic zone (EZ). The success of the surgical treatment is fundamentally determined by the accuracy of presurgical identification of the EZ based on a variety of diagnostic tests. Among them, PISCOM technique is a multimodal imaging processing algorithm, useful for this purpose, yet not incorporated into clinical routine.

This project aims to develop an ergonomic and user-friendly graphical interface that integrates the PISCOM algorithm to make the process become easy and accessible for clinicians. To create the graphical interface, different software environments were studied. The solution chosen was to develop an extension for 3D Slicer, an open-source software package used for medical and biomedical imaging research, and the processing method was therefore adapted to the new platform. The result was assessed with a clinic questionnaire filled out by two nuclear medicine physicians of Hospital Clínic de Barcelona after an introduction session of the developed extension.

The extension was considered to be a user-friendly tool for applying the PISCOM technique, that fulfilled their requirements, and with future potential. Some next steps to improve the user experience were suggested.

Taking everything into account, the extension could substitute other tools for supporting seizure onset zone localization that require more image acquisitions, reducing the radiation over the patient and the cost of the diagnosis.

CONTENT

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

2D	Two Dimensions
3D	Three Dimensions
CIBER-BBN	Centro de Investigación Biomédica en Red de Bioingeniería, Biomateriales y Nanomedicina
COVID-19	Coronavirus disease 2019
DRE	Drug-resistant Epilepsy
EEG	Electroencephalogram
EF	Epileptogenic Focus
EZ	Epileptogenic Zone
FDG	Fluorodeoxyglucose
fMRI	Functional Magnetic Resonance Imaging
FWHM	Full Width at Half Maximum
GIMIAS	Graphical Interface for Medical Image Analysis and Simulation
GUI	Graphical User Interface
ILAE	International League Against Epilepsy
MRI	Magnetic Resonance Imaging
PET	Positron Emission Tomography
PISCOM	PET Interictal Subtracted ictal SPECT Coregistered with MRI
PSF	Point Spread Function
rFDG PET	Reprocessed FDG PET image/ Degraded PET
SOZ	Seizure Onset Zone
SISCOM	Subtraction of Ictal SPECT Co-registered to MRI
SPECT	Single Photon Emission Tomography
SPM	Statistical Parametrical Mapping
UB	Universitat de Barcelona
UI	User Interface
v-EEG	Video-electroencephalogram
WHO	World Health Organization

1. INTRODUCTION

Epilepsy is a neurological disorder characterized by recurrent episodes of paroxysmal brain dysfunction due to a sudden, disorderly, and excessive neuronal discharge [1]. These episodes are commonly called seizures, which cause involuntary movement that may involve a part of the body or the entire body and are sometimes accompanied by loss of consciousness and control of bowel or bladder function (WHO, 2019). It is one of the most common serious neurological disorders; its prevalence ranges from 0.5% to 1% of the population in developed countries, and even higher in developing ones [2].

With an appropriate pharmacological treatment, up to 60~70% of people living with epilepsy could become seizure free [3]. Unfortunately, there is a remaining fraction of affected population that still suffers from seizures despite being under the adequate trials of two tolerated and appropriately chosen and used anti-epileptic drugs (whether as monotherapies or in combination) (ILAE); epileptic seizures control is not achieved with medication [4]. This fraction constitutes the group of patients with *drug-resistant epilepsy*. In these patients, seizures have a focal or partial onset in a limited and specific area of the cerebral cortex.

Currently, the treatment for drug-resistant epilepsy is the surgical resection of the epileptogenic zone (EZ), defined as the "minimum amount of cortex that must be resected to produce seizure freedom" [5]. The success of the surgical treatment is fundamentally determined by the accuracy of presurgical identification and the prediction of the possible sequelae of the intervention. Presurgical identification of the EZ may include several diagnostic tests, defined during the presurgical evaluation: clinical semiology, video-electroencephalogram (v-EEG), neuropsychological tests, psychiatric evaluation, and neuroimaging exploration. When seizures are originated in deep brain structures or when they are rapidly propagated, v-EEG with surface electrodes does not achieve to identify with enough precision the EZ. Electroencephalogram monitoring may be necessary in these cases, requiring surgically placing deep or subdural electrodes. To avoid this invasive procedure, neuroimaging techniques come into play. Structural neuroimaging with magnetic resonance (MRI) has considerably reduced the need of EEG monitoring due to its ability to detect lesions that behave like EZ. However, not all epileptic lesions are visible, some of them can be very subtle, being unnoticed with conventional MRI sequences weighted in T1 or T2 [3]. When multiple foci are suspected, MRI shows no lesions, cortical dysplasias are poorly defined, or when the MRI lesion does not coincide with location of the v-EEG, the EZ cannot be accurately identified and the percentage of failure of the surgery increases. Then, functional neuroimaging scans, such as positron emission tomography (PET) and single-photon emission computed tomography (SPECT), can provide additional essential information to locate the EZ. Both explorations require the previous injection of a radiotracer; and depending on this radiotracer, brain perfusion images (SPECT) or neuroimages of glucose metabolism (PET) can be obtained.

Several imaging processing techniques have been described to improve sensitivity and specificity in the presurgical identification of the seizure onset zone (SOZ) with higher rates of favourable postoperative outcomes. The subtraction of interictal SPECT from the ictal SPECT coregistered with MRI (SISCOM), first described by O'Brien et al. [6], is an example that it is currently used in clinical routine, since several software have been developed to apply it in a simple and precise way. Another example is a voxel-by-voxel statistical analysis that can compare a patient's PET study with a control database in order to locate the hypometabolic zone related to the epileptogenic region [3]. Other image processing techniques exist, although some of them are still proof-of-concept studies. The subtraction of interictal PET from the ictal SPECT coregistered with MRI (PISCOM) can reduce the number of images required for the diagnosis, and so the cost and time, as it substitutes the interictal SPECT for the interictal PET (which usually is taken before SISCOM is required) and is equally valid for the purpose [7]. Additionally, localising the epileptogenic focus following PISCOM methodology can provide important benefits for those patients with refractory epilepsy in which all the previous mentioned functional neuroimaging scans (ictal SPECT, interictal SPECT and PET) are under consideration for an appropriate diagnosis; they can be exempted from one of the neuroimaging acquisitions, thus reducing the exposure to radiation and improving their comfort, especially in those requiring sedation during image acquisition process such as paediatric patients. However, no software tool has been developed yet so that it can be easily applied in clinical routine.

PISCOM was presented in August 2018 [7]. It integrates several preprocessing image techniques for the PET image so it can assimilate features of the SPECT image and both ictal and interictal images can be compared and subtracted. Those required preprocesses are: resampling to obtain the same matrix and voxel size, filtering to achieve similar smoothing, and intensity normalization to correct for differences in the total number of photons detected. Once the preprocessed PET (rFDG PET) is obtained, the same subtraction methods of SISCOM (already integrated in several software and ergonomic graphical interfaces) can be applied, replacing the interictal SPECT image for the interictal rFDG PET. The main limitation that makes PISCOM to remain as a proof-of-concept study is that all these processes are not integrated in an ergonomic graphical interface, and clinicians need it in order to find it easy to use in a daily basis. The goal here is to tackle with this limitation and facilitate bringing PISCOM into clinical practice.

1.1. Objectives

The **main** objective is to develop and assess an ergonomic and user-friendly graphical interface that integrates PISCOM technique in a unique work environment.

This graphical interface is for the *Comité d'Epilèpsia de l'Hospital Clínic* de Barcelona; clinicians and experts in Nuclear Medicine from this multidisciplinary committee will use it to locate the EZ following PISCOM methodology in those previously mentioned patients that might benefit from it. Clinicians require comfort and easiness when using any software tool; if they do not find these features, the final result will be obsolete.

Thus, a **second** objective is to adapt the image processing algorithm to the software environment chosen for the development of the graphical interface.

As a **last** goal, to acquire the required knowledge and formation in image processing techniques and front-end programming to carry out all the objectives stated.

1.2. Methods and structure of the project

Given the current situation of COVID-19 pandemic, this project has been developed mostly in a remote-from-home way. Ideally, it would have been developed in the Laboratori d'Imatges Biomèdiques (Facultat de Medicina de l'Universitat de Barcelona) in collaboration with the Nuclear Medicine department of the Hospital Clinic of Barcelona. However, since it consists of open-source software development and the only material required was a computer, it has been possible to perform all the steps in an online way. Every week, an online meeting (through online communication tools such as Skype or Google Meets) was set to make an update on the progress of the project. When necessary, some of these meetings have been physical. The timeline of the project has been from May 2020 to June 2021, when the final result has to be presented in court.

The body of this project memory is divided in the following parts: background, market analysis, conception engineering, detailed engineering, execution plan, technical feasibility, economic feasibility, regulation and legal aspects, and conclusions and future lines. The aim with this structure is to give a final and clear idea of all the aspects to take into consideration for executing this project.

1.3. Scope

The general scope is to design and programme a friendly-user and ergonomic graphical interface that integrates PISCOM technique so that clinicians from *Comité d'Epilèpsia de l'Hospital Clínic* can use it when required.

The scope breakdown is the following:

- A previous study and deep analysis of the existing software for neuroimaging processing, emphasizing in the front-end code.
- Familiarization with the several scripts that contain the PISCOM technique.
- Design of the graphical interface, taking into consideration the requests of the final users of the product (clinicians).
- Development of the graphical interface, integrating PISCOM necessary processes in it.
- Commissioning: Tests and final corrections. Presentation of the graphical interface.

It is significant to contemplate what is not included in the scope. Since PISCOM technique is already well defined and has been shown to be valid, modifying it is not in the scope. Additionally, the graphical interface is intended to be used, initially, only by Comité d'Epilèpsia de l'Hospital Clínic. For the moment, it falls beyond the scope to take it to other hospitals.

2. BACKGROUND

2.1. State of the art

The different imaging techniques involved in the identification of the EZ in drug-resistant epilepsy are SPECT, PET and MRI.
SPECT and PET are under the basis of nuclear medicine: to use certain properties of isotopes and the energetic particles emitted from the radioactive material previously injected to the patient. This means that radiotracers are taken internally, for example, intravenously or orally, and then external detectors capture and form images from the radiation emitted by them. Both techniques give functional information of the brain activity.
MRI uses another basis. Powerful magnets are used to polarize and excite hydrogen nuclei of water molecules in human tissue, producing a detectable signal which is spatially encoded, resulting in images of the body. With this technique, structural information of the brain is obtained.

On the other hand, to process the images and develop a graphical interface, platforms to write and execute a code are required. There exist several environments: from Jupyter Notebook or Spyder (Python language), to GIMIAS (developed in C++) which provides a workflow-oriented environment for biomedical image computing and simulation, and can be extended through the development of task-specific plug-ins.

Finally, several imaging processing techniques have been described to improve sensitivity and specificity in the presurgical identification of the SOZ with higher rates of favourable postoperative outcomes. Two of them are SISCOM and PISCOM.

2.1.1. Imaging techniques

SPECT

Single photon emission computed tomography imaging instruments provide three-dimensional (tomographic) images of the distribution of radioactive tracer molecules (gamma radiation emitters) that have been introduced into the patient's body (normally through injection into the bloodstream) [8]. The 3D images are reconstructed from a large number of projection images (2D) of body (or just the part of interest) recorded at different angles, which are obtained with gamma camera detectors that can detect the gamma ray emissions from the tracers. These cameras are mounted on a rotating gantry that moves them in a tight circle around the patient, who is lying motionless on a pallet. Figure 1 schematically illustrates the SPECT acquisition and tomographic reconstruction.

The most common uses of SPECT technique are related to neurological disorders, cardiac pathologies, and oncological diseases. Usually, a marker radioisotope is attached to a specific ligand to create a radioligand, whose properties bind it to certain types of tissues. Epileptic seizures are related to changes in regional cerebral blood flow [9]. During an epileptic seizure, both regional cerebral blood flow and metabolism are increased in the epileptic region. Therefore, tracers with the ability to cross the blood-brain barrier and irreversibly bind within the brain cell in quantities proportional to intracerebral blood flow are widely used. ^{99m}Tc−hexamethylpropyleneamine−oxime (^{99m}Tc-HMPAO) and ^{99m}Tc−ethyl cysteinate dime (^{99m}Tc-ECD) are small lipophilic agents that fulfil these properties [3].

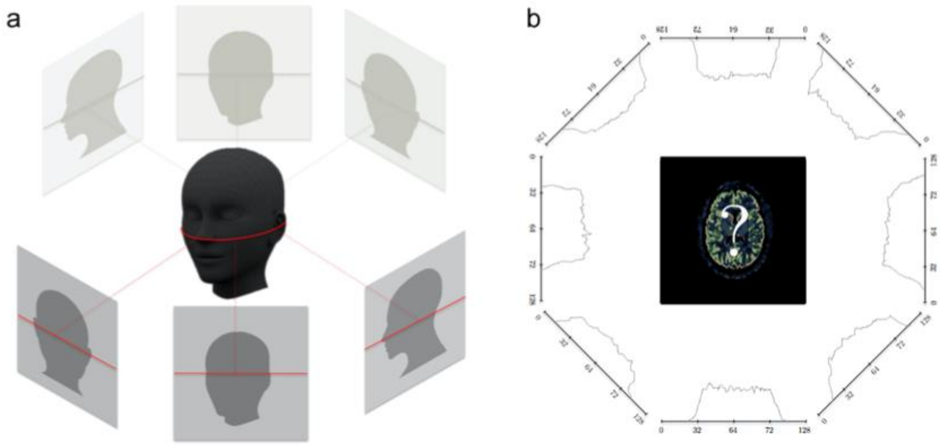


Figure 1: SPECT acquisition and tomographic reconstruction: a) acquisition of projections, red solid lines indicate a section in the brain and its corresponding ideal 1D projections and b) using the intensity distribution of the 1D acquired projections we can reconstruct the axial slice sought. From B. Martí, 2013 [10]

PET

Positron emission tomography is similar to SPECT in its use of radioactive tracer material and detection of gamma rays to create three-dimensional images. In contrast with SPECT, the decay of the radiotracers used with PET scans produce positrons, particles with the same mass as an electron but oppositely charged. Positrons react with electrons up to a few millimetres away (process also called annihilation) causing two gamma photons to be emitted in opposite directions. A PET scanner detects these emissions “coincident” in time, which provides more radiation event localization information and, thus, higher spatial resolution images than SPECT [11]. When two photons are detected at 180º and differences in time detection are very small (4-8 ns), the event is considered as a true coincidence and the location of the positron where the annihilation has taken place along the line is defined by the detection points (usually referred to as line of response or LOR). To estimate the LOR more accurately Time-of-flight (TOF) PET scanners use the time difference between the arrivals of both photons.

The projections obtained during acquisition (raw PET data) can be reconstructed into cross-section images using almost the same algorithms as in SPECT. Nevertheless, the amount of data to be managed is considerably higher than in SPECT, thus, significant computational resources are needed.

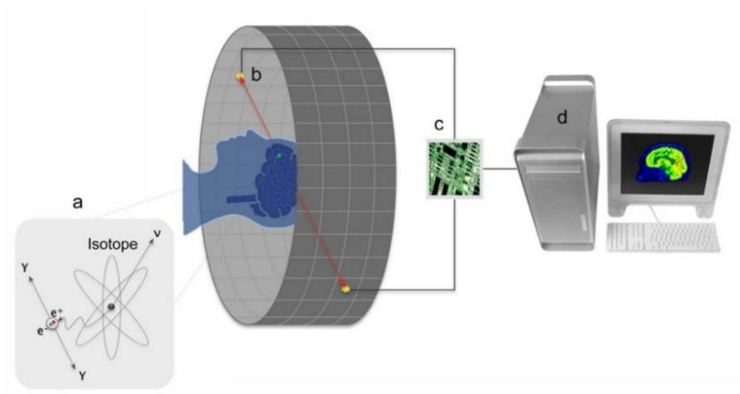


Figure 2: Detection process in PET imaging: a) Positron annihilation resulting in back- to-back gamma photons, b) detection of emitted pair of gamma photons, c) coincidence processing unit and d) computer for image processing and display. From B.Martí, 2013 [10]

The most common clinical applications are related to oncology, cardiology, neurology and psychiatry. In epilepsy, the most widely used radiotracer is ^{18}F fluorodeoxyglucose ([^{18}F]-FDG). It is a glucose analogue and an indirect marker of neuronal metabolism. Like the radiotracers used in SPECT for epilepsy, [^{18}F]-FDG has the ability to cross the blood-brain barrier. In the cell compartment, it is phosphorylated proportionally to the energy demanded by the brain cell [10]. Interictal FDG PET shows the dysfunctional brain areas related to the EZ as hypometabolic glucose-consuming regions. FDG PET has been demonstrated to be more sensitive and accurate than interictal SPECT, usually depicting a larger dysfunctional metabolic brain area than perfusion SPECT [7].

MRI

Magnetic Resonance Imaging (MRI) is a non-invasive and non-ionising imaging technology that produces three dimensional detailed anatomical images. It currently offers the most sensitive way of imaging soft tissue in the brain, since it has the capacity to distinguish different soft tissues of similar density (white matter and grey matter). It is based on an analysis of the relaxation properties of excited nuclei of some atoms (usually hydrogen), which differ depending on the surrounding tissue.

Hydrogen nuclei consist of a single proton that carries a positive electrical charge and is constantly spinning (rotating) around its axis. A moving electrical charge is called a current and following Faraday's law, an oscillating electrical current generates a magnetic field. Thus, protons have their own magnetic fields and behave like little bar magnets. The magnetic field for each proton is known as a magnetic moment. Magnetic moments are normally randomly orientated [12].
By employing powerful magnets to produce a strong magnetic field, protons in the body are forced to align with that field. Then, the group of protons start precessing with a speed proportional to the magnetic field (precession or Larmor frequency). When a radiofrequency current (with the same frequency as the Larmor frequency of the protons) is then pulsed through the patient, the protons are stimulated and spin out of equilibrium, straining against the pull of the magnetic field. Once the radiofrequency field is turned off, the MRI sensors can detect the energy released as the protons

realign with the magnetic field. The time the protons take to realign (relaxation time), as well as the amount of energy released, change depending on the environment and the chemical nature of the molecules [13].
Slice localisation is achieved by using gradient coils to generate a gradient field orientated along a chosen axis. This gradient field alters the strength of the main magnetic field in the chosen direction, so that protons within the gradient field have different Larmor frequencies.
In light of the above, the MRI scanner is composed mainly with main magnet coils, gradient coils, and an integral RF coil. Figure 3 shows the MRI scanner composition.

In refractory epilepsy, MRI detects the epileptogenic lesion in 80% of cases when is located in temporal lobe and 60% in frontal lobe [10]. In case the epileptogenic lesion is beyond the scope of the MRI technique, a structural MRI scan can be a useful anatomic reference for other functional imaging techniques (such as SPECT or PET).

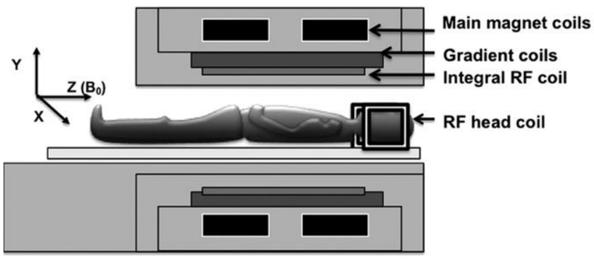


Figure 3: Components of an MRI scanner. From S. Currie, 2013 [12]

2.1.2. Software

Spyder

Spyder is an open-source cross-platform integrated development environment (IDE) for scientific programming in the Python language. It integrates with a number of prominent packages in the scientific Python stack as well as other open-source software [14]. Python is an interactive programming language, and it has a diverse range of options for Graphical User Interface (GUI) Frameworks as well as imaging processing tools.

GIMIAS

GIMIAS (Graphical Interface for Medical Image Analysis and Simulation) is an open source framework distributed under a Berkeley Software Distribution (BDS) licence, developed in C++, based on robust open source libraries. It is designed to be an integrative tool for fast prototyping of medical applications. It provides a workflow-oriented environment for advanced biomedical image computing and simulations, and it can be extended through the development of problem-specific plug-ins [15].

The aim of GIMIAS is to combine tools from different areas of knowledge providing a framework for multi-disciplinary research and clinical study. It is particularly tailored to integrate tools from:

medical imaging, computational modelling, numerical methods and computer graphics to provide scientific developers and researchers with a software framework for building a variety of tools. It allows building medical prototypes for clinical evaluation.

3D Slicer

3D Slicer is a free open-source extensible software application for medical image computing and visualization. It relies on a variety of libraries: VTK, ITK, CTK, CMake, Qt and Python. There is no restriction on use, but Slicer is not approved for clinical use and intended for research. Permissions and compliance with applicable rules are the responsibility of the user. [16,17]

Slicer allows to quickly develop and evaluate new extensions and distribute them to clinical users. Therefore, developers can focus on developing new methods and do not need to spend time with redeveloping basic data import/export, visualization or interaction features.

An extension is a “package” bundling together one or more modules for delivering new functionality to the user. Slicer supports three types of modules: Command Line Interface (CLI), Loadable Modules and Scripted Modules. CLIs are standalone executables with a limited input/output arguments complexity (simple argument types, no user interactions...). They are typically implemented in C++ (using ITK). Recent Slicer versions allow running (almost) any Python script as a CLI module by providing an interface description .xml file. Loadable modules are C++ plugins that are built against Slicer. They define custom GUIs for their specific behaviour as they have full control over the application. Finally, scripted modules are written in Python (easier to develop) and have access to most of Slicer internals [16,17].

Slicer, when downloaded in Windows, includes Qt Designer, which is launched every time to edit the UI of a scripted module. Qt Designer is the Qt tool for designing and building GUIs with Qt Widgets. It allows to compose and customize the windows or dialogs in a what-you-see-is-what-you-get manner and test them using different styles and resolutions. Widgets and forms created with Qt Designer integrate seamlessly with programmed code, using Qt's signals and slots mechanism, so that you can easily assign behaviour to graphical elements [18].

2.1.3. Neuroimaging processing techniques for localizing the epileptogenic focus

SISCOM

SISCOM evaluates the difference in the level of cerebral perfusion or blood flow between a SPECT image obtained during the period of epileptic or ictal crisis, and a SPECT image obtained during a period between crisis (or interictal).

Ictal SPECT can indicate which is the seizure onset zone. The increase in neuronal activity triggered by the epileptic crisis leads to an increase in neuronal metabolism and a regional increase in blood flow, which manifests as focal hyperuptake in SPECT.

On the other hand, interictal SPECT shows an area of hypoperfusion or normal perfusion in the dysfunctional brain areas related to the EZ. However, hypoperfusion is not specific to the EZ, since other brain lesions can produce areas of hypoperfusion in brain SPECT. In addition, the sensitivity of interictal SPECT to locate the EZ is low, so it is usually normal on many occasions. Therefore, the main application of interictal SPECT is to have a baseline study to compare the changes that ictal SPECT presents to facilitate interpretation [3].

Visual assessment of ictal SPECT images and their comparison with interictal SPECT images visualizing homologous sections is a laborious and complex task. For this reason, in recent years, computer programs have been developed that help to carry out this task more precisely and easily. These are programs that registrate both studies, ictal and interictal SPECT, creating an image that shows the differences between both studies, superimposed on the MRI image of the same patient, previously co-registered with the SPECT studies. This widely used methodology, known as SISCOM, was described by O'Brien [6] in 1998.

The workflow of SISCOM Analysis consists of the following steps [10]:

1. Realignment of Ictal and Interictal SPECT studies
2. Subtraction of Ictal and Interictal SPECT studies
3. Generation of a brain mask from the MRI
4. Co-registration of SPECT and MRI studies
5. Fusion of epileptogenic focus (EF) Information with MRI

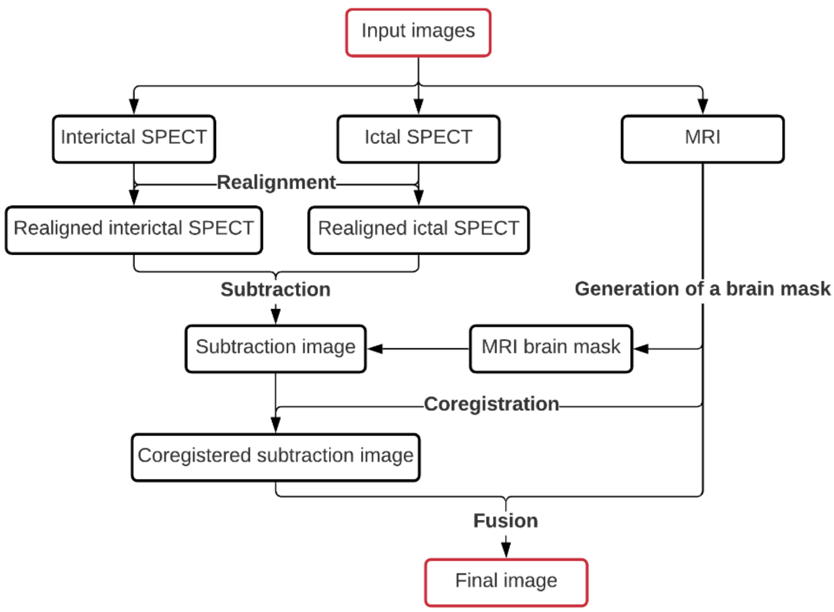


Figure 4: Workflow diagram of SISCOM technique

PISCOM

PISCOM technique arises from the implementation of the SISCOM technique, since it is a modified version of the SISCOM procedure that uses interictal PET instead of the interictal SPECT for seizure onset zone localization.

Interictal dysfunctional brain areas related to the EZ are also represented by ^{18}F-FDG PET as hypometabolic glucose-consuming regions that do not necessarily match the SOZ. This hypometabolism seems to be related with the loss of neuronal cells and the decrease in synaptic impulses due to the constant generation of anomalous electrical activity. Therefore, the hypometabolism of the PET is not only limited to locating the EF, but it is usually more extensive and exceeds the true EZ [3]. Nonetheless, FDG PET offers higher resolution and better signal-to-noise ratio than SPECT, providing valuable information that can modify surgical management and improve intracranial electrode placement. It has been demonstrated to be more sensitive and accurate than interictal SPECT, usually depicting a larger dysfunctional hypometabolic brain area than perfusion SPECT [7]. Thus, in 2018, Perssinotti et al. retrospectively checked if SISCOM technique could be carried out by substituting interictal SPECT for interictal PET in those cases in which the interictal FDG PET test was performed to add information on the localisation of the EF.

The development of the process is the same as that of the SISCOM technique: the subtraction of the interictal PET with the ictal SPECT is performed, followed by the co-registration with the MRI. However, the voxel-to-voxel subtraction between interictal PET and ictal SPECT is not trivial since both images have different characteristics and a preprocessing of FDG PET is necessary to assimilate features. This includes change in matrix and voxel size, smoothing, intensity normalization and resolution adjustment; obtaining a degraded PET image with worse features than the interictal PET image.

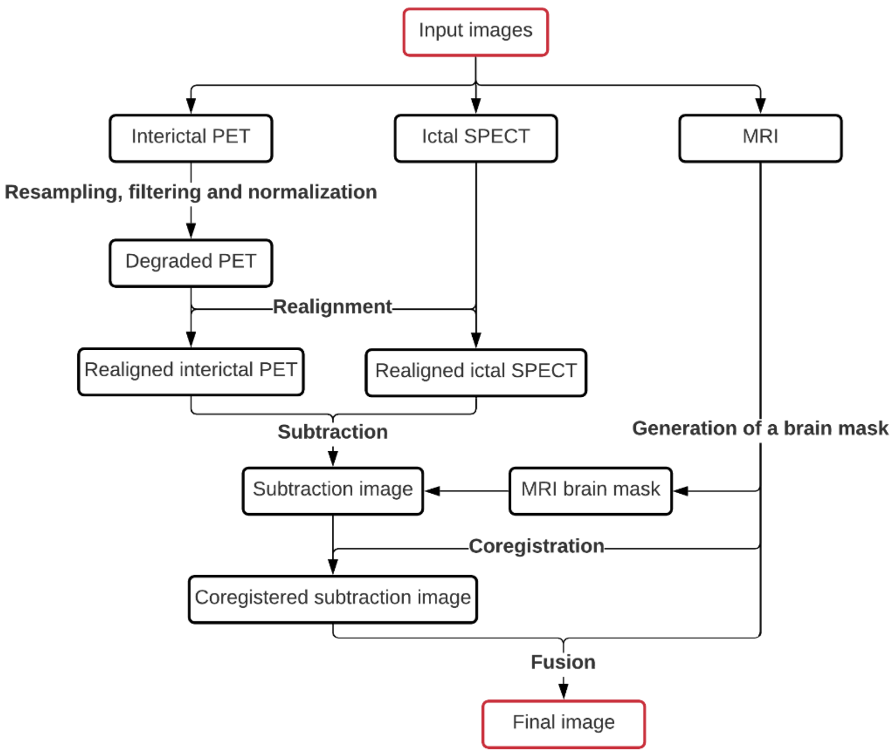


Figure 5: Workflow diagram of PISCOM technique

2.2. State of the situation

Currently, PISCOM methodology has not yet been incorporated into clinical routine. The process is cumbersome, and it requires advanced processing knowledge. Therefore, despite the previously stated significant benefits that it can provide, clinicians still do not follow PISCOM methodology to identify the epileptogenic focus. In addition, further clinical validation is needed apart from the one that was done with the published paper [7] to implement the technique beyond the world of research.

Some widely used examples of commercial and free software for image analysis are image viewers (AMIDE, MRIcro, OsiriX), image reconstruction platforms (STIR, NiftyReg, ASPIRE, QSPECT), image processing and analysis packages (SPM, SnPM) and large image processing tools (ANALYZE, BioImage Suite, 3D Slicer, GIMIAS, FSL) among others. Combining tools of these computer-aided methods, PISCOM analysis can be performed. However, these tools are all oriented to more general-purpose image processing and the whole process is still not comfortable for the clinician.

The fastest way to perform PISCOM analysis is by preprocessing the interictal PET image to make it similar to interictal SPECT, and then following the same steps as the SISCOM analysis but replacing the interictal SPECT image with the rFDG PET image. Some tools to perform specifically SISCOM analysis are: Analyze 12.0 (AnalyzeDirect), Segami SISCOM (Segami Corporation) or *FocusDET*. Analyze 12.0 and Segami tools, however, still need heavy user intervention and high technical skills. On the other hand, *FocusDET* [19] allows to perform the SISCOM analysis by means of a user-friendly interface. The application interface was built under the VPHTk (Virtual Physiological Human Toolkit) project from CIBER-BBN in 2012. Nowadays, it is a fully automated processing software accessible from any browser, with no installation, since it recently was transferred to Qubiotech Health Intelligence S.L. under the name of NEUROCLOUD-SISCOM [20].

In addition, it is also interesting to analyse other software and neuroimaging processing methods that have been developed to locate epileptogenic focus.
The medical imaging software sector is clearly divided into two types of companies, the producers of the scanners (OEM), who make use of their own software, and standalone software developers (software which runs independently of the medical imaging machines). Some competing software packages offer the possibility to analyse SPECT or PET images, such as PNeuro (PMOD TECHNOLOGIES Ltd), NeuroQ (Syntermed) or Hermes Brass (Hermes Medical). *NeuroGam* [21] (GE Medical System, Segami Corp., Columbia, MD, USA) is a statistical software for automated diagnosis of brain perfusion SPECT images that offers a higher sensitivity than EEG or MRI.
On the other hand, in 2018, Long et al., made a comparison of three methods that corregister SPECT images with MRI in patients with epilepsy: SISCOM, STATISCOM and MIMneuro [22]. The study demonstrated that STATISCOM showed the best performance for seizure localization, which was closely followed by MIMneuro.

3. MARKET ANALYSIS

3.1. Sectors to which it is directed

The new graphical interface will be initially directed to the Epilepsy Unit in Hospital Clínic de Barcelona. Epilepsy Unit is integrated in the Institut Clínic de Neurociènies. It focuses its efforts on improving the quality of life of patients with epilepsy and evaluating surgical options in drug-resistant epilepsy. It is one of the centres recognised by the Ministerio de Sanidad as a National Reference Centre (CSUR) for Refractory Epilepsy and surgical treatment of Epilepsy. The Unit has a surgical program in which 100-120 patients with Refractory Epilepsy are studied every year and around 40 surgeries are performed. It is pioneer in the combined use of different imaging techniques to locate the seizure onset area [23].

If the assessment of the final product is positive, it could be presented nationally or internationally. At the national level, there are 8 more CSUR Refractory Epilepsy Units [24]:

- Hospital Universitario La Fe, Comunidad Valenciana
- Hospital Universitario La Princesa, Comunidad de Madrid
- Complejo Hospitalario Universitario de Santiago, Galicia
- Complejo Hospitalario Virgen de las Nieves, Andalucía
- Hospital San Joan de Déu, Catalunya
- Hospital Universitario La Paz, Comunidad de Madrid
- Hospital del Mar, Catalunya
- Hospital de Cruces, País Vasco

At the international level, there exist several organizations that connect professionals in epilepsy worldwide [25]:

- International Bureau for Epilepsy (IBE): it was established in 1961 as an organization of laypersons and professionals interested in the medical and non-medical aspects of epilepsy. Is an international organization for national epilepsy organisations (IBE chapters) that exists to provide support for a strong global network, encourage the development of new chapters in underserved areas of the world, and to encourage communication and collaboration among all members.
- International League Against Epilepsy (ILAE): it was founded in 1909, it is the world's preeminent association of physicians and other health professionals working towards a world where no person's life is limited by epilepsy. ILAE's mission is to ensure that health professionals, patients and their care providers, governments, and the public world-wide have the educational and research resources that are essential in understanding, diagnosing and treating persons with epilepsy.

- e-pilepsy: Pilot European Network (ERN). e-pilepsy established a consortium of 13 centres as associate partners, with a further 15 collaborating centres driving the project. The primary expected outcome of the project is to increase the number and proportion of European children and adults cured of their refractory epilepsy by improving delivery of optimal epilepsy surgery throughout Europe.

If the aim were to sell the product instead of presenting it as an open-source software, the following companies are pioneers with medical-imaging technology improving medical services in the industry [26]:

- Arterys
- VoxelCloud
- Butterfly Network
- Infervision
- Qubiotech Health Intelligence S.L.
- Owkin
- Zebra medical vision
- Curemetrix
- CellmatiQ
- MedyMatch

3.2. Historical evolution of the market and future market prospects

It is significant to study the potential market that will use this platform (users) as well as the market that will indirectly benefit from it (patients), since the evolution of the market is strongly dependent on the number of people affected with refractory epilepsy, which increases the demand on new and accurate diagnostic procedures. The new platform that integrates PISCOM will benefit all these patients diagnosticated with refractory epilepsy in which all the previous mentioned functional neuroimaging scans (ictal SPECT, interictal SPECT and FDG PET) are under consideration for an appropriate identification of the seizure onset zone.

In Spain (2018), the active prevalence of active epilepsy is 8/1.000 inhabitants, which is around 400.000 epilepsy patients. The annual incidence is 31-57/100.000, being that between 12.400 and 22.000 new cases every year. Around a 30% of these 400.000, suffer drug-resistant epilepsy (DRE). It is known that in these cases, over a 5-10% over the total DRE patients are candidate to surgical treatment [27].
Worldwide, epilepsy accounts for 1% of the world's diseases, affecting approximately 70 million people. In the past few decades, despite the continuous development of antiepileptic drugs, there are still more than 30% patients with epilepsy progressing to drug-resistant epilepsy [28]. Since not all studies follow the same definition of DRE, it is difficult to determine the prevalence or incidence

of DRE. Kalilani et al. [29], after analysing a wide range of studies, determined an incidence proportion ranged from 0.06 to 0.51, and a prevalence ranged from 0.11 to 0.58.
Epilepsy is a dynamic condition with a highly variable clinical course among and within patients. Repeated remissions and relapses have been reported in patients with DRE. It is estimated that more than 4% of DRE cases could enter remission. Kalilani et al. also analysed the risk factors associated with DRE and found the following ones: early age at epilepsy onset, symptomatic epilepsy and a history of mental retardation, psychiatric comorbidities, neurologic deficit or abnormal neurologic imaging tests.

It is true that within the last decades, the development of antiepileptic drugs has never stopped, and that possibly, as new drugs are released to the market, the incidence of DRE cases could be decreasing. However, at the moment, refractory epilepsy is still a reality that needs of constantly improved and accurate presurgical identification techniques in order to ensure and increase the success of the surgery.

3.3. Product environment

The field of neuroimaging is in constant flux because of rapid progress over the last few decades. Imaging methods considered experimental today might be standard for epilepsy evaluation in the future. Either hybrid methods using combinations of MRI with SPECT, PET, or magnetoencephalography (MEG), or portable imaging systems, machine learning algorithms, or imaging-based nanoparticle delivery systems to the ictal onset zone… all these techniques hold great promise for patients with epilepsy.

In patients with drug-resistant epilepsies, the focus is on obtaining high-quality images. Advances in neuroimaging techniques are most visible in three areas – structural MRI, fMRI, and EEG combined with fMRI (EEG/fMRI). These techniques help determine epilepsy surgery feasibility and the risks of negative cognitive outcomes after resection. Recent technical refinements have improved delineation of brain anatomy and function, both of which are important for surgery outcomes [30].

4. CONCEPTION DESIGN

4.1. Solutions study

There exist several pathways to develop the new graphical interface. Here, three of them will be presented. The first one consists of recompiling FocusDET to add a plug-in. The second option is to create a new graphical interface that incorporates just the preprocessing of the FDG PET image, with the aim to be complemented with the SISCOM technique integrated in the external server NEUROCLOUD-SISCOM, where FocusDET was transferred. The third and last solution estimated will be to create a completely new extension that integrates the whole PISCOM technique in a unique platform (3D Slicer).

4.1.1. Solution 1: Add a plug-in to FocusDET

FocusDET software and code is currently available in an old computer owned by the UB's Laboratori d'Imatges Biomèdiques. The idea is to get access to the code (C++), and add there the code that corresponds to the *PISCOM Analysis* in a new tab as a new plug-in. The SISCOM Analysis plug-in of FocusDET was developed using GIMIAS. A GIMIAS plug-in is composed of one or more modules, each one performing a specific task. The structure of each module is always the same and divided into: the user interface (`PanelWidget` class) and the processor or algorithm (`Processor` class).

The graphic user interface (GUI) designer used for developing FocusDET was wxGlade, which allows to create wxWid-gets user interfaces, *i.e.* buttons, lists, scrolls, text fields, etc. wxGlade generates GUI codes in Python, C++, Perl, Lisp and XRC using a visual editor to create mouse-operated forms, menus and toolbars [10]. So that the new plug-in is fully integrated in FocusDET, it will be designed using the same GUI, wxGlade.

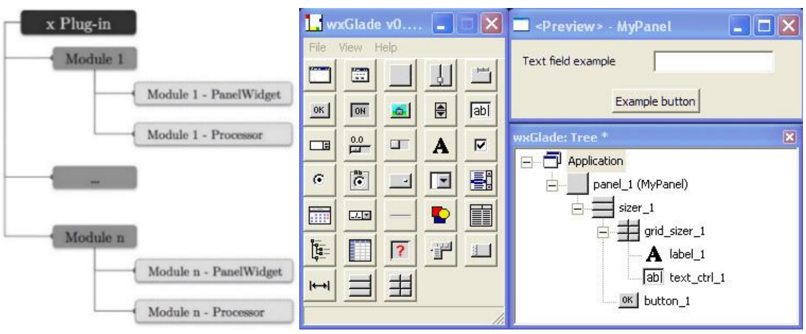


Figure 6: Example of a plug-in structure (left) and example of a graphical user interface (GUI) designed using wxGlade (right). Adapted from B. Martí, 2013 [10]

4.1.2. Solution 2: Develop an independent graphical interface to preprocess FDG PET

This is the simplest solution. The aim here is to just incorporate the first step of the PISCOM methodology to a new application, so that clinicians get the rFDG PET, ready to be used as a substitution of the interictal SPECT in NEUROCLOUD-SISCOM online platform. Since the application will be completely new and developed from scratch, there is more freedom in choosing the programming language.

In this case, Python would be used to develop both front-end and back-end. Python has a huge number of GUI frameworks, from TkInter or PythonQt to several other cross-platform solutions, as well as bindings to platform-specific (also known as “native”) technologies [31].

To obtain the rFDG PET image, the following algorithms will be integrated and applied to FDG PET image:

1. Resampling to obtain the same matrix and voxel size as those of interictal perfusion SPECT.
2. Filtering to achieve similar smoothing between SPECT and PET images.
3. Intensity normalization to correct for differences in the total number of photons detected.

4.1.3. Solution 3: New graphical interface that integrates the whole PISCOM methodology

The previous solution has the handicap that clinicians will have to use two different platforms for the whole process. To tackle with this limitation, this solution consists of creating from scratch a brand-new graphical interface that integrates all the algorithms required for the complete PISCOM analysis.

This solution is an extension of solution 2, thus the programming language chosen will also be Python. Apart from developing the 3 points previously described in solution 2, the following algorithms that refer to the subtraction of interictal rFDG PET to the ictal SPECT and the co-registration with MRI image, will be included:

4. Realignment of Ictal SPECT and Interictal PET Studies (S-P)
5. Subtraction of Ictal SPECT and Interictal PET Studies
6. Generation of an MRI Mask to Extract the Brain Region
7. Co-registration of SPECT, PET and MRI Studies
8. Fusion of EF information with MRI

Among all the different platforms studied in 2.1.2. Software, 3D Slicer is the most complete to develop the graphical interface for the following reasons:

- ✓ It has a tool to create and load scripted modules (in Python language).
- ✓ It launches Qt Designer to edit the UI.
- ✓ It already has a 3D image visualizer.
- ✓ It has a wide documentation for developers, including specific tutorials for new users aiming to contribute to Slicer repository.
- ✓ It has a large community and a discussion forum where problems and errors found in 3D Slicer during development are posted, and later solved by contributors and support staff.

However, Slicer has some limitations that must be considered:

- Slicer integrates 3.6.7 Python version, and only PyPi packages with pip installer can be installed from the Slicer Python terminal. Some python packages that must be built or downloaded from GitHub cannot be used in our final solution since Slicer do not recognize them.

4.2. Proposed solution

Although the easiest solution apparently is to add a plug-in for PISCOM to FocusDET, getting access to the code and compile it might be not possible, since last time this was tried, the compilation gave some errors. Moreover, it was developed in a different Windows operating system version (Windows 7) and trying to migrate it to Windows 10 might consume a lot of time. In addition, the global platform where it was developed (GIMIAS) has not been updated during the last years.

On the other hand, starting from scratch gives the advantage of freely selecting the programming platform and language. FocusDET solution entails writing in C++, and some time would be required to learn this programming language, since I do not have experience with it. However, with solutions 2 and 3, developing in Python is an option, and this learning time saved can be destinated to improve the performance of the algorithm and the UI.

Taking into consideration that integrating all the steps for PISCOM technique into a unique graphical interface once one masters the programming environment should be easy and fast, **solution #3** has been the chosen and proposed for this project.

4.3. Alternative solutions

In case important drawbacks found during the Slicer's extension development make solution 3 nonviable, solution 1 will be the one considered. Despite the previously mentioned limitations it could present, the whole final product would be very similar in terms of user interface to the one that clinicians currently have on their computers, meaning that their adaptation to it will be faster and easier; so, the second goal would be almost surely achieved. Solution 1 offers an initial template where to start working with; the GUI interface is already preestablished and the code from SISCOM Analysis plug-in can be useful for the development of the new PISCOM analysis plug-in.

As a last and simpler option, solution #2 will be selected. Other open-source platforms that use Python language in his whole integrity, thus allowing to use any package compatible with Windows, will be studied. This solution would probably require of additional open-source software to visualize the output images, which is less user-friendly than the others.

5. DETAILED ENGINEERING

The core structures of the PISCOM extension and its main module were created using the Extension Wizard module in 3D Slicer. This Extension Wizard module provides tools to create and manage extensions from within Slicer. The tutorial followed for the process was found in *PerkLab's Slicer bootcamp training materials* [32], under *Scripting and module development tutorial* section, which also describes both the Slicer and the scripted module architecture.

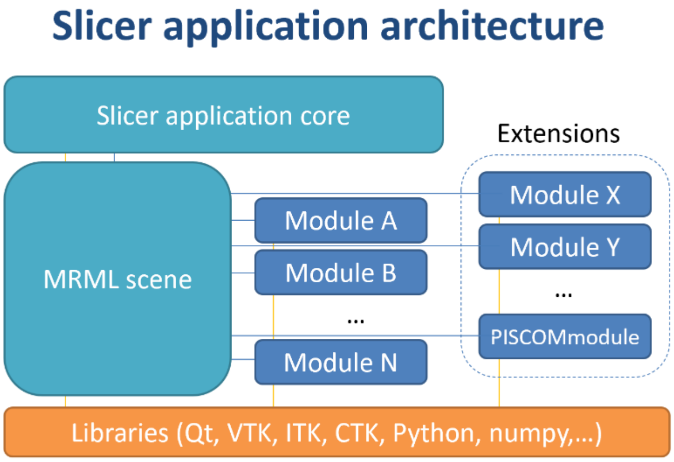


Figure 7: Slicer application architecture. Adapted from Laboratory for Percutaneous Surgery, 2019 [32].

A 3D Slicer extension has the following structure: an image that will be the icon of the extension, a CMakeLists.txt that describes the Extension Metadata and one or various folders, depending on the number of scripted modules. A scripted module folder contains a .py file that defines the logics of the module (the code that will define the actions and functions behind every element of the module UI), a CMakeLists.txt that describes de module metadata, a Testing folder, a _pycache_ folder and a Resources folder which contains two folders: Icon and UI, the former includes the icon of the module and the latter the .ui file that defines the Widgets of the module and it is launched with QtDesigner.

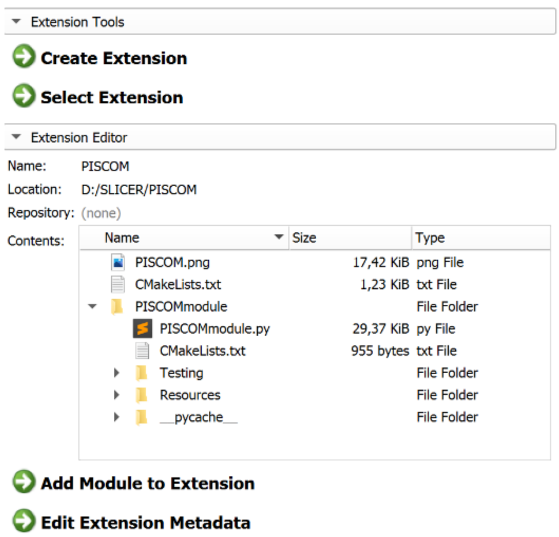


Figure 8: Extension Wizard module. Structure of the PISCOM extension and the PISCOMmodule

When an extension is created with the Extension Wizard module, the folder of the extension containing the CMakeLists.txt file and the .png image of the extension is automatically generated. Then, once the extension is generated, modules can be added to the extension. Slicer asks about the type of module being created. For the PISCOM module, *scripted* was the option chosen. Like in the creation of an extension, all the files previously mentioned that compose the module, are automatically generated. The python script generated is not empty, it contains the skeleton code of the module, which consist of four *classes* containing predefined functions. For the PISCOM module they were:

1. *class* PISCOMmodule(*ScriptedLoadableModule*): defines the __init__ function of the module which includes its metadata: title, categories, dependencies, contributors, the help text and the acknowledgement text.
2. *class* PISCOMmoduleWidget(*ScriptedLoadableModuleWidget*): defines the functions behind the UI of the module.
3. *class* PISCOMmoduleLogic(*ScriptedLoadableModuleLogic*): defines the functions of the PISCOM algorithm that are linked with the apply buttons of the UI in PISCOMmoduleWidget.
4. *class* PISCOMmoduleTest(*ScriptedLoadableModuleTest*): defines the functions for creating a test case of the algorithm.

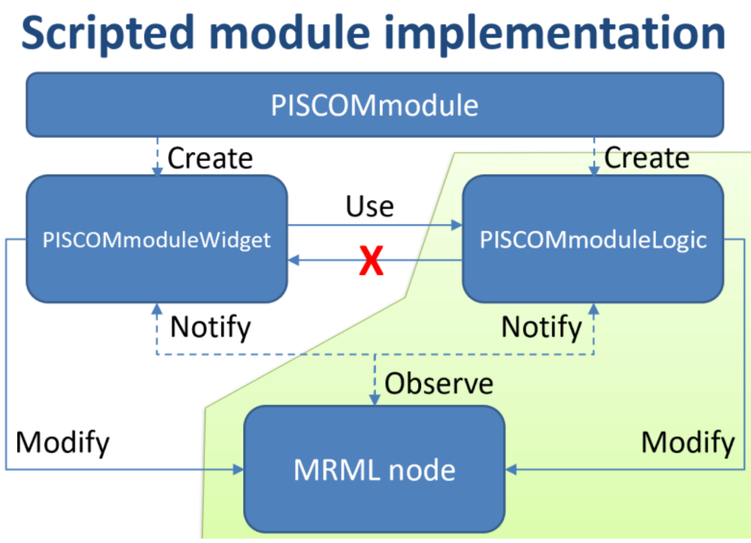


Figure 9: Scripted module implementation. Adapted from Laboratory for Percutaneous Surgery, 2019 [32].

Along with the .py file, a .ui file is also automatically generated with a simple structure of input and output nodes. Figure 10 shows the initial UI structure of a module launched with QtDesigner.

All the extension and module development were performed as a modification over three of the initial files: PISCOMmodule.py, PISCOMmodule.ui and the CMakeLists.txt (of the extension). For the PISCOMmodule.py modifications, the Sublime Text [33] editor was downloaded and defined as the default editor of the .py files in Windows.

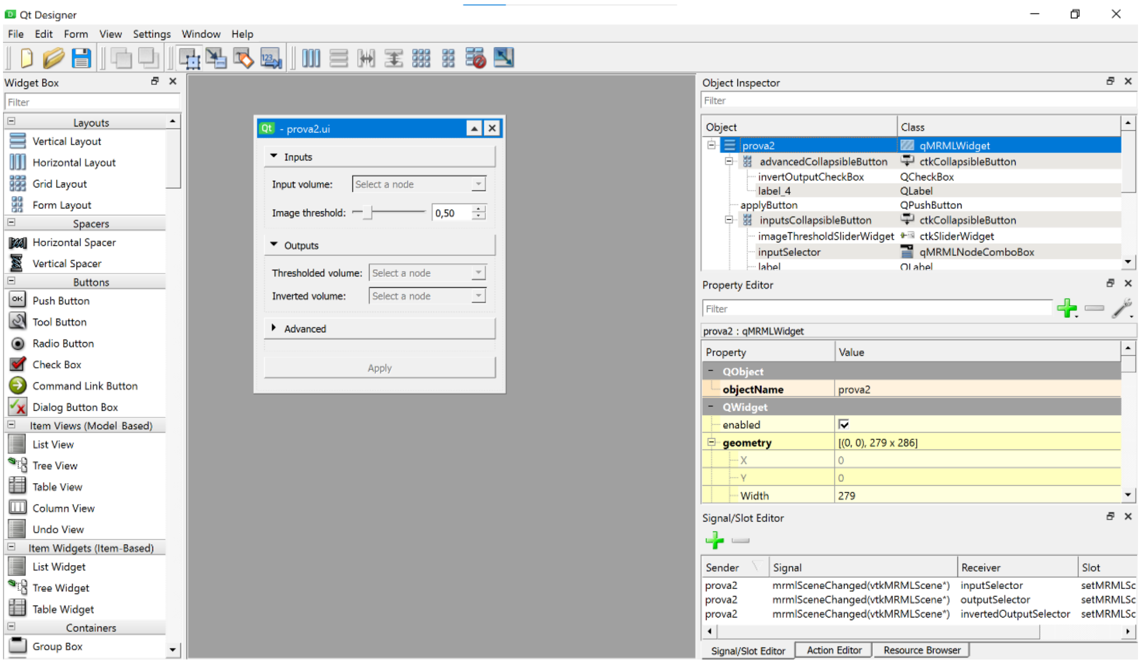


Figure 10: Automatically generated UI of a module

5.1. Detailed and dimensioning diagrams

The algorithm code integrated in PISCOM module was developed by the Biomedical Imaging Laboratory of the UB. During the integration of the algorithm into Slicer, slight modifications in its order were made. At the initial proof-of-concept study, since the PISCOM method was partially performed using FocusDET SISCOM analysis plug-in, the steps were adjusted to those performed by FocusDET software. However, considering that the order of some steps can be altered, keeping the same output image at the same time, and there is no dependence on other software, there is freedom to make this type of modifications.

The generation of an MRI mask to extract the brain region was defined as the first step since it is a required input for the co-registration, the preprocessing of the FDG-PET and the fusion of MRI and subtraction images. On the other hand, instead of aligning the PET and SPECT studies and at the end co-registering them with the MRI, the former step can be omitted if the co-registration with the MRI is performed at the beginning of the process. Thus, co-registration of SPECT, PET and MRI studies was defined as the second step.

Also, the PET preprocessing steps to obtain the degraded PET (rFDG PET), were changed. As mentioned, the FDG PET images resampling to obtain the same matrix and voxel size as those of interictal perfusion SPECT was no longer needed if co-registration and mask generation were the first steps of the method. Secondly, in the existing preprocessing code, intensity normalization is performed before than filtering, so the same structure was maintained in the Slicer's integration. This leads to the following steps:

1. Generation of an MRI mask to extract the brain region
2. Co-registration of SPECT, PET and MRI studies

3. PET preprocessing
    3.1. Intensity normalization to correct for differences in the total number of photons detected.
    3.2. Filtering to achieve similar smoothing between SPECT and PET images.
4. Subtraction
5. Fusion

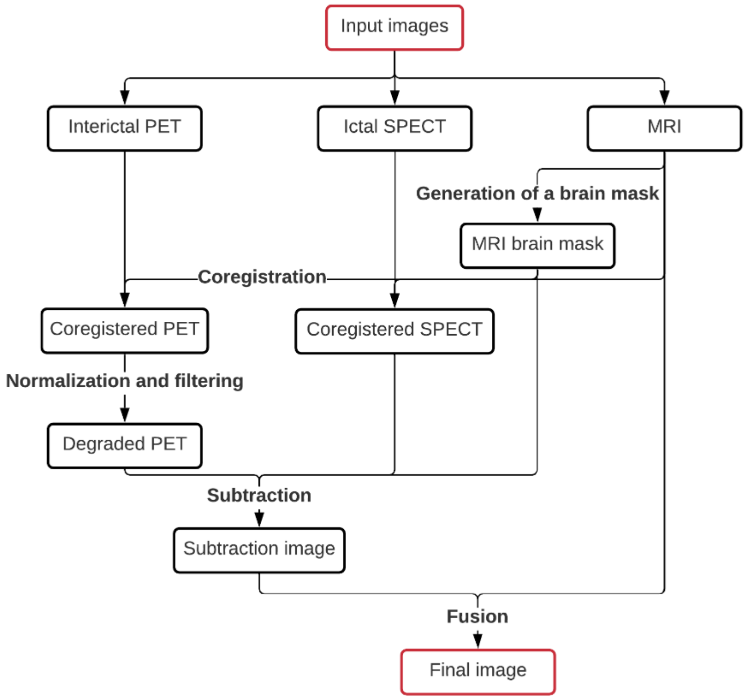


Figure 11: Workflow diagram of the PISCOM algorithm implemented in the 3D Slicer extension

The whole algorithm was divided into 5 blocks: MRI Brain Mask, SPECT/PET – MRI coregistration, Degrade PET, Subtraction and Fusion. Each block is represented by a ctkCollapsibleButton and contains its required input nodes and apply buttons:

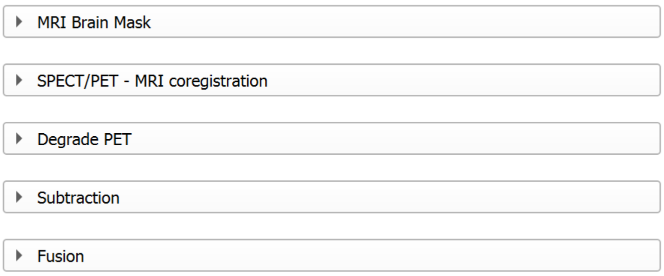


Figure 12: ctkCollapsibleButtons of the PISCOM module

The following chapters will explain the content of each block, and will be organised by 2 sections: first, the UI design will be explained, and then the functionality behind the nodes and buttons.

5.1.1. Generation of an MRI mask to extract the brain region

UI design

The UI design of this block consists of a ctkCollapsibleButton with a Layout in a Form Layout disposition containing: 2 QLabels, a qMRMLNodeComboBox, a QCheckBox and a QPushButton. The distribution of these objects is shown in Figure 13, and Figure 14 shows the object inspector (in QtDesigner) for this block.

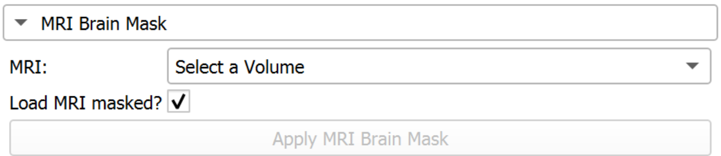


Figure 13: UI design of MRI Brain Mask block

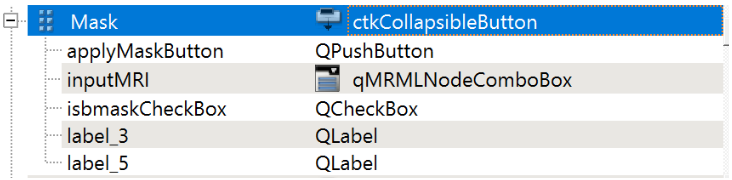


Figure 14: Objects and class of the MRI Mask of the Brain block

The qMRMLNodeComboBox is enabled by default, and the node type is set to vtkMRMLScalarVolumeNode. The editEnabled checkbox is checked so that the user can press the Edit Volume button and be redirected to the Slicer Volumes module (12. Annexes, Volumes) to see the volume information and edit the display features. This is general for all the volume nodes generated in the module.

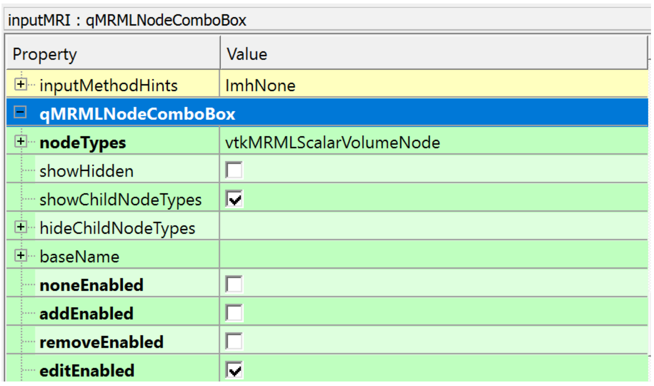
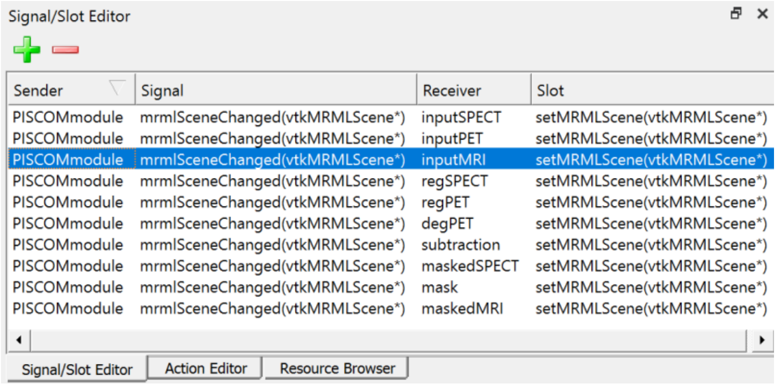


Figure 15: Property editor of inputMRI volume node

Also, all the volume nodes must have a signal/slot generated so that the nodes are connected with the PISCOMmodule.py, which sends a signal when the scene is being changed (e.g. a volume is being selected from de node) [34]. In the Signal/Slot Editor in QtDesigner, the PISCOMmodule is set as the sender of a mrmlSceneChanged(vtkMRMLScene*) signal, and the receiver, in this case inputMRI node, receives the setMRMLScene(vtkMRMLScene*) slot.


Signal/Slot Editor

Sender	Signal	Receiver	Slot
PISCOMmodule	mrmlSceneChanged(vtkMRMLScene*)	inputSPECT	setMRMLScene(vtkMRMLScene*)
PISCOMmodule	mrmlSceneChanged(vtkMRMLScene*)	inputPET	setMRMLScene(vtkMRMLScene*)
PISCOMmodule	mrmlSceneChanged(vtkMRMLScene*)	inputMRI	setMRMLScene(vtkMRMLScene*)
PISCOMmodule	mrmlSceneChanged(vtkMRMLScene*)	regSPECT	setMRMLScene(vtkMRMLScene*)
PISCOMmodule	mrmlSceneChanged(vtkMRMLScene*)	regPET	setMRMLScene(vtkMRMLScene*)
PISCOMmodule	mrmlSceneChanged(vtkMRMLScene*)	degPET	setMRMLScene(vtkMRMLScene*)
PISCOMmodule	mrmlSceneChanged(vtkMRMLScene*)	subtraction	setMRMLScene(vtkMRMLScene*)
PISCOMmodule	mrmlSceneChanged(vtkMRMLScene*)	maskedSPECT	setMRMLScene(vtkMRMLScene*)
PISCOMmodule	mrmlSceneChanged(vtkMRMLScene*)	mask	setMRMLScene(vtkMRMLScene*)
PISCOMmodule	mrmlSceneChanged(vtkMRMLScene*)	maskedMRI	setMRMLScene(vtkMRMLScene*)

Signal/Slot Editor | Action Editor | Resource Browser

Figure 16: Signal/Slot Editor

The QCheckBox checkable and checked properties are set to True. The Apply MRI Brain Mask QPushButton is not checkable when the module is initialized. In PISCOMmodule.py inside the PISCOMmoduleWidget *class*, and the updateGUIFromParameterNode function, it is defined that if the inputMRI is filled, the Apply button is then enabled.

Logics

To generate an MRI mask to extract the brain region, the following python functions, packages, and Slicer modules were studied:

- nilearn.masking.compute_brain_mask [35]
- median-otsu dipy [36]
- DeepBrainSeg [37]
- deepbrain [38]
- Skull Stripper – Slicer [39]
- Swiss Skull Stripper – Slicer [40]

Among all these possible solutions, deepbrain package outperformed extracting the brain region. Its function Extractor() runs a custom U-Net model trained on a variety of manual-verified skull-stripping datasets. However, its latest release is dated in September 2018, and uses a deprecated TensorFlow version, leading to an error if the first lines of its script are not modified. The solution to this problem was found in a github fork [41] and it is documented in the 12.2. Installation manual.

The Apply MRI Brain Mask QPushButton, when clicked, computes the onApplyMaskButton function found in the PISCOMmoduleWidget *class*, which calls the PISCOMmoduleLogic() *class* and tries to execute the maskButton() function. This function, takes the input node (the inputMRI node) and the isbmaskCheckBox boolean, runs the computeMask() function (which uses the Extractor() function) to obtain the brain mask, and saves it and loads it to Slicer. By default, it also applies the mask to the MRI image to obtain a masked MRI. However, the user can uncheck the isbmaskCheckBox and then the masked MRI is not returned as an output. Further detail of what the maskButton() function does and its dependencies can be found commented on the PISCOMmodule.py. Finally, once the maskButton() function returns the outputs, the currentNodeID of the mask and maskedMRI input volume nodes (found in the next blocks), are set to the corresponding obtained outputs, so that the process becomes more automated and the user won't need to manually select those outputs when required.

5.1.2. Co-registration of SPECT, PET and MRI Studies

UI design

The UI design of this block consists of a ctkCollapsibleButton with a Layout in a Form Layout disposition containing: 3 QLabels, 3 qMRMLNodeComboBox, and a QPushButton. The distribution of these objects is shown in Figure 17, and Figure 18 shows the object inspector (in QtDesigner) for this block.

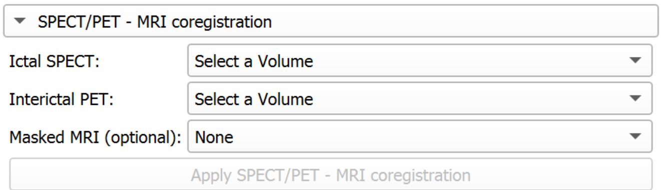


Figure 17: UI design of SPECT/PET – MRI coregistration block

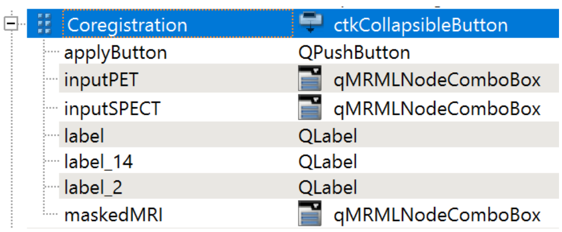


Figure 18: Objects and class of the SPECT/PET – MRI coregistration block

Unlike the previous Mask ctkCollapsibleButton, the *collapsed* property of the Coregistration ctkCollapsibleButton and the next ctkCollapsibleButtons of the module is set to True, so that when the module is initialized, the user recognises the MRI Mask of the Brain block as the first step and uncollapses the next collapsible buttons as the blocks are run.

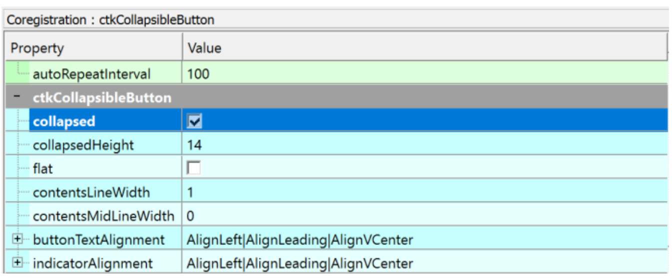


Figure 19: Property editor of Coregistration ctkCollapsibleButton

Differently to the rest of qMRMLNodeComboBox, the maskedMRI node has the noneEnabled property set to True (seen in Figure 13), since it is not necessary that this node is filled to compute the PISCOM algorithm. However, it has been found that the co-registration of the ictal SPECT image is better performed when the fixed volume is the masked MRI rather than the MRI volume, so the masked MRI volume is computed by default in the previous block and this node is automatically filled.

The Apply SPECT/PET – MRI coregistration QPushButton is not checkable when the module is initialized. In PISCOMmodule.py inside the PISCOMmoduleWidget *class*, and the updateGUIFromParameterNode function, it is defined that if the inputMRI, the inputSPECT and the inputPET nodes are filled, the Apply button is then enabled.

Logics

To co-registrate the SPECT and PET volumes to the MRI, the following python projects were studied [42]:

- ANTsPy: this library is useful for co-registering images in a simple way, but it is not implemented in Windows, and the solutions found to build it on Windows required the installation of additional packages (Visual Studio) which would make the installation process much more sophisticated.
- Dipy: it is a python library for the analysis of MR diffusion imaging. The results given by the registration function available in this library were poor.
- PyElastix: it is a thin wrapper around the SimpleElastix toolkit. Its drawback is that it not fully integrates all SimpleElastix features, so the registration option to return and read the output image format in '.nii' is not implemented: the only format allowed is '.mha'
- Nireg: its registration functions gave some errors when used.

Since none of the solutions were successful, the registration extensions inside Slicer were also studied [43], given that it is possible to import the functions from another module in the PISCOMmodule script:

- Slicer Elastix [44]: This extension makes the Elastix medical image registration toolkit [45–47] available in Slicer.
- BRAINSFit – Slicer [48]: This extension includes a module to register a three-dimensional volume to a reference volume (with Mattes Mutual Information by default). The method used is described in [49].

The SlicerElastix module *General Registration (Elastix)* outperformed registering the volumes; it integrates all the .txt parameters files of the *elastix* parameters repository, including the one for PET-MRI registration, which suits for both the interictal PET and ictal SPECT volumes, so it was the solution selected. The extension can be downloaded from the Slicer's Extension Manager, but since some modifications on its components were needed so that the output image is read in *.nii* format (<u>12.1. SlicerElastix modified extension</u>), it was downloaded from the Github repository [44], and the modified version needs to be imported to Slicer from the Extension Wizard module during the PISCOM extension installation process (<u>12.2. Installation manual</u>).

The Apply SPECT/PET – MRI coregistration QPushbutton, when clicked, computes the onApplyButton function found in the PISCOMmoduleWidget *class*, which calls the PISCOMmoduleLogic() *class* and tries to execute the Register() function two times (one for each volume to co-register with the MRI). This function takes as input parameters: 3 input nodes (the inputMRI, the moving volume -SPECT or PET- and the maskedMRI), a list of parameter files names (["Parameters_Rigid.txt", "Par0002.fs.MI.rigid.RandomCoordinate.txt"]) and the desired name of the output file. It applies the registerVolumes() function from the ElastixLogic() to obtain the co-registered volume and save it and load it to Slicer. If the maskedMRI node is None, the registerVolumes() function uses the inputMRI node as the fixed volume, else, the fixed volume used is the maskedMRI node. Further detail of what the Register() function does can be found commented on the PISCOMmodule.py**¡Error! No se encuentra el origen de la referencia.**. Finally, once the Register() function returns the outputs, the currentNodeID of the regSPECT and regPET input volume nodes (found in the next block), are set to the corresponding obtained outputs.

5.1.3. PET preprocessing: intensity normalization and filtering

UI design

The UI design of this block consists of a ctkCollapsibleButton with a Layout in a Form Layout disposition containing: 3 QLabels, 3 qMRMLNodeComboBox, a QPushButton and another ctkCollapsibleButton (which contains a QDoubleSpinBox and a QLabel). The distribution of these objects is shown in Figure 20, and Figure 21 shows the object inspector (in QtDesigner) for this block.

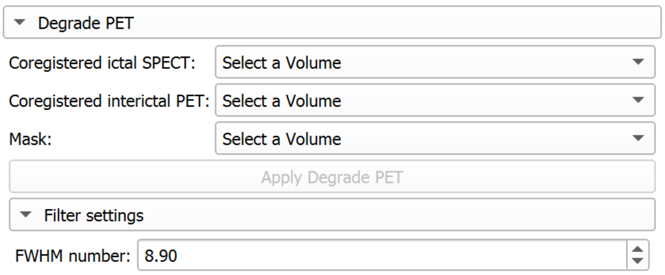


Figure 20: UI design of Degrade PET block

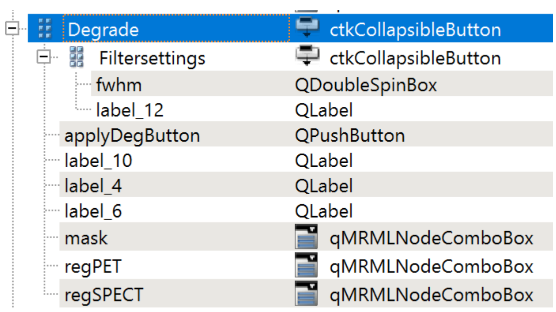


Figure 21: Objects and class of the Degrade PET block

The Apply Degrade PET QPushButton is not checkable when the module is initialized. In PISCOMmodule.py inside the PISCOMmoduleWidget *class*, and the updateGUIFromParameterNode function, it is defined that if the regSPECT, the regPET and the mask nodes are filled, the Apply button is then enabled.

The Filtersettings ctkCollapsibleButton has the property *collapse* set to True, since the value of the QDoubleSpinBox should not be changed by the user unless the SPECT and PET acquisition systems of the images were different to the ones used in the published paper [7]. The Full Width at Half Maximum (FWHM) number of the FDG PET filter was derived from experimental PSF images of the SPECT and PET acquisition systems, thus only if the images came from other equipment, this value should be derived again and changed.

The QDoubleSpinBox has the default value set to 8.9 (the one obtained in [7]) and its relevant properties are shown in Figure 22.

fwhm : QDoubleSpinBox

Property	Value
showGroupSeparator	☐
QDoubleSpinBox	
prefix	
suffix	
decimals	2
minimum	0.000000
maximum	50.000000
singleStep	0.010000
stepType	DefaultStepType
value	8.900000

Figure 22: Property editor fmwh QDoubleSpinBox

Logics

The Apply Degrade PET QPushbutton, when clicked, computes the onApplyDegButton function found in the PISCOMmoduleWidget *class*, which calls the PISCOMmoduleLogic() *class* and tries to execute the degradeButton() function. This function takes as input parameters: the regSPECT, the regPET and mask nodes and the fmwh value of the QDoubleSpinBox. The degradeButton() function runs the degradePET() function to obtain the maskedSPECT, the maskedPET, the PETnorm (normalized without the filter applied) and the PETfiltered (with both normalization and filter applied) images. Then, it saves the output images to the working directory and loads them to Slicer. Further detail of what the degradeButton() function does and its dependencies can be found commented on the PISCOMmodule.py. Finally, once the degradeButton() function returns the outputs, the currentNodeID of the maskedSPECT and degPET input volume nodes (found in the next block), are set to the corresponding obtained outputs.

5.1.4. Subtraction

UI design

The UI design of this block consists of a ctkCollapsibleButton with a Layout in a Form Layout disposition containing: 3 QLabels, 2 qMRMLNodeComboBox, a QCheckBox and a QPushButton. The distribution of these objects is shown in Figure 23, and Figure 24 shows the object inspector (in QtDesigner) for this block.

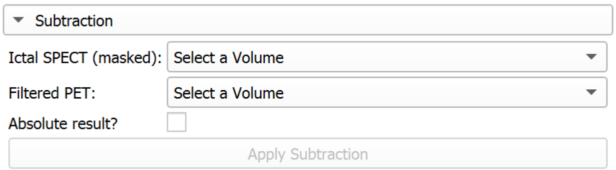


Figure 23: UI design of Subtraction block

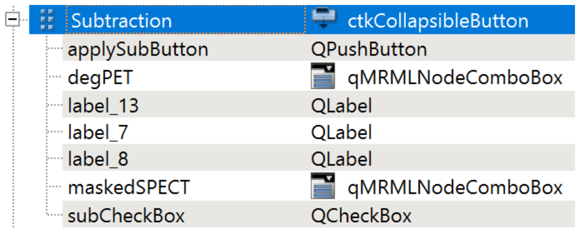


Figure 24: Objects and class of the Subtraction block

The QCheckBox *checkable* property is set to True, and the *checked* property is set to False. The Apply Subtraction QPushButton is not checkable when the module is initialized. In PISCOMmodule.py inside the PISCOMmoduleWidget *class*, and the updateGUIFromParameterNode function, it is defined that if the maskedSPECT and the degPET nodes are filled, the Apply button is then enabled.

Logics

The Apply Subtraction QPushbutton, when clicked, computes the onApplySubButton function found in the PISCOMmoduleWidget *class*, which calls the PISCOMmoduleLogic() *class* and tries to execute the SubButton() function. This function takes as input parameters: the maskedSPECT and the degPET nodes and the boolean corresponding to the subCheckBox. The SubButton() function runs the subtraction() function to obtain the subtraction image, saves the output image to the working directory and loads it to Slicer. If the subCheckBox is checked, instead of computing the relative subtraction image, the subtraction() function computes the absolute subtraction image. Further detail of what the SubButton() function does and its dependencies can be found commented on the PISCOMmodule.py**¡Error! No se encuentra el origen de la referencia.**. Finally, once the SubButton() function returns the outputs, the currentNodeID of the subtraction input volume node (found in the next block), is set to the corresponding obtained output.

5.1.5. Fusion

UI design

The UI design of this block consists of a ctkCollapsibleButton with a Layout in a Form Layout disposition containing: 5 QLabels, a qMRMLNodeComboBox, 2 QSpinBoxs, a QComboBox, a QCheckBox and a QPushButton. The distribution of these objects is shown in Figure 25, and Figure 26 shows the object inspector (in QtDesigner) for this block.

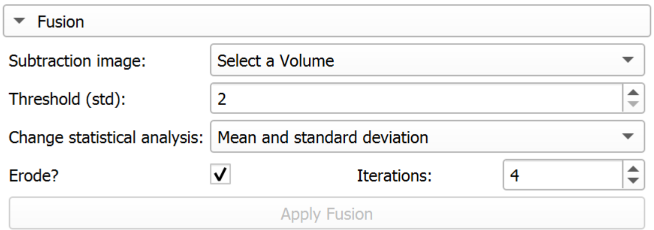
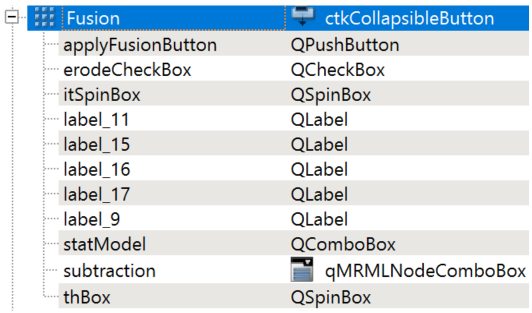


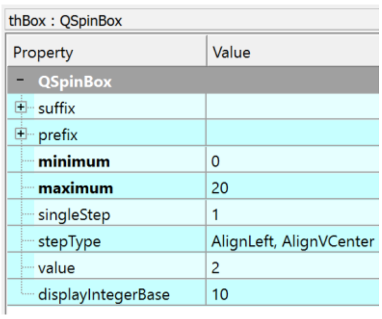
Figure 25: UI design of Subtraction block



Fusion	ctkCollapsibleButton
applyFusionButton	QPushButton
erodeCheckBox	QCheckBox
itSpinBox	QSpinBox
label_11	QLabel
label_15	QLabel
label_16	QLabel
label_17	QLabel
label_9	QLabel
statModel	QComboBox
subtraction	qMRMLNodeComboBox
thBox	QSpinBox

Figure 26: Objects and class of the Subtraction block

The thBox QSpinBox is the input for the threshold value (how many standard deviations over the mean does the user want to select). This value is set to 2 by default, but it can be increased up to 20. The relevant properties of this QSpinBox are shown in Figure 27.



thBox : QSpinBox

Property	Value
QSpinBox	
suffix	
prefix	
minimum	0
maximum	20
singleStep	1
stepType	AlignLeft, AlignVCenter
value	2
displayIntegerBase	10

Figure 27: Property editor for thBox QSpinBox

The statModel QComboBox allows to change the statistic model for the selection of the intensity values higher than the threshold. By default, the mean and the standard deviation (sd) are used to compute the threshold as: th = mean + 2·sd. However, the QComboBox allows to change the statistic model so that the threshold is calculated with the median and the interquartile range (iqr) as: th = median + 2·iqr.

The erodeCheckBox *checkable* and *checked* properties are set to True. It allows the user to indicate if an erosion is desired in the mask, so that the outliers in the extremes of the brain are not selected. By default, it is set to True since the fusion results were found to be more conclusive; the seizure onset zone was better highlighted. The itSpinBox allows the user to indicate how many iterations of erosion wants to apply to the mask when the erodeCheckBox is checked. By default is set to 4. The relevant properties of the itSpinBox are shown in Figure 28.

itSpinBox : QSpinBox

Property	Value
QSpinBox	
suffix	
prefix	
minimum	1
maximum	20
singleStep	1
stepType	DefaultStepType
value	4
displayIntegerBase	10

Figure 28: Property editor for itSpinBox

The Apply Fusion QPushButton is not checkable when the module is initialized. In PISCOMmodule.py inside the PISCOMmoduleWidget *class*, and the updateGUIFromParameterNode function, it is defined that if the subtraction, the inputMRI and the mask nodes are filled, the Apply button is then enabled.

Logics

The Apply Fusion QPushbutton, when clicked, computes the onApplyFusionButton function found in the PISCOMmoduleWidget *class*, which calls the PISCOMmoduleLogic() *class* and tries to execute the FusionButton() function. This function takes as input parameters: the subtraction, the inputMRI and mask nodes, the erodeCheckBox Boolean the thBox value, the statModel index and the itSpinBox value. The FusionButton() function runs the selection() function to obtain the fusion image, saves the output image to the working directory and loads it to Slicer, setting the MRI image as the background and the fusion image as the foreground. If the erodeCheckBox is checked, instead of taking the values of the subtraction image included in the mask previously computed, it erodes this mask and uses the eroded mask to select the values to statistically analyse. Further detail of what the FusionButton() function does and its dependencies can be found commented on the PISCOMmodule.py.

5.1.6. Final result

A QTextBrowser was added to the top of the buttons to recommend to the user to load the three input images to Slicer before running the program, and to have these images in the same folder. This text is separated from the buttons with a vertical Spacer of a fixed 20x10 size. At the bottom of the buttons, another vertical Spacer was placed with an expanding height size.

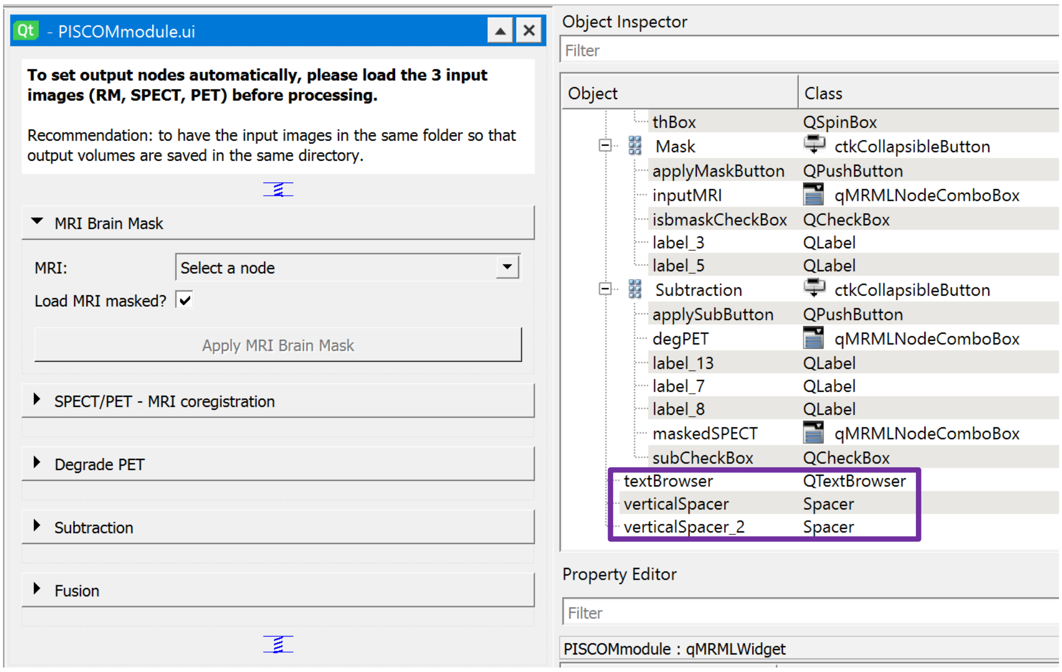


Figure 29: Final distribution of the module UI in QtDesigner

In Figure 30, the visualization of the module's graphical interface when Slicer is initialized is shown.

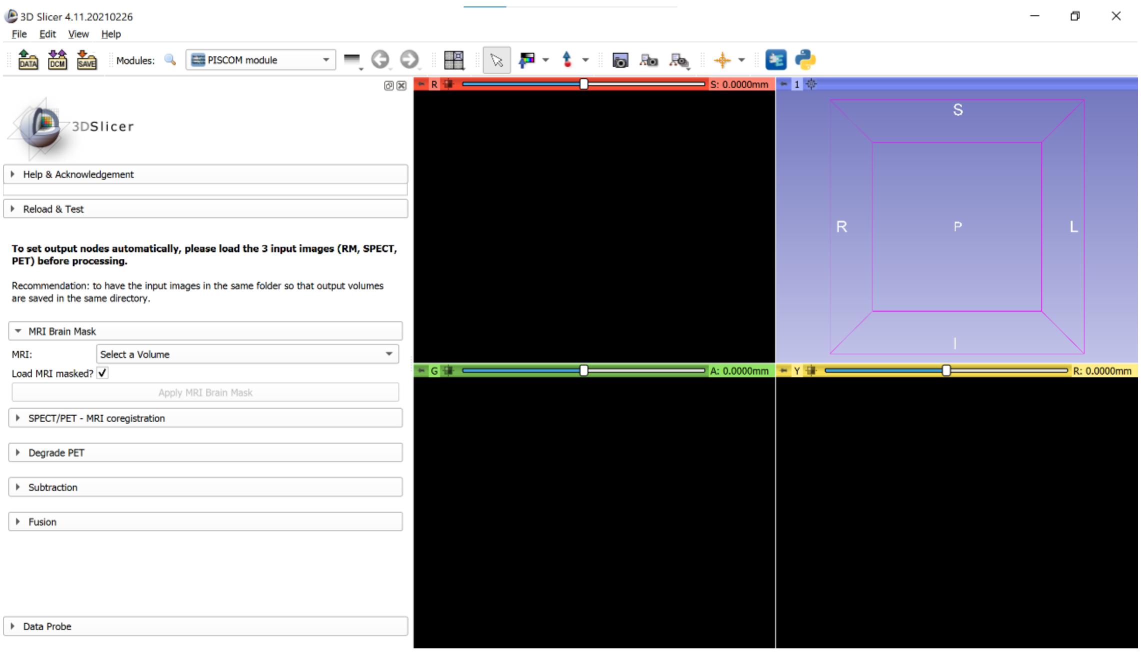


Figure 30: PISCOM module on 3D Slicer

Figure 31 shows the output result after using the PISCOM module.

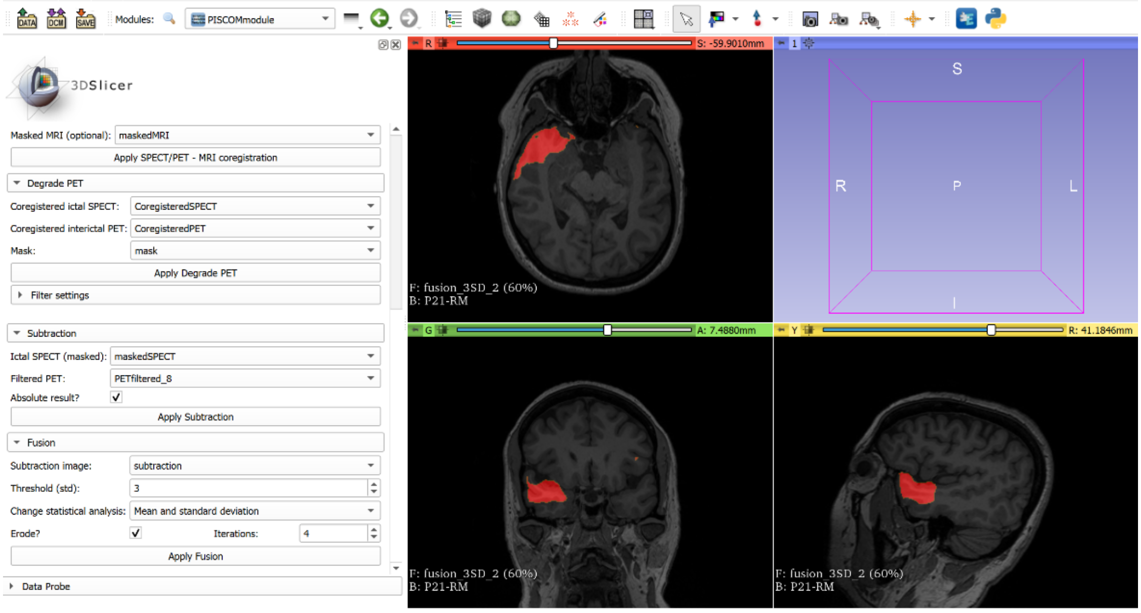


Figure 31: Example of final output of the PISCOM module

Finally, the PISCOM Extension's metadata was modified to describe the extension content and its authors and indicate the dependency on the SlicerElastix extension. Figure 32 shows the modified CMake.Lists.txt file.

```
cmake_minimum_required(VERSION 3.13.4)

project(PISCOM)

#-----------------------------------------------------------------------------
# Extension meta-information
set(EXTENSION_HOMEPAGE "https://www.slicer.org/wiki/Documentation/Nightly/Extensions/PISCOM")
set(EXTENSION_CATEGORY "Examples")
set(EXTENSION_CONTRIBUTORS "Laura Latorre (Universitat de Barcelona), ")
set(EXTENSION_DESCRIPTION "PISCOM extension is composed by one module. PISCOM module integrates

For further information: Perissinotti, A., Niñerola-Baizán, A., Rubí, S. et al.
PISCOM: a new procedure for epilepsy combining ictal SPECT and interictal PET. Eur J Nucl Med M
<a href="https://doi.org/10.1007/s00259-018-4080-6">https://doi.org/10.1007/s00259-018-4080-6")
set(EXTENSION_ICONURL "http://www.example.com/Slicer/Extensions/PISCOM.png")
set(EXTENSION_SCREENSHOTURLS "http://www.example.com/Slicer/Extensions/PISCOM/Screenshots/1.png
set(EXTENSION_DEPENDS "SlicerElastix-master") # Specified as a list or "NA" if no dependencies

#-----------------------------------------------------------------------------
# Extension dependencies
find_package(Slicer REQUIRED)
include(${Slicer_USE_FILE})

#-----------------------------------------------------------------------------
# Extension modules
add_subdirectory(PISCOM)
add_subdirectory(PISCOMmodule)
## NEXT_MODULE

#-----------------------------------------------------------------------------
include(${Slicer_EXTENSION_GENERATE_CONFIG})
include(${Slicer_EXTENSION_CPACK})
```

Figure 32: CMakeLists.txt file of PISCOM extension

5.2. Technical specifications

5.2.1. Installation Manual

An installation manual has been designed to ease the installation process and record all the steps needed to make the extension functional. The module depends on some Python packages that are not installed by default on Slicer and on the Slicer's external extension called *SlicerElastix*, that has been slightly modified so that is adapted to the module needs.

It can be found in 12.2. Installation manual.

5.2.2. User Manual

A user manual has been designed so that the clinicians can follow the instructions necessary to manipulate and interpret the PISCOM extension and the 3D Slicer software utilities. It will be delivered before the software evaluation phase, so that they have an initial idea of the software and the module before evaluating them.

It can be found in 12.3. User Manual.

5.3. Discussion. Limitations and future implementations

Referring to the 1.1. Objectives, one of the goals was to make the graphical interface ergonomic and user-friendly, being that decisive when bringing the software into clinical practice.

To assess the graphical interface, the questionnaire found in 12.4. Software evaluation was generated so that two clinicians from the Department of Nuclear Medicine in Hospital Clínic, Andrés Perissinoti and Xavier Setoain, could fill it out after the first contact with the new software. Clinicians will be the final users of the extension and thus the most indicated ones to perform this evaluation.

The first step of the evaluation was the installation of Slicer and PISCOM extension to an external computer, different to the one where the software was developed. The process can be cumbersome since it is not fully automatized and requires the installation of some python packages from line command at Slicer Python interactor, including the modification of the script of one of these packages installed. Nonetheless, the 12.2. Installation manual created was understood and the installation was successful. However, the process was not tried in the equipment of the hospital, where some installation restrictions remain and what could lead to a significant limitation to consider.

Among the possible drawbacks that Slicer would have over other image processing software could be that the navigation through Slicer might not be intuitive. To tackle this problem, the module was designed in a way that the user would have the minimum possible interaction with Slicer to run the PISCOM algorithm. For instance, the output images are automatically saved to the user working directory, then loaded to Slicer, and when necessary selected automatically in their corresponding input nodes. Another example is the fact that the final output image, the fusion subtraction image, is displayed fused with the MRI image at the end of the process. Additionally, a User Manual (12.3. User Manual) that describes the main functionalities to run the PISCOM module and display the different output volumes was generated. For further information, the Slicer website [17] contains "How to" tutorials with matched sample data sets that demonstrate how to use the 3D Slicer environment.

The clinicians answered that the manual is useful and clear, that the module is not difficult to interpret nor manipulate, and that it is easy to follow the whole process. Thus, this handicap was successfully overcome.

Another concern, linked with the previous one, was that the View controllers and tools for the 2D anatomical viewers offered by Slicer, might be poor compared to other medical image viewers, like ITK-SNAP [50], and that other software would be required for visualizing the final output images. However, during the software evaluation session it was understood that both clinicians were familiarised with similar neuroimaging software and that the tools offered by Slicer were enough for a proper experience. The answer to question "Does the visualization of the images fulfil your needs?" was positive for both clinicians.

Referring to the operating speed of the algorithm, the answer was also positive. Some parts of the initial algorithm were time consuming; the subtraction and fusion steps used *for loops* to operate at a pixel level with the images. This was optimized thanks to the fact that numpy array structures are easy to manipulate without the need of loops commands, using the usual class operators instead. Conversely, the computation of the mask is the most time and graphics consuming, depending on the computer, it can consume up to 30 seconds. Nonetheless, the overall process does not take more than 3 minutes, so the operating speed was found adequate by clinicians.

The structural quality of the module was considered appropriate for clinical use. The module pulldowns, the inputs and the organization do not need any change. Moreover, the output images saved at each block were considered adjusted for the diagnosis; the module does not lack any functionality or output according to the answers of the questionnaire.
Nevertheless, some possible future implementations came up. One of the suggestions was to display ictal, interictal and fusion images simultaneously in the same panel to perform a visual examination of the 3 modalities fused with the MRI all at once at the end of the process. It is known that Slicer offers this visualization modality, in the Slicer's toolbar, when clicking the layout menu icon, the user can choose among different visualization options, and one of them is: Three by three slice. Thus, the functions in *slicer.util* python package should be studied in order to find if there exists one to change the layout view of Slicer and to choose the volumes set to each slice. Then, FusionButton() function would be modified so that those the registered ictal SPECT, the registered interictal PET and the fusion (or PISCOM resulting) image were automatically displayed after clicking on the Apply Fusion QPushButton. Additionally, the display colour maps suggested for the ictal and interictal volumes were Sokoloff, Spectrum and Warm Metal (GE). It should be studied if Slicer offers one of these colour scales.
Another concern was the possibility to import the images in DICOM format to Slicer and reconstruct them to NIFTI format (the one used for in the PISCOM algorithm) with the software. Slicer recognises DICOM images, and, concerning to the reconstruction, the solutions offered should be studied [51].

Furthermore, the need to compare the PISCOM and SISCOM resulting images came to light. The PISCOM module allows to obtain the SISCOM fusion image if the interictal SPECT image is available. The SISCOM output image was obtained using the PISCOM module during the software evaluation. As a result of the success of the PISCOM module in Slicer software, it was suggested to create another module to integrate the SISCOM algorithm.

Overall, the assessment was favourable and promising. Clinicians considered it as a useful and user-friendly tool for applying the PISCOM technique, with future potential, that could be used not only in the Epilepsy Unit of Hospital Clínic de Barcelona, but also in other Epilepsy Units in a future. This could lead to reducing one image acquisition (interictal SPECT) in those patients with refractory epilepsy in which all the functional neuroimaging scans (ictal SPECT, interictal SPECT and interictal PET) are under consideration for an appropriate diagnosis.

6. EXECUTION PLAN

6.1. Tasks and time definition. Phases and milestones

To have an idea of the different tasks to complete and to be able organize them in time, a work-breakdown structure (WBS) has been created.

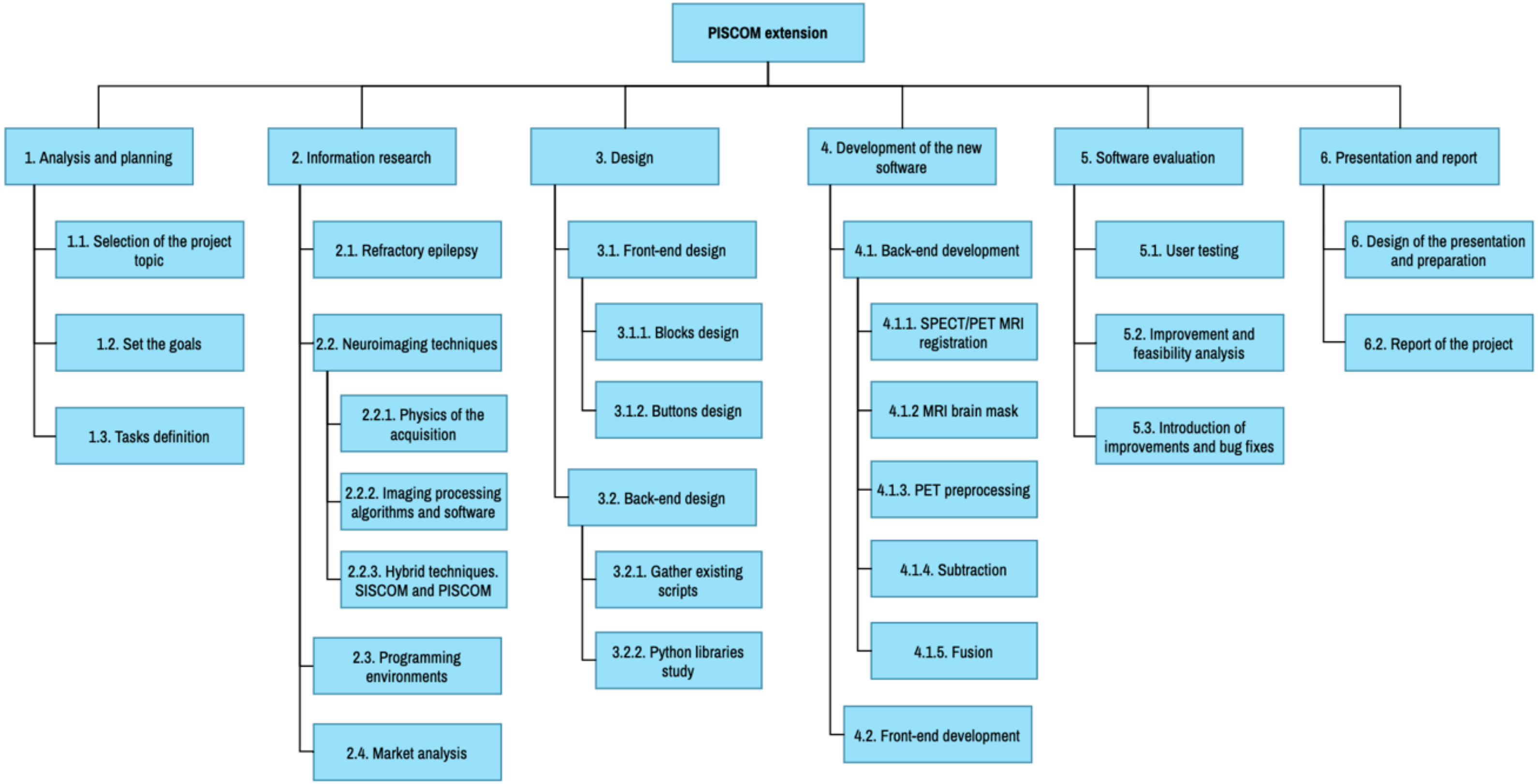


Figure 33: Work-breakdown structure of the project

WBS dictionary

1. **Analysis and planning**: project and tasks definition
   - 1.1. **Selection of the project topic**: the tutor proposes several lines of work and the most appealing is chosen. Deliverable: 07/04/20
   - 1.2. **Set the goals**: definition of the most important points to be achieved at the end of the project. Deliverable: 08/06/20
   - 1.3. **Tasks definition**: creation of a work-breakdown structure that defines the different tasks during the project execution. Deliverable: 08/06/20
2. **Information research**: gathering useful information for the development of the project
   - 2.1. **Refractory epilepsy**: deep research about drug-resistant epilepsy physiopathology, diagnosis and treatment.
   - 2.2. **Neuroimaging techniques**: research on the current neuroimaging techniques to identify the epileptogenic zone.
     - 2.2.1. **Physics of the acquisition**: study of the physics behind each imaging technique.
     - 2.2.2. **Imaging processing algorithms and software**: study of the different imaging processing algorithms and software to obtain the desired image for the diagnosis.
     - 2.2.3. **Hybrid techniques. SISCOM and PISCOM**: deep study on the image processing behind SISCOM and PISCOM analysis.
   - 2.3. **Programming environments**: research and study of the different options to develop a graphical interface.
   - 2.4. **Market analysis**: search for information on the sector to which our application is directed and the possible evolution of this sector over time.
3. **Design**: process that will define how the new software will be designed and structured, and what functions it will perform.
   - 3.1. **Front-end design**: design of the presentation layer
     - 3.1.1. **Blocks design**: design of the layout of all the elements in each block of the algorithm. Includes button positions, text, etc.
     - 3.1.2. **Buttons design**: design of the action that will be associated with every button.
   - 3.2. **Back-end design**: definition of the different functions that the new software will perform.
     - 3.2.1. **Gather existing scripts**: PISCOM algorithms are separated in different scripts. An analysis of the parts of the algorithms that can be used in the integration is performed.
     - 3.2.2. **Python libraries study**: those algorithm steps that need to be reformulated in the new programming environment probably can be performed by an already existing python library.
4. **Development of the new software**: implementation of all the functional and design concepts defined above. Deliverable: 07/05/21

- **4.1. Back-end development**: implementation of the programming code necessary to grant functionality to our application.
  - **4.1.1. SPECT/PET – MRI co-registration**: assessment of the different solutions to perform the co-registration and implementation of the chosen one.
  - **4.1.2. MRI brain mask**: assessment of the different solutions to perform an MRI brain mask and implementation of the chosen one.
  - **4.1.3. PET preprocessing**: integration of the PET preprocessing algorithm.
  - **4.1.4. Subtraction**: integration and optimization of the subtraction algorithm.
  - **4.1.5. Fusion**: integration and optimization of the fusion algorithm.
- **4.2. Front-end development**: implementation of the programming code that defines the visual and graphical way in which the information on each screen will be transmitted to the user.

5. **Software evaluation**: check the correct operation of the software, as well as detect errors and possible improvements. Deliverable: 02/06/21
   - **5.1. User testing**: the software will be tested with a certain number of users of the Nuclear Medicine department.
   - **5.2. Improvement and feasibility analysis**: during the tests, the different errors detected will be recorded and possible improvements will be analysed. Finally, it will be necessary to study how to correct these errors and select those improvements that are feasible.
   - **5.3. Introduction of improvements and bug fixes**: implementing the chosen improvements and correcting bugs in the software.
6. **Presentation and report**: design of the presentation and submission of the report of the project.
   - **6.1. Graphic design of the presentation**: the presentation will be designed to summarize the whole project (10 minutes presentation) in a clear, short, concise and pleasing to the eye format using the Power Point platform. Deliverable: 22/06/21
   - **6.2. Report of the project**: throughout the project's execution, the report will be edited to reach the weekly deliverables proposed by the tutor, until 15th January 2021, when the final submission must be done. Deliverable: 14/06/21

6.2. Precedence analysis and critical path method

In order to determine the time (in hours) that each task will require, the following table has been created, which determines the PERT time as:

$$t_{PERT} = \frac{t_{optimistic} + 4 * t_{normal} + t_{pessimistic}}{6}$$

	Identification	PERT time (h)	Optimistic time (h)	Normal time (h)	Pessimistic time (h)
1.1. Selection of the project topic	A	3	2	3	4
1.2. Set the goals	B	3	2	3	4
1.3. Tasks definition	C	3	2	3	4
2.1. Drug-resistant epilepsy	D	8	6	8	10
2.2. Neuroimaging techniques	E	12,83	10	13	15
2.3. Programming environments	F	6,17	5	6	8
2.4. Market analysis	G	4,17	3	4	6
3.1. Front-end design	H	22,83	20	23	25
3.2. Back-end design	I	25,67	20	26	30
4.1. Back-end development	J	70	65	70	75
4.2. Front-end development	K	70	65	70	75
5.1. User testing	L	2	1	2	3
5.2. Improvement and feasibility analysis	M	3	2	3	4
5.3. Introduction of improvements and bug fixes	N	10,33	7	10	15
6.1. Graphic design of the presentation	O	15	10	15	20
6.2. Report of the project	P	85	80	85	90
	Total	**344**	**300**	**344**	**388**

Table 1: PERT, optimistic, normal and pessimistic time of each task

Since it is an individual project, no task can be done simultaneously, and therefore the critical path will be composed by all the activities, the critical time would be then 344 hours.

6.3. Schedule (timing)

In order to schedule and control the different activities over time, the following GANTT diagram has been created.

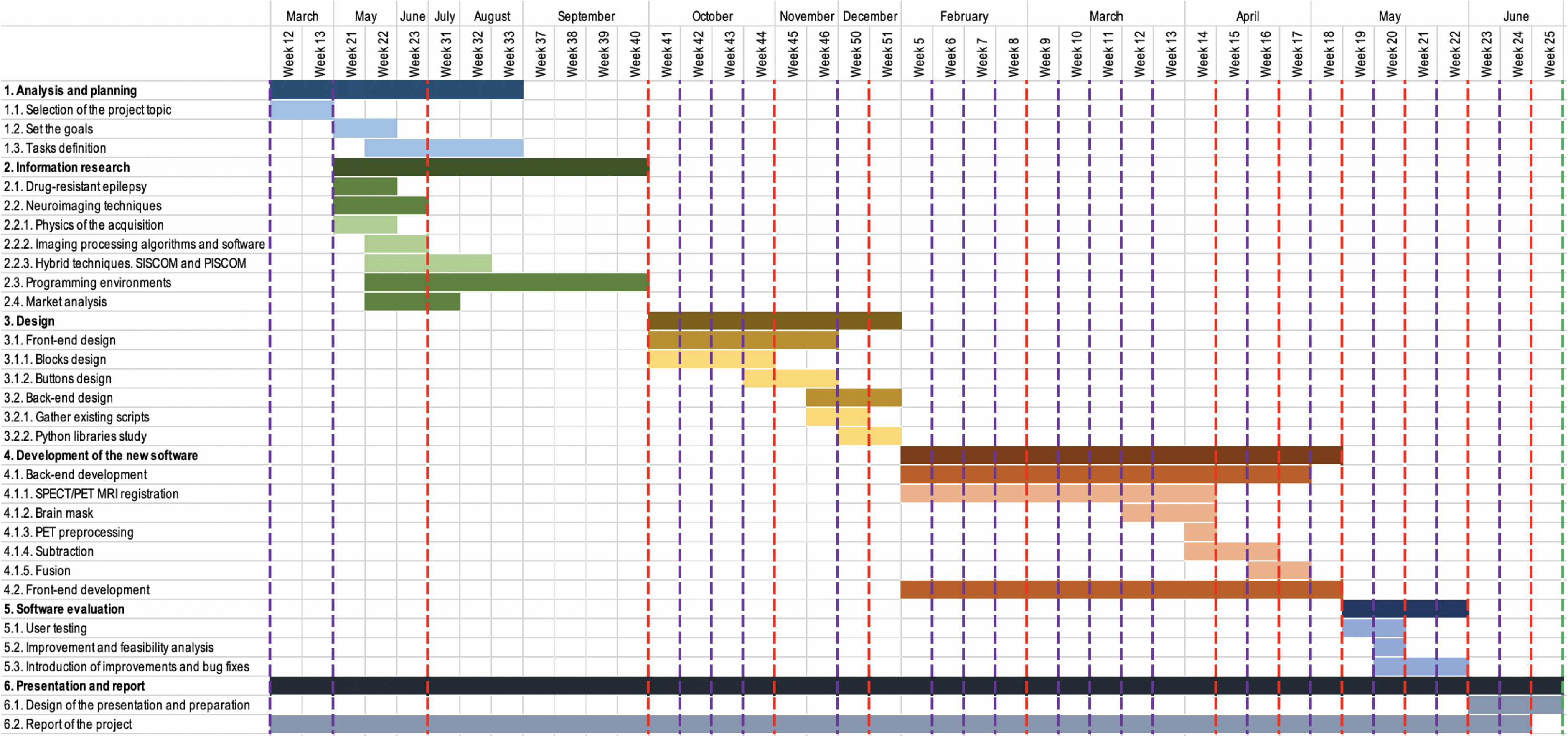


Figure 34: GANTT diagram (red lines are controls and purple lines are meetings)

The activities are distributed over weeks, since is a long project and thus is more visual. The meetings with the tutor are scheduled every week in order to solve ongoing problems or doubts that may appear during the execution of the project. The following controls have been established:

→ **06/06/20**: Revision of the proposal of the project.
→ **02/10/20**: Restart after the summer, planning the design phase.
→ **30/10/20**: Control of the design progress.
→ **11/12/20**: Control of the progress, design phase ends. Development phase starts.
→ **26/02/21**: Control of the development progress.
→ **09/04/21**: Control of the development progress. 3 tasks of this stage end.
→ **23/04/21**: Control of the development progress.
→ **07/05/21**: Development phase ends. Software evaluation phase starts.
→ **21/05/21**: Control of the software evaluation phase.
→ **02/06/21**: Results shown to clinicians.
→ **11/06/21**: Conclusion of the project.
→ **22/06/21**: Presentation of the project.

7. TECHNICAL FEASIBILITY

Figure 35 shows the SWOT diagram that analyses the technical feasibility of the project.

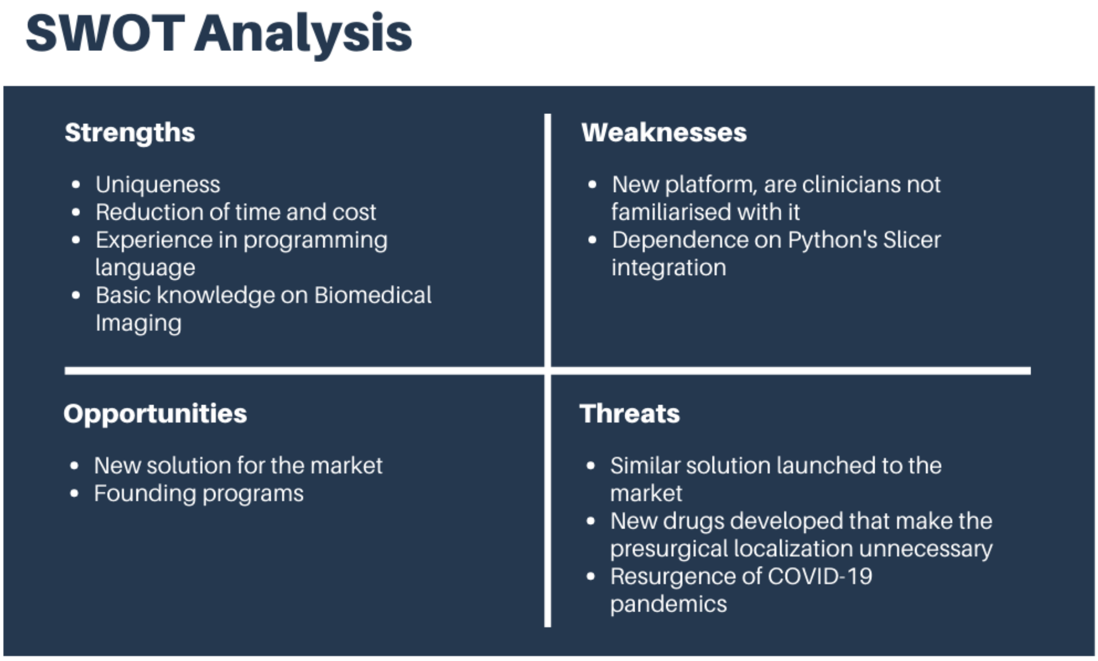


Figure 35: SWOT analysis of the project

STRENGTHS

- *Uniqueness*: PISCOM technique has only been assessed in Hospital Clínic de Barcelona together with Universitat de Barcelona and no software that integrates this technique has been launched to the market.
- *Reduction of time and cost*: The new graphical interface reduces time and cost of the diagnosis process with one acquisition image less.
- *Experience in programming environment*: Previous experience in Python language. The SISCOM algorithm was implemented in Python as a project for a subject in the Biomedical Engineering degree in UB.
- *Basic knowledge on Biomedical Imaging*: I have been learning the basics of image processing during this last year. This will reduce the time required to understand some of the algorithms for processing an image.

WEAKNESSES

- *New platform, clinicians are not familiarised with it*: This might entail a bigger inversion in time to understand the new software, 3D Slicer, and could lead to a less intuitive graphical interface.

- *Dependence on Python's Slicer integration*: Slicer has a built-in Python console. However, not all Python packages can be installed on Slicer; those that do not have a pip command to be installed and need a build process are not recognised by Slicer.

OPORTUNITIES

- *New solution for the market*: It is possible that the results of this project can pave the way to create a commercial solution.
- *Founding programs*: they can be interested in the results of this project. AGAUR (Agència de Gestió d'Ajuts Universitaris i de Recerca) is an instrument of the Generalitat de Catalunya to provide service and support to the people and institutions that are part of the university and research system in Catalonia.

THREATS

- *Similar solution launched to the market*: New solutions to locate the epileptogenic focus with other techniques can be released while the project is being developed.
- *New drugs developed that make the presurgical localization unnecessary*: New drugs that can treat drug-resistant epilepsy can come to light while the project is being developed.
- *Resurgence of COVID-19 pandemics*: This would force reformulating some project methodologies such as the place of development, the communication with people involved and the format of the final presentation.

8. ECONOMICAL FEASIBILITY

The aim here is to study the cost that entails the project's execution.

The budget is divided into several parts:

1. **Materials**

   a. *Hospital equipment*

   To assess the final graphical interface, several medical images were required; at least 5 tests of the extension performance were done. Table 1 shows the cost of each acquisition:

	Unit price (€)	Units	Total (€)
Brain FDG PET interictal	1.121,00	5	5.605,00
Brain perfusion tomography (SPECT) ictal	358,00	5	1.790,00
Brain MRI	203,00	5	1.015,00
Anaesthesia	115,00	15	1.725,00
		Total	**10.135,00€**

Table 2: Cost of the hospital equipment

   b. *IT tools and software*

   Table 2 shows the cost of IT tools and software:

	Unit price (€)	Units	Total (€)
Office 356 licence	7,00/month	12	64,00
MacBook Pro (13-inch, 2019, Two Thunderbolt 3 ports)	1.575,00	1	1.575,00
		Total	**1.639,00€**

Table 3: Cost of IT tools and software

2. **Labour**

To estimate the cost of the student, the main developer of the project, it has been considered the net annual salary of a junior engineer (graduate with an experience of no more than two years), which is around 25.000,00€/year in Spain. This leads to an approximate amount of 13€/hour. Universitat de Barcelona states that the student must dedicate at least 300 hours for developing the final degree project.

The cost of the labour of the tutor is estimated by the net annual salary of a Senior R&D Project Engineer, which is around 60.000,00€/year in Spain. This leads to an approximate amount of 30€/hour. The expected total hours of labour are 80 hours.

	Unit price (€/h)	Units (h)	Total (€)
Student	13,00	300	3.900,00
Tutor	30,00	80	2.400,00
		Total	**6.300,00€**

Table 4: Cost of labour

Final budget

Considering the above, the estimated cost for this project is around **18.074,00€**

9. REGULATION AND LEGAL ASPECTS

The final product of this project is an Slicer extension that allows to implement PISCOM analysis methodology into clinical routine. PISCOM is a neuroimaging processing proof-of-concept study that provides complementary information in the diagnosis and localization of the epileptogenic focus. According to the Medical Device Coordination Group (MDCG) from the European Commission, a medical device is defined as:

*"medical device" means any instrument, apparatus, appliance, **software**, implant, reagent, material or other article intended by the manufacturer to be used, alone or in combination, for human beings for one or more of the following specific medical purposes:*

- *diagnosis, prevention, monitoring, prediction, prognosis, treatment or alleviation of disease,*
- *diagnosis, monitoring, treatment, alleviation of, or compensation for, an injury or disability,*
- *investigation, replacement or modification of the anatomy or of a physiological or pathological process or state,*
- *providing information by means of in vitro examination of specimens derived from the human body, including organ, blood and tissue donations,*

and which does not achieve its principal intended action by pharmacological, immunological or metabolic means, in or on the human body, but which may be assisted in its function by such means. [52]

And a software medical device as:

Medical device software is software that is intended to be used, alone or in combination, for a purpose as specified in the definition of a "medical device" in the medical devices regulation or in vitro diagnostic medical devices regulation. [52]

Slicer is not approved for clinical use and the distributed application is intended for research use. It should be evaluated if the PISCOM extension is intended to be used exclusively as an experimental support in diagnosis, and it would be considered as a medical imaging research tool, so its regulation will be covered under the 3D Slicer license, or if the purpose is to use it for clinical diagnosis instead. The latter would make the extension to be considered as a medical device software, so it would be covered by the Regulation (EU) 2017/745 on Medical Devices (MDR). Additionally, it would be classified as class IIa, since Rule 11a) in Annex VIII states:

MDSW (which is intended to provide information which is used to take decisions with diagnosis or therapeutic purposes) is classified as class IIa. [52]

Therefore, the medical device would need a CE mark before being sold in the EU.

On the other hand, since the software will process patient-related data (medical images) on a relevant scale, it requires special protection. In Europe, the General Data Protection Regulation

(GDPR) is the legal basis for protecting individual's data. This regulation should also be taken into account when designing the software.

10. CONCLUSIONS AND FUTURE DIRECTIONS

The purpose of this project was to develop an ergonomic and user-friendly graphical interface that integrates PISCOM technique in a unique work environment and to assess its structural and performance quality to define if it would be brought into clinical practice.

Different options to develop this tool were studied analysing the already existing software and its usability in a medical environment. It was relevant to consider that final users of the new graphical interface would be clinicians in order to choose the optimal solution. Moreover, the historical evolution of the market that increases the demand on new and accurate diagnostic procedures in seizure onset zone localization -patients diagnosticated with drug-resistant epilepsy- was significant to take into account when estimating the future potential of the developed software.

3D Slicer was selected as the most suitable environment since it offers training materials to develop in Python language, a powerful and well-known technology for image processing and data analysis. The PISCOM algorithm had to be adapted to the Slicer's Python environment, and thus new medical image processing methods were studied for the co-registration and generation of the mask blocks. Moreover, new functionalities and tools were added to boost the user experience.

Based on a usability and quality assessment of the result, performed by two clinicians in the Department of Nuclear Medicine of Hospital Clínic, it can be concluded that the objectives were successfully achieved. The results indicate that the developed Slicer's extension is a useful and user-friendly tool for applying PISCOM technique that could be used not only in the Epilepsy Unit of Hospital Clínic de Barcelona, but also in other Epilepsy Units in a future.

Notwithstanding the future potential it has, further studies on the PISCOM technique are needed to clinically validate the results given in the proof-of-concept study, a process that will be eased and accelerated thanks to the software developed. Thus, the future directions would lie on performing the clinical validation of PISCOM technique, and implementing the suggestions for improving the user experience proposed by the clinicians. The success in bringing PISCOM method to the clinical practice would lead to reduce one image acquisition (interictal SPECT) in some patients with refractory epilepsy.

By way of conclusion, the exposure to radiation could be shortened, and the patient's comfort improved, especially in those requiring sedation during image acquisition process, such as paediatric patients. Additionally, the cost and the time of the diagnosis could be reduced with this technique, optimizing the resources the Health System has to use for the refractory epilepsy treatment.

11. REFERENCES

1. Adams RD, Victor M, Ropper AH. Principles of Neurology. 6th ed. McGraw-Hill; 1998. p313.
2. Xue-Ping W, Hai-Jiao W, Li-Na Z, Xu D, Ling L. Risk factors for drug-resistant epilepsy: A systematic review and meta-analysis. Medicine. 2019;98(30):e16402.
3. Setoain X, Carreño M, Pavía J, Martí-Fuster B, Campos F, Lomeña F. PET y SPECT en la epilepsia. Revista Espanola de Medicina Nuclear e Imagen Molecular. 2014;33(3):165–74.
4. Kwan P, Arzimanoglou A, Berg AT, Brodie MJ, Hauser WA, Mathern G, Moshé SL, Perucca E, Wiebe S, French J. Definition of drug resistant epilepsy: Consensus proposal by the ad hoc Task Force of the ILAE Commission on Therapeutic Strategies. Epilepsia. 2010;51(6):1069–77.
5. Lüders HO, Najm I, Nair D, Widdess-Walsh P, Bingman W. The epileptogenic zone: general principles. Epileptic disorders : international epilepsy journal with videotape. 2006 Aug;8 Suppl 2:S1-9.
6. O'Brien TJ, O'Connor MK, Mullan BP, Brinkmann BH, Hanson D, Jack CR, So EL. Subtraction ictal SPET co-registered to MRI in partial epilepsy: description and technical validation of the method with phantom and patient studies. Nuclear medicine communications. 1998 Jan;19(1):31–45.
7. Perissinotti A, Niñerola-Baizán A, Rubí S, Carreño M, Marti-Fuster B, Aparicio J, Mayoral M, Donaire A, Sanchez-Izquierdo N, Bargalló N, Rumiá J, Boget T, Pons F, Lomeña F, Ros D, Pavía J, Setoain X. PISCOM: a new procedure for epilepsy combining ictal SPECT and interictal PET. European Journal of Nuclear Medicine and Molecular Imaging. 2018;45(13):2358–67.
8. Nuclear Medicine [Internet]. [cited 2020 May 31]. Available from: https://www.nibib.nih.gov/science-education/science-topics/nuclear-medicine
9. Penfield W, von Sántha K, Cipriani A. CEREBRAL BLOOD FLOW DURING INDUCED EPILEPTIFORM SEIZURES IN ANIMALS AND MAN. Journal of Neurophysiology [Internet]. 1939;2(4):257–67. Available from: https://doi.org/10.1152/jn.1939.2.4.257
10. Martí Fuster B. Image processing of emission tomography studies in refractory epilepsy. 2013.
11. Single-photon emission computed tomography - Wikipedia [Internet]. [cited 2020 Jun 1]. Available from: https://en.wikipedia.org/wiki/Single-photon_emission_computed_tomography

12. Currie S, Hoggard N, Craven IJ, Hadjivassiliou M, Wilkinson ID. Understanding MRI: Basic MR physics for physicians. Vol. 89, Postgraduate Medical Journal. The Fellowship of Postgraduate Medicine; 2013. p. 209–23.

13. Magnetic Resonance Imaging (MRI) [Internet]. [cited 2020 Jun 2]. Available from: https://www.nibib.nih.gov/science-education/science-topics/magnetic-resonance-imaging-mri

14. Spyder Website [Internet]. [cited 2020 Jun 2]. Available from: https://www.spyder-ide.org/

15. Introduction [Internet]. [cited 2020 Jun 2]. Available from: http://www.gimias.org/index.php/whatisgimias/introduction

16. Fedorov A, Beichel R, Kalpathy-Cramer J, Finet J, Fillion-Robin J-C, Pujol S, Bauer C, Jennings D, Fennessy F, Sonka M, Buatti J, Aylward S, Miller J v, Pieper S, Kikinis R. 3D Slicer as an image computing platform for the Quantitative Imaging Network. Magnetic resonance imaging. 2012 Nov;30(9):1323–41.

17. 3D Slicer [Internet]. [cited 2021 Mar 4]. Available from: https://www.slicer.org/

18. Qt Designer Manual [Internet]. [cited 2021 May 12]. Available from: https://doc.qt.io/qt-5/qtdesigner-manual.html

19. Focusdet [Internet]. [cited 2020 Jun 5]. Available from: https://www.ciber-bbn.es/programas-transversales/transferencia-industrial/oferta-tecnologica/focusdet

20. Qubiotech [Internet]. [cited 2020 Jun 2]. Available from: https://qubiotech.com/es/neurocloud/siscom

21. Fu P, Zhang F, Gao J, Jing J, Pan L, Li D, Wei L. Neurogam software analysis in epilepsy diagnosis using 99mTc-ECD brain perfusion spect imaging. Medical Science Monitor. 2015;21:2801–8.

22. Long Z, Hanson DP, Mullan BP, Hunt CH, Holmes DR, Brinkmann BH, O'Connor MK. Analysis of Brain SPECT Images Coregistered with MRI in Patients with Epilepsy: Comparison of Three Methods. Journal of Neuroimaging. 2018;28(3):307–12.

23. Unidad de Epilepsia | Hospital Clínic Barcelona [Internet]. [cited 2020 Jun 3]. Available from: https://www.clinicbarcelona.org/unidad/epilepsia

24. UNIDADES DE EPILEPSIA DE REFERENCIA DEL SISTEMA NACIONAL DE SALUD | Alce Epilepsia valencia [Internet]. [cited 2020 Jun 3]. Available from: https://www.alceepilepsia.org/unidades-de-epilepsia-de-referencia-del-sistema-nacional-de-salud/

25. Epilepsy Alliance Europe | Epitarget, EBSACE & Radar CNS [Internet]. [cited 2020 Jun 3]. Available from: https://www.epilepsyallianceeurope.org/

26. Top 10 Medical Imaging Companies | Analytics Insight [Internet]. [cited 2020 Jun 3]. Available from: https://www.analyticsinsight.net/top-10-medical-imaging-companies/

27. Carrasco-García de León S, Burriel-Lobo L, González-Aduna L, Olivié L, Gudín Rodríguez-Magariños M, Vaamonde-Gamo J. Estudio epidemiológico en epilepsia farmacorresistente y criterios quirúrgicos de derivación: nuestra experiencia de 15 años. Neurologia Argentina. 2017 Jan 1;9(1):27–34.

28. Sheng J, Liu S, Qin H, Li B, Zhang X. Drug-Resistant Epilepsy and Surgery. Current Neuropharmacology. 2017;16(1):17–28.

29. Kalilani L, Sun X, Pelgrims B, Noack-Rink M, Villanueva V. The epidemiology of drug-resistant epilepsy: A systematic review and meta-analysis. Epilepsia. 2018;59(12):2179–93.

30. Privitera M. Epilepsy treatment: A futurist view. Epilepsy Currents. 2017;17(4):204–13.

31. GuiProgramming - Python Wiki [Internet]. [cited 2020 Jun 4]. Available from: https://wiki.python.org/moin/GuiProgramming

32. Documentation/Nightly/Training - Slicer Wiki [Internet]. [cited 2021 May 20]. Available from: https://www.slicer.org/wiki/Documentation/Nightly/Training#Tutorials_for_software_developers

33. Sublime Text - Text Editing, Done Right [Internet]. [cited 2021 Jun 7]. Available from: https://www.sublimetext.com/

34. Qt Designer's Signals and Slots Editing Mode | Qt Designer Manual [Internet]. [cited 2021 Jun 8]. Available from: https://doc.qt.io/qt-5/designer-connection-mode.html

35. Nilearn: Statistical Analysis for NeuroImaging in Python — Machine learning for NeuroImaging [Internet]. [cited 2021 Jun 7]. Available from: https://nilearn.github.io/modules/generated/nilearn.masking.compute_brain_mask.html#nilearn.masking.compute_brain_mask

36. DIPY : Docs 1.0.0. - Brain segmentation with median_otsu [Internet]. [cited 2021 Jun 7]. Available from: https://dipy.org/documentation/1.0.0./examples_built/brain_extraction_dwi/

37. DeepBrainSeg · PyPI [Internet]. [cited 2021 Jun 7]. Available from: https://pypi.org/project/DeepBrainSeg/

38. deepbrain · PyPI [Internet]. [cited 2021 Jun 7]. Available from: https://pypi.org/project/deepbrain/

39. Documentation/4.1/Modules/SkullStripper - Slicer Wiki [Internet]. [cited 2021 Jun 7]. Available from: https://www.slicer.org/w/index.php/Documentation/4.1/Modules/SkullStripper

40. Documentation/4.10/Modules/SwissSkullStripper - Slicer Wiki [Internet]. [cited 2021 Jun 7]. Available from: https://www.slicer.org/wiki/Documentation/4.10/Modules/SwissSkullStripper

41. deepbrain/extractor.py at master · rockstreamguy/deepbrain · GitHub [Internet]. [cited 2021 Jun 7]. Available from: https://github.com/rockstreamguy/deepbrain/blob/master/deepbrain/extractor.py

42. List of (non-rigid) image registration projects for Python | pyimreg.github.com [Internet]. [cited 2021 Jun 8]. Available from: http://pyimreg.github.io/

43. Registration — 3D Slicer documentation [Internet]. [cited 2021 Jun 8]. Available from: https://slicer.readthedocs.io/en/latest/user_guide/registration.html

44. GitHub - lassoan/SlicerElastix: This extension makes available Elastix medical image registration toolkit (http://elastix.isi.uu.nl/) available in Slicer. [Internet]. [cited 2021 Jun 8]. Available from: https://github.com/lassoan/SlicerElastix#slicerelastix

45. Klein S, Staring M, Murphy K, Viergever MA, Pluim JPW. elastix: A Toolbox for Intensity-Based Medical Image Registration. IEEE Transactions on Medical Imaging. 2010;29(1):196–205.

46. Shamonin DP, Bron EE, Lelieveldt BPF, Smits M, Klein S, Staring M, Avants B, Scheper TO. Fast parallel image registration on CPU and GPU for diagnostic classification of Alzheimer's disease. 2014; Available from: www.frontiersin.org

47. elastix [Internet]. [cited 2021 Jun 8]. Available from: https://elastix.lumc.nl/

48. Documentation/4.10/Modules/BRAINSFit - Slicer Wiki [Internet]. [cited 2021 Jun 8]. Available from: https://www.slicer.org/wiki/Documentation/4.10/Modules/BRAINSFit

49. Johnson H, Harris G, Williams K. BRAINSFit: mutual information rigid registrations of whole-brain 3D images, using the insight toolkit. Insight J. 2007;57(1):1–11.

50. ITK-SNAP Home [Internet]. [cited 2021 Jun 9]. Available from: http://www.itksnap.org/pmwiki/pmwiki.php?n=Main.HomePage

51. NA-MIC Project Weeks | Website for NA-MIC Project Weeks [Internet]. [cited 2021 Jun 10]. Available from: https://projectweek.na-mic.org/PW27_2018_Boston/Projects/DICOMVolumeReconstruction/

52. European Commission. MDCG 2019 11 Qualification and Classification of Software in Regulation (EU) 2017 / 745 – MDR and Regulation (EU) 2017 / 746 – IVDR. 2019;(October):27.

53. Release elastix 5.0.1 · SuperElastix/elastix · GitHub [Internet]. [cited 2021 Jun 9]. Available from: https://github.com/SuperElastix/elastix/releases/tag/5.0.1

12. ANNEXES

12.1. SlicerElastix modified extension

Some modifications on the SlicerElastix extension were needed so that the output co-registered image is read in .nii format, thus, it was downloaded from the Github repository [44] and modified, so that it could be delivered as a required extension to import on Slicer in the installation process (if the modified SlicerElastix extension is not imported, the PISCOMmodule can't work). The name of the folder of the extension is *SlicerElastix-master*, and inside this folder, the *Elastix* folder corresponds to the *General registration (Elastix)* module, which contains the Elastix.py and the rest of the module files.

Firstly, elastix for Windows [53] was downloaded (the 5.0.1-win64 version) to copy its content to the SlicerElastix-master\Elastix folder, so that the Elastix.py could find these exes and execute them during the Register process.

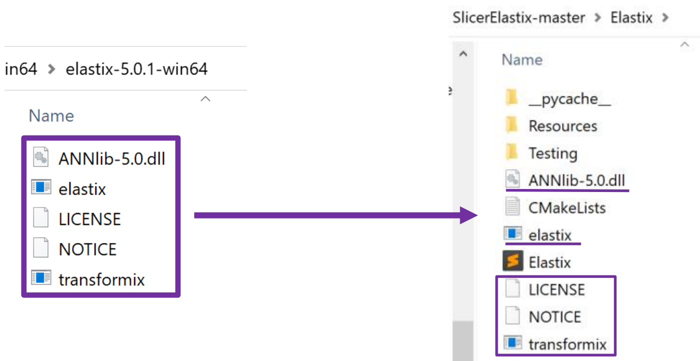


Figure A1: elastix files copy-paste to SlicerElastix-master\Elastix folder

A line was added in the getElasticBinDir function indicating that the elastix files could be found on the same path as the Elastix.py script. It is highlighted in bold in the following piece of the code

```
def getElastixBinDir(self):
  if self.elastixBinDir:
    return self.elastixBinDir

  self.elastixBinDir = self.getCustomElastixBinDir()
  if self.elastixBinDir:
    return self.elastixBinDir

  elastixBinDirCandidates = [
    # install tree
    os.path.join(self.scriptPath, ''),
    os.path.join(self.scriptPath, '..'),
    os.path.join(self.scriptPath, '../../../bin'),
    # build tree
    os.path.join(self.scriptPath, '../../../../bin'),
    os.path.join(self.scriptPath, '../../../../bin/Release'),
    os.path.join(self.scriptPath, '../../../../bin/Debug'),
    os.path.join(self.scriptPath, '../../../../bin/RelWithDebInfo'),
```

```
        os.path.join(self.scriptPath, '../../../../bin/MinSizeRel') ]
```

The PISCOMmodule uses two parameters files as input for the registerVolumes() function: Par0002.fs.MI.rigid.RandomCoordinate.txt and Parameters_Rigid.txt. These files are found inside the Elastix folder, in the following path: SlicerElastix-master\Elastix\Resources \RegistrationParameters. They were modified so that the result image format was 'nii' (instead of the default 'mhd') and the result image pixel type was 'float' (instead of the default 'short'). An example of the modified lines is shown in Figure A2.

```
Parameters_Rigid - Notepad
File Edit Format View Help
// applications a good choice.
(BSplineInterpolationOrder 1)

// Order of B-Spline interpolation used for applying the final
// deformation.
// 3 gives good accuracy; recommended in most cases.
// 1 gives worse accuracy (linear interpolation)
// 0 gives worst accuracy, but is appropriate for binary images
// (masks, segmentations); equivalent to nearest neighbor interpolation.
(FinalBSplineInterpolationOrder 3)

//Default pixel value for pixels that come from outside the picture:
(DefaultPixelValue 0)

// Choose whether to generate the deformed moving image.
// You can save some time by setting this to false, if you are
// only interested in the final (nonrigidly) deformed moving image
// for example.
(WriteResultImage "true")

// The pixel type and format of the resulting deformed moving image
(ResultImagePixelType "float")
(ResultImageFormat "nii")
```

Figure A2: Lines modified of Parameters_Rigid.txt file

The registerVolumes() function found in ElastixLogic() class has an input parameter called outputVolumeNode. If the outputVolumeNode is different to None, then the output volume path is created by joining the temporary directory where the results are created and the "result.mhd" file name and the volume in this path loaded in Slicer. However, since the ResultImageFormat was changed to "nii", the result image name will be "result.nii". Thus, in line 851, this string name was edited.

The following lines to line 851, load the volume in Slicer. However, it is preferred that the resulting image is copied to the working directory of the user (where the MRI, SPECT and PET images are found) and then loaded from that directory, instead of the temporary one that will eventually be deleted. Thus, lines from 850 to 860 where commented.

Additionally, by default, when the Elastix module is initialized, the deleteTemporaryFiles QCheckBox is set to True. However, since in PISCOMmodule.py the temporary path of the result volume is used to copy the image to the user working directory (with shutil.copyfiles python function, see **¡Error! No se encuentra el origen de la referencia.**), line 377 was modified and the deleteTemporaryFiles was set to False. In PISCOMmodule.py, the temporary directory with the registered image created by the Elastix module is deleted to free up computer space (see **¡Error! No se encuentra el origen de la referencia.**).

Finally, at the end of the registerVolumes() function, a line was added (line 885) to return the outputVolumePath. This way, the output volume path can be used in PISCOMmodule.py script.

The following lines contain the previous commented modifications in the code highlighted in bold.

```
373  def __init__(self):
374    ScriptedLoadableModuleLogic.__init__(self)
375    self.logCallback = None
376    self.abortRequested = False
377    self.deleteTemporaryFiles = False
378    self.logStandardOutput = False
379    self.registrationPresets = None
380    self.customElastixBinDirSettingsKey = 'Elastix/CustomElastixPath'

747  def registerVolumes(self, fixedVolumeNode, movingVolumeNode, parameterFilenames = None,
748    outputVolumeNode = None, outputTransformNode = None, fixedVolumeMaskNode = None,
     movingVolumeMaskNode = None, forceDisplacementFieldOutputTransform = False):

849      if outputVolumeNode:
850        #Load volume in Slicer
851        outputVolumePath = os.path.join(resultResampleDir, "result.nii")
852        #[success, loadedOutputVolumeNode] = slicer.util.loadVolume(outputVolumePath,
           #returnNode = True)
853        #if success:
854        #  outputVolumeNode.SetAndObserveImageData(loadedOutputVolumeNode.GetImageData())
855        #  ijkToRas = vtk.vtkMatrix4x4()
856        #  loadedOutputVolumeNode.GetIJKToRASMatrix(ijkToRas)
857        #  outputVolumeNode.SetIJKToRASMatrix(ijkToRas)
858        #  slicer.mrmlScene.RemoveNode(loadedOutputVolumeNode)
859        #else:
860        #  self.addLog("Failed to load output volume from "+outputVolumePath)

878    # Clean up
879    if self.deleteTemporaryFiles:
880      import shutil
881      shutil.rmtree(tempDir)
882
883    self.addLog("Registration is completed")
884
885    return outputVolumePath
```

12.2. Installation manual

The following lines will describe how to install the software required and the PISCOM extension as well as all the packages and dependencies needed to execute the extension.

1. Download and install 3D Slicer
   Link: https://download.slicer.org
   Version used in this project: **4.11.20210226**

2. Install packages and dependencies in 3D Slicer
   a. Open 3D Slicer

b. From Slicer, open the Python Interactor

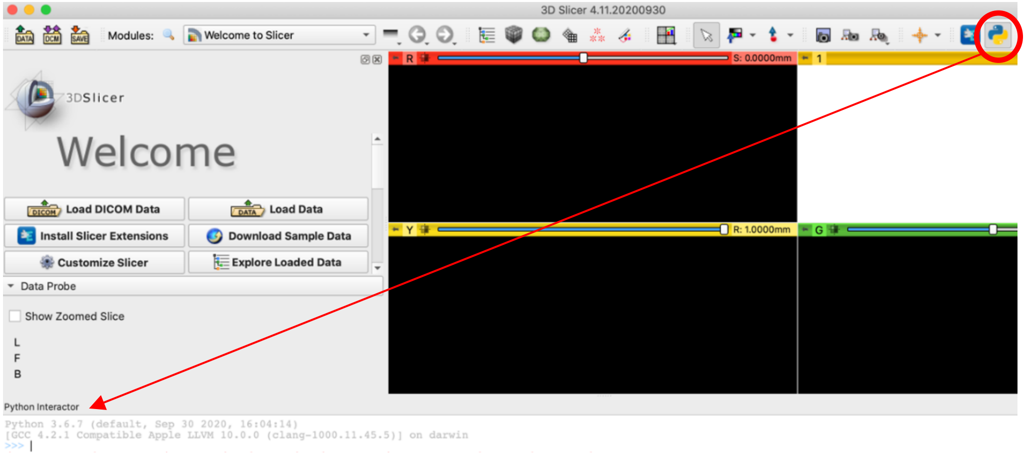


Figure A3: Location of Python interactor

c. Write the following commands in Python interactor:

slicer.util.pip_install('pip --upgrade')

slicer.util.pip_install('scipy')

slicer.util.pip_install('matplotlib')

slicer.util.pip_install('nibabel')

slicer.util.pip_install('nilearn')

slicer.util.pip_install('deepbrain')

d. Deepbrain package (modify the extractor.py script):

i. Open the folder that contains Slicer and go to the following path:
MacOS: ~Contents/lib/Python/lib/python3.6/site-packages/deepbrain/
Windows: ~lib\Python\Lib\site-packages\deepbrain
Default (windows): C:\Users*username*\AppData\Local\NA-MIC\Slicer 4.11.20210226\lib\Python\Lib\site-packages\deepbrain

ii. The deepbrain package uses a function of a deprecated tensorflow version since the package was created in 2018 and has never been updated. To use it without errors, instead of importing tensorflow, a compatible tensorflow with version 1 is required. Open the extractor.py file and substitute the line #1 for the following lines:

```
#import tensorflow as tf
import tensorflow.compat.v1 as tf
tf.disable_v2_behavior()
```

e. Restart 3D Slicer

3. Load the Extensions to Slicer

a. Click the Module searcher button, write Extension Wizard and Switch to the module.

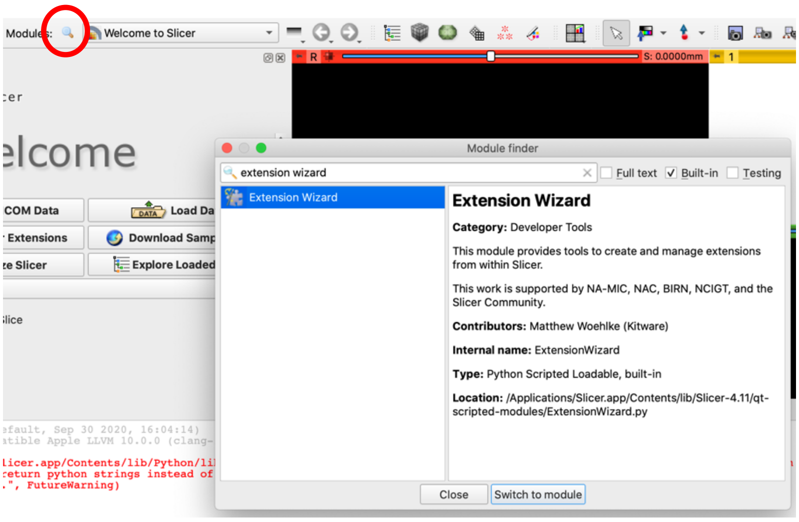


Figure A4: Searching the Extension Wizard module

b. Click the Select Extension button, navigate up to the folder named *Slicer-Elastix master* (which contains the extension) and open it.
c. Say YES to *Would you like to load them now?*
d. Repeat the b) and c) for loading *PISCOM* folder.

4. Restart 3D Slicer

5. Go to **Edit, Application Settings, Modules**. In **Default startup module**, choose Examples→PISCOMmodule

12.3. User Manual

This chapter describes the main functionalities available in Slicer and the PISCOMmodule.

Import images

There are two possible ways for importing images:

1. The fastest: By dragging them into 3D Slicer

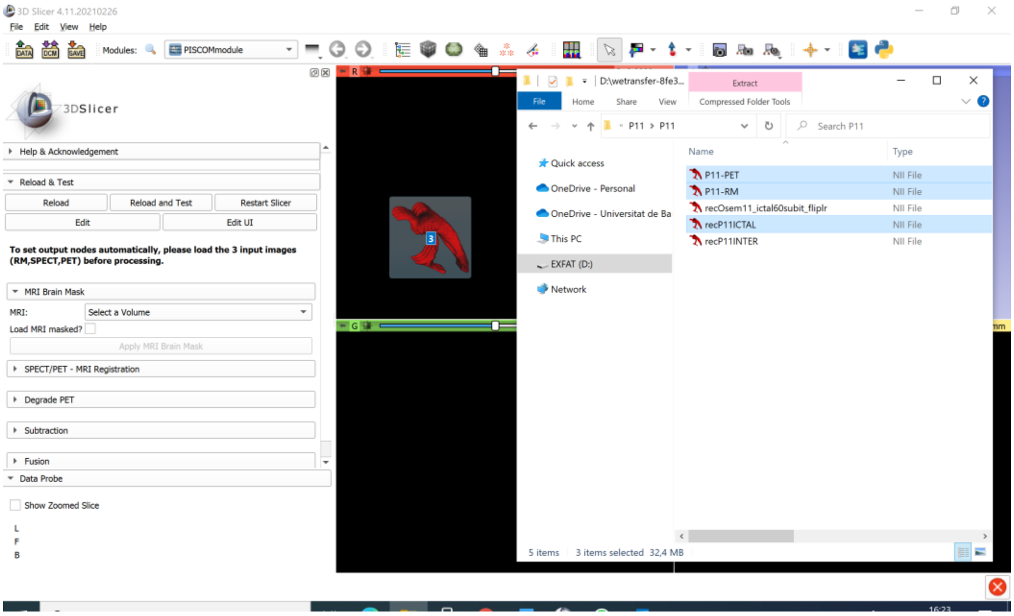


Figure A5: Drag the images into 3D Slicer

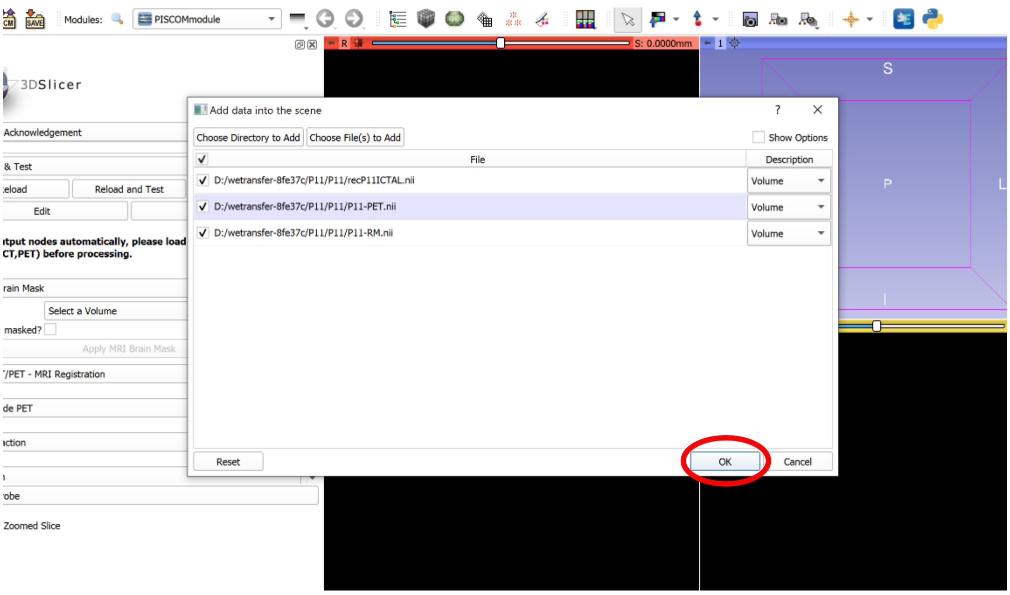


Figure A6: Add data into the scene, OK

2. By using the Add Data Button

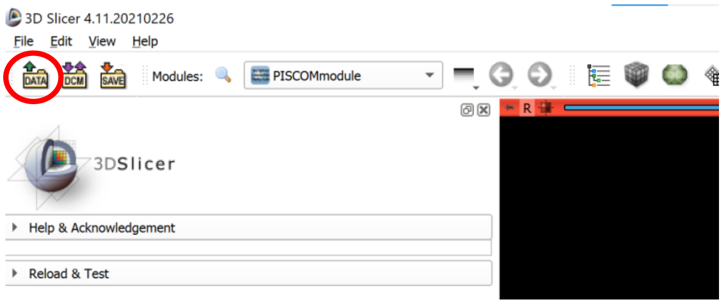


Figure A7: Data Button

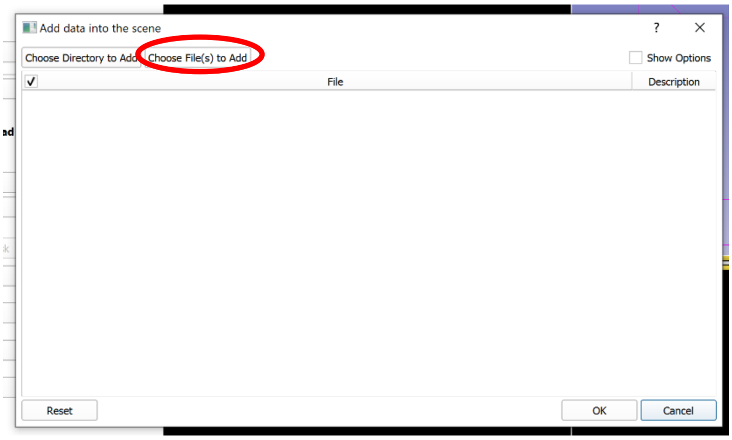


Figure A8: Choose File(s) to Add Button

Go to the files directory and select them, press the Open button and then the OK button (Figure A8).

Recommendation: To have all the images of a patient in a folder named as the patient's identification.

PISCOMmodule

The basic steps for using the PISCOMmodule are described in the following lines.

1. The first step is to select the input volume in the MRI node.

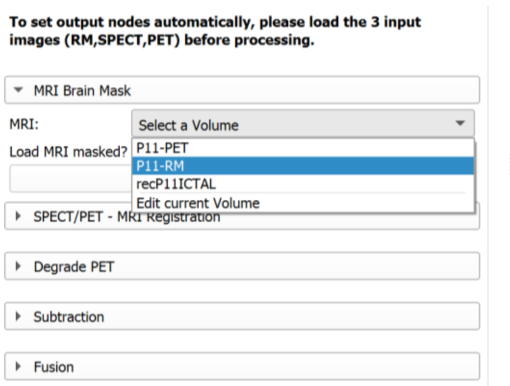


Figure A9: Volume selection in a node

2. The following step is to click on the Apply MRI Brain Mask button.

3. Then select the 2 input volumes in the Ictal SPECT node and the Interictal PET node. Usually, the co-registration algorithm performs best if the fixed volume is the masked MRI, so although it is an optional node, it is recommended to fill it in (done by default).

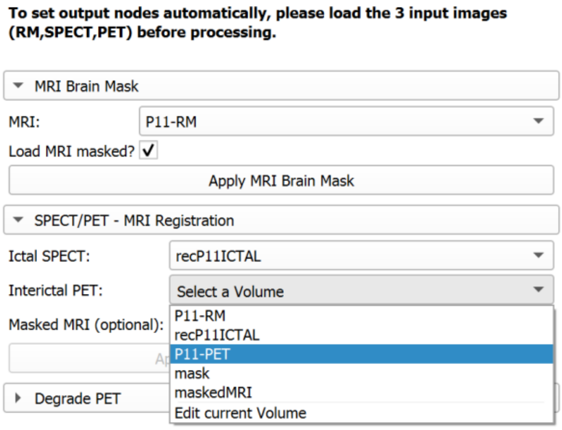


Figure A10: Volume selection in a node 2

4. Click on the Apply SPECT/PET – MRI Coregistration and wait until the output images are loaded.
5. Keep clicking on the Apply buttons until the end.
6. There are some blocks that allow different options in the algorithm.

   The Degrade PET block has a pull-down called Filter settings. By default, it is collapsed. Please, do NOT change this value. The FWHM number of the FDG PET filter was derived from experimental PSF images of the SPECT and PET acquisition systems, thus only if the images came from other acquisition equipment (e.g. from other hospitals), this value should be derived again and changed.

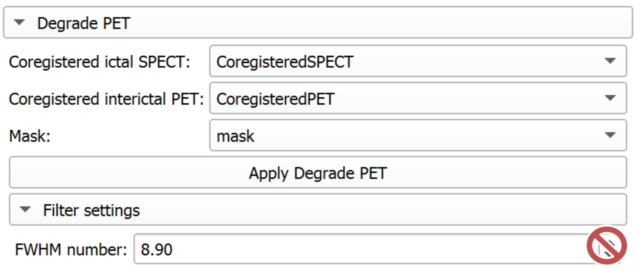


Figure A11: Degrade PET block

In the Subtraction block you can choose if you want to have the relative result of the subtraction (by default) or the absolute result (by checking the CheckBox).

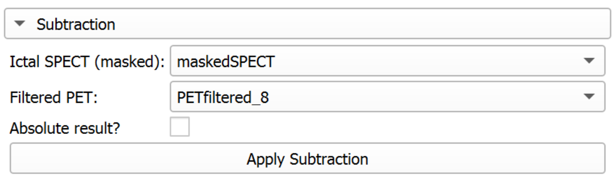


Figure A12: Subtraction block

In the last block, Fusion, you can change the thresholding conditions, the statistical analysis, and the erode iterations to avoid the outliers of the edges.

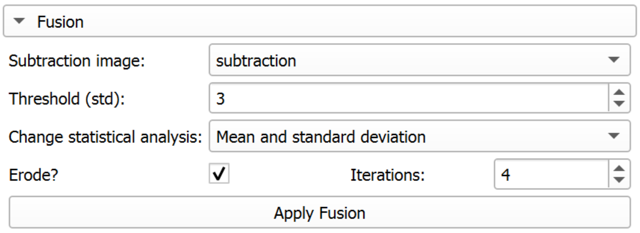


Figure A13: Fusion block. Threshold condition changed.

7. The output image should be similar to the following figure:

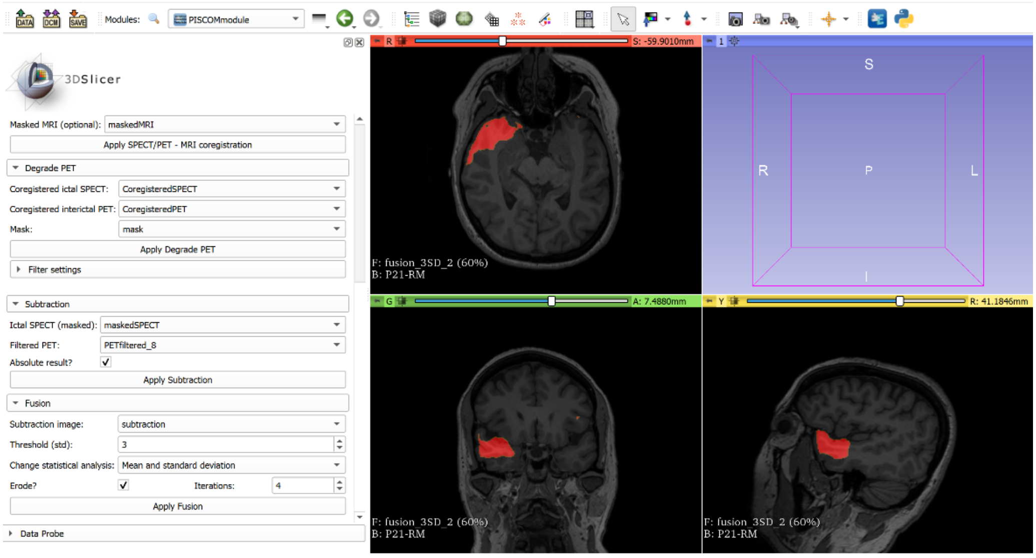


Figure A14: PISCOMmodule final output

View Controllers

There is a module called View Controllers that allows a better experience for visualizing the images. It is found in the pull-down next to the magnifying glass (Figure A15).

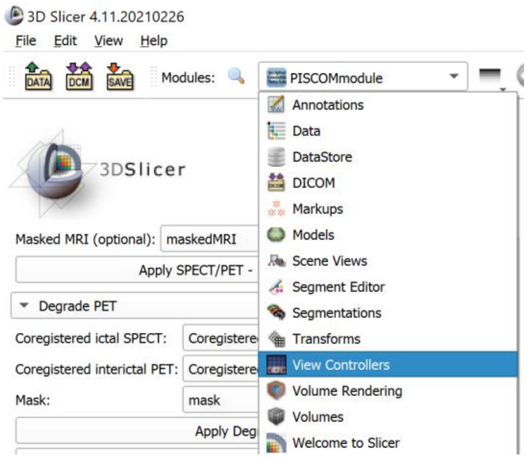


Figure A15: View Controllers location

Among the functionalities it offers there are:

- The eye icon enables the 3D visualization panel.

- You can change the visible slices in background and foreground, and its intensity. This can be useful for checking if the co-registration has been successful.
- The axis icon shows the orientation marker.
- The ruler icon shows the ruler

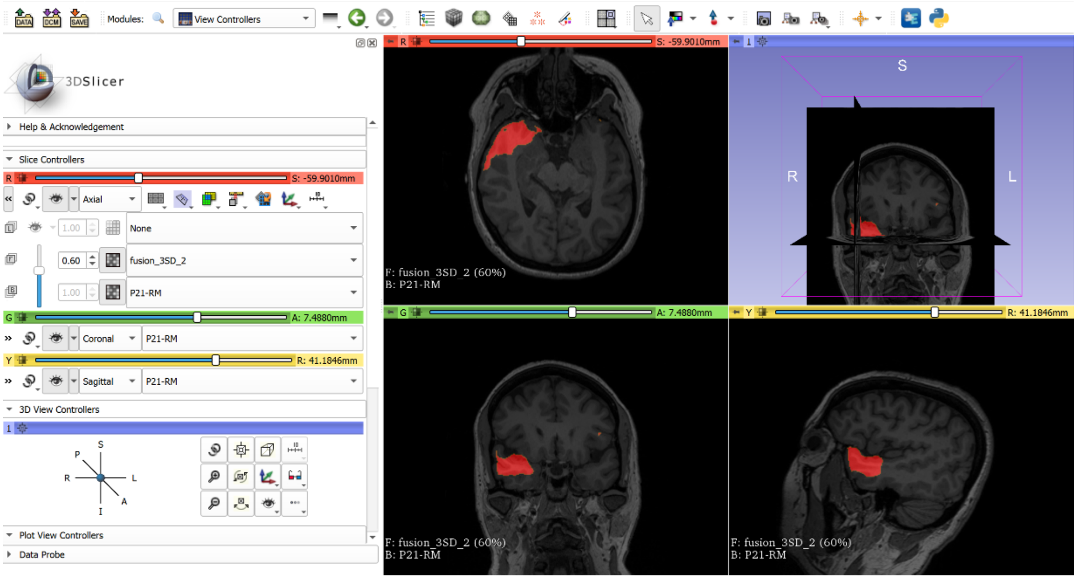


Figure A16: View Controllers panel

*In order to move over one slice of the image with a cursor, press the **MAYUS (or Shift)** button while the mouse is on the desired plane. This can be very useful for moving the slices towards the epileptogenic focus.*

Volumes

The Volumes module allows to see the volume information and to change the colormap of the node. This can be useful for changing the display color of the coregistered outputs.

It is found in the pull-down next to the magnifying glass, like the View controllers module (Figure A15).

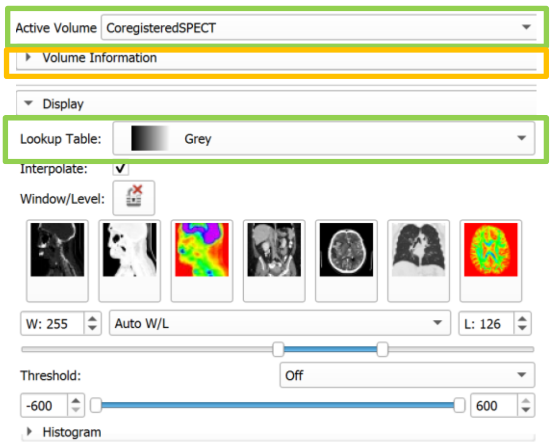


Figure A17: Volumes panel

The first step to change the Display color is to change the active volume to the one that we want to study. After that, in the Display pull-down, change the lookup table to the colormap preferred.

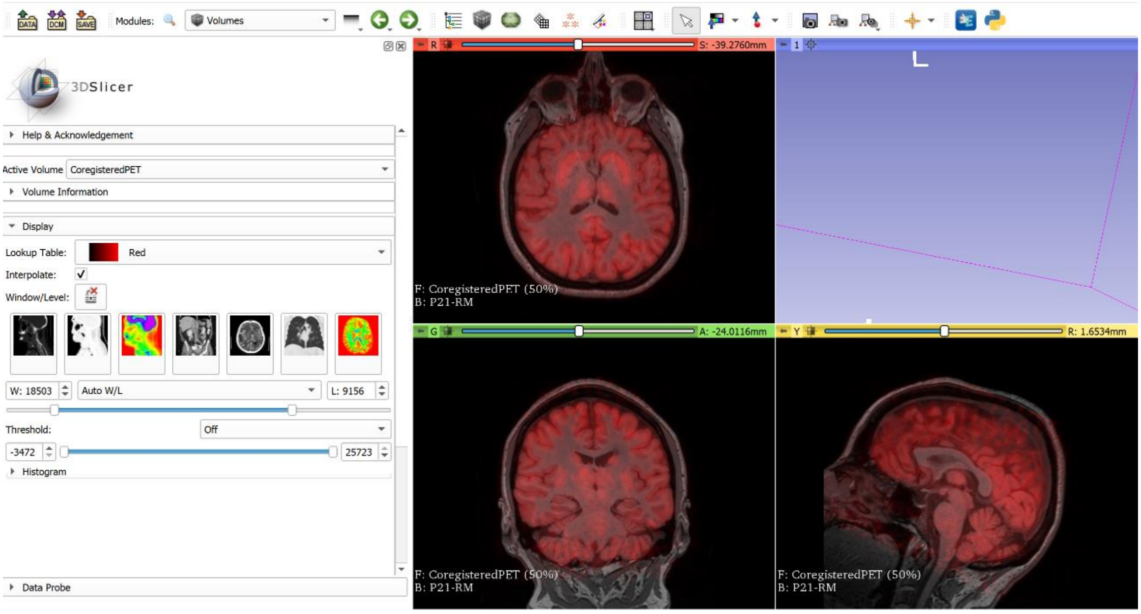


Figure A18: Volumes module example

12.4. Software evaluation

An important point of the main goal is to **assess** the graphical interface, and that means to make a judgment about the final result. The clinicians will be the final users of the extension and thus are the most indicated ones to perform this evaluation.

The following questionnaire/survey has been generated so that clinicians fill it after the first contact with the new software, in the *5.1.User Testing* task of the WBS. The decisive feature that would make the extension to be used in clinical practice is that the graphical interface is ergonomic and easy to use, and therefore the questions are focused on extracting judgements about these features.

The questionnaire was sent with the Google Forms platform; thus, the results were automatically digitalized. The link sent to clinicians was the following: https://forms.gle/cWmzG9qh6ZF88Mmm6.

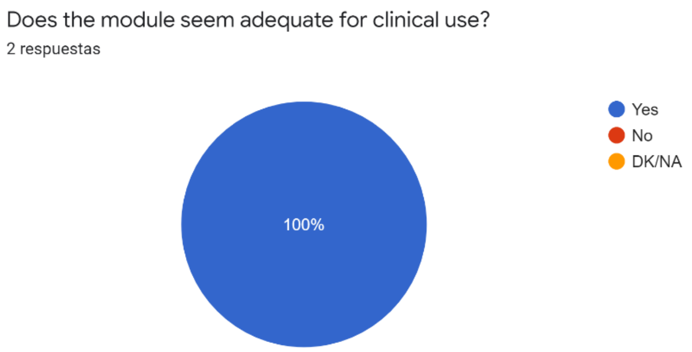
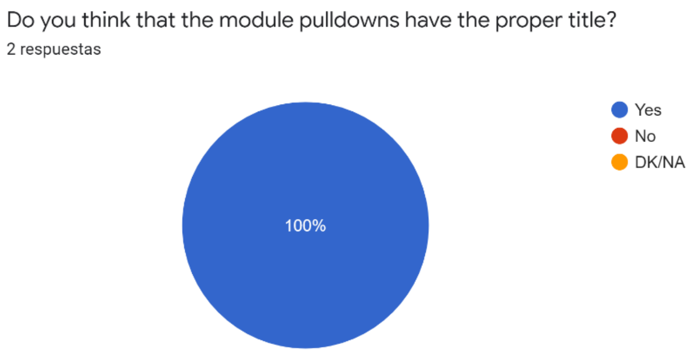
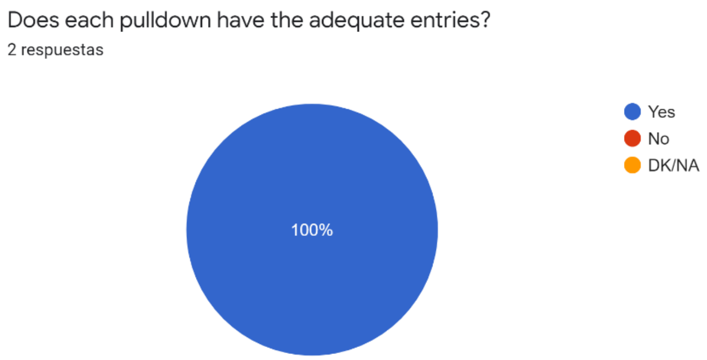
Clinic questionnaire on the PISCOM extension usefulness	**Yes**	**No**	**DK/NA**
Structural quality:			
Does the module seem adequate for clinical use?			
Do you think that the module pulldowns have the proper title?			
Does each pulldown have the adequate entries?			
Do you think any structural change is necessary?			
If yes, please describe it here:___________________________			
Performance:			
Do you think the operating speed is adequate?			
Do you find the manual useful and clear?			
Do you find the module difficult to interpret?			
Do you find the module difficult to manipulate?			
Do you think the output images are adjusted for the diagnosis?			
Do you think the module lacks any functionality or output?			
If yes, please describe it here:___________________________			
Comparison with SISCOM module in FocusDET:			

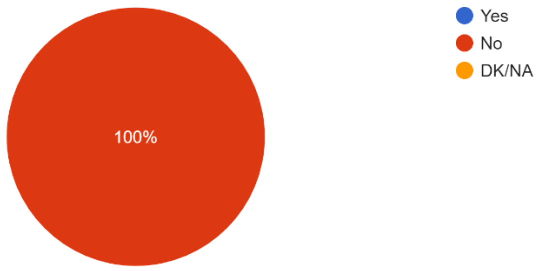
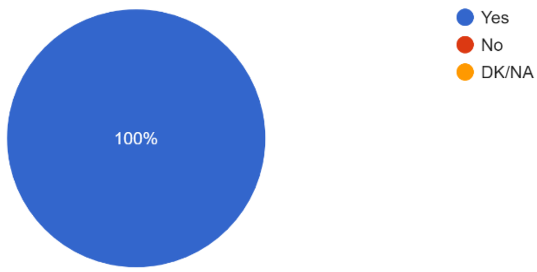
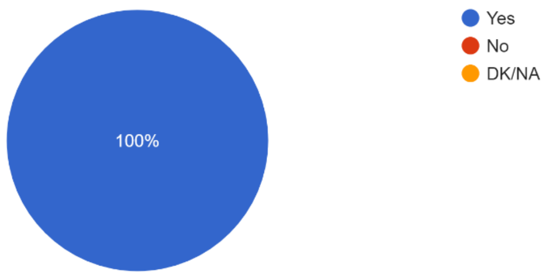
Does it seem to you a more user-friendly tool than the one that performs SISCOM in FocusDET software?			
Does the visualization of the images fulfil your needs?			
It is easy to follow the whole process?			
Usefulness as an element for the localization of the epileptogenic focus:			
In general, do you think it is a useful tool for applying the PISCOM technique?			
Do you think this is a tool that could be used to avoid acquiring interictal SPECT in drug-resistant patients?			
Do you think it is a tool with future potential?			
Do you think it could be used in the Epilepsy Unit of Hospital Clinic de Barcelona?			
Do you think it could be used in other Epilepsy Units?			
Other comments or possible improvements:			

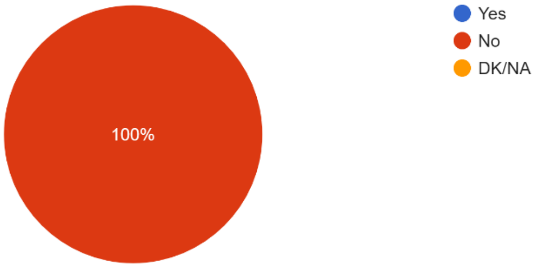
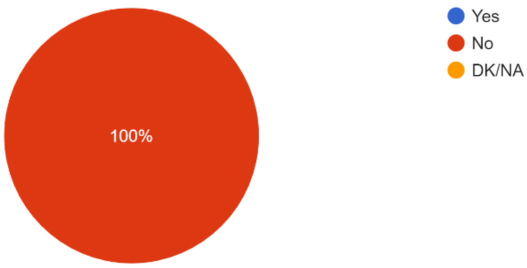
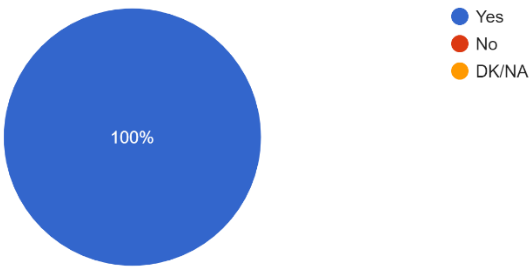
Table A1: Clinic questionnaire on the PISCOM extension usefulness

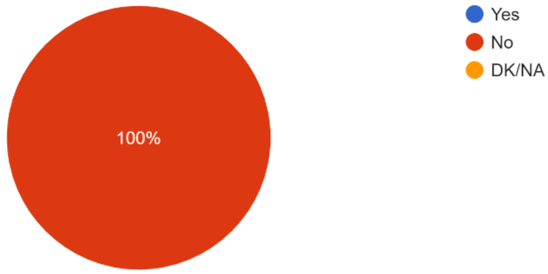
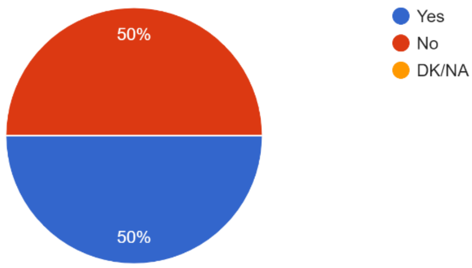
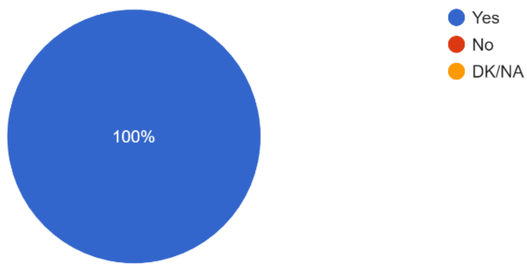
12.4.1. Results of the questionnaire

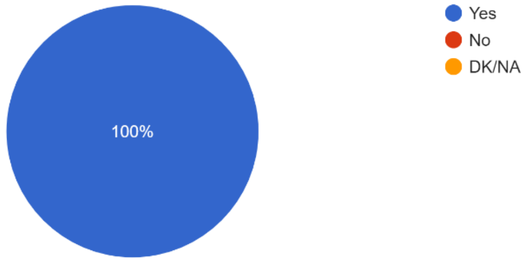
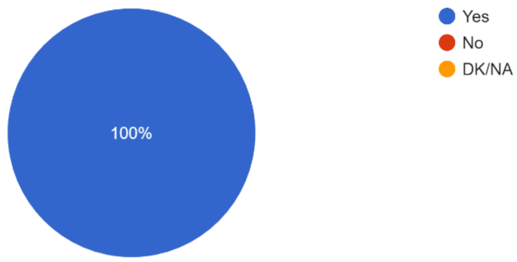
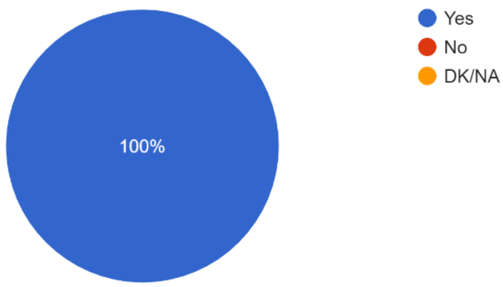
The following Table contains the answers to the questionnaire.

Clinic questionnaire on the PISCOM extension usefulness	**Yes**	**No**	**DK/NA**
Structural quality:			
	2	0	0
	2	0	0
	2	0	0

<table>
<tr><td>Do you think any structural change is necessary?
2 respuestas
</td><td>0</td><td>2</td><td>0</td></tr>
<tr><td colspan="4">If yes, please describe it here: No answers</td></tr>
<tr><td colspan="4">Performance:</td></tr>
<tr><td>Do you think the operating speed is adequate?
2 respuestas
</td><td>2</td><td>0</td><td>0</td></tr>
<tr><td>Do you find the manual useful and clear?
2 respuestas
</td><td>2</td><td>0</td><td>0</td></tr>
</table>

Do you find the module difficult to interpret? 2 respuestas 	0	2	0
Do you find the module difficult to manipulate? 2 respuestas 	0	2	0
Do you think the output images are adjusted for the diagnosis? 2 respuestas 	2	0	0

<table>
<tr><td>Do you think the module lacks any functionality or output?
2 respuestas

</td><td>0</td><td>2</td><td>0</td></tr>
<tr><td colspan="4">If yes, please describe it here: No answers</td></tr>
<tr><td colspan="4">Comparison with SISCOM module in FocusDET:</td></tr>
<tr><td>Does it seem to you a more user-friendly tool than the one that performs SISCOM in FocusDET software?
2 respuestas

</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>Does the visualization of the images fulfil your needs?
2 respuestas

</td><td>2</td><td>0</td><td>0</td></tr>
</table>

It is easy to follow the whole process? 2 respuestas 	2	0	0
Usefulness as an element for the localization of the epileptogenic focus:			
In general, do you think it is a useful tool for applying the PISCOM technique? 2 respuestas 	2	0	0
Do you think this is a tool that could be used to avoid acquiring interictal SPECT in drug-resistant patients? 2 respuestas 	2	0	0

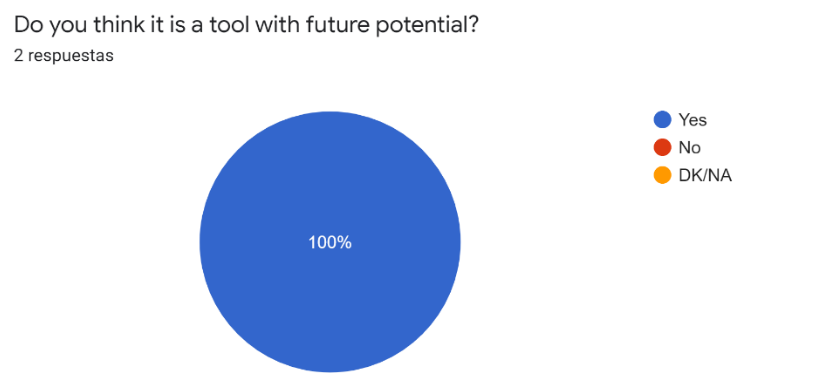
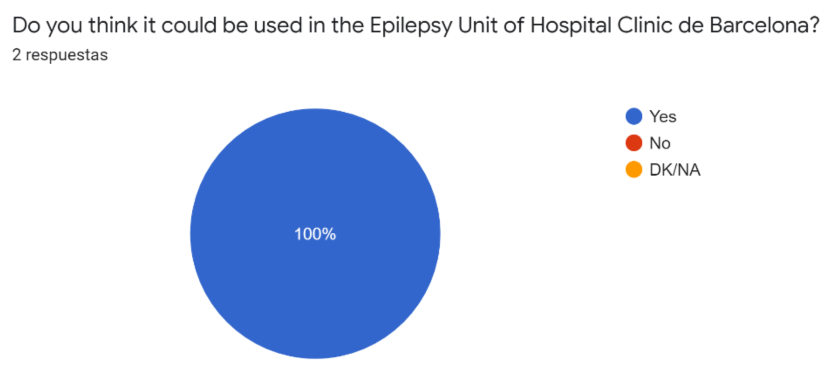
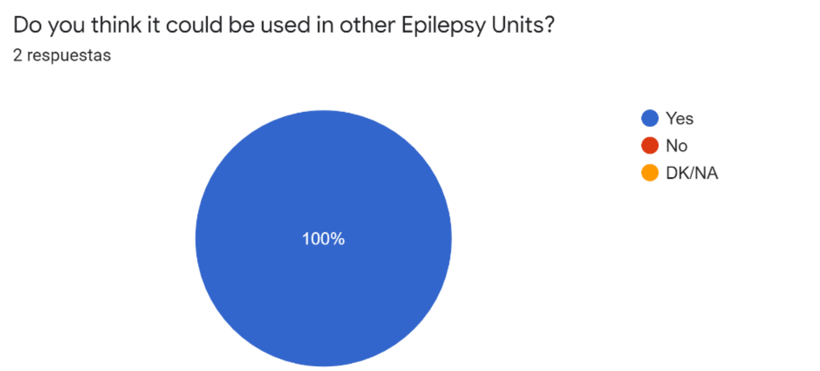
<table>
<tr><td>
</td><td>2</td><td>0</td><td>0</td></tr>
<tr><td>
</td><td>2</td><td>0</td><td>0</td></tr>
<tr><td>
</td><td>2</td><td>0</td><td>0</td></tr>
<tr><td colspan="4">Other comments or possible improvements:</td></tr>
<tr><td colspan="4">Answer 1: It would be really useful to be able to display Ictal, Interictal and PISCOM images simultaneously to perform a visual examination of the 3 modalities fused with the RM all at once. Also, Sokoloff, Specrtum and Warm Metal (GE) color scales could be useful.</td></tr>
</table>

Table A2: Results of the clinic questionnaire on the PISCOM extension usefulness