Conformal energy modulator design optimization using deep learning for FLASH proton therapy treatment planning

Auteur : Jost-Jongen, Elisa
Promoteur(s) : Phillips, Christophe
Faculté : Faculté des Sciences appliquées
Diplôme : Master en ingénieur civil biomédical, à finalité spécialisée
Année académique : 2022-2023
URI/URL : http://hdl.handle.net/2268.2/17725

Conformal energy modulator design optimization using Deep Learning for Flash proton therapy treatment planning

Jost-Jongen Elisa

Master thesis to obtain the degree of Master in Biomedical Engineering

~

Supervisors :

Pr. Phillips
Dr. Labarbe at IBA

Faculty of Applied Sciences University of Liège
Academic Year 2022-2023

Abstract

Radiotherapy is a method used to treat cancers based on the utilization of photons to target and destroy cancer cells. One of the radiotherapy techniques called proton therapy uses the energy released by protons to kill cancer cells. Compared to conventional radiotherapy that uses photons, proton therapy is known to spare more healthy tissues surrounding the tumor. This advantage stems from the particular dose (amount of energy absorbed per kg) profile of protons in matter called the Bragg peak. When proton therapy is delivered at very high dose rates (>40 Gy/s), this technique is called FLASH proton therapy. It has the advantage of sparing even more healthy tissues.

The extra short delivery duration of FLASH proton therapy makes proton administration technically challenging and some delivery methods must be adapted. Importantly, the technique to control the depth dose distribution within the tumor must be entirely rethought. To address this, a new device must be added to the trajectory of the proton beams. This new piece named a Conformal Energy Modulator, is made of a square base on top of which spikes of different heights rise. Its configuration must be tuned according to the tumor shape. The CEM optimization can be achieved by treatment planning systems but current solutions either lack accuracy or take too long to compute.

In this master's thesis, a new method to optimize the CEM quickly and accurately is proposed, relying on an artificial intelligence model called Convolutional Neural Network.

This thesis is divided into 5 main parts. The first part introduces how proton beams are generated and delivered to the patients, how protons interact with matter, and why their dose profile is so interesting from a treatment point of view. Additionally, the challenges posed by the FLASH concept and the required modifications for proton therapy administration to accommodate FLASH PT are discussed. The second part is related to the data acquisition needed for the AI model training. It is performed with a treatment planning system that simulates FLASH PT with the CEM and the resulting dose distribution. The CEMs profiles will serve as outputs and their corresponding dose maps will serve as inputs of the AI model. The third part introduces the concepts of artificial intelligence, neural networks, and convolutional neural networks. The choice of the specific AI model to solve this problem is detailed as well as its architecture. The fourth part discusses the training of the model. The model's ability to predict a CEM given a dose map is evaluated using validation and test sets. Lastly, the performance of the predicted CEMs by the model compared to the initial CEMs of the data set is evaluated through a comparison of their dose maps resulting from FLASH PT simulations. The dose maps comparison is performed with dose-volume histograms and Gamma index evaluation. This comparison highlights the significant similarities observed in the dose maps, indicating the model's excellent training and successful CEM optimization.

Acknowledgements

I would like to thank IBA, who open its doors to me during my internship. I would like to express my gratitude to Dr. Rudi Labarbe who always supported me, followed my progress with carefulness, and made FLASH proton therapy so interesting. Mr. Geoffroy Herbin advised me every week, giving me new ideas to test out and develop without ever limiting my freedom in my work. All the research team welcomed me warmly and made me feel like a real colleague. They made my four months internship captivating.

I would also like to thank my academic supervisor Professor Christophe Phillips. He gave me an additional point of view on my work allowing interesting approaches to test out.

A special thanks to Ph.D. Vincent Boudard who always advised me of his own free will when he had time. His knowledge of deep learning helped me go forward with my work.

Finally, I would like to thank my family who always supported me, and my friends who made my university years unforgettable.

Contents

Chapter 1

Introduction

Proton therapy is a cancer treatment method that has the advantage of sparing more healthy tissues thanks to its specific dose deposition in Bragg peak compared to conventional radiotherapy. The dose refers to the amount of radiation energy absorbed by matter in Gray (= J/kg). The FLASH concept is the way of delivering the dose at high dose rates and that has the yet unexplained advantage of sparing even more healthy tissues while maintaining equal efficacy in tumor growth control. The sparing of more healthy tissues also called the FLASH effect, is still not yet clearly explained but some hypotheses are being considered [14]. Among them, the preservation of stem cell niche, metabolic quiescence, and the reduction of lipid peroxidation and reactive oxygen species are proposed. Despite not fully understanding this phenomenon, the FLASH concept is worth exploring. This FLASH method applied to proton therapy has already been tested on animals and a few humans in the context of clinical studies.

The dose distribution in depth is used in classical proton therapy to obtain a uniform dose in the tumor. Some conventional methods used to achieve this are Pencil Beam Scanning, ridge filter, and range modulator wheel. However, those can not be adapted to FLASH proton therapy due to mechanical limits, too long time duration, or lack of precision. As a replacement for those, a new piece that will be responsible for dose distribution in depth must be added to the trajectory of the proton beams in FLASH. This additional piece, called a Conformal Energy Modulator (CEM), must be designed as a function of the tumor shape and the scattering phenomenon. Scattering is one of the interactions that occur between protons and matter. The CEM structure is made of a square base on top of which spikes of different heights rise. The cross-section and height of the spikes are responsible for the change in weight and range of the proton beams respectively. As a result, the dose becomes distributed in depth in the tumor.

The CEM can be optimized for a given dose distribution objective in a tumor with several treatment planning systems. The Raystation system is a commercial product that is not meant to be used for research purposes as some functionalities are not accessible. MIROpt is used by researchers and the shape of the CEM is optimized using an analytical model relying on the Moliere theory. Its optimization is fast but scattering is not described sufficiently accurately. FlashOpenTPS uses several Monte Carlo simulations performed by MC^2 to perform CEM optimization. Monte Carlo describes well scattering but the computation time is long.

The goal of this master's thesis is to optimize the Conformal Energy Modulator in another way than what already exists while decreasing the computation time and keeping a good accuracy. The use of an AI model is an interesting way to optimize the CEM.

Artificial intelligence (AI) is a domain that is more and more used to solve all sorts of problems by teaching machines how to perform complex tasks. The model takes as input the dose maps and gives the corresponding CEMs as output. A dose map represents the absorbed dose distribution in the tumor in 3D. A CEM is in the form of a 2D elevation map describing the elevation of its spikes. Those CEMs and dose maps can be acquired by FLASHOpenTPS. As the data the model has to handle is complex and multidimensional, the model's complexity must match it. Neural Networks (NN) in the deep learning branch of AI are known to have the capacity to learn complex relationships between the inputs and the outputs. A specific type of NN that has proven to be effective for voluminous and multidimensional data sets with a known grid-like topology is the Convolutional Neural Network. This type of model seems the perfect one for understanding the spatial relationship between the dose maps and the CEMs and thus for the CEM optimization problem.

Chapter 2

Radiotherapy techniques

To understand what FLASH proton therapy is, conventional radiotherapy, proton therapy, and the FLASH concept applied to them are discussed in this section. First, conventional radiotherapy is briefly explained. Then the application of the FLASH concept applied to conventional radiotherapy is described to understand what is FLASH. Afterward, proton therapy is developed more deeply. Lately, FLASH proton therapy and the difference in dose administration compared to proton therapy are explained.

2.1 Conventional radiotherapy

Conventional radiation therapy, also called radiotherapy, is one of the techniques that treat cancer. Radiotherapy uses high doses of ionizing radiation to kill cancer cells and shrink tumors by damaging cells' DNA. The energy released by photons (or sometimes electrons) is the cause of DNA damage. The dose profile of photons is described by energy deposition starting from the surface of the skin, the dose increases down to a couple of centimeters into the body reaching a maximum then decreases exponentially as displayed in Figure 4. Secondary electrons are responsible for depositing dose forward, which causes the dose to build up from the skin surface. In radiotherapy, X-rays (i.e. photons) pass through the whole body. The energy deposited by X-rays in the body is significant and damages the healthy tissues surrounding the tumor. This increases the risk of developing secondary cancers and causes damage to healthy tissue leading to complications. To limit damage to surrounding tissues, radiation beams are aimed from several angles of exposure to intersect at the tumor. The conventional dose rate delivery is 2 Gy/min in a clinical system.

2.2 FLASH radiotherapy

FLASH radiotherapy consists in delivering electrons at ultra-high dose rates, usually higher than 40 Gy/s, in the tumor [8]. It is very complicated to achieve a high dose rate with photons as their production is not very efficient leading to low intensity. The photon intensity is thus too low to get FLASH dose rates. It is the reason why electrons are usually used for FLASH radiotherapy. The dose profile of electrons as a function of the depth in matter is similar to the photons one except that the dose deposited by electrons decreases faster.

The dose delivery lasts a few seconds instead of minutes for conventional radiotherapy, hence the name "FLASH". The short delivery duration allows better precision of the treatment as the motion of the patient can be mitigated for a few seconds. Having better treatment precision leads to less dose in the surrounding tissues and more dose in the target volume. This delivery method shows the advantage of sparing more surrounding tissues and reducing side effects while preserving a high dose in the target volume [8] [16]. FLASH radiotherapy with electrons is still in the research phase. This method has been tested on animals and a few humans in the context of clinical studies. However, this technique is not yet a standardized method used to treat patients.

The sparing of healthy tissues also called the FLASH effect, is still not yet well understood. Several hypotheses are considered to explain this effect [24] [9] [14]. Among them, some are already classified as implausible such as adaptive immunity, oxygen depletion, partial irradiation of blood volume, DNA damage and repair, and a few others. Some plausible mechanisms that could explain the FLASH effect are :

- Preservation of stem cell niche: the extra short irradiation time reduces the impact on stem cell niche which plays a role in tissue repair and regeneration
- Lipid peroxidation and Fenton chemistry: due to the short irradiation period, the production of free and hydroxyl radicals respectively reduces the formation of lipid peroxidation and reactive oxygen species which damage healthy tissues.
- Metabolic hibernation: A transient state of metabolic quiescence in healthy tissues is induced due to the stress produced by the ultra-high dose rate of radiation. This state would reduce the effect of radiation on healthy tissues.

To conclude, the mechanisms of action of the FLASH method are still not clear as of now. However, this domain is worth researching as the FLASH method is a promising way to improve patient treatment.

2.3 Proton therapy

Proton therapy is a specific type of radiotherapy that uses protons instead of photons (or electrons) to damage cells' DNA. Protons are delivered in the form of beams into the body to reach the tumor and irradiate it. Proton therapy spares more healthy tissue than conventional radiotherapy because smaller doses are delivered to tissues upstream of the tumor, and no radiation is deposited downstream. As a result, proton therapy is a less invasive treatment than conventional radiotherapy. Those advantages make proton therapy recommended especially for children who are more sensitive to developing secondary malignancies than adults. The section "Effects of proton therapy"" explains well those advantages and where they come from.

2.3.1 Proton beams generation and administration

Protons used for therapy are typically produced by separating hydrogen atoms into electrons and protons. The protons are accelerated into a cyclotron to gain energy. All

protons that go out of the cyclotron have a maximum energy of around 230 MeV. Protons are then directed into the treatment room with the beam transport system that ensures the right energy and trajectory. Protons energy is tuned while passing through a piece of carbon with variable size called a degrader. Scattered protons are filtered with a brass aperture to keep only protons with the right trajectory. An energy selection system is used to select the protons with the desired energy. Energy loss and scattering will be discussed in the next section. All those components are part of the beam transport system. In the treatment room, the protons are directed to the patient's body through the gantry's arm and nozzle. The gantry is the machine that revolves 360 degrees around the patient allowing to obtain the wanted angle of approach for proton administration. A representation of the cyclotron, the beam transport system, and the gantry is shown in Figure 1.

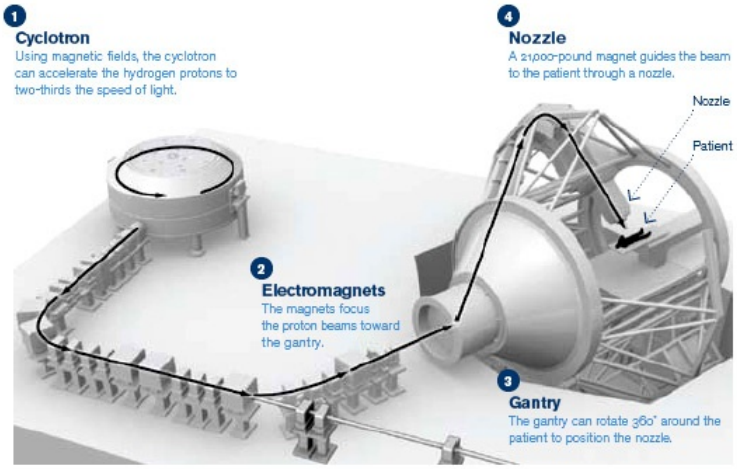

Figure 1: Proton generation and control system (Source: http://www.israelprotontherapy.com)

Quadrupole magnets are used to focus the beam and therefore increase the beam fluence (proton / cm2). Fluence is increased to have a condensed proton beam which is needed for the proton administration technique called pencil beam scanning (PBS). During this process, the proton beams are steered by dipole magnets through the target volume layer by layer and spot by spot to perfectly cover the whole tumor as seen in Figure 2.

The lateral scanning is done by the scanning magnets, which are dipole magnets. The energy of proton beams is modulated by changing the thickness of the degrader according to the specific zone to reach. It is the switching of energy for each layer that mostly takes time. This technique allows a high precision and uniform dose distribution even for complex tumors. Proton beam scanning is an active scanning technique.

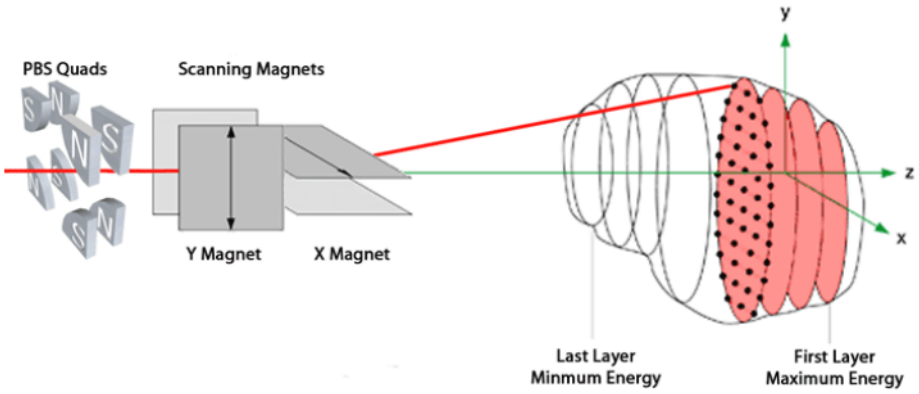

Figure 2: Pencil beam scanning (Source: A. Patriarca - Characterization and implementation of PBS proton therapy techniques

Another proton administration technique which is a passive scattering technique consists of interposing mechanical components in the beam line. High-Z materials called scatterers are added in the proton trajectory for lateral spreading as scattering is induced in those materials. Then spreading in depth is achieved either with a range modulator wheel or by using ridge filters. A range modulator wheel is a rotating wheel with steps of variable thicknesses that turns at a certain frequency. When proton beams pass through it, they will face a piece of matter whose thickness changes with time and allow scanning of the tumor in depth. This wheel is made of low-Z material to reduce scattering as much as possible. A ridge filter consists of a piece of low-density material that has a ridge-like shape. Its thickness is gradually increasing along its length to modulate the range of the proton beams. What is obtained with those administration techniques is a Spread-out Bragg peak (SOBP) in the target volume. The range of a proton beam and the SOBP will be defined in the next section.

2.3.2 Proton interaction with matter

Each time protons come into contact with matter, four types of interactions can occur.

- a) Inelastic Coulombic interaction occurs when a proton passes near an atomic electron and loses its kinetic energy.
- b) Repulsive elastic Coulombic interaction occurs when a proton passes near an atomic nucleus and is deflected from its original straight-line trajectory by the repulsive force from the positive charge of the nucleus. This phenomenon is called Scattering.
- c) Non-elastic nuclear interaction occurs when a proton passes near an atomic nucleus and enters it leading to an emission of secondary particles (electron, neutron, proton, gamma rays,..). This type of interaction is less frequent.
- d) Bremsstrahlung interaction occurs but is negligible for therapeutic proton therapy given the energy range used.

The three first types of interaction are represented in Figure 3.

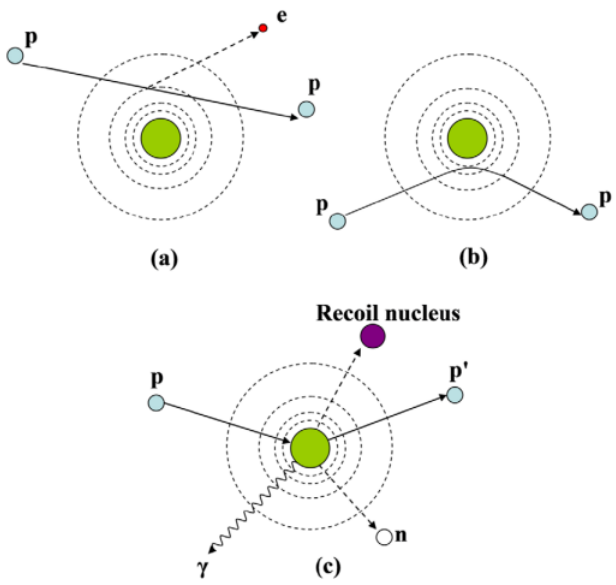

Figure 3: Proton interactions with matter: Inelastic Coulombic interaction(a), Repulsive elastic Coulombic interaction(b), Non-elastic nuclear interaction(c) (Source: W.Dehauser, Physics of proton therapy, R158)

Proton therapy is mainly based on the effect of inelastic Coulombic interactions. The more proton beams interact with matter, the more they release energy until reaching the Bragg peak. The energy loss rate of proton beams depends on many parameters including the velocity, ion charge, and material density. Scattering and nuclear interactions also occur and have to be taken into account as they affect the treatment.

Scattering influences the spatial distribution of dose in a patient due to multiple individual Coulomb scattering events that deflect protons' trajectory by a certain angle for each event. The most rigorous theoretical calculation of multiple Coulomb scattering is the Moliere theory but is very complex [5].

2.3.3 Effects of proton therapy

The dose profile of protons is characterized by an increase of released energy while passing through matter until reaching the maximum, called Bragg peak, and then decreases to zero as protons stop. Proton therapy aims to have the dose peak of protons happening in the tumor and protons trajectory stopping in the tumor [17].

In Figure 4, the relative dose as a function of the depth in a medium is displayed for photons in black and protons in red and blue. The dose is relative to a specific reference dose in order to be normalized. It permits to make comparisons easily between several graphs of the same type. The blue curve represents the Pristine Bragg curve which occurs

when a monoenergetic proton beam irradiates a medium. The peak of the blue curve is the Bragg peak.

Fifty percent of protons stop somewhere between the entrance of the body and a specific depth equal to the range. This depth is located beyond the maximum of the Bragg peak: where the relative dose is 80% of the maximum Bragg peak. The range value depends on the initial energy of the proton beam. From the entrance into the matter to around the end of the range, the gradual proton depletion is caused by protons entering nuclei during nuclear interactions. The other 50% of the protons that are still in movement at the range will all stop in a narrow region in the final drop of the Bragg peak. This region is the region where the dose goes from 80% of the maximum Bragg peak to 0%. Around the end of the range, the steep decrease in the number of protons is due to protons running out of energy, slowing down until stopping and being absorbed by the medium. Secondary particles emitted during nuclear interactions represent a small part of the therapeutic absorbed dose.

The red curve represents the Spread out Bragg curve which occurs when a proton beam that has been modified to increase the axial dimension of the peak region irradiates a medium. The proton beam is modified in a poly-energetic beam either by combining multiple mono-energetic beams of different energies or with a continuously modulated beam. It can be seen that the SOBP covers better the tumor represented in grey than the Bragg peak.

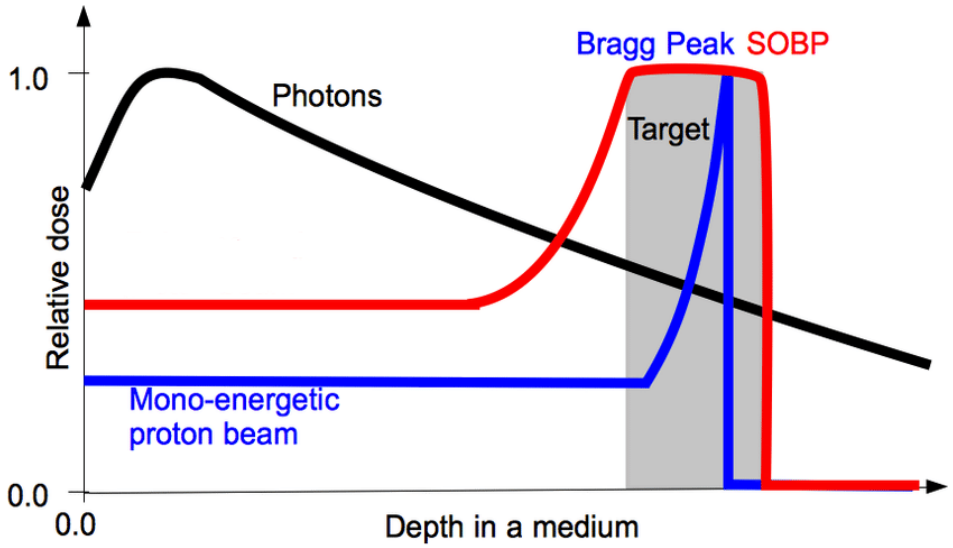

Figure 4: Relative dose as a function of depth in a medium

The SOBP is characterized by a constant dose level over a depth range in a medium. The goal is to combine several Bragg peaks (BP) with different ranges (i.e. depth in the medium) and weights to obtain the wanted shape and width of the SOBP to match the target volume. The range of Bragg peaks can be adjusted by changing the energy of the protons, while the weight of Bragg peaks can be adjusted by changing the number of protons used. The number of protons in a BP can be modified by changing the beam

intensity (proton/s) and keeping the time delivery constant or by changing the delivery time and keeping the intensity constant. The beam intensity is controlled by the proton current coming out of the source located at the center of the cyclotron. Those adjustments are performed in proton beam scanning (PBS) by sending sequentially the multiple BP or in double scattering by the use of a rotating range modulator wheel or a static ridge filter.

The patient irradiation time of proton therapy last about 2 minutes. Proton therapy is typically used to treat non-spreading cancers located near critical organs such as the eyes or brain. The main disadvantage of proton therapy is its huge cost mainly due to the production of protons which is very complex. Currently, it is still unclear which types of cancer are best treated with proton therapy.

2.4 FLASH proton therapy

FLASH proton therapy (FLASH PT) consists in delivering ultra-high dose rates in a short amount of time using protons. It uses the same FLASH concept as the one used on conventional radiotherapy (which used electrons) and the same FLASH effect is observed.

FLASH proton therapy follows the same physics as proton therapy and is generated the same way. The concepts explained in the previous section are thus also true for FLASH proton therapy except for the additional FLASH effect and for a few differences in the administration process.

The method to administrate protons in depth is different for FLASH proton therapy compared to proton therapy that uses PBS. In FLASH PT, there is no degrader that tunes the proton beam's energy. All protons have the same energy of about 230MeV which is to energy at the exit of the cyclotron. The proton beams are sent by the beam transport system (without the degrader) into the gantry, come out of the treatment head called the nozzle, and enter the snout. Like in PBS proton therapy, proton beams are scanned laterally in both directions with two dipole magnets. The scanning allows the distribution of energy in the lateral plane of the tumor. As proton beams all have the same energy, irradiation is not performed layer by layer like for PBS. A new method must be developed to cover the whole tumor homogeneously in depth in a single shot of 3 seconds.

The new method is the creation of a piece called the Conformal Energy Modulator (CEM) located in the snout that is added to the trajectory of protons. Without this piece, all protons' Bragg peaks would occur in a single plane as they have the same energy. Its use is to move back some Bragg peaks by different distances so that the release of energy happens all over the tumor in one shot. This additional layer of matter is composed of a squared base on top of which spikes of different heights rise and resembles the shape of a hedgehog. Due to the different heights of the spikes, Bragg peaks will be distributed in depth over the tumor. The structure of the CEM should be complex and personalized according to the tumor shape of the patient. An example of CEM is displayed in Figure 5.

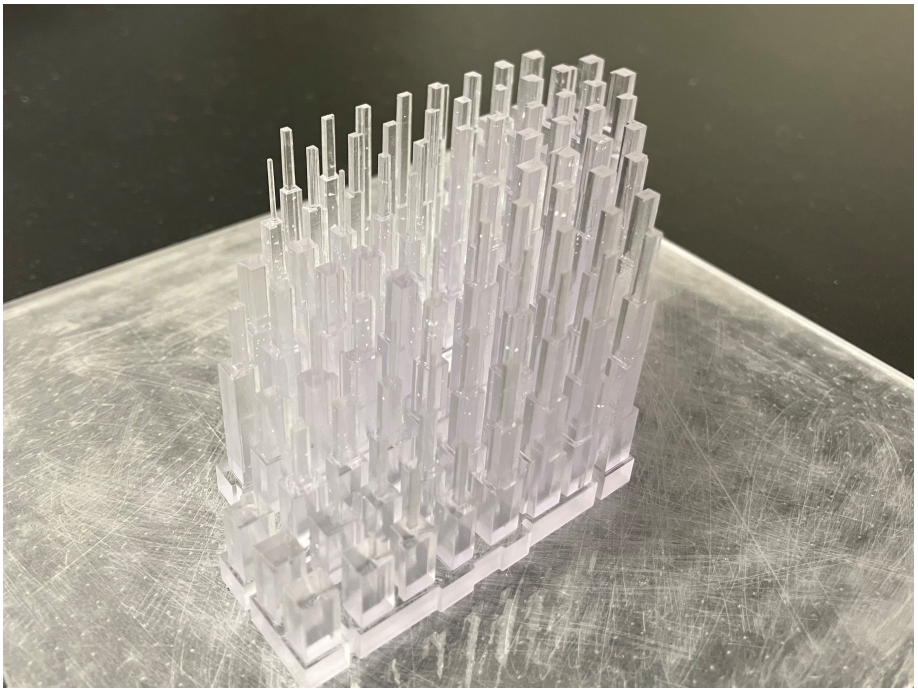

Figure 5: Example of CEM designed with Raystation and 3D printed

Compared to passive scattering in proton therapy, the CEM represents a ridge filter that has been designed asymmetrically. Indeed, a basic ridge filter is symmetric meaning that the same SOBP would be obtained for every patient and would have the form of a parallelepiped. Round shapes would not be achieved with this kind of filter. A range modulator wheel is not doable in FLASH proton therapy due to the extra short dose delivery period that is too constraining. Indeed, the wheel has a mechanical limit for its speed and size that makes the depth covering impossible in 3 seconds. Active scattering with PBS could not be applied either to FLASH as it takes a lot of time to switch the energy for each layer. Those are the reason why a new piece called CEM was necessary to ensure a uniform dose distribution in the tumor. A high-Z material-made CEM would have a high density and therefore would lead to smaller peaks than a low-Z material. Thus lateral scanning is performed with dipole magnets and the CEM is made of low-Z material to reduce scattering.

After the CEM, proton beams will go through a range shifter located in the snout. The purpose of the range shifter is to shift the curve of dose as a function of the depth in tissue to obtain Bragg peaks precisely inside the tumor. The shifting is performed to obtain shallower Bragg peaks. Without this, proton beams would go too deep and Bragg peaks would occur outside of the body or even worse, inside of healthy tissues. This range shifter is rectangular and all proton beams pass through it, the shifting is thus globally the same for every proton beam. After the snout, proton beams are filtered by a brass collimator to keep only protons with the right trajectory. Afterward, proton beams enter the patient's body [18]. An example of FLASH proton therapy assembly is displayed in Figure 6. The CEM is sometimes called "hedgehog" due to its similarity to the animal of the same name. It is the case in this Figure.

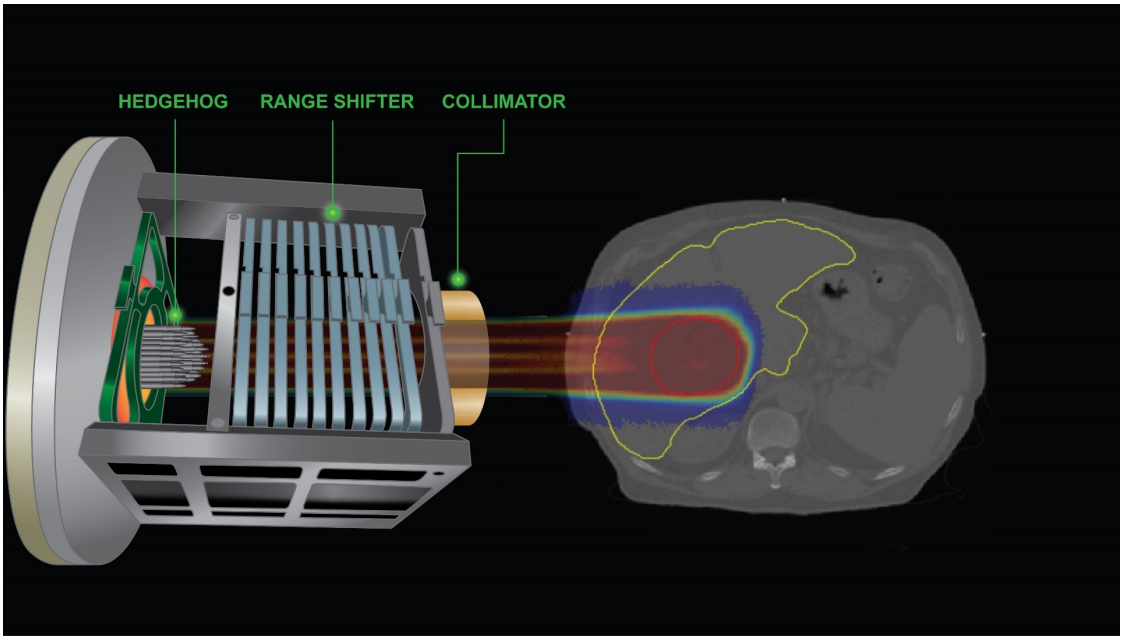

Figure 6: Example of FLASH proton therapy assembly

What must be achieved with the CEM is a Spread-out Bragg peak in the target volume like in proton therapy. The weight of a Bragg peak, i.e. the number of protons of the BP, is proportional to the cross-section of the CEM spike with a given height. The cross-section of each CEM spike is the surface of the spike in the plane perpendicular to the proton beam. The height of the CEM spike (parallel to the proton beam axis) determines the range of the proton beam, which determines the depth of the corresponding Bragg peak.

A sample SOBP is shown in Figure 7. To create this SOBP with a relatively uniform dose across the target volume, one Bragg peak occurring deep in the matter with a high relative dose, and several Bragg peaks with shorter ranges and small relative doses are needed. By summing the Bragg peaks together while taking their respective weight and range into account, an SOBP is expected to result from it.

The general constitution of the CEM leading to this SOBP can be guessed. Indeed, to generate the Bragg peak with a big range and a high relative dose, several CEM spikes with a small height are required. The total cross-section area of all the spikes with the same height generates a large weight.

A Bragg peak with a short range and a small relative dose is produced by a few CEM spikes with big heights. The total cross-section area of those few spikes with the same height generates a small weight. To produce several short-range and small relative dose Bragg peaks, a few spikes with several big heights are required.

It is important to understand that scattering also helps in the dose uniformity in the target volume. Without them, the proton beams would pass through the CEM in a perfectly straight line and the CEM shape would be reflected in the dose distribution. While with scattering, protons of different energies get mixed up while traveling toward the tumor. When reaching the tumor, the protons' energy gets distributed everywhere inside of it in an orthogonal plane leading to a clear SOBP.

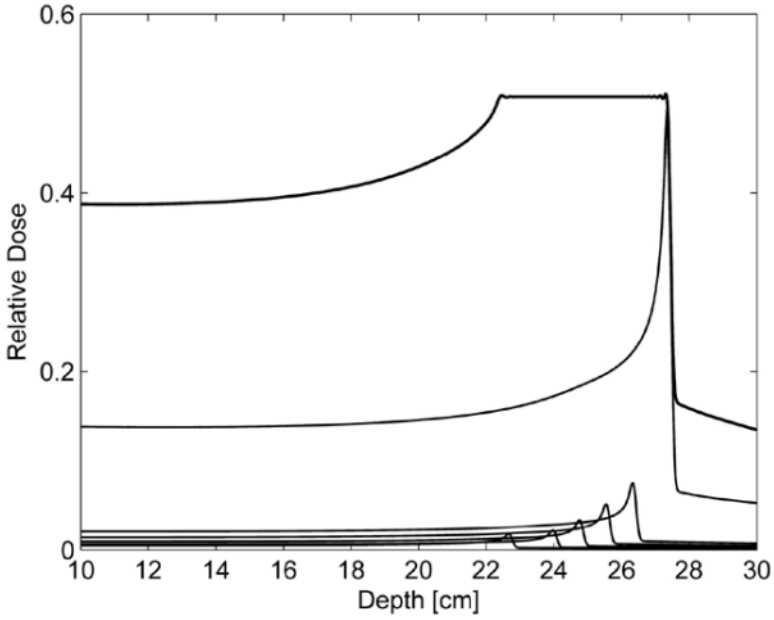

Figure 7: Spread out Bragg Peak composition: Several Bragg Peaks of different ranges and their weighted superposition resulting in a SOBP (Source: Y. Simeonov, 3D Range modulator for scanned particle therapy, p7078)

The exposition time to protons irradiation is reduced leading to a greater quantity of spared normal tissues compared to proton therapy. FLASH proton therapy could be an improvement for tumors that might move when the patient is breathing like lung tumors. Due to the extra short treatment duration, the patient could hold his breath and radiation will occur even more in the target volume while avoiding surrounding tissues. In general, the potential risk of dose displacement increases with the irradiation time [20]. FLASH proton therapy is still in the experimental phase. Experiments were already tested on small animals and a clinical trial will be conducted soon in the USA [18].

Chapter 3

Challenges

One of the main challenges in flash proton therapy is the design of the CEM that makes the dose distribution uniform in depth in the target volume of the patient. The CEM thus must be tuned for each tumor, even for complex ones. Moreover, scattering interactions occur in the CEM in a stochastic way. As a consequence, some protons are deflected from their original direction and pass through several spikes of the CEM, releasing more or less energy than expected if they were traveling in a straight line. Those protons have a consequence on the spatial distribution of dose in the patient. Scattering does play a role in uniform dose distribution. The CEM design must thus consider the tumor volume but also the scattering phenomenon.

Treatment planning consists of defining the grid spots' position and weight, the proton beam's energy, and the CEM's shape. The CEM design must be optimized during treatment planning to be the most suitable for the patient. Treatment planning systems used for CEM designs at IBA are the MIROpt system [1], the FlashOpenTPS system [3], and the Raystation system [4]. MIROpt is mostly used for research purposes while Raystation is a commercial product used for clinical applications. FlashOpenTPS is still in the development phase but some functionalities are already used. This planning system is also used for research purposes. Currently, MIROpt optimizes the shape of the hedgehog using an analytical model relying on the Moliere theory [5]. The optimization is fast but scattering is not described sufficiently accurately.

FLASHOpenTPS uses a tool called MC^2 which performs Monte Carlo simulations [7] to optimize the CEM design. Monte Carlo takes physical processes during particle transport into account with high accuracy, including scattering, which is the most difficult interaction of protons with matter to model. FLASHOpenTPS can build a treatment plan and a CT with the necessary elements required for FLASH PT simulations. If the goal is to perform CEM optimization for a specific tumor, the initial CEM of the treatment plan can be randomly built with the possible spike heights comprised between 0 and the maximum size of the tumor along the depth axis and a CEM size equal to or bigger than the tumor size. The MC^2 tool takes as inputs the CT and the treatment plan created by FLASHOpenTPS and based on those, it simulates the dose deposited in water which represents the outputs. Water is a good tissue substitute (similar density and other properties). In practice, the initial CEM is tested with MC^2, and a first dose map is given. If the dose map is not satisfactory, i.e. if the dose is not homo-

geneously distributed, the design of the CEM is modified and the process is repeated until meeting the desired criteria. However, the optimization with FlashOpenTPS is very slow as several simulations must be performed. One MC^2 simulation lasts several minutes.

The aim of this master thesis is to develop another way of optimizing the design of CEMs using a deep learning model. The deep learning model would be trained based on dose maps as inputs and corresponding CEMs as outputs. The goal of the model would be to predict CEMs suited for the dose maps. Indeed, what clinicians define first is the ideal dose map for the patient and then the corresponding CEM is designed according to it. The CEM must be designed to match the dose map at best, hence the term optimization of the CEM with the AI model. The choice of the AI model is discussed in the next chapter.

The data acquisition for the AI model is performed with FlashOpenTPS. As previously said, FLASHOpenTPS is used to create optimized CEMs according to tumors. Some tumors will be drawn and optimized CEMs will be collected as well as their dose maps. As their acquisition will take time, only a few optimization CEMs will be performed.

FLASHOpenTPS can also perform a single FLASH proton therapy simulation with a given CEM, CT, and the rest of the treatment plan without wanting to perform optimization. This means that the CEM design does not change and no SOBP is expected. Some CEMs can thus be randomly built and FlashOpenTPS would simulate their corresponding dose maps with MC^2. The need of having both randomly built CEMs, and optimized CEMs will be discussed in the next chapter.

When the model is trained with this data set, dose maps of the test set will be given to the trained model. What the model must predict in output is their corresponding CEMs. Afterward, one of the CEMs will be tested on MC^2 and a new resulting dose map will be obtained. Both initial and resulting dose maps will be compared and the goal would be to have the smallest difference between them. Indeed, what matters is the similarity between dose maps and not specifically between CEMs because different CEMs can lead to the same dose map. The dose maps comparison will be performed by analyzing some specific parameters and characteristics such as the Gamma index and the Dose-Volume Histogram [21] [27].

Chapter 4

Data acquisition

This chapter explains how the data acquisition of CEMs and dose maps is performed in different data sets. The assumptions imposed in order to simplify the problem and make it feasible are listed.

4.1 Assumptions

As FLASH proton therapy is in the experimental phase, this technique still has some unknown aspects and effects and data acquisition is limited. Indeed, it is impossible to obtain numerous real data. Moreover, in real cases, many parameters can change in an uncontrolled manner making it difficult to draw conclusions from their comparison. To be able to simulate FLASH proton therapy and collect data, the problem has been simplified by making several assumptions.

The first assumption made is that every CEM will be tested on a parallelepiped-shaped water tank for all cases and not on CT scans of patients. Even though the body density of a person is different for each type of tissue, water is a good tissue substitute, and making simulations on it gives interesting results. This assumption is used in many papers [21] [20].

The second assumption is that the proton beams passing through the CEM are the same for all cases. Proton beams are generated by 16 spots of equal weight (the same number of protons in each spot) that are placed on a 4X4 square spots grid. Each spot center is distant from the others by 7mm and they overlap with each other to obtain a uniform proton fluence map. Proton beams are represented by the fluence as a function of the position in a plane normal to the axis and are represented by a sum of Gaussians. Making weights equal for each spot and having a symmetric grid will allow later to perform data augmentation. However, having the same constant weight for each spot will not lead to a uniform dose distribution in the target volume if this one has a special shape like a sphere for example. Indeed, the fluence (proton/cm) is constant in the 2D plane perpendicular to the beam axis. In the 2D plane, the sphere will project as a circle. On the other hand, the dose has to be delivered in a 3D volume. At the central plane of the sphere, the protons will have to deliver the dose in a depth modulation that is equal to the full diameter of the sphere. However, for planes near the edges of the sphere, the same number of protons will have to deliver the dose but in a much smaller modulation

depth. The dose distribution will thus not be perfectly uniform with constant weight for each spot. However, this assumption makes it possible to use only the dose map in the water tank as input for the AI model. Otherwise, each spot weight must also be taken into account which complexify greatly the problem to solve.

The third assumption made is that the CT with its configuration parameters is unchanged for all cases. Those will be described in the next subsection.

The three first assumptions show that the CEM is changing for each case but not the rest of the settings. The output dose maps provided by MC^2 simulations will thus differ only due to the different CEM shapes. Data comparison is thus feasible. It will allow the Neural Network to assess the FLASH PT functioning in a clearer and faster way.

The fourth assumption made is that the base of the CEM will have the same size for all cases and it is the tuning of the height of each spike that will make the CEM specific for each case. In this project, spikes will have the shape of towers and the base size of the CEM is 20 x 20mm. This size is decided to match the size of the target volume where most of the dose must irradiate.

The fifth assumption made is that each elevation spike can go from 5 mm to 25 mm. This means that the CEM can suit a tumor of 20 mm depth maximum. This value of 20 mm has been chosen for the data set not to be too large and for potential printing to be possible. Indeed if the maximum and minimum heights are too different, it becomes impossible to print a solid CEM that does not bend.

The sixth assumption made is on the number of primary protons to simulate. The higher it is, the less statistical noise there will be but the longer simulations will last. The statistical noise refers to particle interactions that occur in a probabilistic matter causing uncertainty in the results. If more primary protons are simulated, then more statistical samples are obtained. Thus, the simulation results become more statistically representative of the behavior of the proton beam. However, simulating more protons leads to increasing computational time. As a lot of data is needed meaning lots of MC^2 simulations are required, a compromise must be found. A value of $2 * 10^6$ has been chosen. It leads to around 2% of uncertainty on MC^2 simulations. This implies that if two identical simulations are performed, two different dose maps will be obtained. The higher the uncertainty, the more dose maps will be different. In a research context, an uncertainty equal to 2% is sufficient.

4.2 CT

All the data acquisition is performed by modeling FLASH proton therapy using a treatment planning system called FLASHOpenTPS. FLASH proton therapy simulations are conducted in a 100x600x100mm CT that includes all the necessary elements for simulations such as the spot grid, range shifter, collimator, CEM, and the water tank. The water tank, which substitutes the human body, is inserted into the CT from 450mm to 600mm in depth while the origin is defined at the exit of the spot grid. The size of the CT and the depth of water is chosen so that the elements necessary for the FLASH PT can be placed in the CT while having the water tank in the interval of use of the nozzle.

The thickness of the range shifter made of Aluminium is calculated based on the distal distance between the nozzle and the tumor target, the proton beam energy, and the material density. The brass collimator with a 30x30mm square hole is built with a thickness of 65 mm and is used to stop the scattered proton beams. Brass is a high Z material alloy (elements with a high atomic number) that has the property to have a higher stopping power for charged particles like protons. Its slot size is chosen according to the target volume of 20x20x20 and a margin of 5mm on each side to account for uncertainties in beam delivery (and patient setup in real cases). Having a slightly larger slot size compared to the target volume helps ensure that the radiation beam covers the entire target volume.

The plan which consists in the spot grid that generates proton beams is composed of sixteen spots placed in square shape with a spot spacing of 7 mm and a nominal energy of 226 MeV for each spot. The nominal energy accounts for the initial energy of the proton beam before passing through a medium and losing energy. The spot grid is located right in front of the beginning of the CT. The CEM is located 350mm from the isocenter position, which is the center of mass of the target volume. In the CT, proton beams first pass through the CEM, then the range shifter, the collimator, and end up in the water tank. Each element is centered in the position 50x50 in width and height in the CT and is perpendicular to the proton beam's trajectory. The CT is modeled in the form of a 3D matrix whose values correspond to the radio density of each material in the Hounsfield unit. The objective dose in the target volume is set to 2.5 Gy.

A lateral view of this CT scan computed with REGGUI [2] is displayed in Figure 8. The water tank is represented in dark grey while the range shifter is represented in white. The CEM designed for a cubic target volume is the hedgehog-shaped piece located on the left of the CT. The purple line defines the distance between the nozzle and the isocenter position in the water tank. This CT displays only the part of the CT where simulations will be calculated. Its size is 30x30mm in width and height centered in the isocenter position and 600 mm in depth. Its specific size will be discussed later in the dose maps section of this chapter. This specific size explains why the collimator is not displayed in the CT. A resume of some simulation parameters is displayed in a Table in the Appendix.

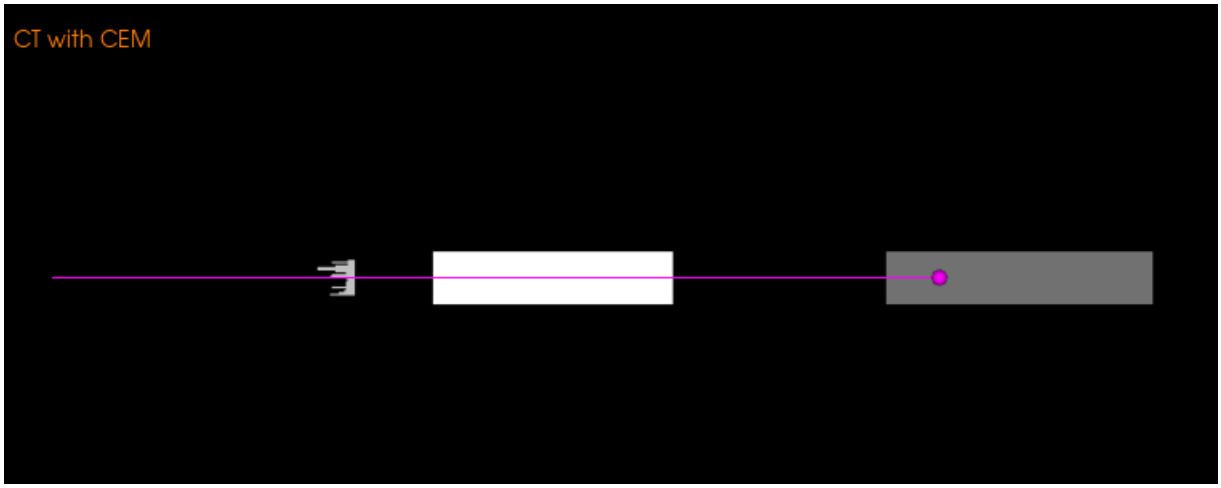

Figure 8: Example of a lateral view of a CT with CEM, range shifter, water tank, and distance (in purple) between the nozzle and the isocenter position

4.3 Conformal Energy Modulators

CEM designs are created in the form of 2D matrices whose size is the CEM base size (20mm x 20mm) and where each pixel (1mm X 1mm) value corresponds to the elevation of each spike. The elevation spike can be equal to any integer value in the interval $[5, 25]$ mm. The minimum size of 5mm of the CEM is comprised in the total thickness of the range shifter. This additional thickness added to the CEM comes from the range shifter being composed of plates of aluminum of a standard size of 4, 8, 12, and 16cm, making it impossible to build a range shifter on any thickness in practice. To solve this issue, an additional layer of plastic is added to the CEM's base. However, this thickness can not exceed 1cm at the risk of warping of the printed CEM. The CEM thickness base must at least be equal to 2mm in order to hold the spikes together. There are 20x20x21! = 2x10e22 possible matrices to generate with these characteristics. The ideal case would be to be able to generate as many CEMs as possible and put them in the data set of the AI model so that the physics is learned quasi-perfectly. In reality, the number of CEMs collected is limited as the training computational time of the neural network and the memory availability need to be taken into account. The same problem would occur for performing Monte Carlo simulations on these matrices to acquire the dose maps.

In practice, all these possible matrices do not give realistic CEMs as the form of the CEMs resembles the form of the tumors [20]. As such, the design of CEMs is performed to obtain a data set close to reality. Two data acquisitions are performed. Both will be used separately for the training of the model.

4.3.1 First acquisition

For the first data acquisition, 8500 CEMs that can be divided into two groups (R and S) are designed. Some are generated randomly (R) and some others are specifically designed to suit a geometric shape (S) representing a fictive tumor in the water tank. The S group

of matrices is used so that the AI model focuses on cases close to reality. Indeed, the purpose of the project is to build an AI model that gives good results with close-to-real case dose maps in inputs. Thus, it might be necessary that the model learns specifically shaped matrices rather than only randomly built matrices to give better results. However, the randomly built matrices could be useful so that the AI model learns real physics instead of possibly learning in the wrong way. Thus, it could be necessary to have CEMs whose design is random and that does not necessarily give a dose map with an SOBP as for real cases. The distribution of those two groups of matrices is 30 % for randomly built ones (group R) and 70 % for specifically built ones (group S).

The R group of matrices is built without FLASHOpenTPS. It is their corresponding dose maps that will be generated by FLASHOpenTPS. The creation of 2079 CEMs for which each spike elevation is decided randomly in the interval $[5, 25]$ is performed. Also, 21 CEMs resembling a range shifter meaning whose spike elevation value is the same over the whole matrix from 5 to 25 are created. Those CEMs will be easy to compare later with their predicted ones and are thus interesting to have in the data set. As it is known that the height of the spikes controls the range of the protons and the number of spikes with a given height controls the weight of the corresponding Bragg peak at that range, it can also be interesting to build CEMs with mostly small or big heights. As such, CEMs with only small height values or big height values or a transitory state between both were generated: 200 CEMs whose spike value is in the range [5,6] to [5,25] and 200 CEMs whose spike value is in the range [5,25] to [24,25] are generated with 10 CEMs for each range. The first group of matrices is composed of 2 500 matrices.

For the S group, matrices were designed in order to obtain specific dose maps with a SOBP occurring in geometric shapes with the help of OpenTPS. Those shapes represent basic tumors inside of which the biggest part of the dose must be distributed. FLASH OpenTPS can draw geometric shapes in a water tank that represent the target volume where the dose needs to be delivered. For each shape, OpenTPS optimizes the design of a CEM by iteratively calling MC^2 for dose computation. As this process takes a lot of computation time, only eight CEMs are generated with this method for eight geometric shapes. Basic forms are used such as a cube and a rectangle. In addition to those, a sphere, two ellipsoids of the same size but in different orientations referred to as ellipsoid 1 and ellipsoid 2, and three ellipsoid agglomerates based on the ellipsoids 1 and 2 are drawn. Those shapes are rounded and concave which are the most encountered characteristics of tumors. The ellipsoid agglomerates resemble potato shapes that could account for less regular tumors. Those were drawn based on the equations of a sphere and an ellipsoid with different parameter values of a,b, and c defined as :

$$(x - x_o)^2 + (y - y_o)^2 + (z - z_o)^2 < R^2 \tag{1}$$

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} < 1 \tag{2}$$

where in this case the target volumes are composed of every pixel that verifies those equations and where x_0, y_o, and z_o are the coordinates of the isocenter position. The geometric shapes are centered in the isocenter position and their sizes are in TABLE 1.

Cube	length = 20	/	/
Sphere	diameter = 20	/	/
Rectangle	length = 20	width = 14	/
Ellipsoid 1	a = 10	b = 7	c=7
Ellipsoid 2	a = 7	b = 10	c=7
Ellipsoid sum 1	$a_1 = 10$ $a_2 = 7$	$b_1 = 7$ $b_2 = 10$	c_1=7 c_2=7
Ellipsoid sum 2	$a_1 = 10$ $a_2 = 7$	$b_1 = 7$ $b_2 = 7$	c_1=7 c_2=10
Ellipsoid sum 3	$a_1 = 10$ $a_2 = 7$ $a_3 = 7$	$b_1 = 7$ $b_2 = 10$ $b_3 = 7$	c_1=7 c_2=7 $c_3 = 10$

Table 1: Geometric shapes used as target volume for the optimization of the 16 CEMs for Group V. a, b, and c are the parameters of the ellipsoids respectively along the width, depth, and height axis of the CT

Note that in this case, the target area where irradiation needs to occur is only in the target volume representing a fictive simplified tumor. It is a case of GTV (gross tumor volume). The GTV is the visible or palpable tumor determined through imaging techniques such as CT scans, MRI scans, or PET scans. In reality, a CTV (Clinical Target Volume) and a PTV (Planning Target Volume) are also needed for patient treatment. A CTV takes into account additional areas where cancer cells may exist out of the GTV. A PTV includes additional margins around the CTV to take uncertainties in the treatment delivery process into account such as patient motion and setup errors.

The resulting optimized CEMs are all of size 30x30. However, as a geometric shape of a specific size in the water tank would give a CEM of very similar size, each optimized CEM is composed of a significant part of size 20x20 or smaller, and all the other elements are equal to 5mm. It is thus possible to reduce the size of the CEM to 20x20 without losing any important information. This size reduction of the CEM and thus of the data set size can speed up the computation time of the AI model. An example of each type of CEMs is displayed in Figure 9. Each type of CEM is named according to its specific target volume for Group S. CEMs that resemble range shifters, that have mostly big values, that have mostly small values, and that are built randomly are respectively named "rangeshi", "big", "small" and "random". What can be observed is that the CEMs designed for a specific target volume are really similar to their target volume. Indeed, a CEM optimized for a spherical target volume as seen in the 7^{th} case really resembles a circle. As for the height of the spikes of Group S, there do not seem to have particular places (edges or center) where the majority of the small or big values are gathered. The distribution of height seems to be uniform for "big", "small" and "random" cases of Group R.

Based on the eight optimized CEMs for specific target volumes, new CEMs can be generated by modifying slightly their design. The modification is performed by adding to each of the spike value a number chosen randomly in the range $[-2, 2]$. This allows the creation of new CEMs whose resulting dose maps are similar to the one obtained for the initial geometric target volume. For each optimized CEMs, 749 new ones are generated.

The eight CEMs and the ones that resulted from a modification of those are part of the data set. They allow the data set to have dose maps that are closer to real cases. The second group of matrices is composed of 6 000 matrices. The 2D matrices representing the 8 500 CEMs are all registered into numpy files and represent half of the data set.

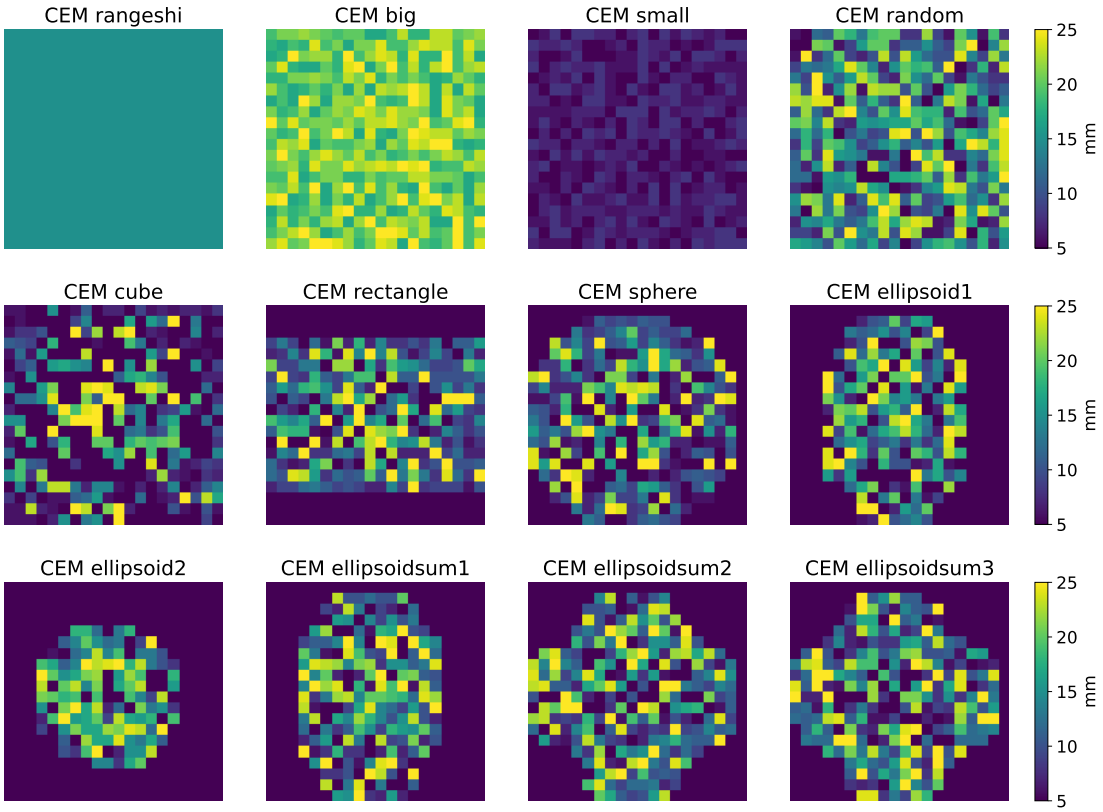

Figure 9: First data acquisition: Example of each type of CEMs

4.3.2 Second acquisition

The second acquisition is composed of CEMs specifically designed to suit geometric shapes like for group S of the first acquisition but with more variability. The eight CEMs previously designed for the eight geometric shapes are part of it. Again, based on the eight optimized CEMs for specific target volumes, new CEMs are generated. The difference with the previous S group is that CEMs created from the eight initial ones have more variability. Indeed, those are generated by adding to each of the spike value a number chosen randomly in the range $[-7, 7]$ instead of $[-2, 2]$. For each optimized CEMs, 188 new ones are generated. Thus, 1500 new CEMs are generated.

Additionally, the same eight geometric shapes but with a smaller size for each are used as target volumes for the creation of eight new optimized CEMs with FLASH OpenTPS. For each optimized CEMs, 188 new ones are generated by adding to each of the spike value a number chosen randomly in the range $[-7, 7]$. In total, the second acquisition is

composed of 3007 CEMs. The size of the 16 geometric shapes used as target volumes is displayed in Table 2. As this data set is a variation of Group S with more variability, it will be called Group V.

Cube	length = 20	/	/
Sphere	diameter = 20	/	/
Rectangle	width = 14	depth = 14	height = 20
Ellipsoid 1	a = 10	b = 7	c=7
Ellipsoid 2	a = 7	b = 10	c=7
Ellipsoid sum 1	$a_1 = 10$ $a_2 = 7$	$b_1 = 7$ $b_2 = 10$	c_1=7 c_2=7
Ellipsoid sum 2	$a_1 = 10$ $a_2 = 7$	$b_1 = 7$ $b_2 = 7$	c_1=7 c_2=10
Ellipsoid sum 3	$a_1 = 10$ $a_2 = 7$ $a_3 = 7$	$b_1 = 7$ $b_2 = 10$ $b_3 = 7$	c_1=7 c_2=7 $c_3 = 10$
Small Cube	length = 16	/	/
Small Sphere	diameter = 16	/	/
Small Rectangle	width = 10	depth = 10	height = 16
Small Ellipsoid 1	a = 8	b = 5	c=5
Small Ellipsoid 2	a = 5	b = 8	c=5
Small Ellipsoid sum 1	$a_1 = 8$ $a_2 = 5$	$b_1 = 5$ $b_2 = 8$	c_1=5 c_2=5
Small Ellipsoid sum 2	$a_1 = 8$ $a_2 = 5$	$b_1 = 5$ $b_2 = 5$	c_1=5 c_2=8
Small Ellipsoid sum 3	$a_1 = 8$ $a_2 = 5$ $a_3 = 5$	$b_1 = 5$ $b_2 = 8$ $b_3 = 5$	c_1=5 c_2=5 $c_3 = 8$

Table 2: Geometric shapes used as target volume for the optimization of the 16 CEMs for Group V. a, b, and c are the parameters of the ellipsoids respectively along the width, depth, and height axis of the CT

An example of each type of CEMs is displayed in Figure 10. What can be observed is that all the CEMs are really similar to their target volume. When comparing the CEMs designed for the same geometric shape with their two different sizes, the height values have different distributions. This confirms the fact that it was necessary to perform CEM optimization for the eight new geometric shapes. Group V is used to verify whether the Neural Network performs well even if trained on more variable data and if overfitting occurs in this case. Indeed, it would mean that the model learns a part of the physics and not only remembers patterns.

The addition of a second size for each geometric shape is used to verify whether the NN really understands the relationship between the edges of the dose maps and the size of the CEM. Group V has fewer pairs of CEMs and dose maps than group S because of a matter of time. However, the results of Group S for only 4 types of CEMs (3 000 pairs)

already gave very good results in terms of MSE when all the data was not fully acquired. Comparison between Group S and Group V will thus still be meaningful.

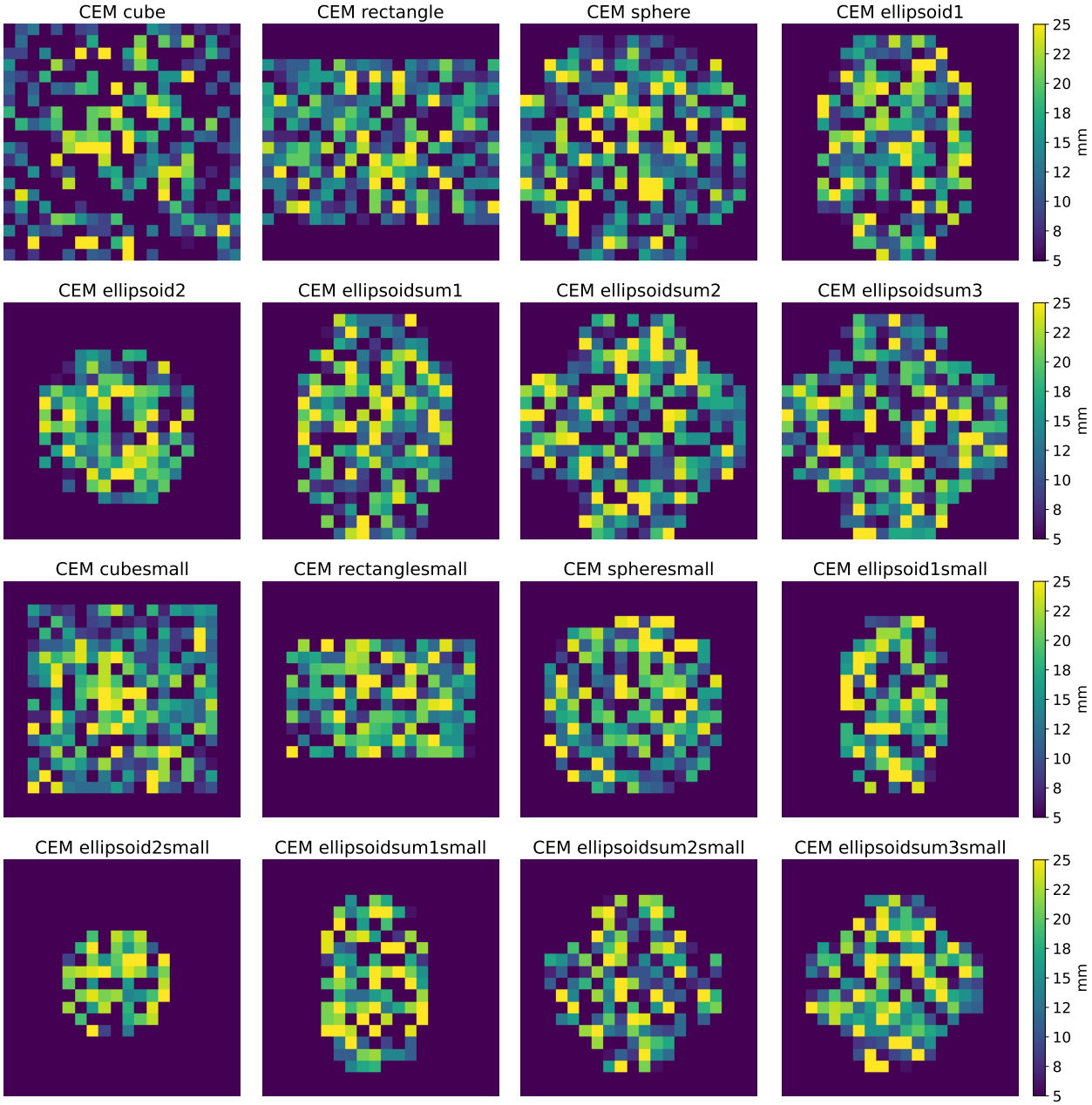

Figure 10: Group V: Example of each type of CEMs

4.4 Dose maps

The second half of the data set is composed of the dose maps resulting from the simulation of FLASH PT in the CT using the CEMs previously built. The process is the same for the first and second acquisitions of CEMS. To obtain those, the matrices representing the elevation map of all the CEMs are loaded with the numpy library. Through OpenTPS functions, the 2D matrices are converted into 3D ones and placed in the CT at the right place in the trajectory of the proton beams. Indeed, each pixel value where lies the CEM is changed from -1049 to 279 HU based on the elevation spike values.

Afterward, the CT scan is tested on MC^2 for every CEM and the corresponding dose map is given as the output. MC^2 simulates the propagation of protons through all the FLASH accessories and the dose deposition in the water phantom.

The interesting part of the dose to register in the CT scan lies only in the water tank as it represents the hypothetical patient. Also, the whole water tank is very big and all the dose does not deposit everywhere in it. Indeed, it is possible to define a region of interest (ROI) that represents the region where lies the higher-level dose distribution. This ROI is a rectangular parallelepiped whose size is set to 30x40x30mm and comprises the 20x20x20mm cubic target volume centered at the isocenter position in addition to 5 mm on each side of this cube and 20 mm in depth after it as margins to ensure most of the dose is included in it.

As a reminder, every geometric shape is equal to or smaller than the 20x20x20mm cube which is why this shape is taken as the reference to draw the ROI. The region of interest is drawn with OpenTPS. As the ROI is the only part of the dose map that will be registered, OpenTPS does not perform simulations over the whole CT but only in the portion in width and height of the ROI. It allows MC^2 simulations to be shorter. The 30x30mm size of the eight optimized CEMs is also due to the simulations being performed in the ROI width and height.

Figure 11 displays the same CT as in Figure 8 with the addition of the dose concentration. It can be observed that the proton beams pass through the CEM, then the range shifter, and finally enter the water tank. The cubic target volume is drawn in green whereas the rectangular parallelepiped region of interest is in red. The dose in the target volume seems uniform and of high value at first glance. The region of interest which is the registered part of the dose map is big enough to include the majority of the dose in and around the target volume.

There are a few high-dose releases before the target volume in the water tank but no organ at risk was defined thus this region is not interesting. If an organ or a region were defined to be at risk before the target volume, then the least dose release should also be attained in it while maintaining the wanted dose in the target volume. The challenge would become more complex.

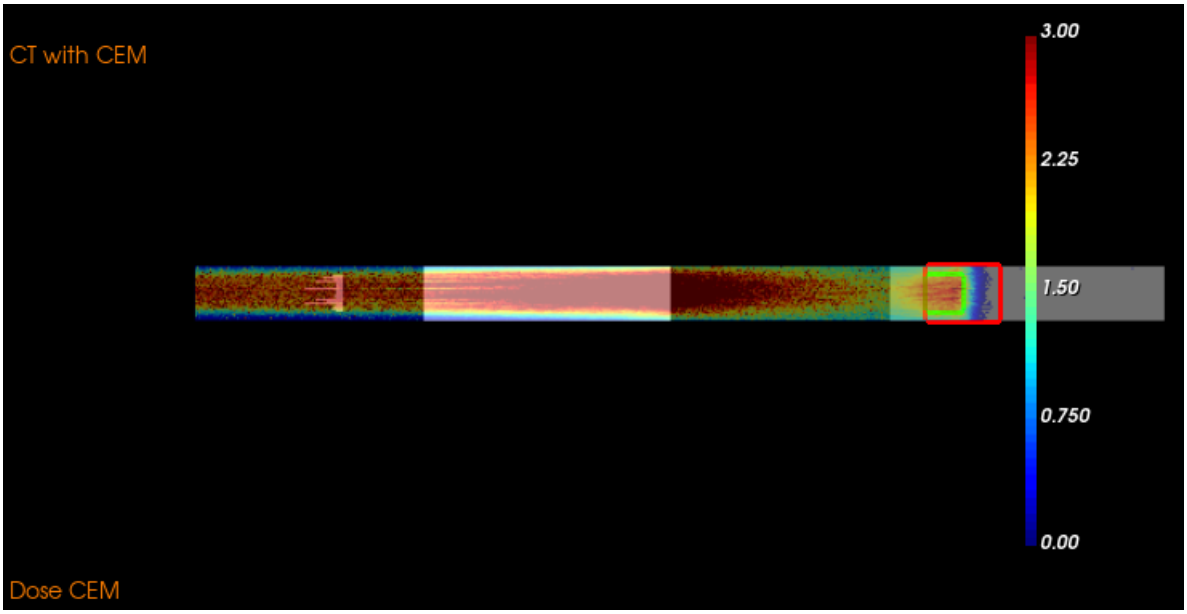

Figure 11: Example of a lateral view of a CT with CEM, range shifter, water tank, and dose map. The color map represents the dose in Gy

A transversal slice of the CT at the center of the target volume is displayed in Figure 12. As previously observed, the highest dose concentrations are located inside the target volume. However, the dose distribution is not perfectly uniform, especially at the edges, as would be expected in real cases.

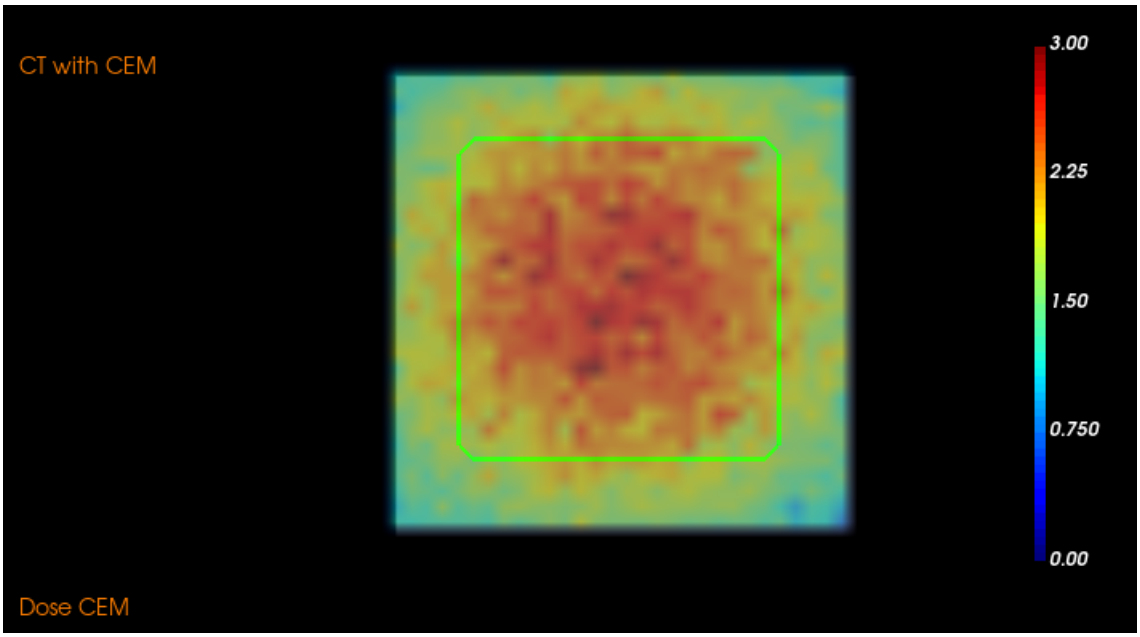

Figure 12: Example of a transversal slice of a CT with the target volume and the dose map

In Figure 13, the dose intensity as a function of the distance in the water phantom for a segment of the CT along the lateral axis is displayed. A SOBP can be seen between the green lines that delimitate the cubic target volume. This SOBP is not perfect as the edges of the target volume have smaller dose intensity. It goes according to observations made in Figure 12. Once again, this is explained by the spot weights being all equal, leading to a not perfectly uniform dose distribution in the target volume.

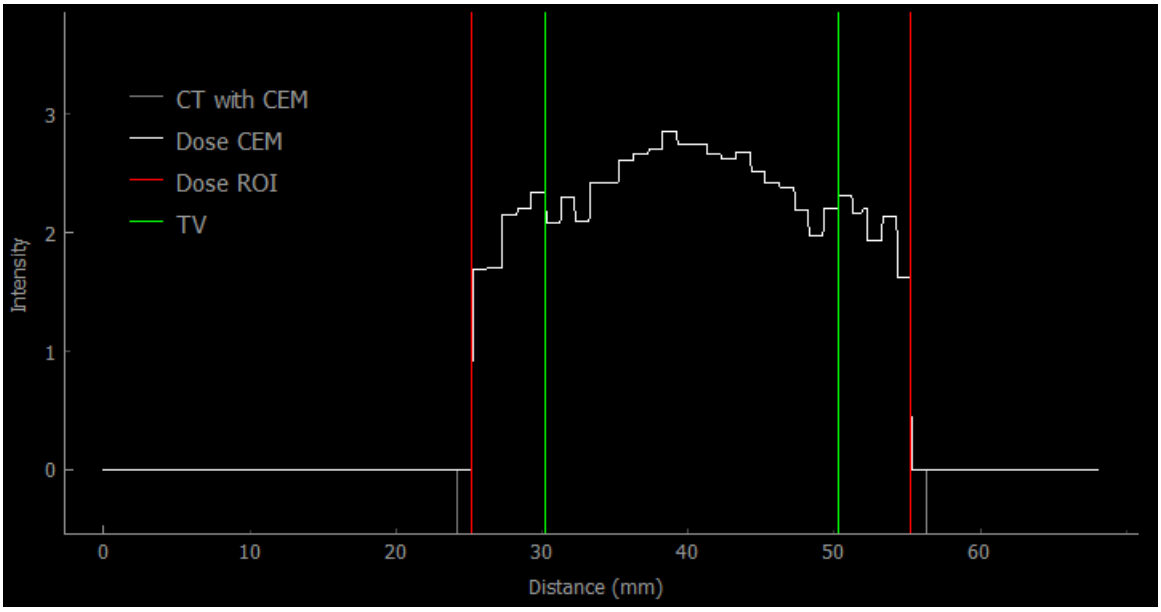

Figure 13: Example of SOBP in the case of a cubic target volume (TV)

The registration of the dose maps is also made in NUMPY format which is easy to handle and transform into tensors for the CNN model. The size reduction of the registered dose maps into the ROI size allows the shortening of the computation time of the CNN model. At this point, the data set for the AI model is complete and is composed of pairs of CEMs and corresponding dose maps.

An approximation of dose maps can be visualized in Figures 14 and 15. The first dose map corresponds to a spherical target volume whereas the second one is the result of a CEM build randomly. The protons' trajectory passes through this rectangular parallelepiped from the bottom to the top and perpendicularly to its surface. It can be seen that the majority of the dose represented in blue is included in the 30x40x30 size rectangular parallelepiped and that the dose deposition stops after the target volume according to the theory. In the first case, the dose looks uniform and has a sphere shape. In the second case, the dose is distributed in a plane.

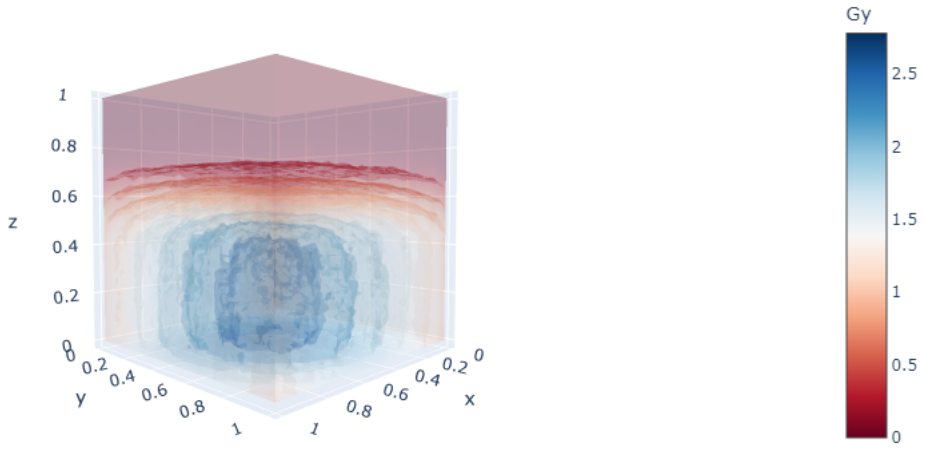

Figure 14: Dose map of a spherical target volume

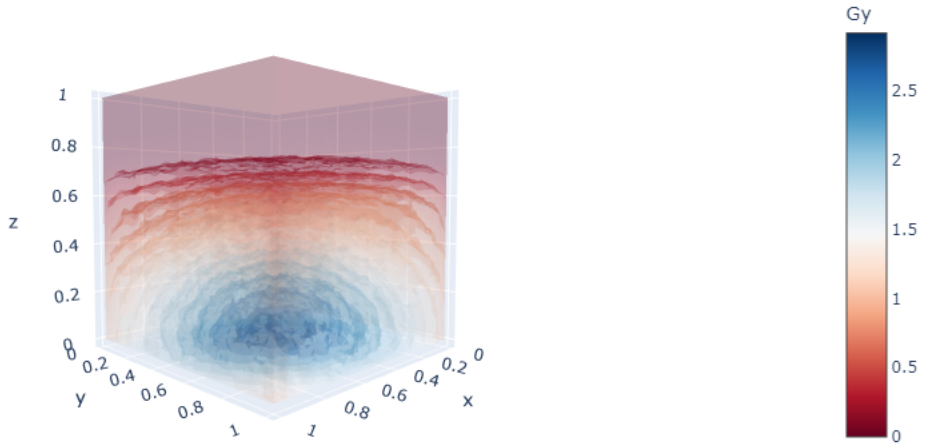

Figure 15: Corresponding dose map of a CEM built randomly

Chapter 5

Convolutional neural networks

This chapter introduces the notions of Artificial intelligence, Neural Networks, and Convolutional Neural Networks. Their functioning and domain applications are briefly explained.

5.1 Artificial Intelligence

Artificial intelligence consists of teaching machines to perform complex tasks that would normally require human intelligence, hence its name. Nowadays, AI is used in many fields such as industrial automation, financial analysis, commercial optimization, and medical diagnosis. The purpose of using AI is to improve the efficiency, productivity, and accuracy of the tasks performed but it also allows to reduce possible human errors and can be cost savings.

5.2 Neural Networks

Deep Learning is a subgroup of Machine Learning which is a subgroup of the Artificial intelligence domain. A Neural Network (NN) is a deep learning process whose architecture and function mimic human neuron functioning. The NN is composed of interconnected nodes called neurons which are arranged into layers. [15] A NN can be applied to several types of problems such as regression, classification, denoising, and anomaly detection problems. A regression problem is a problem for which the predicted output is continuous while a classification problem is where a discrete output is predicted.

Artificial neural network applications can go from easy ones like distance minimization to complex ones such as face recognition and image segmentation. However, the drawback is that a large data set is required for the training of the model and the amount needed depends on the problem's complexity.

In NN, to learn complex relations between the inputs and outputs of the data, nonlinearity is added to the model. This nonlinearity is introduced with the use of activation functions such as RELU (Rectified Linear Unit), Sigmoid, and Tanh functions that are applied to the output of each neuron. Those functions have the particular characteristic to be nonlinear, differentiable, and simple to compute. In addition to introducing nonlinearity, they are easily calculable and they make backpropagation possible. Each

activation function has its particularity which makes them more appropriate for certain kinds of problems.

In general, the first layer of the NN is the layer that yields the inputs, and the last layer yields the outputs. In between, several hidden layers are used to process specific aspects of the inputs. The number of layers and neurons in each layer depends on the complexity of the problem that has to be solved and the desired network architecture. All those layers are interconnected and each neuron is connected to neurons of the next layer by parameters called weights. Depending on the value of the weight, a neuron will excite another neuron in the next layer or not and with a certain intensity. Sometimes, other parameters called bias or intercepts are also describing the relationship between neurons, in addition to the weights.

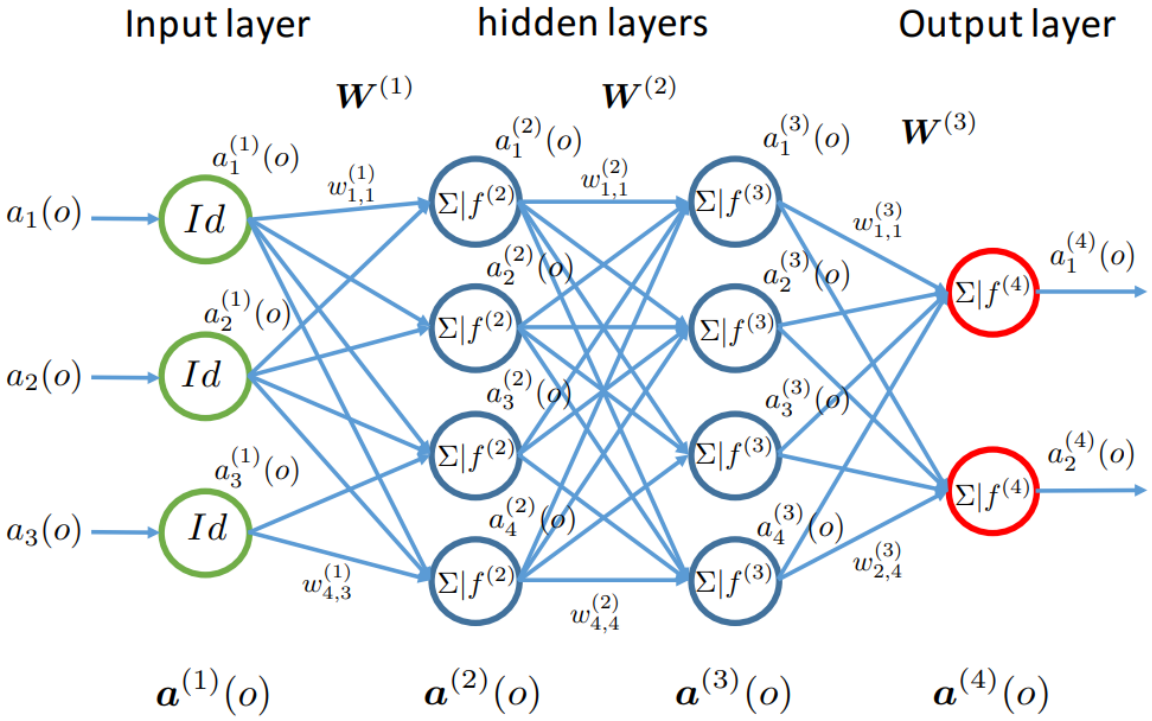

Figure 16: Neural Network architecture: Multilayer Perceptron (Source: L. Wehenkel, P. Geurts, Introduction to Machine Learning, Lecture 5)

To help to visualize a NN and how neurons are connected and predictions are computed, an example of Multilayer perceptron architecture is displayed in Figure 16. A Multilayer perceptron (MLP) is a common Neural Network characterized by having multiple layers of neurons, each fully connected to the next one. [25] In this example, the MLP has three inputs, two outputs, and two hidden layers composed of 4 neurons each. In terms of notation, $a_m^{(l)}(o)$ is the output (also called activation) of the m^{th} neuron of layer l for an object o. An object is an instance in a data set. Weights and bias are represented by w where $w_{m,n}^{(l)}$ is the weight of the edge from neuron n in layer l to neuron m in layer l+1 and the bias of a neuron m in layer l is $w_{m,0}^{(l)}$. The matrix notation for the

outputs and weights are respectively $\mathbf{a}_{(o)}^{(l)}$ and $\mathbf{W}^{(l)}$. The activation function of layer l is represented by $f^{(l)}$. The equation used to predict the output of layer $l+1$ is :

$$a_m^{(l+1)}(o) = f^{(l+1)}(w_{m,0}^{(l)} + \sum_{n=1}^{nbr_l} w_{m,n}^{(l)} a_n^{(l)}(o) \tag{3}$$

The values of the weights must be adjusted so that the model fits the data well. To know whether the model fits the data set or not, a loss function that calculates the difference between the real output and the predicted output is estimated. During the training process, weight values are updated to decrease the loss function along the number of iterations. One way to update the weights is to perform gradient descent and backpropagation [12].

Backpropagation is the algorithm for training neural networks. It computes gradients of the loss function with respect to the weights by the chain rules method. Gradient descent is an optimization algorithm that adjusts parameters, i.e. weights and sometimes bias, to minimize the loss function in the fastest way. The parameter update involves adding a step value to the parameter values, which is large when far from the minimum and small when close to it. It updates weights by propagating errors backward through the network. By using backpropagation and gradient descent iteratively, NN predictions improve on training data. [23]

Usually, the data set onto which the Neural Network must learn is divided into three subsets called the training, validation, and test sets. The training set is the largest subset with which the model is trained. The model updates its weights to reduce the difference between the predicted and true outputs of this data set.

The validation set is used to assess the temporary performance of the model. The loss comparison between the training and validation sets as a function of the number of epochs permits tuning the model hyperparameters that are not trainable. Their tuning allows the model to make a better generalization.

The test set is independent of the model as it has never been seen by the model before. The trained model's final performance on new data is performed on this set. The results of the model on the test set highlight whether the model managed to generalize to new, unseen data or not.

This method of separating the data into 3 subsets is commonly used for any supervised learning problems and is not limited to NNs. Supervised learning is a type of machine learning where labeled inputs and outputs are provided and the purpose is to find a function f of the inputs that approximates at best the output [10].

Another utility to have a validation set is to prevent the model to overfit. Overfitting occurs when the model becomes too specific to the training set and does not learn general patterns between the inputs and the outputs anymore. If the model is too specific to the training set then it will perform poorly to new unseen data. It is important to stop the

training of the model right before overfitting occurs. A way to know the sufficient number of epochs after which the model should stop learning is by looking at the training and validation losses as a function of the number of epochs. If the training and validation loss curves are both decreasing then the model is underfitting meaning that the model still needs to be trained. If the training loss curve still decreases while the validation loss curve starts increasing, then overfitting is happening.

5.3 Convolution and its application with NNs

Convolutional Neural Networks (CNNs) are used for voluminous data sets that have a known grid-like topology such as images. CNNs exploit the spatial relationships between the data points enabling them to efficiently perform computer vision tasks such as image classification, object detection, and segmentation. The name Convolutional Neural Network comes from the mathematical operation the network uses namely the convolution.

"A convolution is a mathematical operation on two functions that produces a third function that expresses how the shape of one is modified by the other". It is defined as :

$$(f * g)(t) = \int_{-\infty}^{+\infty} f(\tau)g(t-\tau)\,d\tau \tag{4}$$

In the CNN case, the function f is the input while the other function g is a kernel filter [11]. A kernel filter is a small matrix of parameters that is applied to portions of the input matrix through convolutions. The filter slides across the input matrix to perform convolution for each position of the input. In practice, performing a convolution consists in computing the dot product between the filter and the input values at each position of the input.

A simple example is shown in Figure 17 where convolution is applied on a 3x3 input matrix with a 2x2 kernel filter. In this example, the computation of the first value of the output matrix is highlighted in blue. The convolution is calculated as followed :

$$0 * 0 + 1 * 1 + 3 * 2 + 4 * 3 = 19 \tag{5}$$

In the case of a 3x3 input matrix, the input can be called a two-order tensor. A tensor is a mathematical object whose value is defined in a vector space. Convolution can be applied to higher-order tensors depending on the size of the specific input of the problem.

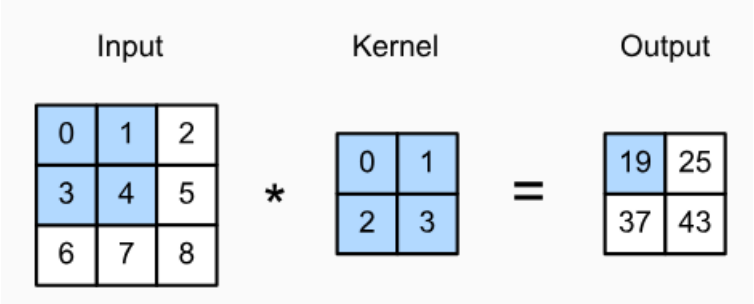

Figure 17: Example of convolution on a $3X3$ rank-2 input tensor with a $2X2$ kernel filter (Source: $https://d2l.ai/chapter_convolutional-neural-networks/conv-layer.html$)

Convolutions are performed to extract features from the input. The output of a convolution is called a feature map. Each feature map element is a function of the corresponding input element and the neighboring input elements. This allows the CNN to capture spatial relationships between the input values enabling the extraction of features. In the case of CNN, several feature maps are computed and the parameters of the kernel filter are optimized during the training. The same kernel filter is applied to produce all the feature maps meaning that the optimization of its parameters is done to obtain the best possible feature maps for the given task.

The three main characteristics of CNNs are sparse interactions, parameter sharing, and equivariant representations that oppose classical NN [11]. In CNN, the use of a smaller kernel filter compared to the input makes that each input unit is not connected with each output unit in opposition to classical NN where they are fully connected. These sparse interactions between the inputs and outputs decrease the number of parameters needed to represent the network. It has the advantages of reducing the memory requirements of the model and improving its efficiency. When the CNN is deep enough, complicated interactions can be learned by the model as units in the deeper layers can indirectly interact with a larger portion of the input.

In a classical NN, each parameter, which is the weights, is used a single time to compute the output of a given layer. In CNN, the parameters of the kernel are the same to compute every feature map. This parameter sharing means that the network must not learn parameters for each location in the input but a single set of parameters optimized for feature detection that is used across the entire input space. Parameter sharing has the advantages of reducing storage requirements and improving efficiency.

In CNN, the output representation changes in the same way that the inputs if the inputs undergo transformations like translations. The output representation is said equivariant to transformations of the input. Equivariant representations are useful, especially in cases where the same features are represented several times in different locations or orientations or with time-series data when it is important to know when features appear in the input.

Convolution in CNN cases is characterized by four parameters called kernel size, padding, strides, and dilation. [15] The kernel size is the size of the filter that is moved

across the input during convolution. The padding is a method used during convolution that adds a frame of zeros around the input whose size is the padding size. It is usually used to decide on the spatial dimension of the feature map. The stride method specifies the step size for which the kernel filter must shift across the signal. This method is mainly used for spatial dimension reduction of the feature map by a constant factor. The dilation method adds rows and columns of zeros in the kernel filter enabling the expansion of the filter without adding weights. It allows an increase in the units' receptive field size without changing the number of parameters.

A convolution layer where convolution is applied on the input does not change the structure of the input as the same linear transformation is applied locally everywhere on it. In addition to using convolution layers, a CNN sometimes uses pooling layers. This method is very useful in cases where the input size is very large. Pooling layers decrease the dimension of the input while preserving its spatial relationships. It consists in replacing some parts of the feature map with a summary statistic.

Two regularization methods that are often used with CNN are dropout and batch normalization. Dropout is a method used to prevent overfitting and to help the model learn generalization instead of memorizing specific patterns. In practice, dropout consists in deactivating some neurons randomly during the forward and backward propagation of the training phase. The dropout rate is set by a parameter p. In this case, the dropout for the last two layers of the encoder is equal to $p = 0.5$ which means that there is a 50% probability that a neuron is deactivated at each iteration in those layers. When a neuron is deactivated, it will not activate neurons in the next layer and its gradient will not be evaluated backward. Some neuron dependency is reduced and the model becomes more robust.

Batch normalization (BN) is another method that acts as a regularizer and is used to speed up the convergence of the model. It consists in normalizing the activations of a layer for each mini-batch by these formulas [13]:

$$x^* = \frac{(x - E(x))}{\sqrt{var(x)}} \tag{6}$$

$$x^{**} = \gamma * x^* + \beta \tag{7}$$

First, the activations referred to as x are normalized by subtracting the mean and dividing it by the standard deviation of the mini-batch according to Equation 6. This normalization is performed along each feature dimension independently. According to Equation 7, the normalized activations are later shifted with the γ and β trainable parameters whose purpose is to find the optimal scale and mean for the normalized activations. As stated in [13], Batch normalization is often used to help stabilize the training and speed up the convergence of the model.

Chapter 6

AI model

This chapter explains the preprocessing applied to the data, and how the data is distributed into the training, validation, and test sets. The choice of the AI model, its architecture, and its parameters are also discussed.

6.1 Preprocessing

Data augmentation can be performed to increase the data set size without performing any additional simulations. The goal of increasing the number of data is to possibly improve the performance of the Neural Network and to show him more diversified data to increase his robustness. In this case, the data augmentation that can be made is by rotating the CEMs and the corresponding dose maps by 90, 180, and 270° along the second axis which is in the direction of the proton trajectory. Indeed, given the symmetry of the spot grid and the equal weight of each spot that sends proton beams, rotating the CEMs and dose maps would keep the FLASH PT physics unchanged. The data set is thus multiplied by four very easily and quickly.

However, any other sort of data augmentation like cutting, or zooming can not be performed as it would alter the physics of FLASH proton therapy. Indeed, changing the size of the dose maps by zooming would modify the relationship of slices along the depth axis and predict the CEM falsely. Changing the values of the dose maps by adding a constant value everywhere would also be wrong as the CEM values will not be impacted proportionally but in a more complex manner as a function of the energy and weight of the proton beams. Cutting is also not an option as it would not take lateral scattering that deviates the protons' trajectory into account. Indeed, if we cut the CEM and dose map into two pieces, a part of the dose in the kept dose map would in fact come from protons passing through the cut side of the CEM. The physics of FLASH proton therapy would be altered.

Normalization of the inputs and the outputs can also be done to facilitate the convergence of learning and improve the accuracy of prediction. By dividing the inputs and outputs by the maximum value of dose concentration and height, which are equal to 5 and 25 respectively, normalization is achieved such that their values fall within the range of $[0, 1]$. This normalization process also facilitates a meaningful comparison between the inputs and between the outputs.

6.2 Choice of AI model

What is asked of the AI model is to extract the features of the 3D dose maps and predict the 2D elevation map of the CEM. The dose maps are voluminous datasets that have a known grid-like topology like images. CNNs are well adapted to work on 2D or 3D images. For example, compared to classical NN, a CNN learns to detect edges represented by high gradients more easily due to the convolution operation. As CNNs exploit the spatial relationships between the data points, they will be able to understand the relationship between the layers along the depth axis of the dose map.

Indeed, it is along this axis that the CEM has an impact on the dose map. The proton beams pass through the CEM and this CEM influences the spread of the dose distribution in depth in the patient. Understanding the relationship of layers along the proton beams' trajectory axis allows the study of the real physics of FLASH proton therapy.

A classic CNN downscales the inputs to extract their features and gives as output a value or a class depending on the type of problem. However, the wanted size of the output is a 20x20 matrix representing the elevation map of the corresponding CEM. A type of CNN structure often used in medical imaging is the U-Net introduced by Ronneberger et al., in [19]. This type of architecture has an encoder part that downscales the inputs into feature maps and a decoder part that upscales the feature maps of the last encoder layer to obtain outputs in the form of matrices. This type of CNN architecture is mostly used for biomedical image segmentation where the input size equals the output size. The goal of this U-Net is to highlight contours and remove unwanted information such as noise or other superficial elements of the image. Its structure is more complex than a basic CNN due to the additional decoder part.

The downscaling and upscaling of the feature maps can be performed by strided and transposed convolutions respectively. A strided convolution is a convolution where the kernel filter slides over the input by more than 1 pixel. The size of the output feature map can thus be smaller than the input size. Strided convolutions permit downsampling that extracts features, and computation time reduction. A transposed convolution is a convolution where padding is applied to the input, meaning that a frame of zeros is added around the input. The size of the output feature map can thus be bigger than the input size. Transposed convolutions permit upsampling that allows the generation of higher-resolution outputs.

In the case of dose maps and CEMs as the data set, the size of the input is not equal to the size of the output. The architecture of a basic U-Net must be modified to start from the 40x30x40 input dose map and obtain the 20x20 corresponding CEM. The size of the output can not be predetermined. The structure of the model must be designed so as to obtain the wanted output size. The number of layers determines the complexity of the model. In order to extract complex features from the input, the CNN must contain multiple layers of kernel filters. Indeed, basic features like edges can be detected in the first layer but the more complex ones such as patterns and shapes are only detected in the last layers. In the case of dose maps being the inputs, four layers are used.

6.3 Architecture

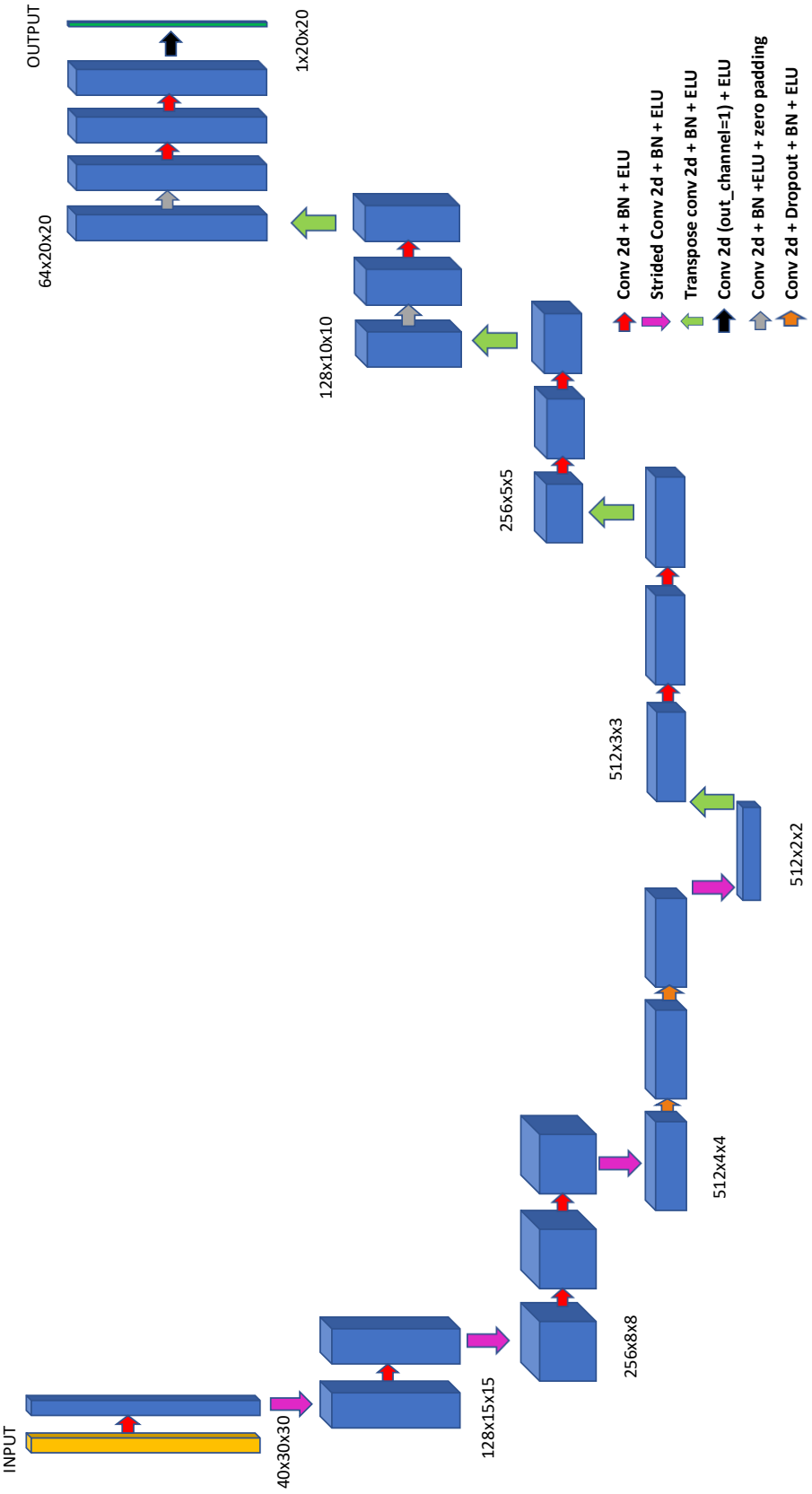

Figure 18: U-Net structure

The architecture of the model is inspired by the U-Net in the article written by V. Boudart which also uses strided and transposed convolutions with Batch normalization and the same number of layers [6]. The rest of the architecture is adapted to this specific case to match the output size.

The architecture of the CNN is composed of 4 layers in the encoder, 4 layers in the decoder, and a last layer to obtain the final output. Its structure is displayed in Figure 18. A layer of the encoder is composed of convolution layers, strided convolution layers, and activation functions. The encoder part takes in input the 40x30x30 dose maps and changes their size to 512x2x2 feature maps.

The activation function used in this architecture is the exponential linear unit (ELU) and is defined as:

$$f(x) = x, if x >= 0 \tag{8}$$

$$f(x) = \alpha * (\exp^{x} - 1), if x < 0 \tag{9}$$

This function has the advantage of making backpropagation easier to compute thanks to being smooth, continuous, and differentiable everywhere. As its positive region is linear, it helps the network to retain information. Moreover, the ELU function is more robust to noise in the inputs than some other activation functions.

In this architecture, the convolution layer is a 2D convolution that does not change the size of the feature maps, that applies batch normalization and sometimes dropout. Its kernel size is equal to 3. At its end, the ELU activation function is applied to introduce nonlinearity. The strided convolution layer is a 2D convolution that doubles the number of feature maps. Its stride parameter is equal to 2, which reduces the second and third dimensions of the feature maps by a factor of 2. Batch normalization and the ELU function are also included in it. The first convolution layer takes as input the dimension of the input to study and gives as output the input number of feature maps decided. The important input dimension is the dimension along the depth axis equal to 40. The initial number of feature maps is set to 32. Its size will be optimized in Chapter 7.

Each layer of the decoder is composed of convolution layers, transposed convolution layers, and activation functions. The encoder part takes in input the 512x2x2 feature maps and gives tensors of size 32x20x20 in output. The transposed convolution layer is a 2D convolution that divides the number of feature maps by a factor of 2. Its stride parameter is equal to 2, which doubles the second and third dimensions of the feature maps. Batch normalization, sometimes zero padding, and the ELU function are also included in it. The last layer of the model is composed of a convolution layer and an activation function. This layer reduces the size of the output down to 1x20x20 which is the size of the CEMs of the dataset.

6.4 Other parameters influencing the model's performance

The loss function used to study the model's performance is the Mean Square Error (MSE). The formula used to calculate it is :

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{10}$$

where n is the number of elements in the training set, y is the true output and $\hat{y}$ is the predicted output. The Mean Square Error has the advantage of penalizing more outliers which might impact badly the CEM design. Indeed, a big difference in the height of the CEM spikes could completely change the dose distribution in the tumor. Other advantages of the MSE are that this loss function is easy to understand and differentiable, which facilitates backpropagation.

An optimizer helps with the weight update of the model while minimizing the loss function. Basically, it helps to optimize the model performance. The optimizer used is Adamax. It is well known for being efficient for computer vision tasks. Its advantages are that this optimizer is robust to the hyperparameters of the model. This means that even if hyperparameters are not precisely chosen according to the problem, the optimizer still performs well. It also has the advantage to adapt the learning rate for each parameter, permitting the U-Net to converge faster.

The weight initializer initializes the weights of the model. The one used is the Normal initializer which initializes weights from a Gaussian distribution with a mean of 0 and a specified standard deviation of 0.02.

The learning rate is the parameter that determines the step size to make when updating the weights. A very high learning rate makes the model converges faster but the risk is to have instability and having the model oscillate around a local minimum in the loss function as in most cases they are non-convex. A too-small learning rate makes the model converges very slowly. The goal is to find the intermediate learning rate between these two cases for this specific problem of CEM optimization. Its value is set to 0.0001 at first.

Ideally, to find the minimum loss over the data, the loss must be computed over all the samples in the training set at each iteration. However, it takes a lot of memory space and time. A solution to this problem is to perform mini-batch gradient descent instead. It consists in evaluating the loss over a small subset of examples at each iteration. The number of training examples, i.e. the batch size, plays a role in the model's performance. The ideal batch size depends on the problem and the variance in the input data. Its ideal size is a compromise between the available data and the model's convergence. Its size value is set to 32 first. The batch size and learning rate values will be optimized later on. Their initial values are set according to values often used in the literature.

6.4 Data distribution

Ensuring a uniform distribution of each CEM type in the training, validation, and test sets is crucial for preventing overfitting and improving the model's performance. Indeed, when the training, validation, and test sets are too dissimilar, overfitting occurs early as the model will become specific to the training data sooner. The dissimilarity shows that the three subsets are not representative of the problem needed to be solved. All the subsets must reflect the overall data variability. If not the case, then the model's performance on the validation and test sets could be biased. It is thus important to have the three subsets of the data uniformly distributed to have good accuracy of the model's performance.

To perform a uniform data distribution, each CEM type is considered a class. Each class is shuffled once to ensure uniformity in the class itself. Otherwise, all range shifter-like CEMs with small values will be in a set and all those with big values will be in another set for example. Each class is then split manually into three groups according to the wanted percentages of 70%, 15%, and 15% for the training, validation, and test sets. When the three sets are created, they are shuffled once again to be sure data is not sorted class by class. Shuffle is made the same way for inputs and outputs to ensure the matching is maintained. This way, a uniform distribution is obtained.

To verify the uniform data distribution in the training, validation, and test sets, histograms are built for the complete data set of the first acquisition and Group V in Figures 19 and 20. As observed, data distribution is uniform in the three sets. Percents are indeed the same for the three histograms of each group. The number of data confirms that there is 70% of the data in the training set and 15% in the validation and test sets.

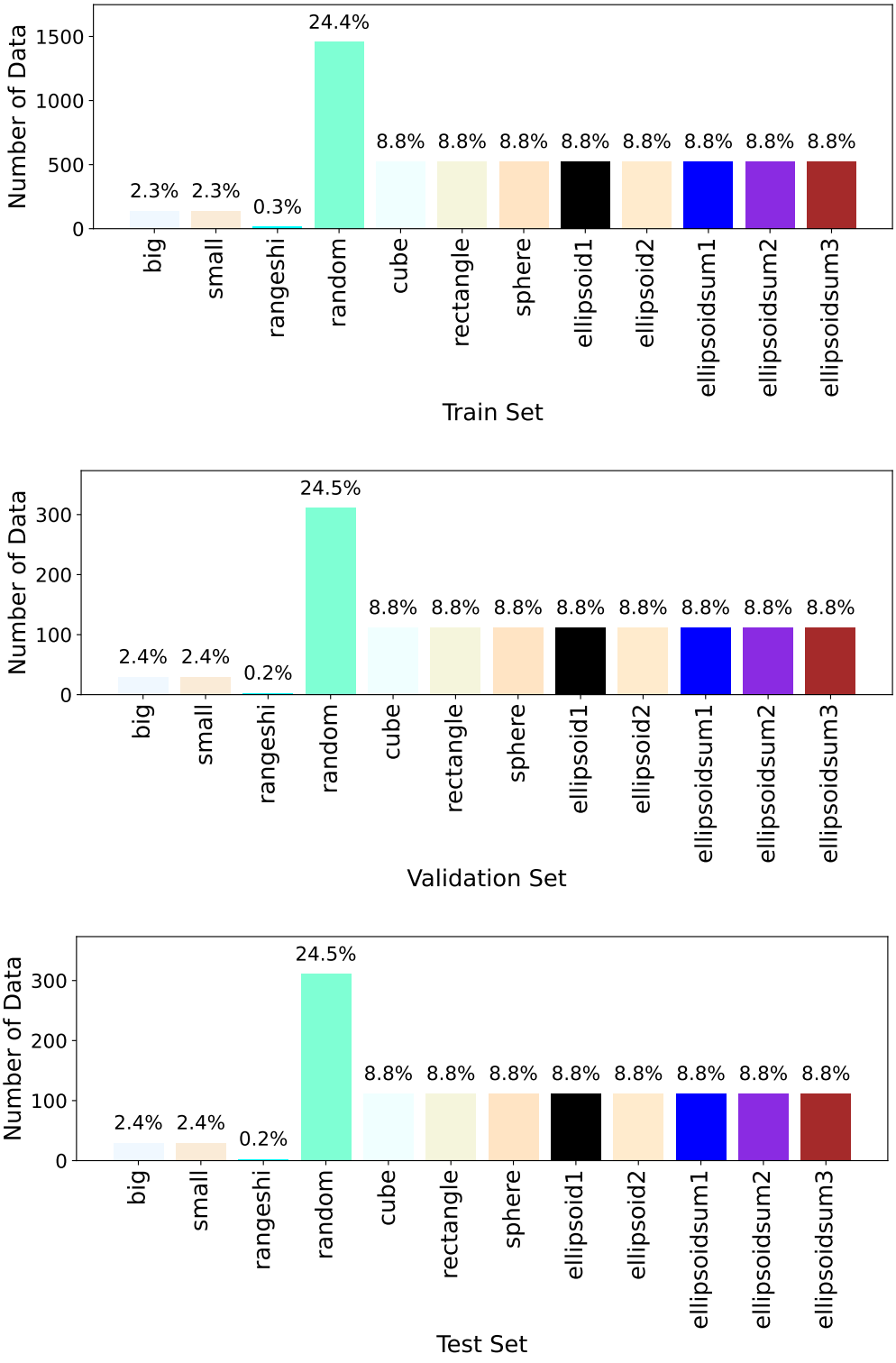

Figure 19: Complete data set: Histogram of the training, validation, and test sets displaying the number and percentage for each type of CEM

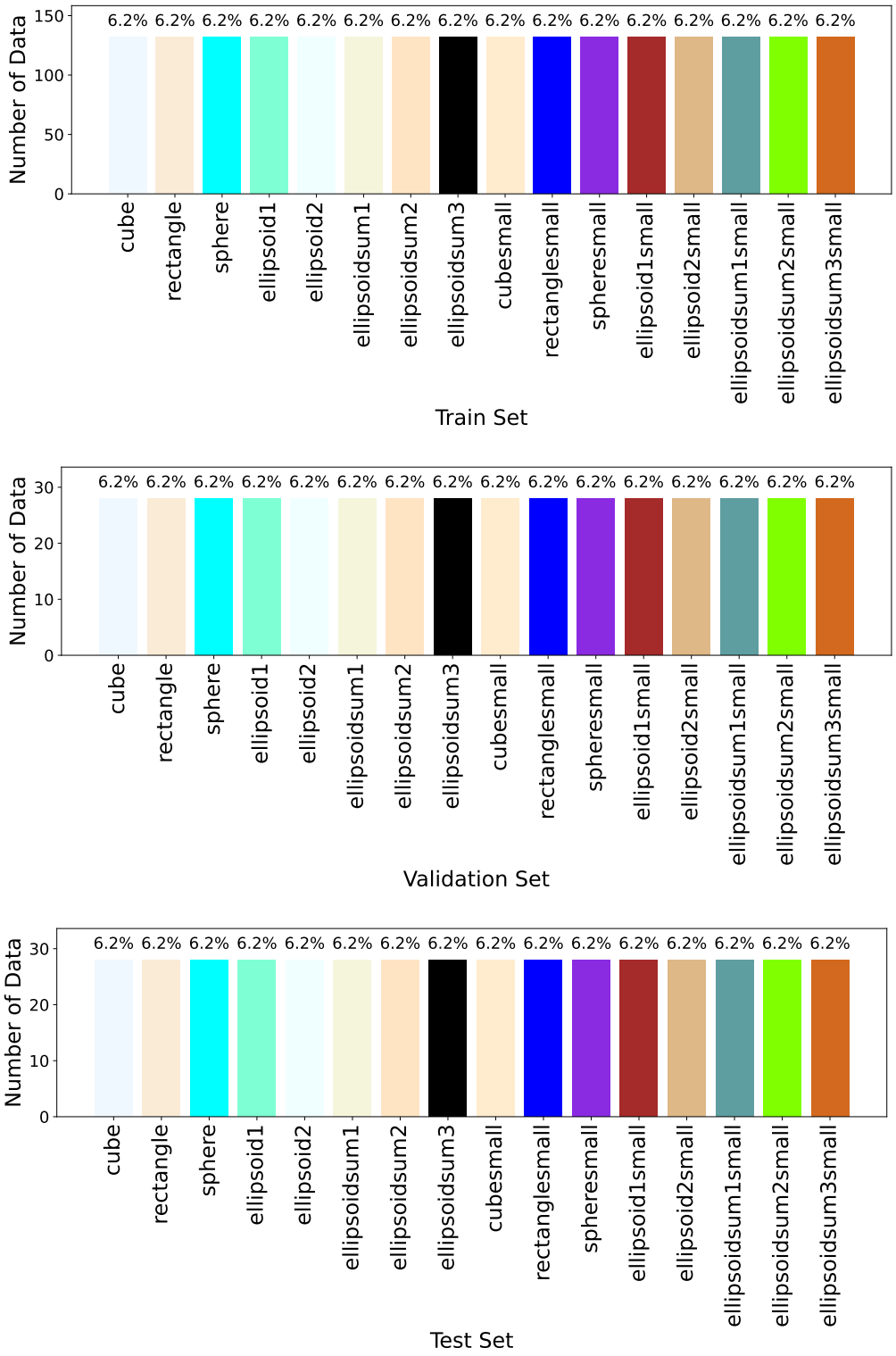

Figure 20: Group V: Histogram of the training, validation, and test sets displaying the number and percentage for each type of CEM

Chapter 7

Training of the AI model with different training data sets and results comparison with validation data sets

7.1 First training of the 4 data groups and validation results

The AI model is trained on four data sets composed of the 2 groups R and S of the first acquisition separately, on the complete data set of the first acquisition and on Group V. Groups R and S are trained separately to understand the contribution of each one to the validation results as they are very different. Moreover, one of the two groups might show better results than the other one, or the grouping of both is needed to obtain the best predictions which is why the model is also trained on the complete set. The data of Group V is a variation of group S with more variability. The model is also trained on this data set. Normalization of the data is already included in the AI model while data augmentation will be used later on.

7.1.1 Group R

Group R of the dataset comprises the 2 500 CEMs built randomly, mainly with small values, mainly with big values, and like range shifters. Those CEMs are not designed to obtain a spread-out Bragg peak.

The MSE loss function of the model as a function of the number of epochs for group R is displayed in Figure 21. From the 1^{st} to the 5^{th} epoch, the MSE value decreased sharply for both the training and the validation sets. The decrease in both curves means that the model learns something. A plateau appears around an MSE equal to 33.4 from the 6^{th} epoch to the 100^{th} epoch and indicates that the model stopped learning new information. It can happen if the loss function has a local minimum and the gradient is stuck in it and can not descend further. From the 90^{th} epoch, the model starts to overfit meaning that the loss value of the validation set starts to increase while the loss value of the training set decreases. This means that the model learns patterns that are specific to the training data set and not real global physics. The optimal number of epochs after which the model should stop learning is thus 100 meaning right before the overfitting of the model. The mean MSE at this epoch is around 32.6.

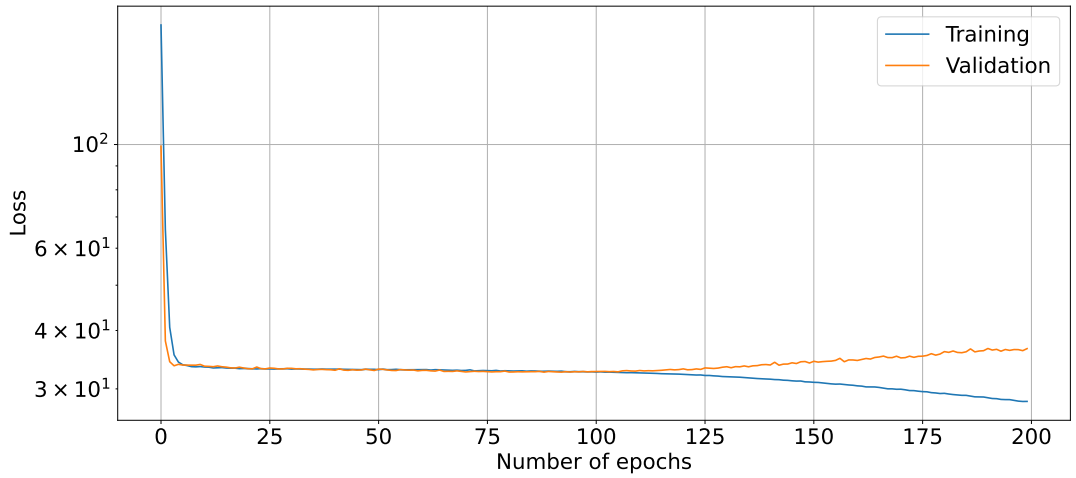

Figure 21: Group R: Losses as a function of the number of epochs for the training and validation sets

In Figure 22, the 10^th slice of the input (dose map) and a comparison between the true outputs and the predicted outputs (CEMs) of the AI model for the validation set is displayed. The first column is composed of a 10^th slice in the depth of the input dose map which corresponds to the slice in the middle of the 20x20x20mm target volume. Without any CEM, the majority of the dose stops at the 20^th slice, and with a CEM with big values, the majority of the dose would be around the 1^st slice. This explains why only a few dose concentration is found in the second case with a big CEM while a lot of dose is found in the third case with a small one.

The true and predicted outputs are respectively in the second and third columns. The last column displays the loss function for each pixel and a global MSE value. Four cases are studied: a CEM built like a range shifter whose height equals 5 mm, a CEM whose values are randomly selected in a range of big values [16,25], a CEM whose values are randomly selected in a range of small values [5,12] and a CEM randomly built.

The results of the model are especially good for the first case where similarities are clearly observed between the true and predicted output images. Moreover, the loss function for each pixel is relatively low and the global MSE value is equal to 1.36. The MSE pixel by pixel shows the specific pixels that were not well predicted. As for the second and third cases, it can be seen that the predicted outputs seem like means of the elements of the true outputs. Indeed, for the second case, the predicted output elements are mostly around 20 (mean big 138 = 20.5) and for the third case, the predicted output elements are mostly around 10 (mean small 288 = 8.5).

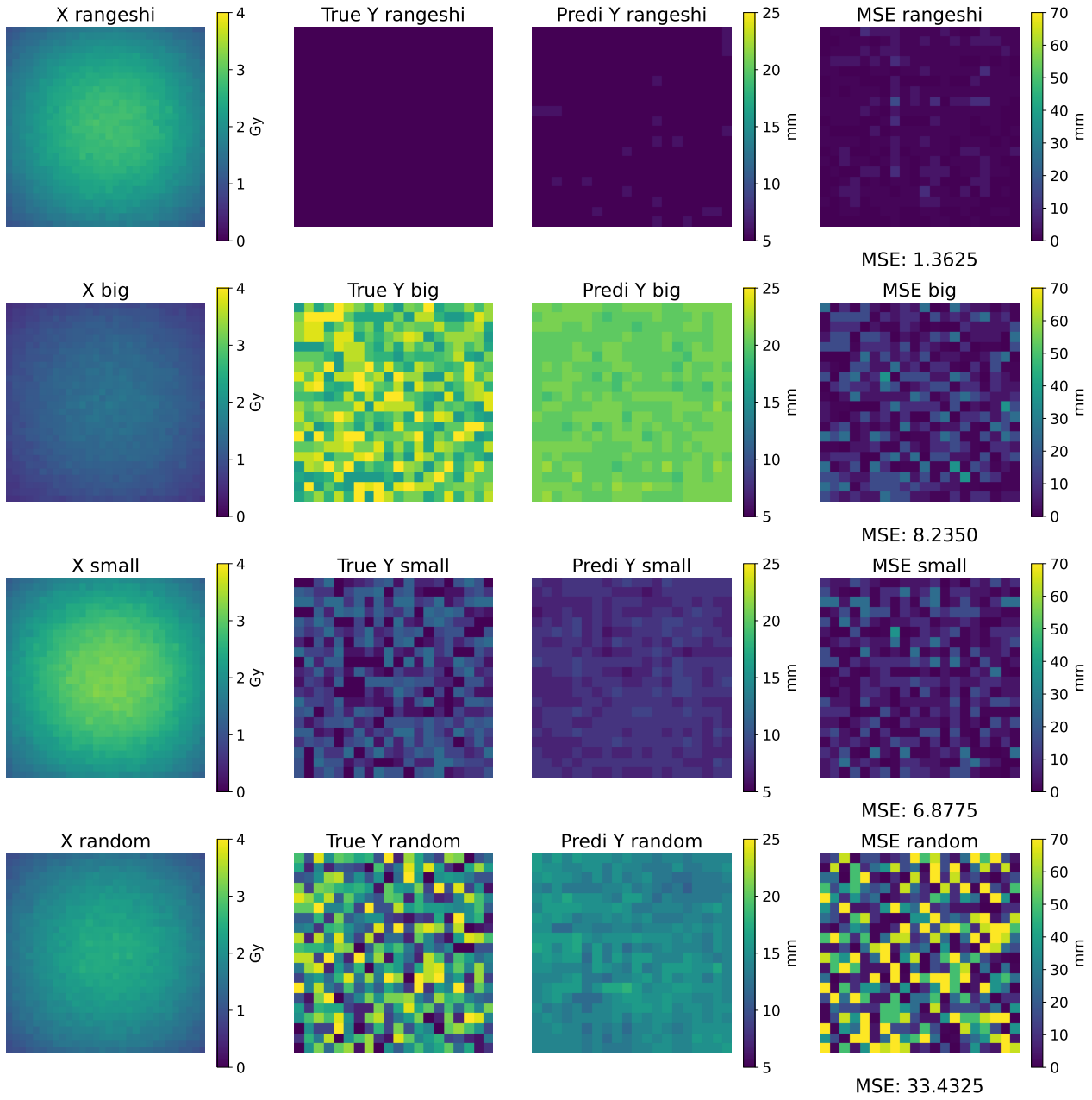

Figure 22: Group R: Input slice, Comparison between the true and predicted outputs of the validation phase, Loss function pixel by pixel and global MSE

The MSE pixel by pixel shows more differences between the outputs and the global MSE is equal to 8.23 and 6.87 for the second and third case respectively. The lase case is the worst of them all with high MSE values pixel by pixel and globally with an MSE of 33.43. It is also seen visually between the outputs. In this case, the predicted output also equals the mean of the true output elements (mean = 15). As there is more variability in the random case (5;25) and the predicted output equals the mean of the true output, the predicted output is the worst one for random cases. The range shifter case CEMs are the best predicted as there is no variability at all. It can be seen that no specific parts of the

20x20 matrix were more badly predicted than the others be it the edges, the corners, or the center for all cases.

In Table 3, the minimum, maximum and mean values of MSE for each type of CEMs of group R are specified. These values confirm the observations made in Figure 22 where the range shifter CEMs are the best-predicted ones, the random CEMs are the worst-predicted ones and the big and small CEMs are in the middle. Overall, the mean MSE for group S equals 33.16 which is consistent with the loss curves obtained for this group.

	Min MSE	Max MSE	Mean MSE
Rangeshifter	0.57	1.36	1.08
Big	1.05	37.82	16.16
Small	2.25	39.37	18.90
Random	30.95	42.25	36.49
Global	0.57	42.25	33.16

Table 3: Group R: Minimum, maximum, and mean values of MSE for each type of CEMs

The mean MSE pixel by pixel of the Group R validation set is displayed in Figure 23. The error is overall well distributed and no particular region has a smaller error. Values are also in accordance with the global MSE of 33.16 of the validation set.

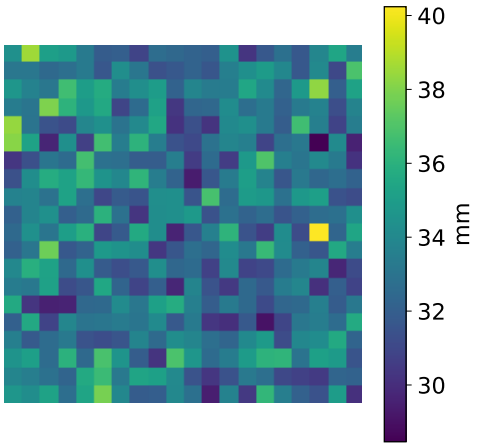

Figure 23: Mean MSE pixel by pixel of the Group R validation set

7.1.2 Group S

Group S of the dataset comprises the 6 000 CEMs built based on the eight CEMs optimized for the eight geometric shapes namely a cube, a rectangle, a sphere, two ellipsoids, and three sums of ellipsoids. Those CEMs are designed so that a spread-out Bragg peak occurs in the geometric shapes just like in real cases. The MSE loss function of the model as a function of the number of epochs for group S is displayed in Figure 24. The loss

function decreases exponentially from the 1^{st} to the 20^{th} epoch for both sets and reaches a plateau. The plateau occurs around an MSE equal to 0.65 which is very small meaning that the model seems to learn a lot.

The model does not overfit which might be explained by too little difference between the training and validation sets. The data might be too similar which makes the model less robust. There is a possibility that the model learns specific patterns that do not generalize well to new data. It can be observed that the validation curve has some bumps which are more frequent during the first dozens of epochs and the phenomenon fades a bit afterwards. This phenomenon of noise can occur when the model is complex with several layers and a lot of parameters to update like for the U-net architecture in this case. It can also be due to the mini-batch gradient descend. The noise does not necessary mean that there is a problem with the training of the model especially since its architecture is complex. What is important is to look at the general tendency of the validation curve. To gain in computation time while keeping a very low MSE value, the optimal number of epochs after which the model should stop learning is 90. The mean MSE at this epoch is around 0.64.

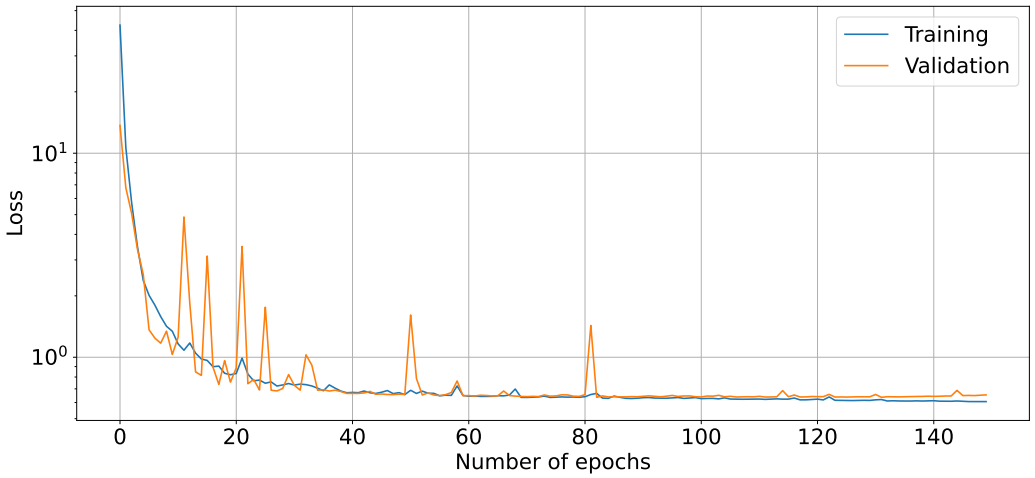

Figure 24: Group S: Losses as a function of the number of epochs for the training and validation sets

In Figure 25, 4 cases are studied: CEMs designed for target volumes whose shapes are a cube, a rectangle, a sphere, and a sum of three ellipsoids (ellipsoids of the same size but perpendicular to each other). In the first column, the 10^th slice shows a pretty high uniform dose concentration of around 3 Gy in the four cases as wanted with a SOPB. However, the resolution is not high enough to distinguish significant differences between them in terms of shape. Moreover, the hypothesis made over the equal spot weights does not permit a clear SOBP in the target volume.

The four cases have excellent results with high visual similarities between the true and predicted outputs and a low MSE both globally and pixel by pixel. Indeed all global MSE are around 1 and mostly all pixels were predicted correctly. Those results are consistent with the loss curves obtained for this group. The model manages to understand the boundaries of the CEMs very accurately in the three last cases where the meaningful part of the CEM does not cover the whole 20x20mm matrix. This means that the model linked the boundaries of the dose maps and those of the CEMs.

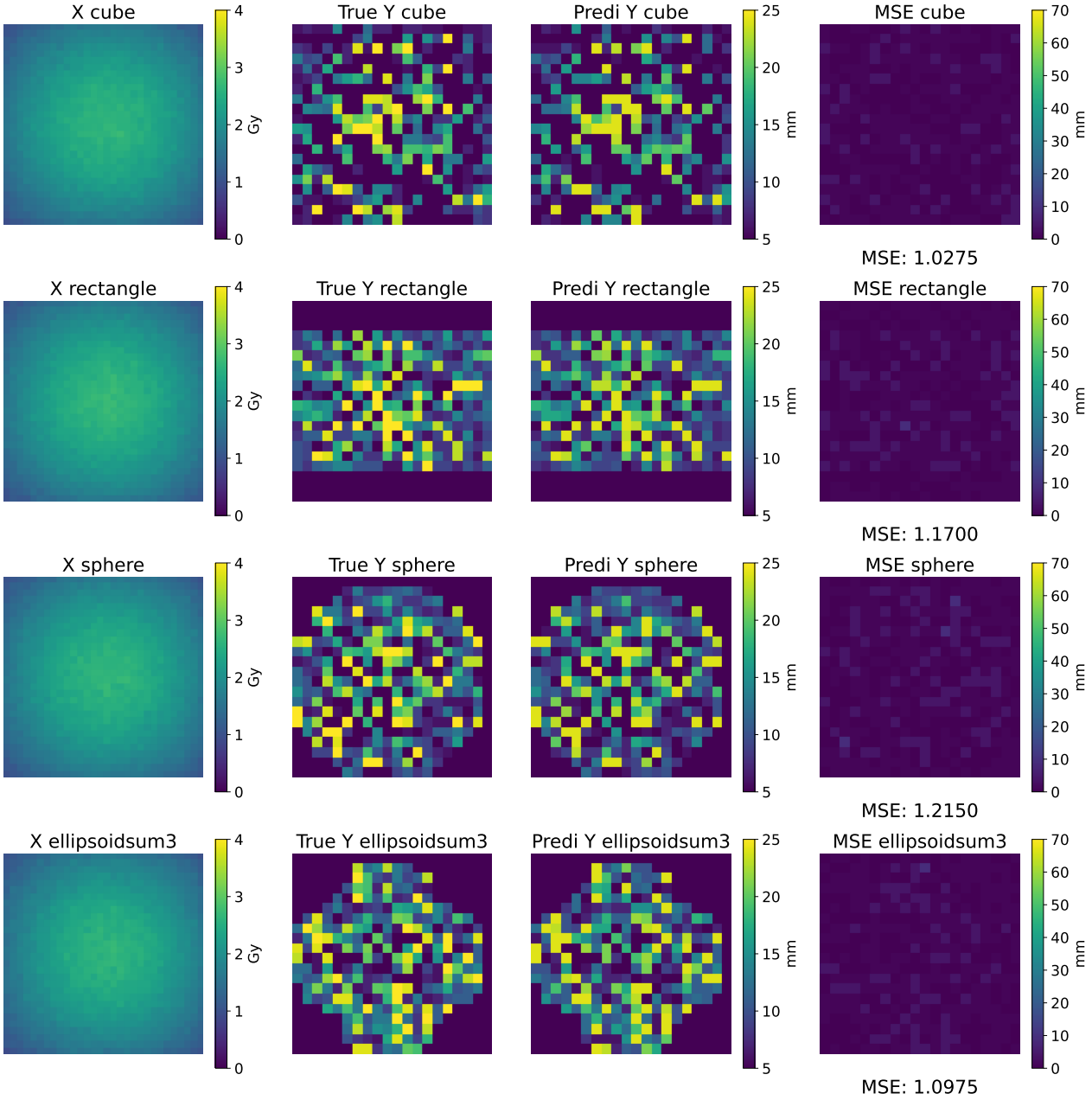

Figure 25: Group S: Input slice, Comparison between the true and predicted outputs of the validation phase, Loss function pixel by pixel and global MSE

In Table 4, the minimum, maximum, and mean values of MSE for each type of CEMs of group S are specified. The TABLE matches completely with the previous observations of the loss curves and output comparison. All types of CEM of group S give good predictions with an MSE around 1.

	Min MSE	Max MSE	Mean MSE
Cube	1.00	1.33	1.11
Rectangle	1.02	1.25	1.14
Sphere	0.98	1.27	1.13
Ellipsoid 1	0.87	1.10	0.96
Ellipsoid 2	0.74	0.97	0.84
Ellipsoid sum 1	0.81	1.10	0.96
Ellipsoid sum 2	0.93	1.18	1.03
Ellipsoid sum 3	0.83	1.25	1.10
Global	0.74	1.33	1.03

Table 4: Group S: Minimum, maximum, and mean values of MSE for each type of CEMs

The mean MSE pixel by pixel of the Group S validation set is displayed in Figure 26. Some particular regions have a smaller error. Those are where the CEM value was equal to the minimum value of 5mm meaning where the part of the CEM was not defined by the geometric shape of the target volume. This could pinpoint the fact that the Neural Network understands well the edges of the dose maps and makes a link with the edges of the meaningful part of the CEM. The error is overall well distributed where the meaningful part of the CEM is defined. Values are also in accordance with the global MSE of 1.03 of the validation set.

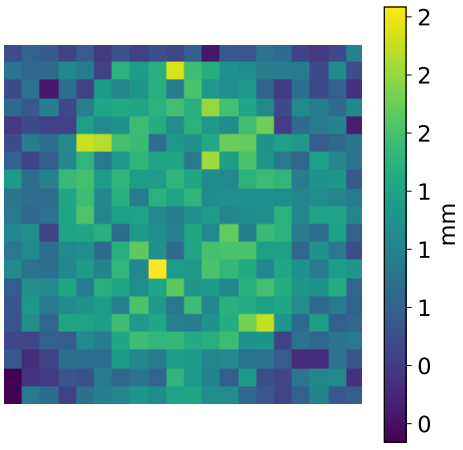

Figure 26: Mean MSE pixel by pixel of the Group S validation set

7.1.3 First data acquisition: Complete data set

The complete dataset comprises the 2 500 CEMs of group R built randomly, mainly with small values, mainly with big values and like range shifters, and the 6 000 CEMs of group S built based on the eight CEMs optimized for the eight geometric shapes. The MSE loss function of the model as a function of the number of epochs for the complete dataset is displayed in Figure 27. The loss curves seem to be a mix of both previous R and S groups. Both train and validation curves decrease exponentially at first to reach a plateau around an MSE equal to 10. For group R the plateau occurs at an MSE of around 33 while for the S group, it was around 0.7. There are also bumps in the validation curve as in the group S case. Overfitting most certainly occurs at a bigger number of epochs due to Group R data. The model could stop learning at epoch 100 where the MSE at this epoch is around 10 as the model does not improve further after it.

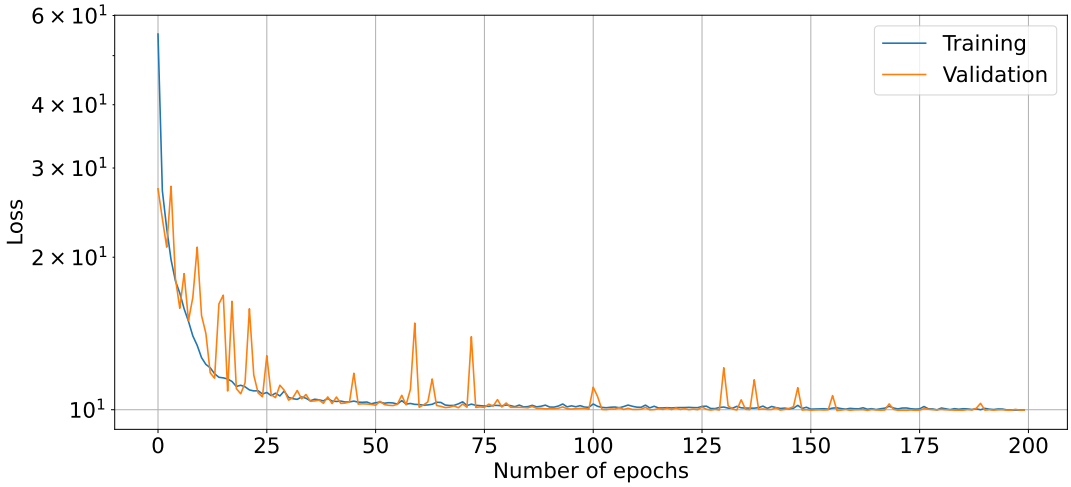

Figure 27: Complete data set: Losses as a function of the number of epochs for the training and validation sets

In Figure 28, 4 cases are studied: a CEM whose values are randomly selected in a range of big values [17,25], a CEM randomly built and CEMs designed for target volumes whose shapes are a cube, and a sum of three ellipsoids. The two first cases have predicted outputs whose values seem like means of the elements of the true outputs. Indeed, the predicted output elements are mostly around 20 (mean = 21) and 16 (mean = 15) for the first and second cases respectively. The first case shows better results than the second one with a global MSE equal to 7.2 and 32.7 respectively. This could be explained by a higher variability of element values in the second case leading to worst predictions. The two last cases have excellent results with high visual similarities between the two outputs. The global MSE of those cases are around 1 and mostly all pixels were predicted correctly as they are dark blue mostly everywhere.

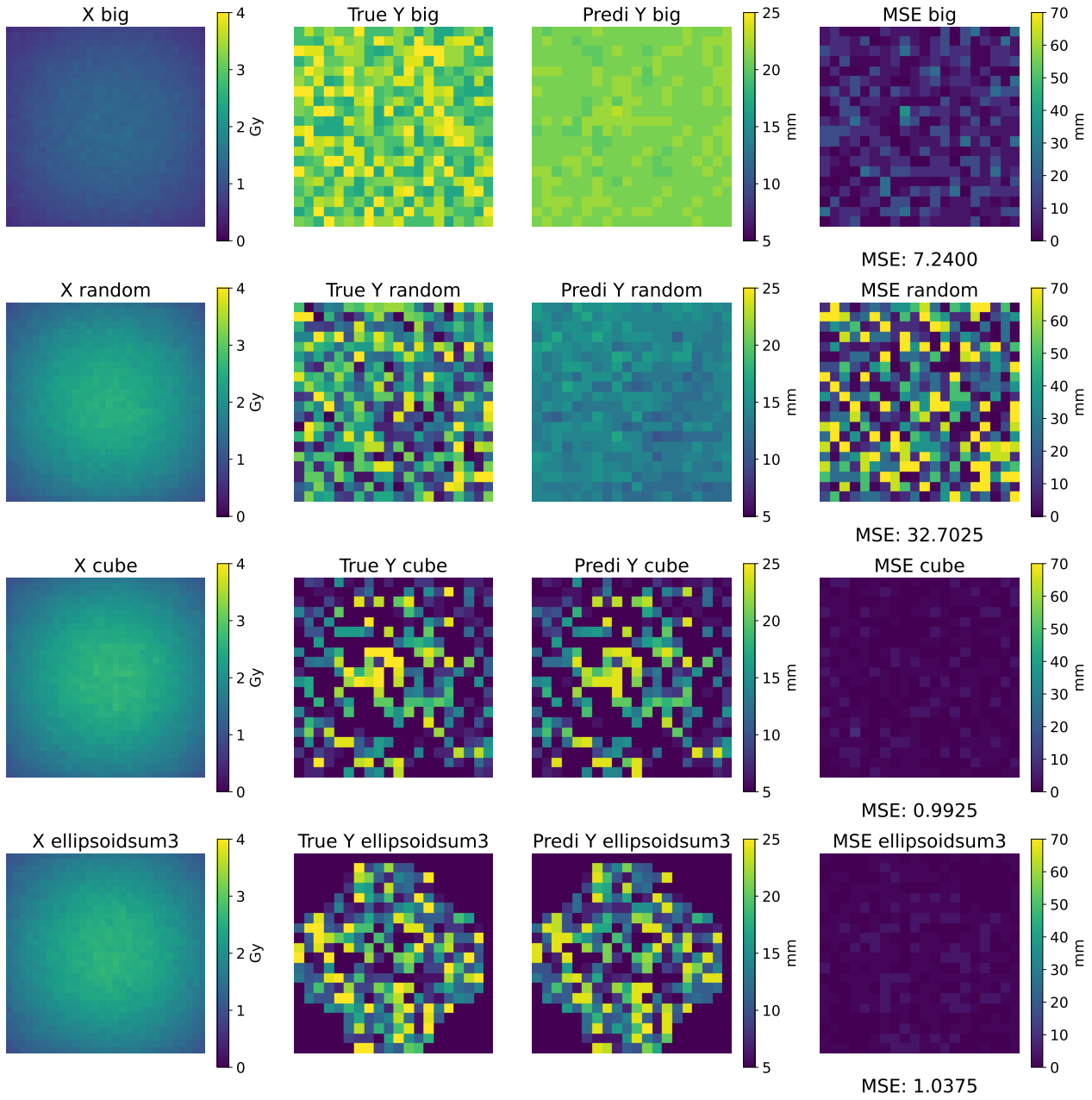

Figure 28: Complete data set: Input slice, Comparison between the true and predicted outputs of the validation phase, Loss function pixel by pixel and global MSE

In Table 5, the minimum, maximum and mean values of MSE for each type of CEMs of the complete data set are specified. The TABLE matches with observations made previously based on Figure 28. All CEMs based on geometric shapes have an MSE of around 1. CEMs like range shifters and whose values are mostly big or small seem to have a smaller MSE when part of the complete data set than of Group R with an MSE of 0.8, 1204, and 13.25 compared to 1.08, 16.16, and 18.90 respectively. However, randomly build CEMs have an MSE of 36.6 when being part of both the complete data set or Group R.

	Min MSE	Max MSE	Mean MSE
Rangeshifter	0.43	1.15	0.80
Big	0.72	30.11	12.47
Small	1.24	38.56	13.25
Random	28.45	42.58	36.42
Cube	0.88	1.21	1.05
Rectangle	0.85	1.18	1.02
Sphere	0.96	1.26	1.06
Ellipsoid 1	0.76	40.07	1.28
Ellipsoid 2	0.58	0.86	076
Ellipsoid sum 1	0.75	0.99	0.88
Ellipsoid sum 2	0.86	1.12	1.00
Ellipsoid sum 3	0.84	1.14	1.01
Global	0.43	42.58	10.24

Table 5: Complete data set: Minimum, maximum and mean values of MSE for each type of CEMs

The mean MSE pixel by pixel of the complete first data acquisition validation set is displayed in Figure 29. The error is overall well distributed and no particular region has a smaller error. Values are also in accordance with the global MSE of 10.14 of the validation set. Even though Group R is composed of 2 500 pairs compared to 6 000 for Group S, The mean MSE distribution is mostly dictated by Group R. This is explained by an MSE of Group S equal to 1 and an MSE of Group R around 30.

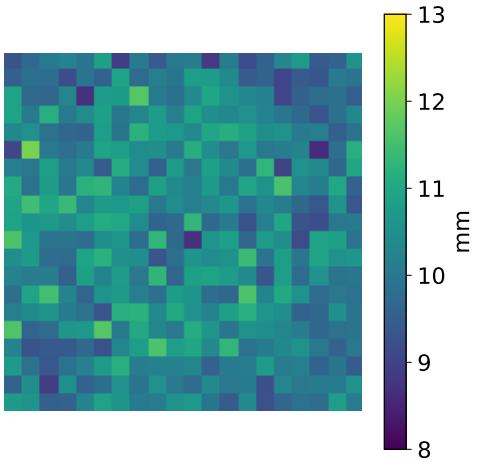

Figure 29: Mean MSE pixel by pixel of the complete data validation set

7.1.4 Group V

Group V comprises the 3007 CEMs built based on the sixteen CEMs optimized for the eight geometric shapes of two different sizes each, namely a cube, a rectangle, a sphere, two ellipsoids, and three sums of ellipsoids. Those CEMs are designed so that a spread-out Bragg peak occurs in the geometric shapes just like in real cases. Compared to Group S, the modified CEMs have more variability as the values added to each element of the CEM are chosen uniformly randomly in the range [-7,7] instead of [-2,2]. Moreover, each geometric shape is drawn in two sizes instead of one.

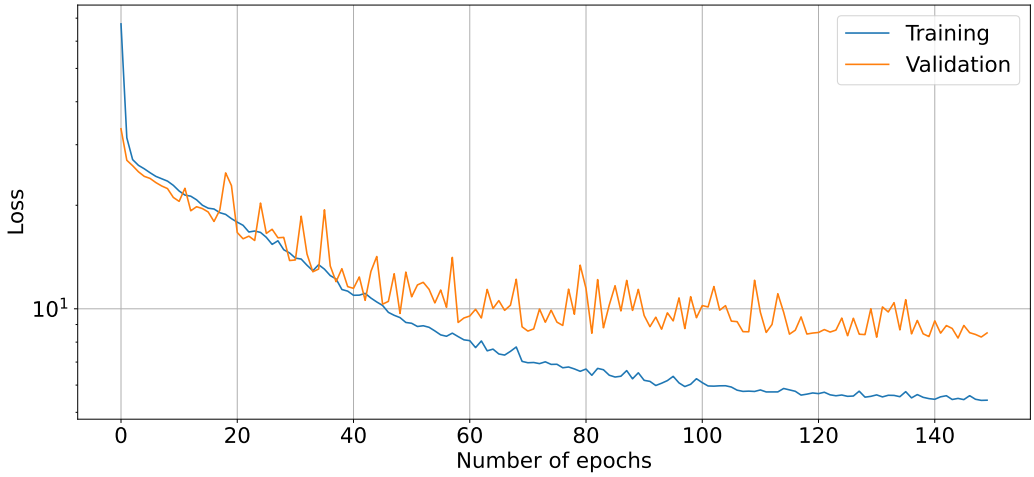

Figure 30: Group V: Losses as a function of the number of epochs for the training and validation sets

The MSE loss function of the model as a function of the number of epochs for Group V is displayed in Figure 30. The loss curves of both the training and validation set decrease to the 70^{th} epoch. After epoch 70, the training curve continues to decrease while the validation curve remains constant around an MSE value of 8.9. The big gap between the training and the validation losses indicates that there is possibly too little data in the training set. The validation curve that stops decreasing while the training curve continues to go down indicates overfitting. This overfitting proves that Group V has more variability and that the training and validation sets are different enough which was not the case for Group S which never overfitted. However, this group is smaller by a factor of 2 compared to Group S. The optimal number of epochs after which the model should stop learning is 150, and the mean MSE at this epoch is around 8.5.

In Figure 31, 4 cases are studied: CEMs designed for target volumes whose shapes are a cube, a rectangle, a small sphere, and a small ellipsoid. The four cases have good results with obvious visual similarities between the true and predicted outputs. Even though it seems at first glance that the model struggles to predict the high values, when looking at the MSE pixel by pixel, the errors are not especially higher for those specific pixels.

The global MSE is quite low for all cases but higher than those of Group S. The model predicts better the pixel equal to the minimal value of 5mm, i.e. where the target volume is not defined. As the last two cases have smaller target volumes, their global MSE is smaller. All those results are consistent with the loss curves obtained for this data set. The model manages to link the boundaries of the dose maps and those of the CEMs as both shape sizes are predicted well.

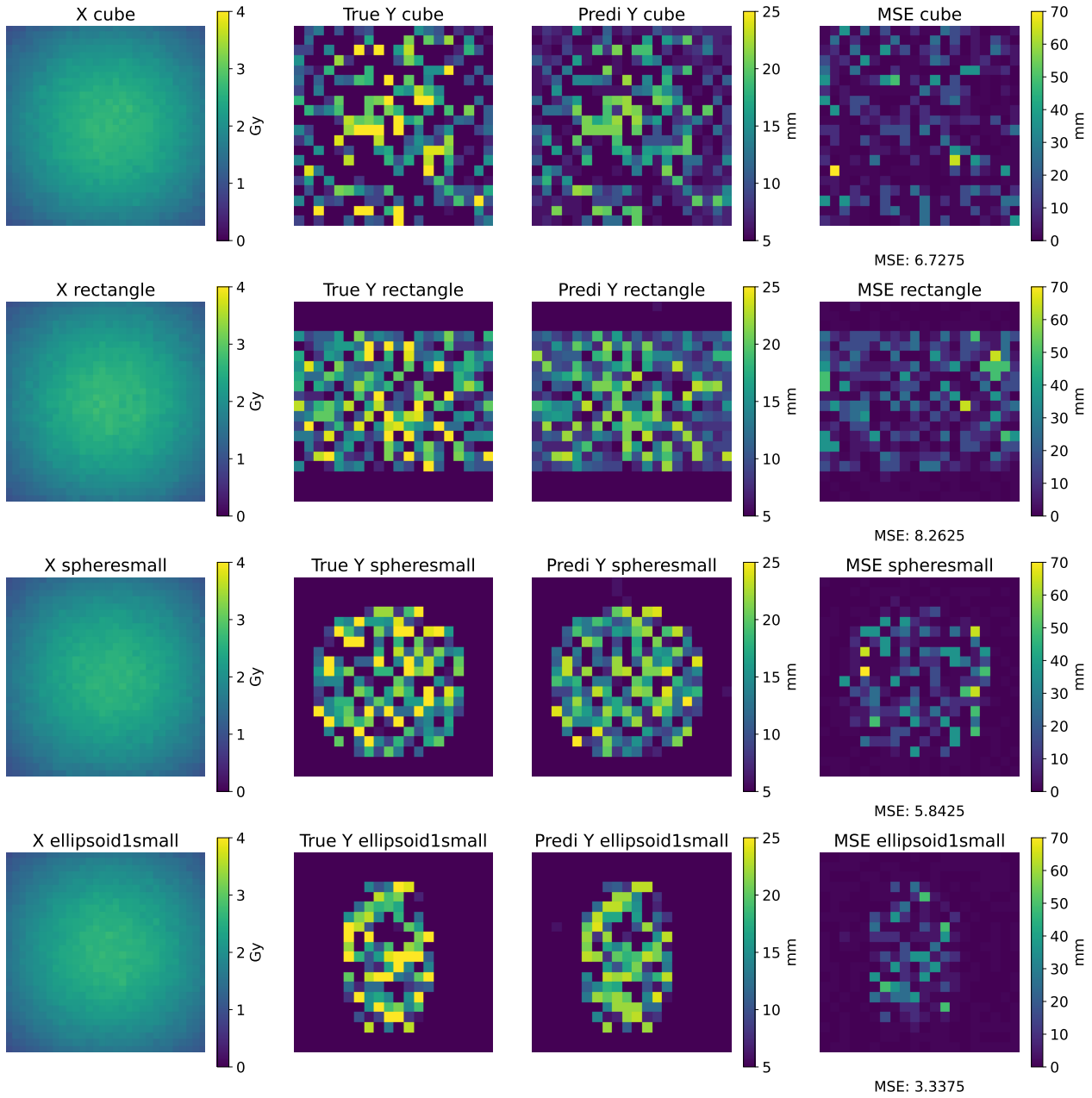

Figure 31: Group V: Input slice, Comparison between the true and predicted outputs of the validation phase, Loss function pixel by pixel and global MSE

In Table 6, the minimum, maximum, and mean values of MSE for each type of CEMs of Group V are specified. The TABLE matches the previous observations of the loss curves and output comparison. The MSEs of the CEM designed for smaller geometric shapes are indeed smaller in general. The maximum MSE is sometimes big which shows that the data set is more variable in this data set than Group S. The global MSE of Group V is equal to 7.12 which is relatively low.

	Min MSE	Max MSE	Mean MSE
Cube	6.36	18.41	8.31
Rectangle	7.31	14.22	8.96
Sphere	2.24	69.14	12.18
Ellipsoid 1	5.06	13.00	6.44
Ellipsoid 2	3.19	17.92	5.12
Ellipsoid sum 1	5.60	45.26	12.95
Ellipsoid sum 2	6.29	33.26	8.18
Ellipsoid sum 3	6.99	9.84	7.53
Small Cube	6.86	43.67	9.40
Small Rectangle	4.20	7.06	4.98
Small Sphere	4.79	11.28	6.07
Small Ellipsoid 1	2.50	28.63	4.64
Small Ellipsoid 2	1.98	2.57	2.24
Small Ellipsoid sum 1	2.63	25.68	5.98
Small Ellipsoid sum 2	4.07	5.68	4.62
Small Ellipsoid sum 3	4.20	25.75	6.29
Global	1.98	69.14	7.12

Table 6: Group V: Minimum, maximum, and mean values of MSE for each type of CEMs

The mean MSE pixel by pixel of Group V validation set is displayed in Figure 32. Like Group S, the regions where the CEM value was equal to the minimum value of 5mm show a smaller error. In this case, those regions are more extended than group S because half of the pairs are designed for smaller target volumes. The error is overall well distributed where the meaningful part of the CEM is defined. Values also follow the global MSE of 7.12 of the validation set.

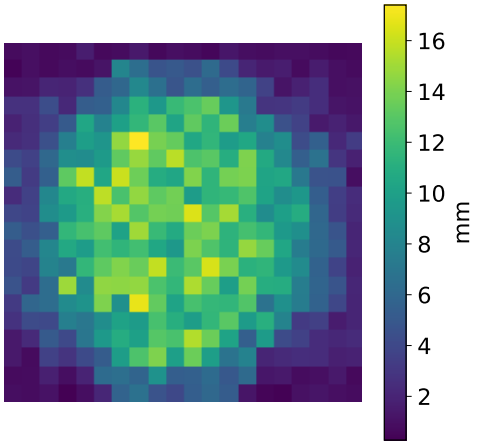

Figure 32: Mean MSE pixel by pixel of Group V validation set

7.1.5 Summary

Group S shows the best results in terms of loss curves, MSE, and visual similarities while Group R has the worst ones. The model does not learn physics when trained on Group R and predicted CEMs are equal to the mean of the true CEMs elements. The loss function chosen, which is the MSE, might not be the best one for Group R. Moreover, the data of Group R is very dissimilar and lacks common features. It is thus possible that the model struggles to learn meaningful relationships within the data. This could explain the high loss value for both the training and validation sets leading to inaccurate predictions.

As for Group S, the validation results are excellent. However, note that it is not possible to predict that the model works well on a completely new data set. The model might not be robust and the real physics might not be learned. Indeed there is the possibility that there is too little variability in this data set which would explain why the model does not overfit.

Adding Groups R and S together into a single data set does not improve the predictions of the model. Indeed the MSE and visual similarities are comparable to Group R and S results. The hypothesis of needing both groups to have better predictions happens to be false.

Group V is overfitting opposite of Group S which shows that the data are more variable. Its predictions are less accurate than Group S's but are not bad either. This model should be more robust due to more variability in its data.

7.2 Training of the model with data augmentation

The size of the data set for Group R, Group S, and Group V is multiplied by four with data augmentation. There is no need to test data augmentation on the complete data of the first acquisition as it has been highlighted that putting Groups R and S together did not improve the results. Each pair of CEMs and dose maps has been rotated by 90, 180, and 270° to increase artificially the number of data. The goal of increasing the number of data is to possibly improve the performance of the Neural Network and to show him more diversified data to increase his robustness. The loss functions of Groups R, S and V with data augmentation are displayed in Figures 33, 34 and 35 respectively.

What can be observed is that data augmentation on Group R did not improve the NN's performance and overfitting occurs around the same MSE of 32.8 at the 75^{th} epoch compared to when data augmentation was not introduced. This means that data augmentation was not sufficient to reduce the MSE and improve predictions. The data of Group R are probably too dissimilar which makes the relationship between the inputs and the outputs too complex for the model to understand.

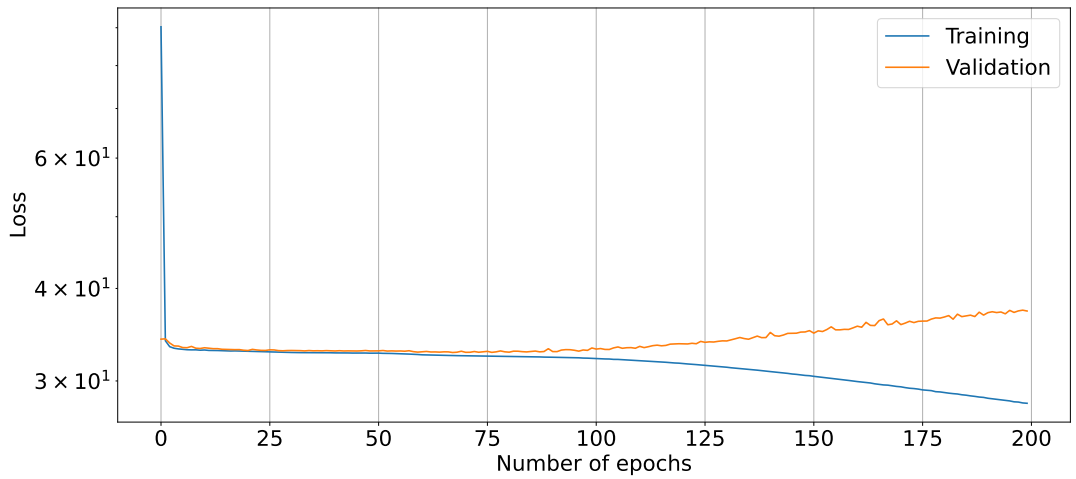

Figure 33: Group R: Losses as a function of the number of epochs for the training and validation sets with data augmentation

As for Figure 34, the model arrives at the same MSE value compared to when data augmentation was not introduced. The validation curve is smoother but decreases more slowly to reach the MSE of 0.65 at only the 140^{th} epoch. The noise reduction is probably due to an increase in the number of data and in the representativeness of the validation set. Overall, the results did not improve with the introduction of data augmentation.

When comparing Figures 30 and 35, the loss curves converge faster with data augmentation. However, the validation curve stagnates at the same MSE of around 8 like without data augmentation. Even an increase in the loss is observed from the 60^{th} epoch.

The predictions do not improve with data augmentation on Group V. The noise is not reduced either. There is still a gap between the loss curves indicating too little data in the training set. To conclude, it is unnecessary to increase the number of data artificially by performing rotations for any data set as the MSE is not reduced while the computation time is increased.

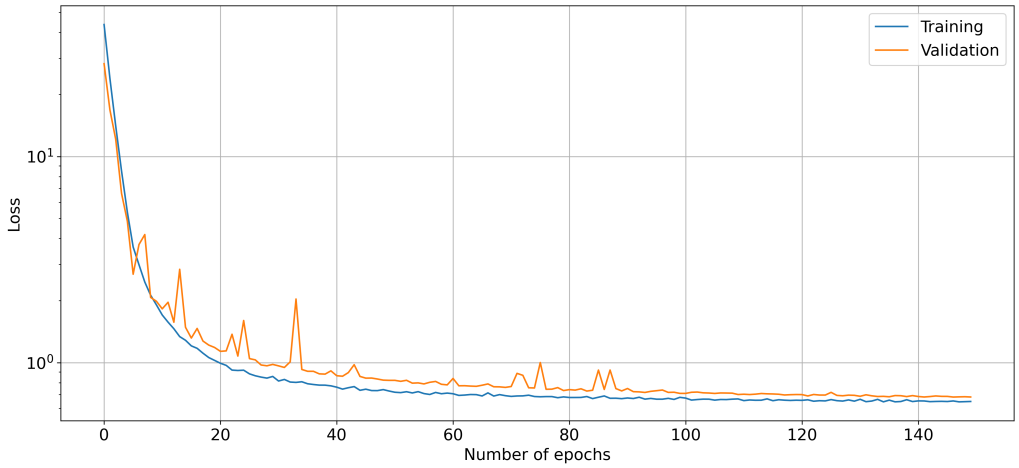

Figure 34: Group S: Losses as a function of the number of epochs for the training and validation sets with data augmentation

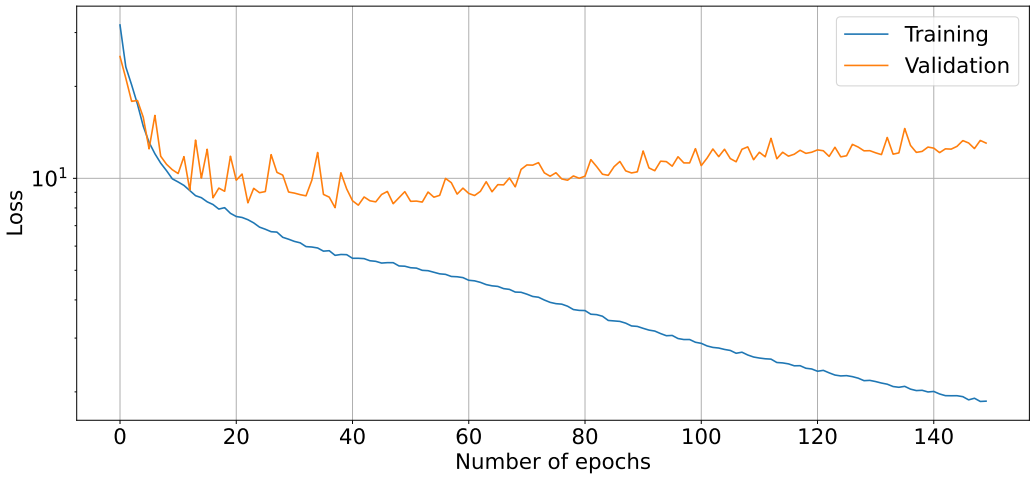

Figure 35: Group V: Losses as a function of the number of epochs for the training and validation sets with data augmentation

7.3 Hyper-parameters tuning

The model has been trained for the 4 groups of data with Group S and Group V being the two best along them. Indeed, putting Groups R and S together did not improve the results and Group R alone did not manage to understand the real physics leading to poor results. Thus, only the parameters of the model trained on Group S and on Group V will be optimized. The goal is to find the best parameters that decrease further the value of the MSE.

What is studied is the impact of three methods embedded in the structure of the model which are Batch normalization, dropout, and the method to downscale the feature maps namely strided convolution or max pooling. A parameter of the model's structure that influences the model's performance is the number of feature maps of the model. The parameters outside of the model structure that have an impact on the model's results are the batch size and the learning rate. The influence of those parameters is also studied in this section.

7.3.1 Group V

Number of feature maps

In Figure 36, the loss function for three different numbers of feature maps as a function of the number of epochs is displayed. The bigger the number of feature maps, the faster the model learns and converges. It is also true that the number of parameters to adjust during the training phase also increases with the number of feature maps hence increasing the computation time.

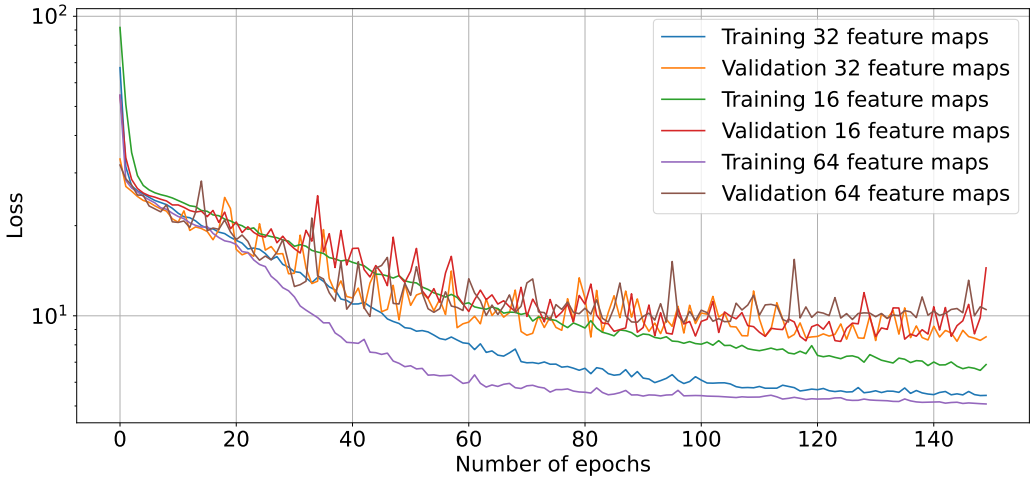

Figure 36: Group V: Losses as a function of the number of epochs for the training and validation sets and three different numbers of feature maps

The model convergence stops at different plateau values depending on the number of feature maps. Indeed, the plateau MSE value equals 10 for 64 feature maps which is

higher than the plateau value for 16 and 32 feature maps. For those number of feature maps, the model ends up at a same MSE value of 8.5. However, it is better to choose a number of feature maps equal to 16. Indeed, the tendency of the model to overfit increases with the number of feature maps and it is good for the model not to learn too quickly.

Batch size and learning rate

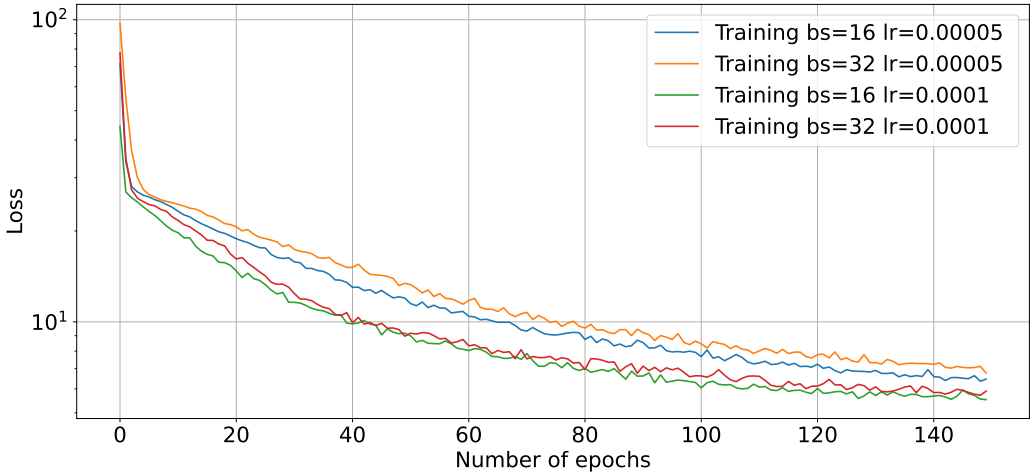

Figure 37: Group V: Losses as a function of the number of epochs for the training set and two different batch sizes and learning rates

In Figures 37 and 38, the loss function for four different batch sizes as a function of the number of epochs is displayed for the training and validation sets respectively. In practice, the batch size and the learning rate must be chosen together. Indeed, the best batch size might certainly be different for two different learning rates. As such, four combinations of batch sizes and learning rates are tested.

The model that converges the best is for a batch size of 32 and a learning rate of 0.0001 for the training phase. However, in the validation phase, the four combinations stop at a similar plateau. The learning rate of 0.0001 leads to more noisy loss curves which is to be avoided. The best compromise is to choose a batch size of 32 and a learning rate of 0.00005 to reduce the noise and shorten the computation time. Indeed, the bigger the batch size is, the shorter the computation time.

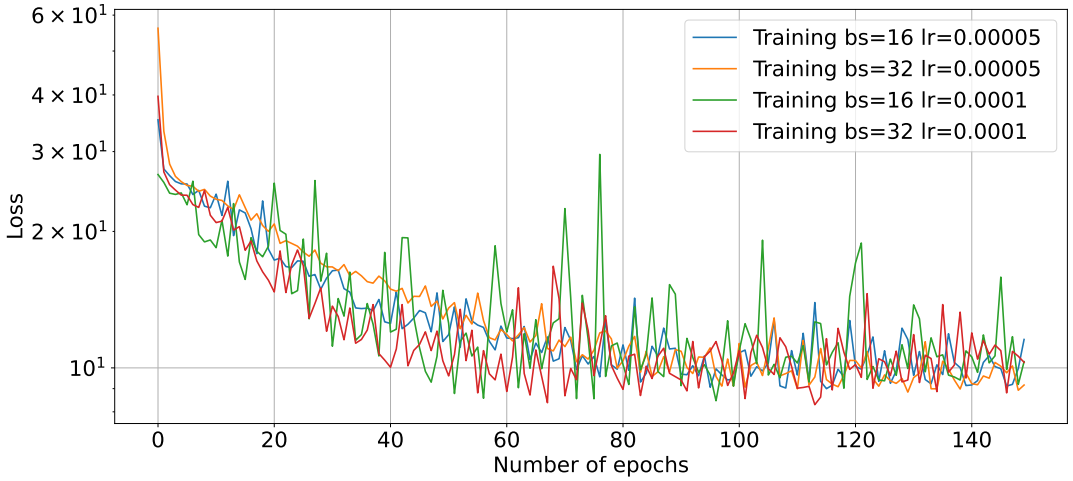

Figure 38: Group V: Losses as a function of the number of epochs for the validation set and two different batch sizes and learning rates

Dropout

As seen in Figure 39, dropout does not really have an effect on the results as the validation loss curves stop at the same plateau. This might be caused by the batch normalization method which makes regularization methods less necessary. Moreover, the noise seems to be more present with dropout and a smooth loss curve is better in theory. It is better to remove dropout from the model structure.

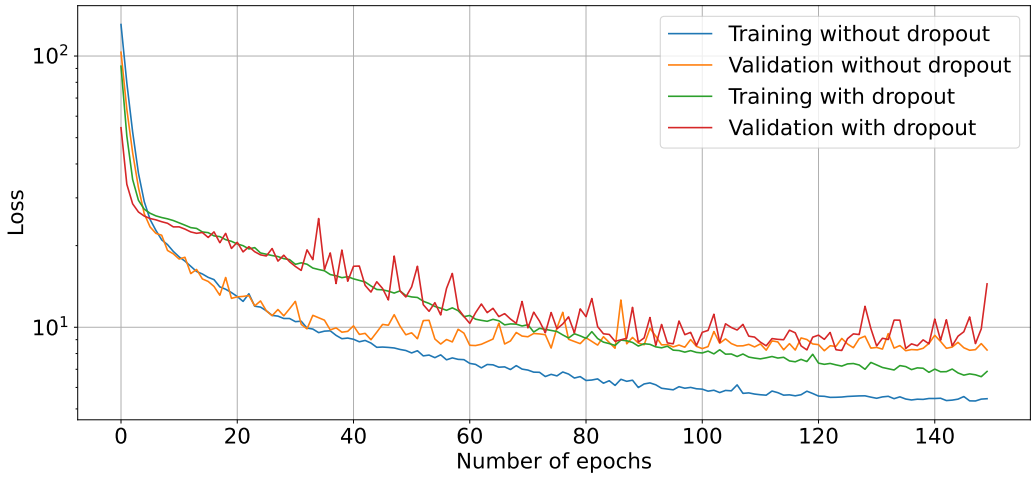

Figure 39: Group V: Losses as a function of the number of epochs for the training and validation sets with and without dropout

Batch normalization

In Figure 40, the loss function with and without batch normalization in the model as a function of the number of epochs is displayed. The model converges very well with batch normalization included in the model whereas the model never learns anything without BN. Having BN in the model is thus the best choice.

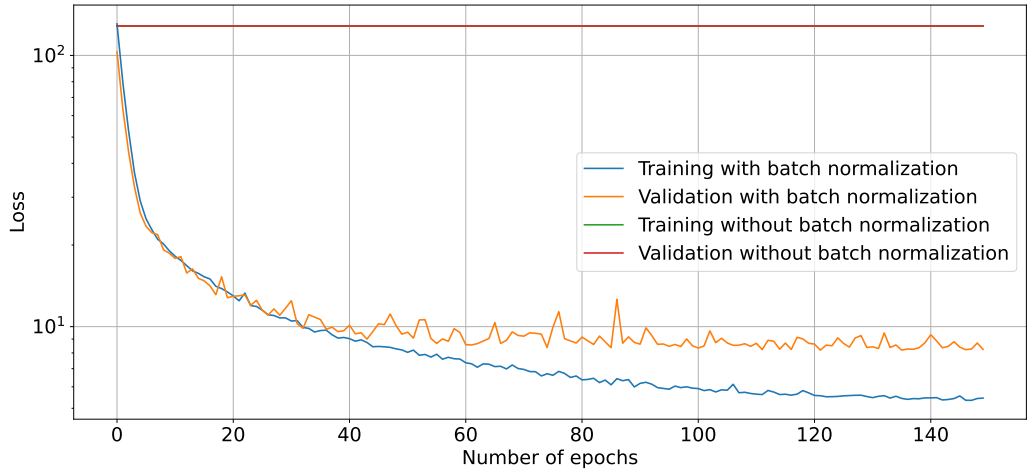

Figure 40: Group V: Losses as a function of the number of epochs for the training and validation sets with and without batch normalization

Strided convolution VS Max pooling

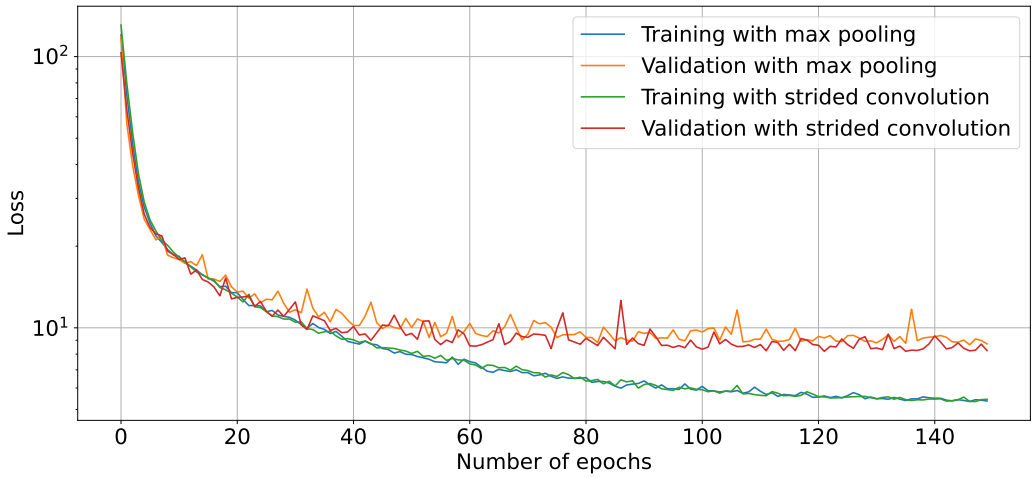

Figure 41: Group V: Losses as a function of the number of epochs for the training and validation sets with Max pooling or with Strided convolution

The validation curve decreases further with strided convolutions than with max pooling. Max pooling replaces local regions of the input feature map with the maximum value found within that region in order to decrease the feature map's size. The most prominent features are kept while less significant details are removed. By doing so, max pooling forced the NN to learn in a certain manner. On the other hand, strided convolution gives more freedom of learning to the Neural Network and increases the model's expressiveness ability. Strided convolutions tend to replace max pooling in several applications [22]. In this case, strided convolution gives the best results.

To conclude, the best parameters for the model trained with Group V are :

- a number of feature maps equal to 16
- a batch size equal to 32
- a learning rate equal to 0.00005
- Batch Normalization used
- Dropout unused
- Strided convolution used

The loss function of the model trained with Group S for its optimized parameters and methods is displayed in Figure 42.

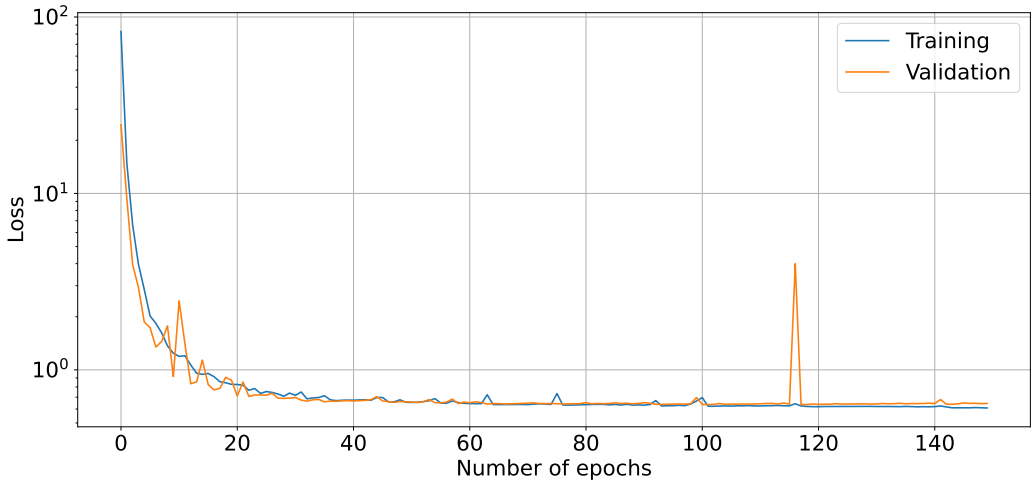

Figure 42: Group S: Losses as a function of the number of epochs for the training and validation sets with optimized parameters

Overall, the parameter optimization did not improve the model's performance greatly for Group V as the model lacks training set examples resulting in a plateau of the validation set MSE. The only method needed for the model's convergence is the use of batch normalization which was already used initially.

7.3.2 Group S

The hyperparameters optimization is also performed for the model trained on the Group S training data set. As the process is repetitive and very similar to the one performed on Group V, the loss curves are displayed in the Appendix chapter. The best parameters for the model trained with Group S are :

- a number of feature maps equal to 32
- a batch size equal to 32
- a learning rate equal to 0.00005
- Batch Normalization used
- Dropout unused
- Strided convolution used

The loss function of the model trained with Group V for its optimized parameters and methods is displayed in Figure 43. The gap between the training and validation loss curves remains even when parameters were optimized. The gap is likely to be due to too few training examples. This hypothesis is justified by the fact that 16 types of different CEMs with only 188 examples each with relatively high variability compose the data set. Compared to Group V, Group S is composed of only 8 different types of CEMs with 750 examples each with low variability in CEMs of the same types. This might explain why there is no gap for Group S and one for Group V.

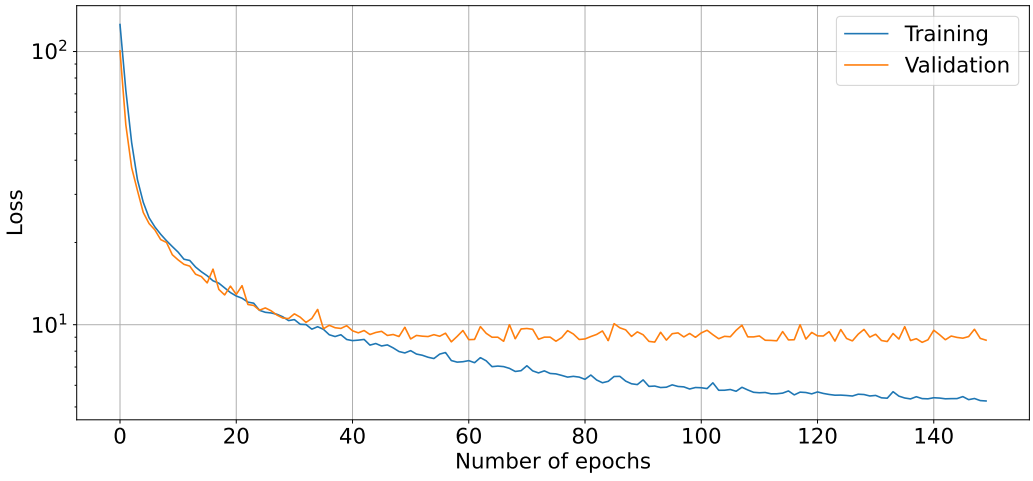

Figure 43: Group V: Losses as a function of the number of epochs for the training and validation sets with optimized parameters

Overall, the parameter optimization did not improve the model's performance greatly for Group S as the two main methods that decreased greatly the MSE was the use of

strided convolutions and batch normalization which were already used by the model initially.

Chapter 8

Test results of AI model

In this chapter, both models trained with Group S and Group V training sets will be tested with their respective test sets. It permits the evaluation of the model's performance on unseen data.

8.1 Testing of the model trained on Group S with Group S test set

The validation set was used to assess the temporary performance of the model and was used as feedback to modify the model. This trained model is now tested on the test set of the data. The test set is used to assess the final evaluation of the model's performance. The model has never seen it before. It gives an unbiased estimate of how well the model is likely to perform on unseen data. The test set also allows to know whether the model has the ability to make generalizations or if it only memorized the training set.

Figure 44 shows that the model obtains good results in terms of visual similarities, MSE pixel by pixel, and global MSE. The results are highly similar to the validation set's results. Table 7 has slightly smaller values compared to the validation set values meaning that the parameters optimization did not have a big impact on them. The mean MSE pixel by pixel also resembles the one for the validation set. All those observations indicate that the model has good performances and was well-trained for this type of data.

Note that the robustness of the model can not be ensured as there is a chance that the model performs well because the sets are too similar. Indeed, this hypothesis was proposed because the model did not show signs of overfitting in the losses as a function of the number of epochs. A future test that could be performed is to design a new geometric shape as the target volume with the same global size of 20x20x20 and create from it CEMs with the same variability as the other ones and test it on the model.

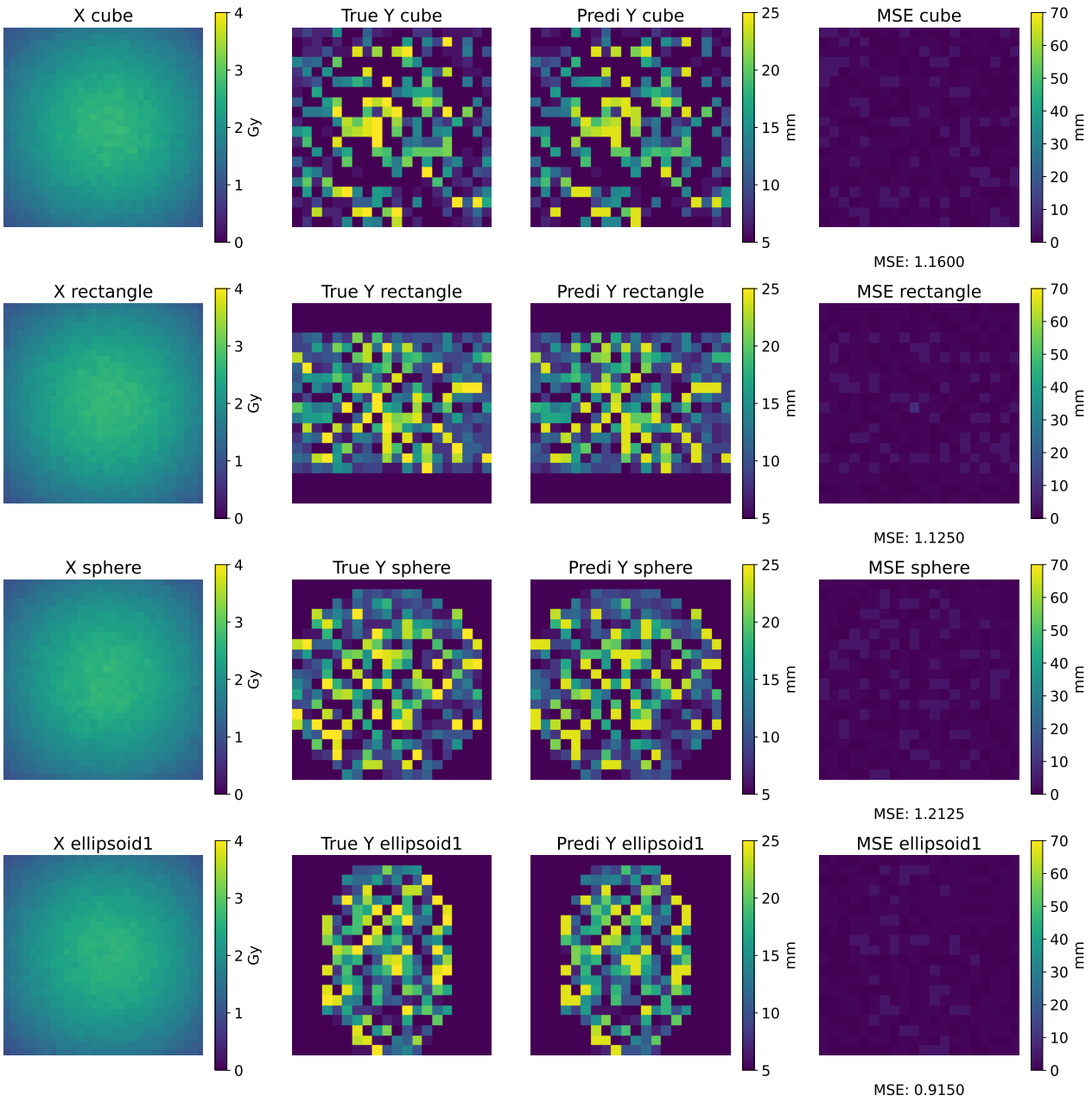

Figure 44: Group S: Input slice, Comparison between the true and predicted outputs of the test phase, Loss function pixel by pixel and global MSE

	Min MSE	Max MSE	Mean MSE
Global	0.69	1.34	1.01

Table 7: Group S: Minimum, maximum, and mean values of MSE

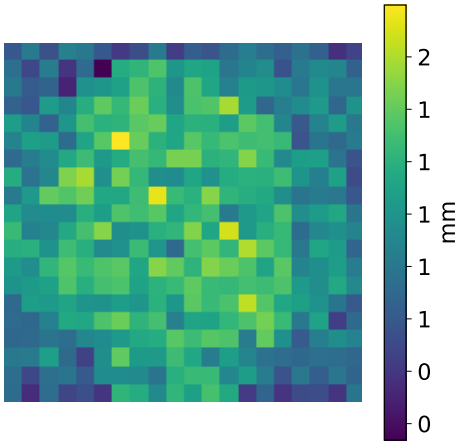

Figure 45: Mean MSE pixel by pixel of the Group S test set

8.2 Testing of the model trained on Group V with Group V test set

The trained model is tested on the test set of Group V and its results are displayed in Table 8 and in Figures 43 and 46. Firstly, the visual similarity between the true and predicted outputs is globally confirmed with small color differences highlighted by the MSE pixel by pixel. The global MSE is not very big and the values seem smaller for the test set than the validation set.

This observation is confirmed by the min, max, and mean MSEs of the whole data set that are reduced. Either the parameters optimization had an impact, either the test set data were more similar to the training set compared to the validation set, or both played a role in the MSE reduction.

The mean MSE pixel by pixel is also similar to the validation set's one with the difference that the values go up to 14mm compared to 16mm which confirms again that the MSE is smaller.

As the model showed overfitting, the model trained on Group V is not likely to have a good performance because of too similar data sets (possibly Group S). This model might thus be more robust to new data compared to Group S. Once again, a new geometric shape defined as the target volume could be designed to test the model's robustness.

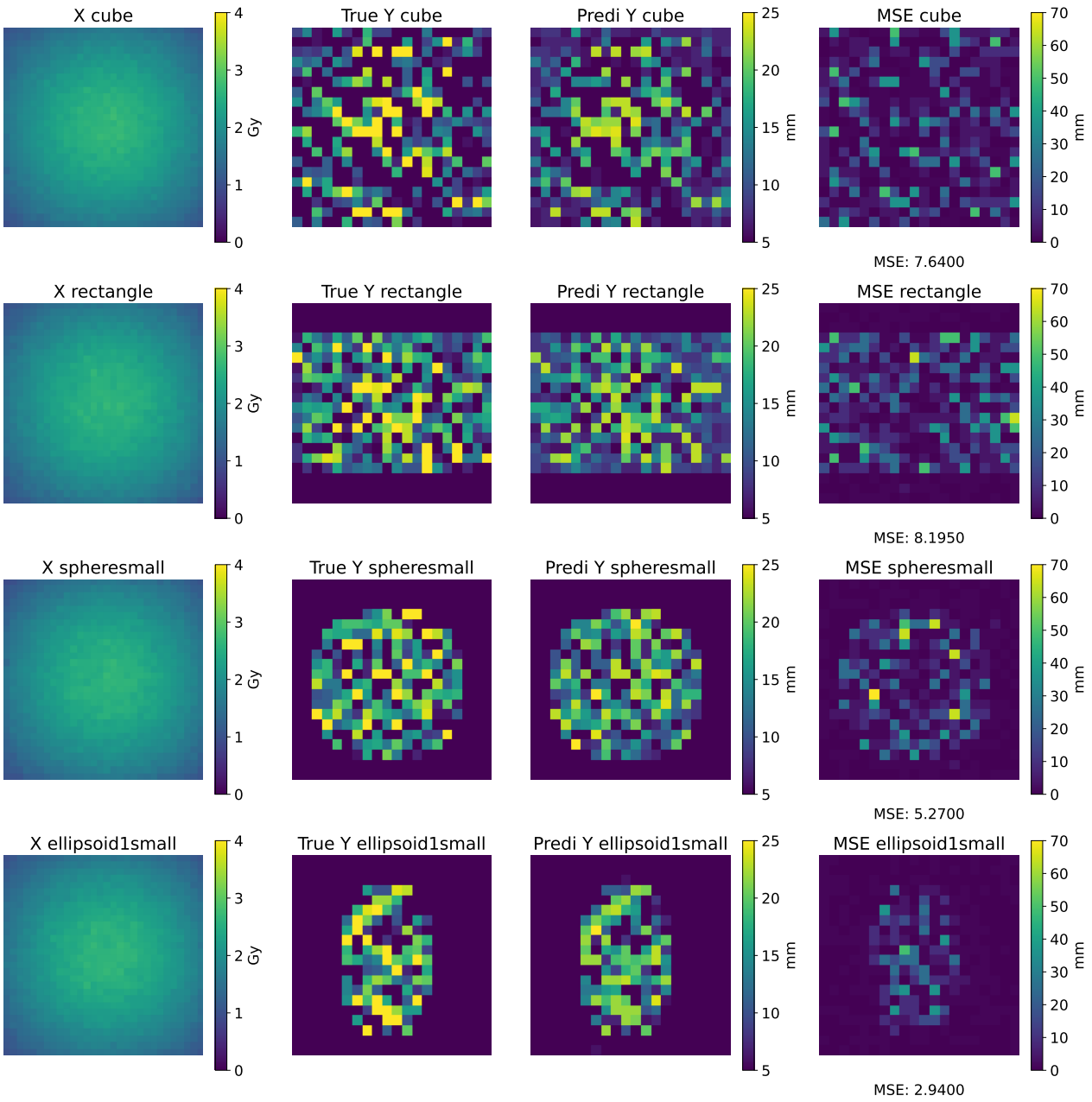

Figure 46: Group V: Input slice, Comparison between the true and predicted outputs of the test phase, Loss function pixel by pixel and global MSE

	Min MSE	Max MSE	Mean MSE
Global	1.16	34.83	6.44

Table 8: Group V: Minimum, maximum, and mean values of MSE

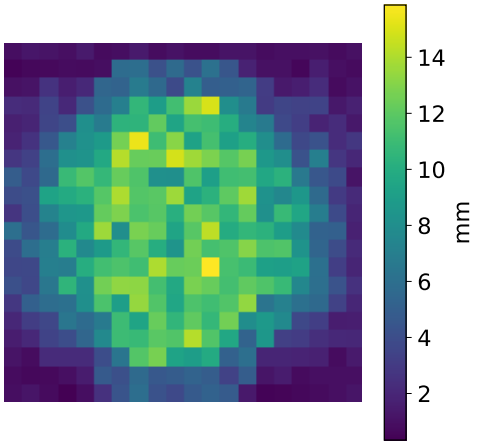

Figure 47: Mean MSE pixel by pixel of Group V test set

Chapter 9

Dose maps comparison

This chapter studies the dose comparison between an initial dose map called a true or reference dose map and a new dose map called a predicted or evaluation dose map. The predicted dose map results from the MC^2 simulation of a predicted CEM. The dose map comparison is done for two different test sets (Group S and Group V). Comparison is performed through dose-volume histograms and Gamma index calculations.

9.1 Dose-Volume Histogram

A predicted CEM of the Group S test set is selected randomly. A new MC^2 simulation is performed with the treatment plan and this predicted CEM. The corresponding dose is registered. The same process is applied to Group V. The predicted CEM chosen in Group S is a cube with a global MSE equal to 1.03. The predicted CEM chosen in Group V is a small rectangle with a global MSE equal to 4.81.

A dose-volume histogram (DVH) is a useful tool to compare dose maps together. The DVH consists in relating radiation dose to tissue volume. It is often used in radiation therapy planning to assess the dose distribution received by a target volume and surrounding healthy tissues in a PTV [21]. The cumulative DVH in Figure 48 plots the relative volume in percent in the cubic target volume as a function of the dose in Gray. It has the particularity to be reversed as a point in the curve specifies that at this specific volume, the dose is equal to or greater than the corresponding dose value. The blue curve represents the reference dose map which was used as data for the model testing. The yellow curve is the evaluation dose map that demonstrates the model's performance.

The goal is to have both curves overlap perfectly. In practice, a small difference between the curves might appear due to statistic noise in MC^2 simulations. Indeed, two simulations of FLASH PT with the exact CEM and treatment plan lead to slightly different dose maps. This difference is very small with a 1% of uncertainty and hence will be neglected in the observations of the DVH. In Figure 48, the curves are very close to each other at each point which implies that the CEM was well optimized. The AI model did well to predict a CEM that would give a dose map suited for the cubic target volume. These results show that having a small MSE between the true and predicted CEMs leads to small differences in their dose maps.

Another piece of information provided by the DVH is that the SOBP is not covering perfectly the whole target volume. Otherwise, the slope of the curve would be higher and the plateau would last until a bit before the objective dose of 2.5. However, this tendency is observed on both curves meaning that this characteristic is coming from the data acquisition and not the AI model itself.

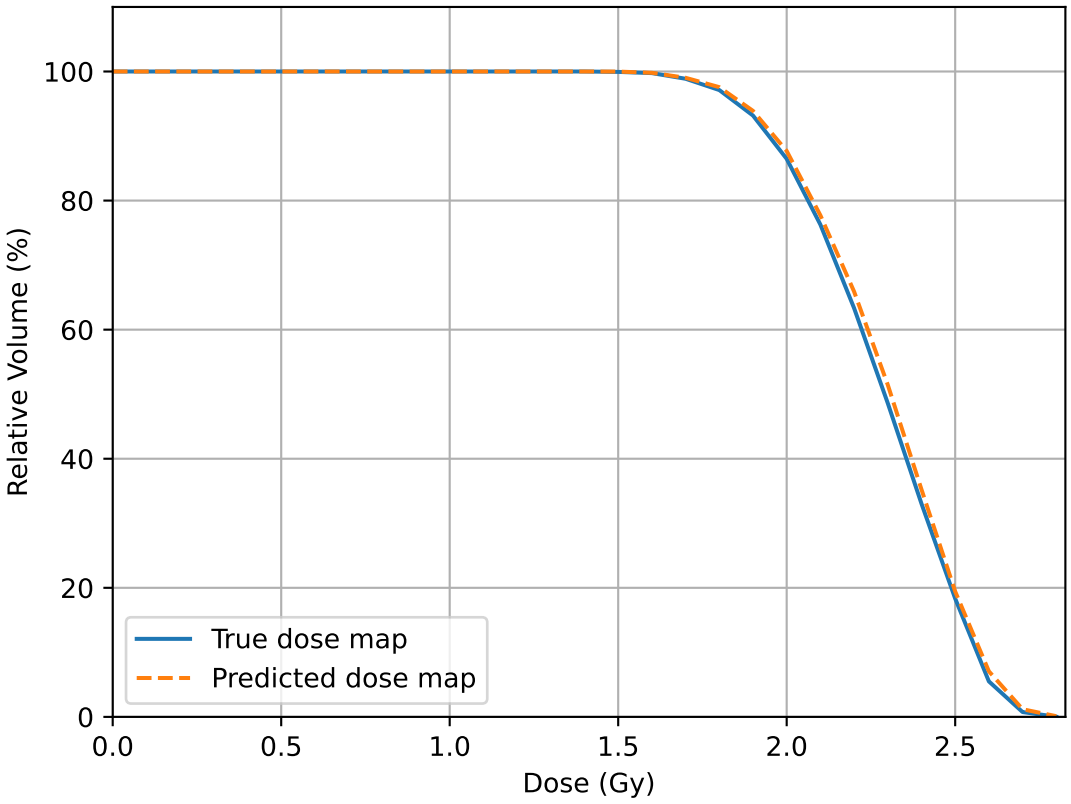

Figure 48: Dose-volume histogram of the true and predicted dose maps from Group S test set

In Figure 49, both curves are close to each other but a gap lies between the reference and the evaluation curves. This implies that the CEM was less well-optimized than in the previous case. Once again, it highlights the fact that dose maps are more different from each other when the MSE between their corresponding CEM is high. In this case, the MSE was equal to 4.81 compared to 1.03 in the previous case.

The SOBP is more extended in this small rectangular parallelepiped target volume as 100 % of the target volume was irradiated by at least 2 Gy. The curves then decrease sharply towards the objective dose of 2.5 Gy.

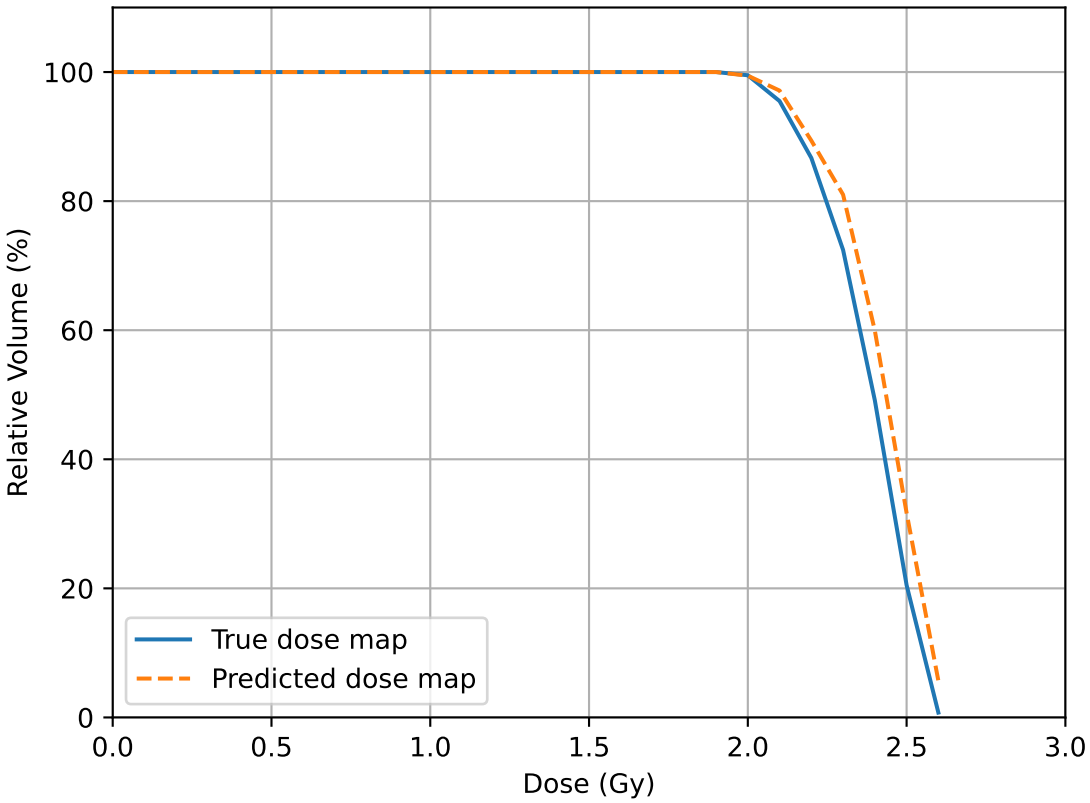

Figure 49: Dose-volume histogram of the true and predicted dose maps from Group V test set

9.2 Gamma Index

The Gamma index method is used to evaluate the measured dose distribution of a radiotherapy technique by comparing it to the calculated dose distribution. The accuracy and quality of the calculations of a treatment planning system can be assessed this way. It is one of the preferred dose distribution comparison method. The Gamma index method takes into consideration the dose difference and the spatial displacement between analyzed points to provide a Gamma-index as a result of comparison [27]. It is calculated according to this formula :

$$\gamma(\overrightarrow{x}) = min(\gamma(\overrightarrow{x}_r, \overrightarrow{x}_p)) = \sqrt{\frac{|\overrightarrow{x}_r, \overrightarrow{x}_p|^2}{dist^2} + \frac{|D(\overrightarrow{x}_r) - D(\overrightarrow{x}_p)|^2}{\nabla D^2}} \tag{11}$$

where $|\overrightarrow{x}_r, \overrightarrow{x}_p|^2$ is the distance between analyzed points, $|D(\overrightarrow{x}_r) - D(\overrightarrow{x}_p)|^2$ is the dose difference, $dist^2$ is the distance threshold, and ∇D^2 is the dose threshold.

A python library called “pymedphys” has a function that computes multidimensional Gamma while using a fast algorithm presented in [26].

With this Gamma matrix, a percentage of successful rate can be calculated. The criteria for an element of the matrix to be sucessful is to have its Gamma index smaller than 1. Indeed, the smaller the Gamma index, the less differences there are between the doses. In is thus possible to quantify how much two multidimensional dose maps are close to each other with a single value.

Put in the context of the CEM optimization problem, the dose comparison is performed as explained above between the 3D reference dose map used for the model testing and the 3D evaluation dose map resulting from MC^2 simulation with the predicted CEM.

Three distance and dose thresholds are tested on the dose map pairs of Groups R and V. Each pair is composed of the reference dose map and the evaluation dose map. The success rate is defined as the number of pixels whose Gamma index is smaller than 1 in percentage. The success rate for each data set and criteria are displayed in Table 9.

	Group S	Group V
dist=3mm, dose coef=3%	99.57%	98.62%
dist=2mm, dose coef=2%	98.11%	94.62%
dist=1mm, dose coef=1%	84.72%	70.68%

Table 9: Success rate based on the Gamma index of two test sets and 3 different criteria

For each set of criteria, Group S shows the highest success rate. This fits with the observations made on the DVHs. Even though Group S has the best results, the success rate of Group V is not bad either. Clinicians agree on having an acceptable similarity between two doses when their success rate is equal to or higher than 95% for a distance threshold of 2mm and a dose percent threshold equal to 2%. Group V's success rate for those criteria is not far from meeting this condition with its value of 94,62 %.

To conclude, both dose-volume histograms and Gamma index showed that the random pair selected in Group S has very similar dose maps meaning that the model was well-trained and that CEM optimization was successful. As for the pair randomly selected in Group V, the dose maps are quite similar and its Gamma index is nearly acceptable from a clinician's point of view which is strict. The model trained on Group V can also be said well trained and CEM optimization was successful.

Chapter 10

Conclusion

FLASH proton therapy is a promising technique that allies the advantages of the FLASH concept and proton therapy. the FLASH effect is the sparing of more healthy tissues while preserving a high dose in the tumor thanks to high dose rate irradiation. The advantage of proton therapy is the sparing of healthy tissues thanks to the particular dose profile of protons in matter called Bragg peak.

One of the challenges of FLASH proton therapy is the design of a piece called Conformal Energy Modulator (CEM) whose role is to ensure the uniform dose distribution in depth in a tumor in a single shot of 3 seconds. Its design must be specifically designed for each patient and must accurately take scattering into account. Scattering is the interaction of protons with matter that deviates protons from their initial trajectory due to the repulsive force from the positive charge of the nucleus.

Some methods to optimize the CEM according to a tumor already exist but have the disadvantages of either taking too much computational time or either be not accurate enough. The goal of this thesis was to find a new way to optimize CEMs that would be fast and accurate. This new way was to use artificial intelligence.

The following chapter of this thesis focused on the data acquisition process employed to train the AI model. Those can be performed with FLASHOpenTPS which is a treatment planning system that simulates dose deposition for FLASH PT. The necessary components for the simulations were defined and some assumptions on the FLASH PT simulations were made to simplify the problem and make it feasible. The data used by the AI model were chosen as the 3D dose maps that represent the dose deposition in water (substitute for a patient) and the CEM under the form of a 2D matrix whose values are the elevation of the CEM spikes. The inputs were the dose maps whereas the outputs were the CEMs. The acquisition of 3 different data sets was performed: CEMs randomly built (Group R), CEMs specifically designed for basic tumors (Group S), and CEMs specifically designed for basic tumors and with more variability (Group V) + their corresponding dose maps.

The following chapter of the thesis described the choice of the AI model and its architecture to match the CEMs and dose map size. A model that appeared to be efficient was the CNN given the multidimensional data set and the necessary features extraction. As the output CEMs are 2D matrices, the U-Net which is known to have an encoder and

decoder part seemed appropriate for this problem.

The model was subsequently trained on Groups R, S, and V individually, as well as on the combined data set of Groups S and R. The model's performance is first evaluated on the validation set. In order to rank the different trained models, several model performance results that could be compared between sets were performed.

According to those, Group R showed the worst results with a high MSE and predicted CEMs being equal to the mean of the true CEMs values. This suggests that the model failed to effectively learn the underlying physics of FLASH PT. The dissimilarity and lack of common features in the data from Group R are likely the main contributing factors, making it challenging for the model to discern any discernible patterns. Group S shows excellent results with high visual similarities between the CEMs and a global MSE smaller than 1. However, given the low variability and absence of overfitting in Group S, there is a possibility that the model may not have adequately learned the real physics underlying the data. It raises concerns about the model's robustness when confronted with entirely new data. Combining both Groups R and S into a single dataset did not yield improvements compared to the results achieved when analyzing Group S and Group R separately. The initial intention of merging the two groups was not efficient, hence this trained group is not further developed. Group V results were good with a rather low MSE around 8 and good visual similarities between the true and predicted outputs. This model showed overfitting which confirms that the data is more variable compared to Group S.

Following the utilization of the validation set to provide feedback on the model's performance, data augmentation, and hyperparameter tuning were conducted. However, data augmentation did not yield further decreases in validation MSE across any of the trained models. As a result, the analysis was halted for the model trained on Group R due to the absence of observable improvements. On the other hand, hyperparameter tuning proved successful for Group V, resulting in a decrease in the MSE loss from 8 to 7. Meanwhile, Group S maintained an MSE of approximately 0.6. To evaluate the model's overall performance, the test set was utilized, and both Groups S and V demonstrated outcomes closely aligned with their respective validation sets. This indicates that both models were effectively trained and possess the capability to perform well when presented with previously unseen data.

In the final chapter, a comparison was made between a dose map from the test set and a dose map generated through FLASH PT simulation using the model's predicted CEM. In Group S, the evaluation of DVH and Gamma index revealed a remarkably close resemblance between the dose maps, suggesting a near-perfect similarity. In the case of Group V, the dose maps exhibited significant similarity, with a Gamma index success rate of 94.6%, which nearly meets the threshold at which clinicians considered a sufficiently acceptable dose map similarity. To conclude, the AI model trained on Group S shows excellent results for CEM optimization though its robustness is not ensured. On the other hand, the model trained on Group V has sufficiently good results though less than Group S but is most likely to be more robust.

For future works, it is worth considering customizing the loss function of the model, particularly for Group R, which exhibited the worst results. Group R's poor performance may have been influenced by an inappropriate choice of loss function. The tuning of a more suitable loss according to this specific problem could improve the results of Group R. Another approach that could be tested is the further complexification of the model's architecture. A variance of the U-Net called V-Net, is known to work better on 3D data. It uses residual connections and 3D convolutional operation to learn generalization. However, this architecture is said to require more tuning of hyperparameters. Some assumptions imposed on the data acquisition to simplify the problem could be removed to get closer to real cases. Among them, the equal spot weight and the water phantom could be modified. Indeed, the spot weights could be tunable and optimized with the CEM. The water phantom could be transformed into human body components such as bones or fat. This would be performed by changing the Hounsfield unit of water into the Hounsfield unit of bones or fat in the CT. However, those modifications complexified greatly the problem as only the CEMs and dose maps would not be sufficient anymore for proper model training.

Chapter 11

Appendix

Parameters optimization on Group S

Number of feature maps

In Figure 50, the loss function for four different numbers of feature maps as a function of the number of epochs is displayed. A number of 8 or 16 feature maps seems too small as the model converges too slowly and the MSE stops at a high plateau. As for 32 or 64 feature maps, the model converges similarly and ends up at the same MSE value. However, it is better to choose a number of feature maps equal to 32. Indeed, the tendency of the model to overfit increases with the number of feature maps.

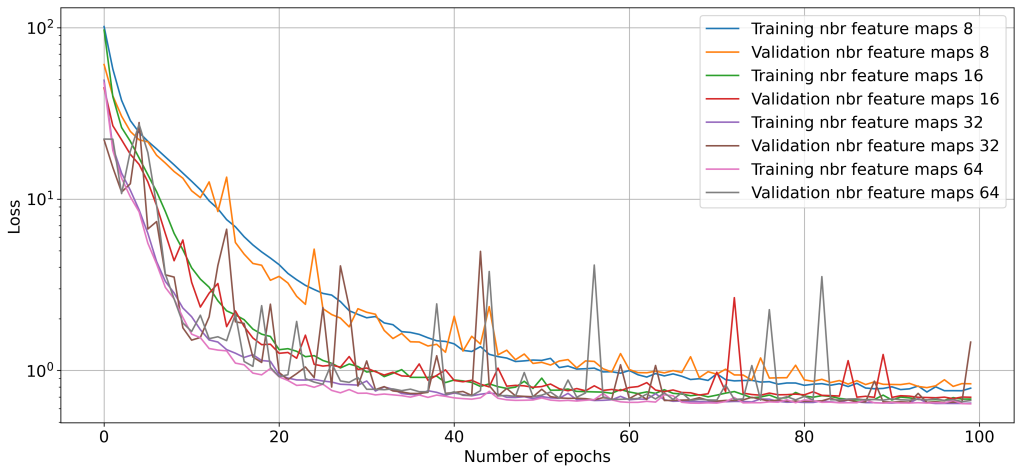

Figure 50: Loss as a function of the number of epochs

Batch size

In Figure 51, the loss function for four different batch sizes as a function of the number of epochs is displayed. The smaller the batch size is, the faster the model converges. However, the smaller the batch size is, the longer the computation time. A good compromise is a batch size equal to 32 that converges fast enough to reach the same plateau as

a batch size equal to 16 over 100 epochs. Batch sizes equal to 64 and 128 converge a bit too slowly and thus a bigger number of epochs would be needed to reach the same MSE.

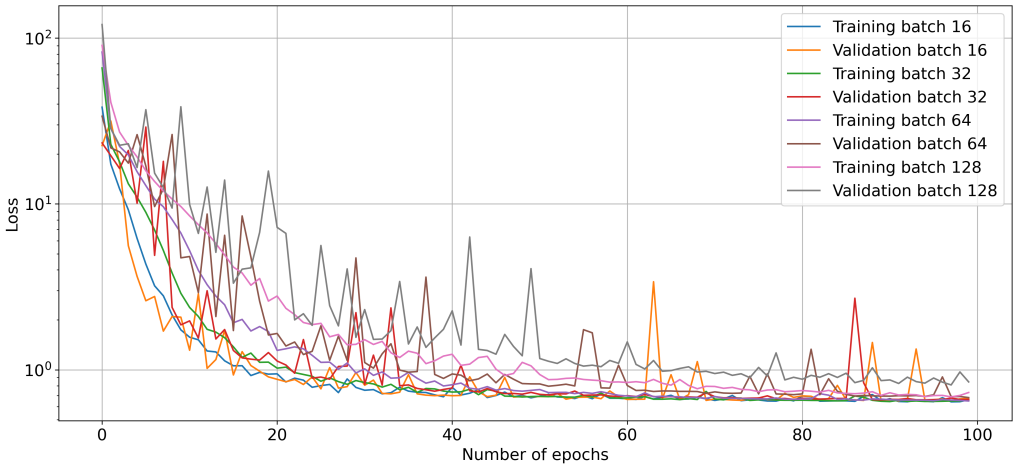

Figure 51: Group S: Losses as a function of the number of epochs for the training and validation set

Learning rate

In Figure 52, it can be seen that the model converges faster for a bigger learning rate. However, the noise amplifies when the learning rate becomes smaller which is very evident in Figure 53. A learning rate of 0.00001 seems to be too small as the model converges very slowly while a learning rate of 0.0001 or above (0.01, 0.001) gives very noisy curves. A good compromise looks to be a learning rate of 0.00005 which converges quickly and which has less noise in its loss curves.

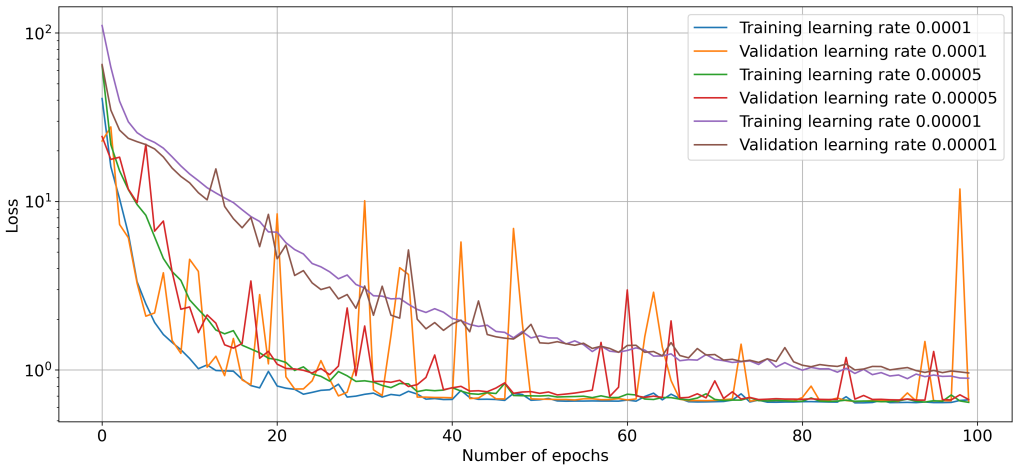

Figure 52: Loss as a function of the number of epochs

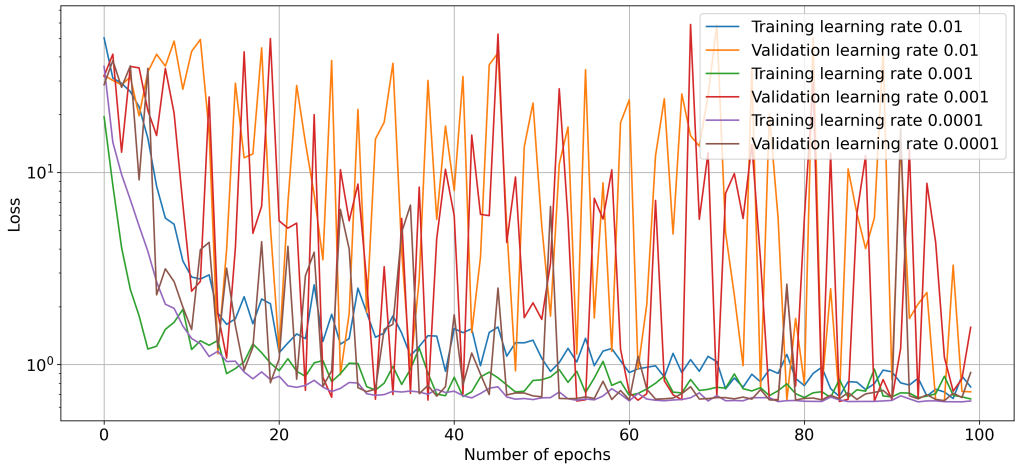

Figure 53: Loss as a function of the number of epochs

Batch normalization

In Figure 54, the loss function with and without batch normalization in the model as a function of the number of epochs is displayed. The model converges much faster with batch normalization included in the model. Moreover, the training and validation losses without BN also become noisy with the number of epochs. To save computation time, having BN in the model is the best choice.

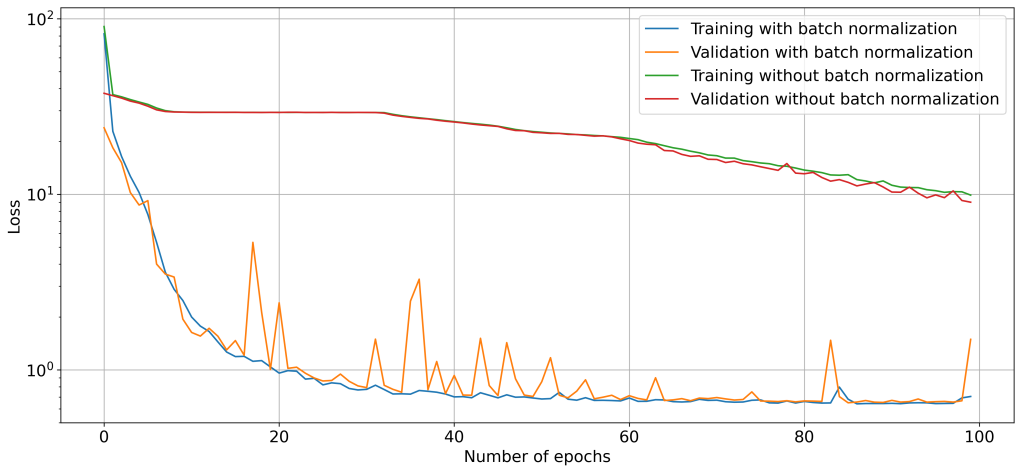

Figure 54: Loss as a function of the number of epochs

Dropout

As seen in Figure 56, dropout does not really have an effect on the results. This might

be caused by the batch normalization method which makes regularization methods less necessary.

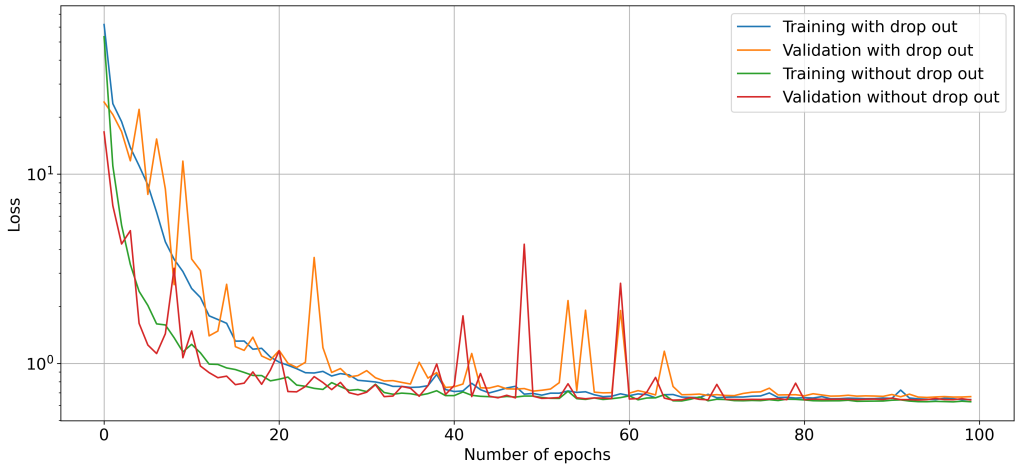

Figure 55: Loss as a function of the number of epochs

Strided convolution VS Max pooling

Strided convolutions lead to a smaller MSE compared to max pooling.

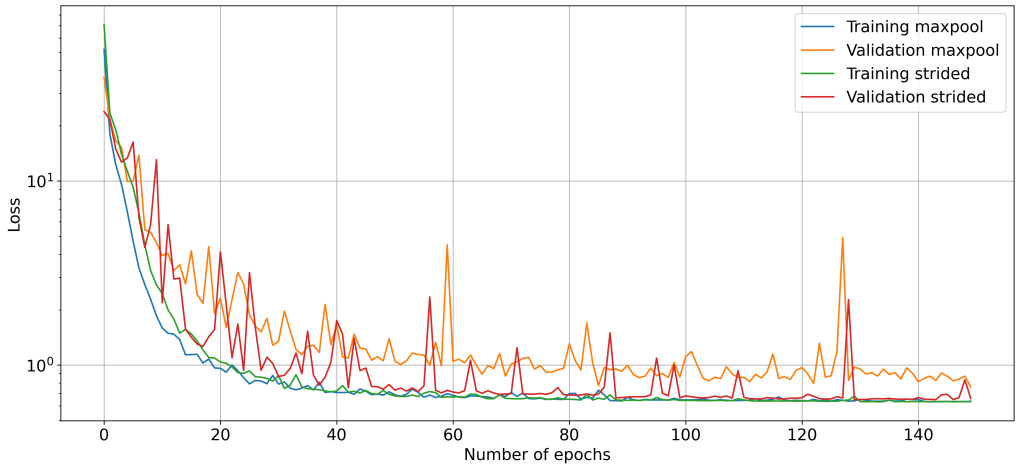

Figure 56: Loss as a function of the number of epochs

Parameters	Values
Nominal energy	226 MeV
Number of spots	16
Spot spacing	7 mm
Gantry angle	0°
Minimal dose objective	2.5 Gy
Maximal dose objective	2.55 Gy
Number of primary protons	$2 * 10^6$
Collimator density	8.5 g/cm^3
Collimator aperture size	25mm $*$ 25 mm
Collimator thickness	65 mm
Range shifter density	2.6 g/cm^3
Range shifter thickness	135 mm
CEM density	1.2 g/cm^3
Distance from CEM to isocenter position	350mm
Isocenter position	(50mm, 480mm, 50mm)
Hounsfield unit of air	-1024 HU

Table 10: Summary of configuration parameters useful for FLASH PT simulations with FLASHOpenTPS

Chapter 12

References

[1] Miropt. `https://openpath.software/software_package/miropt.html`.

[2] Openreggui. `https://openreggui.org/`.

[3] Opentps. `http://opentps.org/`.

[4] Raystation. `https://www.raysearchlabs.com/raystation/`.

[5] H. A. Bethe. Molière's theory of multiple scattering. *Physical Review*, 1953.

[6] V. Boudart and M. Fays. Machine learning algorithm for minute-long burst searches. *Phys. Rev. D*, April 2022.

[7] V. Cobut, Y. Frongillo, J. P. Patau, T. Goulet, M.-J. Fraser, and J.-P. Jay-Gerin. Monte carlo simulation of fast electron and proton tracks in liquid water–i. physical and physicochemical aspects. *Published by Elsevier Science*, 1998.

[8] M. Durante, E. Bräuer-Krisch, and M. Hill. Faster and safer? flash ultra-high dose rate in radiotherapy. *British Journal of Radiology*, 2018.

[9] Vincent Favaudon, Rudi Labarbe, and Charles L. Limoli. Model studies of the role of oxygen in the flash effect. *Medical Physics*, 2020.

[10] Pierre Geurts and Louis Wehenkel. *Classification and Regression Trees*. Institut Montefiore, University of Liège, Belgium, 2022.

[11] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

[12] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Series in Statistics, 2009.

[13] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. 2015.

[14] Charles L. Limoli and Marie-Catherine Vozenin. Reinventing radiobiology in the light of flash radiotherapy. *Annual Review of Cancer Biology*, 2020.

[15] Gilles Louppe. Deep learning lecture 5: Convolutional networks.

[16] Pierre Montay-Gruel, Kristoffer Petersson, Maud Jaccard, Gaël Boivin, Jean-François Germond, Benoit Petit, Raphaël Doenlen, Vincent Favaudon, François Bochud, Claude Bailat, Jean Bourhis, and Marie-Catherine Vozenin. Irradiation in a flash: Unique sparing of memory in mice after whole brain irradiation with dose rates above 100 gy/s. *Radiotherapy and Oncology*, 2017.

[17] Wayne D Newhauser and Rui Zhang. The physics of proton therapy. *Physics in Medicine and Biology*, 2014.

[18] Annalisa Patriarca, Charles Fouillade, Michel Auger, Frédéric Martin, Frédéric Pouzoulet, Catherine Nauraye, Sophie Heinrich, Vincent Favaudon, Samuel Meyroneinc, Rémi Dendale, Alejandro Mazal, Philip Poortmans, Pierre Verrelle, and Ludovic De Marzi. Experimental set-up for flash proton irradiation of small animals using a clinical system. *International Journal of Radiation Oncology*, 2018.

[19] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *Computer Science Department and BIOSS Centre for Biological Signalling Studies, University of Freiburg, Germany*, 2015. `http://lmb.informatik.uni-freiburg.de/`.

[20] Yuri Simeonov, Uli Weber, Petar Penchev, Toke Printz Ringbæk, Christoph Schuy, Stephan Brons, Rita Engenhart-Cabillic, Jens Bliedtner, and Klemens Zink. 3d range-modulator for scanned particle therapy: development, monte carlo simulations and experimental evaluation. 2017.

[21] Yuri Simeonov, Uli Weber, Christoph Schuy, Rita Engenhart-Cabillic, Petar Penchev, Veronika Flatten, and Klemens Zink. Development, monte carlo simulations and experimental evaluation of a 3d range-modulator for a complex target in scanned proton therapy. *Biomedical Physics and Engineering Express*, 2022.

[22] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. 2014.

[23] Josh Starmer. *The StatQuest Illustrated Guide to Machine Learning*. StatQuest, 2020.

[24] Marie-Catherine Vozenin, J.H. Hendry, and C.L. Limoli. Biological benefits of ultra-high dose rate flash radiotherapy: Sleeping beauty awoken. *Clinical Oncology*, 2019.

[25] Louis Wehenkel and Pierre Geurts. (deep) neural networks, 2021.

[26] Markus Wendling, Lambert J Zijp, Leah N McDermott, Ewoud J Smit, Jan-Jakob Sonke, Ben J Mijnheer, and Marcel van Herk. A fast algorithm for gamma evaluation in 3d. *Medical Physics*, Radiation therapy physics, 2007.

[27] Janusz Winiecki, Tomasz Morgaś, Karolina Majewska, and Barbara Drzewiecka. The gamma evaluation method as a routine qa procedure of imrt. *Reports of Practical Oncology & Radiotherapy*, 2009.