Running Head: AI for enhanced non-invasive medical imaging

University of Poitiers **Academic year 2021 - 2022**

SP2MI, UFR Fundamental and Applied Sciences

Master Connected Objects

Internship report

Addressed for the validation of Master 2: Engineering of intelligent objects

Artificial intelligence for enhanced non-invasive medical imaging

Internship Period: 28 Mars, 2022 - 01 September 2022

Submitted by

Marwan AL OMARI

Internship tutor

BOURDON Pascal

XLIM laboratory CNRS 7252

Bât. H1 - SP2MI

11 Bd Marie et Pierre Curie

86073 Poitiers Cedex 9

Referring teacher:

Noel Richard

Table of Contents

List of Figures

List of Tables

Acknowledgments

I devote my sincere thanks first of all to my co-supervisor Pascal Bourdon, and my supervisor Christine Fernandez for the constant follow-up and direction all along the project. Besides, I would like to thank my tutor Noel Richard for his insights on different aspects of project management. Moreover, I would like to thank the XLIM staff, as I had the opportunity to work there and have the chance to establish new connections.

Moreover, I would like to devote my sincere thanks to Campus France for the financing my master studies and a whole year of French language. I really appreciate the life-changing opportunity that had given me to be in France. Besides, I would not forget my family, they have stayed besides me supporting me on the difficult times during my master studies as well as through the internship. I must insist I had no knowledge in French language, I could still remember the days I spent just blocked without advancing trying to figure out some words in French language. But as everything else in this world, it gets better and better as we roll on.

Abstract

Artificial intelligence has advanced in the late years, showing enhanced capabilities in handling data on various applications of computer vision, including classification, segmentation, detection, etc. Generative neural network has proved its effectivity in capturing deeper features during the process of image synthesis. Based on the state-of-the-art, generative networks are competitive in comparison to traditional machine learning algorithms, in learning similar structures between input images. Hence, this research project develops improved single or multiple input and single output architectures for the goal of synthesizing medical imaging. These architectures are important as it saves much times on tumor diagnosis and it would be the cure to avoid injecting additional substance to the brain while extracting T1c or Flair. The models are trained and experimented using hyper-optimization techniques to fine-tune the parameters on the BRATS2018 dataset. Next, the architectures have achieved competitive results in highlighting the state-of-the-art-performance.

Keywords: Artificial intelligence, generative neural networks, deep learning, medical synthesis

Abbreviations and Acronyms

In this section, abbreviated and acronym terms, which used in this report paper, are clarified, and presented to avoid the over repetitiveness of same terms. They are as follows:

Machine **L**earning (ML)

Deep **L**earning (DL)

Normalized **M**ean **A**bsolute **E**rror (NMAE)

Peak **S**ignal-to-**N**oise **R**atio (PSNR)

Structural **S**imilarity **I**ndex **M**easurement (SSIM)

Visual **I**nformation **F**idelity (VIF)

Naturalness **I**mage **Q**uality **E**valuator (NIQE)

Brain **T**umor **S**egmentation **C**hallenge (BRATS)

The **Celeb**Faces **A**ttributes (CelebA)

The **Ra**dboud **F**aces **D**atabase (RaFD)

T1-weighted (T1)

T1 and **C**ontrast-enhance (T1c)

T2-weighted (T2)

Fluid-**A**ttenuated **I**nversion **R**ecovery (Flair)

Convolutional **N**eural **N**etworks (CNNs)

Magnetic **R**esonance **I**maging (MRI)

Generative **A**dversarial **N**etwork (GAN)

Generative **A**dversarial **N**etwork**S** (GANs)

Centre National de la Recherche Scientifique/**S**cientific **R**esearch **N**ational **C**enter (CNRS)

Program **E**valuation and **R**eview **T**echnique (PERT)

Operating **S**ystems (OS)

Variational **A**uto**E**ncoder**S** (VAEs)

Cycle-**C**onsistency (CC)

Convolutional Patched**GANS** (CGANs)

Amazon **M**echanical **T**urk (AMT)

Artificial **N**eural **N**etwork**S** (ANNs)

Deep **N**eural **N**etwork**S** (DNNs)

Recurrent **N**eural **N**etwork**S** (RNNs)

Natural **L**anguage **P**rocessing (NLP)

Machine **T**ranslation (MT)

Artificial **I**ntelligence (AI)

COrona**VI**rus **D**isease **19** (Covid-19)

Intersection-**O**ver-**U**nion (IOU)

Cross **V**alidation (CV)

Computed **T**omography (CT)

Magnetic **R**esonance **I**maging (MRI)

ACCuracy (ACC)

Secure **SH**ell (SSH)

SSH **F**ile **T**ransfer **P**rotocol (SFTP)

Single-**I**nput **S**ingle-**O**utput (SISO)

Multi-**I**nput **S**ingle-**O**utput (MISO)

Multi-**I**nput **M**ulti-**O**utput (MIMO)

Single-**I**nput **M**ulti-**O**utput (SIMO)

BRAin **T**umor **S**egmentation Challenge **2015** (BRATS2015)

BRAin **T**umor **S**egmentation Challenge **2018** (BRATS2018)

Learning **R**ate (LR)

Street **V**iew **H**ouse **N**umber (SVHN)

XLIM Laboratory and Internship Presentations

In this section, a brief introduction is provided on XLIM laboratory as well as its different domain specialties. Then, an overall presentation of the internship position will be discussed along with the objective missions, and project management.

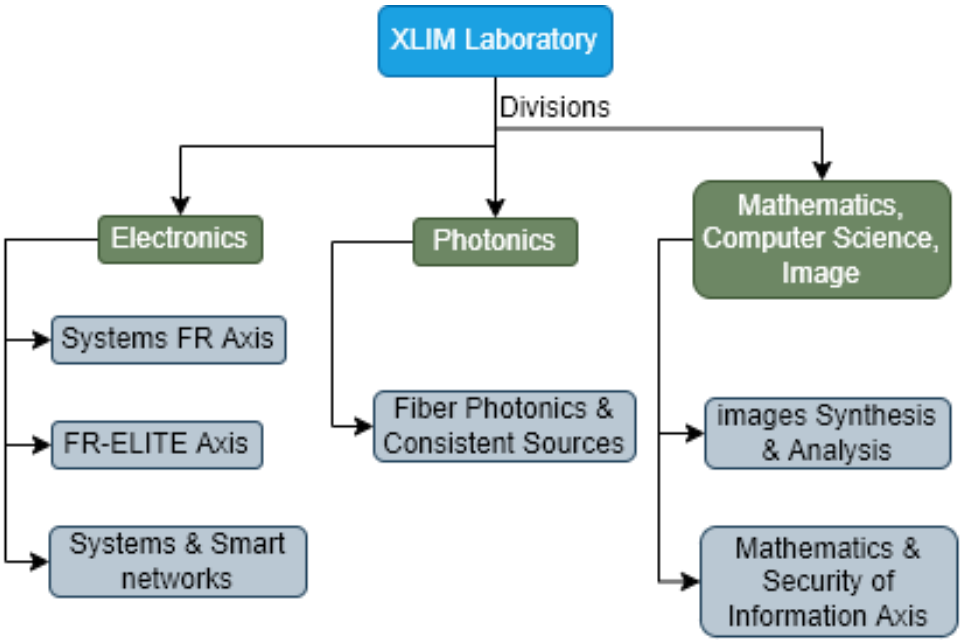
1.1 XLIM Laboratory

XLIM, as a one of the French research centers, is known under the name of CNRS [7], numbered as 7252. The laboratory is greatly centered on electronics, optics, photonics, mathematics, computer science, image processing, telecommunication, network security, bioengineering, and energy. A representation of XLIM divisions is presented below in Figure (1).

XLIM is a multidisciplinary research institute, located on several geographical sites, in Limoges on the sites of the Faculty of Science and Technology, the ENSIL, Ester-Technopole, on the University Campus of Brive; and in Poitiers on the site of the Futuroscope Technopole. It brings together more than 440 teacher-researchers, CNRS researchers, engineers, administrative staff technicians, doctoral and post-doctoral students.

The main research of XLIM is mainly relied on two platforms. On one hand, a platform gives researchers access to technological equipment for the creation of optical structures. For instance, a large pool of lasers and instruments are available for experimental characterization of electronic, optical, electromagnetic, and radiating devices. On the other hand, the laboratory engaged in modeling and simulation activities as it is equipped with a software forge that is available its members and their external collaborators. For that, it offers multiscale simulation of complex systems based on interface models between physics, mathematics, and computer science.

Figure 1- Representation of XLIM laboratory, which consists of 3 scientific poles and 6 scientific axes



1.2 Internship Presentations

During my internship, I have been a research engineer, working on the development and experimentation of DL architectures for solving the problem of generating different MRI sequences, mainly on the generation of fake modalities on BRATS2018. I have acquired many

skills, which will be my next source for knowledge-making and strategy development in the future. Some of the skills I have learned summarized in the following:

1. Connection management and coding at distance network computer using SSH and SFTP
2. Prototyping an algorithm for research enhancement
3. Planning the project and self-management

1.2.1 Specifications

The main specifications of the internships, as following:

1. Developing a solution for constructing MRI sequences using artificial intelligence algorithms.
2. Producing a prototype in Python language.

In the following section, I will describe and give many details on the missions during the whole period of internship.

1.2.2 Description

To begin with, I was responsible in discovering BRATS2018 dataset in medical context aside from its structural and representational complexity. Then, I have had analyzed the dataset in conducting different manipulations in using python programming language. I discovered the data with representational graphs with the help of Matplotlib visualization.

Then, I have developed SIMO and MISO DL architectures, using GANs along with U-Net, to predict certain modalities from one or multiple input. Therefore, I have had to identify the state-of-the-art solutions and algorithms for enhancing the results, including pre-processing methods and algorithms structures. I have experimented different methods for fine-tuning the GANs algorithm, including pipeline and sample testing.

During fact-to-face as well as web meetings, I had to present the results that I came up in each step; so, through presentations, my French language was improving in oral and written communications as I was writing weekly and monthly reports and emails to my supervisor.

In addition to the technical tasks, my skills, as a part of XLIM's Internship program in medical imaging, been developed massively in communication, organization, research independency, analytical vision, curiosity, and creativity. I have learnt to present ideas with good communication skills in both verbal and written forms. The writing is not free of limitations though I have tried my best to keep the writings free from errors.

In daily bases, I was learning new things that had shaped my mind in data science and AI. I believe I have proved myself as an essential member in providing revolutionary ideas and innovative solutions to improve image synthesis. However, I have faced many problems in setting up the environment for training the algorithms as I could not benefit from the fully accessibility to the office computer. I could say that everything depends on hierarchy so if I wanted something, as program supports, I must wait for it as I have no authority on the computer, for example. So, I found out using my personal computer would save much time on developing the different prototypes of GANs architectures.

Nevertheless, I have been a bit limited in my personal computer, which it was crashing in training with huge amount of data. Later, I had the accessibility to a server at distance with Quadro RTX, which I ran all the experimentations on. At the end, I have run the final experimentations in another server at distance that is equipped with Nvidia Tesla.

As for importance, I have found organization as an essential part for time management and task handling as I was responsible for different tasks in every stage of advancement. Data science and DL require dynamic and analytical mind in visualizing fitted solutions while covering a wide literature background in related work in medical imaging, ML or DL, and AI in general. Practically, the literature review has widened my horizons to pass certain challenges and experiences that other researchers have had been through.

In sum, I have really obtained great insights into the key performance and potential capacity of image synthesis through data analysis and algorithmic development. I believe they are rarely manifested in French research laboratories especially after the difficult time of Covid-19 pandemic.

1.3 Project Management

In this section, I am going to present the software I am using in the internship, starting from operating systems, programs, and coding platforms.

1.3.1 Operating systems

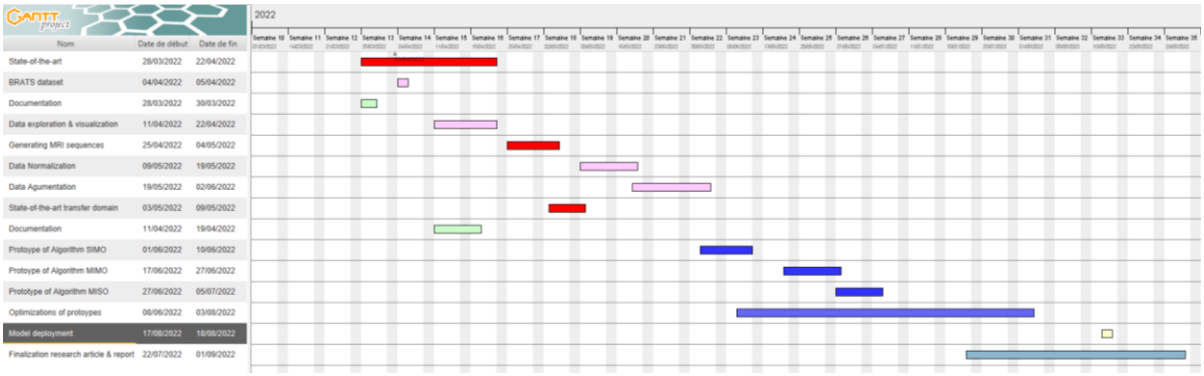
Linux and **Windows** OS are system software that manages computer hardware, software resources, and provides common services for computer programs. So, it is considered the fundamental software that could manage the other software that we use, for example, anaconda.

1.3.2 Software programs

SourceSup service, which operated through RENATER, is a management web platform for Higher Education and French Research organizations. Every member of the network can create a project at the platform in additionally permitting the collaboration of outside people at the projects.

GanttProject break down the work, build a Gantt chart, assign resources, and calculate project costs. In our case, we used to debrief the tasks for the ongoing weeks through performed and still in progress tasks.

Figure 2- GANT diagram - time management of tasks



As in the above Figure (2), we could see the current, finished, and continuous tasks that I am working on. For example, from the period between 28/03/2022 to 22/04/2022, I am covering the state-of-the-art concerning GANs and medical synthesis.

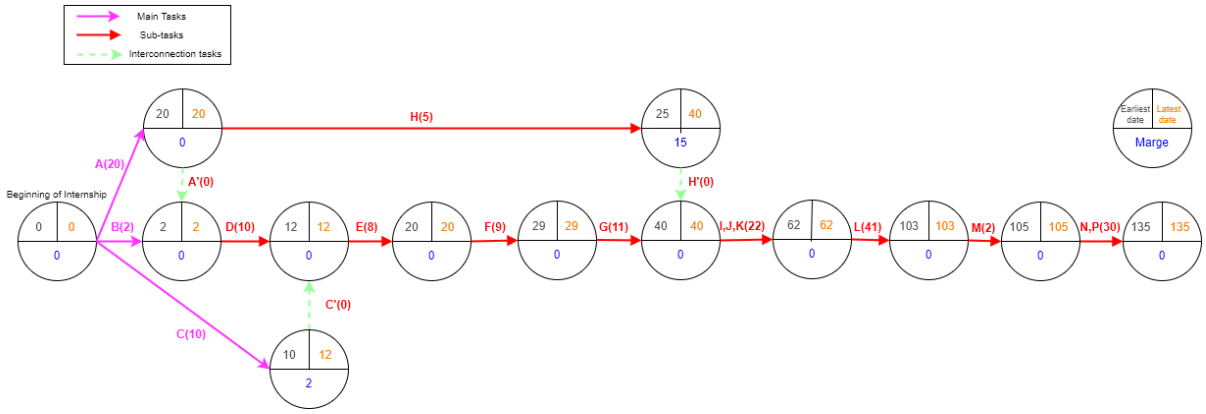
PERT, a tool for planning, organizing, and scheduling tasks within a project, represents the project schedule and categorize individual tasks. PERT charts are like Gantt charts, but with a different structure. In the following Table (1) and Figure (3), I will present the main tasks and its representations in respect to priority and its earlies and latest duration.

Table 1- PERT chart – tasks management

Task	Nature	Duration	Anteriority
A	State-of-the-art	20	-
B	BRATS dataset	2	-
C	Documentation	10	-
D	Data visualization and exploration	10	B
E	Generating MRI sequences	8	D, C
F	Data normalization	9	A, B, C
G	Data augmentation	11	F
H	Enrichment state-of-the-art	5	A

I	Prototype SIMO	8	H, G
J	Prototype MIMO	7	H, G
K	Prototype MISO	7	H, G
L	Optimizations of prototypes	41	I, J, K
M	Model deployment	2	I, J, K, L
N	Writing research article	15	I, J, K
P	Final Report	15	I, J, K
Total	15	24 weeks ≅ 1190 ≅ $170 days$	

Figure 3- PERT chart - time management of tasks



The key element that we could notice in PERT chart, we could estimate the earliest date of completing a task and the state of freedom, as in task H. There, the task could be delayed 15 days more because the other tasks, for example, D, E, or F, they are not dependent on H.

draw.io is free online diagram software that is used to write and simplify the coding and information into a visual schema that is easy to be understood.

Webex, by Cisco, is a software for video conferencing, online meetings, screen share, and webinars. I used it to report results, problems, and complains to the supervisor as well as for planning the weekly meetings.

ENT Zimbra is a collaborative messaging to write collaborators of internship and others, share the messages, create appointments.

Microsoft Office is a text processing application that allows the configurating the attributes of a document, such as layout, the styles of content, and to add their content in a variety of ways and formats to produce documents. It is used to write the weekly reports as well as the final report for the whole internship.

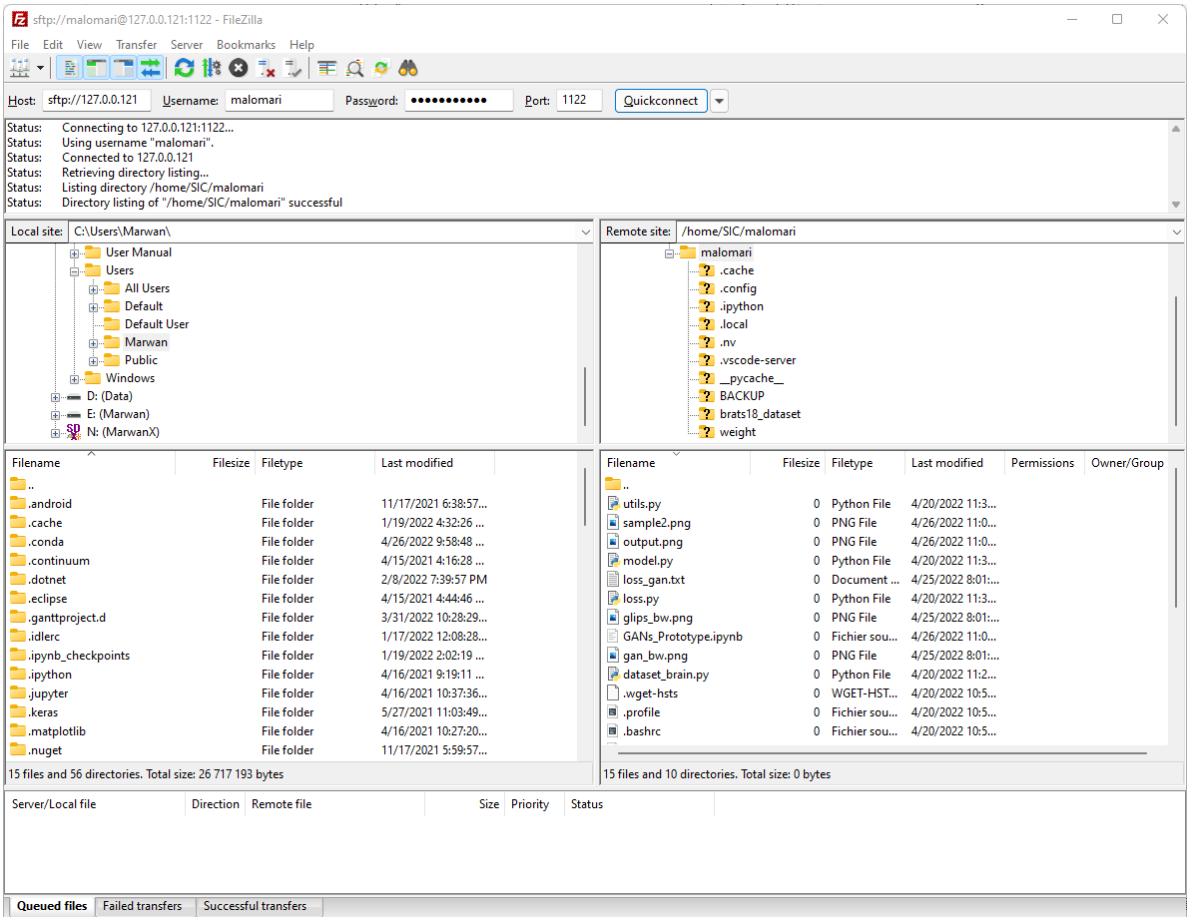
Microsoft PowerPoint is a presentation programs uses slides to convey information rich in multimedia.

Zotero helps to collect, organize, cite, and share your research sources.

FileZilla is a free and open-source, cross-platform FTP application, consisting of FileZilla Client and FileZilla Server. We used FileZilla client, which supports FTP, FTPS and a SFTP

server connection. An example of sftp connection done to the server in question as in the following Figure (4):

Figure 4- SFTP connection using FileZilla



1.3.3 Coding Plateforms

Juypter Notebook is a project that provides services for interactive computing using python. Therefore, as coding in python, Juypter will visualize the results on the same script window.

Anaconda is a distribution of Python programming language for scientific computing including data science, machine learning applications, large-scale data processing, predictive analytics, etc.… It is used as for the aim to simplify package management and deployment.

Microsoft Visual Studio Code, referred to as VS code, is a code editor. We used it to facilitate the SSH connections and code integration in the same window. It is easy to use and handle Jupyter notebook files without a local server opening in the background.

1. Introduction

Image processing employs different and multiple methodologies from hand-forged features to ML for the objectives of detection, classification, and segmentation. As a recent development in AI, DL employs deeper processing units called neurons to learn hierarchical features and representations of data [8]. In this case, DL methods outperform humans' predictions as it is well going into impressive advancement and evolvement in the different fields of science as in computer vision [9].

In the context of medical imaging, the classification of images generalized to distinguishing their different classes and categories in experimentation fine-tuning in auto generative tasks [13]. Generative networks, including collection style transfer, object transfiguration, season transfer, photo enhancement, etc. [15][16].

As for the aim of this internship, we have, therefore, adopted GANs model to generate different attributes from a single and multiple attributes as an input. The proposed architectures save time in predicting attributes and avoiding the injection of certain substance to the patient's brain. The data, being used in this study, is widely used for the improvements of brain lesions diagnosis and segmentation. BRATS2018 includes patient file images, each contains fives modalities (attributes): T1, T1c, T2, Seg, and Flair.

The works of this projects are specified in 5 sections. The first section is already introduced the subject under study. Later in the second section, we would discover the related works that are well provided towards the image synthesis. The section 3 would detail the DL background in the context of medical synthesis along with the dataset. The fourth section would specify the research approach in discussing the pre-processing steps, algorithms, and evaluation metrices. The fifth section, we would present the results of the experiments with dedicated analysis. The sixth section would conclude the whole idea of the research project in highlighting the work challenges along with further improvements in the future. Lastly, there would be the reference list that is numbered all through the paper for navigation simplicity.

2. Related Work

In this section, we would present the main research articles that experimented on open-source datasets from different challenges, for example, CelebA, RaFD or BRATS2015 [4], using GANs for the aim of better understanding the literature and the employed techniques in their approaches with respect to our project study. At the end of this section, we would provide a briefing of the studies in Table (2).

[3] Their multimodal GANs network assessed on four contrasts of T1, T1c, T2, and Flair by calculating NMAE, PSNR, SSIM, VIF, NIQE. They adopted StarGAN strategy [2] in mapping the four MRI modalities through 12 SISO models mapped in one-to-one modality, and a unified GAN to map them together. Their method was evaluated on BRATS2015. It consists of 274 subjects in which, 54 patients with low grade glioma, and 220 patients with high grade glioma. The voxel size of each image is $240 \times 240 \times 155$. For each image, $72 \times 72 \times 72$ patch size extracted and fed into the model. Then, the final estimation of overlapped regions was set to $48 \times 48 \times 48$ with 5-fold cross validation. Most of the important parameters are set to $\alpha = 2$, $\beta = 5$, $\gamma = 10$, $\delta = 2$, and $\mu = 0.1$ for balancing the weights of classification, synthetic consistency, cycle consistency, and adversarial losses, respectively. As a conclusion, for example, with T1 as input modality, the NMAEs for the generated T1c, T2, Flair respectively

are 0.034 ± 0.005, 0.041 ± 0.006, and 0.041 ± 0.006, the PSNRs are 32.353 ± 2.525 dB, 30.016 ± 2.577 dB, and 29.091 ± 2.795 dB, the SSIMs are 0.974 ± 0.059, 0.969 ± 0.059, and 0.959 ± 0.059, the VIF are 0.750 ± 0.087, 0.706 ± 0.097, and 0.654 ± 0.062, and NIQE are 1.396 ± 0.401, 1.511 ± 0.460, and 1.259 ± 0.358, respectively. Finally, their study is limited in spatially co-registered multimodal images before even they were used for training and testing on the small dataset. For future work, they will try to increase the amount of training and testing images through augmentation techniques.

[2] StarGAN, a unified GAN, handles multi-domain image-to-image translations in using a single model of one-to-one discriminator and generator. A single model takes in as input both an image and domain information by one-hot encoding, and learns the mappings between all available domains, using only one generator. Training process generates a target domain label randomly as to finish image translation. Due to the limitations in different models on treating image translation as a mono task that could not be generalized to more than an individual task, the researcher proposed a novel approach that could handle multiple domains using single generator and discriminator. Their approach works fine on facial feature transfer and facial expression synthesis tasks. Their architecture was tested on two datasets. Firstly, CelebA dataset contains 40 labels related to facial features such as hair color, gender, and age. Secondly, RaFD dataset contains 8 labels for facial expressions as like happy, angry, and sad. Besides, to avoid missing values in the dataset, they applied a mask vector of domain label that works on ignoring unknown labels and focus on what is already available. On CelebA, StarGAN achieved 66.2%, 39.1%, 70.6%, 47.4%, 61.5%, 49.8%, 52.2% for Hair color, Gender, Aged, H+G, H+A, G+A, and H+G+A, respectively. On RaFD: StarGAN achieved 2.12 loss with $53.2M \times 1$ parameters. On both CelebA+RaFD, their model learned properly the intended role of a mask vector in image-to-image translations when involving all the labels from multiple datasets altogether. In conclusion, StarGAN generated images of higher visual quality compared to existing methods.

[13] Pix2pix, U-Net generator and CGANs discriminator, combines an adversarial loss with a L1 loss for capturing the low frequencies. Their model progressively downsamples the input images from a high-resolution grid to a bottle neck layer of skip connectors (concatenates the channels from previous layers) followed by U-Net. Their algorithm proved to be effective as the PatchGAN runs patches that could capture high frequencies on image structure. For algorithm's optimization, they had used minibatch SGD and Adam solver, with a 0.0002 LR, and $\beta1 = 0.5$, $\beta2 = 0.999$ momentum parameters. Also, various batch normalization had been used, ranging from 0 to 10. They trained their CGANs on different tasks and datasets, including Cityscapes, GMP Facades, Google Maps photos, BW to color photos, edges of photos, human-drawn sketches, day to night images, thermal to color, and finally missing pixels photos top inpainted photo datasets. They declared even though with small training size; they were able to achieve a descent result on a single Pascal Titan X GPU. As a limitation to measure structure losses in applying traditional metrices as per-pixel mean-squared error, they applied joint method. First, a test of map generation, aerial photo generation, and image colorization conducted for solving graphic problems on AMT. Secondly, a metric system used for recognizing the realistically match of objects in images. As a conclusion, L1 loss, CGAN, and L1+CGAN achieved on cityscapes 86%, 76%, and 83% per-pixel accuracy, and 42%, 28%, and 36% per-class accuracy, respectively.

[14] UNIT combines VAEs with CoCAN, where two generators share weights to learn the shared distribution of images in cross domain. Shared-latent space implies CC mapping between source and target domains interchangeably. They employed weight-sharing constraint to relate the weight of last layers of VAEs that are responsible for extracting encoded high-frequencies and sharing them for decoding. They used ADAM with 0.0001 LR and momentums set to 0.5 and 0.999. The batch size of one image from different domains. Their framework has different parameters of $\lambda 0 = 10$, $\lambda 3 = \lambda 1 = 0.1$ and $\lambda 4 = \lambda 2 = 100$. The encoders consisted of 3 convolutional layers as the front-end and 4 basic residual blocks. The generators consisted of 4 basic residual blocks as the front-end and 3 transposed convolutional layers as the back-end. On the other hand, the discriminators consisted of stacks of convolutional layers with nonlinear LeakyReLU. They used the map dataset, which contained corresponding pairs of images in two domains (satellite image and map). it operated in an unsupervised setting using 1096 satellite images from training set as the first domain and 1098 maps from the validation set as the second domain for $100K$ iterations. The corresponding ground truth maps pixel-wisely where A pixel translation was counted correct if the color difference was within 16 of the ground truth color value. They used the average pixel accuracy over the images in the test set as the performance metric. The algorithm achieved classification accuracies SVHN→MNIST 0.9053, MNIST→USPS 0.9597, USPS→MNIST 0.9358.

[15] DiscoGAN and [16] CycleGAN save information attributes among input and translated images by using a CC loss. However, they all suffered from limited generalization as they are SISO models.

On one hand, DiscoGAN, a cross-domain GANs without any explicit pre-training sets, takes an input image in one domain and generate its correspondence in the other. Their architecture has coupled GANs that force generated image to be a duplicate of the original one, for example, a handbag image, through a reconstruction loss, must be as much as close to images in the shoe domain. In experiment, all input and translated images were $64 \times 64 \times 3$ size and 0.0002 LR, 200 minibatch size, and Adam optimizer with $\beta 1 = 0.5$ and $\beta 2 = 0.999$. Besides, batch normalization was applied to all convolution and deconvolution layers except the first and the last layers. Also, weight decay coefficient varies between $10 - 4$. Data images vary in azimuth rotation from $-90\ ^\circ$ to $+90\ ^\circ$. Their experiments summarized on car to car and face to face. Car dataset consists of rendered images of 3D car models with varying azimuth angles at $15\ ^\circ$ intervals. For evaluation, they translated images in test set and their azimuth angles were predicted using a regressor. The proposed model shows strong correlation between predicted angles of both input and translated images, which successfully discovers azimuth relation between two domains. Between a standard GAN and a reconstructive loss GAN, the generated images do not vary as much as the input images in terms of rotations. In car experiment, the two models suffered from sudden model's collapse. They experimented face conversion by face attributes on CELEBA, facescrub datasets, where only one feature, such as gender or hair color, varies between two domains and all the rest are shared. The translated results have not only similar colors and patterns, but they also have similar level of fashion formality as the input fashion item. Future works will be directed on algorithm modification in handling mixed modalities (e.g., text and image).

On the other hand, CycleGAN, using two CC losses that capture image translation from one domain to another and vice versa, contains three convolutions, residual blocks, and two

fractionally-strided convolutions with ½ stride, and a convolution that maps feature to RGB. Their network contains 6 blocks for 128×128 images and 9 blocks for 256×256. For the discriminator networks they used 70×70 PatchGANs with instance normalization. They used different techniques to stabilize their model. Firstly, LGAN replaces negative log like-hood objective through least-squares loss. Secondly, a discriminator oscillation updates use a history of generated images rather than the ones produced by the latest generators. Important parameters are $\lambda = 10$, 0.0002 Adam LR, batch size of 1, and a linear decay to zero over 100 epochs. The experiments achieved positive results but failed on some translations tasks involving color, geometric, and texture changes, as for example, cat to dog or horse, and a rider on a horse to zebra. Also, semantic weakness was spotted in photo labelling tasks, therefore, translation ambiguity form substantial boundaries. Moreover, the model achieved real labels on Map to Photo $26.8\% \pm 2.8\%$, and $23.2\% \pm 3.4\%$ on Photo to Map. Besides, it achieved 58% per-pixel accuracy, 22% per-class accuracy, and 16% mean class IOU on labels to photo in Cityscapes dataset.

Table 2- Articles Summarization in respect to study number, dataset, used approach, experiments results, challenges and possible future works

Study	Datasets	Approach	Results	Problems and Challenges	Future work
[3]	**Description:** BRATS2015 has 274 subjects of 4 modalities: T1, T1c, T2, and Flair. The size of each image: $240 \times 240 \times 155$. **Pre-processing:** - $72 \times 72 \times 72$ images - $48 \times 48 \times 48$ regions **Validation set:** - 5-fold CV	**Models:** - 12 SISO models and a unified GAN **Configurations:** - Loss parameters: $\alpha = 2$ $\beta = 5$ $\gamma = 10$ $\delta = 2$ $\mu = 0.1$ - Evaluation metrics: NMAE PSNR SSIM VIF NIQE	T1 as input: - NMAEs: $0.034 \pm$ T1c $0.041 \pm$ T2 $0.041 \pm$ Flair - PSNRs: $32.35 \pm$ T1c $30.016 \pm$ T2 $29.09 \pm$ Flair - SSIMs: $0.974 \pm$ T1c $0.969 \pm$ T2 $0.959 \pm$ Flair - VIF: $0.750 \pm$ T1c $0.706 \pm$ T2 $0.65 \pm$ Flair - NIQE: $1.396 \pm$ T1c $1.511 \pm$ T2 $1.259 \pm$ Flair	- Spatially co-registered multimodal images - Small dataset	- Data augmentation techniques
[2]	**Description:** - CelebA dataset contains 40 labels of facial features such as hair color, gender, and age. - RaFD dataset contains 8 labels for facial expressions as like happy, angry, and sad. **Pre-processing:** - Mask vector of domain label	**Models:** - A unified GAN of multiple one-to-one discriminator generator	- ACC on CelebA: 66.2% Hair color 39.1% Gender 70.6% Aged 47.4% H+G 61.5% H+A 49.8% G+A 52.2% H+G+A - Loss on RaFD: 2.12	Not specified	Not specified

Study	Datasets	Approach	Results	Problems and Challenges	Future work
[13]	**Description:** - Cityscapes - GMP Facades - Google Maps - BW to color photos, edges of photos - Humandrawn sketches - Day-to-night images - Thermal to color - Missing pixels photos - Top inpainted photos **Pre-processing:** - Mapping generation - Image colorization - Metric system	**Models:** - U-Net generator and CGANs discriminator **Configuration:** - adversarial and L1 losses - Downsampling - Minibatch - 0.0002 LR SGD and Adam - 0.5 $\beta 1$ & 0.99 $\beta 2$ Momentum - $0-10$ batch size	L1 loss, CGAN, & L1+CGAN accuracies on cityscapes: -Per-pixel: 86% 76% 83% -Per-class: 42% 28% 36%	- Small data size - Structure losses of traditional metrices as per-pixel mean-squared error	Not specified
[14]	**Description:** Map dataset contains pairs of images in two domains: satellite image and map. **Training set:** 1096 satellite images as first domain **Validation set:** 1098 maps as second domain	**Models:** -VAEs -CoCAN -CC loss **Configurations:** - Weight-sharing -0.0001 ADAM LR $-0.5-0.999$ momentums - LeakyReLU - 1 batch size - A pixel translation - $100K$ iterations - Averaged pixel accuracy - Loss parameters: $\lambda 0 = 10$ $\lambda 3 = \lambda 1 = 0.1$ $\lambda 4 = \lambda = 100$	Classification ACC: - SVHN→MNIST: 0.9053 -MNIST→USPS: 0.9597 -USPS→MNIST: 0.9358	Not specified	Not specified
[15]	**Description:** -Car dataset consists of rendered 3D car models with varying azimuth angles at 15 ◦ intervals -CELEBA - Facescrub has many face features such as gender or hair color **Pre-processing:** - $64 \times 64 \times 3$ images	**Models:** - Reconstruction loss. Coupled GANs **Configurations:** - 0.0002 LR - 200 minibatch - batch normalization - $10-4$ weight decay coefficient -Adam optimizer: $\beta 1 = 0.5$ $\beta 2 = 0.999$	- Discovering azimuth relation between 2 domains	- Limited generalization of SISO model - Sudden model collapse	- Algorithm modification in handling mixed modalities (e.g., text and image)

Study	Datasets	Approach	Results	Problems and Challenges	Future work
	- Azimuth rotation from -90 ◦ to $+90$ ◦				
[16]	**Description:** - Cityscapes - Map to Photo	**Models:** - 70×70 PatchGANs - CC loss - 3 convolutions - Residual Blocks - 2 fractionally strided convolutions - ½ stride - RGB convolution - 6 blocks for 128×128 images - 9 blocks for 256×256 images **Configurations:** - Instance normalization - least-squares loss - 0.0002 Adam LR - $\lambda = 10$ - 1 batch size - linear decay to zero over 100 epochs	- Map to Photo: $26.8\% \pm 2.8\%$ - Photo to Map: $23.2\% \pm 3.4\%$ - Cityscapes: 0.58 per-pixel accuracy 0.22 per-class accuracy 0.16 mean class IOU	- Limited generalization of SISO model - Semantic weakness resulting in translation ambiguity - Translations Failure on color, geometric, and texture changes	Not specified

In conclusion, from literature, we could notice a research focus on SIMO method, as we have discussed from the following articles [2][3][15][16]. From there, we devoted our research to introduce MISO architecture, inspired from [2][3], takes the advantages of the leading performance to predict a sole attribute from multiple input.

3. Background

In this section, we would discuss the DL in the context of medical diagnosis in highlighting the GANs architectures as well as detailed description of the BRATS2018 dataset.

3.1 Deep Learning in Medical Diagnosis

AI and ML are the hope in changing the medical diagnosis situation in avoiding acknowledged mistakes and harmful medical errors. They would better the diagnosis of symptoms, which are tricky to spot even by best experts.

DL, as a one part of the broader family of ML methods, based on ANNs with representation learning in reinforcement, supervised, semi-supervised, and unsupervised methods [10]. DL architectures such as DNNs, RNNs and CNNs have been applied to fields including computer vision, NLP, MT, and medical image analysis, where they have produced competitive results that would surpass human's performance [8].

In general, ANNs were inspired by information processing and distributed communication nodes in biological systems. ANNs have differences from biological brains in the tendance to be static and symbolic, while the biological brain tends to be dynamic and analogue. However, the main core of deep learning refers to the use of multiple layers to progressively extract higher-level features from a raw input. For example, in image processing, lower layers may identify edges, while higher layers may identify the concepts relevant to a human such as digits or letters or faces.

Nowadays, DL remains the most promising and widely used ML methodology for radiology and disease detection in general. It comes as no surprise as diagnostic imaging prevails in clinical diagnosis and image recognition. In 2016, Geoffrey Hinton, a notable computer scientist and researcher, predicted that radiologists and specialists who diagnose diseases from medical imaging like X-rays, CT scans, and MRI; they would soon lose their jobs. “People should stop training radiologists right now,” he announced, “It’s obvious that within five years deep learning is going to do better than humans,” [8].

The advancement of DL makes it much useful but not much reliable to the level that Hinton indicted by which machines would replace human experts. However, DL is still used to support the doctors in pre-selection and prioritize cases, but not as a main tool to diagnose patients.

From here, there are diverse use of DL in radiology and other diagnostic practices:

1. Detecting the Neurological Abnormalities
2. Screening of the Common Cancers
3. Identifying the Infections in Kidney & Liver
4. Brain Tumor Detection with High Accuracy
5. Dental Imaging Analysis
6. Detecting the Bone Fractures and Musculoskeletal Injuries

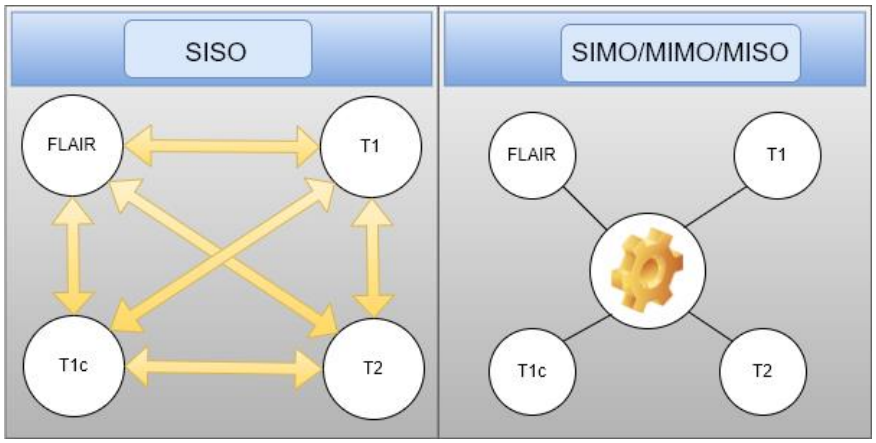
Medical image synthesis is an alternative to multiple pulse sequences for acquiring multiple contracts MRI [3]. MRI scan is non-standardized process across the different institution, cross-modal image synthesis proposed to tackle this challenge in providing missing modalities. It is used in clinical practice due to its capacity in providing useful information, for example the four contracts:

- T1 distinguish white and grey matters.
- T1c assess the change of tumor shape with enhanced demarcation around tumor.
- T2 shows fluid obviously from cortical tissue.
- Flair shows contours of lesion.

In research and medical practices, there are different architectures could be considered whenever applying DL. They are summarized in the following list and Figure (5):

1. SISO: a target image generated from a given source image.
2. MISO: overcome limitations of SISO when source and target images are weakly correlated through learning shared latent representations.
3. MIMO: synthesizing one or more modalities from an input of MRI modalities.
4. SIMO: where only single modality is available as input, but multiple contrasts are necessary in output.

Figure 5- SISO, SIMO, MISO, and MIMO methods in handling MRI modalities



MRI of different modalities could provide complementary information for medical diagnosis, but also it is challenging and costly expensive to access all the modalities. Many methods focus on single modality to a modality synthesize, which marks a grand limitation in generalizing the outcomes to other modalities. Therefore, for each two modalities, we should develop a separate model so we could be able to map them together. To address this challenging problem, we propose a multiple modalities GANs to synthesize three MR modalities (FLAIR, T1 and T1c) from one MR modality T2. Another challenging area is predicting Flair or T1c, as avoiding injecting substance to the patient's brain, from different input at once T1 and T2 or T1, T2, Flair. The proposed architectures will be detailed later in **4.2 Algorithm**.

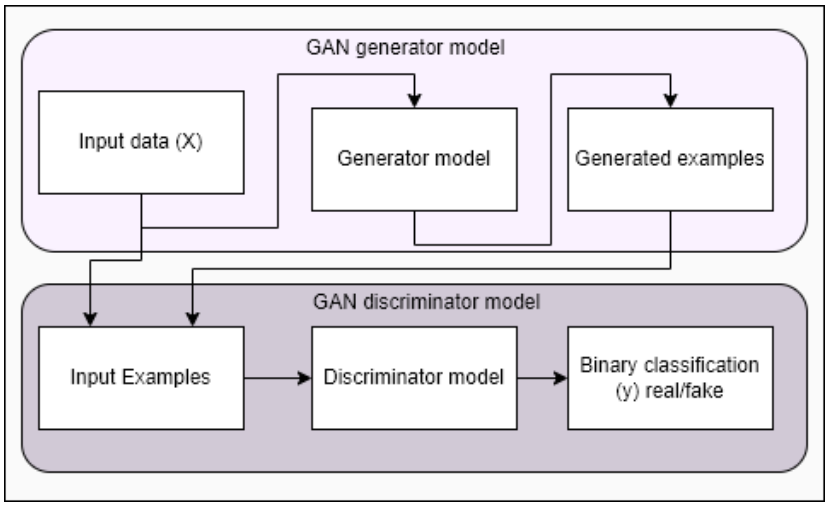
3.2 Generative Adversarial Networks (GANs)

By specifying a high-level target, in producing indistinguishable results from reality, and then automatically learning an appropriate loss function to satisfy that goal, is exactly what GANs do. The loss learning GANs tries to classify whether the output image is real or fake by training a generalized model to minimize this loss. Therefore, blurred images will not be accepted as they will obviously look fake. Because GANs learn an adaptive loss to data, they can be applied to a multitude of tasks that traditionally require different types of loss functions. The adversarial loss makes the key success of GANs, which basically force generated images indistinguishable from real ones.

As a definition, GANs are generative modeling approach using DL methods as in CNNs. In practice, GANs train a generative model by two sub-models: the generator model at one hand, learn to generate new examples by mapping the modalities of the networks' inputs as on the other hand to feed it to discriminator model in classifying the examples as either real or fake.

As we could see in the following figure, GAN generator model is considered as unsupervised learning problem as it generates a batch sample. In return, these entities with the real examples fed into the GANs model. Then, the discriminator model, as considered a supervised learning problem, got updated based on its performance of classification of the sample to either real or fake. The whole concept of the GANs is represented in the following Figure (6):

Figure 6- GANs concept



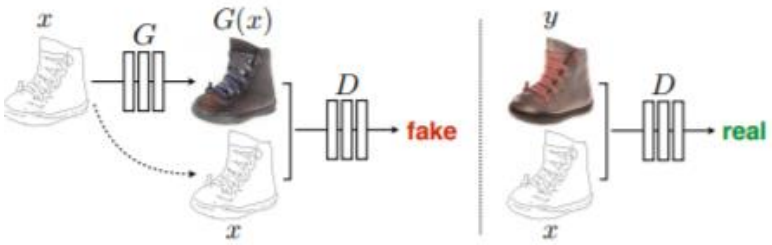
On the other hand, the unified GAN of single generator and discriminator tries to map images from four modalities. Briefly, the generator takes an image with its modality as input to synthesize it to a target modality. The discriminator is capable to differentiate between real and synthesized images on their corresponding modalities.

In mathematical terms, $G: X \rightarrow Y$ such that the distribution of images from $G(X)$ is indistinguishable from the distribution Y using an adversarial loss mapping. GAN learns a mapping from the observed image X and the random noise Z, to y label through $G: \{X, Z\} \rightarrow Y$ [13]. Generator G is trained to produce outputs that are indistinguishable from "real" images by a highly trained discriminator D. G is trained to best generate "fake" images and D tries the best to distinguish the real from fake till they become alike. The learning process is highly under-constrained; we couple it with an inverse mapping $F : Y \rightarrow X$ and introduce a CC loss to enforce $F(G(X)) \approx X$, and vice versa. To keep in mind, there are two different CC losses computations:

1. Forward CC loss: $x \rightarrow G(x) \rightarrow F(G(x)) \approx x$
2. Backward CC loss: $y \rightarrow F(y) \rightarrow G(F(y)) \approx y$

This training procedure is mathematized in Figure (7).

Figure 7- Demonstration of generator-discriminator mapping, which extracted from [13]



U-NET, an image segmentation, helps to reduce the volume of data. The U-Net is composed of two channels. The first channel resembles to the encoder as it captures the context of the image. On the other hand, the decoder is a transposed convolution that works to construct the original

image based on image segments extracted in the process. Basically, the network assembles layers of convolution followed by max pooling later that reduce image density as to reduce the training parameters of network.

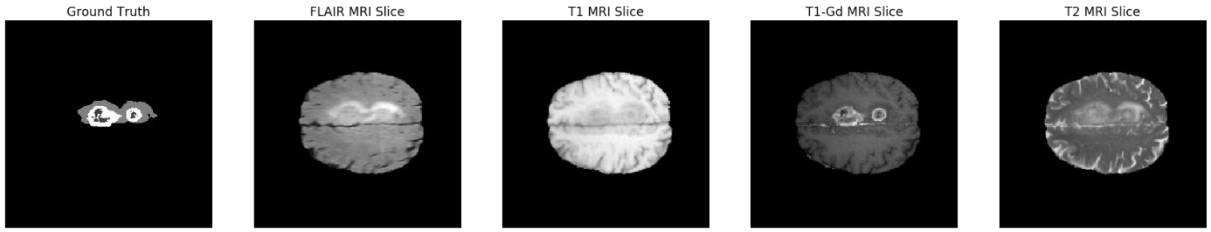
3.5 Dataset

BRATS2018 contains different multimodal scans, available as NIfTI files (.nii.gz), are native (T1), post-contrast T1 weighted (T1c), T2 weighted (T2), and T2 fluid attenuation, which describes Invert recovery (Flair) volumes. They were obtained using different clinical protocols and different scanners from multiple institutions [5][6].

All images were manually segmented by 1 to 4 evaluators using the same annotation protocol, and these annotations were finally approved by an experienced neuroradiologist. The data distributed after simultaneous registration using the same anatomical template, interpolation to the same resolution 1 mm^3, and pretreatment of skull stripping.

The following Figure (8) represents four different modalities along with its segmentation map.

Figure 8 – Data sample demonstrating the four modalities with segmentation map



The dataset contains 285 samples of each modality, distributed over 57 patients each. The following Table (3) summarize the dataset distribution.

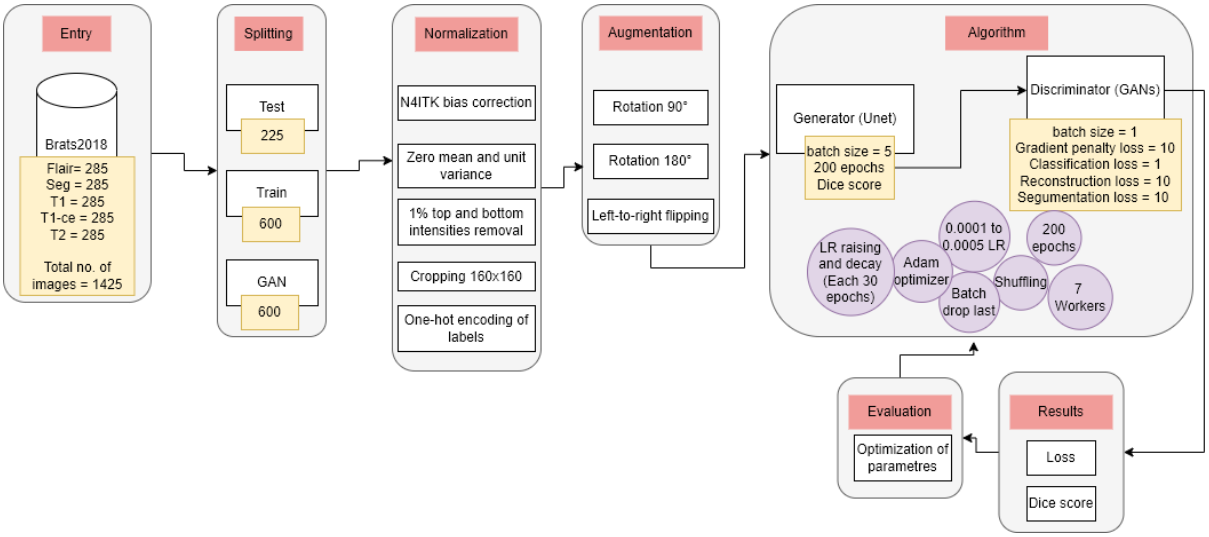
Table 3- BRATS2018 modality distribution over 57 patients

Modality	Distribution
T1	285
T2	285
Flair	285
T1c	285
Seg	285
Total	1425

4. Approach

This section specifies the different steps of the architecture by shading light on data input, split, normalization as well as augmentation techniques, generator-discriminator algorithm with the generated results, and finally evaluation and parameters optimization. The schema is presented in Figure (9).

Figure 9- Program architecture, specifying the different steps of data entry, pre-processing and augmentation, algorithm, results and evaluation metrices



For demonstration, the data is split into 80% training and 20% testing. Then the images are normalized by applying 5 different techniques: bias correction, normalization, intensities removal, cropping and finally one-hot encoding. Later in the process, the normalized images are augmented by applying 3 supplementary techniques: left-to-right flipping, 90° and 180° rotations.

The algorithm, therefore, consists of a generator that takes the input to generate fake modalities for the discriminator to classify into two classes either fake or real. The more the algorithm is unable to distinguish the difference between the reality and the illusion, the algorithm tends to achieve merely perfection.

The generator and the discriminator share certain parameters in common, as detailed in the following:

- Adam optimizer with $0.0001 - 0.0005$ LR decay and raising each 25 epochs.
- 7 computational workers.
- Batch drop last is enabled to ignore the last batch in case the number of examples is not divisible by the batch size.
- Data shuffling.
- 200 Epochs.

On the other hand, the generator and discriminator have different parameters. First, the generator has the following unique parameters:

- 5 batch size.
- Dice score.

The discriminator distinguishes certain parameters, different from the U-Net generator, detailed in the following:

- 1 batch size.
- 10 Gradient penalty loss

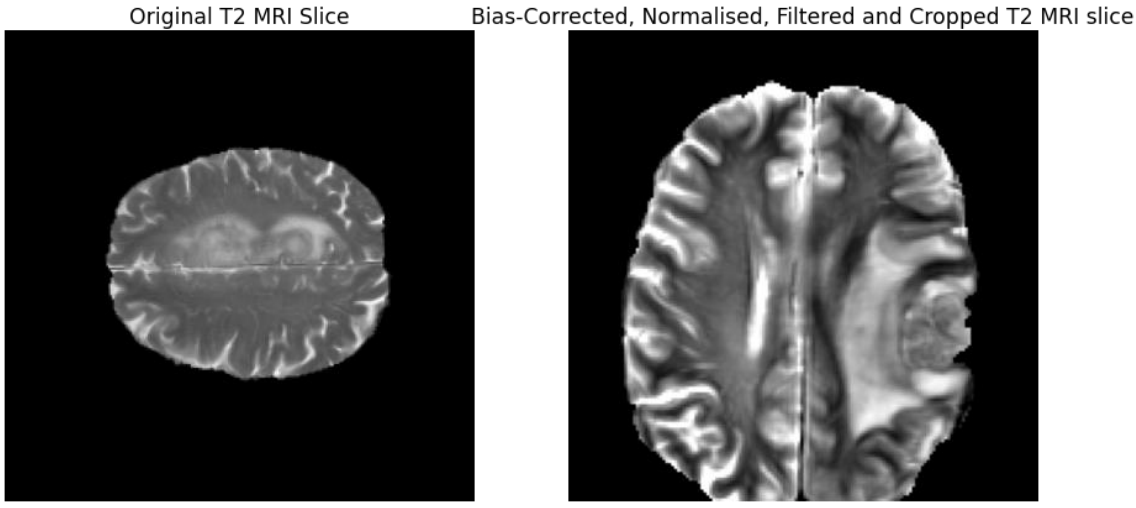
- 1 Classification loss
- 10 Reconstruction loss
- 10 Segmentation loss

After the training process, the results are assessed in considering the loss as well as the dice score. At the end, we adopted parameter optimization based on the performance of each training process. Hence, an overall enhancement is evidently released in optimization process.

4.1 Pre-processing

One-hot encoding of labels, 1% top and bottom intensities removal, zero mean and unit variance, N4ITK bias correction, and $160 \times 160 \times 155$ cropping. A result of the data pre-processing is represented in the following Figure (10).

Figure 10 – The outcome of the pre-processing step, including bias-correction, normalization, filtration, and size cropping



4.1.1 Image Cropping

The entry image is $240 \times 240 \times 155$ (Height, Width, Depth) cropped to $160 \times 160 \times 155$.

4.1.2 One-hot encoding

The labels of the images are transferred to one-hot encoding, which is basically putting 1 whenever there is a tumor, otherwise, it is noted with zero value. The representation of encoding is represented in the following Table (4):

Table 4- One-hot encoding of tumor and non-tumor labels

Patients	Category
Patient 1	Tumor
Patient 2	Non-tumor
...	...

→

Patients	Label
Patient 1	1
Patient 2	0
...	...

4.1.2 Normalization

In the normalization step, the algorithm removes the top and bottom 1% of intensities and then normalize image with zero mean and unit variance. Therefore, feature standardization ensures that the mean of the values of each feature in the data is zero (if the mean is subtracted by the

numerator) and has a unit variance [18]. This method is often used to determine the distribution mean and standard deviation of each characteristic. Then subtract the average from each feature. Then divide the value of each feature (the mean has already been subtracted) by its standard deviation.

$$x' = \frac{x-\bar{x}}{\sigma},$$

where x is the original feature vector, $\bar{x} = \text{average}(x)$ is the mean of feature vector, and σ is its standard deviation.

Furthermore, the non-background regions should be normalized since intensities range from 0 to 5000. The minimum in the normalized slice corresponds to 0 intensity in unnormalized slice, which replaced with -9 to keep track of 0 intensities.

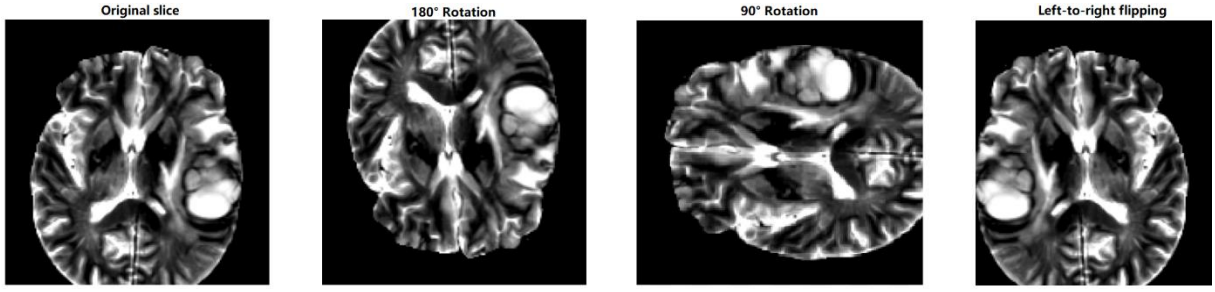
4.1.3 Bias correction

The N4 bias field correction algorithm corrects the low frequency intensity inhomogeneities present in the MRI image data known as biased or enhanced fields. This method has been reported to have been successfully applied as a flat field correction for microscopy data. Furthermore, it starts with a simple parametric model and does not require tissue classification [17].

4.1.2 Data augmentation

Data augmentation is adopted as the dataset is considered a small one for implementing DL and especially generative networks. Therefore, we adopted different augmentation techniques making use of defined functions and the help of TensorFlow through applying 90° and 180° rotations and left to right flipping of the original input image. For example, the following Figure (11) represents a sample of data augmentation that is been applied to <<Brats18_TCIA01_390_1_78 >> slice.

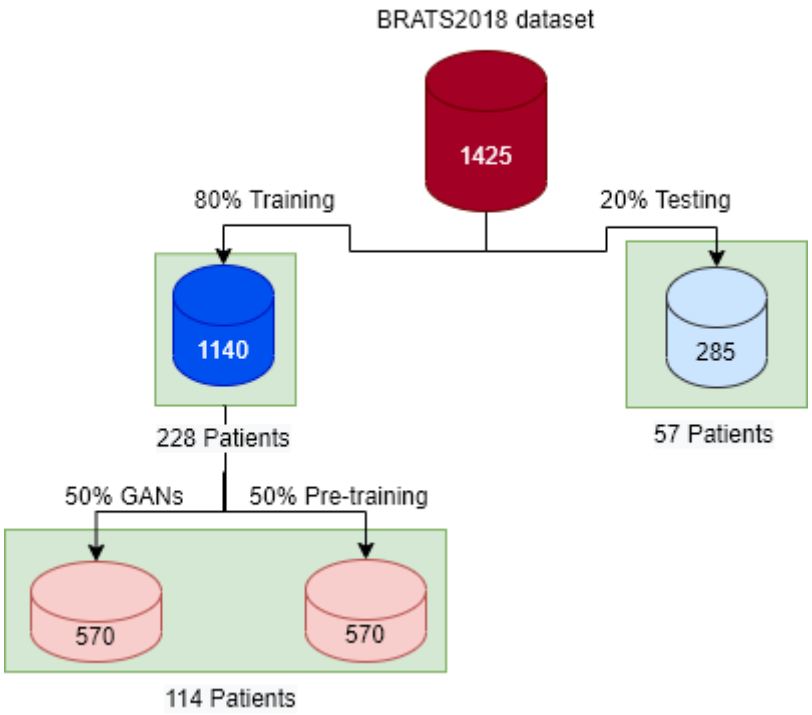
Figure 11- Data augmentation, including: 90° and 180° rotations and left-to-right flipping



4.1.2 Data split

The training data are split into two sets of training and testing sets. Firstly, the training set is 80% of the training dataset and the testing set is 20%. Therefore, the training set contains 1140 images for each generator and discriminator and the testing set includes the rest of images that are 285 pictures in total. The splitting process is detailed and well documented in the following Figure (12):

Figure 12- Data split into testing and training sets



The dataset is split into two parts: 80% training (around 228 patients) and 20% testing (around 57 patients) tests. As we could see from Figure (12), the testing set contains 285 samples divided by 5 modalities: T1, T2, T1c, Flair, and Seg. On the other hand, 1140 training samples are divided into two equal sets. On is for pre-training the U-Net and the other half is for training the whole architecture of GANs.

4.2 Algorithm

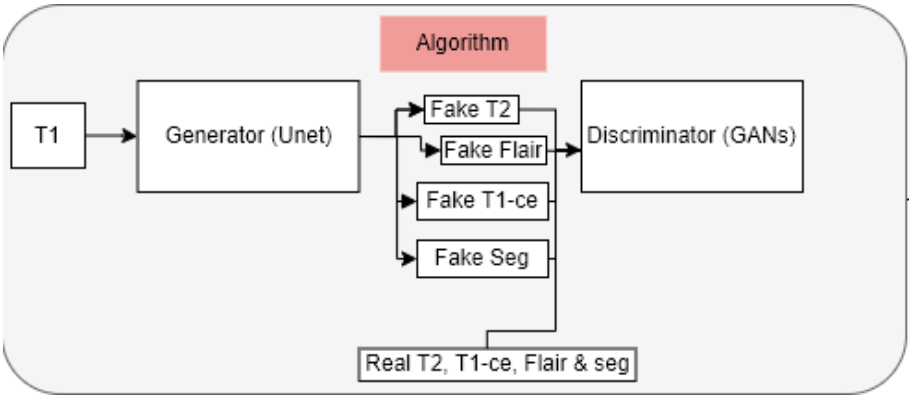
With the existence of many deep CNNs models achieving competitive results in terms of the state-of-the art, I chose U-Net and GANs to fine-tune the dataset in our hands for its efficiency and simplicity. The U-Net, as well described before, handles the segmentation mapping while the GANs algorithm plays the role to discriminate the generated fake images later in the training process.

In the following section, we would present the different algorithm methods that we develop to address the research gap in the literature by 3 different methods, but we would narrow down the experimentation at the end on SIMO and MISO methods as we are aiming at predicting Flair and T1c.

4.2.1 SIMO

SIMO architecture takes a single input, for example, in our case T2 to predict the rest of the modalities by passing by U-Net generator and GAN discriminator. The architecture is well presented in the following Figure (13):

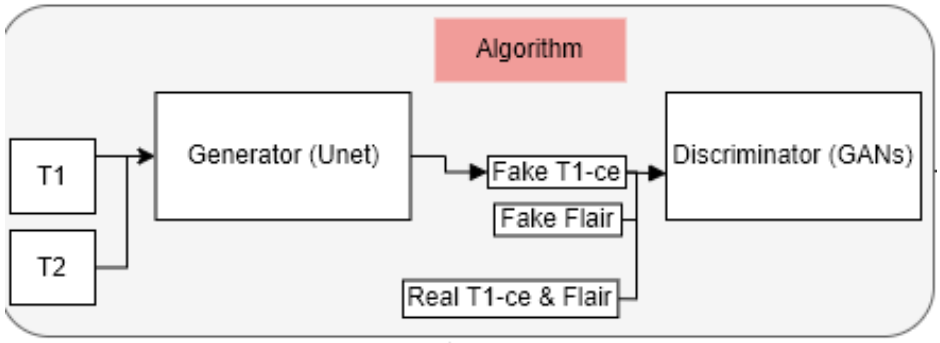
Figure 13- SIMO architecture takes T1 as input, & synthesis T2, Flair, T1c, and segmentation map for discrimination



4.2.2 MIMO

MIMO architecture takes multiple input, for example, in our case T1 and T2 to predict the rest of the modalities by passing by U-Net generator and GAN discriminator. The architecture is well presented in the following Figure (14):

Figure 14- MIMO architecture takes T1 and T2 as input, and synthesis T1c and Flair for discrimination



4.2.3 MISO

MISO architecture takes multiple input, for example, in our case T1 and T2 to predict Flair or T1c by passing by U-Net generator and GAN discriminator. The architecture is well presented in the following Figures (15)(16):

Figure 15- MISO takes T1 and T2 as input, and synthesis Flair for discrimination

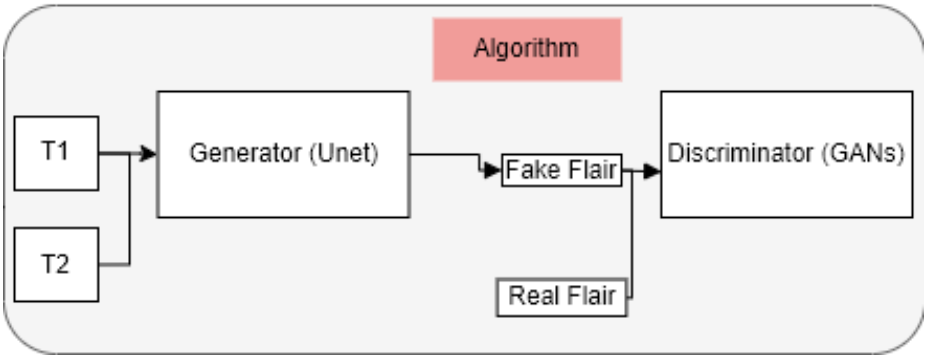
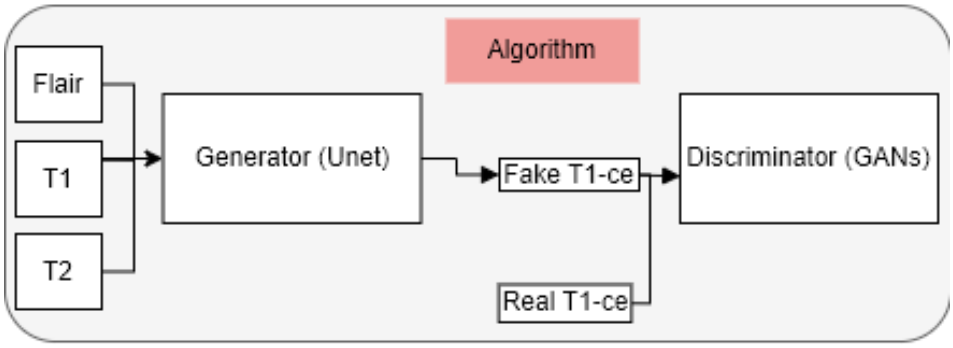


Figure 16- MISO takes T1 and T2, and Flair as input, and synthesis T1c for discrimination



4.3 Evaluation Metrics

The generator U-Net, predicting segmentation map and down-sampling, evaluated by dice score. The GAN discriminator is evaluated by different loss weights, detailed in the following list:

- Gradient penalty
- Classification loss
- Reconstruction loss
- Segmentation loss

5. Experiments and Results

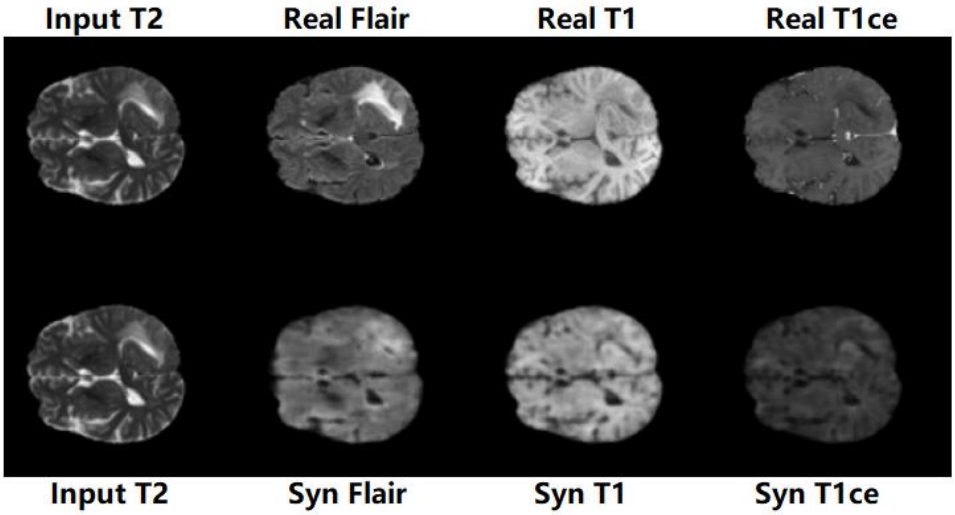
Training the GANs on 55 data examples in specifying these data splitting as 10 tests, 22 train, 23 GAN samples, we have obtained the following results, for first experiment, at the 124 Epoch.

Table 5- Results of GANs on a small test sample of BRATS2018 dataset

d_loss	g_loss	s_loss	d_real	d_fake	d_cls	d_gp	g_fake	g_cls	g_rec	g_seg
-2.23	243.7	0.02	-4.7	2.21	0.1	0.01	-2.2	0.68	243.2	0.023

In Figure (17), we could use a concrete test of our trained model in real and synthesis image. We could notice the generated images are blurry; it is just because we diminished the quality of the images as this experiment run on a low-computational graphic card (RTX 2060).

Figure 17- Results of GANs on unseen example



In the next step, we are going to enhance the results by running a full test of the GANs on the computational resource of the lab, using Quadro RTX 4000.

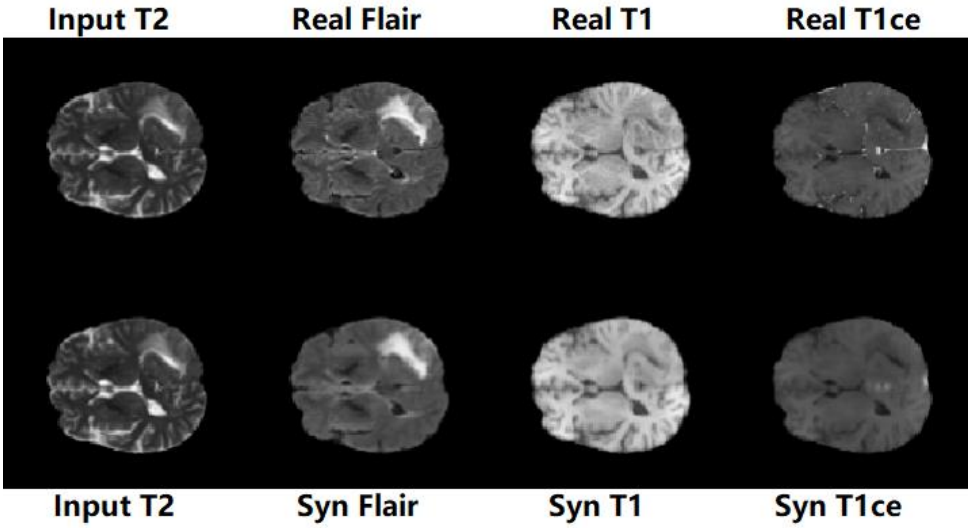
The second experiment is run on 285 data examples, segmented into 35 test sample, 125 train, and 125 GAN samples. The achieved results on 200 epochs:

Table 6- Results of GANs on a small test sample of BRATS2018 dataset

d_loss	g_loss	s_loss	d_real	d_fake	d_cls	d_gp	g_fake	g_cls	g_rec	g_seg
-2.56	203	0.03	-2.05	-0.74	0.006	0.02	0.75	1.13	202.71	0.032

In Figure (18), we used the same concrete test, presented in Figure (17), to assess our trained model in real and synthesis image.

Figure 18- Full dataset Results of GANs on unseen example



6. Conclusion

This section is devoted for the research conclusion. It includes the two subsections of professional and personal conclusions.

6.1 Professional

By the art deep learning, I was able to experiment different settings of architecture on dataset. Through experiments, I have also found that for this dataset, fine-tuning the whole model not only gives better result but also helps the model converges much faster than fine-tuning the rest of the layers.

6.2 Personal

The internship opportunity I had with XLIM laboratory was a great chance to further my learning and professional careers in image processing (precisely, on medical imaging and bioengineering), ML, and DL. Thus, I do consider myself as lucky individual for the having the right position in data and ML engineering. I am so grateful for having the chance to practice many grateful resources of knowledge and skills of independency, research, and personal development. I am with great confidence I could say I am lucky to meeting different specialists in health sector as well as in computer science, including DL and ML. In general, they have led

me of course to the right and desired direction through the internship period from March 28 to September 1 in the year 2022.

6.3 Future Works

We could consider experimenting different loss functions, hyper-parameter optimization, deeper generative networks, and data augmentation, too.

6.4 Challenges

I have faced many difficulties in terms of setting up the Anaconda environment, which was totally not suitable for starting up the proposed CNNs algorithm. Also, the limited computational resources are the most difficult challenge because I could not upgrade my personal laptop to improve the whole results of the research project. The following table (7) specifies the specifications of the laptop that we have tested the architectures on.

Table 7- Laptop specification in which DL architectures trained and tested

Name	MSI GL75 Leopard 10SER
Central processor unit	Intel Core i7-10750H processor (Hexa-Core 2.6 GHz/5 GHz Turbo - 12 Threads - Cache 12M)
RAM	16 GB of 2666 MHz DDR4 RAM (2x8 GB)
Graphic card	NVIDIA GeForce RTX 2060 graphics chip with 6 GB of dedicated GDDR6 memory

6.5 Risk Analysis

In this section, I will present the encountered risks during the whole period of the internship.

6.5.1 Administration user non-grant

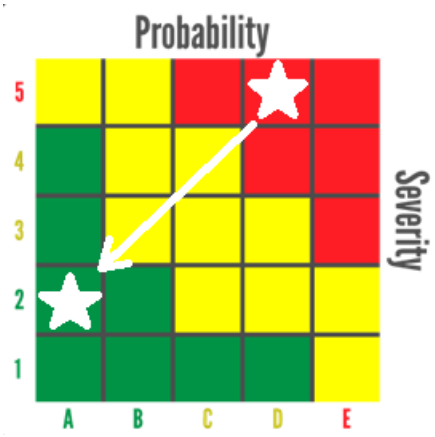
In XLIM laboratory, the administrative right is not given to an intern, for a temporary period. The problem is having no rights to install the applications by myself as well as managing programming environment seemed impossible. From that, I have taken the following solutions to demine the risk:

1) Using my personal computer to conduct the prototype test.
2) Installing applications with *[1].tar extensions to avoid the administration rights.
3) Acquiring two different machines at distance to finalize the final prototypes.
4) Working with python virtual environment as anaconda asks a lot for admin privileges.

After taking in account the above solutions, therefore, the risk diminished substantially as we could visualize in the following Figure (19):

[1] The name of the application

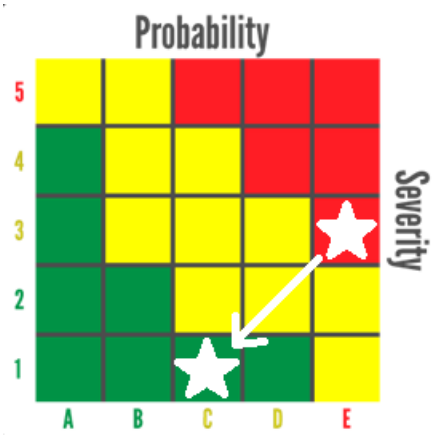
Figure 19- Risk probability/severity matrix for risk of administration user non-grant



6.5.2 The ambiguity of specification

At the beginning of the internship, the specification demanded during the whole period of stage was a bit too broad and not narrow to a focus research area. Later, after a meeting with the educational tutor, he set me up to organize a whole planning map for the whole internship. For that, I have done GanttProject as well as PERT to overshadow the forthcoming tasks and the current tasks at hand. Henceforth, the gravity of the problem reduced with a lower probability.

Figure 20- Risk probability/severity matrix for risk of specification ambiguity



7. References

[1] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. In: Navab, N., Hornegger, J., Wells, W., Frangi, A. (eds) Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. MICCAI 2015. Lecture Notes in Computer Science, vol 9351. Springer, Cham. https://doi.org/10.1007/978-3-319-24574-4_28

[2] Choi, Y., Choi, M., Kim, M.S., Ha, J., Kim, S., & Choo, J. (2018). StarGAN: Unified Generative Adversarial Networks for Multi-domain Image-to-Image Translation. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 8789-8797.

[3] Dai, X., Lei, Y., Fu, Y., Curran, W.J., Liu, T., Mao, H., & Yang, X. (2020). Multimodal MRI Synthesis Using Unified Generative Adversarial Networks. Medical physics. https://doi.org/10.1002/mp.14539

[4] Menze et al. (2015). The Multimodal Brain Tumor Image Segmentation Benchmark (BRATS). IEEE transactions on medical imaging, 34(10), 1993–2024. https://doi.org/10.1109/TMI.2014.2377694

[5] Bakas, S., Akbari, H., Sotiras, A., Bilello, M., Rozycki, M., Kirby, J.S., Freymann, J.B., Farahani, K., & Davatzikos, C. (2017). Advancing The Cancer Genome Atlas glioma MRI collections with expert segmentation labels and radiomic features. Scientific Data, 4.

[6] Bakas et al. (2018). Identifying the Best Machine Learning Algorithms for Brain Tumor Segmentation, Progression Assessment, and Overall Survival Prediction in the BRATS Challenge. ArXiv, abs/1811.02629.

[7] Xlim Présentation. (n.d.). Retrieved May 10, 2021, from https://www.xlim.fr/laboratoire/presentation

[8] Standford University. (2017). RSNA 2017: Rads who use AI will replace rads who don't. Retrieved May 3, 2021, from https://aimi.stanford.edu/news/rsna-2017-rads-who-use-ai-will-replace-rads-who-don-t

[9] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition.

[10] Meskó, B., & Görög, M. (2020). A short guide for medical professionals in the era of artificial intelligence. npj Digit. Med. 3, 126. https://doi.org/10.1038/s41746-020-00333-z

[11] Krizhevsky, A., Sutskever, I., & G. E. Hinton. (2012). Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems, pages1097–1105.

[12] K. He, X. Zhang, S. Ren, & J. Sun. (2015). Deep residual learning for image recognition. arXiv preprint arXiv:1512.03385.

[13] Isola, P., Zhu, J., Zhou, T., & Efros, A. (2017). Image-to-Image Translation with Conditional Adversarial Networks. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 5967-5976.

[14] Liu, M.-Y., Breuel, T., & Kautz, J. (2017). Unsupervised image-to-image translation networks. arXiv preprint arXiv:1703.00848.

[15] Kim, T., Cha, M., Kim, H., Lee, J.K., & Kim, J. (2017). Learning to Discover Cross-Domain Relations with Generative Adversarial Networks. ICML. https://doi.org/10.48550/arXiv.1703.05192

[16] Zhu, J., Park, T., Isola, P., & Efros, A.A. (2017). Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks. 2017 IEEE International Conference on Computer Vision (ICCV), 2242-2251.

[17] Tustison, N. J., Avants, B. B., Cook, P. A., Zheng, Y., Egan, A., Yushkevich, P. A., & Gee, J. C. (2010). N4ITK: improved N3 bias correction. IEEE transactions on medical imaging, 29(6), 1310–1320. https://doi.org/10.1109/TMI.2010.2046908

[18] Grus, Joel (2015). Data Science from Scratch. Sebastopol, CA: O'Reilly. pp. 99, 100. ISBN 978-1-491-90142-7.

Annex

CelebA dataset contains 202,599 face images of celebrities, each annotated with 40 binary attributes. They crop the images, then resize them as 128×128. Randomly, they selected 2,000 images as test set, and they used all remaining images for training data. They constructed seven domains using the following attributes: hair color (black, blond, brown), gender (male/female), and age (young/old).

RaFD dataset consists of 4,824 images collected from 67 participants, who made eight facial expressions in three different gaze directions and angles. The images are cropped to 256 × 256, centralized faces, and finally resized to 128 × 128.