THESIS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

Deep Learning Methods for Classification of Gliomas and Their Molecular Subtypes

From Central Learning to Federated Learning

Muhaddisa Barat Ali

CHALMERS

Department of Electrical Engineering
Chalmers University of Technology
Gothenburg, Sweden, 2023

Deep Learning Methods for Classification of Gliomas and Their Molecular Subtypes
From Central Learning to Federated Learning

Muhaddisa Barat Ali
ISBN: 978-91-7905-903-3

Doktorsavhandlingar vid Chalmers tekniska högskola
Ny serie nr 5369
ISSN 0346-718X
This thesis has been prepared using LaTeX.

Department of Electrical Engineering
Chalmers University of Technology
SE-412 96 Gothenburg, Sweden
Phone: +46 (0)31 772 1000
www.chalmers.se

Printed by Chalmers Reproservice
Gothenburg, Sweden, August, 2023

Abstract

The most common type of brain cancer in adults are gliomas. Under the updated 2016 World Health Organization (WHO) tumor classification in central nervous system (CNS), identification of molecular subtypes of gliomas is important. For low grade gliomas (LGGs), prediction of molecular subtypes by observing magnetic resonance imaging (MRI) scans might be difficult without taking biopsy. With the development of machine learning (ML) methods such as deep learning (DL), molecular based classification methods have shown promising results from MRI scans that may assist clinicians for prognosis and deciding on a treatment strategy. However, DL requires large amount of training datasets with tumor class labels and tumor boundary annotations. Manual annotation of tumor boundary is a time consuming and expensive process.

The thesis is based on the work developed in five papers on gliomas and their molecular subtypes. We propose novel methods that provide improved performance. The proposed methods consist of a multi-stream convolutional autoencoder (CAE)-based classifier, a deep convolutional generative adversarial network (DCGAN) to enlarge the training dataset, a CycleGAN to handle domain shift, a novel federated learning (FL) scheme to allow local client-based training with dataset protection, and employing bounding boxes to MRIs when tumor boundary annotations are not available.

Experimental results showed that DCGAN generated MRIs have enlarged the original training dataset size and have improved the classification performance on test sets. CycleGAN showed good domain adaptation on multiple source datasets and improved the classification performance. The proposed FL scheme showed a slightly degraded performance as compare to that of central learning (CL) approach while protecting dataset privacy. Using tumor bounding boxes showed to be an alternative approach to tumor boundary annotation for tumor classification and segmentation, with a trade-off between a slight decrease in performance and saving time in manual marking by clinicians. The proposed methods may benefit the future research in bringing DL tools into clinical practice for assisting tumor diagnosis and help the decision making process.

Keywords: Deep learning, convolutional NN, generative adversarial network, CycleGAN, convolutional autoencoder, glioma subtype classification, 1p/19q codeletion, IDH mutation, federated learning, multi-stream U-Net.

List of Publications

This thesis is based on the following publications:

[A] **Muhaddisa Barat Ali**, Irene Yu-Hua Gu and Asgeir Store Jakola, "Multi-stream Convolutional Autoencoder and 2D Generative Adversarial Network for Glioma Classification". In: Vento, M., Percannella, G. (eds) *CAIP: International Conference on Computer Analysis of Images and Patterns, 2019*, Lecture Notes in Computer Science, Vol. 11678, pp. 234-245, Springer, Cham, https://doi.org/10.1007/978-3-030-29888-3_19.

[B] **Muhaddisa Barat Ali**, Irene Yu-Hua Gu, Mitchel Berger, Johan Pallud, Derek Southwell, Georg Widhalm, Alexandre Roux, Tomás Gomez Vecchio and Asgeir Store Jakola., "Domain Mapping and Deep Learning from Multiple MRI Clinical Datasets for Prediction of Molecular Subtypes in Low Grade Gliomas". *Brain Sciences (2020)*, Vol. 10, No. 7, pp. 463-483, MDPI, https://doi.org/10.3390/brainsci10070463.

[C] **Muhaddisa Barat Ali**, Irene Yu-Hua Gu, Mitchel S. Berger and Asgeir Store Jakola., "A Novel Federated Deep Learning Scheme for Glioma and its Subtype Classification ". *Frontiers in Neuroscience (2023)*, Vol. 17, ISSN. 1662-453X, https://doi.org/10.3389/fnins. 2023.1181703.

[D] **Muhaddisa Barat Ali**, Irene Yu-Hua Gu, Alice Lidemar, Mitchel Berger, Derek Southwell and Asgeir Store Jakola., "Prediction of Glioma-Subtypes: Comparison of Performance on a DL Classifier using Bounding Box Areas Versus Annotated Tumors". *BMC Biomedical Engineering (2022)*, Vol. 4, No. 1, pp. 4-15, Springer, https://doi.org/10.1186/s42490-022-00061-3.

[E] **Muhaddisa Barat Ali**, Xiaohan Bai, Irene Yu-Hua Gu, Mitchel S. Berger and Asgeir Store Jakola., "A Feasibility Study on Deep Learning Based Brain Tumor Segmentation Using 2D Ellipse Box Areas". *Sensors (2022)*, Vol. 22, No. 14, pp. 5292-5304, MDPI, https://doi.org/10.3390/s22145292.

Other publications by the author, not included in this thesis, are:

[F] Eddie de Dios, **Muhaddisa Barat Ali**, Irene Yu-Hua Gu, Tomás Gomez Vecchio, Chenjie Ge and Asgeir Store Jakola, "Introduction to Deep Learning

in Clinical Neuroscience ". *Machine Learning in Clinical Neuroscience: Foundations and Applications*, Springer International Publishing, 2022. Vol. 134, pp. 79-89, https://doi.org/10.1007/978-3-030-85292-4_11.

[G] **Muhaddisa Barat Ali** and Emanuele Olivetti, "Classification-based Tests for Neuroimaging Data Analysis: Comparison of Best Practices.". *In IEEE International Workshop on Pattern Recognition in Neuroimaging (PRNI)*, 2016, pp. 1-4, https://doi.org/10.1109/ PRNI.2016.7552335.

Acknowledgments

First and foremost, I would like to express my deepest gratitude to my supervisor Irene Yu-Hua Gu for her immense support and helpful guidance throughout this research journey. I would like to express my great appreciation to my co-supervisor Asgeir Store Jakola for his helpful suggestions and guidance from the medical perspective. Thanks for the time and effort spent on providing medical data.

Further, I wish to express my gratitude to my current and former colleagues in the signal processing group. In particular, I am grateful to my office mates Jakob Klintberg and Chenjie Ge. Thanks for always being willing to help and for sharing the PhD struggles.

I would like to thanks Chalmers University of Technology, Sweden and the Electrical Engineering Department for providing me the chance to pursue my PhD studies. I would also like to extend my gratitude to all co-authors and collaborators in this research.

Finally, I would like to express my deepest gratitude to my parents, my husband Zakir Hussain and my kids (Zain and Manha) who believed in me and made this journey easier.

Muhaddisa Barat Ali, Gothenburg, July 2023.

Acronyms

MRI:	Magnetic Resonance Imaging
MRIs:	Magnetic Resonance Images
WHO:	World Health Organization
CNS:	Central Nervous System
CNN:	Convolutional Neural Network
FC:	Fully Connected
GAN:	Generative Adversarial Network
FLAIR:	T2-Weighted Fluid Attenuated Inversion Recovery
HGG:	High Grade Glioma
LGG:	Low Grade Glioma
IDH:	Isocitrate Dehydrogenase
ReLU:	Rectifier Linear Unit
T1:	T1-Weighted
T1ce:	Post Contrast Enhanced T1-Weighted
GBM:	Glioblastoma Multiforme
GPU:	Graphical Processing Unit
SGD:	Stochastic Gradient Descent
BN:	Batch normalization
CAE:	Convolutional Autoencoder
PCA:	Principal Component Analysis
MSE:	Mean Square Error

DCGAN:	Deep Convolutional Generative Adversarial Network
ROI:	Region of Interest
GT:	Ground Truth
EtFedDyn:	Extended Federated Dynamic Regularization
PD:	Proton Density
TR:	Repetition Time
TE:	Echo Time
CSF:	Cerebrospinal Fluid
Gad:	Gadolinium
AI:	Artificial Intelligence
ML:	Machine Learning
DL:	Deep Learning
SVM:	Support Vector Machine
PET:	Positron Emission Technology
FL:	Federated Learning
fMRI:	Functional Magnetic Resonance Imaging
ANN:	Artificial Neural Network
Adagrad:	Adaptive Gradient Algorithm
RMSprop:	Root Mean Square Propagation
Adam:	Adaptive Moment Estimation
SAE:	Stacked Autoencoder
WGAN:	Wasserstein Generative Adversarial Network
cGAN:	Conditional Generative Adversarial Network

InfoGAN:	Information Maximizing Generative Adversarial Network
SGAN:	Semi-supervised Generative Adversarial Network
LSGAN:	Least Square Generative Adversarial Network
vid2vid:	Video to Video Synthesis
CoGAN:	Coupled Generative Adversarial Network
GDPR:	General Data Protection Regulation
HIPAA:	Health Insurance Portability and Accountability
CL:	Central Learning
FedAvg:	Federated Average
LSTM:	Long Short Term Memory Network
SCAFFOLD:	Stochastic Controlled Averaging for Federated Learning
FedDyn:	Federated Dynamic Regularization
FG:	Foreground
BG:	Background
JS:	Jensen Shannon

Contents

Part I

Introductory chapters

CHAPTER 1

Introduction

1.1 Magnetic Resonance Imaging (MRI)

MRI technology and scanning sequences have evolved recently due to the advancement of medical technology and testing equipment [1]. It combines the advantages of anatomy, function and imaging that produce high resolution images for the inside of the body with a magnetic field and radio waves. It reveals detail tissue information and is frequently used for pre-operative diagnosis, intra-operative treatment and post-operative examination of the body organs [2]. These body organs include mostly non-bony parts or soft tissues, e.g., brain, spinal cord and nerves, tendons, ligaments and muscles.

MRI scanner is a tube shaped equipment that generates a strong magnetic field. A closed bore MRI machine uses a ring of magnet that measures the magnetic strength between 0.5 T (Tesla) to 3 T as shown in Figure 1.1. Since the human body consists of 60% water, MRI scanner can measure the content and the behavior of water in different tissues. Water molecules contain hydrogen atoms, which have magnetic nature because of the presence of sub-atomic particles called protons that carry a positive charge. When a patient is placed inside the machine, the MRI scanner produces a magnetic field that aligns the

hydrogen atoms inside the body with the magnetic field. A radio frequency current is then passed through the body from the scanner that interrupts the alignment of hydrogen atoms, causing the protons in them to produce electromagnetic signals. When the burst of radio frequency stops, the protons return to their original alignment and emit radio signals. The realignment speed in different tissues of the body is different that causes the change in the emitted signals. These signals are picked up by a receiver in the scanner, where a sensor detects and measures these signals, which are used further to develop detailed 3D images.

Figure 1.1: Example of an MRI scanner [3].

MRI is a frequently used imaging test for brain and spinal cord to help brain related disease diagnosis, e.g., exploring the tumor type at the molecular level. It can show the differences between the white matter and the grey matter of brain. To describe a MRI scan appearance with respect to the grey matter as a reference, the terms hyperintense and hypointense are usually used. Anything brighter than the grey matter in the scan is hyperintense (brain tumor), while anything darker than the grey matter is hypointense (bone/skull). Clinicians use different MRI sequences, referred usually as MRI modalities, to examine the brain anatomy. The signal intensity is determined by a number of parameters in these modalities, which include: longitudinal relaxation time (T1), transverse relaxation time (T2), proton density (PD), the repetition time (TR) and echo time (TE) of radio frequency pulses among

many. By changing these pulse sequence parameters, different MR image contrasts can be generated as shown in Figure 1.2 and each of them is described below [4]:

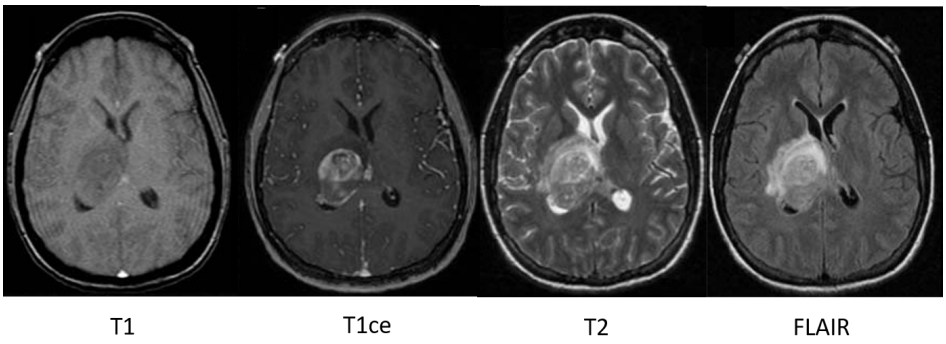

Figure 1.2: Axial slices of MR scans with glioma from four modalities. **Left to right:** T1-weighted MRI, T1-weighted contrast enhanced MRI, T2-weighted MRI and T2-weighted FLAIR MRI.

- **T1-weighted MRI:** This modality is produced by using short TR and TE times and shows the structures that are made of fat. The cerebrospinal fluid (CSF) is shown as grey, grey matter as grey, white matter as light and the bones as dark in the image. It is useful to examine normal anatomy of brain.
- **T1-weighted contrast enhanced (T1ce) MRI:** It is an enhanced version of T1-weighted images and is obtained by infusing a non-toxic paramagnetic contrast enhancement agent called as Gadolinium (Gad). When this agent is injected during the scan, it changes signal intensities by shortening T1 relaxation time and thereby increases the signal intensity. T1ce MR images are useful for showing vascular structures and breakdown in the blood-brain barrier, e.g., tumors etc.
- **T2-weighted MRI:** This modality shows the structure with a high amount of water. It is produced by using long TR and TE pulses. The CSF is shown as bright, grey matter as grey, white matter as darker grey and the bones as black in the image.
- **T2-weighted fluid attenuated inversion recovery (FLAIR) MRI:** This modality is similar to a T2-weighted scan except that the TR and

TE times are very long. This causes abnormalities to look bright and normal CSF fluid to look dark in the image, differentiating it from T2 scan.

Different modalities show different contrast of MRIs as summarized in Table 1.1. An abnormality (e.g., tumor) can form a complicated structure making it hard to identify and locate its region from a single modality. Therefore, multiple modalities are usually used to look for rich sources of information.

Table 1.1: The difference of images in different modalities [5]

Brain tissue	T1	T1ce	T2	FLAIR
CSF	Grey	Dark	Bright	Dark
Cortex	Grey	Grey	Light Grey	Light Grey
White matter	Light	Dark Grey	Dark Grey	Dark Grey
Tumor	Dark	Dark	Bright	Bright
Fat	Bright	Bright	Light	Light
Inflammation (infection etc.)	Dark	Dark	Bright	Bright

1.2 Gliomas and their Subtypes

Glioma is one of the primary type of brain cancer that originates in the gluey supportive cells, called glial cells. Glial cells are found around nerve cells which help them function. As a glioma grows, it puts pressure on that part of brain or spinal cord tissue and causes symptoms depending on which part of the brain or spinal cord it grows. The type of glial cells involved in tumor defines the type of a glioma and its genetic features. These genetic features are important to know to assist in prognosis and treatment management. Some gliomas grow slowly and are not considered aggressive. Others are considered aggressive that affect the brain functions and eventually becomes deadly, depending on their growth rate and location in brain. Glioma treatment includes surgery, radiation therapy, chemotherapy and many more.

Glioma Grading and Cell Types

Glioma types are defined by the type of glial cells involved, which can be either astrocytomas, ependymomas or oligodendrogliomas. Diffuse gliomas have high infiltrate growth to their surrounding tissues. On the other hand, non-diffuse gliomas have clear boundary and belong to either pilocytic astrocytomas or ependymoma group. Considering the aggressiveness of the tumor, World Health Organization (WHO) has categorized them into four grades (I-IV) [6].

Grade 1: Grade 1 gliomas belong to the glial cell type pilocytic astrocytomas. They have slow growth rate and are non-invasive with well-defined boundaries carrying better prognosis. For this reason, they can be removed by surgery with low chances of recurrence. These gliomas happen usually in children and young adults also called as low grade gliomas (LGGs).

Grade 2: Grade 2 gliomas emerge from astrocytes (the supportive cells around the neurons), they are referred to as low grade diffuse astrocytomas. These gliomas tend to spread in the surrounding of healthy tissues and do not show clearly defined boundaries. For this reason, it can be challenging to remove them completely through surgery and are usually treated with radiation and chemotherapy, depending on their size and location. They have better prognosis compared to grade 3 and grade 4 gliomas and are put into low grade category. However, astrocytomas often progress into high grade gliomas.

When grade 2 gliomas emerge in oligodendrocytes (cells that wrap around nerve fibers to provide support), they are called as oligodendrogliomas. Their occurrence is relatively rare with slow growth rate. Since they occur in brain regions that control major functions of body, they are difficult to be removed. Instead, radiation and chemotherapy might be suggested. They usually happen in adults.

Grade 3: Grade 3 gliomas are also called anaplastic gliomas, which means that tumor cells divide themselves rapidly. Sometimes, they appear as aggressive forms of their respective grade 2, otherwise they originate as grade 3. They have fast growth rate and spread quickly with more chances of turning into grade 4. They are more challenging to treat than grade 2 gliomas.

Grade 4: Grade 4 gliomas are highly malignant brain tumors with shortest survival rate. They are the most aggressive forms of astrocytomas also called as glioblastomas multiforme (GBM). Primary glioblastomas originate as grade 4 and develop quickly, while secondary glioblastomas appear as progressed forms of lower grade gliomas. These are rarely found in children and most commonly in older adults.

For treatment management and clinical decision making, it is of great importance to know the glioma grade. It helps to predict tumor progression over time for planing the treatment to prolong survival and improve patient's quality of life [7]. Table 1.2 summarizes glioma grades with their corresponding occurrences and 5-year survival rates.

Table 1.2: WHO grading of gliomas [6] and their occurrence rate [7] and 5-year survival rates [5]

Glioma Grade	Glioma Type	Occurrence rate in primary brain tumors	5-year survival rate
1	Pilocytic astrocytoma	15.6%	curable
2	Diffuse astrocytoma	2-5%	50%
	Oligodendroglioma	1.4%	80%
3	Anaplastic astrocytoma	1-2%	30%
	Anaplastic oligodendroglioma	1.4%	80%
4	Glioblastoma	14.9%	5%

Molecular Subtypes of Gliomas

Glioma grades can be observed from the brain MRI of a patient. Unlike glioma grades, some molecular biomarkers are not visible from MRIs [6]. These molecular subtypes are defined after WHO's revision of tumor classification of Central Nervous System (CNS) in 2016. Among many subtypes, two of the biomarkers are isocitrate dehydrogenase (IDH) and 1p/19q codeletion, which are considered as important biomarkers. They are associated with better prognosis and are sensitive to oncological treatment [5], [8], [9]. There-

fore, knowing this information prior to surgery is of great value and provides good assistance in treatment planning.

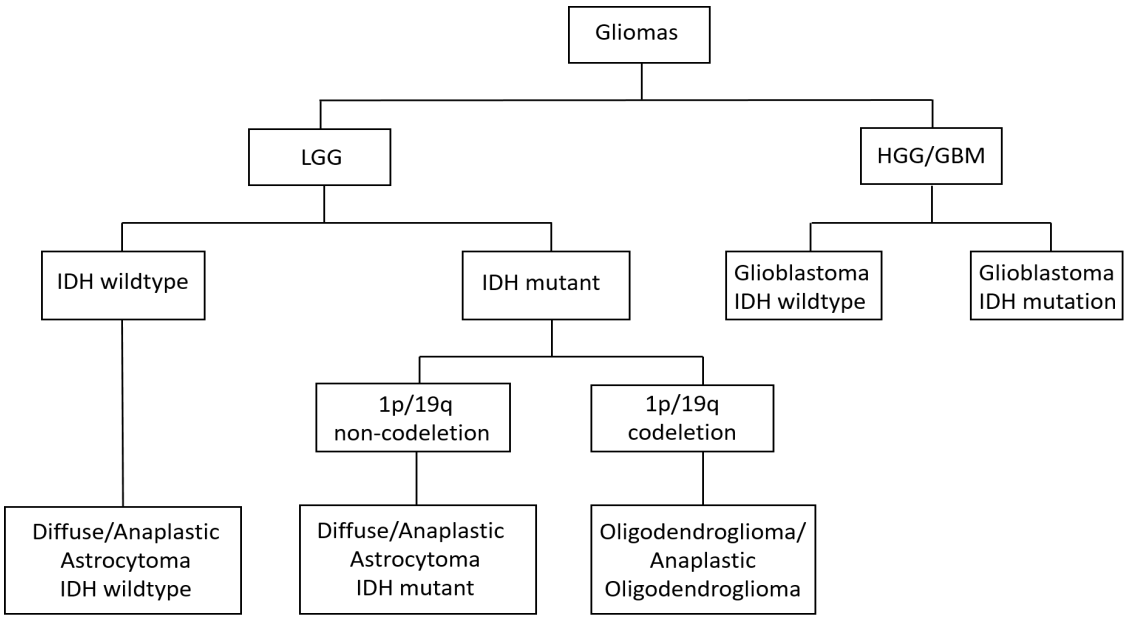

Figure 1.3: Summary of gliomas and their subtypes[10].

Figure 1.3 depicts a detail flowchart on types of gliomas and their molecular subtypes. LGG subtypes are associated with the mutations in the isocitrate dehydrogenase (IDH) gene and the abnormal missing sections of chromosomes 1p and 19q called as 1p/19q co-deletion. Glioblastomas or HGGs are classified into two distinct entities: IDH mutantion GBM and wildtype GBM [11].

Oligodendrogliomas (grade 2 type) are defined by the presence of an IDH mutation and a 1p/19q codeletion. Another grade 2 type, anaplastic oligodendrogliomas, when progresses over time turn into grade 3, otherwise it originally evolves as grade 3. On the other hand, diffuse astrocytomas without 1p/19q codeletion are either classified as IDH wildtype or IDH-mutant diffuse astrocytomas. Anaplastic astrocytomas can have abnormal genetic signatures including mutations in IDH genes. IDH wildtype diffuse astrocytomas are clinically like glioblastomas with poor prognosis. However, recent studies have shown that some low grade IDH wildtype astrocytomas lack both the molecular glioblastoma signature and genetic alterations suggesting the existence of low grade IDH wild type astrocytomas [12]. They grow fast and are prone to early recurrence, while IDH mutant diffuse astrocytomas show an

in-between behavior between oligodendrogliomas and IDH wildtype diffuse astrocytomas. They have slow progression to that of wildtype and their chances of recurrences are not high. IDH mutation is found in 70-80% of morphologically defined LGGs [13]. The survival rate for LGG IDH mutated patients is relatively higher than IDH wild-type. To identify these subtypes, tissue diagnosis is performed through invasive methods (e.g. biopsy, resection), that come with inherent risks.

In WHO grade IV glioblastomas (GBM), IDH mutations are also found frequent in secondary GBM (emerged from LGGs) as compared to primary GBM, where they are seen rare [14]. The frequent occurrence of IDH mutations in secondary GBM suggests that LGGs with IDH mutation often appear again to a higher grade. IDH mutation is found in approximately 10% of morphologically defined GBM. Studies have shown that, the presence of IDH mutations in GBM can predict a favourable disease outcome with prolonged survival.

1.2.1 Clinical Diagnosis

Clinical diagnosis of brain tumor consists of a series of tests based on patient's symptoms [15]. These tests usually consist of following steps [16]:

- A neurological exam is done to test functioning of different parts of brain and to locate the problematic brain area.
- Imaging tests such as MRI is conducted to get a detailed picture of the suspected area. It can be MRI of brain, spinal cord, or both that depends on the type of the tumor and its chances of spreading in the CNS. A neurological exam will be done to decide which modality of MRI can be used.
- A sample of the tumor tissue is usually needed to confirm the final diagnosis. It is done through a process called biopsy. A certain brain function might be affected by biopsy. Stereotactic needle biopsy is another alternative to remove a sample tissue [17]. It includes drilling a small hole in the skull and inserting a thin needle to take out a tissue sample, where the path of needle is planned through imaging scans.
- After biopsy, doctors could use the tissue samples for a desired treatment strategy. Other genetic tests are further conducted to know the molecular biomarkers in cells [11].

Glioma grades can be seen by clinicians by observing different modalities of MRI scans that show the status of different tissue densities in brain. However, the molecular subtypes can barely be seen with the naked-eyes from MRIs. Hence, it is important to look for other effective noninvasive ML diagnostic tools for assisting the diagnosis of brain tumor.

1.2.2 Machine Learning (ML) Assisted Diagnosis

Knowledge of molecular biomarkers assists in better prognosis and timely treatment [18], [19]. Despite the potential benefits of tumor biomarker testing, challenges still remain. Medical images contain information beyond visual perception, employing ML methods can help extracting useful information from these images and can help medical doctors in their diagnostic process. These methods can make future predictions using pre-existing MRIs along with their biopsy information. They also offer the possibility of combining the datasets, enlarging the training dataset by augmentation and over-coming the domain shift issues for effective learning. These benefits make ML methods more affordable to manage the surgical and maintenance costs. Such automatic decision making processes work quicker, save the labor and provide the clinicians with tools to choose. Hence, ML based assisted diagnostic methods show the potential to improve care system for patients.

1.3 Related Work on Brain MRIs

AI technologies such as machine learning (ML) methods have emerged to be promising tools for characterizing the gliomas and their subtypes, that can be roughly divided in two paradigms: classification based on conventional ML methods, and those based on deep learning (DL) methods. Other than classification, DL is capable of offering solutions to several other challenges. This thesis mainly addresses some of the challenges such as: small training dataset size, class imbalance, domain shift, dataset protection in hospitals, and training dataset without tumor boundary annotations.

Conventional Machine Learning (ML) Methods

There has been numerous studies on classification methods based on hand-crafted features. Kang [20] proposed a histogram based analysis for glioma grading. It performed analysis of diffusion coefficient maps over the whole tumor area by using diffusion weighted MRIs. Zhou [21] applied a combination of histogram, shape, texture features and age data to a random forest algorithm for IDH mutation and 1p/19q codeletion prediction. Han [22] generated radiomics signature from MRIs and performed analysis to predict 1p/19q codeletion status. Yu [23] also used radiomics based features for IDH mutation prediction. Van der Voort [24] used support vector machine (SVM) on 78 extracted MR image features together with age and gender information for 1p/19q prediction. Zhang [25] proposed a SVM-based recursive feature elimination to find the optimal features for IDH mutation detection. These methods used conventional ML methods with hand-crafted features from MRIs. Defining these features for subtypes of gliomas could be difficult.

Deep Learning (DL) Methods

Classification: DL methods can offer effective solutions for gliomas and their subtypes by automatically learning of relevant features from medical images. Studies have provided different solutions for the classification of gliomas and their molecular subtypes using automatic feature learning. Several methods have applied CNN models on MRIs. However, different studies have used different datasets. Some of them have combined datasets from different owners and the classifers were trained through central learning (CL). Matsui [10] proposed a residual network-based model that performed 3 class prediction of molecular subtypes. They applied a residual network-based model on scanned images (MRI, positron emission tomogrpahy (PET) and CT) and on some numerical characteristics of patients. Liang [26] used 3D DensNets on multiple modalities of MRIs for IDH mutation prediction. Ge [27] proposed a semi-supervised learning to make full use of unlabeled training dataset, where CNN features were incorporated into a graph-based framework to learn the labels of the unlabeled dataset. A DL based classifier was then trained to classify glioma types and IDH mutation. Van der Voort [28] proposed a multi-task CNN that used 3D MRIs to predict IDH mutation status, 1p/19q co-deletion

status, grade of the tumor while also performing segmentation simultaneously on 1508 patients. Chang [29] trained a residual CNN on four MRI modalities for IDH prediction. Li [30] performed tumor segmentation using a 6 layer CNN. A set of ten features were obtained from last fully connected layer, which were size normalized by Fisher vector coding. It was later followed by a SVM classifier for IDH mutation prediction. Ge [31] used GAN to generate missing MRI modalities from the training dataset for the prediction of IDH genotype. Cluceru [32] used a CNN on diffusion-weighted images for 3-class prediction (IDHwt, IDHmut-intact and IDHmut-codel).

Data Augmentation: DL networks generalize poorly on the unseen test set, when the training dataset is not sufficiently large. Data augmentation alleviates this by enlarging the training dataset. However, conventional ways of data augmentation offer only a limited plausible alternative. Recently, GAN based augmentation techniques have been successfully used to produce images close to real images new to human eyes [33]. Jendele [34] proposed a CycleGAN that is able to translate images from one class to another and used this property to augment the training dataset into a bigger and more balanced training dataset. Qi [35] proposed attention-guided CycleGAN to create tumors in normal MRIs and return normal MRIs from tumor ones. Frid-Adhar [36] suggested that training separate DCGAN for each lesion class led to improved classification performance than training unified AC-GAN for all classes. Further, Mok [37] used cGAN to augment MRIs from training dataset for robust segmentation.

Domain Mapping: In medical area, collecting large amount of data from a single source is difficult due to several reasons, such as limited number of patient cohorts in hospitals and high cost of image acquisition etc. However, combining several datasets from different sources to enlarge the size of the training dataset often leads to poor test performance due to the domain shift issue. Therefore, most algorithms use a single dataset for their training that often suffer from the problem of small and unbalance dataset. Different studies have shown different ways to map the datasets to a common target/reference domain, where network learning and prediction can take place. Bashyam [38] proposed a modified CycleGAN to perform domain mapping and tested on two example problems; age prediction from MRI dataset and

classification of schizophrenia using 9 datasets. Gao [39] used a modified CycleGAN to standardize the intensity distribution of MRIs using T2-FLAIR images from 4 different hospitals for LGG/HGG classification. Guan [40] proposed an attention-guided framework that consisted of an encoder for feature extraction, an attention model to locate dementia affected brain regions and a generative network based domain transfer module. Further for enhancing image quality, Qu [41] proposed to produce 7T T1-weighted images from 3T images, by fusing information from spatial domain and wavelet domain.

Federated Learning (FL): Combining medical datasets from multiple hospitals poses another challenge related to dataset protection. Recently, a federated learning (FL) approach was introduced to allow training a central DL model without violating data privacy regulations. FL approach has encouraged the hospitals to contribute in training for a specific task without the need to share their dataset. Regarding this, only a few studies on medical images involve classification and majority are proposed on segmentation task. Li [42] proposed FL based classification model on fMRI dataset, where the algorithm altered the shared local model weights by a randomization mechanism. Its aim was to tighten the privacy so that patients information could not be recovered from the model gradients or weights. Huang [43] performed FL based classification on brain metastasis identification using T1 MRI images and DeepMedic neural network. Nalawade [44] performed brain tumor segmentation by proposing an aggregation logic to combine knowledge gained across multiple institutions. Yi [45] performed FL based brain tumor segmentation, where inception module and dense block were introduced into standard U-Net to comprise SU-Net. Although FL provides a way to obtain generalizability of models trained on MRI dataset from different hospitals, its solution still suffers from domain shift of datasets. Domain shift can happen because of several reasons, e.g., different scanners/scanner settings. Guo [46] proposed a cross-site modeling method to overcome the domain shift issue during collaboration. The method provided a supervision to align the latent space distribution between the source domain to a target domain.

Tumor Segmentation: Other than classification, there exists many studies on brain tumor segmentation. The first successful model for segmentation is U-Net [47], since then many variants of U-Net [48], [49] have been reported.

Wang [50] performed brain-wise normalization and used two patching strategies for training a 3D U-Net. Kim [51] performed segmentation in a two-step setup, first an initial segmentation map was obtained from 2D U-Nets which was together used by 3D U-Net to produce the final segmentation map. Shi [52] used U-Net by adding increased number of channels that enabled extraction of rich diverse features from multi-modality scans. Some other works used CNN for segmentation. Sun [53] proposed a CNN based computationally efficient model with reduced number of parameters. Das [54] used 3D CNN in a cascaded way to first extract whole tumor area followed by the core and then enhance core tumor areas. Shan [55] suggested a lightweight 3D CNN with improved depth that used different size of convolution kernels to aggregate features. Ramin [56] proposed a cascaded CNN to speed up the learning. Most of these segmentation methods used annotated tumors for training deep networks.

1.4 Thesis Aims and Scopes

The thesis focuses on some DL methods for classification of gliomas and their biomarker molecular subtypes aimed at assisting medical doctors in their diagnostic process. The included papers focus on methods to improve the performance with respect to (i) test accuracy (ii) enlarging the training dataset size (iii) training on multiple datasets (iv) using training dataset without/with just a few GT tumor boundary annotations.

The methods and contributions are focused on the following sub-problems:

- **CAE based classifier to improve the performance:** A multi-stream convolutional autoencoder (CAE) based classifier is employed with a 2-round training strategy for effective feature learning and feature fusion.
- **GAN generated data to enlarge the training dataset:** DL methods need large amount of balanced training dataset. A DCGAN is proposed that generates synthetic data distribution as the original data for multi-modality MRIs, which enlarges the training dataset size.
- **Domain mapping among different datasets to avoid domain shift:** Clinical datasets are often small in size. MRIs are acquired with

different scanners/ scanner settings at different hospitals. Simply combining them for training a DL classifier do not offer improved test performance. A framework is proposed that uses CycleGAN to map clinical MRIs from different source domains to a target domain without loosing the molecular-marker information. Further, a multi-stream CAE based classifier is employed for the classification of 1p/19q codeletion and IDH mutation.

- **Federated learning (FL) for dataset protection:** Training datasets consisting of molecular based glioma subtypes are usually small in size, as they are obtained from different patient cohorts with different scanners/scanner settings. Furthermore, such datasets often suffer from class imbalance. If those datasets are combined to train a DL classifier using central learning (CL) approach, data privacy issues often pose a constraint. Despite, many studies on FL, few studies can be found on gliomas and their subtype classification. We propose a novel FL scheme, where datasets from different hospitals participate in a collaborative learning without providing their datasets to a central model.

- **Using tumor bounding box areas when GT tumor boundaries are missing:** DL based classification often requires GT tumor boundaries which are usually manually drawn by medical experts. Manual drawing of tumor boundaries puts high demand on clinicians. By using ellipse shaped tumor boundaries, the proposed method explores whether the classification performance is affected compare to that of using GT tumor boundary annotations.

- **Brain tumor segmentation based on using 2D ellipse box areas:** Supervised DL networks require training datasets with annotated tumor annotations by medical experts to perform segmentation, which is a time consuming process. This study is based on the datasets that have only a small number of tumor boundary annotations. By using foreground (FG) and background (BG) tumor boxes on unannotated MRIs, this method explores whether segmentation performance is affected as compare to that of using all GT tumor boundary annotations.

Scopes and Limitations: The studies in this thesis were conducted on four different datasets containing multiple modality MRIs as summarized in

Table 1.3. Some of these datasets have missing tumor boundary annotations. The thesis has addressed this issue and offered alternative solutions when developing and evaluating the proposed methods.

Table 1.3: Summary of datasets used in our studies.

Name	Modality	Papers	Brain tumor type
MICCAI'17	T1/T1ce/T2/FLAIR	A, C, E	LGG/HGG
TCGA	T1/T1ce/T2/FLAIR	C, D	LGG/HGG, IDH mut/wt
US	T1ce/FLAIR	B, C, D	LGG, IDH mut/wt 1p/19 codel/non-codel
France	T1ce/FLAIR	B	LGG, IDH mut/wt 1p/19 codel/non-codel

The included papers A, B, C, D are based on 2 class classification. Due to small sizes of the datasets, the proposed methods cannot offer complete solutions to the clinical problems at the moment. Therefore, test should be conducted when large training datasets become available. The included papers do not include technical details of data acquisition, pre-processing and labeling/annotating MRIs.

1.5 Thesis Outline

The thesis proposes DL methods for gliomas and their subtype classification. It consists of two parts. Part I includes the introduction part on the research background. The remainder of part I is organized as follows: Chapter 2 reviews several background theories and methods on which the proposed studies are built. Chapter 3 summarizes the main work and contributions of the proposed methods. Finally, conclusions and future works are discussed in Chapter 4. Part II includes the five papers that the thesis is based upon.

CHAPTER 2

Background Theories and Methods

2.1 Deep Learning (DL) for Classification of Brain Tumors

Deep learning (DL) is a class of machine learning (ML) methods that may automatically learn features on raw input data. Success of DL could not have been possible without the increased computational power of modern GPUs (Graphical Processing Units) [57] and access to large labeled datasets. DL methods have been developed and applied successfully in many areas of applications, that use different learning approaches, such as supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning. Classification is a two-step process: training or learning phase and the test or evaluation phase. The test accuracy indicates whether a classifier is capable of classifying the unseen data.

A DL network tries to mimic a human brain through combination of data inputs, weights and bias where these elements work together to recognize and classify the objects. The network has a deep architecture that consists of mul-

tiple layers to learn representations from data in different abstraction levels [58]. The key aspects of DL is that multiple layers of features are learned automatically. Unlike conventional ML methods, where features are defined by human experts before extraction, DL eliminates this requirement. The network accepts data at the input layer and makes the final prediction at the output layer. However, DL networks are not always simple in different applications and there are certain types to address specific problems.

2.1.1 Deep Convolutional Neural Network

Convolutional Neural Network (CNN), in theory, is able to learn any functions and is known as universal function approximator. The basic layers that build a CNN architecture consist of 3 types. These are convolutional layers, pooling or subsampling layers and fully connected (FC) layers as shown in Figure 2.1. Its architecture is designed to take advantage of the 2D structure of an input, such as image or a speech signal. This is obtained with weight connections followed by some form of pooling that produces translation invariant features. Compared to fully connected networks, CNNs have fewer weight connections with the same number of hidden units.

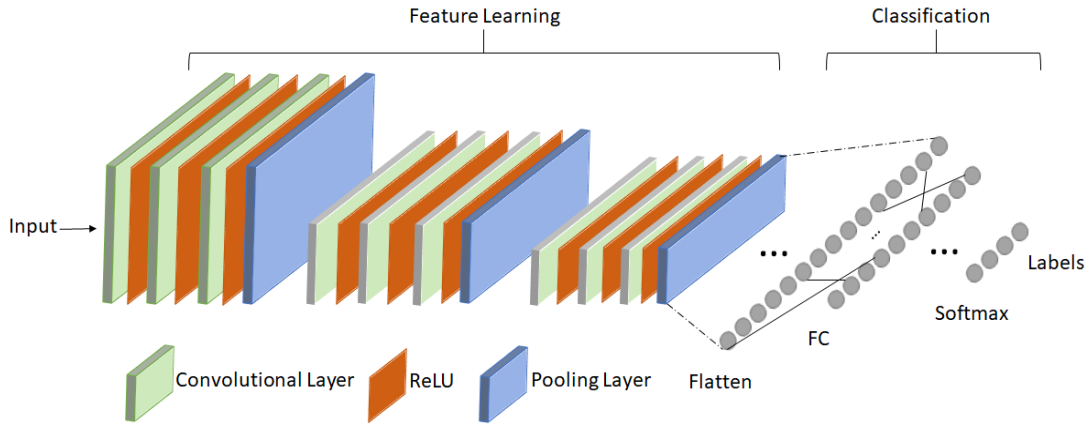

Figure 2.1: Example of a Convolutional Neural Network (CNN) [59]. **ReLU:** Rectifier linear unit, **FC:** Fully connected

Convolutional layers: The purpose of the convolutional layers is to extract image characteristics by using automatically learned filters. Each convolutional layer contains several learnable filters. The convolutional layer uses

these filters to perform convolution operations by scanning the input with respect to its dimensions resulting in a number of feature maps. The layer is convolved across the input dimension with a step size called stride. If the stride size is big, it skips more pixels and the output appears as a smaller feature map. The reduced size of feature map can cause image borders to vanish. However, it can be avoided by zero padding around the input feature maps. If stride is one, a convolutional layer only changes the input volume depth-wise leaving its size unchanged. The first few convolutional layers extract low level features, such as edges, lines, corners etc., while the higher layers assemble these low level features into more complex features. The depth of the network and the layer sizes decide the learned filters ability to recognize high-level features. The convolutional property enables translation invariance, that is, similar patterns in different parts of the image is processed in similar manner.

Pooling layers: The aim of a pooling layer is to introduce non-linearity as well as to reduce the parameter space. It is applied on feature maps obtained from convolutional layers. Two common choices are max pooling and average pooling. Max pooling computes the maximum of a local patch of units, and average pooling computes the mean of a local patch of units. Computing the convolution and max pooling function on any local patch of units are separately depicted in Figure 2.2.

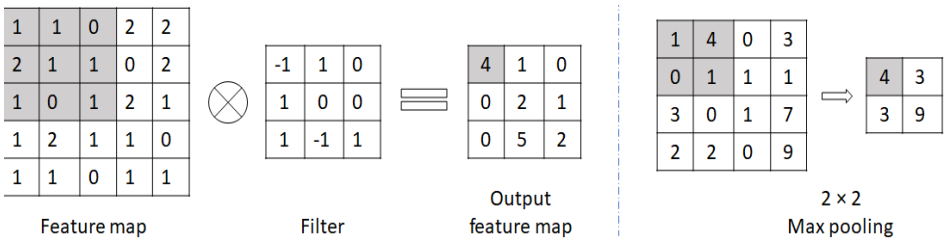

Figure 2.2: Examples on convolution and maxpooling functions [59]. **Left:** Example of a 2D convolution operation with stride =1. **Right:** Example of a 2D maxpooling function with stride =2.

Fully connected (FC) layer: The FC layers are stacked on top of the convolutional layers to flatten the multi-dimensional output of the last convolutional layer into one dimension. These layers end with an output layer that consists of same number of neurons as the number of classes.

Activation function: It is an important part of the design of a CNN. All the nodes in a layer use the same activation function. In a convolutional layer, the result of the local weighted sum is usually passed through a non-linear operator. An activation function is differentiable, which is required during network training when performing backpropagation of the error to update the weights. Different activation functions are used in different layers of the network. In the hidden layers, its selection would handle the learning of the network. In the output layer, its selection would define the type of prediction.

Commonly used activation functions for any hidden layer are:

- *ReLU:* It is a preferred activation function for hidden layers of a CNN because it is less susceptible to vanishing gradient. However, it may cause saturation. It returns 0 for negative values, otherwise, returns the value x for a given input x as:

$$\text{ReLU}(x) = \max(0, x). \tag{2.1}$$

  This activation function has other variants such as leaky ReLU [60] and parametric ReLU [61].

- *Sigmoid:* It is also called as logistic function. It takes any real value as the input and gives the output values between 0 and 1. For large input values, the output is close to 1, while for smaller input values, the output is closer to 0. Due to this reason, it may cause vanishing gradient issue during network training. For a given input x, the sigmoid function is computed as:

$$\text{Sigmoid}(x) = \frac{1}{(1 + e^{-x})}. \tag{2.2}$$

- *Tanh:* A tangent hyperbolic activation function or Tanh is very similar to a sigmoid function with the same S-shape. It accepts any real input value and produces output between -1 and 1. Large input values results in output close to 1 and small values close to -1. Like sigmoid function, it may also cause vanishing gradient. For a given input x, the Tanh function is computed as:

$$\text{Tanh}(x) = \frac{e^{x} - e^{-x}}{(e^{x} + e^{-x})}. \tag{2.3}$$

Commonly used activation functions for an output layer are:

- *Linear:* This function is directly proportional to the input, i.e., the weighted sum of neurons. It is mostly used for regression.

- *Sigmoid:* Similar to (2.2), this function outputs values between 0 and 1. It is mostly used for two class classification, where the output layer has one node.

- *Softmax:* This function accepts real value inputs and computes probabilities of the output units. It is frequently used for multi-class classification problem, where the output layer has one node per class. If x is a given input and the m is the number of output nodes, the softmax function is computed as:

$$\text{Softmax}(x) = \frac{e^x}{\sum_{j=1}^{m} e^{x_j}}. \tag{2.4}$$

Cost function: This function is used for computing DL network errors. The aim is to search through weight parameter spaces that minimize the cost function and produce the lowest error. A classification problem is defined as predicting the probability of an input example belonging to a specific class. If the problem is a binary classification, there would be two classes with high probability for a positive class and a low probability for a negative class. If it is a multiple-class classification, then the probability of examples belonging to each class are predicted. In such cases, a commonly used loss function is the cross-entropy loss [62]. Let p_y denote the probability of the true label and $p_{\hat{y}}$ the predicted probability for N number of training examples, the cross-entropy loss is defined as:

$$L = -\frac{1}{N}\sum_{i=1}^{N}(p_{y_i}\log(p_{\hat{y}_i}) + (1-p_{y_i})\log(1-p_{\hat{y}_i})). \tag{2.5}$$

A dynamically scaled cross entropy loss is called a focal loss function. It is used when training examples have high class imbalance. The examples from the major class usually comprise the main loss and dominate the gradients. Focal loss tries to downweigh the confidence in predicting the easy class during training and allows the model to focus more on examples from the hard class.

To balance the importance between the major and minor classes, a balanced variant of focal loss is defined as:

$$L = -\frac{1}{N}\sum_{i=1}^{N}(\alpha\ \hat{q_i}^{\gamma} p_{y_i}\ \log(p_{\hat{y}_i}) + (1-\alpha)\ (p_{\hat{y}_i})^{\gamma} q_i\ \log(\hat{q}_i)), \tag{2.6}$$

where $p_{\hat{y}_i}$ and $\hat{q}_i = (1-p_{\hat{y}_i})$ are the predicted probability, p_{y_i} and $q_i = (1-p_{y_i})$ are the true probability of training examples as true labels, $\alpha \in [0,1]$ is the weighting factor for major class and $(1-\alpha)$ for minor class, $\hat{q}$ is a modulating factor and γ is a focusing parameter. When $\gamma = 0$, (2.6) is same as the weighted cross-entropy loss. Choosing $\gamma > 0$ reduces the relative loss for easy class examples while putting more focus on hard class examples.

For pixel-wise classification or regression where a real-value quantity is to be predicted, for instance, in autoencoders, a regression loss function also referred to as the mean square error (MSE). It is calculated as the average of the squared difference between the the predicted pixel values $\hat{y_i}$ and actual pixel values y_i as:

$$L = -\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y_i})^2. \tag{2.7}$$

CNN training: During training, the network searches through weight parameter spaces that may minimize the difference between the network's prediction and the GT labels. For such purpose, gradient decent method aka "batch gradient descent" is usually used. It computes the gradient of the cost function $\nabla_{\mathbf{w}} L(\mathbf{w})$ w.r.t. to the weight parameters $\mathbf{w}$ for the whole training dataset as:

$$\mathbf{w} = \mathbf{w} - \eta\ .\ \nabla_{\mathbf{w}} L(\mathbf{w}), \tag{2.8}$$

where gradients are computed using backpropagation [63] by moving in a backward direction to adjust the weights in each network layer. Since we compute the gradients for the entire dataset for just one update, batch gradient descent can be very slow which might not fit in memory. It guarantees to converge to the global minimum for convex error functions and to local minimum for non-convex functions. In contrast to batch gradient descent, stochastic gradient descent (SGD) [64] computes a weight update for each training example x_i and GT label l_i given as follows:

$$\mathbf{w} = \mathbf{w} - \eta \; . \; \nabla_{\mathbf{w}} L(\mathbf{w}; x_i; l_i). \tag{2.9}$$

It converges relatively faster with a high variance that could cause fluctuations and instability. An intermediate option could be a mini-batch gradient descent that computes an update for every batch of n training examples, give as follows:

$$\mathbf{w} = \mathbf{w} - \eta \; . \; \nabla_{\mathbf{w}} L(\mathbf{w}; x_{i:i+n}; l_{i:i+n}). \tag{2.10}$$

This reduces the variance of weight updates leading to more stable convergence and enables computing the gradients efficiently. The term SGD is commonly employed also to mini-batch gradient descent. To improve the convergence, some variants of SGD are used such as, adaptive gradient algorithm (Adagrad) [65], root mean square propagation (RMSprop) [66] and adaptive moment estimation (Adam) [67]. Moreover, batch normalization (BN) [68] also helps to speed up the training. Despite the potential of CNNs, they are prone to over-fitting when the training dataset size is small. To tackle this problem, there are certain ways such as using data augmentation to enlarge the training dataset size, or using different regularization techniques. For the latter method, some common choices are using L_1/L_2 regularization to penalize large weights and adding random dropouts to FC layers.

Several CNN Models

A CNN model has the capacity of exploiting the spatial or temporal correlation from input datasets. A few successful example models are listed below:

- *LeNet* [69]: It was the first CNN model and was originally developed to categorize hand written digits of the MNIST dataset. It consisted of seven layers with its own trainable parameters that effectively performed classification. However, it was not effective enough for bigger images and large number of classes.
- *AlexNet* [70]: It was a major breakthrough with improved architecture on ImageNet dataset and was the winner of ILSVR-2012. The architecture was similar to LeNet but with more included filters, that enabled classifying more object classes. Its architecture consisted of 5 convolutional layers followed by max pooling layers and 3 FC layers to classify

1000 classes. All the layers used ReLUs as activation function and FC layers used dropouts to overcome over-fitting. The last layer used the softmax function.

- *ZFNet* [71]: It was the winner of ILSVRC 2013. ZFNet proposed a fascinating network whose aim was to statistically visualise network performance by analysing neuron activation. It was the outcome of some improvement on AlexNet where some hyper-parameters were adjusted like, expanding the size of middle convolutional layers, lowering stride value and employing the filter size of 77 in the first layer.

- *GoogleNet* [72]: It was the winner of ILSVRC 2014. GoogleNet proposed an inception module. It was an improved version of the original LeNet design and consisted of 22 layers (27 with pooling layers) with nine inception modules stacked on top of each other. The other versions were later introduced such as inception-V4.

- *VGGNet* [73]: It was the winner of ILSVRC 2014. VGGNet consisted of a deep architecture extending up to 16-19 weight layers. Its main idea was to know how a network could be made more dense. The network architecture was simple, with pooling and fully linked layers. VGGNet replaced the bigger size filters with a stack of 3×3 filters presenting that the simultaneous placement of small size (3×3) filters could provide the effect of a big size filter. Hence, the parameters were reduced with the benefit of low computational complexity.

- *ResNet* [74]: It was the winner of ILSVRC-2015, and has been one of the popular and effective DL networks. It consisted of a residual block which was designed on the concept of skip-connections and used a lot of batch-normalization. The new added layers learned something new from the previous layers without sacrificing speed. The network design was inspired by VGGNet-19 and had a 34 layer network with added shortcut and skip connections. The gradient flow reached to the earlier layers through the shortcut links and overcame the vanishing gradient issue.

- *DenseNet* [75]: It was proposed to overcome the vanishing gradient issue in the same way as ResNet. It used cross-layer connection to solve this problem. It had a thin-layer structure, but as the number of feature

maps increased, its computation became costly. The network's information flow improved by giving each layer direct access to the gradients through the loss function.

These CNN models performed well on ImageNet dataset but these methods need yet to be explored on medical image datasets. Medical datasets are different because of different image modalities, small dataset size and other varying details.

2.1.2 Autoencoders

A stacked autoencoder (SAE) is a specific type of network, which is designed for learning a compressed representation to encode the input such that the decoder output would become as close as possible to the input [76]. Its main idea is to learn an informative representation in an unsupervised manner that can be further used for supervised learning. It consists of an encoder part and a decoder part. The encoder learns the features of inputs and generates feature codes. The decoder reconstructs the input from the feature codes. The encoder and decoder parts are usually DL networks containing non-linear activation functions. If the non-linear operations are dropped, it would perform similar to that of principal component analysis (PCA) [77]. Autoenocder learns a non-linear manifold instead of just learning a low dimensional hyperplane to represent data.

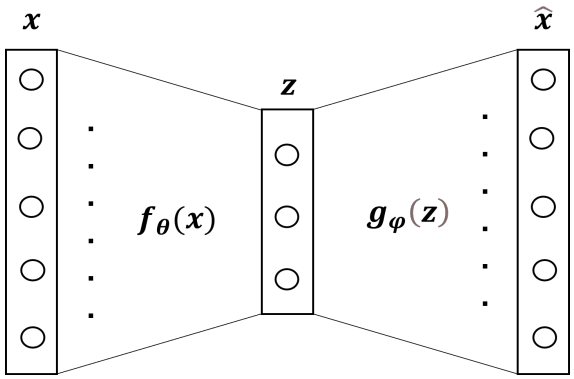

Figure 2.3: The basic structure of an autoencoder.

Let x be the input for the encoder, $\hat{x}$ be the decoder output and $z =$

$f_\theta(x)$ is the output from the encoder and $\hat{x} = g_\phi(z)$ represent the decoder. Further, let θ and ϕ represent the optimized weights for the encoder and decoder respectively. The network is trained in an unsupervised manner that minimizes the mean square error (MSE) between the input and the output images for N number of images given as:

$$L(x, \hat{x}) = -\frac{1}{N}\sum_{i=1}^{N} |g_\phi(f_\theta(x)) - x|^2. \tag{2.11}$$

SAE is trained so that it should be sensitive enough towards reconstructing the input by minimizing the loss function. At the same time, it should be insensitive towards the input and not to just memorize it. In the cost function, a regularization term is usually added to introduce constraints and learn useful representation from the input. The training is performed using SGD to perform the reconstruction well and is called as pre-training step. Once the network is pre-trained, the encoder part acts like a feature descriptor and a classifier can be added for supervised refined-training. The idea from SAE is widely used in compressing data [78], where it reduces storage space and speeds up the time for computation. It improves performance by discarding redundant variables [79]. It is also used for visualizing high dimension data [80] and for reducing noise from input data.

Another AE network is formed by using convolutional layers called as convolutional autoencoder (CAE) [81]. This replaces the basic structure of a SAE by replacing fully connected layers to convolutional layers with pooling layers to extract feature codes in the encoder part. The layers in the decoder part is replaced by deconvolutional layers with upsampling layers to reconstruct the input image from the feature codes as shown in Figure 2.4. Convoltuional layers in CAE are better to capture spatial information form image datasets with fewer weights compared to SAE.

Other Autoencoder Models

By adding different regularization terms in the cost function, many variants of autoencoders have been introduced, for example:

- *Sparse AE* [82]: This model enforced sparsity regularization on the hid-

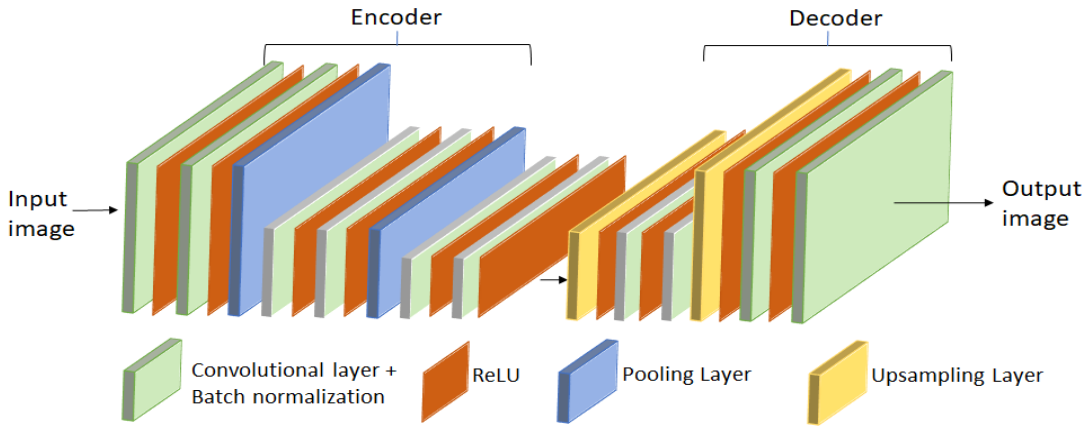

Figure 2.4: Example of a Convolutional Autoencoder (CAE) [59].

den activation instead of the weights. It was done in two ways: inducing sparsity by applying L_1 regularization or by using KL-divergence. The former method added L_1 norm of hidden activation to the objective function which activated the highest values nodes while making others zero called as the sparse penalty term. The latter method measured the distance between two probability distributions, where one distribution was computed from the activation of neurons that were assumed as Bernoulli variables, and the other distribution was empirically computed from the probabilities of the hidden neurons.

- *Denoising AE* [83]: This model can be considered as a robust autoencoder that was used for error correction. It learned to reconstruct the original undistorted input from its noise corrupted copy. Hence, it avoided to copy the input to the output, and learned instead the actual features of the input.

- *Contractive AE* [84]: This model resisted input perturbations and enforced a robust feature extraction to contract a neighborhood of inputs into a smaller neighbourhood of outputs. It worked similar to a denoising autoenocoder where the decoder resists the noises. On the contrary, contractive autoencoder added a regularization term in its objective function that penalized large derivatives of hidden layer activations with respect to the input training examples.

- *Variational AE* [85]: This model improved the representation capabili-

ties of basic autoencoders. This generative model attempted to encode inputs through a probabilistic distribution instead of an arbitrary function. The input examples were encoded (recognition model) into two parameters to approximate the real posterior distribution over the latent space. It was assumed that the latent space had already a prior distribution as a normal Gaussian distribution. When a point was randomly sampled from the distribution, the decoder (generative model) mapped that latent space point to the reconstructed input data.

2.1.3 Central Learning (CL)

It is a learning method in which the datasets from several local clients are combined into a pool of dataset to train a central model. As shown in Figure 2.5, dataset from each local client is shared centrally. The central model combines all datasets from all local clients, extracts features and then trains a DL network on the combined datasets. This results in a single integrated central model which is evaluated either on the individual test set by each local client or on the combined test sets. This learning method offers low level of privacy because sharing data centrally discloses the sensitive information of the datasets. However, it enables enlarging the dataset size and allows training a more generalized DL network. The public datasets, obtained from different hospitals, can be used and shared centrally to train a classifer. In this thesis, we have assumed a DL network trained on a single MRI dataset similar to a CL approach, since the training method is similar to that when it is performed on a combination of datasets.

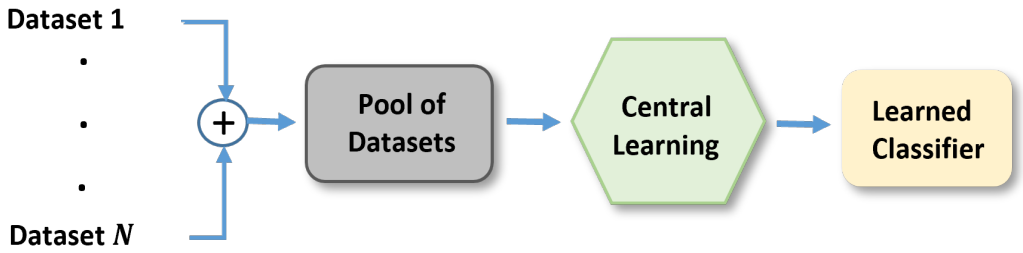

Figure 2.5: Depiction of a CL method on centrally shared pool of datasets from N local datasets.

2.1.4 Federated Learning (FL)

The European General Data Protection Regulation (GDPR) [86] and the United States Health Insurance Portability and Accountability (HIPAA) [87] have put regulatory constraints on sharing personal health information. Such data protection constraints the medical data sharing and hence training DL networks become limited on using to local data only. This in-turn limits the research on finding effective automatic medical diagnostic methods. Recently, federated learning (FL) is introduced as a collaborative learning approach that allows training of a central model while protecting the privacy of datasets. This learning method does not need the local datasets to be shared centrally. It allows the local DL networks to be trained locally and the network weights are instead shared with a central model as shown in Figure 2.6. A set of local DL networks are trained locally on the local datasets and the weights of the local networks or gradients are sent to the central model for aggregation. The central model sends back the updated network weights or gradients to each local networks for further training. One complete round of such a training is called a communication round, and it continues until the central model converges.

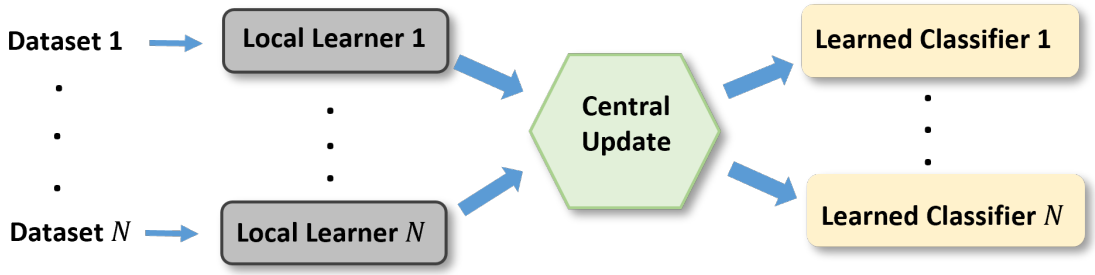

Figure 2.6: Depiction of a FL method with N locally learned networks on N private datasets, where network weights are transferred to the central model for the update and redistribution.

There are different basic categories of FL such as horizontal FL, vertical FL, federated transfer learning, cross-silo FL and cross-device FL [88]. In this thesis we are interested in horizontal FL frameworks, where each local client has dataset obtained from different group of people with same features to train a shared DL network.

Challenges in Federated Learning

FL is seen as an effective way to address data protection issue and has attracted many research interests [89], [90] in different applications [91], [92]. Despite many FL methods have been introduced, challenges remain, some of which can be listed below [93]:

- *Data Heterogeneity:* Medical datasets are diverse from many aspects. They consists of different modalities and characteristics and are acquired from different brands of scanner machines with different settings. Such inhomogeneous data distribution poses a challenge in FL training. In practice, simple FL [94] fails to overcome this problem and suffers from client drift. This problem may cause the central model not to reach to an optimal solution. For intuition, Figure 2.7 depicts the scenario of client drift caused by data heterogeneity, where client 1 performs more local updates than client 2, and the central model update strays more towards the local minimum of x_1^*, while away from the global minimum x^*.

- *Convergence Speed:* The factors such as data heterogeneity, computation limitations and partial client participation greatly impact the convergence speed of central model in FL. A large number of communication rounds is usually required that may induce delay in central model convergence. This could be costly in terms of network resources. Furthermore, local dataset may have different convergence speed.

- *Communication Cost:* Frequent communication between client models to/from the central model could be costly. One way to reduce this cost could be to push optimization burden on the client sides [96]. As discussed previously, delay in convergence speed could also put burden on communication cost.

- *Class Imbalance:* Different geographic areas can have different patient cohorts and disease distribution. For instance, due to ozone hole, the countries at southern hemisphere can have more skin cancer patients than the ones on northern hemisphere. This affects the label distribution. Such proportion of classes in the client training datasets have great impact on FL performance and may lead a central model towards

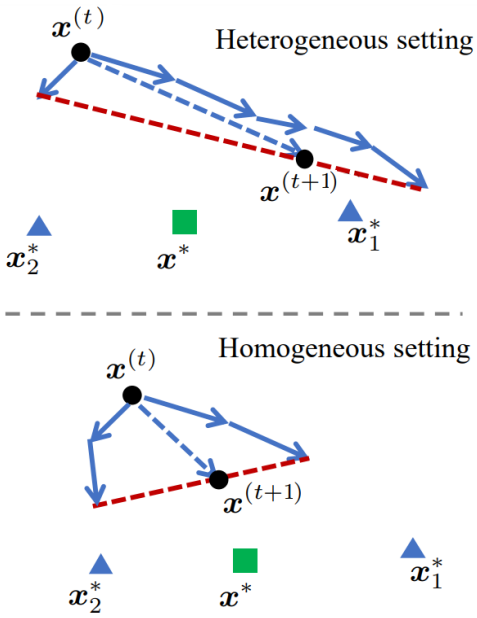

Figure 2.7: Depiction of model updates in the parameter space [95]. **Top row:** Heterogeneous setting. **Bottm row:** Homogeneous setting. **Green square:** Global minimum point. **Blue triangle:** Local minima of local objectives of client 1 and client 2 respectively.

the wrong direction and hence deteriorate the performance. Therefore, it is crucial to develop new methods for mitigating the effect of class imbalance in FL [97].

Some FL Frameworks

FL frameworks can be realized from different design architectures and computation plans, but their basic operational aim remains the same, i.e., to combine knowledge learned from different private datasets. Several FL frameworks are briefly reviewed as follows:

Federated Average (FedAvg) [94]: It is a basic FL algorithm developed by Google and is an effective optimization method in FL setting. The steps involved in the training process of FedAvg are depicted in Figure 2.8. In the start, the weights of DL network in the central model and client models are different from each other. The central model sends its weights to the clients before training the local DL networks. After initialization, the local clients

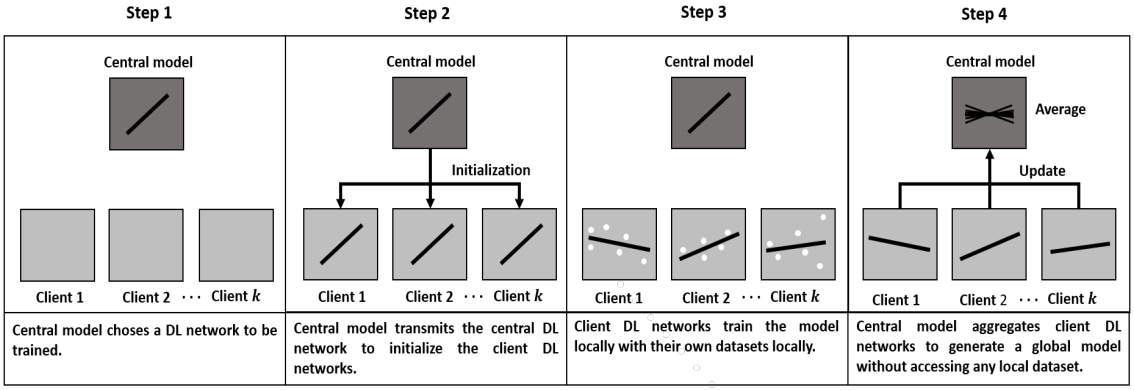

Figure 2.8: Steps in learning process of FedAvg.

train their DL networks on their own local datasets with stochastic gradient descent (SGD) for a number of epochs. After the training process in each client network is completed, each client communicates its updated network weights to the central model. The central model takes the average of these weights and update its own network weights. The central model then passes them back to the client networks for the next communication round. This algorithm has addressed the problems of client availability, class imbalance and heterogeneity. In each communication round $t = 1, 2, \cdots, T$ the central model sends its model to k clients from the randomly selected subset of S_t. Each of k clients trains and updates its network weights on its local dataset as follows:

$$\mathbf{w}_k^t = \mathbf{w}_k^t - \eta \, \nabla F_k(\mathbf{w}_k^t), \tag{2.12}$$

where η is the learning rate. The local objective function for the kth client is represented by $\nabla F_k(\mathbf{w}_k) = \sum_{k \in S_t} \frac{|n_k|}{n} \Delta \mathbf{w}_k^t$ given that n_k denotes the data samples of kth client and $n = \sum_{k \in S_t} n_k$ denotes the total data samples. The local updated network weights are communicated to the central model, where the central model weights $\mathbf{w}_c$ are computed by aggregating the local model weights $\mathbf{w}_k$. The iteration continues until convergence, or communication round T is reached. This algorithm was implemented using a multi-layer perceptron, different CNN architectures and a two layer long-short term memory (LSTM) network.

FedProx [98]: FedProx has improved FedAvg in many matters. FedProx al-

lows non-uniform amount of work across client models which in FedAvg is same across clients without considering their underlying system constraints, and if any client fails to work within a specified time window, it is dropped instead [99]. FedProx supports robust convergence on heterogeneous datasets, while FedAvg diverges empirically in heterogeneous data settings [94]. To support non-uniform amount of work across client models, FedProx introduces an additional proximal term in the local objective as follows:

$$\mathbf{w}_k^t = \mathbf{w}_k^t - \eta \, \nabla F_k(\mathbf{w}_k^t) + \mu(\mathbf{w}_k^t - \mathbf{w}_c^{t-1}), \tag{2.13}$$

where $\mathbf{w}_c^{t-1}$ is the central model weights sent to the clients and $\mu > 0$ is a tunable regularization parameter that controls the step size in FedProx and requires a careful selection. The proximal term pulls the local client model backward closer to the central model and helps the training in two ways. Firstly, it avoids the effort of manual setting of local epochs and restricts the local updates to remain closer to the initial central model. Secondly, it efficiently incorporates variable amount of work for local clients caused by the dataset heterogeneity.

FedNova [95]: Another variant of FedAvg is FedNova that improves the stability and overall accuracy of FL in heterogeneous settings. FedNova was introduced while considering the following situation: given the same time constraint when clients have different computational power; or given the same number of epochs and batch sizes clients have different sizes of datasets. In such scenarios, the clients with more local updates get biased towards its local optimum, and this in turn impacts the global optimum. Thus, to ensure that the central model updates are not biased, FedNova normalizes and scales the local updates of clients based on the number of local steps. In other words, it uses momentum to assign correct weights to the local weights of clients as follows:

$$\mathbf{w}_k^t = \mathbf{w}_k^t - \eta \frac{\sum_{k \in S_t} |n_k| \tau_k}{n} \frac{\nabla F_k(\mathbf{w}_k^t)}{\tau_k}, \tag{2.14}$$

where τ_k is an arbitrary scalar value that changes across each round and is computed as $\tau_k = En_k/B$ for local epochs E and mini-batch size B for each local client. The local averaged gradient is multiplied with τ_k while computing the aggregated gradient. This normalized averaging method eliminates

objective inconsistency and provides fast convergence.

SCAFFOLD [98]: Stochastic controlled averaging for FL (SCAFFOLD) is introduced to overcome the unstable training and slow convergence in FedAvg. SCAFFOLD introduces control variate (variance reduction) to avoid client drift in the local updates. This control variate shows the difference between the update direction of the central model and in the gradient direction of each client during its local model updates. In this way, SCAFFOLD corrects the direction of local updates by adding drift during training as follows:

$$\mathbf{w}_k^t = \mathbf{w}_k^t - \eta(\nabla F_k(\mathbf{w}_k^t) - \mathbf{c}_k + \mathbf{c}), \tag{2.15}$$

where $\mathbf{c}$ is the central model control variate and $\mathbf{c}_k$ is the kth client model's variate that can be updated in two ways. When $\mathbf{c}_k$ is set to zero, SCAFFOLD becomes FedAvg.

FedDyn [96]: To control the client drift, FL with dynamic regularization (FedDyn) is introduced. It adds a regularization term to control the client drift and speed up the convergence to reduce the communication cost. Unlike SCAFFOLD that applies control variate on the client side, FedDyn partly applies it on the clients and partly on the central model side. The regularization term, in FedDyn, dynamically aligns global and local convergence points, and avoids client drifting from happening. The local model weights are updated as follows:

$$\mathbf{w}_k^t = \mathbf{w}_k^t - \eta[-\nabla F_k(\mathbf{w}_k^t) - \nabla F_k(\mathbf{w}_k^{t-1}) - \alpha(\mathbf{w}_c^{t-1} - \mathbf{w}_k^t)], \tag{2.16}$$

where α is a regularization term, and $(\mathbf{w}_c^{t-1} - \mathbf{w}_k^t)$ is a penalty term that is dynamically modified. The local model weights are then transmitted to the central model for computing central model gradients and updating the central model weights.

2.2 Generative Adversarial Networks (GANs)

2.2.1 GAN Basics

A GAN is a DL method that learns from original data distribution and generates synthetic data with similar distribution. The first GAN was introduced by Goodfellow [33] as shown in Figure 2.9. It consists of two DL networks: a generator G that accepts an input variable $\mathbf{z}$ and generates fake images, and a discriminator D that distinguishes whether the generated images are fake or real.

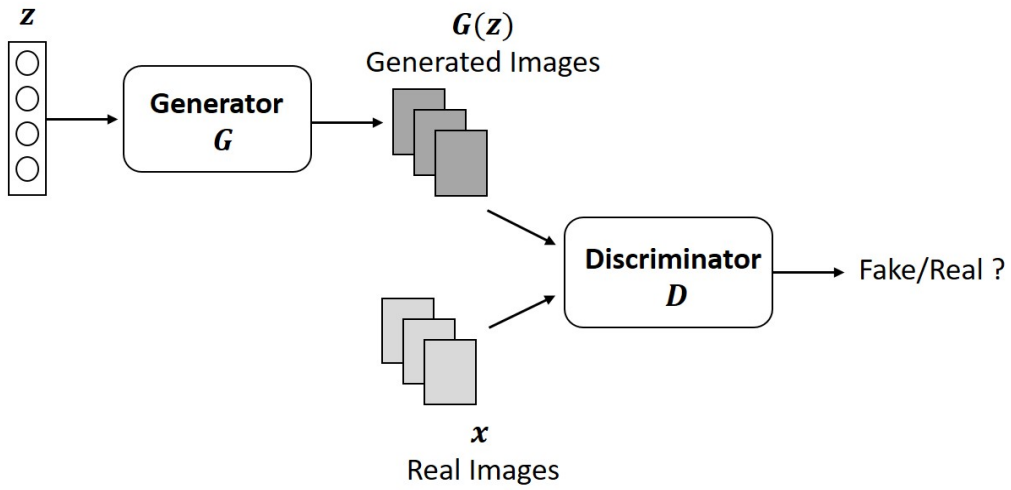

Figure 2.9: Generative Adversarial Network.

Given a prior distribution $p_{\mathbf{z}}(\mathbf{z})$ (usually a Gaussian distribution) of input variable $\mathbf{z}$, the generator $G(\mathbf{x};\theta_g)$ learns a mapping from $\mathbf{z}$ to a data generated distribution p_g over the target data $\mathbf{x}$, where θ_d are the learnable parameters of G. While the discriminator $D(\mathbf{x};\theta_d)$ learns with parameters θ_d to discriminate between $G(z)$ the generated samples coming from generated distribution p_g and the real image samples $\mathbf{x}$ from data distribution p_{data}. During training, both networks learn simultaneously, with G aiming to minimize $\log(1 - D(G(\mathbf{z})))$, and D aiming to maximize the loss function. G tries to generate images $G(\mathbf{z})$ with high probability to look real and obtain the goal $p_g = p_{data}$, while D tries to learn distinguishing between p_g and p_{data}. In other words, D and G aim to play the two-player minmax game by optimizing the following function:

$$\min_G \max_D V(G,D) = \mathbb{E}_{\mathbf{x}\sim p_{data}(\mathbf{x})} \log D(\mathbf{x}) + \mathbb{E}_{\mathbf{z}} \sim p_{\mathbf{z}(\mathbf{z})} \log(1 - D(G(\mathbf{z}))), \quad (2.17)$$

where D maximizes the objective function using gradient ascent, and G minimizes it using gradient descent. In practice, when D is close to optimal solution, it provides sufficient feedback to G to change its gradients. However, a highly precise D, where $D(x) = 1$ and $D(G(\mathbf{z})) = 0$ reduces the loss function to 0, resulting in 0 gradients and hence G gets a little feedback. A reasonable approach could be to maximize $\mathbb{E}[\log(D(G(\mathbf{z})))]$ rather than to minimize $\log(1 - D(G(\mathbf{z})))$. In this way, D and G would update iteratively, such that, when D is optimized for k steps, G is updated for one step on the mini-batch using stochastic gradient descent.

2.2.2 GAN Variants

There exists different types of GANs, which can vary mainly in 3 aspects as shown in Figure 2.10: based on architecture, based on the condition and objective functions of the discriminator and generators. These variants aim at improving the performance and image generation, a few of them are discussed below.

DCGAN [100]: Deep convolutional GAN (DCGAN) is introduced by using convolutional layers resulting in stable training and producing high resolution images. Firstly, it replaces the pooling layer with fractional-strided convolutions for G, and strided convolutions for D. Secondly, batch normalization is used in both G and D after each convoltuional layer, that helps to keep generated images and real ones centering at zero. This helps to deal with the training problem that might have caused either by poor initialization or by vanishng gradients. Finally, ReLU activation functions are used for hidden layers in G, except Tanh activation function for the last layer, while LeakyReLU for all G layers. Fully connected layers are used at the input of G and at the output of D. It has the drawback of suffering from mode collapse where it might not learn to produce some samples of the dataset. DCGAN has been used for data augmentation in [36] and for classification in [101], [102].

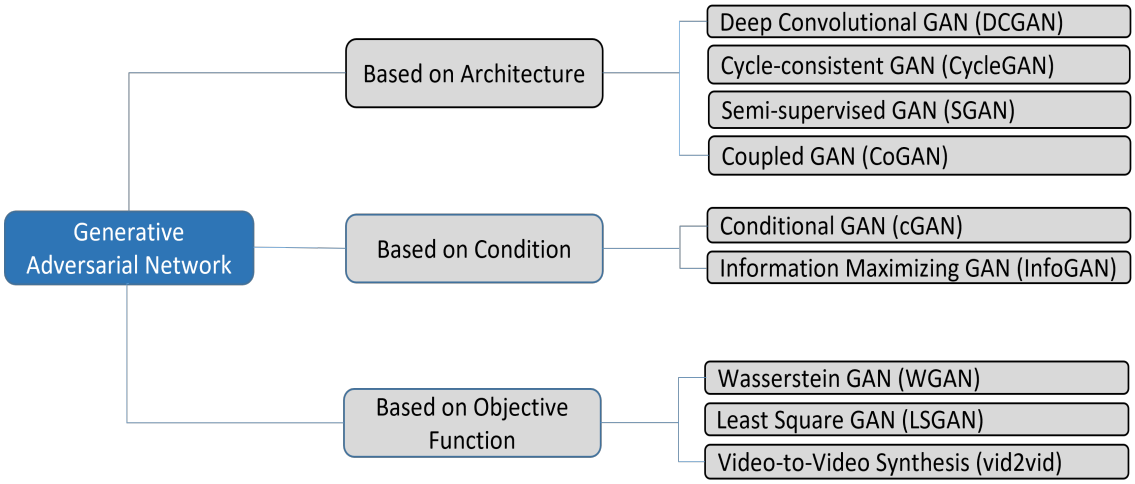

Figure 2.10: GAN taxonomy. There exists many other types of GANs which are out of scope of this thesis.

CycleGAN [103]: CycleGAN performs transferring of images from one domain to another domain. It employs cycle-consistency loss to guarantee the relation between input image and the corresponding output image as shown in Figure 2.11. CycleGAN consists of two generators X and Y and two dis-

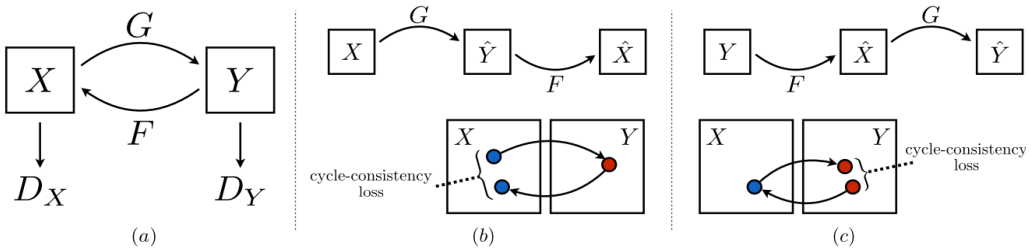

Figure 2.11: A schematic of cycleGAN taken from [103]. (a) G an F are the two mappings with their corresponding discrimnators D_Y and D_X. (b) Forward consistency loss is encouraged by $F(G(x)) \approx x$. (c) Backward consistency loss is encouraged by $G(F(y)) \approx y$.

criminators D_X and D_Y. Two mapping functions G and F are computed with their associated discriminators D_Y and D_X. Mapping G is conducted such that $G : X \rightarrow Y$ where D_Y learns to discriminate between $\hat{y} = G(x), x \in X$ and $y \in Y$. To cope with the unpaired inputs and outputs in the two domains,

an additional constrain is required such that the transformations remain cycle consistent. In other words, if x is transformed to $\hat{y}$ and is transformed back, then $\hat{x} \approx x$. This is obtained by using the other mapping function F in the cycle which maps $F : Y \rightarrow X$ and D_X is trained to discriminate between $\hat{x} = F(\hat{y})$ and x. Both the mapping functions are learned through the adversarial losses simultaneously with the additional cycle consistency loss for $F(G(x)) \approx x$ and $G(F(y)) \approx y$ to perform domain-to-domain transformation.

Semi-supervised Generative Adversarial Network (SGAN) [104]: The design of this GAN is based on the idea of multi-headed discriminator that can use either softmax function or sigmoid function. In this way, the discriminator D learns better and impacts the learning of G. The discriminator has a total of $N + 1$ outputs corresponding to N data classes with an additional fake class. SGAN is called semi-supervised because it uses labels for half of the mini-batch that has been drawn from the data generating distribution. However, SGAN limits the diversity of generated images. SGAN has been used in [105].

Coupled Generative Adversarial Networks (CoGAN) [106]: This GAN framework generates pairs of corresponding images in two different domains. It consists of two pairs of GANs, each responsible to generate images in one domain. It employs a simple weight-sharing constraint as shown in Figure 2.12 that enables learning of a joint distribution of images from multiple domains without them to be paired-images. This weight-sharing strategy reduces the

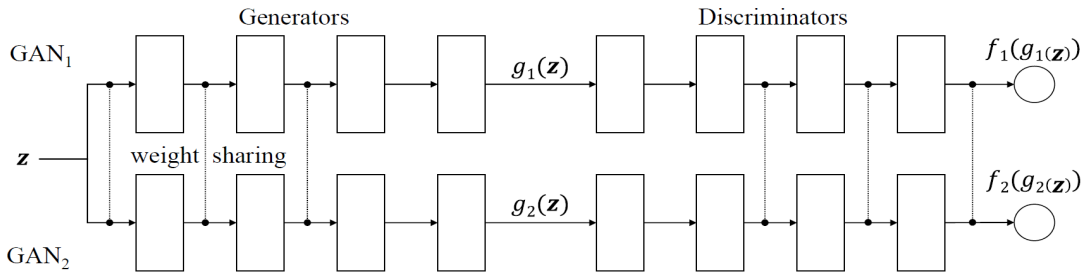

Figure 2.12: A schematic of CoGAN taken from [106]. It consists of two GANs; GAN_1 and GAN_2 with generators g_1 and g_2, two discriminators f_1 and f_2 and a common input z. The weights of first few layers of generators and the last few layers of discriminators are shared which allows the model to learn the join distribution of images without any supervision.

number of parameters. The training is done through back propagation alternatively, i.e., first train two discriminators one after the other, and then train two generators one after the other. The process repeats until convergence.

Conditional GAN (cGAN) [107]: A traditional GAN generates random images from the given input dataset irrespective of its class label. Conditional GAN (cGAN) allows the image generation with adding extra information to control the generated data images. Both D and G models are conditioned on class labels encoded as a one-hot vector. Further, they are concatenated with $\mathbf{z}$ in G and $\mathbf{x}$ in D on an embedding layer followed by a FC layer with linear function to scale this layer to the real input size before being presented to the model. For instance, if the images have class labels, they can be conditioned using their discrete class labels, but if the condition is any image, cGAN can perform image-to-image translation [108]–[111]. Although, cGAN requires pairs of input and output images for training, it is hard to obtain when applying in domain adaptation. cGAN has successfully been used for data augmentation [37], [112] and for synthesizing good quality images from text [113]. The loss function of cGAN is given as follows:

$$\min_{G}\max_{D} V(G,D) = \mathbb{E}_{\mathbf{x}\sim p_{data}(\mathbf{x})}[\log D(\mathbf{x}|c)] + \mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}(\mathbf{z})}}\log[(1 - D(G(\mathbf{z}|\mathbf{c})))]. \tag{2.18}$$

Information Maximizing GAN (InfoGAN) [114]: It is an extended version of cGAN that learns to maximize the mutual information between the conditional variable and the generated data to learn disentangled representation in an unsupervised manner. The noise vector is decomposed into two parts: in-compressible noise z and latent code c. A new classifier Q is defined that learns to estimate c given by $Q(c|x)$, where Q and D share all convolutional layers except the fully connected ones. The loss function of InfoGAN is the regularized form of cGAN, give as:

$$\min_{G}\max_{D} V_I(G,D) = V(G,D) - \lambda I(c; G/z, c)), \tag{2.19}$$

where $V(G,D)$ is objective function for cGAN, the difference is that the discriminator in InfoGAN does not take c as input. InfoGAN is good at disentangling objects and discovering visual concepts from the existing images.

Wasserstein GAN (WGAN) [115]: In traditional GANs, the model could suffer from instability and mode collapse. WGAN is introduced to reduce the Jensen-Shannon (JS) divergence with Earth-Mover (EM) distance. EM distance is differentiable and effectively solves mode collapse. It allows stable training of the GAN that supports the equilibrium between G and D, and provides high quality gradients to train G. The WGAN cost function is give as follows:

$$\min_{G} \max_{D \in \mathcal{D}} V(G, D) = \mathbb{E}_{\mathbf{x} \sim P_{data(\mathbf{x})}}[D(\mathbf{x})] - \mathbb{E}_{\mathbf{z} \sim p_{\mathbf{z}(\mathbf{z})}} \log[D(\mathbf{z})], \tag{2.20}$$

where $\mathcal{D}$ is the set of 1-Lipschitz function. Enforcing this constraint restricts the weights to lie within a compact space. This leads to large gradients being clipped off resulting in unstable training. WGANs have been increasingly used in computer vision applications for synthesizing good quality images.

Least Square GAN (LSGAN) [116]: The main idea of LSGAN is to replace sigmoid cross entropy loss with least-square loss function. In original GAN, the discriminator is considered as a classifier with sigmoid cross entropy loss function that leads to vanishing gradient problem. When the fake images are classified as real images, it creates no error. It considers those images on the correct side of the decision boundary even though, in real, those images are far from the real ones. LSGAN uses the least square losses, V_G, for the generator and V_D for the discriminator that enables generation of high quality images and penalizes the images that are far from the real ones to sustain more stability. These losses are given as below:

$$\min_{D} V_D = \frac{1}{2}\mathbb{E}_{\mathbf{x} \sim p_{data}(\mathbf{x})}[(D(\mathbf{x}) - a)^2] + \frac{1}{2}\mathbb{E}_{\mathbf{z} \sim p_{\mathbf{z}}(\mathbf{z})}[(D(G(\mathbf{z})) - b)^2],$$

$$\min_{G} V_G = \frac{1}{2}\mathbb{E}_{\mathbf{z} \sim p_{\mathbf{z}(\mathbf{z})}}[(D(G(\mathbf{z})) - a)^2], \tag{2.21}$$

where a is the label for real images and b for the generated images.

Video-to-Video Synthesis (vid2vid) [117]: Like image-to-image mapping, this GAN aims at learning a mapping function from, e.g., a human pose video, and synthesizes realistic videos of the given example images according to the

video poses. This GAN uses cGAN with a Markov assumption, where the t-th frame is generated considering the previous L frames, including the source images and the generated images. Since the consecutive frames in a video have redundant information, the next frames can be estimated by warping the current frame given that the optical flow is known. The adversarial loss of vid2vid is given as follows:

$$\max_{D} \min_{G} V_D = \mathbb{E}_{(\mathbf{x}_1^T, \mathbf{s}_1^T)}[\log D(\mathbf{x}_1^T, \mathbf{s}_1^T)] + \mathbb{E}_{\mathbf{s}_1^T}[\log(1 - D(G(\mathbf{s}_1^T), \mathbf{s}_1^T))], \quad (2.22)$$

where $\mathbf{s}_1^T$ are the sequence of source video frames and $\mathbf{x}_1^T$ are the sequence of corresponding real video frames. vid2vid learns a mapping function to convert $\mathbf{s}_1^T$ to a sequence of $\mathbf{x}_1^T$. However, this model does not respond well against unseen objects and scenes and requires re-training for the new input. This generalization problem has been solved in few shot video-to-video synthesis [118], which can synthesize videos of unseen inputs.

2.2.3 Applications of GANs

GANs were first used in non-medical images with promising results. This includes style transfer [119], super resolution [120], [121], sequential data generation [122], image-to-image translation [103], [108], [123], data augmentation [124], domain adaptation [125], object detection [126], image registration [127], [128] etc. These methods were later extended to medical images. We keep our focus limited on data augmentation and domain mapping as the GAN applications which are discussed below.

Data Augmentation

It is a technique to increase the training dataset size by adding synthetic data similar to the input data distribution. DL network performance relies on the size of training dataset, which is a constraint in medical area, where the datasets are usually small in size. Some medical datasets have more number of one class images than the other, i.e., class imabalance images. Data augmentation is one of the most common ways to increase the number of training images for improving classification performance and balancing the classes. Typical ways of augmentation are random rotation, shift, zoom, elastic de-

formations, scaling and flipping [47], [129], [130]. Augmentation based on DL methods produces more realistic results, e.g., GAN augmentation may synthesize MRIs by changing tumor size, tumor location and contextual details from the input image distribution. Qi [35] used an attention guided CycleGAN for MRIs to augment a tumor in normal MRIs and returned normal MRIs from the ones with tumor. Mok [37] proposed a cGAN to generate generic augmented images that led to an improved dice score from brain tumor segmentation network. GANs have also been used for generating synthetic images for many other medical images. Motamed [131] proposed Inception-augmentation GAN (IAGAN) on chest X-rays for the detection of pneumonia. The ability of GAN was improved to learn more details from the training dataset while maintaining the spatial information. Another study [132] on chest X-ray images used AC-GAN for generating synthetic images to improve the classification performance. Frid-Adar [36] trained a separate DCGAN for each class of liver lesion that led to improved classification performance. Tekchandani [133] generated augmented images of benign and malignant mediastinal lymph nodes by using different GAN architectures such as cGAN, DCGAN, WGAN, AC-GAN and InfoGAN.

Domain Mapping

Many GAN based DL networks have been introduced for mapping the input image to an output with a different appearance but with the same underlying structure. This image transformation is more challenging for medical images than the visual images, because the amount of detailed structure presented in medical images could be distorted in the process. Medical images could be of different dimensions and contrasts that offer diversity in diagnostic choices. At the same time, it is considered a challenge to translate them among different modalities or among various acquisitions within one modality. Various GANs (cGAN, CycleGAN, StarGAN and inforGAN) have been used to perform domain mapping for inter or intra-modality. Among them, CycleGAN has been most popular because of its capability of dealing with unpaired data images. Yang [134] used a structure constrained CycleGAN [103] on unpaired images to transform brain MRIs to CT scans. The structure consistency loss between the synthetic and real images was defined based on modality independent neighborhood descriptor to constrain structural consistency. Instead

of random selection of training images, a position based selection strategy was employed. Armaniousa [135] proposed MedGAN to perform PET-CT transformation, correction of MR motion artefact and PET image denoising on brain images. The generator architecture (CasNet) consisted of a cascaded U-net to produce sharp transformed image. The discriminator was used as a trainable feature extractor that penalizes the difference between transformed image and the desired modality. Nie [136] used a patch-based GAN to transform MRIs to CT images. It used a CNN as a generator and proposed an auto-context model for image refinement. Lie [86] used a CycleGAN in which the generator used a dense block-based network to produce CT images from the brain MRIs. Yang [137] developed a cGAN based framework that exploited low and high level features for cross-modality (between T1, T2 and FLAIR) transformation. Most of the above methods performed mapping between only two domains at a single time. On the contrary, StarGAN [138] performed a multi domain transformation using a single generator where the domains had no feature mismatch. The generator took the target domain as an additional input. Another study that performed mapping among multiple domains was seen in RadialGAN [139] that overcame feature mismatch.

Despite GANs have shown promising results in many applications, they have rarely been used to generate synthetic data and perform domain mapping on the datasets of molecular subtypes of gliomas, where retaining the subtle molecular information of tumors in MRIs is important. This motivate us to generate synthetic MRIs containing gliomas that may assist to improve performance in molecular- based classification.

2.3 Deep Learning (DL) for Tumor Segmentation

Brain tumor segmentation aims at localization of tumor area and its subregions with respect to surrounding tissues. The output of a segmentation is an image of the same dimension as that of the input image, where each pixel is assigned a label indicating that the pixel belongs to which subregion of tumor area. A manual annotation, marked by a medical expert is reffered to as the ground truth (GT) label/annotation. Figure 2.13 shows an example of a FLAIR-MRI with GT tumor annotation with its subregions: necrotic and non-enhancing tumor (red), peritumoral edema (green) and enhancing tumor

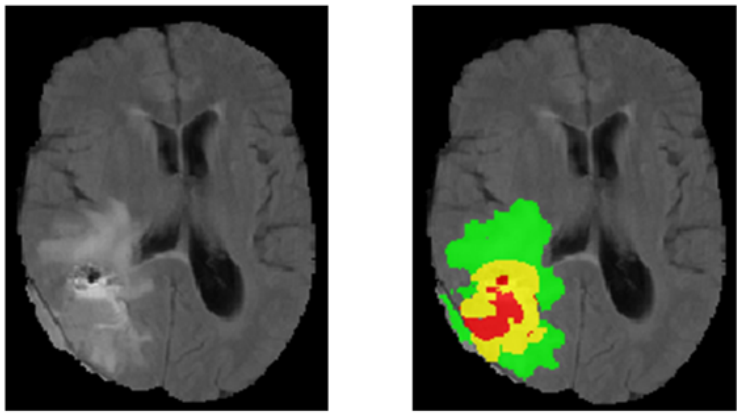

Figure 2.13: Axial slice of FLAIR-MRI with its GT annotation from BraTS'17 dataset. **Red:** Necrotic and non-enhancing tumor. **Green:** Peritumoral edema. **Yellow:** Enhancing tumor.

(yellow). Manual segmentation is a time-consuming process and the quality of annotation is highly dependant on the medical expert's skill. Further, there exists high probability of inter-observer variations. Thus, manual segmentation is not very practically feasible when the dataset size is large.

2.3.1 U-Net

It is a U-shape DL architecture that is developed for the segmentation of medical images. It consists of an encoder and a decoder [140]. The encoder part covers down-sampling operations to learn context, where regular convolutional layers with max pooling layers contract the size of the input image as the depth proceeds. The U-Net architecture is depicted in Figure 2.14. The encoder part is further proceeded with successive upsampling layers that localize the high resolution features. Since the U-shape of the model is symmetric, a large number of feature channels are allowed to propagate information to higher resolution layers assisting to generate a more precise segmentation map. The network uses input images and their corresponding segmentation maps for supervised training the network with SGD.

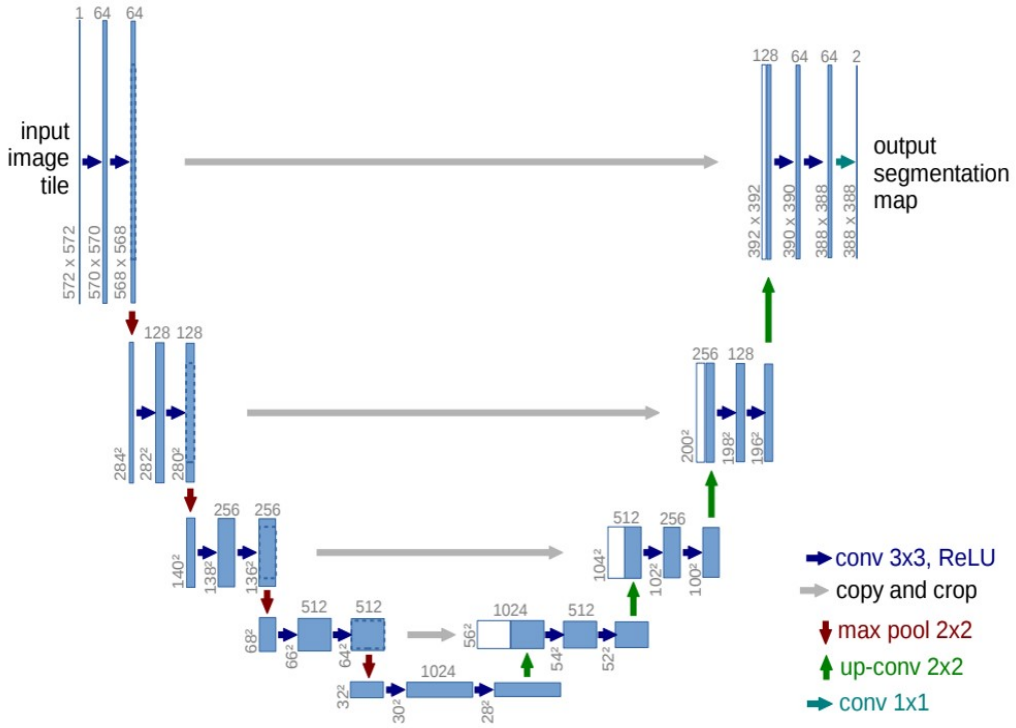

Figure 2.14: U-Net architecture proposed by Olaf Ronneberger et al. [140]

2.3.2 Other DL Models for Segmentation

There exist several challenges in segmenting medical images compared to the non-medical ones. The segmentation algorithms are often specific to applications, image modality and the body organ under study. Typically, medical scans have both low contrast details and also moderate to high level of noise. Image ambiguities such as partial volume effects (more prominent in brain images) and inter and intra-patient variability further increase the challenge. Many algorithms have been introduced to improve the segmentation performance on brain MRIs, a few of them are discussed below.

V-Net[141]: V-Net is proposed to use a volumetric CNN that performs segmentation on 3D MRI prostate volumes. An objective function based on the Dice overlapping coefficient is optimised during an end-to-end training. The network architecture consists of a compression path and a decompression path. The compression path comprises of different stages that operates at different

resolutions. Each stage consists of one to three convolutional layers and is formulated such that they learn a residual function which increases the convergence rate. At the output layer, a softmax function is used to predict the probability of each voxel that belongs to foreground and to background, and Dice overlap coefficient is optimized between predicted binary segmentation volumeand the GT binary volume.

VoxResNet[142]: This study proposes a novel voxel wise residual network (VoxResNet) for the segmentation of the key brain tissues (WM, GM ans CSF) from 3D MRIs. It extends the 2D residual learning into a 3D variant with a deeper architecture that consists of 25 layers. Residual learning trains such a deep network to alleviate the degradation problem such that the performance gain achieved by increasing the network depth is fully leveraged. During training, multi-modality and multi-level contextual information are integrated, such that the complementary information of different modalities assist in improving the segmentation performance. For evaluation metrics, Dice coefficient, Hausdorff distance and absolute volume difference are used. To further boost the performance, an auto-context version of VaxResNet is introduced that uses different levels of contextual information.

2.3.3 Segmentation Evaluation

Image segmentation aims at finding a segmentation map/label $\mathcal{L}$, that is as similar to the GT label $\mathcal{L}_{GT}$ as possible:

$$\mathcal{L}^* = \arg \max_{\mathcal{L}} S(\mathcal{L}, \mathcal{L}_{GT}), \tag{2.23}$$

where S measures the similarity between the two labels. Segmentation algorithms are first trained on a training image dataset with GT labels that maximizes the similarity in (2.23). After that, the algorithms are tested on unseen images to predict the tumor areas. Out of many choices, one commonly used similarity metric to evaluate the segmentation results is the Dice coefficient:

$$S_{\text{DICE}} = \frac{2|\mathcal{L} \cap \mathcal{L}_{GT}|}{|\mathcal{L}| + |\mathcal{L}_{\mathcal{GT}}|}, \tag{2.24}$$

where both $\mathcal{L}$ and $\mathcal{L}_{GT}$ are label binary maps in binary for one class. For labels with multi-class, the mean Dice metric over all classes is computed. Another commonly used metric is the Jaccard index (Intersection over Union), given as:

$$S_{\text{JACCARD}} = \frac{|\mathcal{L} \cap \mathcal{L}_{GT}|}{|\mathcal{L}| \cup |\mathcal{L}_{\mathcal{GT}}|}. \tag{2.25}$$

The two metrics are related by $S_{\text{DICE}} = \frac{2S_{\text{JACCARD}}}{(1+S_{\text{JACCARD}})}$. Both metrics take values from 0 to 1, with values closer to 1 the better. Other similarity metrics have also been found in literature such as Hausdorff distance and the mean surface distance, that are used when qualitative segmentation shape is important.

CHAPTER 3

Summary of the Main Contributions

This chapter describes the methods developed in this thesis and are depicted in Figure 3.1.

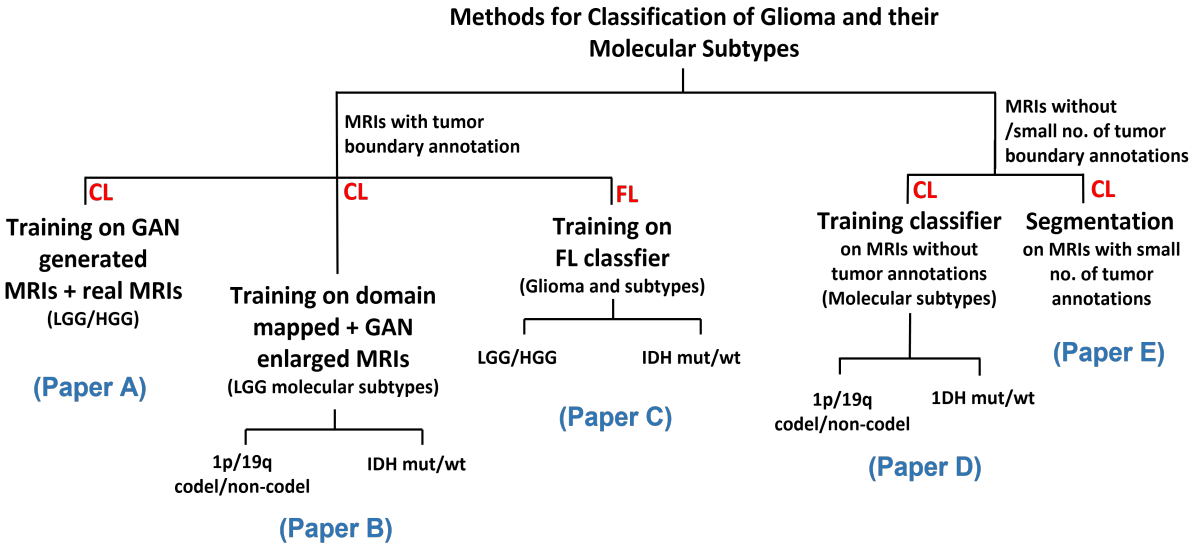

Figure 3.1: A summary of the topics developed in the thesis.

The proposed methods on the classification of glioma and their subtypes and the tumor segmentation, as depicted in Figure 3.1, are briefly described as follows:

- **Enlarging training dataset:** DL networks need large datasets to provide reliable performance. Brain tumor dataset is often small in size. Motivated by this, we propose to tackle this problem by enlarging the dataset. Paper A and B address this issue.
- **Domain mapping:** Datasets from different hospitals suffer from domain shift, due to different patient cohorts, scanners and scanner settings. Therefore, combining them likely would not help to increase the classification performance. Motivated by this issue, a domain mapping method is proposed. Paper B and C address this issue.
- **Training on multiple datasets with individual dataset protection:** Sharing hospital datasets puts constraints under data protection regulations, which limits training dataset size in central learning (CL). Motivated by this issue, we employ a federated learning (FL) method where hospitals may hold their own datasets while training local DL networks that aggregate to converge a central model. Paper C addresses this issue.
- **Training on datasets tackling tumor boundary annotations:** Manually drawn tumor annotation is required for supervised DL. However, this requires time and medical expertise. To mitigate this problem we propose: 1) training a classifier on MRIs with ellipse bounding box tumor areas instead of GT tumor boundary annotations. 2) performing segmentation where a large number of MRIs use bounding box tumor areas and a small number of MRIs use GT annotation. Paper D and E address this issue.

3.1 GAN for Data Augmentation and Multi-Stream CAE for Glioma Classification

(Summary of Paper A)

Sub-problem addressed: (1) When training dataset is too small, DL networks do not provide reliable performance. (2) Improved performance is required for glioma (LGG/HGG) classification.

Motivations: A DCGAN is employed to generate synthetic MRIs for multiple modalities. Hence, one can use GAN augmented MRIs in addition to the real MRIs for training. A multi-stream convolutional autoencoder (CAE) is proposed for feature learning of MRIs and fusion for glioma classification.

Basic idea: The basic idea behind this study is to enlarge the training dataset by generating synthetic MRIs with the similar distribution. The proposed classifier learns complementary information from multi-modality MRIs which are fused followed by classification.

Contributions:

- Employing a deep convolutional GAN (DCGAN) for generating multi-modality MRIs to enlarge and balance the training dataset.
- Proposing a 3-stream CAE-based classifier for end-to-end glioma feature learning, feature fusion and classification.
- Employing a 2-round training strategy: pre-training on GAN synthetic MRIs followed by refined-training on original MRIs.
- Evaluating the performance and comparing with the state-of-the-art methods.

Proposed method: The pipeline of the proposed framework is shown in Figure 3.2, where the training 2D MRIs from multiple modalities (T1ce, T2 and FLAIR) are fed into a DCGAN to generate synthetic MRIs for each

modality. The training of the network is conducted in two rounds: pre-training and refined-training. Based on the number of MRI modalities available, 3 streams of CAEs are trained in the pre-training round, where each stream is trained separately for feature learning on individual MRI modality. Each stream of CAE has an encoder (6 convolutional layers with max pooling layers in between) and a decoder (5 layers of deconvolutional layers each followed by an upsampling layer). The encoders at all streams are fused in the fusion layer. Finally, the network is refined-trained on original MRIs to further improve the learned features.

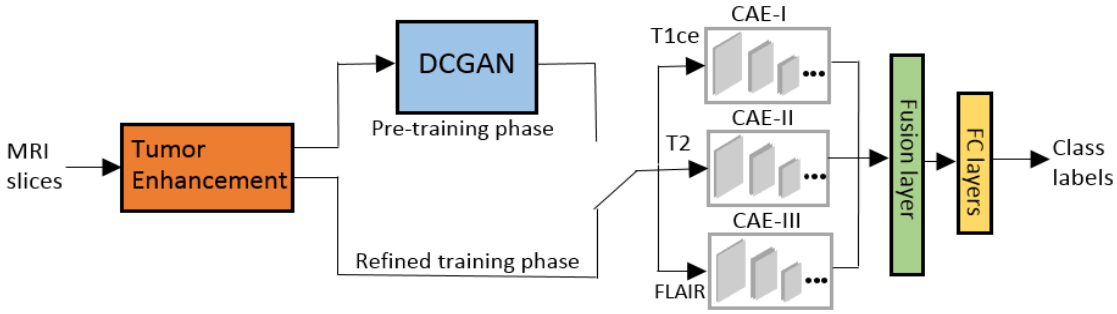

Figure 3.2: The block diagram of the proposed scheme for glioma classification.

Evaluation: The proposed method was conducted on MICCAI BraTS'17 dataset for glioma (HGG/LGG: 210/75 patients) classification. The classification performance of the proposed method on individual MRI modality and on the fused multi-stream network are shown in Table 3.1. Observing the 4th column of Table 3.1, the fusion of feature information shows a boost in test accuracy compared to the individual MRI-modality.

Table 3.1: Average test performance (over 5 runs) of the proposed scheme on individual MRI-modality and multi-modality inputs.

Acc. %(| σ |) on T1ce	Acc. %(| σ |) on FLAIR	Acc. %(| σ |) on T2	Acc. %(| σ |) on 3-Modality Fusion
86.90(0.61)	81.75(1.78)	73.25(1.65)	92.04(1.03)

To observe the effect of data augmentation on performance improvement, different sizes of augmented MRIs have been tested as shown in Figure 3.3. In-

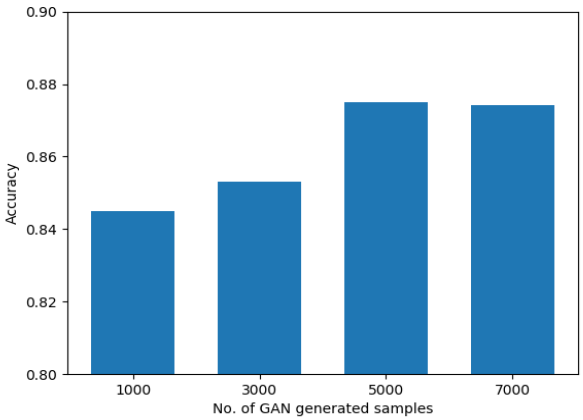

Figure 3.3: Effect of DCGAN augmented image size on test accuracy for T1ce-MRIs.

creasing the number of augmented MRIs have shown noticeable performance improvement up to 5000 images.

Comparison: A performance comparison is done on test accuracy with a few existing studies based on the same dataset for LGG/HGG classification as shown in Table 3.2. The results from the proposed method shows improved performance.

Table 3.2: Comparison with existing methods for HGG/LGG classification using BraTS dataset.

Method	# of Subjects	Test Accuracy (%)
Ye[143]	274	82.10
Ge[144]	285	88.07
Ge[145]	285	90.87
Proposed Scheme	285	92.04

Conclusion: Enlarging the training dataset by DCGAN augmented MRIs has shown to improve the performance. The multi-stream CAE-based classifier shows that the fusion of multi-modality MRIs gives complementary information to improve the classification performance.

3.2 Glioma Molecular-Subtype Classification based on Domain Mapped Data

(Summary of Paper B)

Sub-problem addressed: (1) Lack of efficient methods for combining several small MRI datasets into a large training dataset due to domain shift, while maintaining the molecular subtle information in augmented MRIs. (2) To identify whether molecular subtype information in LGG MRIs is retained after employing domain mapping.

Motivations: Identifying molecular subtypes of LGG is important for prognosis and timely treatment. A brain tumor sample is often used obtained through biopsy. DL methods might help to assist diagnosis by learning from existing MRIs with available biopsy information. However, such methods need large training dataset. MRI datasets are often obtained from local hospitals with different acquisition protocols. This calls for an efficient domain mapping method that may allow to combine datasets into a common domain, while retaining the subtle molecular information in MRIs.

Basic idea: This study is based on the main idea of enlarging the size of the training dataset from multiple MRI datasets. For this purpose, a framework is proposed that uses CycleGAN to map clinical MRIs to a target domain which can retain the molecular subtype information.

Contributions:

- Mapping several small datatsets to a common domain by using CycleGAN, while retaining tumor characteristics on the molecular level.
- Further enlarging and balancing the training dataset in different classes by using DCGAN.
- Using bounding boxes for MRIs instead of GT tumor boundary annotations.
- Applying a two round training strategy for effective feature learning. Pre-training a multi-stream CAE on DCGAN augmented data, and

refined-training on MRIs.

Proposed Method: The proposed scheme is shown in Figure 3.4, that consists of three modules: (i) mapping datasets to a common domain by unpaired-CycleGAN; (ii) augmenting the training dataset by DCGAN to further enlarge and balance the training dataset in different classes; and (iii) employing a 2 round training strategy on a multi-stream CAE classifier. Two modality MRIs (T1ce, FLAIR) are fed into the CycleGAN to perform mapping from a source domain A to produce $\tilde{A}$ mapped MRIs. In our experiments, we only map one source dataset to a target domain B. The total dataset D is obtained as $D = \{\tilde{A} \cup B\}$. To further enlarge the size and balance the classes in the training dataset, $\tilde{D}_{train}$ is obtained using DCGAN. The tumor extraction block fixes tight rectangular bounding boxes around ROIs. A 2-round training strategy is then performed on a multi-stream CAE classifier [146].

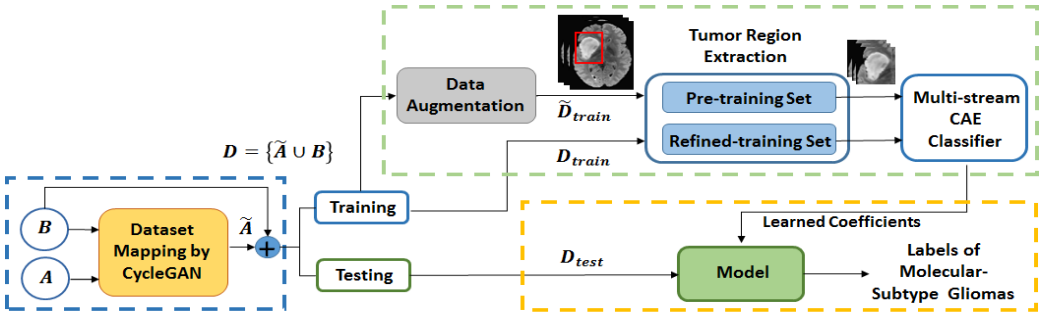

Figure 3.4: The block diagram of the proposed scheme. Blue dash box: domain mapping; Green dash box: feature learning step; Yellow dash box: testing step

Figure 3.5 shows the concept of an unpaired-CycleGAN for domain mapping from a source domain to a target domain.

Evaluation: This method is tested on two datasets (USA and France) for two cases of molecular subtypes in LGG MRIs. Case-A is used for 1p/19q codeletion prediction while case-B is used for IDH mutation prediction. Since USA dataset proved to give better performance for both case studies than the France dataset, it was chosen as the target domain. Figure 3.6 shows before and after mapped MRIs from France dataset domain to US dataset domain.

The performance was tested with different number of augmented MRIs and the one with the best test performance was selected. The network was then

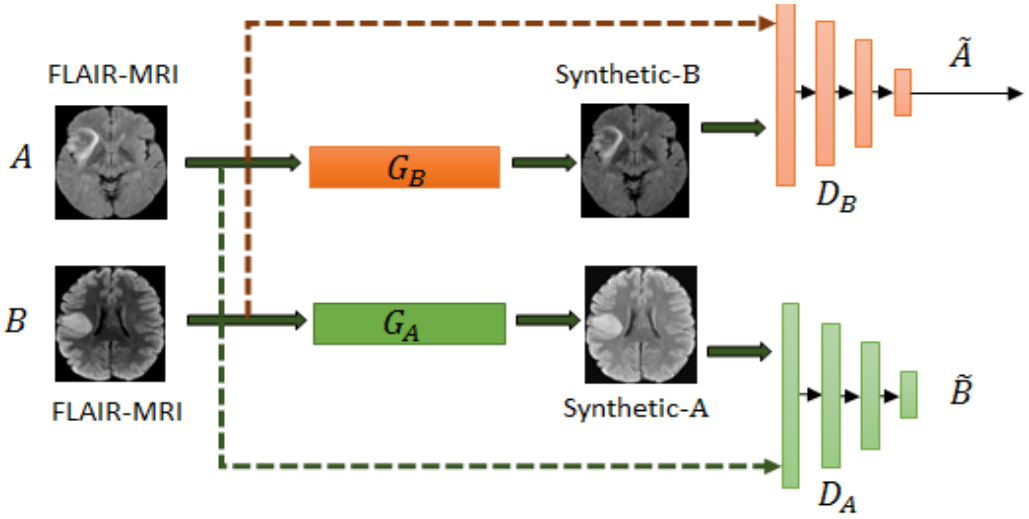

Figure 3.5: Illustration of domain mapping from domain A to domain B for FLAIR-MRIs. G_A and G_B are generators and D_A and D_B are the discriminators.

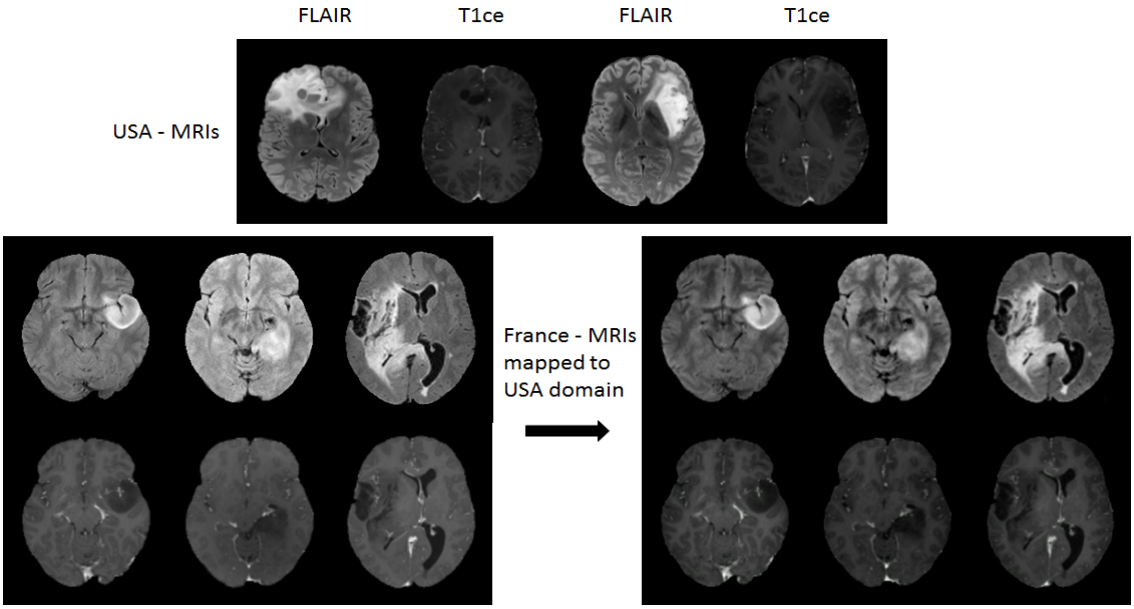

Figure 3.6: Visual inspection of 2D images for FLAIR-MRIs mapped from a source domain to a target domain.

trained using a 2-round training strategy. The mean test performance for the classification is shown in Table 3.3.

Comparison: Figure 3.7 shows the comparison of classification performance

Table 3.3: Average test accuracy of 2 datasets (over 5 runs) with domain mapping for two case studies.

Case study	Test Accuracy %(| σ |)
Case-A (1p/19q codel/non-codel)	74.81(0.98)
Case-B (IDH mut/wt)	81.19(3.70)

based on using domain mapping and without using it on both case studies. In

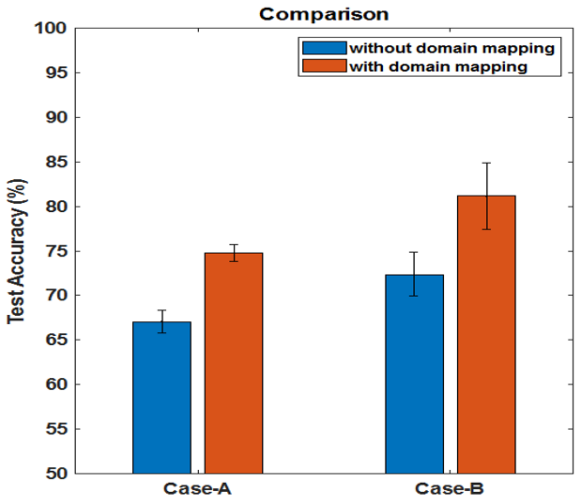

Figure 3.7: Comparison of the test results for classification with/without domain mapping.

case-A, the test accuracy (74.81%) was improved by 7.78% for 1p/19q codeletion prediction. In case study-B, the test accuracy (81.19%) was improved by 8.81% for IDH mutation.

Conclusion: The experimental results have indicated that the proposed domain mapping approach is effective while retaining the molecular information in LGG MRIs.

3.3 Federated Deep Learning for Gliomas and Their Molecular-Subtype Classification

(Summary of Paper C)

Sub-problem addressed: (1) Training datasets from individual hospital are usually small. However, under data protection regulations, sharing datasets could be difficult. (2) In addition to the above problem, datasets from different hospitals have domain mismatches.

Motivation: FL techniques have so far not been studied for classification of molecular subtypes of gliomas. This study is aimed at training several participating hospitals on their private datasets to update a central model without sharing their datasets centrally.

Basic idea: We extend the existing FedDyn by replacing a new cost function to mitigate the problem of class imbalance and also employing a multi-stream architecture in the network. A comparison is made to that of a central learning (CL) approach to see whether the FL-based classification performs well and whether it is possible to replace the FL scheme with the CL ones.

Contributions:

- Proposing EtFedDyn classifier that is an extension of FedDyn with focal loss cost function to mitigate the class imbalance problem with a multi-stream network architecture for multi-modality MRIs.
- Additionally, applying domain mapping to overcome the domain shift among multiple datasets.
- Adding a 3D scan-based post-processing based on a majority voting-based criterion.
- Comparing the performance with that of a CL approach to see whether CL approaches can be replaced by FL ones.

Proposed Method: The proposed scheme is shown in Figure 3.8 that consists of domain mapping, individual hospital local learners and central model update in training process. In testing process, a 3D-based post-processing step is added.

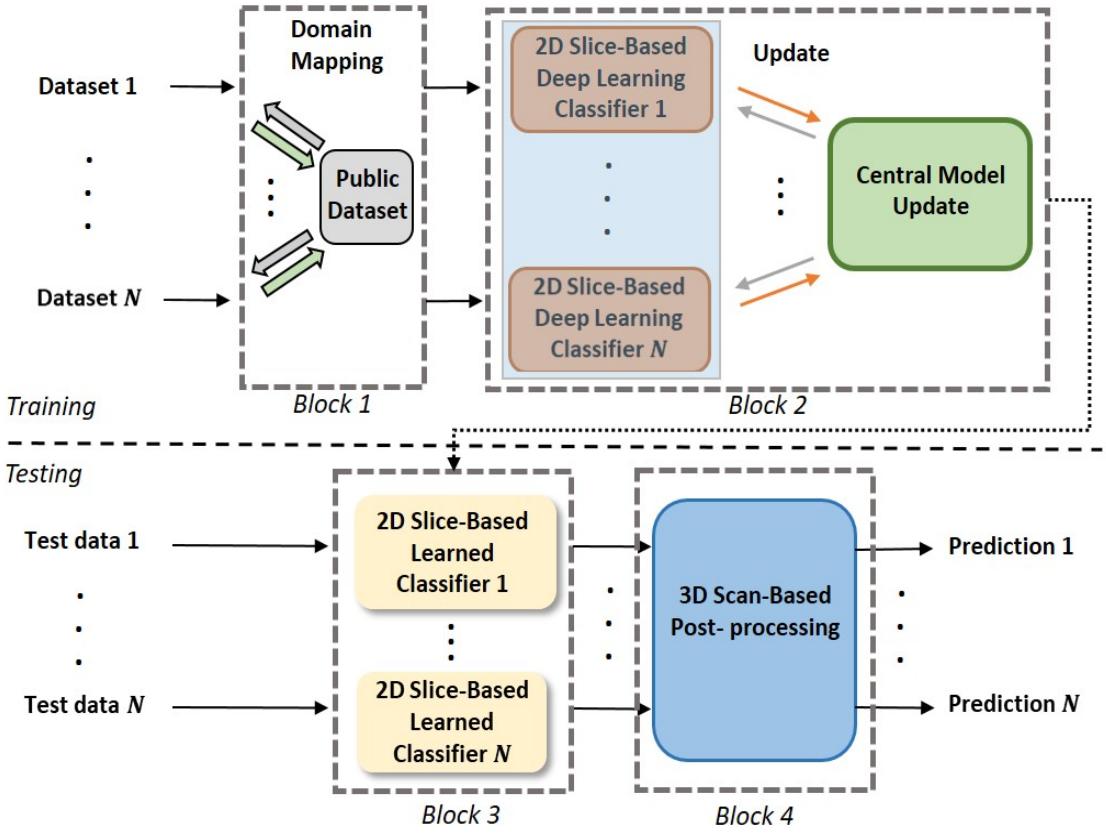

Figure 3.8: The pipeline of the proposed scheme using FL-based ETFedDyn classifier for prediction of gliomas and their molecular subtype.

The proposed 2D EtFedDyn classifier is an extension of FedDyn [96], but with a focal loss cost function and multi-stream setting. The focal loss function tackles class imbalances by emphasizing on the errors caused by hard class and downeighs the confidence in predicting the easy class during training. The network consists of 2-streams of CNNs and weighted sum of features from both streams are computed at feature fusion layer, such that weights may be learned adaptively based on their modality-specific features. During training, each local learner trains the network on its own dataset and communicates the updated weights for the central model update, and this continues

until convergence. During testing, the predicted 2D MRIs are fed to a post-processing step for making a 3D scan-based prediction by using a majority voting-based criterion.

Evaluation: The proposed scheme is tested on two case studies. Case A study was conducted on US and TCGA dataset for IDH mutation and wild-type prediction. Case B study was conducted for glioma grades (LGG/HGG) by partitioning MICCAI'17 datasets into two local clients (MICCAI 1 and MICCAI 2) according to patient scans. The overall performance from the proposed scheme on these case studies is shown in Table 3.4.

Table 3.4: Average test performance (over 5 runs) of proposed 3D scan-based FL scheme on the test sets for two case studies.

Case Study	Dataset	3D Acc. %(| σ |)	3D Sen. %(| σ |)	3D Spe. %(| σ |)
A	TCGA	85.46(3.53)	78.18(7.27)	89.09(4.63)
(IDH mut/wt)	US	75.56(2.72)	78.57(6.38)	65.00(12.25)
B	MICCAI 1	89.28(2.26)	79.99(6.99)	92.38(3.81)
(LGG/HGG)	MICCAI 2	90.72(1.75)	82.85(5.71)	93.33(2.34)

Detailed empirical analysis was conducted to verify the contributions of individual parts of scheme on both datasets and is summarized below:

- The 3D prediction results are compared with the performance of 2D EtFedDyn classifier. The effect of adding post-processing has improved accuracy by (TCGA: 2.11%, US: 2.23%) on case A study and (MICCAI 1: 1.81%, MICCAI 2: 2.39%) on case B study.
- The effect of using different loss functions with EtFedDyn classifier has been examined. Focal loss function improves accuracy by (TCGA: 1.66%, US: 3.25%) on case A study and (MICCAI 1: 1.19%, MICCAI 2: 1.85%) on case B study.
- Comparison of the performance from the proposed EtFedDyn to that of FedAvg classifier showed that EtFedDyn has improved the accuracy by (TCGA: 1.05%, US: 1.55%) on case A study and (MICCAI 1: 1.23%,

MICCAI 2: 1.81%) on case B study. It has also improved the convergence speed by nearly 50%.

- The domain mapping effect was evaluated on case A study only, since the datasets were obtained from different sources. It has improved the test accuracy by (TCGA: 0.4%, US: 1.85%).

Comparison: The difference between the CL approach and the proposed FL approach was evaluated and is shown in Table 3.5.

Table 3.5: Performance comparison on 3D scan-based test results of the proposed FL vs. its corresponding CL scheme on 2 case studies.

Case Study	Proposed FL %(| σ |)	Corresponding CL %(| σ |)	Performance Difference
A	81.96(2.88)	83.13(2.94)	-1.17
B	89.88(1.68)	90.71(1.33)	-0.83

Conclusion: From the experimental results on both case studies, one can observe that there is a slight reduction in performance of the proposed FL as comparing with the corresponding CL ones on the datasets used. This indicates that FL can be a promising option to replace the CL classifier.

3.4 Glioma Subtype Classification without Using Tumor Boundary Annotations

(Summary of Paper D)

Subproblem addressed: Supervised training of a DL network requires previously existing biopsy labels and tumor boundary annotations in the datasets. However, manually marking tumor boundaries by medical experts is a time consuming process. This study aims at looking for an alternative approach.

Motivation: Inspired by computer vision community's research on visual object tracking by using bounding boxes, this study explores the use of ellipse bounding boxes for tumor areas instead of using GT tumor boundary annotation for training a DL classifier.

Basic idea: This study is aimed at using ellipse bounding boxes for tumor areas and at exploring whether the classification performance degrades when tight tumor bounding boxes are used.

Contributions:

- Proposing an alternate paradigm for extracting tumor areas using ellipse bounding boxes from MRIs without GT tumor boundary annotations.
- Training and testing the classifier on MRIs with ellipse tumor bounding box areas for two datasets.
- Comparing the classification performance of the network trained on bounding box tumor areas to that of the one trained with GT annotated MRIs.

Propose Method: The block diagram of the proposed approach is shown in Figure 3.9, where the blue dotted block is for tumor region extraction that separates the tumor areas either by using tight ellipse bounding boxes (point **a'**) or by using GT tumor annotation (point **b'**). The shape of the bounding

boxes is chosen as ellipse considering the shapes of tumors. A multi-stream 2D CNN is first trained on MRIs with bounding box tumor areas obtained at point **a'** in Figure 3.9. The classification performance obtained is then compared with the same network trained on MRIs with GT tumor annotations obtained at point **b'** in Figure 3.9.

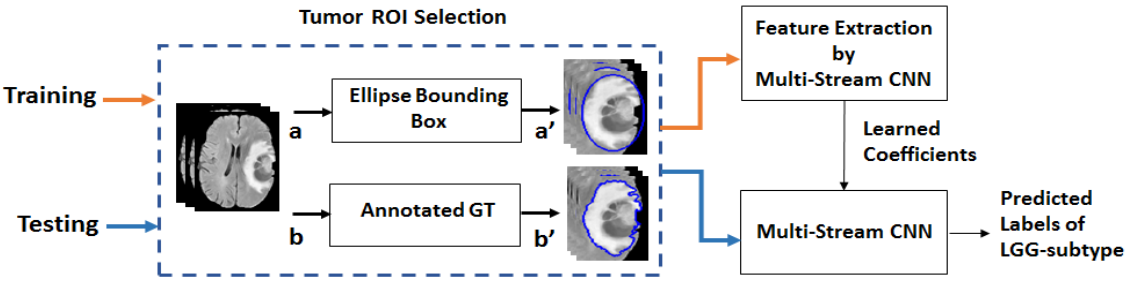

Figure 3.9: The pipeline of the method based on proposed strategy. **Blue dash box:** Tumor areas separated by ellipse bounding box and tumor boundary annotations. **Orange arrow:** Training phase. **Blue arrow:** Testing phase.

Evaluation: This method was tested on two datasets. TCGA dataset consists of LGG/HGG gliomas and US dataset consists of only LGGs. Table

Table 3.6: Comparison of the average test results (over 5 runs) for glioma subtype using ellipse bounding box tumor areas. Case-A for US dataset (1p/19q prediction). Case-B for TCGA dataset (IDH genotype).

Case Study	Acc. %(| σ |)	Sen. %(| σ |)	Spe. %(| σ |)
A (1p/19q codel)	69.86(2.46)	74.20(4.39)	64.60(1.92)
B (IDH mut)	79.50(2.12)	72.32(1.67)	86.65(3.28)

3.6 shows the experimental results. One can observe that the classification performance on US dataset seems more challenging due to non-enhanced hyperintensive tumor areas in LGGs.

Comparison: A comparison was performed on training the DL network on ellipse tumor areas to that of the network trained on GT tumor annotated

areas. Comparing the performances in Table 3.7, one can see a slight degraded performance on the test datasets using ellipse bounding boxes by 2.92% in US dataset and by 3.23% in TCGA dataset.

Table 3.7: Performance difference on average prediction results (over 5 runs) by using GT tumor areas and ellipse tumor bounding box areas for training.

Case Study	Ellipse Tumor Area Acc. %(| σ |)	GT Tumor Area Acc. %(| σ |)	Difference
A	69.86(2.46)	72.78(1.45)	-2.92
B	79.50(2.12)	82.73(1.82)	-3.23

Conclusion: Some non-tumor pixels were included in ellipse bounding box areas. This caused slight degradation in classification performance. The method can be used as a trade-off in terms of saving annotation time and accepting a small performance degradation.

3.5 Brain Tumor Segmentation using 2D Ellipse Box Tumor Areas

(Summary of Paper E)

Subproblem addressed: Supervised brain tumor segmentation requires GT tumor boundary annotations. However, obtaining tumor boundary annotation from medical experts is a time consuming process. This study aims at seeking an alternative approach of segmentation when a small part of the dataset has GT tumor annotation, and the remaining large part of the dataset is without annotation.

Motivation: This study explores the possibility of replacing two bounding box (foreground (FG) and background (BG)) areas with GT tumor annotation, and uses large amount of a dataset without GT annotations along with small amount (<20) with annotation to perform segmentation.

Basic idea: This study is aimed at training a multi-stream deep network for tumor segmentation by using a large amount of ellipse tumor bounding boxes and a small amount of GT annotated MRIs. To investigate the possibility whether GT tumor annotations could be replaced by ellipse tumor bounding box areas without much drop in segmentation performance.

Contributions:

- Studying the feasibility of using 2D FG and BG ellipse box areas on MRIs for pre-training the segmentation network and refined-training on a small number of MRIs with annotated tumors (<20 MRI scans).

- Using multi-modality MRIs on a multi-stream U-Net, where features learned from all modalities are fused for segmentation.

- Comparing the performance using the same network trained entirely on MRIs with tumor annotations.

Proposed Method: The block diagram of the proposed method is shown in Figure 3.10.

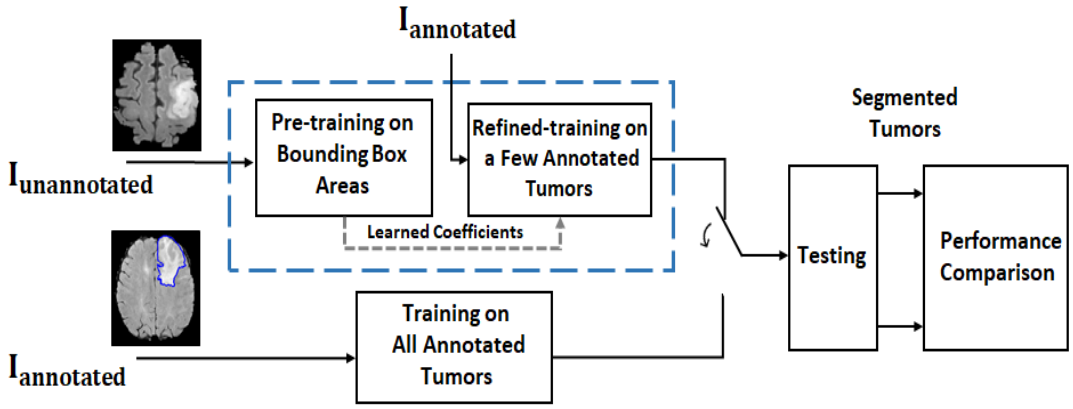

Figure 3.10: The block diagram for designed case studies for MR brain tumor segmentation.

In the blue dashed box, the network for glioma segmentation is trained by a large percentage of unannotated images using ellipse box areas and a small number of annotated images. The network performance is then compared with the same network trained on all annotated tumor images. The FG and BG tumor areas are used for pre-training the network, where FG is the interior area of the tumor region and BG is the exterior area of a relatively large ellipse containing normal brain tissues as shown in Figure 3.11. Segmentation is performed using a multi-stream U-Net, where 4 modalities of MRIs are used as the input to the four-stream of U-Net. The features learned at the ends of each streams are fused at a fusion layer and is further proceeded for predicting segmentation. For pre-training, a large part of the training dataset with ellipse box areas are used and for refined-training a small part (<20 patient MRIs) with tumor annotations are used.

Evaluation: The experiments are conducted on MICCAI'17. The confusion matrix from the test results are reported in Table 3.8(a).

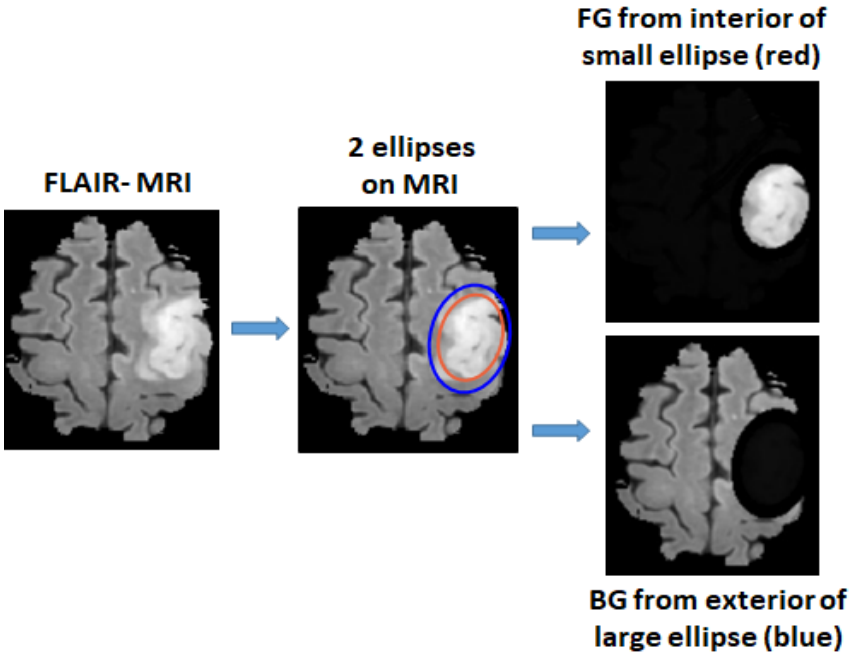

Figure 3.11: FG and BG tumor areas defined by two ellipse areas.

Table 3.8: Average test results (over 5 runs) on MICCAI dataset. (a) Confusion Matrix. (b) Performance evaluation.

(a) Confusion Matrix

Predicted\True	Tumor(| σ |)	Non-tumor(| σ |)
Tumor	83.88(0.08)	1.54(0.09)
Non-tumor	16.12(0.08)	98.46(0.09)

(b) Performance evaluation

Tumor Acc. %(| σ |)	Dice Score (| σ |)	Jaccard Index (| σ |)
83.88(0.08)	0.8407(0.0006)	0.7233(0.0028)

Performance is further evaluated on tumor accuracy, dice score and Jaccard index on tumor areas and included in Table 3.8(b). One can observe that the average dice score, the Jaccard index values show reasonably good performance on MICCAI.

Comparison: The performance of the proposed paradigm is compared with the performance of the same network but trained and tested on all GT annotated MRIs and is reported in Table 3.9. The aim of this comparison is to see how much the performance degrades and to see if it is feasible to use such a paradigm.

Table 3.9: Comparison of the test results on test accuracy and dice score.

Method	Tumor Acc.(%)	Dice Score
Proposed	83.88	0.8407
Conventional	92.66	0.9001
Degradation	-8.78	-0.0594

Conclusion: The comparison results have indicated that the proposed method is rather effective without significant degradation at the cost of tedious task of manual annotation.

CHAPTER 4

Conclusion

This thesis proposes several efficient DL networks for glioma grades and their molecular subtypes, as well as for tumor segmentation. The contribution of thesis are on developing several methods including; multi-stream CAE for classification; DCGAN for enlarging the training MRI dataset; CycleGAN for mapping datasets to a target domain; FL for individual dataset protection in a collaborative training; using bounding box strategy for MRIs without tumor boundary annotations. Experimental results on several datasets have shown that:

- For data augmentation, using DCGAN to enlarge the training dataset size has enhanced the classification performance.
- For classification, using multi-stream CAE network and fusion of features from multi-modality MRIs have improved the performance.
- For domain mapping, CycleGAN has successfully overcome domain shifts while retaining subtle molecular information. This has provided combining datasets effectively from multiple hospitals for classification.
- The proposed FL scheme has obtained good classification performance.

The performance degradation of FL compare to that of CL is small. This promising result provides the future direction to replace FL approach with CL.

- For classification, using bounding box tumor areas has shown feasibility with a slight degradation in performance as compare to that of using GT tumor boundary annotation. This method provides a trade-off between saving annotation time and slight performance degradation.
- For segmentation, training MRIs without tumor annotation using FG and BG ellipse tumor areas have shown to be effective. The method provides a trade-off between annotation cost and a slight performance degradation as compare to that of using all GT tumor boundary annotation.

4.1 Future Work

AI assisted diagnosis on brain tumor and their molecular subtypes is required for open research area. Our study shows that using datasets with pre-existing biopsy information, DL performs effectively for brain tumor diagnosis from MRIs. Current performance still requires more improvement to be clinically used. To improve the performance, more training dataset is required.

FL provides promising results for dataset protection, where each hospital participate in training without sharing their datatset with others. Compared to CL approach, it has shown a small performance degradation. This gives a good research direction for improving the DL performance that tackles large training dataset issue with protection of datasets from individual owners.

References

[1] "Magnetic resonance imaging (mri)," *https://www.nibib.nih.gov/science-education/science-topics/magnetic-resonance-imaging-mri*, 2023.

[2] K. Raza and N. K. Singh, "A tour of unsupervised deep learning for medical image analysis," *Current Medical Imaging*, vol. 17, no. 9, pp. 1059–1077, 2021.

[3] "Magnetic resonance imaging (mri)," *https://www.independentimaging.com/what-to-expect-from-an-mri-scan-of-the-head-and-brain/mri- scanner/*, 2023.

[4] D. C. Preston, "Magnetic resonance imaging (mri) of the brain and spine: Basics," *MRI Basics, Case Med*, vol. 30, 2006.

[5] C. E. Fuller and A. Perry, "Molecular diagnostics in central nervous system tumors," *Advances in anatomic pathology*, vol. 12, no. 4, pp. 180–194, 2005.

[6] D. N. Louis, A. Perry, G. Reifenberger, *et al.*, "The 2016 world health organization classification of tumors of the central nervous system: A summary," *Acta neuropathologica*, vol. 131, no. 6, pp. 803–820, 2016.

[7] C. Kruchko, Q. T. Ostrom, H. Gittleman, and J. S. Barnholtz-Sloan, "The cbtrus story: Providing accurate population-based statistics on brain and other central nervous system tumors for everyone," 2018.

[8] M. M. Wijnenga, S. R. van der Voort, P. J. French, *et al.*, "Differences in spatial distribution between who 2016 low-grade glioma molecular subgroups," *Neuro-oncology advances*, vol. 1, no. 1, vdz001, 2019.

[9] D. Delev, D. H. Heiland, P. Franco, *et al.*, "Surgical management of lower-grade glioma in the spotlight of the 2016 who classification system," *Journal of neuro-oncology*, vol. 141, no. 1, pp. 223–233, 2019.

[10] Y. Matsui, T. Maruyama, M. Nitta, *et al.*, "Prediction of lower-grade glioma molecular subtypes using deep learning," *Journal of neuro-oncology*, vol. 146, no. 2, pp. 321–327, 2020.

[11] D. N. Louis, A. Perry, G. Reifenberger, *et al.*, "The 2016 world health organization classification of tumors of the central nervous system: A summary," *Acta neuropathologica*, vol. 131, pp. 803–820, 2016.

[12] T. Komori, "Grading of adult diffuse gliomas according to the 2021 who classification of tumors of the central nervous system," *Laboratory Investigation*, vol. 102, no. 2, pp. 126–133, 2022.

[13] D. W. Parsons, S. Jones, X. Zhang, *et al.*, "An integrated genomic analysis of human glioblastoma multiforme," *science*, vol. 321, no. 5897, pp. 1807–1812, 2008.

[14] S. Nobusawa, T. Watanabe, P. Kleihues, and H. Ohgaki, "Idh1 mutations as molecular signature and predictive factor of secondary glioblastomas," *Clinical Cancer Research*, vol. 15, no. 19, pp. 6002–6007, 2009.

[15] C. Perez, "Brady's principles and practice of radiation oncology," *Edil. Lippincot Phil*, vol. 1, pp. 462–4631, 2008.

[16] "Diagnosis treatment," *https://www.mayoclinic.org/diseases-conditions/brain-tumor/diagnosis-treatment/drc-20350088*, 2023.

[17] P. T. Chandrasoma, M. M. Smith, and M. L. Apuzzo, "Stereotactic biopsy in the diagnosis of brain masses: Comparison of results of biopsy and resected surgical specimen," *Neurosurgery*, vol. 24, no. 2, pp. 160–165, 1989.

[18] J. Beiko, D. Suki, K. R. Hess, *et al.*, "Idh1 mutant malignant astrocytomas are more amenable to surgical resection and have a survival benefit associated with maximal surgical resection," *Neuro-oncology*, vol. 16, no. 1, pp. 81–91, 2014.

[19] P. Yang, W. Zhang, Y. Wang, *et al.*, "Idh mutation and mgmt promoter methylation in glioblastoma: Results of a prospective registry," *Oncotarget*, vol. 6, no. 38, p. 40 896, 2015.

[20] Y. Kang, S. H. Choi, Y.-J. Kim, *et al.*, "Gliomas: Histogram analysis of apparent diffusion coefficient maps with standard-or high-b-value diffusion-weighted mr imaging—correlation with tumor grade," *Radiology*, vol. 261, no. 3, pp. 882–890, 2011.

[21] H. Zhou, K. Chang, H. X. Bai, *et al.*, "Machine learning reveals multimodal mri patterns predictive of isocitrate dehydrogenase and 1p/19q status in diffuse low-and high-grade gliomas," *Journal of neuro-oncology*, vol. 142, pp. 299–307, 2019.

[22] Y. Han, Z. Xie, Y. Zang, *et al.*, "Non-invasive genotype prediction of chromosome 1p/19q co-deletion by development and validation of an mri-based radiomics signature in lower-grade gliomas," *Journal of neuro-oncology*, vol. 140, pp. 297–306, 2018.

[23] J. Yu, Z. Shi, Y. Lian, *et al.*, "Noninvasive idh1 mutation estimation based on a quantitative radiomics approach for grade ii glioma," *European radiology*, vol. 27, pp. 3509–3522, 2017.

[24] S. R. van der Voort, F. Incekara, M. M. Wijnenga, *et al.*, "Predicting the 1p/19q codeletion status of presumed low-grade glioma with an externally validated machine learning algorithm," *Clinical Cancer Research*, vol. 25, no. 24, pp. 7455–7462, 2019.

[25] X. Zhang, Q. Tian, L. Wang, *et al.*, "Radiomics strategy for molecular subtype stratification of lower-grade glioma: Detecting idh and tp53 mutations based on multimodal mri," *Journal of Magnetic Resonance Imaging*, vol. 48, no. 4, pp. 916–926, 2018.

[26] S. Liang, R. Zhang, D. Liang, *et al.*, "Multimodal 3d densenet for idh genotype prediction in gliomas," *Genes*, vol. 9, no. 8, p. 382, 2018.

[27] C. Ge, I. Y.-H. Gu, A. S. Jakola, and J. Yang, "Deep semi-supervised learning for brain tumor classification," *BMC Medical Imaging*, vol. 20, no. 1, pp. 1–11, 2020.

[28] S. R. van der Voort, F. Incekara, M. M. Wijnenga, *et al.*, "Combined molecular subtyping, grading, and segmentation of glioma using multi-task deep learning," *Neuro-oncology*, vol. 25, no. 2, pp. 279–289, 2023.

[29] K. Chang, H. X. Bai, H. Zhou, *et al.*, "Residual convolutional neural network for the determination of idh status in low-and high-grade gliomas from mr imagingneural network for determination of idh status in gliomas," *Clinical Cancer Research*, vol. 24, no. 5, pp. 1073–1081, 2018.

[30] Z. Li, Y. Wang, J. Yu, Y. Guo, and W. Cao, "Deep learning based radiomics (dlr) and its usage in noninvasive idh1 prediction for low grade glioma," *Scientific reports*, vol. 7, no. 1, pp. 1–11, 2017.

[31] C. Ge, I. Y.-H. Gu, A. S. Jakola, and J. Yang, "Enlarged training dataset by pairwise gans for molecular-based brain tumor classification," *IEEE access*, vol. 8, pp. 22 560–22 570, 2020.

[32] J. Cluceru, Y. Interian, J. J. Phillips, *et al.*, "Improving the noninvasive classification of glioma genetic subtype with deep learning and diffusion-weighted imaging," *Neuro-oncology*, vol. 24, no. 4, pp. 639–652, 2022.

[33] I. Goodfellow, J. Pouget-Abadie, M. Mirza, *et al.*, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014.

[34] L. Jendele, O. Skopek, A. S. Becker, and E. Konukoglu, "Adversarial augmentation for enhancing classification of mammography images," *arXiv preprint arXiv:1902.07762*, 2019.

[35] C. Qi, J. Chen, G. Xu, Z. Xu, T. Lukasiewicz, and Y. Liu, "Sag-gan: Semi-supervised attention-guided gans for data augmentation on medical images," *arXiv preprint arXiv:2011.07534*, 2020.

[36] M. Frid-Adar, I. Diamant, E. Klang, M. Amitai, J. Goldberger, and H. Greenspan, "Gan-based synthetic medical image augmentation for increased cnn performance in liver lesion classification," *Neurocomputing*, vol. 321, pp. 321–331, 2018.

[37] T. C. Mok and A. C. Chung, "Learning data augmentation for brain tumor segmentation with coarse-to-fine generative adversarial networks," in *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries: 4th International Workshop, BrainLes 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 16, 2018, Revised Selected Papers, Part I 4*, Springer, 2019, pp. 70–80.

[38] V. M. Bashyam, J. Doshi, G. Erus, *et al.*, "Medical image harmonization using deep learning based canonical mapping: Toward robust and generalizable learning in imaging," *arXiv preprint arXiv:2010.05355*, 2020.

[39] Y. Gao, Y. Liu, Y. Wang, Z. Shi, and J. Yu, "A universal intensity standardization method based on a many-to-one weak-paired cycle generative adversarial network for magnetic resonance images," *IEEE transactions on medical imaging*, vol. 38, no. 9, pp. 2059–2069, 2019.

[40] H. Guan, Y. Liu, E. Yang, P.-T. Yap, D. Shen, and M. Liu, "Multi-site mri harmonization via attention-guided deep domain adaptation for brain disorder identification," *Medical image analysis*, vol. 71, p. 102 076, 2021.

[41] L. Qu, Y. Zhang, S. Wang, P.-T. Yap, and D. Shen, "Synthesized 7t mri from 3t mri via deep learning in spatial and wavelet domains," *Medical image analysis*, vol. 62, p. 101 663, 2020.

[42] X. Li, Y. Gu, N. Dvornek, L. H. Staib, P. Ventola, and J. S. Duncan, "Multi-site fmri analysis using privacy-preserving federated learning and domain adaptation: Abide results," *Medical Image Analysis*, vol. 65, p. 101 765, 2020.

[43] Y. Huang, C. Bert, S. Fischer, *et al.*, "Continual learning for peer-to-peer federated learning: A study on automated brain metastasis identification," *arXiv preprint arXiv:2204.13591*, 2022.

[44] S. Nalawade, C. Ganesh, B. Wagner, *et al.*, "Federated learning for brain tumor segmentation using mri and transformers," in *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries: 7th International Workshop, BrainLes 2021, Held in Conjunction with MICCAI 2021, Virtual Event, September 27, 2021, Revised Selected Papers, Part II*, Springer, 2022, pp. 444–454.

[45] L. Yi, J. Zhang, R. Zhang, J. Shi, G. Wang, and X. Liu, "Su-net: An efficient encoder-decoder model of federated learning for brain tumor segmentation," in *Artificial Neural Networks and Machine Learning–ICANN 2020: 29th International Conference on Artificial Neural Networks, Bratislava, Slovakia, September 15–18, 2020, Proceedings, Part I*, Springer, 2020, pp. 761–773.

[46] P. Guo, P. Wang, J. Zhou, S. Jiang, and V. M. Patel, "Multi-institutional collaborations for improving deep learning-based magnetic resonance image reconstruction using federated learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 2423–2432.

[47] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, Springer, 2015, pp. 234–241.

[48] F. I. Diakogiannis, F. Waldner, P. Caccetta, and C. Wu, "Resunet-a: A deep learning framework for semantic segmentation of remotely sensed data," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 162, pp. 94–114, 2020.

[49] H. Huang, L. Lin, R. Tong, *et al.*, "Unet 3+: A full-scale connected unet for medical image segmentation," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2020, pp. 1055–1059.

[50] F. Wang, R. Jiang, L. Zheng, C. Meng, and B. Biswal, "3d u-net based brain tumor segmentation and survival days prediction," in *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries: 5th International Workshop, BrainLes 2019, Held in Conjunction with MICCAI 2019, Shenzhen, China, October 17, 2019, Revised Selected Papers, Part I*, Springer, 2020, pp. 131–141.

[51] S. Kim, M. Luna, P. Chikontwe, and S. H. Park, "Two-step u-nets for brain tumor segmentation and random forest with radiomics for survival time prediction," in *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries: 5th International Workshop, BrainLes 2019, Held in Conjunction with MICCAI 2019, Shenzhen, China, October 17, 2019, Revised Selected Papers, Part I 5*, Springer, 2020, pp. 200–209.

[52] W. Shi, E. Pang, Q. Wu, and F. Lin, "Brain tumor segmentation using dense channels 2d u-net and multiple feature extraction network," in *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries: 5th International Workshop, BrainLes 2019, Held in Conjunc-*

tion with MICCAI 2019, Shenzhen, China, October 17, 2019, Revised Selected Papers, Part I 5, Springer, 2020, pp. 273–283.

[53] Y. Sun and C. Wang, "A computation-efficient cnn system for high-quality brain tumor segmentation," *Biomedical Signal Processing and Control*, vol. 74, p. 103 475, 2022.

[54] S. Das, M. K. Swain, G. Nayak, and S. Saxena, "Brain tumor segmentation from 3d mri slices using cascading convolutional neural network," in *Advances in Electronics, Communication and Computing: Select Proceedings of ETAEERE 2020*, Springer, 2021, pp. 119–126.

[55] C. Shan, Q. Li, and C.-H. Wang, "Brain tumor segmentation using automatic 3d multi-channel feature selection convolutional neural network.," *Journal of Imaging Science & Technology*, vol. 66, no. 6, 2022.

[56] R. Ranjbarzadeh, A. Bagherian Kasgari, S. Jafarzadeh Ghoushchi, S. Anari, M. Naseri, and M. Bendechache, "Brain tumor segmentation based on deep learning and an attention mechanism using mri multi-modalities brain images," *Scientific Reports*, vol. 11, no. 1, pp. 1–17, 2021.

[57] R. Raina, A. Madhavan, and A. Y. Ng, "Large-scale deep unsupervised learning using graphics processors," in *Proceedings of the 26th annual international conference on machine learning*, 2009, pp. 873–880.

[58] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[59] M. B. Ali, *Deep Learning Methods for Classification of Glioma and Its Molecular Subtypes*. Chalmers Tekniska Hogskola (Sweden), 2021.

[60] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Icml*, 2010.

[61] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1026–1034.

[62] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[63] Y. LeCun, B. Boser, J. S. Denker, *et al.*, "Backpropagation applied to handwritten zip code recognition," *Neural computation*, vol. 1, no. 4, pp. 541–551, 1989.

[64] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016.

[65] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization.," *Journal of machine learning research*, vol. 12, no. 7, 2011.

[66] M. D. Zeiler, "Adadelta: An adaptive learning rate method," *arXiv preprint arXiv:1212.5701*, 2012.

[67] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[68] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International conference on machine learning*, PMLR, 2015, pp. 448–456.

[69] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[70] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, pp. 1097–1105, 2012.

[71] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," *CoRR*, vol. abs/1311.2901, 2013.

[72] C. Szegedy, W. Liu, Y. Jia, *et al.*, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[73] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[74] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[75] F. Iandola, M. Moskewicz, S. Karayev, R. Girshick, T. Darrell, and K. Keutzer, "Densenet: Implementing efficient convnet descriptor pyramids," *arXiv preprint arXiv:1404.1869*, 2014.

[76] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *science*, vol. 313, no. 5786, pp. 504–507, 2006.

[77] H. Abdi and L. J. Williams, "Principal component analysis," *Wiley interdisciplinary reviews: computational statistics*, vol. 2, no. 4, pp. 433–459, 2010.

[78] P. Baldi, "Autoencoders, unsupervised learning, and deep architectures," in *Proceedings of ICML workshop on unsupervised and transfer learning*, JMLR Workshop and Conference Proceedings, 2012, pp. 37–49.

[79] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[80] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne.," *Journal of machine learning research*, vol. 9, no. 11, 2008.

[81] J. Masci, U. Meier, D. Cireşan, and J. Schmidhuber, "Stacked convolutional auto-encoders for hierarchical feature extraction," in *International conference on artificial neural networks*, Springer, 2011, pp. 52–59.

[82] A. Ng *et al.*, "Sparse autoencoder," *CS294A Lecture notes*, vol. 72, no. 2011, pp. 1–19, 2011.

[83] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, P.-A. Manzagol, and L. Bottou, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion.," *Journal of machine learning research*, vol. 11, no. 12, 2010.

[84] S. Rifai, G. Mesnil, P. Vincent, *et al.*, "Higher order contractive auto-encoder," in *Joint European conference on machine learning and knowledge discovery in databases*, Springer, 2011, pp. 645–660.

[85] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[86] Y. Lei, J. Harms, T. Wang, *et al.*, "Mri-only based synthetic ct generation using dense cycle consistent generative adversarial networks," *Medical physics*, vol. 46, no. 8, pp. 3565–3581, 2019.

[87] H. U.S., "Department of health human services," *https://www.hhs.gov /hipaa/for-professionals/ index.html (accessed on 03 May 2023)*, 2023.

[88] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 10, no. 2, pp. 1–19, 2019.

[89] Z. Dai, B. K. H. Low, and P. Jaillet, "Federated bayesian optimization via thompson sampling," *Advances in Neural Information Processing Systems*, vol. 33, pp. 9687–9699, 2020.

[90] C. He, M. Annavaram, and S. Avestimehr, "Group knowledge transfer: Federated learning of large cnns at the edge," *Advances in Neural Information Processing Systems*, vol. 33, pp. 14 068–14 080, 2020.

[91] G. A. Kaissis, M. R. Makowski, D. Rückert, and R. F. Braren, "Secure, privacy-preserving and federated machine learning in medical imaging," *Nature Machine Intelligence*, vol. 2, no. 6, pp. 305–311, 2020.

[92] A. Hard, K. Rao, R. Mathews, *et al.*, "Federated learning for mobile keyboard prediction," *arXiv preprint arXiv:1811.03604*, 2018.

[93] N. Rieke, J. Hancox, W. Li, *et al.*, "The future of digital health with federated learning," *NPJ digital medicine*, vol. 3, no. 1, p. 119, 2020.

[94] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial intelligence and statistics*, PMLR, 2017, pp. 1273–1282.

[95] J. Wang, Q. Liu, H. Liang, G. Joshi, and H. V. Poor, "Tackling the objective inconsistency problem in heterogeneous federated optimization," *Advances in neural information processing systems*, vol. 33, pp. 7611–7623, 2020.

[96] D. A. E. Acar, Y. Zhao, R. M. Navarro, M. Mattina, P. N. Whatmough, and V. Saligrama, "Federated learning based on dynamic regularization," *arXiv preprint arXiv:2111.04263*, 2021.

[97] L. Wang, S. Xu, X. Wang, and Q. Zhu, "Addressing class imbalance in federated learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, 2021, pp. 10 165–10 173.

[98] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *Proceedings of Machine learning and systems*, vol. 2, pp. 429–450, 2020.

[99] K. Bonawitz, H. Eichner, W. Grieskamp, *et al.*, "Towards federated learning at scale: System design," *Proceedings of machine learning and systems*, vol. 1, pp. 374–388, 2019.

[100] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.

[101] M. Li, H. Tang, M. D. Chan, X. Zhou, and X. Qian, "Dc-al gan: Pseudoprogression and true tumor progression of glioblastoma multiform image classification based on dcgan and alexnet," *Medical physics*, vol. 47, no. 3, pp. 1139–1150, 2020.

[102] A. Madani, M. Moradi, A. Karargyris, and T. Syeda-Mahmood, "Semi-supervised learning with generative adversarial networks for chest x-ray classification with ability of data domain adaptation," in *2018 IEEE 15th International symposium on biomedical imaging (ISBI 2018)*, IEEE, 2018, pp. 1038–1042.

[103] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2223–2232.

[104] A. Odena, "Semi-supervised learning with generative adversarial networks," *arXiv preprint arXiv:1606.01583*, 2016.

[105] M. Fahim Sikder, "Bangla handwritten digit recognition and generation," in *Proceedings of International Joint Conference on Computational Intelligence: IJCCI 2018*, Springer, 2020, pp. 547–556.

[106] M. Liu and O. Tuzel, "Coupled generative adversarial networks," *CoRR*, vol. abs/1606.07536, 2016.

[107] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *CoRR*, vol. abs/1411.1784, 2014.

[108] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1125–1134.

[109] Y. Jiang, H. Chen, M. Loew, and H. Ko, "Covid-19 ct image synthesis with a conditional generative adversarial network," *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 2, pp. 441–452, 2020.

[110] L. Karacan, Z. Akata, A. Erdem, and E. Erdem, "Learning to generate images of outdoor scenes from attributes and semantic layouts," *arXiv preprint arXiv:1612.00215*, 2016.

[111] P. Sangkloy, J. Lu, C. Fang, F. Yu, and J. Hays, "Scribbler: Controlling deep image synthesis with sketch and color," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 5400–5409.

[112] A. Mobiny, P. A. Cicalese, S. Zare, *et al.*, "Radiologist-level covid-19 detection using ct scans with detail-oriented capsule networks," *arXiv preprint arXiv:2004.07407*, 2020.

[113] H. Zhang, T. Xu, H. Li, *et al.*, "Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 5907–5915.

[114] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, "Infogan: Interpretable representation learning by information maximizing generative adversarial nets," *Advances in neural information processing systems*, vol. 29, 2016.

[115] M. Arjovsky, S. Chintala, and L. Bottou, *Wasserstein gan*, 2017.

[116] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. Paul Smolley, "Least squares generative adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2794–2802.

[117] T. Wang, M. Liu, A. Tao, G. Liu, J. Kautz, and B. Catanzaro, "Few-shot video-to-video synthesis," *CoRR*, vol. abs/1910.12713, 2019.

[118] ——, "Few-shot video-to-video synthesis," *arXiv preprint arXiv:1910.12713*, 2019.

[119] C. Hu, Y. Ding, and Y. Li, "Image style transfer based on generative adversarial network," in *2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, IEEE, vol. 1, 2020, pp. 2098–2102.

[120] C. Ledig, L. Theis, F. Huszár, *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4681–4690.

[121] R. Ma, B. Zhang, and H. Hu, "Gaussian pyramid of conditional generative adversarial network for real-world noisy image denoising," *Neural Processing Letters*, vol. 51, pp. 2669–2684, 2020.

[122] L. Yu, W. Zhang, J. Wang, and Y. Yu, "Seqgan: Sequence generative adversarial nets with policy gradient," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 31, 2017.

[123] Z. Zhang, L. Yang, and Y. Zheng, "Translating and segmenting multimodal medical volumes with cycle-and shape-consistency generative adversarial network," in *Proceedings of the IEEE conference on computer vision and pattern Recognition*, 2018, pp. 9242–9251.

[124] J. Yang, Z. Zhao, H. Zhang, and Y. Shi, "Data augmentation for x-ray prohibited item images using generative adversarial networks," *IEEE Access*, vol. 7, pp. 28 894–28 902, 2019.

[125] Y. Ganin, E. Ustinova, H. Ajakan, *et al.*, "Domain-adversarial training of neural networks," *The journal of machine learning research*, vol. 17, no. 1, pp. 2096–2030, 2016.

[126] K. Ehsani, R. Mottaghi, and A. Farhadi, "Segan: Segmenting and generating the invisible," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 6144–6153.

[127] X. Zhang, W. Jian, Y. Chen, and S. Yang, "Deform-gan: An unsupervised learning model for deformable registration," *arXiv preprint arXiv:2002.11430*, 2020.

[128] J. Qiao, Q. Lai, Y. Li, T. Lan, C. Yu, and X. Wang, "A gan based multi-contrast modalities medical image registration approach," in *2020 IEEE International Conference on Image Processing (ICIP)*, IEEE, 2020, pp. 3000–3004.

[129] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," in *2016 fourth international conference on 3D vision (3DV)*, Ieee, 2016, pp. 565–571.

[130] H. Shen, R. Wang, J. Zhang, and S. J. McKenna, "Boundary-aware fully convolutional network for brain tumor segmentation," in *Medical Image Computing and Computer-Assisted Intervention- MICCAI 2017: 20th International Conference, Quebec City, QC, Canada, September 11-13, 2017, Proceedings, Part II 20*, Springer, 2017, pp. 433–441.

[131] S. Motamed and F. Khalvati, "Inception augmentation generative adversarial network," 2020.

[132] A. Waheed, M. Goyal, D. Gupta, A. Khanna, F. Al-Turjman, and P. R. Pinheiro, "Covidgan: Data augmentation using auxiliary classifier gan for improved covid-19 detection," *Ieee Access*, vol. 8, pp. 91 916–91 923, 2020.

[133] H. Tekchandani, S. Verma, and N. Londhe, "Performance improvement of mediastinal lymph node severity detection using gan and inception network," *Computer Methods and Programs in Biomedicine*, vol. 194, p. 105 478, 2020.

[134] H. Yang, J. Sun, A. Carass, *et al.*, "Unpaired brain mr-to-ct synthesis using a structure-constrained cyclegan," in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: 4th International Workshop, DLMIA 2018, and 8th International Workshop, ML-CDS 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 20, 2018, Proceedings 4*, Springer, 2018, pp. 174–182.

[135] K. Armanious, C. Jiang, M. Fischer, *et al.*, "Medgan: Medical image translation using gans," *Computerized medical imaging and graphics*, vol. 79, p. 101 684, 2020.

[136] D. Nie, R. Trullo, J. Lian, *et al.*, "Medical image synthesis with context-aware generative adversarial networks," in *Medical Image Computing and Computer Assisted Intervention- MICCAI 2017: 20th International Conference, Quebec City, QC, Canada, September 11-13, 2017, Proceedings, Part III 20*, Springer, 2017, pp. 417–425.

[137] Q. Yang, N. Li, Z. Zhao, X. Fan, E. I. Chang, Y. Xu, *et al.*, "Mri cross-modality image-to-image translation," *Scientific reports*, vol. 10, no. 1, pp. 1–18, 2020.

[138] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "Stargan: Unified generative adversarial networks for multi-domain image-to-image translation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8789–8797.

[139] J. Yoon, J. Jordon, and M. Schaar, "Radialgan: Leveraging multiple datasets to improve target-specific predictive models using generative adversarial networks," in *International Conference on Machine Learning*, PMLR, 2018, pp. 5699–5707.

[140] T. Falk, D. Mai, R. Bensch, *et al.*, "U-net: Deep learning for cell counting, detection, and morphometry," *Nature methods*, vol. 16, no. 1, pp. 67–70, 2019.

[141] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," in *2016 fourth international conference on 3D vision (3DV)*, Ieee, 2016, pp. 565–571.

[142] H. Chen, Q. Dou, L. Yu, J. Qin, and P.-A. Heng, "Voxresnet: Deep voxelwise residual networks for brain segmentation from 3d mr images," *NeuroImage*, vol. 170, pp. 446–455, 2018.

[143] F. Ye, J. Pu, J. Wang, Y. Li, and H. Zha, "Glioma grading based on 3d multimodal convolutional neural network and privileged learning," in *2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, IEEE, 2017, pp. 759–763.

[144] C. Ge, Q. Qu, I. Y.-H. Gu, and A. S. Jakola, "3d multi-scale convolutional networks for glioma grading using mr images," in *2018 25th IEEE International Conference on Image Processing (ICIP)*, IEEE, 2018, pp. 141–145.

[145] C. Ge, I. Y.-H. Gu, A. S. Jakola, and J. Yang, "Deep learning and multi-sensor fusion for glioma classification using multistream 2d convolutional networks," in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, IEEE, 2018, pp. 5894–5897.

[146] M. B. Ali, I. Y.-H. Gu, and A. S. Jakola, "Multi-stream convolutional autoencoder and 2d generative adversarial network for glioma classification," in *International Conference on Computer Analysis of Images and Patterns*, Springer, 2019, pp. 234–245.