Review

Survey on Recent Trends in Medical Image Classification Using Semi-Supervised Learning

Zahra Solatidehkordi * and Imran Zualkernan

Department of Computer Science and Engineering, American University of Sharjah, Sharjah 26666, United Arab Emirates
* Correspondence: g00059068@aus.edu

Abstract: Training machine learning and deep learning models for medical image classification is a challenging task due to a lack of large, high-quality labeled datasets. As the labeling of medical images requires considerable time and effort from medical experts, models need to be specifically designed to train on low amounts of labeled data. Therefore, an application of semi-supervised learning (SSL) methods provides one potential solution. SSL methods use a combination of a small number of labeled datasets with a much larger number of unlabeled datasets to achieve successful predictions by leveraging the information gained through unsupervised learning to improve the supervised model. This paper provides a comprehensive survey of the latest SSL methods proposed for medical image classification tasks.

Keywords: deep learning; semi-supervised learning; medical image analysis; classification

1. Introduction

Recent years have seen an increase in the digitalization of health records as well as a gradual increase in computational power, enabling the use of machine learning and deep learning algorithms in various areas of medicine such as diagnosis, outcome prediction, and clinical decision-making [1,2]. Medical images are a major component of patients’ electronic health records, making medical image analysis an active field of research for machine learning and deep learning [2]. Currently, the extraction of information from medical images is performed manually by radiologists and other professionals in the healthcare field, making the process dependent on the individual’s experience and knowledge, vulnerable to human error, and time-consuming. In order to address these issues, extensive research has been performed on the usage of machine learning and deep learning in medical image analysis to achieve more efficient and precise results [3]. The tasks studied in the context of medical image analysis include segmentation, detection, de-noising, reconstruction, and classification [4]. This survey paper focuses on recent studies in image classification only.

Image classification involves the assignment of one or more labels to an image based on the features extracted from it. The process consists of various steps, including preprocessing, feature extraction, feature selection, and classification. The extracted features can be low-level such as color, shape, intensity, texture, boundary, and position information, as well as high-level such as bag-of-words, scale-invariant feature transform, and fisher vector [5,6]. In traditional machine learning methods, features are generally extracted separately and then fed into the classification model. On the other hand, deep learning algorithms allow the feature extraction and classification steps to be combined in one network [5]. In addition, deep learning models apply end-to-end learning, in which the model receives a dataset of labeled images and extracts descriptive, hierarchical, and highly representative features with respect to each label which the model then uses for the classification task [5,7]. Deep learning methods can combine complex and low-level features and...
are less vulnerable to human errors [6]. Research has shown that deep learning models frequently outperform traditional machine learning algorithms in image classification tasks [5,8–10]. However, deep learning methods have their own limitations, such as the requirement for more time and higher computing power. Furthermore, traditional machine learning methods may achieve better results on smaller datasets [11–13].

Medical image classification plays an essential role in the analysis and diagnosis of diseases as well as clinical care and treatment. Medical images can be of different modalities based on the biomedical devices used to generate them. These modalities include Projectional Radiography (X-rays), Computed Tomography (CT), Magnetic Resonance Imaging (MRI), and Ultrasound Imaging [14]. X-ray is the most common medical imaging technique in which electromagnetic waves are used to create a two-dimensional representation of the body’s internal structure based on the varying rates of wave absorption in tissues with different densities. CT is a computerized procedure used to generate cross-sectional X-ray images, which can be combined to create a three-dimensional image, providing a more detailed representation of the region compared to conventional X-rays [15]. MRI generates three-dimensional anatomical images based on the behavior of the atomic nuclei in response to powerful magnetic fields and radiofrequency pulses [16]. Ultrasound imaging uses high-frequency sound waves to generate two-dimensional images of organs and tissues based on the reflection of the sound waves from different body structures [17]. In addition to these, dermoscopic images are also often used for medical image classification. These are two-dimensional images visualizing the subsurface skin structures. The benchmark datasets most used for research on medical image classification are the National Institutes of Health (NIH) Chest X-ray14 dataset and the International Skin Imaging Collaboration 2018 (ISIC2018) Skin Lesion Analysis dataset. The ChestX-ray14 dataset consists of 112,120 chest x-rays of 30,805 patients. 53.9% of the images are normal, and 46.1% are abnormal. The abnormal images are labeled with 14 different pathologies. Each image can have one or more labels, with 40.1% of abnormal images being multilabeled, making the dataset suitable for multi-label classification. The ISIC2018 dataset consists of 10,015 skin lesion dermoscopy images labeled with seven types of skin lesions. Each image belongs to one class, making the dataset suitable for multi-class classification.

One major issue facing medical image analysis is the lack of large datasets of high-quality labeled samples because manual labeling of images is a labor-intensive, expensive task [18]. However, obtaining unlabeled medical data from clinical sites is relatively easier [19]. Therefore, semi-supervised learning (SSL) is appropriate for medical image analysis because SSL uses a combination of both labeled and unlabeled data for learning. The goal of SSL is to use unsupervised learning with unlabeled data to improve the performance of the supervised model developed from labeled data [20]. This paper provides a survey of some of the latest SSL methods developed for the purpose of medical image classification.

This paper is organized as follows: Section 2 discusses commonly used SSL approaches, Section 3 explains the survey methodology, Section 4 presents the review of recent SSL approaches, Section 5 offers a summary of the results, and Section 6 discusses gaps and future work.

2. Background

This section provides background information about commonly used SSL approaches. Most current SSL methods rely on three assumptions. First, the smoothness assumption which states that if two points in a high-density region are close, their output labels must also be close. Second, the cluster assumption states that the points belonging to one class tend to form a cluster, meaning if certain points are in the same cluster, they likely belong to the same class. Lastly, the manifold assumption states that samples that are close in high-dimensional space are also close in a low-dimensional manifold [21,22]. The state-of-the-art (SOTA) semi-supervised methods for medical image classification can be divided into three broad categories of consistency-based, graph-based and adversarial [19,23]. Each of these is described next.
2.1. Consistency-Based Methods

In consistency-based methods, a model is encouraged to give consistent results on perturbed versions of the same inputs, such as input images with Gaussian noise added [21]. In other words, if an input $x_i$ is classified as class $c$, then the perturbed input $x_i'$ must also be classified as class $c$. As previous research has shown, an ensemble of neural networks can considerably improve the ability of the network to generalize [24-26]. One of the most dominant consistency-based SSL models is the Mean Teacher (MT) [27], which is based on the Temporal Ensembling [24] method. This method involves self-ensembling in neural networks, wherein the results of the same network over different epochs are combined during training to infer new training targets from the unlabeled data. Each training epoch has the same network but different regularization and input augmentation conditions to enforce consistency. Due to the use of dropout regularization, the prediction generated from each epoch is from an individual sub-network, making the final result an ensemble prediction of many different sub-networks. An exponential moving average (EMA) prediction is maintained for each training instance and updated after each training step with the new predictions. As a result, the EMA prediction is the ensemble result of the model generated after the latest training step and all the models generated in the previous steps. The predictions obtained through the temporal self-ensembling method are closer to the real labels of the unlabeled part of the dataset, and as such, these inferred labels can be used as training targets for unlabeled data. The Mean Teacher (MT) [27] builds on the temporal self-ensembling method. The model takes on the role of a student and a teacher; as a student, the model normally learns, while as a teacher, the model creates targets for learning which the student then uses to train. The teacher is an ensemble of the student model, similar to the previous method. This model uses the average weights of the models as opposed to the average of the predictions. Thus, the teacher maintains the EMA weights of the student model. At training step $t$, the EMA($\theta_t$) of successive $\theta$ weights is defined as:

$$\theta_t = \alpha \theta_{t-1} + (1 - \alpha) \theta_t$$

where $\alpha$ is a smoothing coefficient hyperparameter. Mean Teacher maintains a total cost function which is a weighted sum of the consistency cost and classification cost. The consistency cost is calculated by comparing the outputs of the teacher and student, while the classification cost involves comparing the student’s prediction and the target label. The Mean Teacher method allows the model to work with large datasets and improves classification accuracy.

2.2. Adversarial Methods

We will discuss two commonly used methods called Virtual Adversarial Training (VAT) and Generative Adversarial Network (GAN) methods in this category. Adversarial training methods involve the presence of “adversarial attacks,” which cause various errors in prediction [28]. These attacks are small perturbations to the input data applied in the direction where the model’s label assignment is most sensitive [29]. In VAT, adversarial attacks are generated and incorporated into the training of the model in order to increase its robustness. The aim of this method is to achieve high smoothness at each data point in order to prevent small perturbations from causing classification errors. The loss function used is the local distributional smoothness (LDS) loss, a measure of the robustness of the model against local perturbation around each input data point. The LDS is calculated as follows:

$$LDS(x, \theta) = D[p(y \mid x, \hat{\theta}), p(y \mid x, + r_{adv}, \theta)]$$

wherein $x$ is a labeled or unlabeled data point and $D[p, p']$ is a function that evaluates the divergence between the distributions $p$ and $p'$. $p(y \mid x, \hat{\theta})$ is an approximation of the true distribution of the output label calculated based on probabilistically generated
"virtual" labels. Thus, the goal is to minimize the divergence between the distribution of 
and its transformed version \( x_\epsilon + r_{adv} \), making the model more robust to adversarial
perturbations. The full objective function of VAT is as follows:

\[
\ell(D, \theta) + \alpha R_{adv}(D, D_{uu}, \theta)
\]

wherein \( \ell(D, \theta) \) is the negative log-likelihood of the labeled dataset, and \( \alpha \) is the regu-
larization coefficient which controls the relative balance between \( \ell(D, \theta) \) and \( R_{adv} \).
\( R_{adv} \) is the regularization term which is defined as the average of the loss \( LDS(x, \theta) \)
over all input data points.

In GAN-based methods, the adversary is a generator network whose role is to create
fake images that fit the distribution of the real data as closely as possible. This generator
competes against a discriminator network whose role is to distinguish the fake data from
the real data. As the two networks compete, they both gradually improve. The optimal
state of the model is reached when the generator has learned to create perfectly realistic
samples which the discriminator cannot distinguish from the real data [30]. This ability to
synthesize samples with high levels of realism can be used to solve the issue of scarcity of
labeled data. Additionally, the classifier training can also be embedded directly into the
GAN framework for improved generalization and robustness [4].

Generative Adversarial Nets (GANs) [31] are deep neural network architectures in-
volving two networks; a generator model (G) and a discriminator model (D). The name
“adversarial” comes from the fact that G and D are pitted against each other; G generates
samples while D learns to distinguish between the samples generated by G and samples
from the real data. The competition between the two networks leads to both networks
improving over time, with G generating more realistic images increasingly until they are
undistinguishable from the real samples.

Deep Convolutional Generative Adversarial Networks (DCGAN) [32] is a type of
GAN that uses Convolutional Neural Networks (CNN) in its generator in order to create
a higher quality generator model and achieve stable training across different datasets.
Some of the architecture guidelines of DCGAN include the replacement of pooling layers
with stride convolutions in D and fractional-strided convolutions in G, the usage of Batch
Normalization in both G and D, the elimination of fully connected hidden layers for
deeper architectures, the usage of the ReLU activation in G for all layers except the output
which uses the Tanh function, and lastly, usage of LeakyReLU activation in D for all lay-
ners.

Auxiliary Classifier Generative Adversarial Networks (ACGAN) [33] is a model
based on GAN that uses label conditioning in image synthesis, resulting in higher quality
samples with increased global coherence. In this method, every sample created by G has
a class label in addition to noise. In turn, D has to estimate both the likelihood of the image
being correct and the likelihood of the label is correct. The addition of label conditioning
in the model results in higher-quality generated images and more stabilized training.

2.3. Graph-Based Methods

In graph-based methods, graphs are constructed based on the data, where nodes rep-
resent samples (labeled or unlabeled), and edges represent neighborhood relations be-
tween the samples [22]. Generally, graph-based methods in SSL work by propagating la-
bles from labeled samples to unlabeled samples based on the information gained from the
graph [21]. Graph-based techniques are popular in recent works as they tend to be convex,
scalable, and effective [22]. Additionally, their strong mathematical properties allow for
fast computation [34]. A graph is represented by the notation \( G = (V, E) \) with \( V \) denot-
ing the nodes of the graph and \( E \) represents edges between two nodes. Each node is a sample from the data and each edge represents the relationship between two samples.
Graphs are called directed if the edges have directions and undirected if they do not.
Graphs can also be weighted in a weighted graph \( G = (V, E, W) \), \( w_{ij} \) represents the
weight of the edge between nodes i and j, and \( W \) is a matrix of size \( N \times N \) which contains
the weights of all edges. If \( w_{ij} = w_{ji} \), the \( W \) matrix is symmetric. If \( w_{ij} = 0 \), this signifies that the nodes \( i \) and \( j \) are not connected [22,35]. The degree of a vertex \( v_i \) is defined as \( d_i = \sum_{j=1}^{n} w_{ij} \). The degree matrix \( D \) is a diagonal matrix in which the values on the diagonal are \( d_1, \ldots, d_n \) [35].

The unnormalized Graph Laplacian matrix is defined as \( L = D - W \), while the normalized graph Laplacian matrix is defined as:

\[
L_{\text{sym}} = D^{-1/2}LD^{-1/2} = I - D^{-1/2}WD^{-1/2}
\]

where \( I \) is an identity matrix.

The Graph \( p \)-Laplacian is a nonlinear generalization of the graph Laplacian, which aims to improve the representation of the local structure information [36]. The unnormalized and normalized matrix forms of the graph \( p \)-Laplacian operator \( \Delta_p \) are as follows:

\[
\left( \Delta_p (u) \right)_i = \sum_{j \in V} M_{ij} \varphi_p (f_i - f_j)
\]

\[
\left( \Delta_p (n) \right)_i = \frac{1}{d_i^p} \sum_{j \in V} M_{ij} \varphi_p (f_i - f_j)
\]

where \( \varphi_p (x) = |x|^{p-1} \text{sign} (x) \), \( d_i \) is the degree function of the graph, \( M_{ij} \) is the weight of an edge, \( f \) is the eigenvector of the graph \( p \)-Laplacian operator, and \( p \geq 1 \) is a parameter set by the user.

2.4. Other Methods

In addition to the three main categories, we also review papers implementing methods that are less common in the recent literature. For example, one such method is pseudo-labeling, where a supervised learning model is trained on the labeled data and then used to classify a part of the unlabeled data. The pseudo-labels generated with high confidence are then added to the labeled data, and the model has trained again [37]. Traditional pseudo-labeling methods have several disadvantages, such as their tendency to be biased toward the majority class and their inability to be seamlessly adapted to multi-label and multi-class problems. Confidence-based pseudo-labeling leads to the selection of samples from the majority class as they tend to have higher confidence, resulting in a majority-biased classification model [37]. As such, pseudo-labeling methods are usually used in conjunction with consistency-based, adversarial, or graph-based methods in order to mitigate these disadvantages. In addition to pseudo-labeling, papers addressing Federated Learning, Learning by Association, Contrastive Learning, and MixMatch will also be reviewed.

Federated Learning (FL) is a decentralized system that allows training to be conducted on distributed data sources such as different hospitals separately. The parameters of the locally trained models can be used to update the main algorithm without exchanging private information, making FL an appropriate privacy-reserving method of data collaboration between different institutions. In semi-supervised FL, each of the local datasets can be labeled or unlabeled [38].

The Learning by Association (LA) [39] model assumes that good embeddings that belong to the same class will have high similarity. It builds association cycles through a “walk” from the embedding vector A obtained from a batch of labeled data and embedding vector B obtained from a batch of unlabeled data. The “walker” goes from A to B and then back to A based on mutual similarities. If walker ends up in the same class that it started from, the cycle is considered correct. Thus, the goal is to maximize the possibility of a correct association cycle. The “walker” loss is defined as:

\[
L_{\text{walker}} = H(T, P^{aba})
\]

with \( H \) representing the cross-entropy between the association cycle probabilities \( P^{aba} \) and the uniform target distribution of correct association cycles \( T \). Additionally, Haeusser et al. [39] argue that instead of making associations only among “easy” samples, it is beneficial to “visit” all unlabeled samples in order to make the best use of the dataset and
encourage embeddings that generalize better. Thus, the visit loss $L_{\text{visit}}$ is proposed, which is the cross-entropy between the visit probabilities $p^{\text{visit}}$ and the uniform target distribution $V$:

$$L_{\text{visit}} = H(V, p^{\text{visit}})$$  \hspace{1cm} (8)

Contrastive Learning (CL) [40] is a form of self-supervised learning which involves generating augmentations of datapoints to learn the higher-level features of a dataset from unlabeled samples. The augmentations of the same datapoint are considered positive samples and encouraged to be close to each other, while the augmentations of different datapoints are considered negative samples and encouraged to be far apart. The goal of this process is to create a model with higher generalization capability across different data distributions. Joint Contrastive Learning (JCL) [41] is a form of CL in which an infinite number of key-query pairs are generated from unlabeled data. As opposed to each key-query pair being compared independently as in the original CL method, multiple key-query pairs containing positive keys paired with the same shared query $q$ are encouraged to be close. This leads to more consistent representations in each instance-specific class.

MixMatch is a recent SSL method proposed by Berthelot et al. [42] that aims to unify three different loss terms, such as entropy minimization loss, which encourages the generation of labels with high confidence. The consistency regularization loss encourages output consistency in the presence of perturbations, and the generic regularization loss encourages better generalization. Consistency regularization is achieved by augmenting both labeled and unlabeled images with standard augmentation techniques (random horizontal flips and crops for images) and enforcing consistency in their results. Entropy minimization is achieved by sharpening the target distribution for unlabeled data. Lastly, traditional regularization is applied using a weight decay that penalizes the $L_2$ norm of the model parameters. Additionally, the Mixed Sample Data Augmentation (MSDA) method MixUp [43] is used for both regularization and semi-supervised learning. MSDA methods involve mixing different samples and their labels based on a particular policy to create an augmented dataset. The MixUp approach involves training the model on convex combinations of pairs of images and their labels to encourage linear behavior in-between training samples. Thus, the algorithm works as follows. A batch of labeled data $X$ and a batch of unlabeled data $U$ is fed to the model, and augmentations are applied to both. The label predictions of all augmentations of $U$ are computed and averaged, and sharpening is applied to the average prediction. The augmented labeled dataset $\hat{X}$ and the augmented unlabeled dataset with guessed labels $\hat{U}$ are combined and shuffled to obtain set $W$. Then, MixUp is applied to the labeled dataset $\hat{X}$ and entries from $W$ to obtain the processed labeled dataset $X'$; in the same way, MixUp is applied to the unlabeled dataset $\hat{U}$ and the rest of $W$ to obtain the processed unlabeled dataset $U'$. After the processed labeled and unlabeled (with guessed labels) batches $X'$ and $U'$ are obtained, they are used for semi-supervised training with the loss function $\mathcal{L}_X + \lambda_U \mathcal{L}_U$ which combines the supervised loss $\mathcal{L}_X$ and the unsupervised loss $\mathcal{L}_U$ with the hyperparameter $\lambda_U$. $\mathcal{L}_X$ is the cross-entropy loss obtained from training on $X'$ and $\mathcal{L}_U$ is the squared $L_2$ loss obtained from training on $U'$. FixMatch [44] is an SSL method similar to MixMatch that combines different existing SSL approaches, including consistency regularization and pseudo-labeling. FixMatch encourages label consistency between the pseudo-label of a weakly-augmented version of an input image and the pseudo-label of a strongly augmented version of the same image. CoMatch [45] is another SSL method that combines entropy minimization, consistency regularization, contrastive learning, and graph-based SSL. In this method, the training data is represented by a class probability and a low-dimensional embedding which evolve in a co-training framework. A smoothness constraint is applied to the class probabilities to improve pseudo-labels which are then used to train the classification head and projection head with a cross-entropy loss and graph-based contrastive loss respectively.
3. Survey Methodology

We used a variety of sources to search for journal articles and conference papers. These sources include the following databases and digital libraries: IEEE Xplore, ACM Digital Library, Springer, Elsevier, ScienceDirect, ResearchGate, Google Scholar, Scopus, MDPI online database, and PubMed. The distribution of papers read from different sources can be seen in Figure 1. The main keywords used for our searches were medical, image, and semi-supervised. Other searches included the addition of the terms “analysis” and “classification.” Additionally, for each category, the keywords related to that category were added to the search; consistency, adversarial, graph, GANs. Results were filtered for articles published between the years 2019 and 2022, and sorting was conducted based on relevance and number of citations when possible.

![Figure 1. Percentage of studies read from different sources.](image)

The exclusion and inclusion of research articles were decided first by a preliminary abstract analysis, then by a full review of the article. The research papers which focused only on medical image segmentation and did not contain a dedicated section on classification were excluded. The inclusion criteria for our research were as follows:

- The study must have a primary focus on semi-supervised learning.
- The study must include a clear report on the performance of the classification model.
- The study must present an in-depth description of the model architecture.

The exclusion criteria of our research were as follows:

- The study is not peer-reviewed or indexed in a reputable database.
- The study does not propose any significant addition or change to previously existing deep learning models.
- The study presents vague descriptions of the experimentation and classification results.

For the popular benchmark datasets X-ray14 and ISIC2018, the models which did not achieve results comparable to existing state-of-the-art models were excluded.

As the latest survey on this topic covered papers published in 2018 and prior [46], our paper presents research published between 2019 and 2022. Additionally, while [46] offered a general overview of unsupervised and semi-supervised methods in medical image analysis, this paper is specifically about semi-supervised learning for classification tasks and provides a more in-depth description of each of the models presented.
4. Survey Results

We have categorized the methods that were reviewed into five types of SSL approaches: consistency-based, adversarial, graph-based, other approaches, and hybrid methods. Figure 2 shows a diagram of the methods belonging to each category as well as the various datasets used for experimentation. The application of each method is discussed next.

Figure 2. Diagram of the types of SSL approaches and the datasets used to test each method.

4.1. Consistency-Based Methods

Relation-driven Self-Ensembling Model (SRC-MT) [19]: This method employs the Mean Teacher framework with a consistency-enforcing approach. In addition, this method explores the intrinsic relation between the images, which is generally overlooked in consistency-based methods, including Mean Teacher. Studying the relation between the images in an unsupervised fashion helps extract useful information from the unlabeled data [47,48]. The SRC-MT method proposes a novel Sample Relation Consistency (SRC) paradigm which enforces the consistency of the relation between two images after
perturbation; if two images have a similarity before perturbation, this relation should be preserved after they are perturbed. In other words, the relationship between input samples \( x_1 \) and \( x_2 \) must be the same as the relationship between perturbed samples \( x_1' \) and \( x_2' \). Thus, this method achieves consistency in both labeling and relations after perturbation. The total objective functions of the framework are as follows:

\[
L = L_s + \alpha L_u, \text{ with } L_u = L_c + \beta L_{src}
\]

where \( L_s \) is the supervised objective and \( L_u \) the unsupervised objective. \( L_u \) is made up of the standard consistency loss \( L_c \) and the relation consistency loss \( L_{src} \). \( \alpha \) is the trade-off weight between unsupervised and supervised loss, and \( \beta \) is a hyperparameter to balance \( L_c \) and \( L_{src} \).

SRC-MT was tested on two public benchmark medical image datasets; thorax disease classification with ChestX-ray14 for multilabel classification and ISIC2018: Skin Lesion Analysis for single label classification. Perturbations such as random rotation, horizontal and vertical flips, and translation were applied to the input images for consistency enforcing. For the skin lesion dataset, the images were all resized to 224 \( \times \) 224. The dataset was divided into 70% training, 10% validation, and 20% testing. DenseNet121 [23], pre-trained on ImageNet [49], was used as the network backbone. For the ChestX-ray14 dataset, the images were resized to 384 \( \times \) 384, and the same data split as the previous dataset was used. DenseNet169 [23], pre-trained on ImageNet, was used as the network backbone. For comparison, the upper bound and baseline performances were calculated; the upper bound was the performance of the fully supervised model trained on 100% labeled data, and the baseline was the performance of the fully supervised model trained on 20% of the data. Additionally, the model was compared to other semi-supervised learning methods, including the self-training-based method [50], SS-DCGAN [51], the Temporal Ensembling method, and the Mean Teacher. The training set used for the proposed method consisted of 20% labeled and 80% unlabeled data. The metrics used to gauge the performance of the model for single-label classification was AUC, sensitivity, specificity, accuracy, and F1 scores. For the single-label classification task on the skin lesion dataset, SRC-MT achieved the highest AUC, sensitivity, accuracy, and F1 scores compared to the baseline and other semi-supervised models. SRC-MT also achieved the second-highest specificity. Additionally, the AUC score of SRC-MT (93.58) was 3% higher than the baseline and 2% less than the upper bound and the F1 score (60.68) was 8% more than the baseline and 10% less than the upper bound. For the multi-labeled classification on the Chest-Xray14 dataset, the metric used was AUC, and for comparison the upper bound and baseline as well as the GraphX\textsuperscript{MT} method were used. The AUC scores were calculated for each of the 14 labels and averaged. The SRC-MT achieved a score of 79.23, approximately 0.4% higher than the Mean Teacher, 1.2% higher than GraphX\textsuperscript{MT}, 2% higher than the baseline, and 2% lower than the upper bound.

NoTeacher (NoT) [52]: In the Mean Teacher, the consistency target, which is the teacher, relies on the EMA of the student. In other words, the teacher’s weight is an ensemble of student weights. As such, the weights of the student and teacher are tightly coupled. This leads to the model being vulnerable to confirmation bias [53]; the teacher continues to reinforce what it already believes. NoTeacher is a consistency-based SSL framework that addressed this issue by employing two independent networks, thus removing the need for the time-averaging component. The NoTeacher framework works as follows: two random augmentations are applied to an input \( x \), which can be a 2D or 3D sample. These augmentations result in samples \( x_1 \) and \( x_2 \), which becomes the input of \( F_1 \) and \( F_2 \), two networks with similar architectures. The outputs are named \( f_1^s \) and \( f_2^s \) for labeled inputs and \( f_1^u \) and \( f_2^u \) for unlabeled inputs. Next, a loss function is calculated to enforce prediction consistency between \( F_1 \) and \( F_2 \). This loss function consists of a consistency loss and a supervised cross-entropy loss; if \( x_1 \) and \( x_2 \) are augmented versions of the same input \( x \), then \( f_1 \) resulting from \( x_1 \) and \( f_2 \) resulting from \( x_2 \) must be similar. Additionally, if \( x \) is a labeled input, then the outputs of both networks should match the
ground truth target $y$. Lastly, the total loss is backpropagated to update network parameters. The NoTeacher method is like the Mean Teacher as they both employ two networks with similar architectures. However, the NoTeacher method introduces two differences; first, it removes the EMA, making the networks entirely independent, and second, NoTeacher bases its loss function on a graphical model. The graphical model involves a graph consisting of $f_1$, $f_2$ and $y$ as the nodes, each connected to a consensus function $f_c \in [0,1]$, which enforces mutual agreement of the outputs on labeled and unlabeled data. The three datasets used for testing this model were Chest X-ray14, Radiological Society of North America (RSNA) Brain CT, and the Knee MRNet dataset containing knee MRI exams. The backbone networks used for each dataset were DenseNet121 [23], DenseNet169 [23], and MRNet [54], respectively, all pre-trained on ImageNet. All inputs were normalized based on ImageNet statistics before training. For all the datasets, the experiment was repeated six or seven times with different percentages of the dataset being labeled. The Chest X-ray14 dataset was used for multi-label classification. It was divided into 70% training, 10% validation, and 20% testing. The AUROC scores of the method were compared to the supervised baseline as well as other semi-supervised methods, including Mean Teacher, VAT, Pseudo-labeling, and the previously discussed SRC-MT. The NoTeacher method consistently achieved higher scores than the previous methods. It achieved a score of 82.10 at 50% labeled data, 1% less than the supervised model at 100% labeled data. The SRC-MT method attained lower scores than MT at all labeling budgets below 20%, but a higher score at 20%. The RSNA Brain CT dataset was used for multi labeled classification. It includes 19,520 CT brain scans from the RSNA 2019 challenge. Each slice in these scans is labeled with five binary labels corresponding to the absence or presence of five different forms of hemorrhage. 14.3% of the images include the presence of at least one form of hemorrhage, with 30.1% including multiple forms. 85.7% of the data are samples with no abnormalities. The dataset was divided into 60% training, 20% validation and 20% testing. NoTeacher achieved consistently higher scores than both the supervised baseline and MT. It also achieved overall higher scores than the other semi-supervised methods. On the other hand, MT provided lower scores than the supervised baseline at higher percentages of labeled data. Lastly, the Knee MRNet dataset consists of 1370 data points labeled with the presence or absence of abnormality, leading to a single labeled classification task. 80.6% of the data points contain abnormalities. 1130 training scans and 120 testing scans were used, with 20% of the training scans being randomly sampled for the validation set. Similar to the previous experiments, NoTeacher achieved higher scores than both the supervised baseline and the other semi-supervised methods. Additionally, at the labeled data percentage of 50%, NoTeacher managed to achieve a score higher than the supervised baseline at 100% labeled data.

Self-supervised Mean Teacher for Semi-supervised (S2MTS) [55]: S2MTS is a consistency-based SSL method for chest x-ray classification, which uses the Mean Teacher framework. This method consists of two learning stages. In the first stage, the student-teacher model is pre-trained in a self-supervised manner on the labeled and unlabeled data using JCL [41]. JCL involves learning an infinite number of key-query pairs generated from unlabeled data with the goal of enforcing dependencies between different key-query pairs containing the same query, which leads to more consistent representations in each instance-specific class [41]. Thus, each query $q_i$ paired simultaneously with multiple of its positive keys $k_{i,m}^+$ must return a low loss value. The loss of each $(q_i, k_{i,m}^+)$ pair is defined as follows:

$$L_{i,m} = \log \frac{\exp\left[\frac{1}{\tau} q_i^T k_{i,m}^+\right]}{\sum_{k_{i,j}^+} \exp\left[\frac{1}{\tau} q_i^T k_{i,j}^+\right]}$$

where $\tau$ is the temperature hyperparameter, $k_{i,m}^+$ is the $m^{th}$ positive key of $q_i$ and $k_{i,j}^-$ is the $j^{th}$ negative key of $q_i$. Thus, the total JCL loss can be calculated as follows:
\[ \ell_p(D^X, \theta_L, \theta_U^*) = -\frac{1}{|D^X|} \frac{1}{M} \sum_{i=1}^{|D^X|} \sum_{m=1}^M [\mathcal{L}_{IM}] \]  

where \( D^X \) is the set of labeled and unlabeled images, and \( M \) is the number of positive keys. In the second stage, the pre-trained student-teacher model is fine-tuned using the original Mean Teacher approach by maintaining an EMA following Equation (1). The datasets used for testing were the Chest X-ray14, ISIC2018 Skin Lesion Analysis, and CheXpert. The CheXpert dataset contains 220,000 images labeled with 14 diseases, wherein each image can have more than one label. The backbone model used for all datasets was DenseNet21 [23], with the two-layer multi-layer perceptron (MLP) projection head being replaced with a three-layer MLP. The Chest X-ray14 dataset was resized to \( 512 \times 512 \), and the ISIC2018 and CheXpert datasets were resized to \( 224 \times 244 \). The data augmentations used were random resize, random crop, random horizontal flipping, and random rotation. The Chest X-ray14 dataset was evaluated on different percentages of labeled data (2%, 5%, 10%, 15%, 20%), resulting in AUC scores of 74.69, 78.96, 79.90, 80.31, 81.06, and 82.51. For the CheXpert dataset, the model achieved AUC scores of 66.15, 67.85, 70.83, 71.37, and 71.58 for 100, 200, 300, 400, and 500 labeled samples, respectively. Lastly, the results obtained on the ISIC2018 dataset with 20% labeled samples were an AUC score of 94.71% and an F1 score of 62.67.

A summary of the methods discussed in this section is provided in Table 1. Three consistency-based methods were reviewed: SRC-MT, NoT, and S⁵MTS. For each model, the table contains the datasets used for testing as well as the best performance score obtained on each dataset.

<table>
<thead>
<tr>
<th>Method</th>
<th>Type</th>
<th>Dataset Used</th>
<th>Best Results</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>ISIC2018</td>
<td>ISIC2018: AUC 93.58, F1 60.68</td>
</tr>
<tr>
<td>NoT [52]</td>
<td>Consistency-based</td>
<td>Chest X-ray14, RSNA Brain CT, Knee MRNet</td>
<td>X-ray14: AUC 79.49</td>
</tr>
<tr>
<td></td>
<td></td>
<td>RSNA: AUC 93.31</td>
<td>MRNet: AUC 91.49</td>
</tr>
<tr>
<td>S⁵MTS [55]</td>
<td>Consistency-based</td>
<td>Chest X-ray14, ISIC2018, CheXpert</td>
<td>X-ray14: AUC 82.51</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ISIC2018, CheXpert</td>
<td>CheXpert: AUC 94.71, F1 62.67</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

4.2. Adversarial Methods

Uncertainty-Guided Virtual Adversarial Training With Batch Nuclear-Norm Optimization [56]: This method incorporates batch-nuclear norm (BNN) optimization [57] to prevent overfitting to the labeled data and to improve the diversity and discriminability of the model. As proposed by Cui et al. [57], the nuclear-norm \( \| P(\theta) \|_* \) of the \( m \times n \) prediction matrix \( P(\theta) \) is calculated as follows:

\[ \| P(\theta) \|_* = \sum_{i=1}^{\min(m,n)} \sigma_i(P(\theta)) \]  

where \( \sigma_i(P(\theta)) \) represents the \( i \)th largest singular value of \( P(\theta) \). As mentioned previously, the goal of incorporating BNN optimization is to improve generalization and avoid overfitting to the labeled data. This is achieved by minimizing the BNN of the labeled data and maximizing the BNN of the unlabeled data batch. Thus, the labeled BNN loss \( L_{IBNN} \) and unlabeled BNN loss \( L_{uBNN} \) are defined as follows:

\[ L_{IBNN} = \frac{1}{B_l} \| P_l(\theta) \|_* \]  

\[ L_{uBNN} = -\frac{1}{B_u} \| P_u(\theta) \|_* \]
B_l and B_u represent the size of the labeled and unlabeled datasets, while \( \| P_l(\theta) \| \) and \( \| P_u(\theta) \| \) represent the nuclear norm of the labeled and unlabeled prediction matrices, respectively. In addition to BNN, the model incorporates uncertainty guidance during VAT loss computation to filter out unlabeled samples close to the decision boundary. The uncertainty \( U^i \) is calculated for each unlabeled sample \( x^i_u \) in a batch, and predictions with high uncertainty are filtered out to ensure that only reliable targets are used for learning. \( U^i \) is calculated as follows:

\[
U^i = - \sum_{j=1}^{c} p_u^{i,j} \log (p_u^{i,j}), i \in 1 \ldots B_u
\]  

(15)

with \( p_u^{i,j} \) representing the predicted probability of \( x^i_u \) for the \( j \)th class and \( C \) denoting the number of classes. Thus the model is trained using the following losses: the cross-entropy loss of the supervised model \( (L_{cls}) \), the VAT loss calculated from labeled data \( (L_{vat}) \), the uncertainty-guided VAT loss computed from the unlabeled data \( (L_{uvat}) \), and the BNN losses \( L_{BNN} \) and \( L_{UBNN} \). The overall loss for the labeled data is the combination of all losses calculated over the labeled data:

\[
L_l = L_{cls} + \lambda_{vat} L_{vat} + \lambda_{BNN} L_{BNN}
\]  

(16)

Similarly, the unlabeled loss can be calculated as follows:

\[
L_u = \lambda_{vat} L_{uvat} + \lambda_{UBNN} L_{UBNN}
\]  

(17)

wherein \( \lambda_{vat} \), \( \lambda_{BNN} \) and \( \lambda_{UBNN} \) are weighting coefficients. The overall objective function is the addition of the supervised and unsupervised losses \( L_l + L_u \). The model was tested on the ISIC2018 and Chest X-ray14 datasets as well as an in-house collected hip x-ray dataset composed of 26,075 x-ray images labeled into four categories. The ISIC2018 and hip x-ray images were resized into 224 × 224 images. DenseNet121 [23], pre-trained on ImageNet, was used as the backbone network for both datasets. The X-ray14 dataset was resized into 256 × 256 images with DenseNet169 [23] as the backbone network. The model achieved an AUC score of 96.04 and an F1 score of 69.67 on the ISIC2018 dataset with 20% labeled data. On the Chest X-ray14 dataset, the method achieved AUC scores of 69.75, 74.50, 77.52, 79.49, and 80.69 on labeled percentages of 2%, 5%, 10%, 15%, and 20%. Lastly, the hip x-ray dataset generated an AUC score of 92.43 and an F1 score of 72.39 with 20% labeled data. Other state-of-the-art methods such as MT, SRC-MT, and VAT were also tested on this dataset achieving AUC scores of 92.00, 91.40, 91.66, and F1 scores of 70.52, 70.90, and 70.57, respectively.

Semi-supervised Adversarial Classification (SSAC) [58]: this GAN-based method involves a reconstruction network \( R \), and a supervised classification network \( C \). Learnable transition (\( T \)) layers are used to transfer the image representation ability learned by network \( R \) to \( C \). \( R \) is an adversarial autoencoder-based unsupervised network consisting of a generator \( G \) and a discriminator \( D \). \( G \) contains an encoder and a decoder. The encoder, which has an architecture similar to ResNet [59], takes \( 64 \times 64 \) patches, and the decoder generates reconstructed patches of the same size. \( D \) is a deep convolutional neural network containing four convolutional layers. \( C \) contains two parts, one with the same architecture as the encoder of \( R \), and one a fully connected layer with two neurons, separated by a global average pooling (GAP) layer. There is no parameter sharing between \( R \) and \( C \); the learnable \( T \) layers, with each layer consisting of a \( 1 \times 1 \) convolutional layer, transfers the feature maps obtained by \( R \) to the corresponding blocks in \( C \). In the experimentation, \( C \) was pre-trained on ImageNet, and \( R \) was pre-trained on both the labeled and unlabeled data. The loss function was defined as follows:

\[
L_{SSAC}(X_m) = \lambda_{l_m}(l_{mse}(G(X_m), X_m) - [l_{bc}(D(G(X_m)), 0) + l_{bc}(D(X_m), 1)]) + l_{bc}(C(X_m), Y_m)
\]  

(18)
where $X_m$ is the $m$th input sample and $\lambda_1$ is a weighting parameter. The three terms in the function represent the mean square reconstruction loss of $G$, the cross-entropy adversarial loss of $D$, and the supervised classification loss, respectively. Two datasets were used for experimentation. The first dataset was a database of lung nodules on CT-scans from the lung image database consortium (LIDC), and the image database resource initiative (IDRI) called LIDC-IDRI. The second dataset was the Tianchi Lung Nodule Detection dataset. The LIDC-IDRI dataset contains 1018 chest CT scans from which the nodules of diameter 3mm to 30 mm were identified and annotated. From the Tianchi Lung Nodule Detection dataset, 1000 patients’ data was used to extract 1227 unlabeled nodules. In total, the combination of the two datasets resulted in 644 malignant, 1301 benign, and 1839 unlabeled samples, making this a single-label classification task. For this task, the MK-SSAC model was constructed, which consists of nine knowledge-based collaborative SSAC models (KBC-SSAC) trained on the patches extracted from the lung nodules whose predictions were subsequently combined by two output neurons. The model was tested five independent times with 10-fold cross-validation. The metrics calculated to gauge the results were accuracy, sensitivity, specificity, and AUC. The performance of the model was compared to other SSL methods (MK-CatGAN, MK-AAE, MK-Ladder Network), hand-crafted feature-based methods (3D GLCM + SVM, MVF + SVM), and DCNN-based methods (Fuse-TSD, MV-KBC). The average scores achieved by the model were 92.53 accuracies, 84.94 sensitivity, 96.28 specificities, and 95.81 AUC. The model obtained the highest accuracy and AUC scores as well as the second-highest sensitivity and specificity among all the methods compared. Additionally, the MK-SSAC model was tested with different percentages of the labeled data (100%, 80%, 60%, 40%, and 20%). From 20% to 100%, the scores increased by about 2% for accuracy and 3% for AUC. In comparison, the supervised MK-C model was also tested with the same percentages of labeled data and showed an increase in 7% in accuracy and 5% in AUC. These results show that the MK-SSAC model has relative robustness to variation in the number of labeled data.

Retinal Image Synthesis and Semi-Supervised Learning for Glaucoma Assessment (SS-DCGAN) [51]: SS-DCGAN is a model created to synthesize retinal images and determine the presence or absence of glaucoma. This model is based on the DCGAN architecture with a modification in the last output layer of $D$: one neuron for synthesis and three neurons for training the glaucoma classifier. $D$ thus becomes a classifier that has to label each sample as either Normal, Glaucoma, or Synthetic. The loss function of the method was as follows:

$$L = L_{\text{supervised}} + L_{\text{unsupervised}}$$ (19)

where $L_{\text{supervised}}$ for K classes is the cross-entropy loss function:

$$L_{\text{supervised}} = -\mathbb{E}_{x,y \sim p_{\text{data}}(x,y)} \log (p_{\text{model}}(y \mid x, y < K + 1))$$ (20)

and $L_{\text{unsupervised}}$ is GAN’s two-player minimax game:

$$L_{\text{unsupervised}} = -\left\{\mathbb{E}_{x \sim p_{\text{data}}(x)} \log D(x) + \mathbb{E}_{z \sim p_{z}(z)} \log (1 - D(G(z)))\right\}$$ (21)

where $D(x)$ is the probability of $x$ being from the real data and $G(z)$ is the probability of $z$ being from the generator. The dataset used for experimentation was a combination of fourteen public databases with a total of 86,926 images. Samples with glaucoma or normal labels were divided into a 70% training set (669 glaucomas and 981 normal) and a 30% testing set (287 glaucomas and 420 normal). The labeled training set, as well as all unlabeled images (84,569 samples), were used to train the model. The metrics calculated were AUC, F1-score, specificity, and sensitivity. The final results were compared to three previous models: ResNet50 [59] as well as the CNNs proposed by Alghamdi et al. [60] and Chen et al. [61]. SS-DCGAN achieved the highest AUC (0.90) and F1-score (0.84), as well as the second-highest specificity and sensitivity scores among the models.
Bi-Modality Medical Image Synthesis Using Semi-Supervised Sequential Generative Adversarial Networks [62]: multi-modality imaging involves the simultaneous usage of two or more imaging modalities for one examination [63]. For example, single photon emission computed tomography (SPECT), magnetic resonance imaging (MRI), and positron emission tomography (PET) can be used to combine optical, magnetic, and radioactive reporters to detect abnormalities in the epileptic brain. Common examples of bi-modality images are PET-SPECT and PET-CT [63]. The model proposed by Yang et al. was intended to synthesize high-quality bi-modality medical images using GANs by creating two sequential generative networks, each corresponding to one modality. The first modality in the sequence is the one with the least complexity, measured automatically by a complexity measurer algorithm. The goal of this is to decrease the difficulty of generating the second, more complex modality by conditioning it on the first. The proposed generator network is trained via semi-supervised learning in order to generate realistic images with a large diversity. The supervised learning model learns the joint distribution of the different modalities, while the unsupervised learning model learns the marginal distribution of the different modalities via adversarial learning. The architecture of the generator is as follows: first, a real image of modality 1 is encoded into a low-dimensional latent vector, which is then mapped to a fake image of the same modality by a decoder. For the second image, an image-to-image translator is used to create a fake image of modality 2 with the help of information gained from the previously generated fake image of modality 1. In the supervised training, the original paired images are provided, and so for each pair of fake images generated by the generator, the corresponding true pair in the original data can be found. Thus, the loss function of the supervised training is the pixel-wise reconstruction loss:

\[
L_s = \mathbb{E}_{I^1, I^2 \sim p(I^1, I^2)} \left[ \| I^1 - \hat{I}^1 \| + \| I^2 - \hat{I}^2 \| \right]
\]

where \( \hat{I}^1 \) and \( \hat{I}^2 \) represent the fake images while \( I^1 \) and \( I^2 \) represent the real images and \( \| x - \hat{x} \| \) represents the average Manhattan-distance calculated pixel-wise between intensities of images \( x \) and \( \hat{x} \). In order to counter a severe overfitting issue present in the supervised learning model due to the small number of labeled images available, an unsupervised learning model is also deployed. In the unsupervised model, the generator is trained using unpaired images and noise vectors rather than encodings. The model aims to minimize the Wasserstein (\( W \)) distances between the fake and real images. As such, the loss function of the unsupervised generator is:

\[
L_{unsup} = W^1 + W^2
\]

where \( W^1 \) is the \( W \) distance between the real and fake images of modality 1 and \( W^2 \) is the \( W \) distance between the real and fake images of modality 2. The semi-supervised training is carried out as follows: in one iteration, the generator is trained in a semi-supervised manner with a set of paired training images. In the next iteration, the decoder and image translator are trained in an unsupervised manner using a set of unpaired images. The supervised and unsupervised training are alternated for 40,000 iterations. The usage of supervised learning allows for the creation of correctly paired images, while unsupervised training allows for higher realism and diversity. The images generated by this model were used as real training data in a single-label prostate cancer classification task in which each pair of images is classified as clinically significant (CS) or non-clinically significant (nonCS). The classifier was trained using 483 synthetic multimodal images of type ADC-T2w, and the testing set contained 50 real CS images and 50 real nonCS images. The evaluation metrics used were the Inception Score (IS) and Fréchet Inception Distance (FID) for quality assessment of the synthetic images, Mutual Information Distance (MID) to assess the correctness of the pairings between ADC and T2w synthetic images, and finally, the accuracy of the classification task. When compared with four state-of-the-arts GAN-based image synthesis methods (Costa et al. [64], CoGAN [65], CycleGAN [66], and pix2pix [67]),
the proposed model achieved better results in all metrics, with a classification accuracy of 93%.

A summary of the methods discussed in this section is provided in Table 2. Three adversarial methods have been reviewed: Uncertainty-guided VAT with BNN, SSAC, and Bi-Modality Image Synthesis. For each model, the table contains the specific adversarial approach (VAT-based or GAN-based), the datasets used for testing, as well as the best performance score obtained on each dataset.

Table 2. Comparison of adversarial methods.

<table>
<thead>
<tr>
<th>Method</th>
<th>Type</th>
<th>Dataset Used</th>
<th>Best Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSAC [58]</td>
<td>Adversarial (GAN-based)</td>
<td>LIDC-IDRI and Tianchi Lung Nodule dataset combined</td>
<td>AUC 92.82, Accuracy 90.58</td>
</tr>
<tr>
<td>Bi-Modality Image Synthesis [62]</td>
<td>Adversarial (GAN-based)</td>
<td>Locally collected dataset and PROSTATEx combined</td>
<td>Accuracy 93</td>
</tr>
</tbody>
</table>

4.3. Graph-Based Methods

GraphXNET: GraphXNET is a semi-supervised graph-based framework that performs a classification task with an extremely small number of labeled samples and a large number of unlabeled samples. The function used in this model is $\Delta_l(u) = |WD^{-1}u|$, based on the normalized graph p-Laplacian with $p=1$. The algorithm is as follows: for each class $k$, the model assumes that there exists a set of labeled nodes $I_k \subset \{1 \ldots l\}$. For each class $k$, a variable $u^k$ is chosen that has values in all nodes of the graph. Assume the total number of classes is $L$. For all unlabeled nodes $i > l$, the $L$ chosen variables are coupled with the following constraint: $\sum_{k=1}^{L} u^k_i = 0, \forall i > l$. Using a small positive number $\epsilon$, the following constraint is applied:

$\begin{cases} u^k_i \geq \epsilon & \text{if } i \in I_k \\ u^k_i \leq -\epsilon & \text{if } i \in I_k \text{ and } k' \neq k \end{cases}$

(24)

The goal of the model is then to minimize the normalized ratios $\sum_k \frac{\Delta_l(u^k)}{|u^k|}$. This model was evaluated using the ChestX-ray14 dataset. The results of the classification task were compared to two previously existing state-of-the-art deep learning methods for X-ray classification proposed by Wang et al. [68] and Yao et al. [69]. For both methods, 70% of the data was labeled, whereas, for the GraphXNET method, only 20% was labeled. The metric used for comparison was the average of the AUC score over all 14 classes. GraphXNET achieved the highest average score (0.78) despite using a much lower number of labeled samples. Additionally, when tested with three different, random partitions of data, GraphXNET proved to be more stable to variations in the data partition.

Graph-Embedded Random Forest [70]: This method makes the standard random forest algorithm more reliable when dealing with a low number of labeled samples. With the traditional method, insufficient training data leads to a limited depth, an inaccurate prediction model for leaf nodes, and a sub-optimal splitting strategy [71]. In the graph-based model, Gu et al. [70] improved the splitting strategy by replacing the information gain algorithm with a graph-embedded entropy. The goal is to utilize the local structure of unlabeled data to achieve higher reliability when using a small number of labeled data while maintaining the advantages of the random forest, such as low computational complexity.
burden and robustness towards overfitting. The loss function is a sum of the supervised loss and a graph Laplacian regularization term. First, a graph \( G = (V, E, W) \) is constructed using labeled and unlabeled data. The nodes represent the training samples and \( W \) is the symmetric weight matrix, calculated as follows:

\[
W_{ij} = \begin{cases} 
\exp \left( \frac{-\|x_i - x_j\|^2}{\sigma^2} \right) & \text{if } (x_i, x_j) \text{ are neighbors} \\
0 & \text{if } (x_i, x_j) \text{ are not neighbors}
\end{cases}
\]  

(25)

Based on the label information for unlabeled samples gained from the graph embedding, the new information gained is as follows:

\[
G_m(w, \tau, X_l, Y_l, X_u) = G_m(S) - \left( |S_l|G_m(S_l) + |S_u|G_m(S_u) \right) / |S|
\]

(26)

where \( S \) is the node, \( S_l \) is the left child node, \( S_u \) is the right child node, \( \tau \) is a threshold, \( X_l \) and \( Y_l \) represent labeled instances and the class labels, respectively, and \( X_u \) represents unlabeled instances. The medical image datasets used for the classification task were the Digital Retinal Images for Vessel Extraction (DRIVE) 2D retinal vessel dataset and the Big Neuron 3D neuronal dataset. For both datasets, 40,000 samples were randomly selected from the training and testing sets, respectively, each containing 20,000 positive and 20,000 negative samples. The results were compared with the standard random forest trained on labeled data only, as well as the RobustNode method [71] trained on both labeled and unlabeled data. The results showed considerable improvement over both the standard random forest and the RobustNode method. For the DRIVE dataset, training with a labeled dataset size of 400 achieved an accuracy of 79.42%. For the Big Neuron dataset, training with 1500 labeled data achieved an accuracy of 74.16%.

Colorectal Cancer Tissue Classification Using Semi-Supervised Hypergraph Convolutional Network [72]: Bakht et al. [72] propose a hypergraph-based approach for colorectal cancer (CRC) classification. Hypergraphs allow more complex relationships between nodes compared to standard graphs by allowing one edge to join multiple nodes. The images used for classification were CRC Whole Slide Images (WSIs), high-resolution images captured from a microscope slide representing tissue structures that can be used to identify malignancy. First, patches of size \( 224 \times 224 \) are extracted from the images, and a VGG-19 [73] model in feed-forward mode is used to extract the feature matrix \( X \) from the set of \( n \) patches. The hypergraph \( G(V, E, W) \) is then constructed from \( X \) by connecting each vertex with its \( k \) nearest neighbors. The hypergraph is represented by a vertex-edge probabilistic incidence matrix \( H \) of size \( n \times n \) as follows:

\[
h(n, e) = \begin{cases} 
\exp \left( \frac{-d}{p_{\text{max}} d_{\text{avg}}} \right), & \text{if } n \in e \\
0, & \text{if } n \notin e
\end{cases}
\]

(27)

where \( d \) is the Euclidean distance between the current node and the neighbor node, \( d_{\text{avg}} \) is the average Euclidean distance between the \( k \) neighbors, and \( p_{\text{max}} \) is the maximum probability. The degrees of each vertex \( v \in V \) and edge \( e \in E \) were calculated as follows:

\[
d(v) = \sum_{e \in v} h(v, e), d(e) = \sum_{v \in e} h(v, e)
\]

(28)

The combination of all node and edge degrees results in the diagonal matrices \( D_v \) and \( D_e \). In the classification step, \( X \) and \( H \) are fed to a hypergraph neural network (HGNN) consisting of three hidden convolutional layers and a SoftMax classification layer. Spectral graph convolution is used for representation learning as follows:

\[
X^{L+1} = \sigma \left( D_v^{-1/2} H W D_v^{-1} H^T D_v^{-1/2} X^L \theta \right)
\]

(29)
where $\sigma$ is the activation function, $\theta$ is a parameter to be learned during training, and $W$ is one’s diagonal matrix. Layer $L$ outputs $X^{L+1}$ which is received as an input by layer $L+1$. The model was tested on a dataset of CRC WSI containing 4995 patches of size $224 \times 224$ representing 7 different tissue types. 70% of the dataset was used for training and 30% for testing. The model achieved an F1 score of 0.94 for the seven-class classification task.

A summary of the methods discussed in this section is provided in Table 3. Three graph-based methods have been reviewed: GraphX$^{\text{NET}}$, Graph-Embedded RF, and CRC with HGNNet. For each model, the table contains the datasets used for testing as well as the best performance score obtained on each dataset.

### Table 3. Comparison of graph-based methods.

<table>
<thead>
<tr>
<th>Method</th>
<th>Type</th>
<th>Dataset Used</th>
<th>Best Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>GraphX$^{\text{NET}}$ [34]</td>
<td>Graph-based</td>
<td>Chest X-ray14</td>
<td>AUC 0.78</td>
</tr>
<tr>
<td>Graph-Embedded RF [70]</td>
<td>Graph-based</td>
<td>DRIVE and Big Neuron 3D neuronal</td>
<td>DRIVE: accuracy 79.42 Big Neuron: accuracy 74.16</td>
</tr>
<tr>
<td>CRC with HGNNet [72]</td>
<td>Graph-based</td>
<td>CRC WSI dataset</td>
<td>F1 0.94</td>
</tr>
</tbody>
</table>

#### 4.4. Other Methods

The following section provides reviews of methods that do not neatly fit into the previous three categories.

Anti-curriculum Pseudo-labeling for Semi-supervised Medical Image Classification (ACPL) [37]: Liu et al. [37] propose a pseudo-labeling based image classification method for the Chest X-Ray14 and ISIC2018 Skin Lesion Analysis datasets. The ACPL approach aims to address the disadvantages of traditional pseudo-labeling and develop a model capable of achieving state-of-the-art results on par with consistency-based methods. ACPL relies on the existence of a distribution shift between labeled and unlabeled data. The unlabeled samples chosen for pseudo-labeling are the ones located as far as possible from the labeled data’s distribution, making them more likely to belong to the minority class and, thus, more informative, resulting in a more balanced training process. The informativeness of a sample is determined using the cross-distribution sample informativeness (CDSI) measure, which computes the closeness of the unlabeled sample to the set of most informative labeled samples called the anchor set ($\mathcal{D}_a$). The CDSI was calculated as follows:

$$ h(f_\theta(x), \mathcal{D}_a) = \begin{cases} 1, & p_{\gamma}(\xi = \text{high} \mid x, \mathcal{D}_a) > \tau \\ 0, & \text{otherwise} \end{cases} $$

in which $\xi$ denotes the information content random variable (low, medium, high), $\gamma$ represents the parameters of the Gaussian Mixture Model (GMM) and $\tau = \max[p_{\gamma}(\xi = \text{low} \mid x, \mathcal{D}_a), p_{\gamma}(\xi = \text{medium} \mid x, \mathcal{D}_a)]$. Once the set of most informative unlabeled samples is selected, the informative mixup (IM) method is used for pseudo-labeling. This method mixes the labels obtained from the K-nearest neighbor (KNN) classification with the labels obtained from the model $p_\theta(\cdot)$, where $p_\theta(x) = \sigma(f_\theta(x))$ wherein $f_\theta(x)$ is the input image feature and $\sigma(\cdot)$ is the final activation function to calculate an output in $[0,1]$.$^{[9]}$. The IM method carries out the pseudo-labeling process by calculating a linear combination of the model prediction $p_\theta(x)$ and the KNN prediction is weighted by the density score. Following the pseudo-labeling stage, the Anchor Set Purification (ASP) algorithm is used to select the most informative pseudo-labeled samples to be added to the anchor set. The backbone model used for experimentation was DenseNet-121 [23] for both datasets. The images in the X-Ray14 dataset were resized to $512 \times 512$. The training was performed with batch size 16 and a learning rate of 0.05 for 20 epochs in the first step and then 50 epochs, where the anchor set was updated with new pseudo-labeled samples every 10 epochs.
The model achieved AUC scores of 74.69, 79.96, 79.90, 80.31, and 81.06 for labeled dataset sizes of 2%, 5%, 10%, 15%, and 20%, respectively. For the ISIC2018 dataset, images were resized to $224 \times 224$. The training was conducted with batch size 32 and a learning rate of 0.001 for 40 epochs in the warm-up stage, followed by 100 epochs, with the anchor set being updated with ASP every 20 epochs. The model achieved AUC and F1-scores of 94.36 and 62.23 at a labeled set size of 20%.

Federated Semi-supervised Medical Image Classification via Inter-client Relation Matching [74]: Liu et al. [74] propose an SSL method based on FL. Existing SSL methods cannot be used reliably for FL as they depend on the labeled dataset being accessible during training. However, in the case of FL, the local data at a particular location may be entirely unlabeled. As such, Liu et al. [74] build an interaction between the learning of labeled and unlabeled data by leveraging inherent disease relationships which are independent of a specific local dataset. This client-invariant disease relation information can be extracted from the labeled data and used to supervise the learning of unlabeled data to ensure it captures similar disease relationships. An uncertainty-based algorithm is used to filter out unreliable pseudo-labels. The backbone FL framework of this method follows the standard FL paradigm in which the central server sends the global parameters $\theta$ to every client $k$. The model is trained at each $k$ with a local objective function $L^k$ and the updated local parameters are sent back to the central server and aggregated to update the global model. In the proposed model, local parameters are aggregated using weights proportional to the size of their database following the federated averaging algorithm (FedAvg) method [75]. The local learning objectives use the cross-entropy loss at labeled clients and the consistency regularization mechanism at unlabeled clients with transformations such as rotation, translation, and flip. The inter-client relation matching (IRM) method is introduced for disease relation estimation to assist learning at unlabeled clients, in which the disease relation matrix $M^d$ is derived from the data’s pre-SoftMax features at each labeled client and averaged to obtain the matrix $M$ representing the general release relation. The model was tested on two datasets; ISIC2018 and the RSNA 2019 Brain CT dataset. From the RSNA dataset, 25,000 slices were sampled and divided into 70%, 10%, and 20% splits for training, validation, and testing. The ISIC2018 images were resized to $224 \times 224$ and split similarly. The FL setting was simulated by partitioning the training set into 10 local clients, of which 20% (two clients) were labeled. DenseNet121 was used as the backbone model. On the RSNA dataset, the model achieved an AUC score of 87.56 and F1-score of 59.86. On the ISIC2018 dataset, it achieved an AUC score of 92.46 and F1-score of 55.81.

Semi-HIC [76]: Su et al. [76] propose an SSL method called Semi-HIC for the classification of histopathological images, which are representations of biopsy samples captured by a microscope. This model used a patch-wise CNN to learn embeddings of labeled and unlabeled images, wherein three convolutional blocks are used to extract low-level detail information, and four cascaded inception blocks are used to create discriminative representations for patches. Semi-HIC uses a novel loss function obtained from combining an association cycle consistency (ACC) loss and a maximal conditional association (MCA) loss. These losses are intended to address the two major challenges in histopathological image classification, which are inter-class similarity and intra-class variation. They are based on the walker loss and visit loss proposed by Haeusser et al. [39] in their LA method. The original visit loss is vulnerable to the inter-class similarity in histopathological images, as a cancerous labeled embedding $E^l_k$ and unlabeled embedding $E^u_k$ with the underlying non-cancerous class may have high similarity. Su et al.’s MCA loss aims to address this by separately calculating the visit probability of non-cancerous embeddings $p^{vn}_k$ and the visit probability of cancerous embeddings $p^{vc}_k$. If the unlabeled embedding $E_k^u$ has an underlying cancerous class, it is expected to have a high $p^{vc}_k$ and low $p^{vn}_k$ and vice versa. Thus, the MCA loss $L_{MCA}$ is defined as the conditional entropy between $p^{vn}_k$ and $p^{vc}_k$.
where $M$ is the size of $E^n$. The goal is to maximize the difference between $p_k^{uc}$ and $p_k^{um}$ for the model to learn discriminative representations for unlabeled patches. The ACC loss, on the other hand, applies modifications to the walker loss in order to make the model robust to intra-class variation. This is achieved by penalizing association cycles that start and end at labeled embeddings belonging to the same image. The ACC loss was calculated as follows:

$$L_{ACC} = -\frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{N} \varphi(h_i, h_j) p_{ij}^{u} \log \left( p_{ij}^{u} \right)$$  \hspace{1cm} (32)$$

representing the cross-entropy between the association cycle probabilities $p_{ij}^{u}$ and uniform target distribution $p_T$. The penalty factor $\varphi(h_i, h_j)$ is 1 if the association cycle does not start and end at the same image ($h_i \neq h_j$) and 5 if it does. Finally, the total loss function of the model is:

$$L_{total} = \lambda L_{BCE} + \gamma L_{ACC} + \beta L_{MCA}$$  \hspace{1cm} (33)$$

wherein $L_{BCE}$ is the binary cross-entropy loss of the CNN model. $\lambda$, $\gamma$, and $\beta$ are the weights of each loss which are set to 1, 1, and 0.75, respectively. This model was evaluated on the Bioimaging2015 and the Grand Challenge on Breast Cancer Histology (BACH) datasets. The Bioimaging2015 dataset contains 249 training and 36 testing images of cancerous and non-cancerous histopathological images. The training set is split into 222 training and 27 validation images. The BACH dataset contains 400 cancerous and non-cancerous histopathological images split with the ratio 70:10:20 into the training, validation, and testing sets. After patch extraction, the size of the dataset is 8420, 1163, and 2369 patches for training, validation, and testing, respectively. 5-fold cross-validation was carried out on the BACH dataset, while the average of 5 runs was computed for the Bioimaging2015 dataset. For the BACH dataset, the model obtained error rates of 30.38, 24.87, and 15.46 for 20, 40, and 80 labeled images per class, respectively. For the Bioimaging2015 dataset, the model achieved error rates of 30.13, 28.39, and 25.23 for 20, 40, and 80 labeled images per class, respectively. The model proved to be more successful than state-of-the-art supervised models for histopathology image classification.

A summary of the methods discussed in this section is provided in Table 4. Three methods have been reviewed: ACPL, FL with IRM, and Semi-HIC. For each model, the table contains the type of approach to SSL, the datasets used for testing, as well as the best performance score obtained on each dataset.

<table>
<thead>
<tr>
<th>Method</th>
<th>Type</th>
<th>Dataset Used</th>
<th>Best Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>FL with IRM [74]</td>
<td>Federated learning</td>
<td>RSNA Brain CT and ISIC2018 Skin Lesion Analysis</td>
<td>RSNA: AUC 87.56, F1 59.86 ISIC2018: AUC 92.46, F1 55.81</td>
</tr>
<tr>
<td>Semi-HIC [76]</td>
<td>Learning by association</td>
<td>Bioimaging2015 and BACH</td>
<td>Bioimaging2015: error rate 25.23 BACH: error rate 15.46</td>
</tr>
</tbody>
</table>

4.5. Hybrid Methods

This section covers a few models which combine two or more of the previously reviewed approaches.
Local and Global Consistency Regularized Method [77]: This method is both consistency-based and graph-based. This method uses the Mean Teacher framework and also enforces the local and global consistency of the data [78]. Local consistency is the tendency of instances from the same class to be in the same area in the feature space, and global consistency is the tendency of instances from the same global structure to have the same label. This method integrates label propagation (LP) to enforce local and global consistency. LP is a semi-supervised learning algorithm that involves the propagation of labels from labeled samples to unlabeled samples based on their closeness, which is determined by the affinity matrix. For an unlabeled instance \( x \), the label is calculated by taking the weighted average of the labeled instances close to \( x \); then, the new label of \( x \) can be propagated to other neighboring unlabeled data. Next, a graph is constructed based on the labels created by the LP algorithm as well as the ground truth labels:

\[
A_{ij} = \begin{cases} 
1, & \text{if } y_i = y_j \\
0, & \text{otherwise} 
\end{cases}
\]  

(34)

in which \( y_i \) and \( y_j \) represent the labels of the data. Contrastive Siamese loss [79] is used in order to enforce local and global consistencies by pulling the instances from the same class closer and pushing the ones from different classes farther apart:

\[
L_s = \begin{cases} 
\| z_i - z_j \|^2, & \text{if } A_{ij} = 1 \\
\max(0, m - \| z_i - z_j \|^2), & \text{if } A_{ij} = 0
\end{cases}
\]  

(35)

where \( m \) is a hyperparameter and \( z \) is the feature vector extracted from the intermediate layers of the student network. The final loss function is as follows:

\[
L_{\text{total}} = L_{\text{mt}} + w(\tau) \left( \lambda_{g_1} \sum_{x_i, x_j \in X_l} L_{s_1} + \lambda_{g_2} \sum_{x_i, x_j \in X_u} L_{s_2} \right)
\]  

(36)

where the total loss is a weighted addition of the Mean Teacher loss (\( L_{\text{mt}} \)) and the graph-based loss (\( L_{s_1} \) and \( L_{s_2} \)). \( \lambda_{g_1} \) is the weight of the loss calculated on the labeled instances and \( \lambda_{g_2} \) is the weight of the loss calculated on both labeled and unlabeled instances. The loss on the unlabeled instances is not calculated separately as the labels predicted by the LP algorithm are noisy, and their inclusion may harm the performance of the method. Testing was conducted using two datasets, the Multi-organ Nucleus Segmentation (MoNuSeg) dataset containing 22,462 samples and the Ki-67 nucleus dataset containing 17,516 samples. In both datasets, each sample is labeled with one of four types, leading to a single labeled classification task. For both datasets, 80% of the data was used for training and 20% for testing. From the training data, 20% was used for validation. The model architecture is a 13-layer convolutional neural network similar to the one proposed in the original Mean Teacher method [27]. The metric used to gauge the method’s performance is F1. Testing was conducted at different percentages of labeled data (5, 10, 25, 50, 100 for MoNuSeg and 1, 5, 10, 100 for Ki-67). For the MoNuSeg dataset, the proposed method achieved higher scores than both the supervised baseline and the Mean Teacher for all percentages, notably in the 5% and 10% percentages, where it obtained scores approximately 2% higher than the Mean Teacher. The highest score obtained was 76.89 for 50% of labeled data, and the lowest was 75.02 for 5% of labeled data. In the case of the Ki-67 dataset, the proposed method also achieved a higher score than the supervised baseline and Mean Teacher, the highest increase from Mean Teacher being approximately 2.5% at the 5% labeled percentage. The highest score was 79.79 for 10% of labeled data, and the lowest was 74.9 for 1% of labeled data.

Deep virtual adversarial self-training with consistency regularization [80]: This model is both adversarial and consistency-based. It consists of three parts: self-training, consistency regularization, and virtual adversarial training. As previously mentioned, self-training involves using the model itself to generate labels for unlabeled samples, which will then become part of the labeled training set in the next iteration. The model
outputs a probability distribution over all classes, and the generated label is only kept when the highest probability is above a predetermined threshold. Consistency enforcement is applied to both labeled and unlabeled samples. For the labeled samples, weak augmentation is applied, and the generated label is encouraged to be consistent with the ground truth label. For unlabeled samples, weak augmentation is applied, then a pseudo-label is generated by the self-training model; the same input image is then strongly augmented, and the new prediction must be consistent with the previously obtained pseudo-label. Lastly, virtual adversarial training is applied to improve the robustness of the model and strengthen its generalization capability. Thus, the loss function of the model becomes the weighted sum of three losses: the supervised cross-entropy loss $\ell_s$ for labeled data, the regularization loss $\ell_r$ for unlabeled data and the virtual adversarial training loss $\ell_{vat}$ applied to both labeled and unlabeled data.

$$\mathcal{L} = \ell_s + \alpha \cdot \ell_r + \beta \cdot \ell_{vat}$$  \hspace{1cm} (37)

where $\alpha$ and $\beta$ are the weighting coefficients. Experimentation was performed using two datasets; a breast ultrasound (BRUS) dataset and an optical coherence tomography (OCT) dataset. The BRUS dataset contains 39,904 images, 22,026 of which are labeled as malignant (14,557) or benign (7469). The other 17,878 images are unlabeled. The labeled data were randomly split into 34% training (7662), 34% validation (7579), and 32% testing (6785). The OCT dataset contains 109,309 images, each labeled with four different types (three diseases and one normal). There were 20,000 images taken for the validation and testing datasets, respectively, and 1% of the remaining data (691 images) was taken for the labeled dataset, whereas the rest was considered unlabeled. The network architecture used for this experimentation is DenseNet-121. The weak augmentations include horizontal and/or vertical flips with a 50% probability, as well as randomly applied horizontal and/or vertical translations by less than 20% of the image width or height. Strong augmentation includes transformations such as color channel reduction, contrast maximization, solarization, rotation, and posterization. The results are compared with Mean Teacher, VAT, MixMatch [42], and FixMatch [44], as well as the fully supervised DenseNet-121 as the baseline. The metrics calculated were the accuracy, F1, AUC, and Kappa scores. The proposed method achieved higher scores than all the compared methods on both datasets, with an F1-score of 0.88 on the BRUS dataset and a Macro F1-score of 0.91 on the OCT dataset.

Pseudo-labeling generative adversarial networks (PLGAN) [81]: This model proposed by Mao et al. [81] is based on pseudo-labeling and GANs. It additionally incorporates CL and MixMatch. The training process of this method is comprised of 4 steps; (1) pretraining, (2) image generation, (3) finetuning, and (4) pseudo-labeling. In step 1, the feature layer of ResNet50 [59] is pre-trained using CL to extract key image features. In step 2, GANs are used to generate images that simulate the real distribution of the labeled images with random Gaussian noise. In step 3, the cross-entropy loss is used to classify the images created by the generator in order to finetune the discriminator for classification. The model contains two classifiers, a global and a local classifier, both of which use two convolutional blocks to extract feature information. In step 4, the MixMatch approach is used for pseudo-labeling. The trained generator is used to create additional unlabeled samples in order to expand the unlabeled dataset, and these generated images are combined with the original dataset. In the same fashion, the pseudo-labels created for both the original and generated samples are combined to create the complete set of pseudo-labels. The overall loss function of the model contains four loss functions for each of the 4 steps. The loss function for the first step is the infoNCE [82] loss for CL and the reconstruction loss for intermediate vectors to avoid CL pattern collapse. For the second step, the loss function is the least squares loss. The loss of the semi-supervised finetuning step is the addition of the supervised cross-entropy loss and the unsupervised loss. Lastly, the fourth step uses the loss function of MixMatch. The model was tested on a dataset of optical coherence tomography (OCT) images for the classification of retinal degeneration, with
each sample belonging to one of four categories (three disease labels and one normal label). Additionally, it was tested on a COVID-19 dataset, a brain tumor MRI dataset, and a chest x-ray dataset. For the OCT and chest x-ray datasets, 100 labeled and 1000 unlabeled images were used. In the MRI dataset, 100 labeled images and 480 unlabeled images were selected. Lastly, 200 labeled and 200 unlabeled images were used for the COVID-19 dataset. The results achieved by the model were classification accuracies of 87.06, 89.50%, 80.50%, and 96.80% for the OCT, COVID-19, MRI, and x-ray datasets, respectively. In addition to PLGAN, the authors developed PLGAN+, which uses Comatch [45] for pseudo-labeling. This model achieved accuracy scores of 97.10%, 89.30%, 86.80%, and 97.50% for the OCT, COVID-19, MRI, and x-ray datasets, respectively. These results were compared to the results gained by several other state-of-the-art SSL models, including Comatch, MixMatch, FixMatch, and VAT. Overall, PLGAN+ achieved more favorable results than all other models.

Semi-supervised medical image classification based on CamMix [83]: Guo et al. [83] proposed an SSL model for medical image classification similar to MixMatch, which combines several SSL approaches. The consistency-based approach is used on unlabeled data to generate pseudo-labels that are robust to various augmentations. The authors argue that the MixUp method, which mixes samples by linearly interpolating the input samples and labels, results in unnatural mixed samples. Thus, they proposed CamMix, a novel MSDA method that mixes pairs of input samples and labels based on the class activation mask generated from the predictions of both labeled and unlabeled samples. As in MixMatch, entropy minimization is achieved by sharpening the target distribution for unlabeled data. The class activation map is obtained as follows for each batch $b$ at each epoch:

$$\text{GradMaxCam}^b = \max \left( \text{ReLU} \left( \sum_k w_k^b A^k \right) \right)$$

(38)

where $w_k^b$ is the weight of feature map $k$ for batch $b$ and $A$. $w_k^b$ is calculated as follows:

$$w_k^b = \frac{1}{Z} \sum_i \sum_j \frac{\partial Y^b}{\partial A_{ij}}$$

(39)

where $A_{ij}^k$ is the pixel value at location $(i, j)$ of $k$, $Z$ represents the number of pixels in $k$ and $Y^b$ denotes the maximum prediction score of batch $b$ generated by the classification model. Random threshold $\lambda \in [0,1]$ is applied to the gray-level $\text{GradMaxCam}^b$ to obtain the binary mask CamMask. The pixels higher than $1-\lambda$ are set to 1, and all others are set to 0. The CamMix algorithm takes a batch of labeled and unlabeled data and their corresponding predictions, including weak and strong augmentations of the data, and generates a mixed batch of original samples and shuffled samples, whose corresponding labels are mixed based on the number of pixels in CamMask. Thus, considering the original samples $input_1$ and the shuffled samples $input_2$, the CamMask is obtained by calculating $\text{GradMaxCam} \left( input_1 , 1 - \lambda \right)$, and the parameter $\text{lam}$ is calculated based on the number of pixels in the CamMask as follows:

$$\text{lam} = \text{sum(CamMask == 1)} / (\text{CamMask.size(0) * CamMask.size(1)})$$

(40)

The mixed batch $\text{mixed_input}$ is obtained from $input_1$ and $input_2$ by calculating the following:

$$\text{mixed_input} = input_1 \ast \text{CamMask} + input_2 \ast (1 - \text{CamMask})$$

(41)

Lastly, the overall loss of the model is defined as:

$$l = \text{criterion(logits, target_1) \ast lam + criterion(logits, target_2) \ast (1 - lam)}$$

(42)
wherein $\text{logits} = \text{model}(\text{mixed_input})$. The model was tested with a ResNet-18 on two datasets: The Interstitial Lung Disease (ILD) dataset and the ISIC2018 dataset. Instead of the F1-score, the metric used for evaluation was $F_{avg}$, the average F1-score over the different classes. The ILD dataset contains 109 CT scans with an average of 25 slices per case for a total of 2795 slices. All regions of interest were cropped into patches of size $32 \times 32$, resulting in approximately 7000 labeled patches and 19,000 unlabeled patches. Six typical lung patterns were used for the classification task. By using CamMix, the model achieved an $F_{avg}$ score of 95.34, higher than previous state-of-the-art MSDA methods for the ILD dataset, such as MixUp, CutMix [84], and Fmix [85]. For the ISIC2018 dataset, the model was trained with 20% labeled and 80% unlabeled data, obtaining an AUC score of 94.04 and $F_{avg}$ of 78.15. These scores were higher than the scores obtained by the previously mentioned MSDA methods. The SRC-MT model was also used for comparison and achieved a $F_{avg}$ score of 78.15, higher than the one obtained by CamMix. On the other hand, it obtained a lower AUC score.

A summary of methods discussed in this section is provided in Table 5. Three hybrid methods have been reviewed: Local and Global Consistency Regularized, PLGAN, SSL with CamMix, and Wan et al.’s method. We have reported the SSL methods which have been combined to create each hybrid method, as well as the datasets used for testing each model and the best performance score obtained on each dataset.

Table 5. Comparison of hybrid methods.

<table>
<thead>
<tr>
<th>Method</th>
<th>Type</th>
<th>Dataset Used</th>
<th>Best Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local and Global Consistency Regularized [77]</td>
<td>Consistency-based + Graph-based</td>
<td>MoNuSeg and Ki-67</td>
<td>MoNuSeg: F1 76.89</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Ki-67: F1 79.79</td>
</tr>
<tr>
<td>Wang et al. [80]</td>
<td>Consistency-based + Adversarial (VAT)</td>
<td>Breast Ultrasound (BRUS) and OCT</td>
<td>BRUS: F1 0.88</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>OCT: Macro F1 0.91</td>
</tr>
</tbody>
</table>
| PLGAN [81]                    | Pseudo-labeling + Adversarial (GANs) | OCT, COVID-19, Brain Tumor MRI, Chest X-Ray | Accuracies:  
|                               |                           |                             | OCT: 97.10             |
|                               |                           |                             | COVID-19: 89.30           |
|                               |                           |                             | MRI: 86.80               |
|                               |                           |                             | X-Ray: 97.50             |
| SSL with CamMix [83]          | Consistency-based + CamMix (MSDA) | ILD dataset, ISIC2018        | ILD: AUC 95.34            |
|                               |                           |                             | ISIC2018: AUC 94.04      |

5. Summary of Results

Table 6 shows a comparison of the models which were tested on the Chest X-Ray14 dataset for multi-label classification and/or the ISIC2018 for multi-class classification. For both datasets, all models used 20% of the data as the labeled dataset and 80% as the unlabeled dataset. Among them, the highest score on the Chest X-ray14 dataset was achieved by the $S^2\text{MTS}^2$ method (AUC of 82.51), followed by ACPL (AUC of 81.77). On the ISIC2018 dataset, the highest score was achieved by Uncertainty-guided VAT with BNN with an F1 score of 69.67 and AUC of 96.04, considerably higher than all other models. It is worth noting that ACPL obtained relatively high scores on both datasets, showing that pseudolabeling can indeed achieve state-of-the-art results by improving the measure used for the selection of informative pseudo-labels.
Table 6. Comparison of methods tested on the Chest X-ray14 and ISIC2018 datasets, at 20% labeled data for all models.

<table>
<thead>
<tr>
<th>Model</th>
<th>Type</th>
<th>Chest X-ray14</th>
<th>ISIC2018</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>AUC</td>
<td>AUC</td>
<td>F1</td>
</tr>
<tr>
<td>SRC-MT [19]</td>
<td>Consistency-based</td>
<td>79.23</td>
<td>93.58</td>
</tr>
<tr>
<td>NoT [52]</td>
<td>Consistency-based</td>
<td>79.49</td>
<td>_</td>
</tr>
<tr>
<td>S\textsuperscript{2}MTS\textsuperscript{2} [55]</td>
<td>Consistency-based</td>
<td>82.51</td>
<td>94.71</td>
</tr>
<tr>
<td>Uncertainty-guided VAT with BNN [56]</td>
<td>Adversarial (VAT-based)</td>
<td>80.69</td>
<td>96.04</td>
</tr>
<tr>
<td>Graph\textsuperscript{X}NET [34]</td>
<td>Graph-based</td>
<td>78.00</td>
<td>_</td>
</tr>
<tr>
<td>ACPL [37]</td>
<td>Pseudo-labeling</td>
<td>81.77</td>
<td>94.36</td>
</tr>
<tr>
<td>FL with IRM [74]</td>
<td>Federated learning</td>
<td>_</td>
<td>92.46</td>
</tr>
<tr>
<td>SSL with CamMix [83]</td>
<td>Consistency-based + CamMix (MSDA)</td>
<td>_</td>
<td>94.04</td>
</tr>
</tbody>
</table>

Figure 3 showcases the results obtained by different methods for each dataset. Among the GAN-based methods, the bi-modality image synthesis model is notable as it allows a more comprehensive look at a certain medical problem by combining different types of scans of the same area, leading to a more successful diagnosis. Despite the usefulness of bi-modality and multi-modality medical image diagnosis, labeled data in this category is even more scarce, making the semi-supervised image synthesis method to create more labeled inputs a promising method.

Hybrid models which use a combination of different SSL approaches are popular in recent research and show much promise. Most of these models incorporate consistency regularization, including the MixMatch method, which has gained considerable attention. Hybrid models inspired by or incorporating MixMatch, as well as DSNA approaches such as MixUp, are common. Among the hybrid methods, the consistency-based adversarial method proposed by Wang et al. achieved impressive results with a labeled dataset of only 691 images. However, this study assumes that the labeled and unlabeled data have the same distribution. Therefore, further experimentation with differently distributed data is needed.
6. Gaps and Future Work

As one would expect, the performance of the models varied with different percentages of labeled data. Some methods, such as SRC-MT and NoTeacher, revealed up to a 10% difference in score from 1% to 10% labeled data. GraphX\textsuperscript{NET} showed an increase in more than 20%, with an AUC score of 0.53 and 0.78 for 2% and 20% labeled datasets, respectively. Other methods had smaller differences: the SSAC model’s AUC score increased by 3% from 20% to 100% labeled data, and the bi-modality synthesis model increased by 4% from 1% to 10% labeled data. Further research can be conducted on accomplishing higher robustness to changes in the percentage of labeled data and achieving successful training on lower numbers of labeled data.

As mentioned earlier, the areas of bi and multi-modality image synthesis is an area that can be studied further. Additionally, given the promise shown by the ACPL method, it may be beneficial to explore other strategies for measuring confidence and uncertainty.
in order to achieve improved results with the traditional pseudo-labeling method. Another area of learning that may be worth exploring is active learning. In this method, instead of using an existing dataset, a few valuable unlabeled samples are carefully selected, and a medical expert is asked to label them. As the samples to be labeled are chosen specifically based on the information they offer, this method results in a smaller amount of labeled data needed and, therefore, less time and effort in labeling from experts.

Lastly, more thorough experimentation is needed to gauge the performance of these models in a real-world clinical setting and with larger patient cohorts.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

**References**


