

Data Augmentation in Cancer Image Classification Problem with Quantum GAN

Muhammad Bagus Andra

bagus.andra.mail@gmail.com

National Research and Innovation Agency

Asri R. Yuliani

National Research and Innovation Agency

Jimmy A. Kadar

National Research and Innovation Agency

Ade Ramdan

National Research and Innovation Agency

Hilman F. Pardede

National Research and Innovation Agency

Research Article

Keywords: Cancer Classification, GAN, Generative Model, Quantum Computing, Quantum Machine Learning, Quantum GAN

Posted Date: November 15th, 2024

DOI: <https://doi.org/10.21203/rs.3.rs-5191116/v1>

License:  This work is licensed under a Creative Commons Attribution 4.0 International License.
[Read Full License](#)

Additional Declarations: No competing interests reported.

Version of Record: A version of this preprint was published at Quantum Machine Intelligence on September 15th, 2025. See the published version at <https://doi.org/10.1007/s42484-025-00315-y>.

Data Augmentation in Cancer Image Classification Problem with Quantum GAN

M. Bagus Andra^{1*}, Asri R. Yuliani^{1†}, Jimmy A. Kadar^{1†},
Ade Ramdan^{1†}, Hilman F. Pardede^{1†}

¹*Research Center for Artificial Intelligence and Cyber Security,
National Research and Innovation Agency , Street, Sangkuriang, Dago,
40135, Bandung, Indonesia.

*Corresponding author(s). E-mail(s): bagus.andra.mail@gmail.com;

†These authors contributed equally to this work.

Abstract

In this paper, we propose a novel approach leveraging Quantum Generative Adversarial Networks (QGANs) for data augmentation in cancer image classification tasks. Cancer diagnosis from medical images plays a crucial role in early detection and treatment planning. However, imbalance nature of medical image data pose significant challenges for training accurate classification models. Our proposed methodology utilizes QGANs to generate synthetic medical images that closely mimic the characteristics of real cancer images classes, effectively addressing the data imbalance issue. Our QGAN architecture, which combines quantum circuits for the generator with a classical neural networks for the discriminator, was trained to synthesize medical images for augmentation. The performance of the convolutional neural networks (CNN) classifier was evaluated using accuracy and the area under the receiver operating characteristics curve (AUC-ROC) as metrics. Results show that augmented datasets significantly improve the classification performance. For instance, the AUC-ROC for the Basal Cell Carcinoma (BCC) class increased by 0.031, and the Dermatofibroma (DF) class saw a reduction in validation loss by 0.06. Moreover, segmentation-based QGAN training demonstrated considerable advantages in computational efficiency. The training time was reduced to just one hour, and inference times were less than a second per image.

Keywords: Cancer Classification, GAN, Generative Model, Quantum Computing, Quantum Machine Learning, Quantum GAN

1 Introduction

The advent of GPU (Graphics Processing Unit) technology in recent years has shaped a new era for computer vision and the machine learning field. The computation power and parallelization of GPU enable a sophisticated architecture of deep learning model that gives the machine the capability to have several levels of visual perception of the input image. This capability makes it possible to perform a broad range of visual tasks from one that could be classified as low level such as recognizing edges and identifying similar regions to much more complex and high-level tasks such as scene understanding and Gaussian Splatting [1]. Many scientific fields have greatly benefited from the wide applications and the remarkable performance of these methods including medical imaging, medical video, and clinical analysis. Studies have shown the importance of medical imaging for the health industry [2] and is critical to help in early detecting, diagnosing, and treating diseases in a timely manner [3] [4].

The reason for the emergence of machine learning applications in medical image processing is twofold – first, the advancement of imaging technology and processing power enables scientists and medical institutes to collect clinical data quickly. Applying image detail enhancement and accurate region segmentation can be applied to clinical data such as brain CT scans and roentgen of lung images, resulting in a high-quality dataset. Some of the examples are the DermNet skin diseases atlas, which has 23 image classes and 23,000 images [5],

The Cancer Imaging Archive (TCIA), which records images of 37 types of cancer [6], and PCam(PatchCamelyon) classification dataset, which contains around 327,000 images of lymph node segment scans [7] [8]. With the help of machine learning algorithms, accurate prediction models could be developed using these data. Second, with the vast number of datasets and their high availability, manually extracting features becomes impractical, and approaches that can learn from raw data and automatically turn it into representative low-dimensional features are more appealing to examine [?]. The supervised and semi-supervised learning methods with deep learning are appropriate for this purpose and could take advantage of the sheer amount of accessible data to capture the generality of the subject domain. A plethora of deep learning architectures have been proposed throughout the years. Convolutional neural networks (CNNs) stand out to be arguably the most successful in dealing with image data. CNNs, like traditional neural networks, are modeled after neurons found in human and animal brains. Specifically, the visual cortex part of the brain is made up of a complicated sequence of cells. The architecture was made popular by Le-cun in 1980 [9] when they integrated the backpropagation method to optimize the kernel coefficients directly from the two-dimensional array. There are three main advantages of using CNN in image data, as outlined by Goodfellow et. al. [10] equivalent representations, sparse interactions, and parameter sharing. In contrast to traditional fully connected neural networks, CNNs use shared weights and local connections to optimize 2D input-data structures such as picture signals while also using fewer parameters to simplify training and speed up network performance. This emulates the real visual cortex cells in the brain. These cells sense small portions of a scene rather than the entire scene.

CNNs have also been used widely in the field of medical image processing, helping researchers uncover novel approaches and making an important advance in its interpretation. In the subject of pathology, researchers have used CNN to localize nucleus centroids and diagnose papillary cancer on thyroid fine-needle aspiration cytology smears [11] [12]. It has also been utilized in ophthalmology for the early diagnosis of diabetic retinopathy, allowing medical professionals to treat patients earlier and avoid total blindness [13]. The model uses binocular fundus images as input and learns their correlation to create prediction [14]. Various radiology techniques, such as X-rays, computed tomography, magnetic resonance imaging, and ultrasound, are critical in detecting and treating diseases, and they generate massive amounts of data. CNNs have gained popularity in radiology imaging and analysis due to their ability to process and interpret large datasets. Rajpurkar et.al. trained a 121-layers CNN to detect 14 classes of diseases from chest X-ray images [15] which was trained on the ChestX-ray14 dataset that was released by Wang et al. [16]. Remarkably, this model surpasses the average performance of radiologists on the F1 metric. Amidst the COVID-19 pandemic, many radiologists and physicians are actively embracing CNNs and other deep-learning methods to expedite the detection of pneumonia, thereby enhancing lung imaging analysis [17] [18].

Despite the significant advancements offered by CNN in the analysis of medical images, and the wealth of information and datasets available for training, several important challenges still persist. Research conducted by Jürgen Weese and Cristian Lorenz [19] provides an excellent summary of the four major challenges in medical image analysis that are likely to emerge in the near future. Figure 1 illustrates the block diagram of the approach necessary to address these challenges, as outlined in their research paper.

Among the four challenges described by [19], efficient ground truth generation is the most important when using a deep learning model to perform a detection or classification task, as the model's performance is closely related to the quality of the training dataset and ground truth for validation. However, medical image datasets are frequently imbalanced. A problem is considered imbalanced when one class within the available data is represented by a significantly smaller number of instances compared to other classes [20]. This often occurs because the incidence of anomalies or diseases is typically much lower than that of normal cases. As a result, the dataset's classes tend to be heavily skewed toward a single class, usually the negative class in disease diagnostics or the benign cells in cancer detection scenarios resulting in high false negative results which could lead to a low-occurrence high risk scenario that is crucial to be minimized in the medical field where the patients well-being is at stake.

Class imbalance has been a prominent problem in data mining and machine learning. This issue arises in domains where the classifier needs to identify rare yet significant cases [21], such as detecting fraudulent phone calls, intrusions, manufacturing process failures, or diagnosing uncommon medical diseases. In these contexts, the imbalance impedes the ability of traditional classification algorithms to accurately identify the cases of interest most of the time. It is important to address this issue as misclassifying a minority instance typically incurs a higher cost than misclassifying a majority one. Moreover, in a specific case, misclassifying a rare event can result in

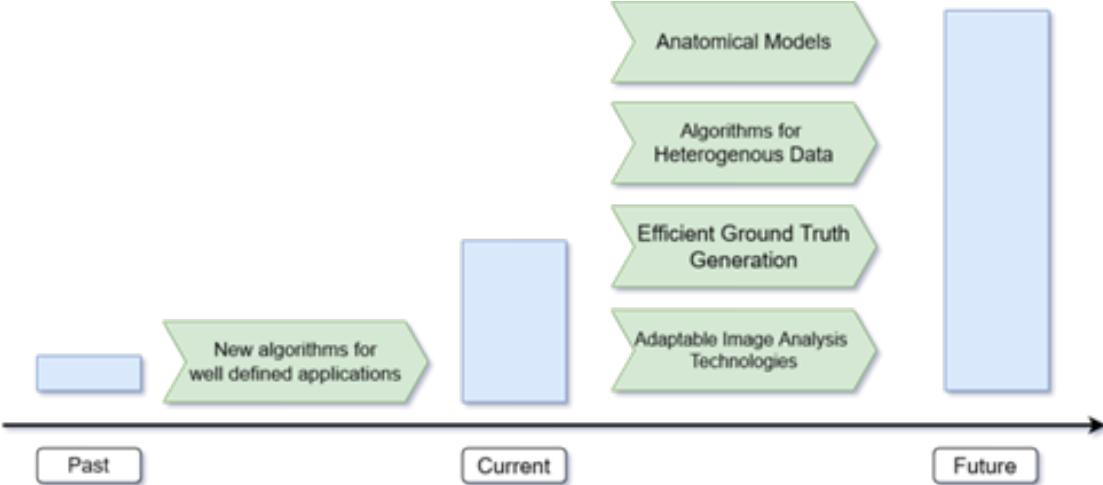


Fig. 1: Diagram depicting the evolution of main challenges for medical image analysis (adapted from [19])

more severe consequences than misclassifying a common event. For instance, in the context of cancerous cell detection in medical diagnosis, misclassifying non-cancerous cells may necessitate additional clinical tests. However, misclassifying cancerous cells poses significant health risks.

There have been many efforts to deal with the imbalance problems in the classical machine learning approaches. In contrast to most solutions, we aim to explore the solution by utilizing Quantum Computing (QC). The introduction of QC is a paradigm change in computer technology, with the potential to alter a wide range of fields. The notion of quantum mechanics, which is central to this area, evolved from atomic physics in the early 1900s. Richard Feynman advocated the use of quantum physics for computing in 1981 [?].

Quantum computers differ from classical computers by using qubits, which, unlike classical bits that exist in states of either 0 or 1, may represent both states simultaneously. Quantum information represents a quantum system's state and is required for quantum computers to function properly. These systems use unique quantum mechanical features, such as superposition and entanglement, to execute calculations [22]. The current level of QC is known as the Noisy Intermediate-Scale Quantum Era (NISQ). These systems can operate with fewer than 100 qubits but are not suitable for larger qubit sizes due to noise sensitivity and potential for quantum decoherence [23]. Generally, Today's QC falls into three categories: quantum computing, quantum information, and quantum cryptography. The capacity to use the huge array of permutations enabled by qubits is what gives QC its significant processing capability.

Notably, adding each qubit will double the quantum computer's memory capacity. In classical computing, to represent a system of N bits, we require N binary numbers. In quantum systems, the two possible definite states are denoted as $|0\rangle$ and $|1\rangle$. For a bipartite quantum system, a general state can be represented as

$\Phi = \alpha|00\rangle + \beta|01\rangle + \gamma|10\rangle + \delta|11\rangle$. From this representation, it is evident that a two-qubit quantum system can encode the equivalent of four classical bits of information ($\alpha, \beta, \gamma, \delta$). Similarly, an N-qubit quantum system can hold bits of classical information [24]. QC can be understood through the lens of a mathematical model akin to a Turing machine, which conceptually defines the operations of a machine that manipulates symbols on a tape according to a set of rules. Quantum computers are considered universal Turing machines. They leverage the principle of superposition in quantum mechanics to enable quantum parallelism. This capability allows quantum computers to perform certain probabilistic computations much faster than classical computers. Due to its high computing capability, QC is primarily utilized in scientific research where the computational demands exceed the capabilities of classical computers to deliver results within a feasible timeframe. For example, quantum computers are used to perform complex simulations such as protein folding and DNA assembly, as detailed in [25]. However, recently researchers are examining the potential of merging QC and machine learning to manage traditional data using machine learning techniques. Quantum Machine Learning (QML) is a new sub-discipline of study that combines machine learning theory and QC. QML intends to provide quantum applications for various machine learning methods, using the processing power of quantum computers and the scalability and learning capacity of these algorithms.

Quantum versions of common machine learning algorithms have already been created. Narayanan and Menneer [26] discussed the theoretical design of a Quantum Neural Network (QNN) architecture and its potential performance compared to traditional systems. Rebentrost et al. [27] developed Quantum Support Vector Machines (QSVM) to solve least-squares SVM with the HHL method [28], which generates the hyperplane by matrix inversion. In 2014, Wiebe et al. [29] introduced a quantum version of k-nearest neighbour that uses Euclidean distance and amplitude estimation to compute nearest neighbour without measurement. In 2018, Dang et al. [30] proposed an image classification model using quantum k-nearest neighbour and parallel computation. The model enhanced classification accuracy and productivity. Schuld et al. [31] suggested quantum linear regression, a variation of classical linear regression that runs in an exponential period with N features and uses quantum data as information.

Quantum computers are being used to implement deep learning algorithms, which need large storage and processing time. Quantum Generative Adversarial Networks (QGAN) [32][33][34] and their implementation in [35] use a superconducting quantum processor to generate real-world handwritten digit pictures. Quantum Wasserstein Generative Adversarial Networks (Quantum WGAN) [36] improve the scalability and stability of quantum generative adversarial model training on quantum machines. Quantum Boltzmann Machines [37][38] are also examples of these algorithms.

In this paper, we propose Quantum Generative Adversarial Network as an oversampling method to generate new images from the minority class. The synthetic data thus generated is used to oversample the respective class, thereby augmenting the dataset. Various classification methods are then applied to both the pre-augmented and augmented datasets to evaluate the impact of data augmentation on model performance. The model's effectiveness is assessed using the AUC-ROC and accuracy metrics. The main contributions of our article are as follows:

- Development of a novel QGAN, trained on a cancer dataset, to synthesize a new training dataset for the minority classes.
- Construction of an optimized GAN architecture that leverages the principles of QC and qubit encoding to maximize space efficiency and meet input dimension requirements.
- Application of Fuzzy C-means Clustering as an image segmentation technique to preprocess training data. This approach significantly enhances the efficiency of QGAN and reducing the training and inference time significantly.

Evaluation of our methods shows the advantages of using QGAN over classical GANs and other augmentation methods by comparing the performance of the classifier model. Our assessment indicates the QGAN model have the capability to capture the underlying representations of each class by examining the resulting synthesized data. Moreover, the augmented data also helps the model to reduce overfitting that is indicated by AUC-ROC score. The structure of this paper is organized as follows: Section 2 introduces several traditional data augmentation techniques, detailing how methods such as geometric transformations and color space adjustments are utilized in medical image analysis, and also explores the role of classical Generative Adversarial Networks (GANs) in generating synthetic medical images to address dataset imbalance. Section 3 offers a comprehensive description of the c-fuzzy segmentation method that is employed as preprocessing and the proposed QGAN architecture. Section 4 describes the classification methods employed, parameter used in the classification model, and the dataset used for the experiment. In Section 5, we explore the results of the experiment and we perform a thorough evaluation of the proposed method, including the classification performance, input feature evaluation and comparison to other data augmentation technique. In the final section of this paper, we conclude our research by presenting the summary and final verdict of our study.

2 Literature Review

2.1 Data Augmentation Method

Many real-world applications create data using a skewed distribution, resulting in an unequal distribution of classes within a dataset. In such instances, one class usually has significantly more samples than others, known as the majority class, while the class with fewer samples is known as the minority class. If the distribution of classes is imbalanced, the classifier may have difficulty determining decision boundaries. In the worst-case situation, the classifier may identify all instances as belonging to the majority class while completely ignoring the minority class

Given the significance of this issue, a wide array of techniques has been developed to tackle the problem of class imbalance. These techniques can be categorized into three groups based on their approach to handling class disparities. First, algorithm-level approaches involve proposing specific algorithms or modifying existing ones to better manage class imbalance. Second, data-level techniques add an extra processing step to mitigate the effects of skewed class distribution, including methods like undersampling and oversampling. Finally, hybrid-level methods merge both algorithmic and data-level

strategies, such as boosting and cost-sensitive learning, to create a more comprehensive solution [?]. This section will mainly focus on data-level techniques as the objective of our research is to use a generative model to augment the dataset and obtain equal class distribution

2.1.1 Basic Image Manipulation

The simplest and most common method in data augmentation in the image is by applying basic image manipulation such as geometric transformation (flipping, rotation, cropping) and modifying color space to the original image to create additional image data. With various transformations, the new data could be considered as new variations to the original ones. By doing so, a similar object with a different perspective or skewed shape could help the model avoid overfit [39]. For example, cropping could be used to provide an effect similar to translations but with a reduced dimension of the image input. Performing augmentations in the color channel space is another straightforward and effective strategy that can be implemented with minimal complexity. Simple color augmentations can include isolating a single-color channel, such as Red (R), Green (G), or Blue (B). More advanced color augmentations can be achieved by creating a color histogram that describes the distribution of colors within the image. This histogram serves as a detailed representation of the intensity and frequency of different colors in the image, allowing for sophisticated manipulations based on this color profile. Figure 2. Shows the result of applying color augmentation to an image.

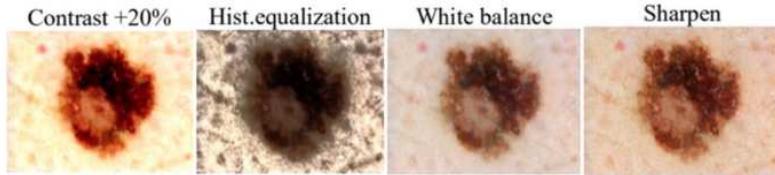


Fig. 2: Example of Medical Image with Color Augmentation

2.1.2 Image Mixing and Random Erasing

Mixing images by averaging their pixel values might seem counterintuitive as a data augmentation technique. Visually, the resulting images might not appear to be a useful transformation to a human observer. However, Inoue [40] demonstrated that this method of pairing samples could be turned into an effective augmentation strategy. In their experiment, two images are first randomly cropped from 256x256 pixels to 224x224 pixels and then randomly flipped horizontally. These images are subsequently blended by averaging their pixel values across each of the RGB channels. The resulting mixed image is then used to train a classification model, with the label of the new image assigned to match that of the first randomly selected image (Figure 3). This strategy is then expanded upon in [41], which resulted in better performance than the baseline models.

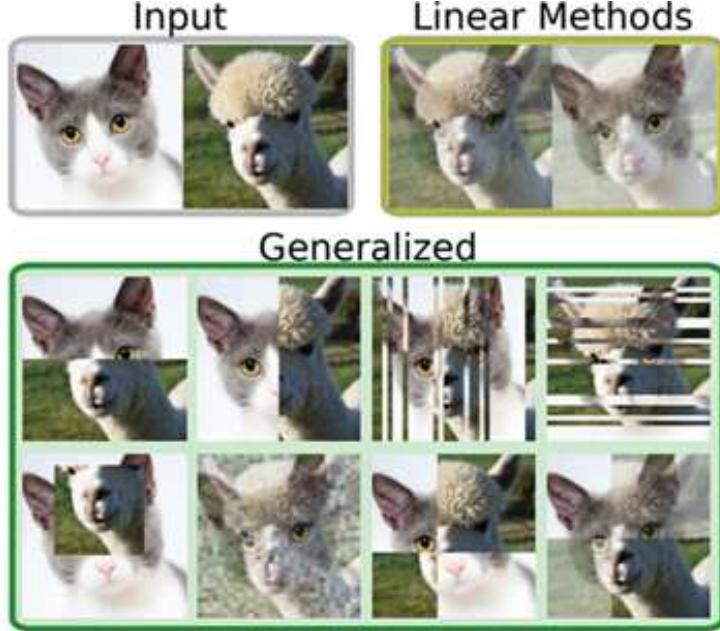


Fig. 3: Results of using Image mixing from [41]

Another approach for data augmentation proposed by Zhong et al [42] is random erasing. The method takes inspiration from the dropout regularization mechanism which can be seen as a similar operation to dropout except it's done in input data space rather than latent space. This technique addresses issues with image recognition caused by occlusion. Occlusion occurs when specific elements of an object are unclear. Random erasing prevents overfitting by encouraging the model to learn additional descriptive properties of a picture. It involves randomly selecting an $n \times m$ patch of an image and masking it with 0s, 255s, mean pixel values, or random values rendering that portion of the image to be pure black or white in terms of color representation. A similar approach has also been studied by Taylor [43] and by Mikolajczyk and Grochowski [44] called Cutout Regularization that combines random erasing with a deep learning model designed for inpainting. Figure 4 depicts the random erasing process in the input dataset.

2.1.3 Generative Model Based Data Augmentation

The generative model has gained popularity with the rise of complex deep-learning architecture. Generative modeling involves constructing fictional instances from a dataset with similar features to the original set. The most well-known generative model is arguably Generative Adversarial Networks (GANs). GANs are a unique type of neural network that involves the simultaneous training of two networks called generator (G) and discriminator (D). Generator focuses on generating images, while the other specializes in the discrimination of images generated by generators and real



Fig. 4: Random Erasing Process on the Image Dataset

ones. This adversarial training approach has captured the attention of both the academic and industrial sectors due to its effectiveness in mitigating domain shifts and its capability to generate new image samples. GANs have demonstrated state-of-the-art performance in a variety of image generation tasks, such as text-to-image synthesis [45], super-resolution [46], and image-to-image translation [47].

GAN could be defined as a generative model that was designed for directly drawing samples from the desired data distribution without the need to explicitly model the underlying probability density function. The input to G, z is pure random noise sampled from a prior distribution $p(z)$, which is commonly chosen to be a Gaussian or a uniform distribution for simplicity. The output of G, x_g is expected to have visual similarity with the real sample x_r that is drawn from the real data distribution $p_r(x)$. We denote the nonlinear mapping function learned by G parametrized by θ_g as $x_g = G(z; \theta_g)$. The input to D is either a real or generated sample. The output of D, y_1 is a single value indicating the probability of the input being a real or fake sample. The mapping learned by D parametrized by θ_d is denoted as $y_1 = D(x; \theta_d)$. The generated samples form a distribution $p_g(x)$ which is desired to be an approximation of $p_r(x)$ after successful training. D 's objective is to differentiate these two groups of images whereas the generator G is trained to confuse the discriminator D as much as possible. Intuitively, G could be viewed as a forger trying to produce some quality counterfeit material, and D could be regarded as the police officer trying to detect the forged items. In an alternative view, we can perceive G as receiving a reward signal from D depending on whether the generated data is accurate or not. The gradient information is back propagated from D to G , so G adapts its parameters in order to produce an output image that can fool D . The training objectives of D and G can be expressed mathematically as:

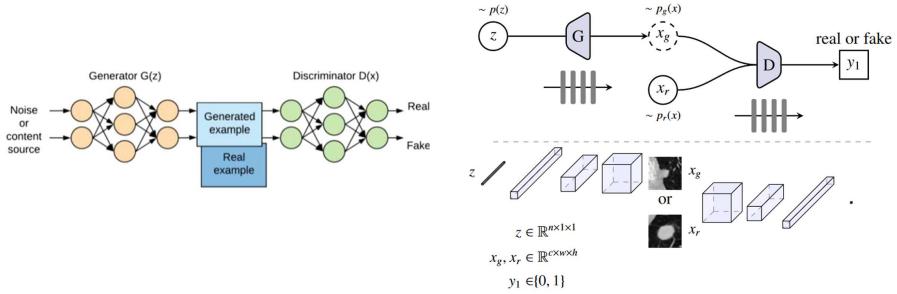


Fig. 5: Schematic view of the vanilla GAN for synthesis of lung nodule on CT images. Top of the figure shows the network configuration. The part below shows the input, output and the internal feature representations of the generator G and discriminator D . G transforms a sample from into a generated nodule. D is a binary classifier that differentiates the generated and real images of lung nodule formed by and respectively.

$$\begin{aligned} \mathcal{L}_D^{GAN} &= \max_D \mathbb{E}_{x_r \sim p_r(x)} [\log D(x_r)] + \mathbb{E}_{x_g \sim p_g(x)} [\log(1 - D(x_g))], \\ \mathcal{L}_G^{GAN} &= \min_G \mathbb{E}_{x_g \sim p_g(x)} [\log(1 - D(x_g))]. \end{aligned} \quad (1)$$

Where L_D^{GAN} and L_G^{GAN} are the loss of D and G respectively. As can be seen, D is simply a binary classifier with a maximum log-likelihood objective. If the discriminator D is trained to optimal before the next generator G updates, then minimizing L_G^{GAN} is proven to be equivalent to minimizing the Jensen Shannon (JS) divergence between $p_r(x)$ and $p_g(x)$. The desired outcome after training is that samples formed by x_g should approximate the real data distribution $p_r(x)$. The overall GAN scheme is depicted by Figure 5.

The capability of GANs makes them employed in medical imaging as well. The first on the ability of the generative component of GANs, which can aid in understanding the structure of training data and creating new images. GANs show great promise in addressing data scarcity and patient privacy. The second is discrimination, with discriminator D acting as a learned prior for normal images and a regularizer/detector for anomalies in images. Much research has shown the application of GAN in medical image synthesis with great success. Frid-Adar et al. [48] evaluated the effectiveness of using DCGANs for generating medical images of liver lesions. They employed the architecture shown above to produce images with dimensions of $64 \times 64 \times 1$, representing liver lesion CT scans. Their initial dataset comprised 182 CT scans, including 53 cysts, 64 metastases, and 65 hemangiomas. Initially, after applying traditional augmentation techniques, they achieved a sensitivity of 78.6% and a specificity of 88.4%. However, upon incorporating samples generated by DCGANs, they observed an increase in sensitivity to 85.7% and specificity to 92.4%.

In a study conducted by Wolterink et al. [49] on synthesizing brain CT images from MR images, the authors discovered that training with unpaired images yielded

better results than using aligned images. Hiasa et al. [50] further incorporated gradient consistency loss in the training to improve accuracy at the boundaries. Mok and Chung [51]. utilized cGAN (conditional Generative Adversarial Network) to augment training images for brain tumor segmentation. The generator in their setup was conditioned on a segmentation map and produced brain MR images using a coarse-to-fine approach. To enhance the clarity and delineation of the tumor in the generated images, they specifically designed the generator to include the tumor boundaries during the image generation process.

2.2 Quantum Computation in Medical Image

The capability of quantum computer to utilize parallel computing has facilitated the development of Quantum Machine Learning approaches. QML integrates and applies the properties of quantum computing within traditional machine learning methods. To improve the speed and efficiency of machine learning compared to traditional classical computing, QML focuses on the design and implementation of quantum software [52]. QML incorporates quantum algorithms in its framework, which could potentially surpass classical ML algorithms in solving certain problems. A variety of QML algorithms have been proposed, including quantum reinforcement learning (QRL), quantum principal component analysis, quantum support vector machines (QSVM), and quantum neural networks (QNN) [53].

Figure 6 illustrates data processed in QML in contrast to the classical machine learning (CML). Although the field of QML is still in its infancy, researchers have already begun applying it across multiple disciplines such as Chemistry, Industrial, healthcare, physics, and also in medical imaging. Andrea el. A. [54] proposes a Hybrid QCNN architecture for radiological image classification tasks. The model employs an additional quantum convolutional layer and combines it with the classical convolutional layer. The research investigates several quantum circuits designs, incorporating higher encoding, basic entangling layers, and strongly entangling layers. The results are promising, demonstrating that the Quantum Convolutional Neural Network (QCNN) consistently outperforms its classical counterpart, CNN, in terms of accuracy across all tested variations. Natansh et al. [55] conducted a similar study focused on classifying medical images. They utilized a quantum-assisted neural network in a hybrid architecture and also explored a pure variant known as the orthogonal quantum neural network. The experiments were conducted both in simulation and on actual quantum computer hardware provided by IBM. While the simulations yielded excellent results, the performance observed on the real quantum hardware was less impressive, providing a more realistic expectation of outcomes in real-world applications.

3 Proposed Method

In the current study, we explore two methodologies for utilizing QGANs to produce synthetic images. The first approach involves employing raw images as the input dataset without any modification except resizing. In the second approach, we trained the dataset with images that has gone through a pre-processing stage where feature extraction is executed prior to training the GAN. The trained GAN is utilized

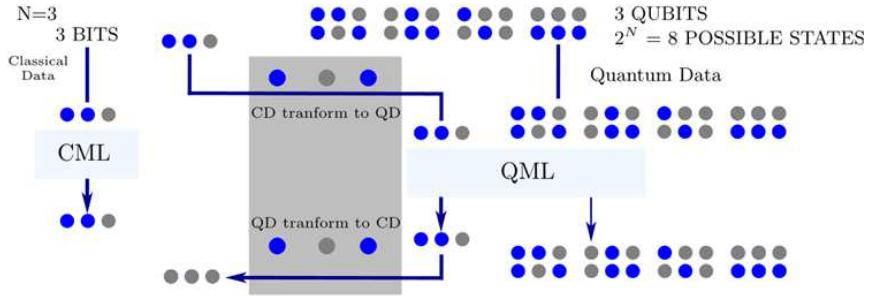


Fig. 6: How data is being processed in classical machine learning and quantum machine learning (adapted from [56]).

to generate both raw and segmented images corresponding to the class targeted for oversampling. The original dataset is then augmented with these newly synthesized images, which are employed to train a classifier. Following this, the performance of the classifier model is evaluated. The entire process of employing the QGAN is depicted in Figure 7, while Figure 8 illustrates the classification process.

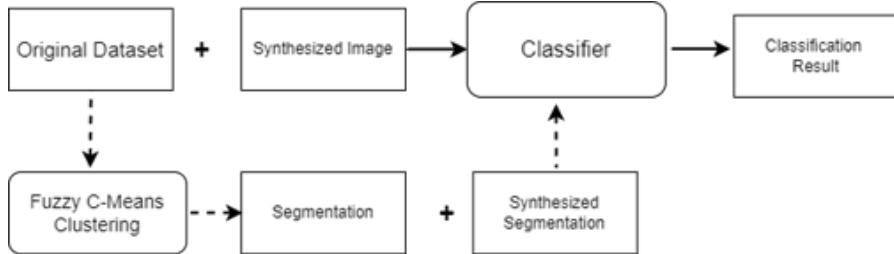


Fig. 7: The process of applying Quantum GAN for generating new synthesis data for both raw image and segmentation

3.1 Image Segmentation using Fuzzy C-means Clustering

The feature extraction is accomplished using a hybrid segmentation technique that combines C-fuzzy means and K-means clustering algorithms. This process unfolds in two stages. Initially, the image is converted into grayscale, followed by the application of K-means clustering to partition the image and identify clusters within it. Subsequently, the detected clusters are processed using C-fuzzy means clustering [57]. This two-step approach significantly reduces the processing time by pre-clustering the image before it undergoes FCM processing., utilizing fuzzy logic to determine this similarity. This algorithm has demonstrated its utility by yielding effective results in the segmentation of noisy images with minimal human intervention. Several studies have successfully employed this algorithm for segmenting skin lesions [58] and the retina

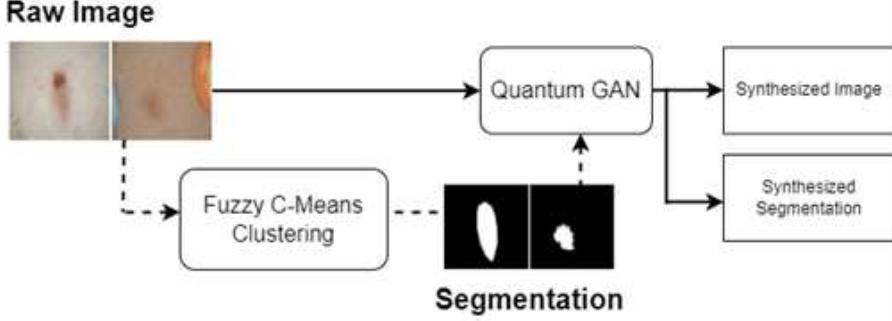


Fig. 8: Augmentation of original dataset by the mean of oversampling with the synthesized data and using it to train and evaluate classifier model

[59]. Fuzzy C-means is derived by minimizing the objective function of C-Means, which is defined as follows:

$$J_m = \sum_{j=1}^C \sum_{i=1}^N u_{ij}^m \|x_i - c_j\|^2 \quad (2)$$

For the Fuzzy C-means algorithm, each x_i represents a data point within the dataset, c_j is the center of a cluster, and m is a real value greater than 1 that affects the fuzziness of the clustering. The term $\|x_i - c_j\|^2$ quantifies the squared Euclidean distance between the data point x_i and the cluster center c_j . u_{ij} denotes the membership degree of the data point x_i in the in Fuzzy C-means is defined as:

$$u_{ij} = \frac{1}{\sum_{k=1}^c \left(\frac{\|x_i - c_k\|}{\|x_i - c_j\|} \right)^{\frac{2}{m-1}}} \quad (3)$$

1. **Initialization:** Determine the initial cluster centers, c_1, c_2, c_3 , and c_4
2. **Membership Calculation:** : Compute the degree of membership u_{ij} for each data point being analyzed.
3. **Centroid Update:** Update the cluster centers using the following formula:

$$c_j = \frac{\sum_{i=1}^N u_{ij}^m x_i}{\sum_{i=1}^N u_{ij}^m} \quad (4)$$

4. **Membership Update:** Recalculate the degree of membership u_{ij} for each data point with respect to the new cluster centers

Repeat steps 2 and 3 until the cluster centers stabilize, indicating minimal or no change between successive iterations. This convergence suggests that the optimal

cluster centers have been found, effectively partitioning the data according to the FCM algorithm's criteria. The example of a resulted image from the FCM process is depicted in Figure 9.

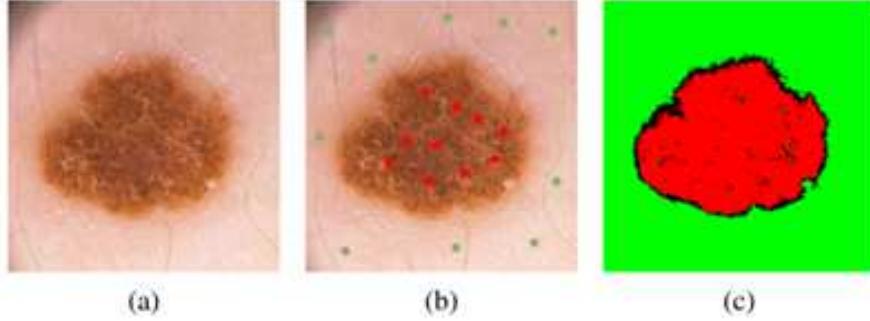


Fig. 9: Segmentation process on skin lesion image using C-fuzzy segmentation method (a) is the input image, (b) depicts the centroid calculated and (c) is the segmentation result.

3.2 Quantum Generative Adversarial Network

While numerous other studies have utilized QML for medical imaging, primarily focusing on segmentation and classification tasks, the application of QML to generative networks such as GANs remains largely unexplored. Specifically, how these quantum-enhanced generative models might address issues related to imbalanced datasets has received minimal attention. In this study, we aim to explore and analyze the capabilities of GAN implemented on a quantum circuit, utilized for generating a synthetic dataset to enhance the representation of underrepresented classes through oversampling. CNN as the classifier model is employed to classify the dataset both before and after the augmentation process. The results are then compared to assess the impact of data augmentation on the performance of the classifier. Furthermore, we also examine the ability of the GAN to generate synthesized images that effectively capture the underlying representations or features of each class.

The concept of a QGAN is similar to that of classical GAN, the architecture consists of two parts that is a generator G and a discriminator D . The generator G takes a latent vector z sampled from a specific distribution and aims to produce output $G(z)$ such that $P_g(G(z)) \approx P_{data}(x)$, effectively attempting to deceive the discriminator D . Concurrently, D strives to differentiate the true samples $x \sim P_{data}(x)$ from the generated samples $G(z)$. Distinct from classical GANs, both the generator and the discriminator in quantum GANs are constructed using quantum circuits. More specifically, we consider a quantum device equipped with N-qubits and a circuit depth of $O(\text{poly}(N))$. The feature dimension of the training examples is denoted by M . We have developed two adaptable strategies—the patch strategy and the batch strategy—to effectively leverage the available quantum resources, particularly under the conditions

of where it's $N < \|\log M\|$ and $N > \|\log M\|$ respectively. This patch method is adapted from the work of Huang et. al. [36]. The design of the quantum patch GAN is tailored to generate high-dimensional features using limited quantum resources. Conversely, the quantum batch GAN is designed for parallel training when sufficient resources are available. This approach allows our proposal to flexibly adapt to and maximize the use of available quantum resources. The quantum patch GAN, designed for scenarios where $N < \|\log M\|$, includes a quantum generator paired with a classical discriminator.

A notable advantage of the quantum generator is its potentially greater expressive power, which might enable it to better fit data distributions than its classical counterparts as it is enhanced capability stems from the unique properties of QC, such as superposition and entanglement. The quantum generator G is comprised of a series of sub-generators $\{G_t\}_{t=1}^T$, where each G_t is a parameterized quantum circuit (PQC), $U_{G_t}(\theta_t)$. Each sub-generator G_t is designed to transform an input state $|z\rangle$ into an output state $|G_t(z)\rangle = U_{G_t}(\theta_t)|z\rangle$, which represents a distinct segment of the high dimensional feature vectors. These sub-generators, scalable with $T \sim O\left(\frac{\log M}{N}\right)$, can be constructed on distributed quantum devices for parallel training or on a single quantum device for sequential training. The synthesized sample \tilde{x} is generated by measuring the states $\{|G_t(z)\rangle\}_{t=1}^T$ along the computation basis. Using both \tilde{x} and the true sample x , the loss function L is used to optimize the trainable parameters θ and γ for the generator G and discriminator D that is described by:

$$\min_{\theta} \max_{\gamma} \mathcal{L}(D_{\gamma}(G_{\theta}(\mathbf{z})), D_{\gamma}(\mathbf{x})) \quad (5)$$

Where $\mathcal{L}(D_{\gamma}(G_{\theta}(\mathbf{z})), D_{\gamma}(\mathbf{x})) = \mathbb{E}_{\mathbf{x} \sim P_{\text{data}}(\mathbf{x})} [\log D_{\gamma}(\mathbf{x})] + \mathbb{E}_{\mathbf{z} \sim P(\mathbf{z})} [\log(1 - D_{\gamma}(G_{\theta}(\mathbf{z})))]$.

$(G_{\theta}(\mathbf{z}))$, where $P_{\text{data}}(\mathbf{x})$ refers to the distribution of training dataset, and $P(\mathbf{z})$ is the probability distribution of the latent variable z . The operation of the quantum patch GAN works as follows. Initially, a latent vector $|z\rangle$, drawn from the designated latent space, serves as the input to a quantum generator, G . This generator comprises T sub-generators each constituted by a parameterized quantum circuit (PQC), $U_{G_t}(\theta_t)$. Subsequently, the image synthesis involves measuring the resultant quantum states $\{U_{G_t}(\theta_t)|z\rangle\}_{t=1}^T$ along the computational basis, thereby producing the generated image. This image, along with a corresponding real image, is then sequentially fed into a classical discriminator, D (also highlighted in the pink region). The discriminator's output is employed by a classical optimizer to adjust the trainable parameters of both G and D based on the classification results. This process delineates a single iteration of the model's training cycle. The overall structure of the patch quantum generator is depicted in Figure 10.

In addition to employing the patch quantum generator method, we have incorporated several enhancement techniques as proposed by reference [57]. These enhancements include quantum generator augmentation and quantum discriminator enhancement. These methods are designed to improve the robustness and performance of the quantum generative adversarial network by refining both the generative and discriminative components of the model. In the augmentation of the quantum generator \widehat{UG} , a quantum-enhanced term \widehat{VG} then incorporated into the generator's

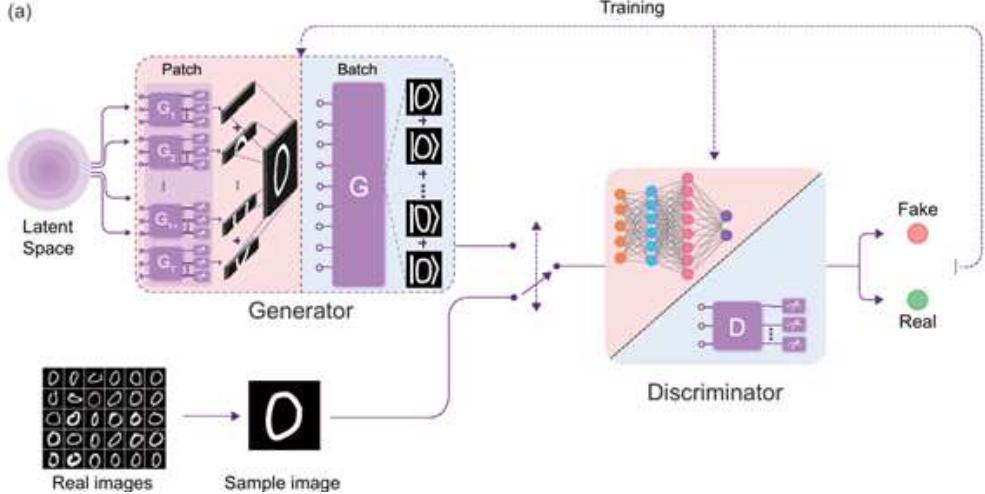


Fig. 10: Diagram of patch quantum GAN mechanism as adapted from [36]

Hamiltonian. This term modifies the evolution of the generator according to the equation:

$$\widehat{UG} = e^{-i(\widehat{HG} + \lambda\widehat{VG})t} \quad (6)$$

where \widehat{HG} represents the original Hamiltonian of the quantum generator, t denotes the evolution time, and λ is a parameter that modulates the influence of the quantum-enhanced term. The addition of \widehat{VG} is strategically designed to leverage particular quantum effects that are favorable for enhancing the training dynamics of the system. As for the discriminator, \widehat{UD} similarly received an enhancement through the employment of a quantum-enhanced term \widehat{VD} in its Hamiltonian. The modified Hamiltonian is given by:

$$\widehat{UD} = e^{-i(\widehat{HD} + \lambda\widehat{VD})t} \quad (7)$$

where \widehat{HD} is the original Hamiltonian of the quantum discriminator, and λ controls the influence of the quantum-enhanced term. This augmentation is devised to increase the discriminating capabilities of the quantum discriminator, enabling a higher capability to discriminate between true and generated quantum states.

4 Dataset and Classifier

In this section, we offer a comprehensive overview of the dataset used for this study, detailing its sources, characteristics, and relevance to the objectives of our research. Additionally, we describe the architecture of the Convolutional Neural Network employed as the classifier in this experiment. This includes a discussion on the configuration of layers, the chosen parameters for the network, and how the model is optimized for the specific challenges presented by the dataset.

4.1 HAM10000 Dataset

The HAM10000 ("Human Against Machine with 10000 training images") dataset [60] [61] is a comprehensive skin cancer dataset composed of dermatoscopic images gathered from diverse populations, using different acquisition and storage modalities. The dataset ultimately comprises 10,015 dermatoscopic images, making it suitable for academic machine learning research. It includes a representative assortment of all significant diagnostic categories within the domain of pigmented lesions, such as Actinic keratoses and intraepithelial carcinoma/Bowen's disease (akiec), basal cell carcinoma (bcc), benign keratosis-like lesions (solar lentigines/seborrheic keratoses and lichen-planus like keratoses(bkl)), dermatofibroma (df), melanoma (mel), melanocytic nevi (nv), and vascular lesions (angiomas, angiokeratomas, pyogenic granulomas, and hemorrhage (vasc)). The dataset encompasses six distinct classes, with data collected from individuals ranging in age from 0 to 85 years and including both genders. Figure 11 showcases various data samples, while Figure 12 visualizes the distribution of classes within the dataset.

As evident from the plot, the class distribution in the dataset is significantly skewed towards the melanocytic nevi (nv) class, which refers to moles. This skew is understandable given that in data collection processes, particularly in contexts like cancer detection, negative data (non-disease cases) is typically more abundant compared to positive (disease) samples. The significant disparity in sample quantity and the imbalanced distribution of classes could potentially lead to an overfitting model, a scenario that must be avoided in critical tasks such as cancer detection. This situation exemplifies where data augmentation plays a crucial role. By evenly distributing the classes, data augmentation not only improves the quality of the dataset but also enhances the performance of the model.

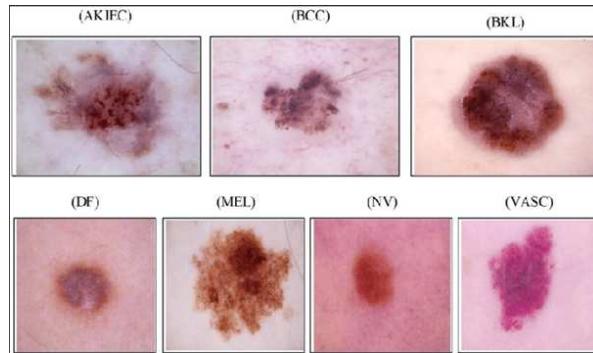


Fig. 11: Several images from HAM10000 dataset with its corresponding classes

4.2 Convolutional Neural Network Classifier

For the experiment, we are utilizing CNN as the classifier model. As discussed in the previous section, CNNs have been demonstrated to be one of the most effective deep

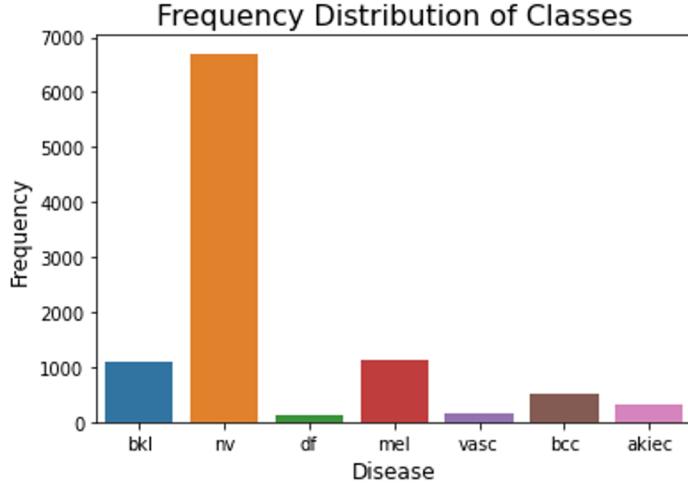


Fig. 12: Class frequency distribution of HAM10000 dataset

learning architectures for handling image data. This makes them a suitable foundation for assessing the effectiveness of our data augmentation methods. In CNNs, the convolutional layer performs an operation where an input image and a convolution filter are combined to produce an output feature map. The essential components of CNNs typically include multiple convolutional layers, with pooling layers in between, followed by one or more fully connected layers. These layers work in tandem to extract and reduce feature dimensions, enabling the network to efficiently learn spatial hierarchies in data. These layers are then stacked to achieve deep network architecture that could learn the underlying pattern in the input imgae. The convolutional layers compute new pixel values from a linear combination of nearby pixel values in the preceding feature map, expressed as:

$$x_{ij}^{(\ell)} = \sum_{a,b=1}^{Pw} w_{a,b} x_{i+a,j+b}^{(\ell-1)} \quad (8)$$

Where $w_{(a,b)}$ represents the weights of a ww filter matrix. Pooling layers then follow, aimed at reducing the size of the feature map. This reduction is typically achieved by performing operations such as taking the maximum value (max pooling) from a set of contiguous pixels. These layers are frequently paired with the application of a nonlinear activation function to introduce non-linearity into the model, helping it to learn more complex patterns. Once the feature map size is sufficiently reduced, the final output is computed using a fully connected layer that depends on all remaining pixels. This layer integrates all the learned features across the image into a final classification outcome which is then associated with one of the dataset labels. A typical CNN model architecture is depicted in Figure 13.

In this experiment, the classification process using the CNN proceeds as follows: the input image is initially fed into the input layer of the CNN. The input images is then transformed into lower dimensional feature space by applying four convolutional

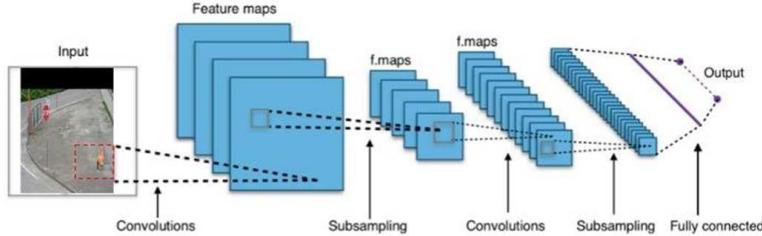


Fig. 13: Components of a typical CNN model architecture

layers are applied, each with a kernel size of 3 and a stride value of 1. Following these convolutional layers, the feature map is directed into a fully connected network. This network includes two dropout layers inserted between the layers to help prevent overfitting. The process concludes at the output layer which uses softmax function that performs the final classification task. The parameters used for the network are detailed in Table 1 and Table 2.

Table 1: Parameter utilized for training the CNN model

Learning Rate	15×10^{-4}
Loss Function	Binary Cross Entropy
Optimizer	ADAM
Normalization Layer	Batch Normalization 2 D
Output Activation Function	Sigmoid
No. of Parameter	460 k

Table 2: Specification of the convolutional layer

Kernel Size	3
Stride	1
Activation Function	ReLU
Normalization Layer	Batch Normalization 2D
Maximum Pooling Layer	2 x 2

5 Experiment

In this section, we conduct a series of experiments to evaluate the performance of the proposed Quantum GAN model. The HAM10000 skin cancer dataset, which effectively encapsulates the challenges this research aims to address, will be used for these

experiments. The Quantum GAN will be trained specifically on classes that are under-represented in the dataset and subsequently employed to augment the existing dataset. CNN as the classifier model will then be trained on this augmented dataset to perform. The classification performance of the model before and after augmentation is then evaluated.

5.1 Classification Performance Evaluation

For this experiment, the QGAN is trained on each class within the HAM10000 dataset, excluding the melanocytic nevi (nv) class. The trained models are then employed to generate new images for each class, which are subsequently used to augment the original dataset. The training process utilizes segmented images as input features. The distribution of classes following augmentation is depicted in Figure 14.

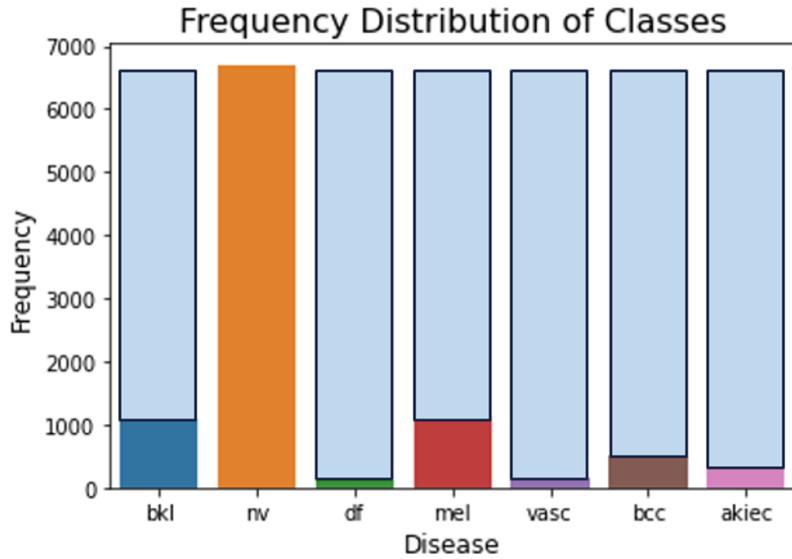


Fig. 14: Distribution of the HAM10000 image classes post data augmentation. The teal-colored plot illustrates the volume of synthesized data added to the original dataset

The QGAN architecture employed is as outlined in Section 3.2, utilizing a patch-based approach. In this configuration, the generator consists of several Parameterized Quantum Circuits (PQCs), each serving as a sub-patch generator. These sub-patches are ultimately combined to form a single coherent image. On the other hand, the discriminator component of the system employs a traditional fully connected neural network architecture. This hybrid approach leverages the strengths of QC in the generation process while maintaining the reliability of classical methods for discrimination. The parameters employed for the QGAN are detailed in Table 3 and 4.

Table 3: The specifications of the proposed generator

No. of Qubit	7
No. of Aux. Qubit	1
No. of Sub-generator	16
PQC Depth	10
Learning rate	0.1
Epoch	2000

Table 4: The specifications of the proposed discriminator

No. of Layers	5
Activation Function	ReLU
Output Layer Activation	Sigmoid
PQC Depth	10
Learning rate	5×10^{-3}
Epoch	2000
Loss Function	Binary Cross Entropy

5.1.1 BKL vs NV

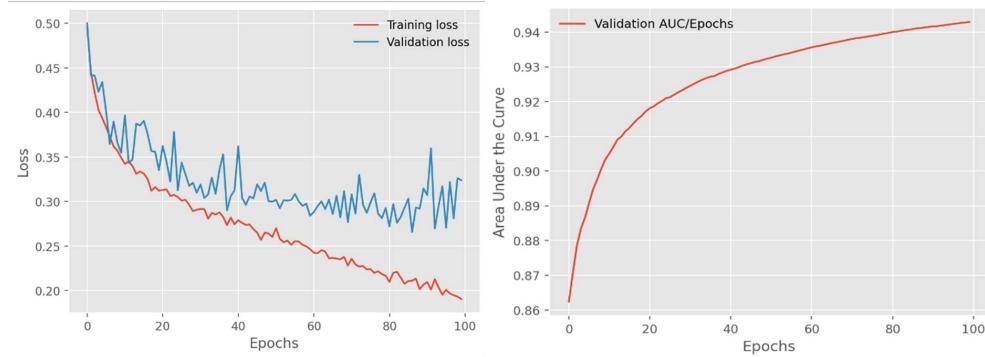


Fig. 15: Loss and AUC-ROC curve for BKL vs NV

Table 5: Performance comparison with and without augmentation for BKL vs NV

	Without Augmentation	With Augmentation
Train Loss	0.198	0.177
Validation Loss	0.340	0.320
AUC/ROC	0.956	0.971

5.1.2 DF vs NV



Fig. 16: Loss and AUC-ROC curve for DF vs NV

Table 6: Performance comparison with and without augmentation for DF vs NV

	Without Augmentation	With Augmentation
Train Loss	0.205	0.255
Validation Loss	0.277	0.210
AUC/ROC	0.953	0.970

5.1.3 MEL vs NV

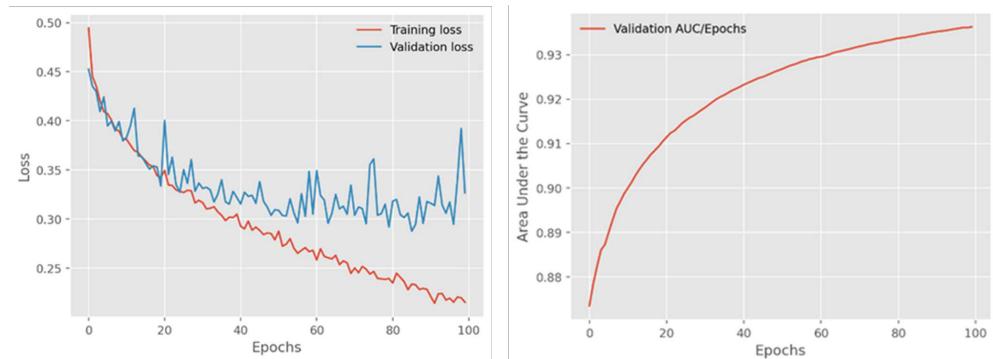


Fig. 17: Loss and AUC-ROC curve for MEL vs NV

Table 7: Performance comparison with and without augmentation for MEL vs NV

	Without Augmentation	With Augmentation
Train Loss	0.198	0.216
Validation Loss	0.372	0.331
AUC/ROC	0.92	0.931

5.1.4 BCC vs NV

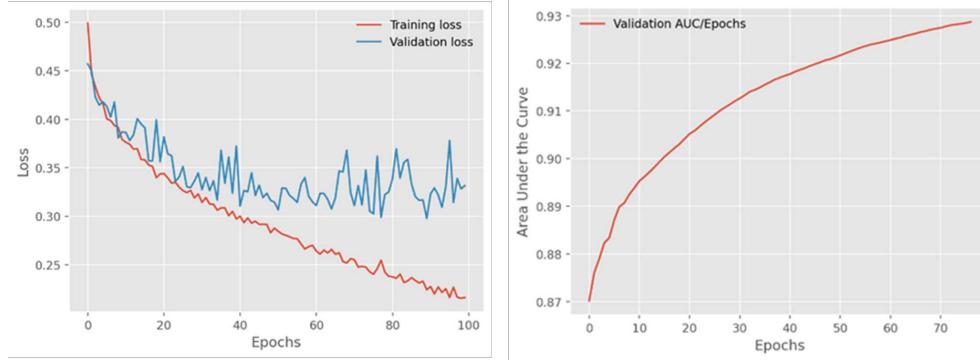


Fig. 18: Loss and AUC-ROC curve for BCC vs NV

Table 8: Performance comparison with and without augmentation for BCC vs NV

	Without Augmentation	With Augmentation
Train Loss	0.247	0.215
Validation Loss	0.335	0.326
AUC/ROC	0.905	0.936

5.1.5 AKIEC vs NV

Table 9: Performance comparison with and without augmentation for AKIEC vs NV

	Without Augmentation	With Augmentation
Train Loss	0.210	0.225
Validation Loss	0.337	0.315
AUC/ROC	0.912	0.925



Fig. 19: Loss and AUC-ROC curve for AKIEC vs NV

5.1.6 Result Discussion

This experiment aims to observe the impact of adding synthesized data as data augmentation to the classification performance of the model. Among the 6 class in the dataset, NV class is selected as the baseline for comparison in a one-to-one binary classification experiment as compared to other class, NV represent the majority of the dataset. By comparing the class with size discrepancy one-to-one to the NV class we could see whether the augmentation method is effective for every type of class and it could generalize well. While multiclass classification could be performed we argue that the similar size of the remainder class other than NV will result in the situation that does not reflect the impact of the augmentation technique.

From the results presented above, a consistent trend is observed across all evaluations: there is a decrease in validation loss and an increase in AUC/ROC scores when comparing classifications on augmented datasets to those without augmentation. The training progress of the classifier across all classes also displays a similar trend, where it successfully minimizes the training loss to a significantly low value while achieving remarkable scores on validation tests. This trend further corroborates the effectiveness of the model in handling diverse data scenarios, affirming its capability to generalize well across different sets of data. The increase in AUC/ROC scores indicates that the classifier model, when trained with augmented data, is better able to differentiate between the two classes. This enhancement in model performance underscores the effectiveness of data augmentation in improving the robustness of the classifier.

The improvement in validation loss is particularly notable, with a reduction of up to 0.06 in the Dermatofibroma (DF) class, indicating that the additional newly generated data contributes significantly to better generalization for this class. A substantial improvement is also observed in the Melanoma (MEL) class, with a decrease in validation loss of approximately 0.041. These improvements demonstrate the tangible benefits of data augmentation in enhancing model performance across less represented classes.

The significant increase in the AUC/ROC score for the Basal Cell Carcinoma (BCC) class, with an improvement of 0.031, underscores the effectiveness of the augmented data in capturing the distinct characteristics of the BCC class. This

enhancement allows the classifier to more effectively distinguish between BCC and other classes, particularly when contrasted with the Melanocytic Nevi (NV) class used as the baseline. The introduction of synthesized images that closely represent the nuanced features of BCC appears to have crucially contributed to improving the classifier's diagnostic accuracy, thus highlighting the pivotal role of targeted data augmentation in medical image analysis, especially in cases where certain conditions are underrepresented in the training dataset.

The observation regarding the slight increase in training loss for the Dermatofibroma (DF), Melanoma (MEL), and Actinic Keratoses (AKIEC) classes, despite the higher AUC/ROC scores, offers an insightful perspective on the model's behavior during training. Typically, a reduction in training loss is associated with improved model performance, but in this context, the slight increase in loss could indicate that the model is avoiding overfitting the training data. This scenario suggests that while the model may not perform as optimally on the training data itself (as reflected by the higher training loss), it is developing a better-generalized performance on unseen data, which is crucial for reliable medical diagnostics. The augmentation appears to introduce sufficient variability and complexity into the training process, thereby preventing the model from merely memorizing the training data and encouraging it to learn more robust features that apply to a broader range of data. This is corroborated by the increased AUC/ROC scores, which signify enhanced capability in discriminating between different classes, particularly in distinguishing between disease and non-disease cases.

Thus, the trade-off between higher training loss and increased validation metrics like AUC/ROC is a positive indication that the model is becoming more adept at handling real-world variations in the data, which is often the objective in medical image analysis and other applications where the ability to generalize is more critical than achieving minimal loss on the training set.

5.2 Input Feature Evaluation

This next experiment presents an interesting comparison in the performance of the QGAN when trained on different types of input images: raw original images, raw resized images, and resized segmented images. The choice to experiment with these varying input formats is to explore how the QGAN handles complexity and detail in image data.

The first input type consists of raw original images that are unmodified and contain all details and noise from the capture process, which might challenge the QGAN due to high variability and the presence of irrelevant information. The second type, raw resized images, involves scaling down the original images to a uniform size of 24x24 pixels, which may introduce or remove details, potentially affecting the GAN's ability to learn relevant features. The third type is resized segmented images, similar to those used in previous experiments. Here, the segmentation process isolates important features of the images by removing background noise and focusing on areas of interest, potentially making it easier for the QGAN to learn and generate relevant features. The target class for augmentation is Dermatofibroma (DF), chosen due to its strong performance in prior experiments. DF will be compared against the NV (melanocytic

nevi) class, allowing for a focused analysis of how well the QGAN can generate and augment data specifically for DF under varying input conditions.

The classifier model parameters remain unchanged to maintain consistency in measuring the impact of the input variations on the performance of the generated images. This control ensures that any observed differences in classifier performance can be attributed to changes in the GAN’s training data rather than alterations in the classifier configuration. Table 10 shows the result of this experiment.

Table 10: Performance comparison of classifier model with various input data

	Raw Image	Raw Image + Augment	Raw Resized	Raw Resized + Augment	Resized Segmentation	Resized Segmentation + Augment
Train Loss	0.124	0.274	0.184	0.362	0.205	0.255
Validation Loss	0.245	0.393	0.288	0.464	0.277	0.210
AUC/ROC	0.975	0.902	0.934	0.873	0.953	0.970
Training Time	5 hours	8 hours	3 hours	4 hours	1 hour	1.5 hours
Inference Time	10 seconds	12 seconds	3 seconds	3.5 seconds	<1 second	<1 second

The results from the experiment offer valuable insights into the capabilities of the QGAN when dealing with different types of image inputs. Interestingly, contrary to initial expectations, the use of raw, unmodified images for training the classifier led to the best performance outcomes. This superior performance can be attributed to the fact that raw images retain the most original data, providing CNN with a rich array of information to extract and learn from. The complexity and richness of these images seem to have allowed CNN to better capture and utilize the inherent features for classification tasks.

However, the results took an unexpected turn when it came to the performance enhancement promised by the synthesized images generated by the QGAN. Instead of improving the classifier’s performance, the augmentation with synthesized images resulted in a decline. The results obtained align with those observed for resized raw images. Without data augmentation, the performance of these images is quite commendable. However, the reduction in dimensions leads to a loss of information, resulting in poorer performance compared to that of resized segmented images. Upon applying data augmentation, a further decline in performance is noted. Conversely, the performance of resized segmented images with augmentation is notably better. This improvement can likely be attributed to the feature extraction process, which isolates the most critical structural elements of the image, thereby facilitating the model’s ability to distinguish between classes more effectively. In scenarios involving augmentation, raw resized images exhibit the poorest performance across all cases, whereas

resized segmented images augmented with additional data show a marked performance improvement compared to their non-augmented counterparts.

In terms of inference speed and training time, our proposed method using segmented images provides considerable advantages, significantly accelerating the training process. Specifically, the training duration is reduced to merely one hour, and the inference time is expedited to less than a second. This enhancement in speed facilitates more efficient iterations and potentially quicker improvements in model accuracy through more frequent tuning and validation. Although the overall performance of using segmented images is slightly less optimal compared to using raw images, the marginal difference in performance is compensated by substantial gains in processing efficiency. This trade-off is particularly beneficial in scenarios where computational resources are limited or when rapid processing is prioritized.

5.3 Data Augmentation Method Comparison

In the final experiment, we aim to evaluate the efficacy of our proposed QGAN for data augmentation by comparing its results with those of another established augmentation method. This experiment mirrors the methodology outlined in section 5.2, where DF represents the underrepresented class selected for augmentation. The objective is to enhance the classification accuracy of a two-class model using different augmentation techniques. Specifically, the training dataset for the DF class will be augmented using three distinct methods: basic geometric manipulations, a classical GAN, and our proposed QGAN. The raw full image and segmented image are both used as input features for comparison. This comparative analysis seeks to analyze the relative improvements each method offers in the context of enhancing model performance. The data augmentation method used for comparison is as follows:

5.3.1 Geometric Manipulation

For the initial data augmentation technique, we employed basic geometric manipulations on the training dataset. The specific operations conducted included rotating the images by 90, 180, and 270 degrees, as well as cropping and shifting. All operations were performed while maintaining the original image dimensions. This process was repeated until the number of images in the DF class was equal to that of the NV class, thus ensuring a balanced dataset for subsequent classification tasks.

5.3.2 Classical GAN

The second augmentation method involved the use of a Classical Generative Adversarial Network to generate synthetic images. All images from the DF dataset were utilized to train the GAN model, which employs the vanilla GAN architecture as detailed in Section 2.1.3. In a manner akin to the first method, synthetic data was generated until the number of samples in the DF class matched that of the NV class, thereby ensuring a balanced dataset for improved model training and evaluation. An example of a generated image from the classic GAN is shown in Figure 20. For the final comparative analysis, the proposed QGAN architecture was utilized to generate synthetic images,

similar to those produced by the classical GAN. Samples of the images generated from the proposed QGAN are illustrated in Figure 21

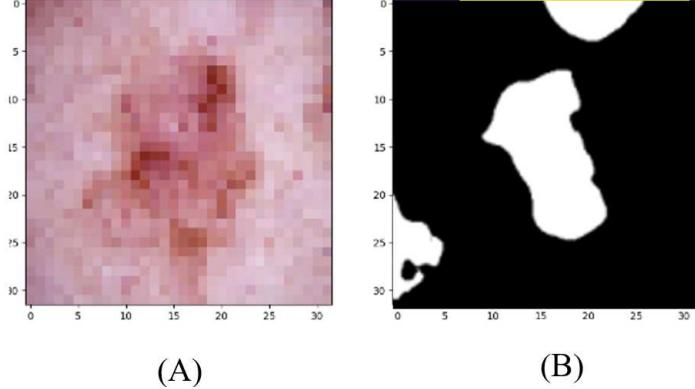


Fig. 20: Generated Images from Classical GAN: (A) shows a synthesized image derived from the original raw image, and (B) displays the synthesized segmentation.

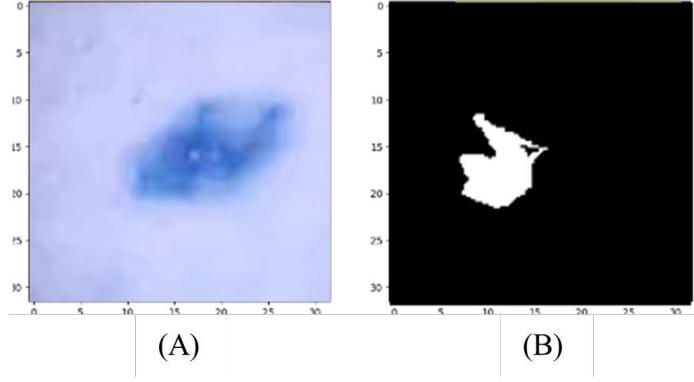


Fig. 21: Generated Images from QGAN: (A) shows a synthesized image derived from the original raw image, and (B) displays the synthesized segmentation.

5.3.3 Image Mixing and Random Erasing

The third augmentation method adopts the image mixing and random patch erasing as described in Section 2. For this augmentation, training image for each class are sampled randomly matching the amount of augmented image used in the QGAN and CGAN methods. This randomly sampled images are then divided further into two group. Random erasing is performed on one group while image mixing is performed on the other one. These images are then combined with the original images to be used

for training. The process is done using image processing library (OpenCV). The patch that is mixed and erased is randomly selected from the 2x2 image array in the size of 1/16 of the image size.

5.3.4 Result Discussion

The classifier model, maintaining the same architecture and parameters as discussed in the previous section, was trained with each case of the augmented dataset. It was then employed to classify the DF and NV classes using each augmentation method. The comparative effectiveness of these methods was assessed based on accuracy and AUC/ROC scores. The results of these experiments are presented in Table 11.

Table 11: Performance comparison of various data augmentation methods

	Geometry manipulation		Image Mixing and Erasing		Classical GAN		QGAN	
	Raw Image	Segment	Raw Image	Segment	Raw Image	Segment	Raw Image	Segment
Accuracy	0.87	0.92	0.84	0.88	0.98	0.95	0.89	0.96
AUC/ROC	0.82	0.96	0.86	0.94	0.98	0.97	0.87	0.97

From the results, it is evident that augmentation using simple geometric manipulations as data augmentation results in the model that does not perform too well in the classification task. Specifically, incorporating geometrically manipulated data into the synthetic dataset resulted in an accuracy of 0.87 when distinguishing between the NV and DF classes. This reduced performance can likely be attributed to the fact that the rotation and shifting operations did not adequately capture the defining characteristics and uniqueness of the respective classes. Consequently, these operations may have introduced excessive variability into the DF class, rendering the data similar to that of other classes.

However, this method still performs better when compared to the image mixing and random erasing which performs generally worst compared to other methods. This result could be attributed to the fact that mixing and erasing operation in this specific dataset reduces the information of the class significantly as most of the feature in this dataset is represented by the border and shape of the object of interest. Interestingly, the introduction of the mixing and erased images in the dataset improves the AUC-ROC score of the model when using raw image as the input feature. While this might be due to the regularization effect that's occurred after the addition of the synthesized image, it is arguable whether this method is suitable for this specific case. The comparative analysis between the classical GAN and the Quantum GAN provides deeper insights into the performance of these models. Utilizing raw image data as input features, the classical GAN demonstrated superior ability in generating images that closely resemble those of the original class and effectively capture its properties. This is evidenced by its high performance, achieving an accuracy of 0.98 and an AUC/ROC score of 0.982. In contrast, the Quantum GAN faced challenges in

accurately generating raw images, which affected the model’s performance, resulting in a lower accuracy of 0.89 when augmenting the dataset.

However, when examining the performance of the model augmented using segmentation images. In this scenario, the QGAN emerges as superior compared to other models, achieving an accuracy score of 0.95 and an AUC/ROC score of 0.975. This superior performance can be attributed to the architecture of our proposed QGAN, which utilizes the patch method. This architectural approach enables the model to better capture the localized properties of the class, which is crucial for segmentation models that rely heavily on the structural shape and contour of the image. Consequently, this gives our proposed model a distinct advantage in handling segmentation tasks.

6 Conclusion

In this study, we have presented the use of a QGAN model for data augmentation in the context of the HAM10000 skin cancer dataset. The proposed QGAN model was specifically trained on under-represented classes within the dataset and successfully generated synthetic images to balance class distribution. This augmentation approach, when compared with traditional data augmentation techniques, demonstrates a significant improvement in model performance in terms of generality and robustness of the classifier model.

Our experimental results shows that the use of augmented data through QGAN notably enhances the generality and AUC-ROC scores across various classes. Moreover, the comparison between raw images, raw resized images, and segmented images as inputs for the QGAN showed that while raw images retained the most information leading to higher initial performance, segmented images, when augmented, showed a marked improvement in model performance. This indicates that the pre-processing and structuring of input data play a significant role in the efficacy of synthetic data augmentation.

Acknowledgements. The studies are funded by Research and Innovation for Advanced Indonesia Program with contract number: B- 838/II.7.5/FR.06/5/2023.

Ethic Approval. This research study was performed without the involvement of human and animal subjects. The data used in the research study is publicly available and any data that includes specimens and/or samples collected from human subjects have undergone an ethics clearance process in their respective original publications.

References

- [1] Kerbl, B., Kopanas, G., Leimkühler, T. & Drettakis, G. 3D Gaussian Splatting for Real-Time Radiance Field Rendering (2023). URL <http://arxiv.org/abs/2308.04079>. ArXiv:2308.04079 [cs].

- [2] Zhu, Z., Wang, S.-H. & Zhang, Y.-D. A Survey of Convolutional Neural Network in Breast Cancer. *Computer Modeling in Engineering & Sciences* **136**, 2127–2172 (2023). URL <https://www.techscience.com/CMES/v136n3/51805>.
- [3] Sannasi Chakravarthy, S. R., Bharanidharan, N. & Rajaguru, H. Multi-Deep CNN based Experimentations for Early Diagnosis of Breast Cancer. *IETE Journal of Research* **69**, 7326–7341 (2023). URL <https://www.tandfonline.com/doi/full/10.1080/03772063.2022.2028584>.
- [4] Susilo, A. B. & Sugiharti, E. Accuracy Enhancement in Early Detection of Breast Cancer on Mammogram Images with Convolutional Neural Network (CNN) Methods using Data Augmentation and Transfer Learning. *Journal of Advances in Information Systems and Technology* **3**, 9–16 (2021). URL <https://journal.unnes.ac.id/sju/index.php/jaist/article/view/49012>.
- [5] DermNet, N. Dermnet | Dermatology Resource. URL <http://dermnetz.org>.
- [6] TCIA. Welcome to the Cancer Imaging Archive. *The Cancer Imaging Archive (TCIA)* URL www.cancerimagingarchive.net.
- [7] Veeling, B. S., Linmans, J., Winkens, J., Cohen, T. & Welling, M. Rotation Equivariant CNNs for Digital Pathology (2018). URL <http://arxiv.org/abs/1806.03962>. ArXiv:1806.03962 [cs, stat].
- [8] Ehteshami Bejnordi, B. *et al.* Diagnostic Assessment of Deep Learning Algorithms for Detection of Lymph Node Metastases in Women With Breast Cancer. *JAMA* **318**, 2199 (2017). URL <http://jama.jamanetwork.com/article.aspx?doi=10.1001/jama.2017.14585>.
- [9] Zhu, Z., Wang, S.-H. & Zhang, Y.-D. A Survey of Convolutional Neural Network in Breast Cancer. *Computer Modeling in Engineering & Sciences* **136**, 2127–2172 (2023). URL <https://www.techscience.com/CMES/v136n3/51805>.
- [10] Goodfellow, I., Bengio, Y. & Courville, A. *Deep Learning* (MIT Press, 2016). <http://www.deeplearningbook.org>.
- [11] Xie, Y. *et al.* in *Deep Voting: A Robust Approach Toward Nucleus Localization in Microscopy Images* (eds Navab, N., Hornegger, J., Wells, W. M. & Frangi, A. F.) *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, Vol. 9351 374–382 (Springer International Publishing, Cham, 2015). URL http://link.springer.com/10.1007/978-3-319-24574-4_45. Series Title: Lecture Notes in Computer Science.
- [12] Sanyal, P., Mukherjee, T., Barui, S., Das, A. & Gangopadhyay, P. Artificial Intelligence in Cytopathology: A Neural Network to Identify Papillary Carcinoma on Thyroid Fine-Needle Aspiration Cytology Smears. *Journal of Pathology Informatics* **9**, 43 (2018). URL <https://linkinghub.elsevier.com/retrieve/pii/>

S215335392200356X.

- [13] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J. & Wojna, Z. Rethinking the Inception Architecture for Computer Vision 2818–2826 (2016).
- [14] Zeng, X., Chen, H., Luo, Y. & Ye, W. Automated Diabetic Retinopathy Detection Based on Binocular Siamese-Like Convolutional Neural Network. *IEEE Access* **7**, 30744–30753 (2019). URL <https://ieeexplore.ieee.org/document/8660434/>.
- [15] Ehrhart, I. C. *et al.* Coronary vascular and myocardial responses to carotid body stimulation in the dog. *The American Journal of Physiology* **229**, 754–760 (1975).
- [16] Wang, X. *et al.* in *ChestX-ray: Hospital-Scale Chest X-ray Database and Benchmarks on Weakly Supervised Classification and Localization of Common Thorax Diseases* (eds Lu, L., Wang, X., Carneiro, G. & Yang, L.) *Deep Learning and Convolutional Neural Networks for Medical Imaging and Clinical Informatics* 369–392 (Springer International Publishing, Cham, 2019). URL http://link.springer.com/10.1007/978-3-030-13969-8_18. Series Title: Advances in Computer Vision and Pattern Recognition.
- [17] Wang, L., Lin, Z. Q. & Wong, A. COVID-Net: a tailored deep convolutional neural network design for detection of COVID-19 cases from chest X-ray images. *Scientific Reports* **10**, 19549 (2020). URL <https://www.nature.com/articles/s41598-020-76550-z>.
- [18] Militante, N. V., Sammy V. a nd Dionisio & Sibbaluca, B. G. Pneumonia and COVID-19 Detection using Convolutional Neural Networks 1–6 (2020). URL <https://ieeexplore.ieee.org/document/9243290/>.
- [19] Weese, J. & Lorenz, C. Four challenges in medical image analysis from an industrial perspective. *Medical Image Analysis* **33**, 44–49 (2016). URL <https://linkinghub.elsevier.com/retrieve/pii/S1361841516300998>.
- [20] Japkowicz, N. & Stephen, S. The class imbalance problem: A systematic study1. *Intelligent Data Analysis* **6**, 429–449 (2002). URL <https://www.medra.org/servlet/aliasResolver?alias=iospress&doi=10.3233/IDA-2002-6504>.
- [21] Barandela, R., Sanchez, J., Garcia, V. & Rangel, E. Strategies for learning in class imbalance problems. *Pattern Recognition* **36**, 849–851 (2003). URL <https://linkinghub.elsevier.com/retrieve/pii/S0031320302002571>.
- [22] Maheshwari, D., Garcia-Zapirain, B. & Sierra-Sosa, D. Quantum Machine Learning Applications in the Biomedical Domain: A Systematic Review. *IEEE Access* **10**, 80463–80484 (2022). URL <https://ieeexplore.ieee.org/document/9844718/>.
- [23] Akhalwaya, I. Y. *et al.* Topological data analysis on noisy quantum computers (2024). URL <http://arxiv.org/abs/2209.09371>. ArXiv:2209.09371 [quant-ph].

- [24] Mishra, N. *et al.* in *Quantum Machine Learning: A Review and Current Status* (eds Sharma, N., Chakrabarti, A., Balas, V. E. & Martinovic, J.) *Data Management, Analytics and Innovation*, Vol. 1175 101–145 (Springer Singapore, Singapore, 2021). URL http://link.springer.com/10.1007/978-981-15-5619-7_8. Series Title: Advances in Intelligent Systems and Computing.
- [25] Zeguendry, A., Jarir, Z. & Quafafou, M. Quantum Machine Learning: A Review and Case Studies. *Entropy* **25**, 287 (2023). URL <https://www.mdpi.com/1099-4300/25/2/287>.
- [26] Menneer, T. & Narayanan, A. Quantum-inspired neural networks **95**, 27–30 (1995).
- [27] Rebentrost, P., Mohseni, M. & Lloyd, S. Quantum support vector machine for big data classification. *Physical Review Letters* **113**, 130503 (2014). URL <http://arxiv.org/abs/1307.0471>. ArXiv:1307.0471 [quant-ph].
- [28] Harrow, A. W., Hassidim, A. & Lloyd, S. Quantum Algorithm for Linear Systems of Equations. *Physical Review Letters* **103**, 150502 (2009). URL <https://link.aps.org/doi/10.1103/PhysRevLett.103.150502>.
- [29] Wiebe, N., Kapoor, A. & Svore, K. Quantum Algorithms for Nearest-Neighbor Methods for Supervised and Unsupervised Learning (2014). URL <http://arxiv.org/abs/1401.2142>. ArXiv:1401.2142 [quant-ph].
- [30] Zhou, N.-R., Liu, X.-X., Chen, Y.-L. & Du, N.-S. Quantum K-Nearest-Neighbor Image Classification Algorithm Based on K-L Transform. *International Journal of Theoretical Physics* **60**, 1209–1224 (2021). URL <https://link.springer.com/10.1007/s10773-021-04747-7>.
- [31] Schuld, M., Sinayskiy, I. & Petruccione, F. Prediction by linear regression on a quantum computer. *Physical Review A* **94**, 022342 (2016). URL <https://link.aps.org/doi/10.1103/PhysRevA.94.022342>.
- [32] Lloyd, S. & Weedbrook, C. Quantum Generative Adversarial Learning. *Physical Review Letters* **121**, 040502 (2018). URL <https://link.aps.org/doi/10.1103/PhysRevLett.121.040502>.
- [33] Dallaire-Demers, P.-L. & Killoran, N. Quantum generative adversarial networks. *Physical Review A* **98**, 012324 (2018). URL <http://arxiv.org/abs/1804.08641>. ArXiv:1804.08641 [quant-ph].
- [34] Situ, H., He, Z., Wang, Y., Li, L. & Zheng, S. Quantum generative adversarial network for generating discrete distribution. *Information Sciences* **538**, 193–208 (2020). URL <https://linkinghub.elsevier.com/retrieve/pii/S0020025520305545>.

- [35] Chakrabarti, S., Huang, Y., Li, T., Feizi, S. & Wu, X. Quantum Wasserstein Generative Adversarial Networks (2019). URL <http://arxiv.org/abs/1911.00111>. ArXiv:1911.00111 [quant-ph].
- [36] Huang, H.-L. *et al.* Experimental Quantum Generative Adversarial Networks for Image Generation. *Physical Review Applied* **16**, 024051 (2021). URL <https://link.aps.org/doi/10.1103/PhysRevApplied.16.024051>.
- [37] Kieferová, M. & Wiebe, N. Tomography and generative training with quantum Boltzmann machines. *Physical Review A* **96**, 062327 (2017). URL <https://link.aps.org/doi/10.1103/PhysRevA.96.062327>.
- [38] Amin, M. H., Andriyash, E., Rolfe, J., Kulchytskyy, B. & Melko, R. Quantum Boltzmann Machine. *Physical Review X* **8**, 021050 (2018). URL <http://arxiv.org/abs/1601.02036>. ArXiv:1601.02036 [quant-ph].
- [39] Shorten, C. & Khoshgoftaar, T. M. A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data* **6**, 60 (2019). URL <https://journalofbigdata.springeropen.com/articles/10.1186/s40537-019-0197-0>.
- [40] Inoue, H. Data Augmentation by Pairing Samples for Images Classification (2018). URL <http://arxiv.org/abs/1801.02929>. ArXiv:1801.02929 [cs, stat].
- [41] Summers, C. & Dinneen, M. J. Improved Mixed-Example Data Augmentation (2019). URL <http://arxiv.org/abs/1805.11272>. ArXiv:1805.11272 [cs].
- [42] Zhong, Z., Zheng, L., Kang, G., Li, S. & Yang, Y. Random Erasing Data Augmentation (2017). URL <http://arxiv.org/abs/1708.04896>. ArXiv:1708.04896 [cs].
- [43] DeVries, T. & Taylor, G. W. Improved Regularization of Convolutional Neural Networks with Cutout (2017). URL <http://arxiv.org/abs/1708.04552>. ArXiv:1708.04552 [cs].
- [44] Mikolajczyk, A. & Grochowski, M. Data augmentation for improving deep learning in image classification problem 117–122 (2018). URL <https://ieeexplore.ieee.org/document/8388338/>.
- [45] Xu, T. *et al.* AttnGAN: Fine-Grained Text to Image Generation with Attentional Generative Adversarial Networks (2017). URL <http://arxiv.org/abs/1711.10485>. ArXiv:1711.10485 [cs].
- [46] Ledig, C. *et al.* Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network (2017). URL <http://arxiv.org/abs/1609.04802>. ArXiv:1609.04802 [cs, stat].

- [47] Zhu, J.-Y., Park, T., Isola, P. & Efros, A. A. Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks (2020). URL <http://arxiv.org/abs/1703.10593>. ArXiv:1703.10593 [cs].
- [48] Frid-Adar, M. *et al.* GAN-based Synthetic Medical Image Augmentation for increased CNN Performance in Liver Lesion Classification. *Neurocomputing* **321**, 321–331 (2018). URL <http://arxiv.org/abs/1803.01229>. ArXiv:1803.01229 [cs, stat].
- [49] Wolterink, J. M. *et al.* Deep MR to CT Synthesis using Unpaired Data (2017). URL <http://arxiv.org/abs/1708.01155>. ArXiv:1708.01155 [cs].
- [50] Hiasa, Y. *et al.* Cross-modality image synthesis from unpaired data using CycleGAN: Effects of gradient consistency loss and training data size (2018). URL <http://arxiv.org/abs/1803.06629>. ArXiv:1803.06629 [cs].
- [51] Mok, T. C. W. & Chung, A. C. S. Learning Data Augmentation for Brain Tumor Segmentation with Coarse-to-Fine Generative Adversarial Networks **11383**, 70–80 (2019). URL <http://arxiv.org/abs/1805.11291>. ArXiv:1805.11291 [cs].
- [52] Houssein, E. H., Abohashima, Z., Elhoseny, M. & Mohamed, W. M. Machine learning in the quantum realm: The state-of-the-art, challenges, and future vision. *Expert Systems with Applications* **194**, 116512 (2022). URL <https://linkinghub.elsevier.com/retrieve/pii/S0957417422000136>.
- [53] Dong, D. & Petersen, I. R. Quantum estimation, control and learning: Opportunities and challenges. *Annual Reviews in Control* **54**, 243–251 (2022). URL <https://linkinghub.elsevier.com/retrieve/pii/S1367578822000281>.
- [54] Matic, A., Monnet, M., Lorenz, J. M., Schachtner, B. & Messerer, T. Quantum-classical convolutional neural networks in radiological image classification (2022). URL <https://arxiv.org/abs/2204.12390>. Version Number: 2.
- [55] Mathur, N. *et al.* Medical image classification via quantum neural networks (2022). URL <http://arxiv.org/abs/2109.01831>. ArXiv:2109.01831 [quant-ph].
- [56] Khan, T. M. & Robles-Kelly, A. Machine Learning: Quantum vs Classical. *IEEE Access* **8**, 219275–219294 (2020). URL <https://ieeexplore.ieee.org/document/9274431/>.
- [57] Sari, C. A., Sari, W. S. & Rahmalan, H. A Combination of K-Means and Fuzzy C-Means for Brain Tumor Identification. *Scientific Journal of Informatics* **8**, 76–83 (2021). URL <https://journal.unnes.ac.id/nju/index.php/sji/article/view/29357>.
- [58] Zhou, H., Schaefer, G., Sadka, A. H. & Celebi, M. E. Anisotropic Mean Shift Based Fuzzy C-Means Segmentation of Dermoscopy Images. *IEEE Journal of Selected Topics in Signal Processing* **3**, 26–34 (2009). URL <http://ieeexplore.ieee.org>.

- org/document/4786538/.
- [59] Mittapalli, P. S. & Kande, G. B. Segmentation of optic disk and optic cup from digital fundus images for the assessment of glaucoma. *Biomedical Signal Processing and Control* **24**, 34–46 (2016). URL <https://linkinghub.elsevier.com/retrieve/pii/S1746809415001512>.
 - [60] Codella, N. *et al.* Skin Lesion Analysis Toward Melanoma Detection 2018: A Challenge Hosted by the International Skin Imaging Collaboration (ISIC) (2019). URL <http://arxiv.org/abs/1902.03368>. ArXiv:1902.03368 [cs].
 - [61] Tschandl, P., Rosendahl, C. & Kittler, H. The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *Scientific Data* **5**, 180161 (2018). URL <https://www.nature.com/articles/sdata2018161>.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- [DataAugmentationwithQGANgabstract.png](#)