CHALMERS
UNIVERSITY OF TECHNOLOGY

UNIVERSITY OF GOTHENBURG

Improving echocardiogram view classification using diffusion models

Master's thesis in Computer science and engineering

LUIS AREVALO

ANOUKA RANBY

Department of Computer Science and Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
UNIVERSITY OF GOTHENBURG
Gothenburg, Sweden 2023

Master's thesis 2023

Improving echocardiogram view classification using diffusion models

LUIS AREVALO
ANOUKA RANBY

UNIVERSITY OF
GOTHENBURG

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Computer Science and Engineering
Chalmers University of Technology
University of Gothenburg
Gothenburg, Sweden 2023

Improving echocardiogram view classification using diffusion models
LUIS AREVALO
ANOUKA RANBY

Supervisor: Yinan Yu, Department of Computer Science and Engineering
Advisor: Charlotte von Numers, AstraZeneca
Examiner: Richard Johansson, Department of Computer Science and Engineering

Master's Thesis 2023
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg
SE-412 96 Gothenburg
Telephone +46 31 772 1000

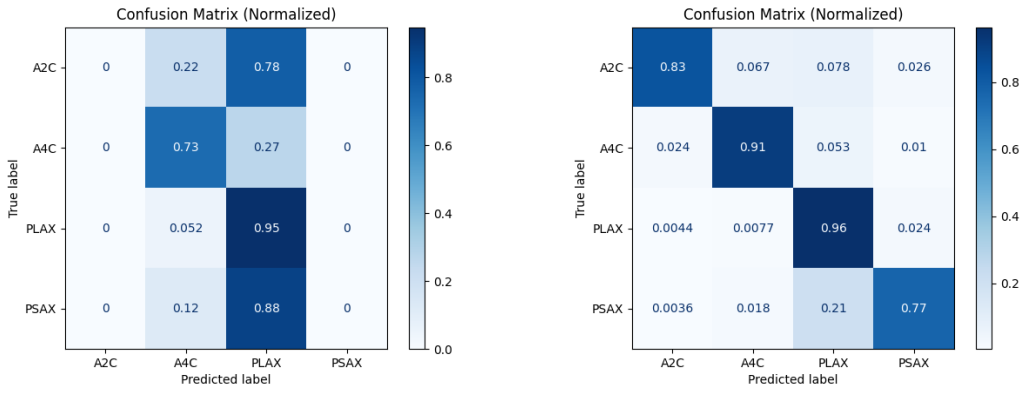
Typeset in LaTeX
Gothenburg, Sweden 2023

Improving echocardiogram view classification using diffusion models
LUIS AREVALO
ANOUKA RANBY
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg

Abstract

In the field of medical science datasets are often highly imbalanced, where rare datapoints are of high importance. This study aims to explore the usage of synthetic datasets to improve the classification of echocardiogram views. We join together different echocardiogram datasets (EchoNet-LVH, EchoNet-Dynamic, TMED-2) to form a custom imbalanced dataset (N=38,915) including Apical two-chamber (A2C), Apical four-chamber (A4C), Parasternal long axis (PLAX), and Parasternal short axis (PSAX) views. We study the results of training a diffusion model on differently sized subsets of this dataset. For each size of real subset available, we train two echocardiogram view classifiers on (i) the real data subset and (ii) on the synthetic subset, generated from training on the real subset.

Our results show that synthetic data can be used to improve echocardiogram view classification performance. Specifically we prove that the classification performance of minority classes is significantly improved when the imbalanced dataset is limited. The percentage of images that get correctly classified, for the minority classes A2C and PSAX, increases from 0 to 0.83 and from 0 to 0.77 respectively, when the real subset is limited to 1,000 examples.



(a) Trained on real data.

(b) Trained on synthetic data.

Figure 0.1: Normalized Confusion Matrix: VGG16 echocardiogram view classifier.

Keywords: Computer, science, computer science, engineering, project, artificial intelligence, machine learning, deep neural networks, diffusion models, synthetic data, echocardiogram classification.

Acknowledgements

We would like to express our deepest gratitude to both Charlotte von Numers and Yinan Yu for their unwavering feedback, support and encouragement throughout this entire project. Your enthusiasm for this work and its potential was palpable in every discussion we had, and served as a huge source of inspiration and motivation. The expertise, insights, and input you brought to our conversations have been invaluable assets without which this work could not have been carried out.

Luis Arevalo & Anouka Ranby, Gothenburg, 2023-06-27

Contents

List of Figures

List of Tables

List of Acronyms

TTE Transthoracic Echocardiogram

A2C Apical two-chamber

A4C Apical four-chamber

A5C Apical five-chamber

PLAX Parasternal long axis

PSAX Parasternal short axis

IVC Inferior vena cava

ED End diastole

ES End Systole

CAMUS Cardiac Acquisitions for Multi-structure Ultrasound Segmentation

TMED2 Tufts Medical Echocardiogram Dataset Version 2

NIH National Institute of Health

KS Kolmogorov-Smirnov

DARTS Differentiable architecture search

GAN Generative Adversarial Network

CNN Convolutional Neural Network

AUC Area under the Curve

PRC Precision-Recall Curve

AUPRC Area Under the Precision-Recall Curve

FID Fréchet inception distance

GPU Graphics Processing Unit

PCA Principal Component Analysis

t-SNE t-Distributed Stochastic Neighbor Embedding

1

Introduction

Transthoracic Echocardiogram (TTE) are ultrasound images or videos of the heart (see Figure 1.1) that can be collected in 2D and 3D, where the first is the common format for heart assessment. They are frequently used for cardiac condition evaluation in healthcare and clinical trials. As part of their clinical trials, AstraZeneca currently collects echocardiogram data from their subjects. Hereafter, when referring to echocardiograms, we implicitly mean TTE 2D images.

This data collection process entails the subject coming to AstraZeneca's and partnering facilities to have a sonographer record echocardiogram video sequences of their heart while lying down. Importantly, these sequences must be long enough to show at least a few heartbeats, typically a few seconds. Moreover, the echocardiogram records different views of the heart depending on the exact position and angle of the probe on the subject's chest. In order to properly conduct their studies, AstraZeneca requires these ultrasound images to be of good enough quality and to cover a certain variety of views. Recordings are therefore screened and verified by a sonographer before being further used for health evaluations or in clinical trials.

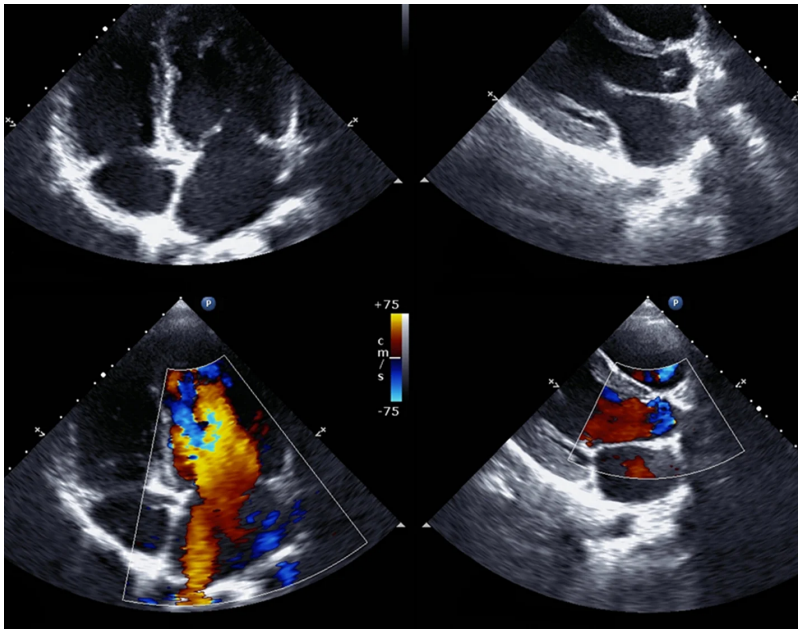


Figure 1.1: Echocardiogram images.

1.1 Problem Description

The activities of recording echocardiograms and validating quality and views captured are generally conducted at separate instances and times. If there are faults in quality, or if key views are missing, the subject may need to be called back for an additional echocardiogram recording. The risk of not having immediate quality and view detection measures implemented at the initial echocardiogram recording session is patients' drop out of clinical trials, inefficient use of resources and high costs. Figure 1.2 shows an estimated process for the collection of echocardiograms as part of clinical trials with challenges highlighted in red.

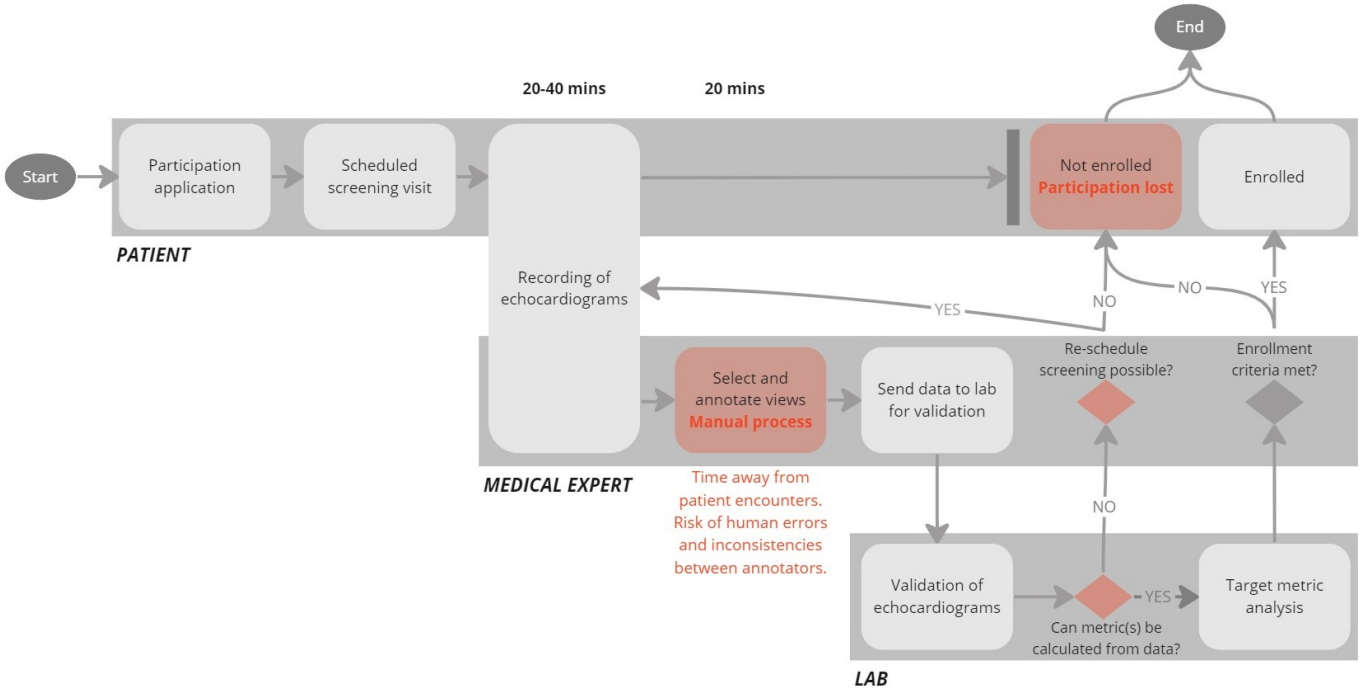


Figure 1.2: Example workflow for clinical trials.

Acquiring high volumes of medical data of premium quality is a key problem in medical sciences. Data points of primary interests are those representing rare conditions, seen as minority classes, resulting in most imbalanced datasets. In addition, medical data must be safeguarded with strict regulatory frameworks to protect the privacy of patients and their health information.

Partially automating the echocardiogram screening process by verifying views, which is currently performed manually, could lead to expedited subject processing, more efficient use of resources, and potentially even to faster patient diagnoses. The outcome of training a classification model is closely linked to the quantity and quality of the data it is trained on. Current limitations on the availability of annotated echocardiograms is a factor that could hinder classifier performance. Extending available datasets by collecting real echocardiograms is difficult, in part due to patient privacy concerns, but it is also time consuming. Additionally, the annotation

process is expensive, as it requires highly trained professionals, whose time is especially valuable.

The lack of annotated subclass-level view data also constitutes an impediment to the training of a classifier. Given the expensive and time-consuming nature of echocardiogram data acquisition and annotation, a promising alternative is instead generating synthetic data on which to train the classifier. If such a model were to train on and synthesize echocardiogram views for which data is currently readily available, and if these synthetic images can be shown to improve performance of a view classifier, then it would constitute a proof-of-concept that synthetic echocardiograms can be used to complement real training data.

1.2 Project Aim

This project aims to tackle the issue of view detection specifically, as opposed to the echocardiogram quality aspect. We use a generative diffusion model to generate synthetic echocardiograms with a diversity similar to that of the original training data, such that they become indistinguishable from real echocardiograms to a human expert. These synthetic echocardiograms are then used in an attempt to improve the performance of a view classifier on real echocardiograms. Our hope is to show that synthetic data can be used to overcome challenges related to imbalanced datasets, and to address the data privacy concerns by utilising anonymous synthetic echocardiograms.

1.3 Research Questions

Based on the aim of the project, we formulate two research questions:

1. Is it possible to improve view classification performance of real echocardiograms with synthetic data?

2. Can diffusion models generate synthetic echocardiograms that look realistic enough to become indistinguishable from real echocardiograms to a human expert?

Firstly, a diffusion model is trained to generate synthetic echocardiograms with, ideally, good enough quality and diversity that they are indistinguishable from real echocardiograms to a human expert. Secondly, yet primarily, these synthetic images are used to train a classifier to ultimately try to improve the performance of classifying real echocardiogram images.

Successful results could prove helpful in the medical sciences, as they can potentially lead to saved costs, time and resources, and potentially faster diagnoses for patients suffering from heart conditions. In addition, it could prove helpful in overcoming challenges with medical datasets, such as imbalance and data privacy concerns.

1.4 Approach

Several studies have attempted to automate echocardiogram view detection by a variety of methods, machine learning models being particularly prevalent in the recent literature [1] [2] [3] [4] [5]. These studies vary in datasets, classification algorithms, and classification performance. However, a common theme among most of the studies conducted on this topic is the focus on a particular set of echocardiogram views, namely A2C, A4C, PLAX and PSAX. These categorical views can be referred to as the highest hierarchy views since they can be further divided into subclasses of views.

The focus on these top-level views is due in part to their common clinical use [4]. This clinical relevance might also explain why most publicly available echocardiogram datasets focus on these views in particular. The lack of annotated data on views at the subclass level, however, might explain why classifiers able to distinguish between different subclasses of views are largely absent from the literature.

Synthetic data generation can be used to complement limited medical datasets in a way that guarantees that privacy regulations are not violated, and is also a less labour-intensive data acquisition approach. However, generating medical data is difficult due to the complexity of organs and classification performance often depends on subtle changes in images. Recently, generative models have emerged within the field of deep learning and computer vision as an efficient mean to generate synthetic images. They are trained to generate new images similar to the ones they were trained on by learning the underlying distribution of the training data.

Generative Adversarial Networks (GANs) have been state-of-the-art in image synthesis in general and within the medical field specifically [6]. With extended research and usage, GANs have also become known to suffer from mode collapse, unstable training behavior, difficulty in capturing true diversity [6], vanishing gradient issues, along with non-convergence and hyperparameter sensitivity [7].

Diffusion models have lately gained traction for image synthesis as proving capable of generating high-quality images [8] [9] [10] [11]. Recent studies shows that diffusion models outperform state-of-the-art GANs in image generation tasks, both in general [12] and in the medical field [13]. This holds true whether they are conditioned on specific labels [14] or left unconditioned [15].

The research undertaken revealed a lack of knowledge shared regarding the application of diffusion models for the synthesis of echocardiograms conditioned on particular views.

1.5 Limitations

Certain limitations adhere in the context of this thesis, each outlined in this section.

Focus on diffusion models: This research focuses exclusively on diffusion models as the chosen type of generative model architecture. Although diffusion models have

demonstrated their efficacy in various applications, restricting the investigation to a single type of generative model may limit the breadth of insights and understanding that could be gained from exploring alternative models or methods for extending datasets.

Echocardiogram views: There is a limited number of echocardiogram views explored in this project due to views present in the datasets used. This means that the analysis and findings are restricted to the data and classes available. Consequently, the conclusions drawn from this research and their generalizability to other cardiac views should be studied further.

Bias in data: Data used in this study may contain biases stemming from various factors, such as: sample selection, class label imbalances, potential socioeconomic backgrounds of admitted patients, clinical routines, and the equipment used to acquire data. If these biases exist in the training data, they will likely influence the knowledge gained by diffusion models during training. As a result, bias found in the training data may be translated to synthetic images generated. Additionally, when classification models, trained on biased data, are used at inference with data from a different distribution, they may fail to perform effectively.

Limited usage rights for non-commercial purposes: Usage rights associated with the data involved in this research are restricted to non-commercial use only. Consequently, the findings and outcomes of this study cannot be directly utilized or applied in commercial endeavors or by AstraZeneca. However, despite this limitation, the research still holds value as a proof-of-concept, showcasing the feasibility and potential benefits of employing generative models in the medical field.

Potential risk of synthetic images generated: Generative models learn patterns found in real echocardiograms to generate new images that mimic the real ones. The risk of this process is that synthetic images could include close copies of original images and therefore become a patient privacy concern. There is also a possibility that generated images may contain anatomical inaccuracies or inconsistencies, which could lead to incorrect interpretations, misdiagnoses, or ineffective treatment plans if used improperly.

It is important to emphasize that the purpose of this research is not to claim medical correctness but rather to explore the potential of diffusion models in generating synthetic echocardiograms to improve the performance of a downstream view classification model.

1.6 Ethical Considerations

Ethical concerns related to the use of synthetic medical data is closely related to the impacts it could have in the future and specifically for patients' health and life. Here are some immediate concerns, although not an exhaustive list:

- Regulatory frameworks: Integration and implementation of synthetic data in the healthcare sector requires careful consideration and governance of regulatory frameworks, such as data protection, accountability and validation

standards, to ensure reliable and ethical use of synthetic data in healthcare applications.

- Patient privacy concerns: By substituting actual patient data with synthetic counterparts, healthcare organizations can minimize the risk of privacy breaches, distribute and repurpose data more. Nevertheless, caution must be exercised in the event of biased synthetic data being employed for decision-making and diagnoses, as this could result in erroneous or discriminatory outcomes. Similarly, the generation of images using real patient data to produce potentially misleading or inaccurate representations demands careful scrutiny to uphold the standards and ensure accuracy and integrity of medical imaging.
- Environmental footprint: Environmental consequences stemming from the computational requirements of running deep learning models raise concerns regarding their negative impact. Lower-income communities may experience heightened vulnerability due to limited resources and capacity to adapt to environmental damages stemming from activities leaving a negative footprint. This highlights the importance of environmentally responsible model developments and usage of resources.

Overall, while generative models have the potential to generate realistic-looking medical images, there is a need for caution regarding usage of synthetically generated contents. Ongoing validation, expert input, and integration with clinical expertise are necessary to ensure that the generated images are used appropriately for medical applications.

1.7 Thesis Outline

Subsequent section (2) of this report is structured to provide a view of most recent studies published in relation to state-of-the-art echocardiogram view classification and synthetic data generation with diffusion models. Following that is Section 3 with an in depth presentation of the image and video datasets used, along with the preprocessing techniques applied. Section 4 provides an overview of the project pipeline, evaluation approaches applied per diffusion and classification tasks and the explanation to experiments carried out. Experiment headlines are organised to have a corresponding headline in the Results section (5) to easily connect the experiment design to its corresponding outcome. Finally, we present a concluding section (6) answering our research questions, stating our contributions and suggestions to future work.

2

Theory

This section highlights recent studies of echocardiogram view classification using state-of-the-art deep learning networks. Specifically we highlight the potential benefits of using synthetic datasets to overcome challenges of typically scarce medical datasets. Lastly, the concept of diffusion models is introduced, explaining how it works and can be used to generate images that mimic original data distributions.

2.1 View Classification

The process of classifying echocardiograms views (including A2C, A4C, PLAX, PSAX) has been attempted and documented in the research. However, few studies present a successful classification of larger collection of views maintained in small datasets. We will briefly outline most recent studies that attempt to solve echocardiogram view classification using state-of-the-art deep learning architectures. A more comprehensive overview of echocardiogram view classification research can be found in [16].

[17] proposes a real-time system including view classification for four views (among them A4C and PLAX). The system uses a self-supervised echocardiogram specific representation built with a small MobileNetV2 autoencoder trained on EchoNet-Dynamic dataset (112x112 image resolution). The encoder is modified by having fewer layers and a fuzzy pooling layer replacing the average pooling layer to better handle image speckle noise (grainy noise due to interference of ultrasound waves). A light-weight multi-head model then identifies unique views and dismisses low quality samples. View classification performance results reported 0.985 Area under the Curve (AUC) as compared to 0.963 with DenseNet161, 0.958 when using VGG16 and 0.951 with ResNet18, all with echo-specific weights.

Another real-time solution for view classification was presented in [5] identifying 14 different echocardiogram views using neural architecture search, more specifically Differentiable architecture search (DARTS), to automate the model architecture design process. With focus on a minimized network architecture with maximal prediction accuracy, they also aimed to include subclasses of echocardiogram views and assess the model performance based on different input image resolutions and training dataset sizes. PLAX, PSAX and A4C views were the majority classes. Macro average Precision and Recall (average overall metrics of per-view performance) and F1-Score (harmonic mean of Precision and Recall) was used as evaluation metrics.

Two different DARTS model architectures were evaluated and compared to VGG16, ResNet18 and DenseNet201. Results showed that accuracy performance converged between the larger input image resolutions (96x96, 128x128) for all five models as compared to the smaller resolutions (32x32, 64x64) where the drop was coming from specific views such as A4C right ventricle. The best performing model was their 2-cell-DARTS on 128x128 images having 0.951 in F1 score, where it was most challenging to detect Apical five-chamber (A5C) view as being mistaken for other apical views while views with distinct characteristics were easier to distinguish. Regarding training dataset size, the larger the better accuracy performance generally.

Identification of views is often done by classifying individual frames collected from video sequences. Using deep neural network architecture with the ability to consider cardiac movements may result in advancements in performances. [18] was Inspired by recent work in the field of human action recognition, processing spatial and temporal factors in videos. A total of 8,732 videos were utilised and had 14 classes represented unequally (including A2C, A4C, PLAX) and were labelled by experts. The first 40 frames of each video were resized to 299x299 or 224x224 pixels and normalized to a value between 0 and 1 at training time. The study compared single-frame 2D Convolutional Neural Network (CNN) (Xception, DenseNet121, ResNet, InceptionV3, VGG16), multi-frame time-distributed CNN (TD Xception), spatio-temporal 3D CNN (C3D, Inception3D). The error rate of the latter two was half of the best performing 2D CNN, holding promise to making use of information from the cardiac cycle as oppose to isolated frames.

Three state-of-the-art architectures, VGG16, DenseNet and ResNet, were exploited to use knowledge distillation to build an ensemble of lightweight deep learning architectures [19]. The task was to classify 12 echocardiogram views (from Apical, Parasternal, Subcostal, Suprasternal windows) from an imbalanced dataset of 16,612 videos, equal to a 807,908 frames resized to 80x80 grayscale pixels, labeled by a cardiologist. The combination of lightweight models achieved comparable performance to an ensemble of the original architectures. Among the contributions were a lightweight system that achieved a six times speedup at inference by utilising only around 1% of the original model parameters, a method beneficial to mobile application and real-time scenarios.

With the ambition to use minimal data while maximizing the performance accuracy, a deep convolutional neural network inspired from VGG16 was used to classify 15 echocardiogram views [3]. A datasets of 267 labeled echocardiograms (videos and images), resulting in a dataset with 223,787 grayscale images with 60x80 pixel resolution scaled to values between 0 and 1. The model achieved 0.964 in F1-score on videos and 0.904 on still images on average. Minority classes and views with distinct features (m-mode) had lowest performance.

These most recent studies have attempted to classify multiple views available in echocardiography. However, the definition of views is somewhat inconsistent, for example, considering Continuous Wave Doppler, Pulsed Wave Doppler and Motion-mode the same view, or Aortic Vale, Mitral Vale, Left Ventricle and Right Ventricular Annulus were all considered Motion-mode [3]. Overall, the problem of very small

dataset samples are not represented, while the problem of seeing under-represented classes seem to be persistent across studies.

Repeatedly seen in studies mentioned above are VGG16 [20], ResNet18 [21], DenseNet121 [22] and InceptionV3 [23]. These are all state-of-the-art CNN that will be further explored in the classification task of this project. Each has its own characteristics, laid out in detail in their respective research papers, yet they all share a key component as they include convolutional layers. These make up the fundamental building blocks that capture spatial patterns and features from input images. This allows the networks to learn hierarchical representations of the data, starting from low-level features to high-level semantic information.

2.2 Image Synthesis

Large deep neural networks outlined in Section 2.1 are know to be data hungry. As large volumes of annotated high-quality medical data is limited, an approach to extending crucial datasets is data augmentation by artificially manufacturing data. This is not only beneficial from a resource perspective, but synthetic data may also allow for wider distributions of medical data without violating patient privacy, since the data being distributed is not real patient data. Re-purposing datasets and reproducing medical research results would also become easier with the usage of synthetic data.

Synthetic data can be created through the use of generative algorithms, producing data of desirable volumes that can be customized to fit specific needs [24]. In the medical field, synthesized data can cover the diversity and quality of the datasets it was trained on, and emulate underrepresented classes to improve the adaptability of machine learning models at inference. For example, datasets can be balanced by generating data conditioned on specific classes.

There are limitations and risks associated with the usage of synthetic data in the medical field. Examples of such scenarios are unknown or new conditions, such as COVID-19, that may not have been part of the data a generative model was trained on [25]. Synthetic data can also be used maliciously by creating contents that may be inaccurate or misleading, such as anatomically incorrect images.

2.3 Diffusion Models

Modeling of complex data distributions can be achieved through the utilization of a Markov chain. This approach gradually transforms a well-behaved distribution into the desired complex target distribution of the data by means of incremental steps [8]. The underlying principle involves the progressive diffusion of the initial images through the addition of small amounts of Gaussian noise at each step, over multiple iterations. At the completion of this Markov chain process, the resulting images conform to the well-behaved distribution, typically a zero-mean identity-covariance Gaussian distribution $\mathcal{N}(\mathbf{0}, \mathbf{I})$.

By capturing the input-output image pairs at each stage of the diffusion process, a model can be trained to effectively reverse this diffusion process. Specifically, the model is trained to accurately reconstruct the input image based on the corresponding output at each time step, thereby learning to effectively denoise the images step by step. Finally, running the learned reverse diffusion process on randomly sampled noise results in generated images that mimic the distribution of the original data [8].

In [8], the Gaussian forward diffusion process is defined as:

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t;\ \sqrt{1-\beta_t}\mathbf{x}_{t-1},\ \beta_t\mathbf{I}),$$

where $\mathbf{x}_{t-1}$ and $\mathbf{x}_t$ are the input and output images of the diffusion step, respectively, and β_t is the diffusion rate, which can vary for different time steps.

Similarly, the learnable reverse diffusion process is:

$$p(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1};, \mathbf{f}_\mu(\mathbf{x}_t, t),\ \mathbf{f}_\Sigma(\mathbf{x}_t, t)),$$

where $\mathbf{f}_\mu(\mathbf{x}_t, t)$ and $\mathbf{f}_\Sigma(\mathbf{x}_t, t)$ are functions defining the mean and covariance of the reverse diffusion process. These functions are defined as neural networks whose parameters can then be learned.

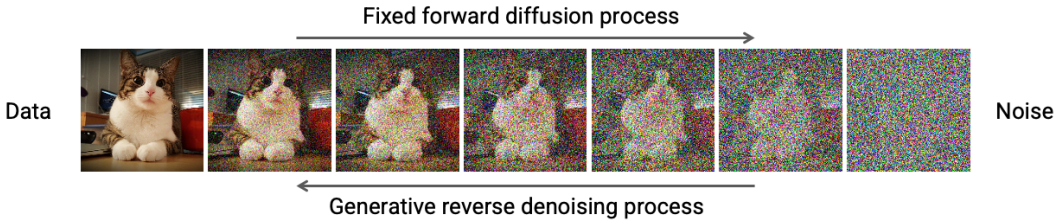


Figure 2.1: Image diffusion process.

At their inception, diffusion models showed good results with simpler toy data, but their formulations entailed some complexities that may have prevented them from taking off right at the outset. With a few clever simplifications, however, it was shown that diffusion models could achieve remarkable results in image generation on more complex real-world data [10]. Diffusion rates β_t and covariances $\Sigma(\mathbf{x_t}, t)$ can potentially be learned, but setting them at fixed values can prove a beneficial gain of simplicity at the cost of flexibility. Similarly, [10] introduces a reparameterization trick:

Let:

$$\alpha_t = (1-\beta_t), \quad \text{and} \quad \bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s.$$

Then:

$$q(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t;\, \sqrt{\bar{\alpha}_t}\mathbf{x}_0,\, (1-\bar{\alpha}_t)\mathbf{I}),$$

which is a simplified formulation for sampling x_t at an arbitrary time step directly from x_0. This can be further reparameterized to:

$$\mathbf{x}_t(\mathbf{x}_0,\, \boldsymbol{\epsilon}) = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{(1-\bar{\alpha}_t)}\boldsymbol{\epsilon}, \quad \text{where} \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0},\, \mathbf{I})$$

Using this reparameterization, they arrive to a simplified training objective:

$$\mathbb{E}_{\mathbf{x}_0,\, \boldsymbol{\epsilon}} = \left[\frac{\beta_t^2}{2\sigma_t^2\alpha_t(1-\bar{\alpha}_t)}||\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon},\, t)||^2\right],$$

where $\sigma_t^2\mathbf{I}$ is the untrained time dependent covariance of the reverse diffusion process (normally set to $\sigma_t^2 = \beta_t$), $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0},\, \mathbf{I})$, and $\boldsymbol{\epsilon}_\theta$ is a function approximator that learns to predict $\boldsymbol{\epsilon}$ from $\mathbf{x}_t$.

Further research into diffusion models re-established the importance of learning variances in the reverse diffusion process [11]. This, however, necessitated modifying the simplified loss function [10] proposed in earlier work, as it provided no learning signal for the variance. A hybrid loss function was designed, which allowed learning both means and variances of the reverse diffusion process with Improved Denoising Diffusion Probabilistic Models (Improved Diffusion) [11]. Learning variances in addition to means led to higher quality sampled images.

Yet another improvement to diffusion models came in the form of the cosine noise schedule, introduced as an alternative to the default linear noise schedule [11]. The linear noise schedule, which dictates how β_t evolved throughout diffusion time steps, was found to be suboptimal for low resolution images (32x32 and 64x64). In essence, a linear noise schedule destroys the image information too quickly to make the end of the forward diffusion process informative during training. In contrast, a cosine noise schedule was found to destroy the information contained within training images more gradually, which in turn proved to work better with low resolution images. Use of this cosine noise schedule led to lower FID than those achieved with a linear noise schedule, which would indicate higher quality in generated samples.

Figure 2.2: Forward diffusion process using linear (top) and cosine (bottom) noise schedules at evenly spaced intervals [11].

Since their introduction in [8], and especially after many useful improvements arising from research in the area [10], [11], [26], diffusion models have exploded in popularity and have been used to great effect for a variety of tasks, such as text-to-image generation, image inpainting, to name a few. A few notable examples of trained diffusion models include Dall-E 2, Stable Diffusion, and Midjourney.

(a) Dall-E 2 (b) Stable Diffusion (c) Midjourney

Figure 2.3: Example images from notable diffusion models

It was demonstrated in "Diffusion Probabilistic Models beat GANs on Medical Images" (Medfusion) the potential of diffusion models applied specifically in the context of medical imaging [13]. Diffusion models have exhibited outstanding performance when training on different datasets comprising ophthalmological, radiological, and histological data, surpassing other models like Generative Adversarial Network (GAN) across multiple evaluation metrics used to assess image sample quality and diversity (e.g., FID, Precision, Recall). Notably, diffusion models have shown a distinct advantage over GANs in that they do not generate the typical artifacts commonly observed in GAN-generated images. This characteristic presents an advantage for diffusion models when it comes to utilizing their generated samples as training data for machine learning models that are expected to perform well on real data.

In this thesis work, we aim to continue this line of research using echocardiogram images to train several diffusion models, and leveraging the generated synthetic data for training an echocardiogram view classifier.

3

Data

In this project, five distinct echocardiogram datasets were employed, each comprising a composite of images and videos featuring A2C, A4C, PLAX, and PSAX views of the heart at varying states. The TMED2, EchoNet-LVH and EchoNet-Dynamic datasets are used for training, tuning and validation of diffusion and classification models. The CAMUS and Unity datasets are solely used for testing classification performance under domain shift.

To enable subsequent modeling, each dataset underwent a preprocessing procedure tailored to its inherent characteristics, resulting in a standardized set of images with uniform formatting. This section outlines the salient attributes of echocardiograms, including their intricate nature, describes each dataset, and provides a comprehensive account of the corresponding preprocessing methodologies for images and videos.

3.1 Echocardiograms

Standard echocardiogram studies usually involve recording multiple videos (50-100) and images of one heart at varied angles and positions using different techniques. Echocardiogram images are complex due to the intricate anatomy and physiology of the heart. The heart is a dynamic organ, and the different views captured during an echocardiogram can provide different insights into its structure and function. The complexity in acquiring echocardiograms and the image variability itself can make it challenging to accurately classify the correct view of the obtained image.

Factors that have impact on the complexity of echocardiogram images include the patient's heart rate, the patient's body, and the presence of any obstructions, such as tissues. Additionally, the various imaging machines and techniques used to capture echocardiograms and the experience of the operator can increase this complexity.

The area of the echocardioagram that displays the ultrasound image of the heart is referred to as the cone area, based on its shape. Depending on the machine settings used, the size and magnification of the cone area can vary. Echocardiograms usually have overlaying information such as view labels, meta data (text related to the patient) and electrocardiographic signal curves indicating the heart beat. They can also contain overlaying coloring generated from a Doppler echocardiogram, which can help determine the speed and direction of blood flow.

Datasets used were originally video sequences capturing a couple of hearts beats over

a few seconds, or still images capturing the heart at different states of cardiac cycle. The following sections detail video and image specific datasets and the processing applied to each.

3.2 Image Preprocessing

Images from datasets encompass different complexities and characteristics. They all had varied file formats, color channels, image dimensions, pixel intensities and outlier data. Due to the variability of echocardiograms there are a number of preprocessing steps carried out before feeding images to deep learning models. Processing techniques used are explained and motivated here:

- Consistent file format: Changing the original file formats from video avi and medical mhd to png images to ensure a consistency in the file format that is fed to the models.

- Grayscale: The key area of echocardiograms is the cone itself. The cone is grayscale, at least to the human eye, and the color channels from the original datasets does not provide additional information. Hence, making sure that all images are grayscale, with one color channel instead of three color channels (Reg, Green, Blue), reduces the complexity to the model.

- Resize: Resizing the images to a standard size to ensure that they can be processed by the machine learning models. This was achieved by down-sampling images with Open CV's cubic interpolation to a standard 112x112 pixels, if not already in that shape.

- Center-crop: When an image is resized or transformed without cropping, the aspect ratio can change, leading to a distorted appearance. By center-cropping, we ensure that the most important part of the image remains intact while eliminating the unwanted parts.

- Normalize: The grayscale echocardiogram images contains pixel intensity values on a varied scale, between 0 and 255, due to the differences in machine settings. It is therefore important to normalize these to ensure that images have similar brightness and contrast levels, which can improve the performance of the classifier. Normalization is done as part of the PyTorch data loader before feeding images to models into a range of values between -1 to 1.

- Outlier removal: Discarding outliers from the datasets, such as images with anomalies (color Doppler, images containing multiple views, abnormal coloring of images), to ultimately improve the performance of the classifier. Example of outliers identified in the dataset can be seen in Figure 3.8 and 3.9 where the latter exemplify outliers usable after converting to grayscale images.

3.3 CAMUS Dataset

The CAMUS dataset [27] contains 2,000 echocardiogram 2D-images collected from 500 patients during clinical exams at the University Hospital of St Etienne (France). Data was collected with varied acquisition settings resulting in different qualities and pathological cases, as typically seen in a general clinical setting. Two views and a total of four images were collected per patient: apical two-chamber (A2C) and four-chamber (A4C) at both End diastole (ED) and End Systole (ES), which can be seen in more detail in Figure 3.1. For some patients it was not possible to collect the four-chamber view and instead a five-chamber view was acquired. A GE Vivid E95 ultrasound scanner was used with a GE M5S probe for all patient screenings.

Original images were downloaded in mhd file format with one color channel (grayscale), with varied heights and widths, and two label views (A2C and A4C, at ED and ES) being part of the file name. There was no overlay information but solely the echocardiogram cone area. The dataset is balanced between the two views (1,000 A2C, 1,000 A4C) and all 2,000 images were converted from mhd to 8-bit arrays with Simple ITK python library, down-sampled to 112 by 112 pixels, and saved in png file format.

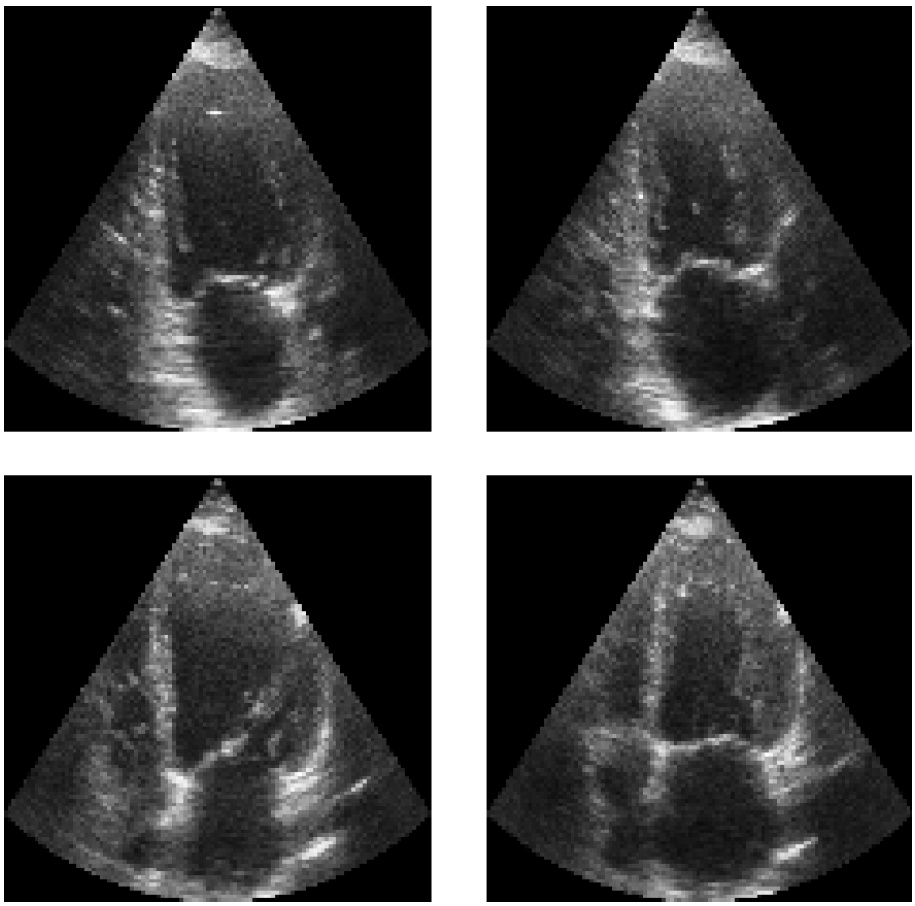


Figure 3.1: Example images from CAMUS dataset showing one patient and A2C (top row) and A4C (bottom row) echocardiogram views at both end diastole (left column) and end systole (right column), post processing.

3.4 Unity Imaging Collaborative Dataset

From the Unity dataset [28], 7,231 echocardiograms of A2C, A4C and PLAX views will be utilised. These have been labelled by experts of the UK group of cardiologist's and sonographer's called Unity Imaging Collaborative. The data is organised in a manner where each echocardiogram video was named by a unique hexadecimal code from which multiple frames can be selected. There is no given naming convention to identify individual patients. Though, when echocardiograms were used for the research purpose it was originally created for, data was divided to have no overlap between unique hexadecimal codes in between training and validation sets. Hence, the same approach was adopted for this project.

The original images were downloaded as png files in 3-color channels (RGB) with varying overlay information and color segmentation of echocardiograms. Several image resolutions were identified across images, all center-cropped, converted to grayscale and resized to 112x112 resolution before being used for testing in this project.

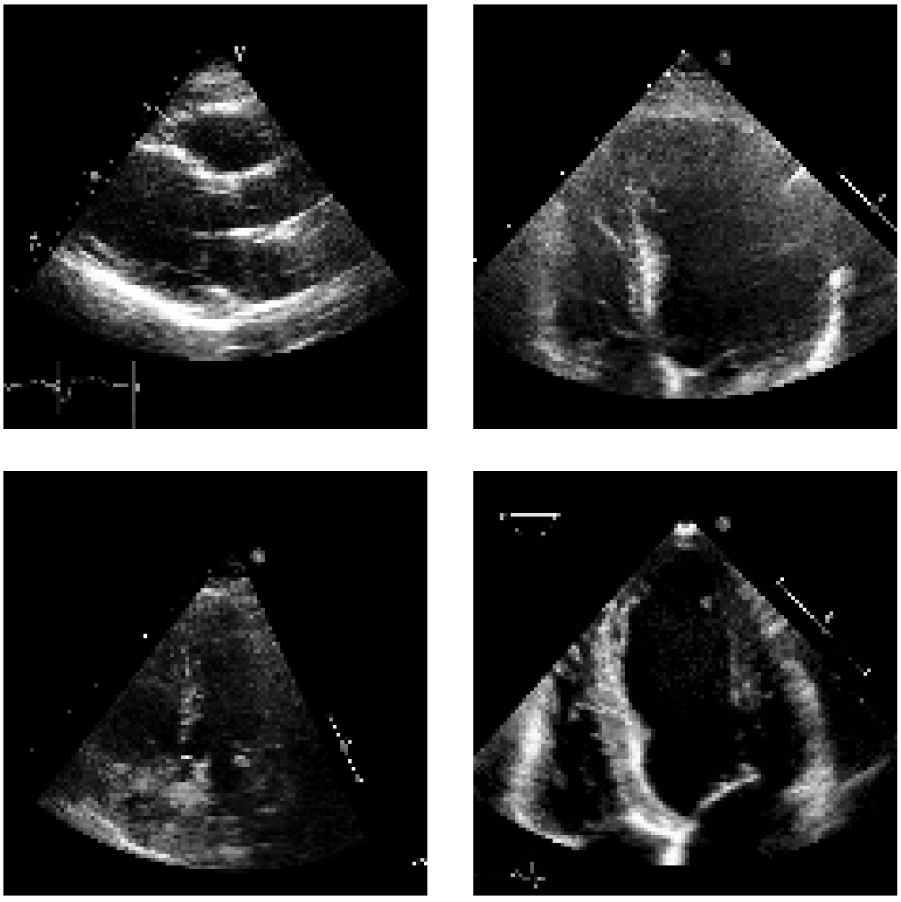


Figure 3.2: Randomly selected Unity images of PLAX views after being downsized and converted to grayscale.

3.5 TMED2 Dataset

The TMED2 [29] includes 17,707 two-dimensional echocardiogram images collected from 1,280 patients. The view labels represented in the dataset are A2C, A4C, PLAX and PSAX, and are imbalanced with a heavier weight towards PLAX and A4C views.

The collection process was conducted as part of routine clinical care at Tufts Medical Center during 2011-2020. Certified sonographers labeled each image with a annotation tool and were instructed to label multiple examples of all four views for each patient. This resulted in that the number of images per patient and view label varied while the majority of patients had about 1-15 images per view. For this reason, up to 15 images were selected per patient, in order to use as much of the dataset as possible while avoiding over representation of a few patients that had a much higher number of images per view.

Images were originally available in 112x112 pixel resolution and grayscale in png format where doppler images, m-mode image and colored images were excluded. Hence, there was no need for additional image processing.

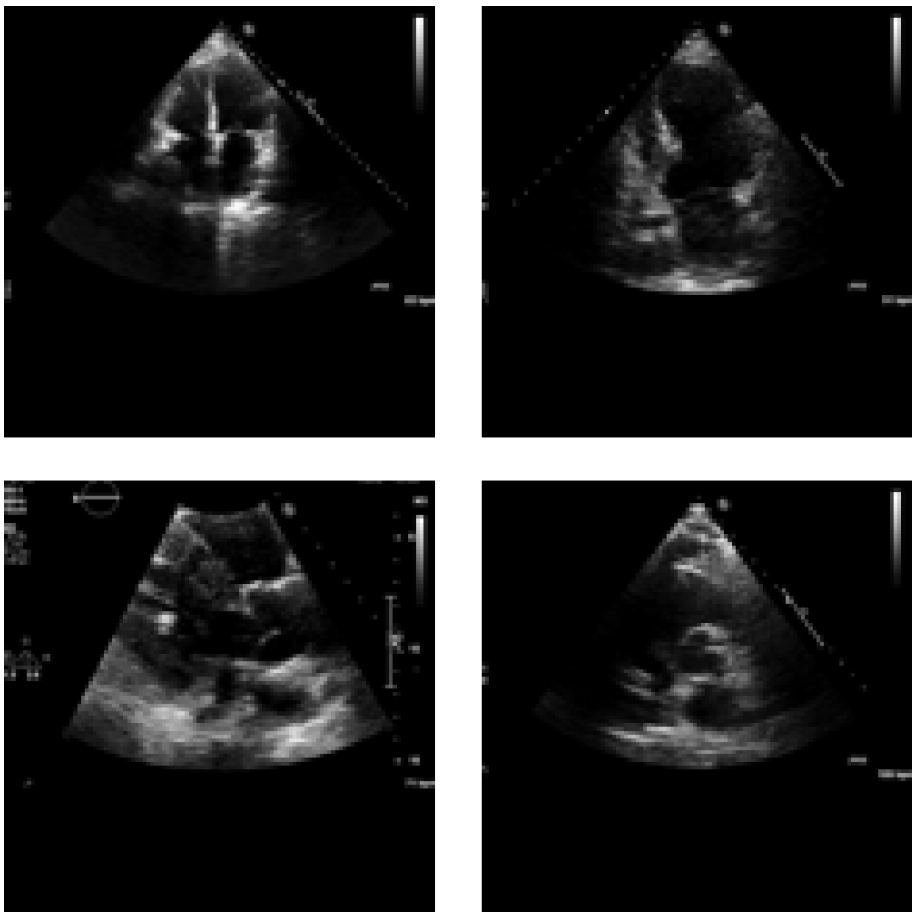


Figure 3.3: Randomly selected TMED2 images showing A4C (top left), A2C (top right), PLAX (bottom left) and PSAX (bottom right) views.

3.6 Stanford EchoNet-Dynamic and EchoNet-LVH Datasets

The EchoNet-Dynamic and EchoNet-LVH datasets (hereafter referred to simply as Dynamic and LVH, respectively) were published by the Stanford University Hospital (US) and contain 10,030 labeled A4C echocardiogram videos and 12,000 labeled PLAX echocardiogram, respectively. Recordings were collected as part of routine clinical care by trained sonographers in an academic medical center with lab settings typical for echocardiogram acquisition. Varied ultrasound machines (iE33, Sonos, Acuson SC2000, Epiq 5G, Epiq 7C) were used to gather Dynamic videos while devices used in collection of LVH are not explicitly stated. Both datasets provide cropped and masked videos to exclude overlay information outside of the cone area.

Videos can be accessed through the Stanford AI in Medicine and Imaging (AIMI) center. Each video sequence comes in avi video file format, three color channels (RGB), and is about 3-4 seconds long. Sequences represent a number of heart beats to capture varied states of the heart view and contain about 50 frames per second (frames are still images of a video at certain points in time). Example frames from a heart beat video sequence is shown in Figure 3.4 for Dynamic and Figure 3.5 for LVH dataset. There were about 100-275 frames per video for Dynamic and 75-250 frames per video for LVH dataset (excluding very short or very long outliers), that are visualized in Figure 3.6 and 3.7. One final frame per video was then selected at random to account for the various states throughout the heart beat cycle. However, before sampling frames from the videos, outlier videos had to be identified and handled appropriately. This process is outlined in the following section. The final frames selected were center-cropped, down-sampled to 112 by 112 pixels, converted to grayscale and saved in png format. Thus, we end up with 10,026 A4C images (from 10,026 Dynamic videos) and 11,182 PLAX images (from 11,182 LVH videos).

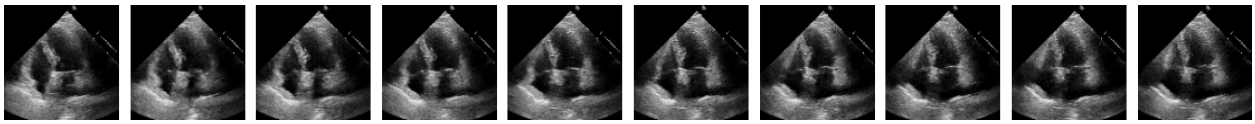


Figure 3.4: Example sequence of frames from one Dynamic video (A4C view).

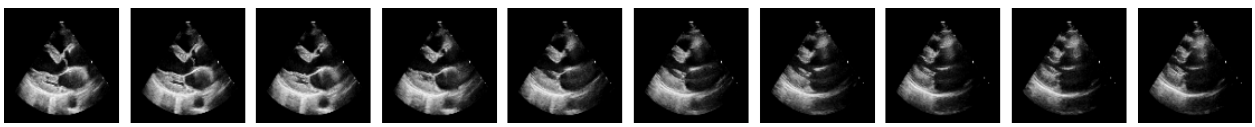


Figure 3.5: Example sequence of frames from one LVH video (PLAX view).

The distribution of frames count per dataset is depicted in Figure 3.6 and 3.7.

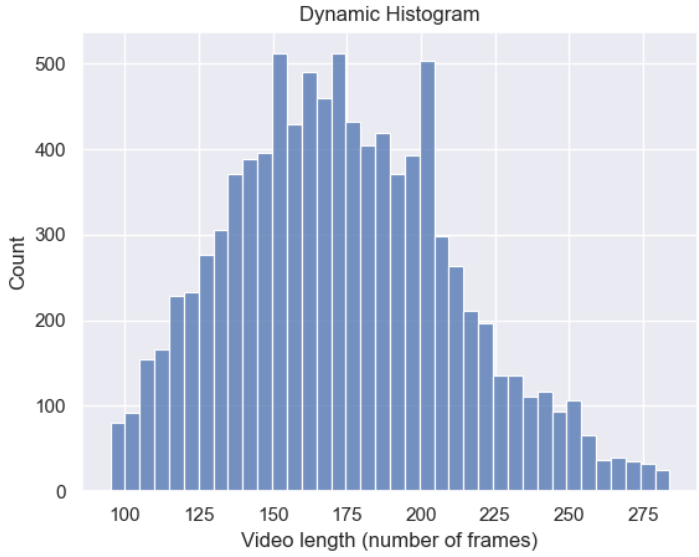


Figure 3.6: Frame count distribution of Dynamic dataset with 95% of frames centered around the mean.

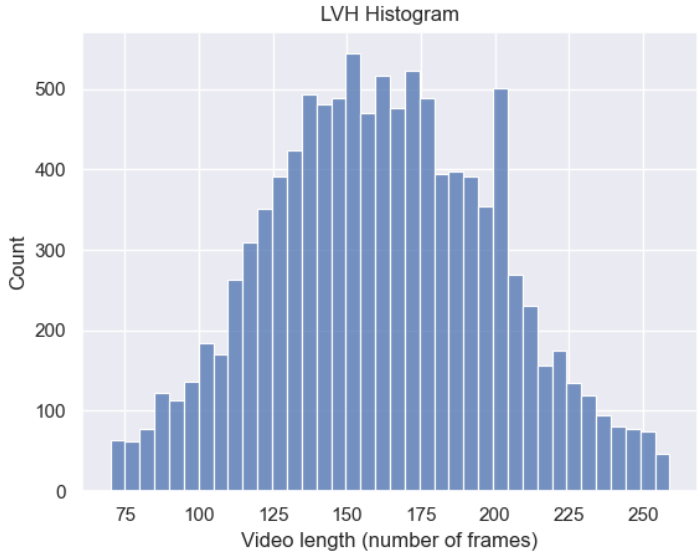


Figure 3.7: Frame count distribution of LVH dataset with 95% of frames centered around the mean.

3.7 Video Preprocessing

The video preprocessing pipeline involved two key steps: video validation and frame selection. The video validation process aimed to identify valid videos for use in the analysis, filtering out videos with dual views, colored Doppler views, or abnormal coloring (Figure 3.8). Subsequently, a subset of frames were selected with an approach

that avoids bias and accounts for variability in frames selected during different states of the heart beat cycle.

3.7.1 Video Validation

The majority of the analyzed videos appeared to be predominantly grayscale to the human observer, with occasional colored overlay information. To limit the model's complexity while mitigating the risk of information loss, it is beneficial to convert the original three-color channels into one-color channel. Prior to this conversion, it was necessary to validate which videos contained colored pixels and assess the usability of such videos.

We developed two approaches for video validation. The initial approach involved extracting the first frame of each video and calculating the pixel-wise differences between the intensities of the three color channels (red-green, green-blue, blue-red). This was based on the assumptions that a video sequence is likely to have similar pixel intensities across frames, and that grayscale pixels have similar or identical values across all channels (RGB). Any pixel whose maximum difference across these channels exceeded a specified threshold was classified as a color pixel.

To account for videos in which the echocardiogram cone was grayscale but contained minor colored overlay information, we introduced a grayscale proportion threshold. Videos whose ratio of grayscale-to-total pixels was below this threshold were deemed to contain too much color, and were excluded from the subsequent analysis. On the other hand, videos whose grayscale proportion exceeded the threshold were classified as grayscale and continued on to frame selection. While this approach seemed sound on paper, in practice we encountered inconsistent results. Therefore, we deemed it necessary to reevaluate our methodology.

We eventually adopted a second approach to video validation, which involved analyzing the first frame of each video to calculate the maximum absolute difference in intensities across the color channels for each pixel. These values were stored in an array for each video. Videos were categorized into distributions based on Kolmogorov-Smirnov (KS) tests with a significance level of 0.05. The first video was assigned to a distribution, and subsequent videos were compared to it using KS tests. If the null hypothesis could not be rejected, the new video was grouped with the initial video's distribution. However, if the null hypothesis was rejected, the new video was not assigned to any distribution and was saved for re-evaluation during the next iteration. This process was repeated for all videos until every video was categorized into a distribution. To ensure data quality, we manually inspected one example frame from each distribution and identified unusable outliers (Figure 3.8). The entire distribution to which an unusable outlier example belonged was considered unreliable, and its videos were excluded from further analysis. The remaining videos were deemed valid for subsequent frame selection. It is important to mention that some examples may appear to be outliers in terms of color scheme (Figure 3.9), but can still be used after converting them to grayscale. Such examples were considered as usable outliers. Since all extracted frames were to be converted from three channel format (RGB) to one channel format (grayscale), no further preprocessing

was required for these usable outliers.

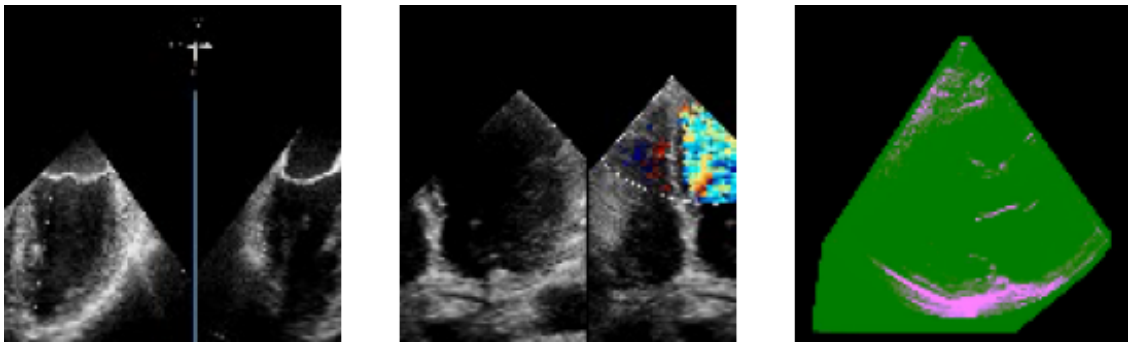


Figure 3.8: Identified outlier echocardiograms that were excluded from the dataset. From left to right: image including split views, Doppler image with coloring, abnormal coloring.

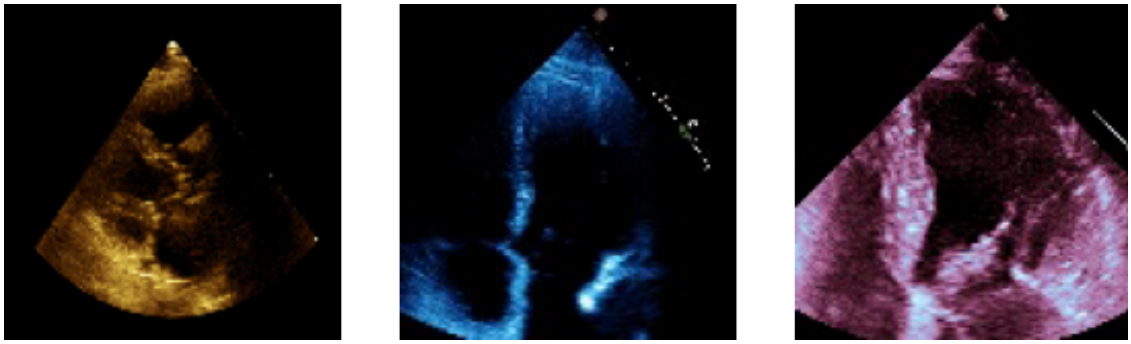


Figure 3.9: Outlier echocardiograms identified but included in the dataset after preprocessing.

3.7.2 Frame Selection

We devised an involved approach to frame selection that aimed to capture diverse frames, in terms of pixel intensity values, from each video. This was based on the assumption that consecutive frames in a video are likely to be similar. To achieve this goal, we constructed a frame similarity matrix by comparing each frame in a video to every other frame. Specifically, we utilized mean squared error (MSE) to measure the similarity between pairs of frames.

Using the frame similarity matrix, we then applied a video-specific similarity threshold to select only frames that were dissimilar from each other. This was accomplished by identifying the 0.25 quantile of the similarity score for each video and selecting only the frames whose similarity score fell below this threshold. By selecting frames with low similarity, we were able to capture a diverse range of frames from each video and thereby increase the variability of our training data.

Although our original approach of selecting frames based on their dissimilarity yielded satisfactory results, we eventually decided to instead select a single random frame per video. Our motivation was to avoid overrepresentation of images

extracted from video datasets in our final training set, thereby achieving a reasonable balance across source datasets. Additionally, this approach yields variability of data by allowing selection of frames from arbitrary moments in the heart beat sequence and avoids bias in sampling.

3.8 Dataset Summary

In summary, it is important to know the limits of each dataset by its own characteristics. Despite all being collected in a typical clinical setting, these characteristics may not generalise well to other echocardiogram datasets acquired with other operational techniques, staff, devices and settings. Additionally, both video processing and image processing techniques had to be applied before arriving at our final datasets: our main 4-class dataset, used for training, validation and testing of diffusion and classification models, and our secondary 3-class dataset used for testing purposes only.

The 4-class (A2C, A4C, PLAX, PSAX views) dataset thus contains images from TMED2, Dynamic and LVH, and totals 38,915 echocardiogram images. This 4-class dataset is further split into train, validation and test sets (80%/10%/10%), detailed in Table 3.1. A total of 9231 images are derived from CAMUS and Unity datasets with 2,000 and 7,231 images respectively. Those images belong to 3 classes (A2C, A4C, PLAX) as shown in Table 3.3. An additional five smaller subsets of training data are created from the 4-class training split to explore the impact of dataset size on diffusion and classification performance.

Table 3.1: Summary of 4-class dataset with train, validation and test splits.

Split/Dataset	LVH	Dynamic	TMED2	Total	Share
Train	8,945	8,020	14,165	31,130	80%
Validation	1,118	1,002	1,775	3,895	10%
Test	1,119	1,004	1,767	3,890	10%
Total	11,182	10,026	17,707	38,915	
Share	29%	26%	46%		100%

Table 3.2: Imbalance between classes shown for 4-class dataset.

Split/Dataset	A2C	A4C	PLAX	PSAX	Total	Share
Train	2,752	11,635	14,573	2,170	31,130	80%
Validation	345	1,465	1,807	278	3,895	10%
Test	335	1,431	1,845	279	3,890	10%
Total	3,432	14,531	18,225	2,727	38,915	
Share	9%	37%	47%	7%		100%

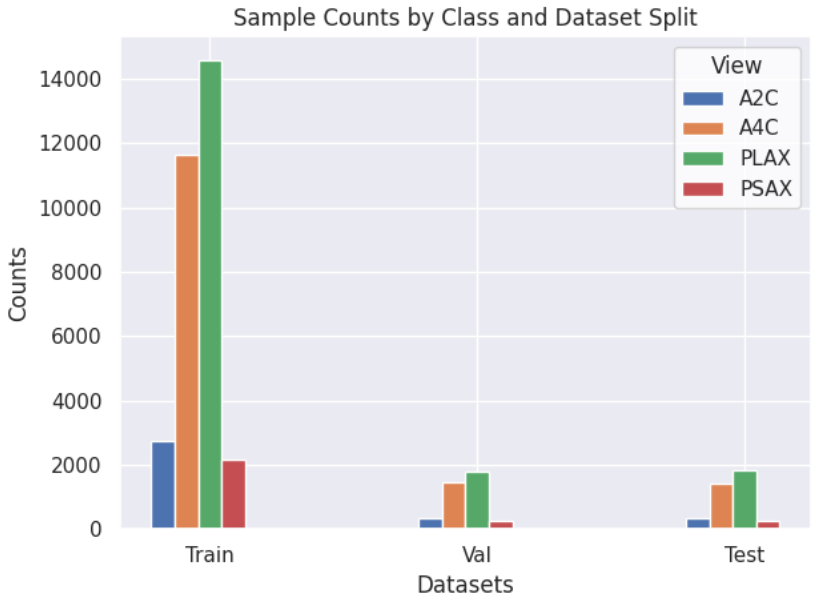


Figure 3.10: 4-class dataset (N=38,915) detail by view and split.

Table 3.3: Summary of 3-class dataset only used for testing.

	CAMUS		Unity			
	A2C	A4C	A2C	A4C	PLAX	Total
Items	1000	1000	1029	3311	2891	9231

A total of six real training data subsets are used in experiments outlined in the remaining parts of this report, and are summarized in Table 3.4. The proportion of images belonging to each dataset (LVH, Dynamic, TMED2) and class is kept in all data splits and subsets created. Individual patients are kept separated with no overlap between different splits of the data where patient information was available (only Unity did not provide patient IDs but hexadecimal codes). For example, images from one patient, if part of the training set, will not be part of validation nor test sets, and vice versa.

Table 3.4: Summary of six training subsets of real data created from the training set of the original 4-class dataset.

Size	Views:				Datasets:		
	A2C	A4C	PLAX	PSAX	Dynamic	LVH	TMED2
1,000	89	374	468	69	253	289	458
2,000	176	748	937	139	515	574	911
4,000	354	1,496	1,872	278	1,045	1,162	1,793
8,000	707	2,990	3,745	558	2,070	2,289	3,641
16,000	1,415	5,980	7,490	1,115	4,077	4,576	7,347
31,130	2,752	11,635	14,573	2,170	8,020	8,945	14,165

4

Methods

This section provides an overview of the project along with a definition and motivation of evaluation metrics used to assess synthetic images and view classification. Finally, we describe the specific experiments carried out in relation to diffusion models, generating images and echocardiogram view classification. Both this section and the following Section 5 use the same subheadings for individual experiments to facilitate matching experiments' methodology and results.

4.1 Pipeline Overview

The first natural step to answer our second research question, related to exploring the impact of classification performance by using synthetic data, is to establish a baseline classification performance of real data alone. The next step, in order to make the most out of the real data available, would be to use the same real dataset to train a diffusion model to generate a greater volume of synthetic images. These synthetic images would in turn be used to train a separate classifier. We would then be able to compare the performance between the baseline classifier trained on real data against the one trained on synthetic data. The purpose would be to explore the potential performance difference between these two approaches:

- using real data available, as is, to train a classifier
- using real data available to train a diffusion model, sampling from it a larger synthetic dataset, and using the synthetic data to train a classifier

One assumption is that the size of the imbalanced original dataset may matter. For this reason, we explore increasingly smaller subsets of the original training dataset. As this analysis would be limited to explore classification performance when trained on either real or synthetic data exclusively, we also explore whether real data augmented with synthetic data could yield valuable results. The aim is to investigate what the optimal use of synthetic data could be in echocardiogram view classification.

In order to answer our research questions we have developed a project pipeline for this thesis work. This pipeline entails the customized data to be used, generation of synthetic echocardiograms, and evaluating classifiers trained with the different datasets. The project pipeline is depicted below in Figure 4.1.

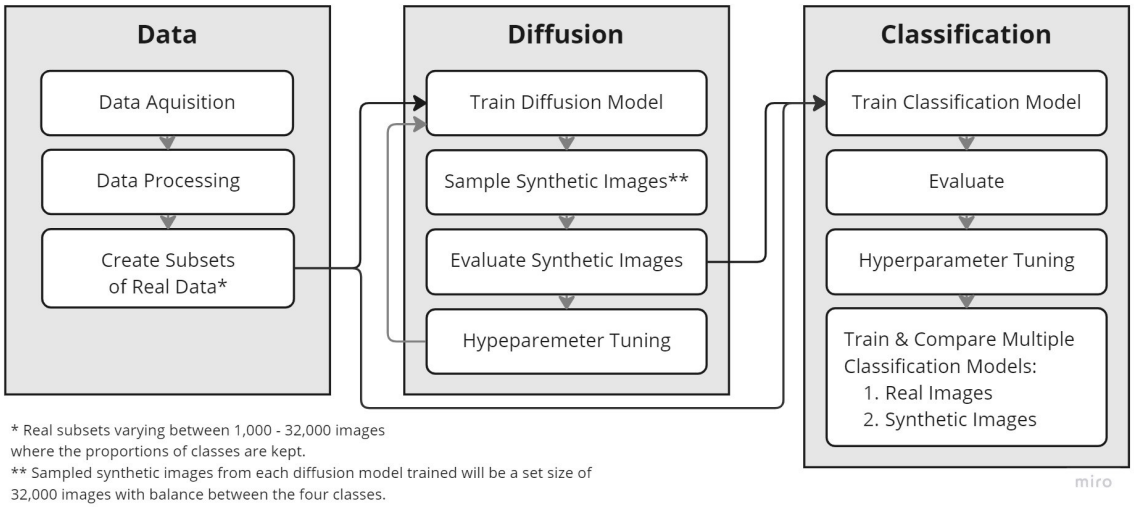


Figure 4.1: Project pipeline overview

The pipeline includes the following steps:

1. Acquire datasets from research institutes.
2. Preprocess image and video data to exclude outliers and configure final images to a consistent format.
3. Build custom datasets of varied sizes, keeping the proportions of imbalanced classes unchanged.
4. Choose a Diffusion model architecture based on the literature review and initial experimental results.
5. Use the architecture to train an initial diffusion model, sample images, evaluate and tune hyperparameters.
6. Train multiple diffusion models with real datasets of different sizes to sample synthetic image datasets of a fixed size.
7. Evaluate the generated images against the validation data by calculating FID.
8. Choose classification model architectures based on the literature review and evaluate with initial experimental results.
9. Train classification models with real and synthetic datasets of varying sizes.
10. Evaluate and compare classification models' performances on the real validation set.

4.2 Evaluation Approach

In this section we provide a brief introduction to our evaluation approaches covering, both synthetic data generation and echocardiogram view classification.

4.2.1 Synthetic Images

Evaluating generative models is still an area of active research. However, a common approach is to compare the distributions of real and synthetic images. FID [30] is outlined below as the main quantitative metric for evaluating synthetic images. A more qualitative evaluation approach is taken by visualizing the features of synthetic and real image using dimensionality reduction, and by having medical experts surveyed on the task of distinguishing real echocardiograms from synthetic ones.

4.2.1.1 Quantitative Assessment

We employ FID as an overall metric for judging performance of a trained diffusion model. FID is a measure of similarity between two sets of images. It computes feature representations using a pretrained Inception V3 network for both sets of images, fits a Gaussian distribution to each set of feature representations, and finally computes the Fréchet distance between the two. As a measure of distance between these two distributions, a lower FID implies a higher similarity between the two sets of images. We thus use FID to evaluate synthetic echocardiogram images against real echocardiogram images.

4.2.1.2 Feature visualizations

The feature representations of real and synthetic images are taken from the activation of the second to last output layer of a CNN architecture to be trained on the 80% training dataset. These are then comprised into a two-dimensionsal t-SNE embedding scaled between 0 to 1, before being plotted. The purpose of this visualizations is to explore how the real and synthetic images position in relation to each other. This can help illustrate how diverse the synthetic echocardiograms are when compared to the real data.

4.2.1.3 Human Evaluation

A survey will be shared with persons with varied medical experience and specialties, with the assumption that they would be better than random at distinguishing between real and synthetic echocardiograms. They will be presented with 50 image pairs, each showing one real and one synthetic echocardiogram displayed next to each other. Then they will be asked to identify which out of the two images that was the real echocardiogram by picking left or right. An example image pair is shown in Figure 4.2. The placement (left or right) of the real echocardiogram was decided at random. Both images in each pair belong to the same view. The survey will only include views from the majority classes (A4C and PLAX). Real images will be randomly selected from the training set while synthetic echocardiogram will be generated from our diffusion model trained on the full 4-class training dataset. At the end of the survey participants will be able to describe and share what their general approach to distinguish between real and synthetic echocardiograms was.

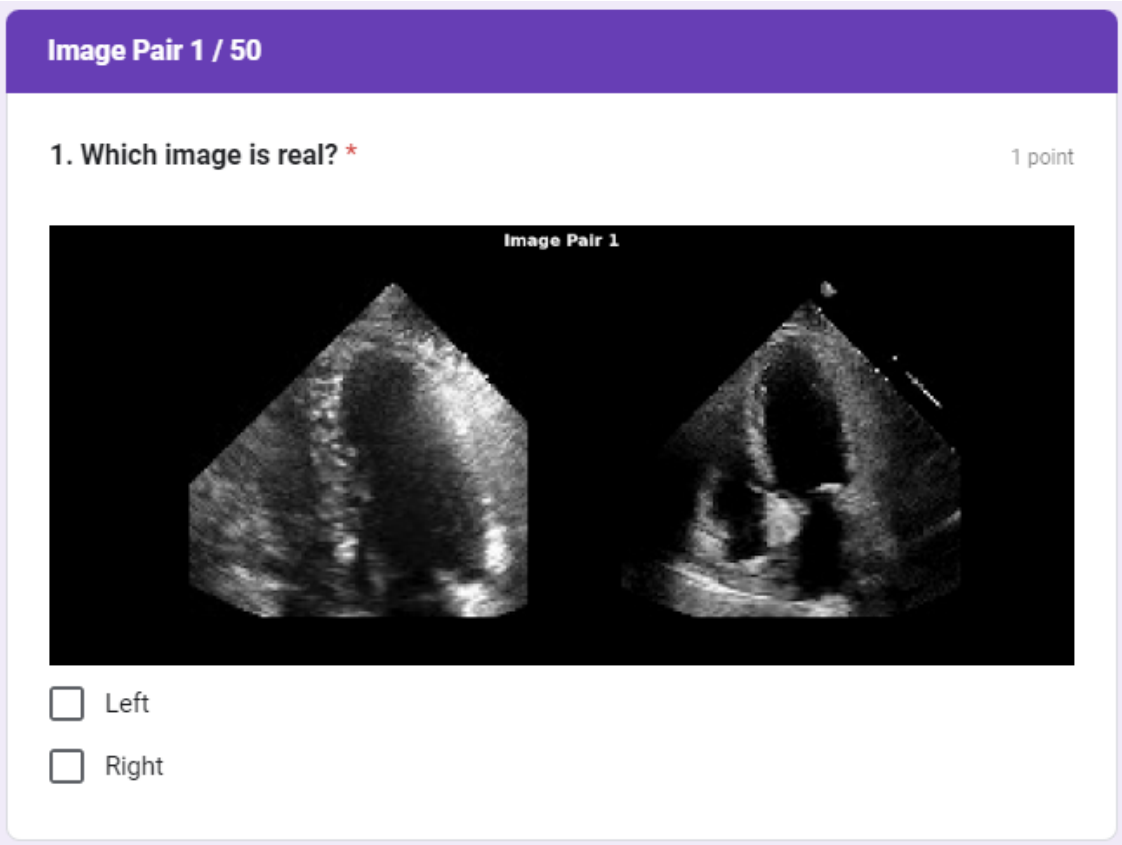


Figure 4.2: Simple game to test whether a human subject can distinguish real from synthetic echocardiograms.

4.2.1.4 Limitations of Synthetic Evaluation

Synthetic data evaluation is still a recent and relevant research area and continuously being developed. Considering this, there are some limitations to the current evaluation approach:

- Creation of synthetic images of high diversity has been a persistent challenge for generative models. While a model that consistently produces high-quality output is desirable, it is of limited usefulness if it is only capable of generating a limited variety of images.
- Quantitative FID measures is based on complex calculations making them difficult to interpret and comprehend.
- FID is less valuable on its own. For example, if generating images with one diffusion model and calculating FID between those images and the original training images, it is difficult to identify what score constitutes good or bad performance without any context or comparison. The aim in this project is to use FID as a means to compare and evaluate different diffusion models to one another, and assess how long they should be trained for. In order to ensure a fair comparison between different sets of synthetic images, they should all be of the same size (i.e. number of images) and their FID should all be computed

against the same set of real images, to limit confounding factors.

- There is a lack of measures to assess the realism of synthetic images, meaning the generated images could potentially be anatomically incorrect or otherwise inaccurate.
- The current evaluation approach does not measure potential copies or close copies of original patient images in the synthetic data generated, which evokes a patient privacy concern.
- There are currently no measures taken to quantify potential bias found in synthetic data other than the known characteristics laid out in the data section. Generally, to avoid having bias translated into synthetic images one should train on a dataset known to have captured the proper diversity of conditions of the population and clinical phenotypes. The larger the dataset the greater the likelihood to cover for a wider diversity. However, due to patient data privacy, distributing medical data is complex, and there is a risk of violating regulations and leak sensitive information.
- FID is currently calculated using the InceptionV3, pretrained on ImageNet, which learned features that may be very different from those relevant to echocardiograms. A more reasonable approach would be to use a model trained on echocardiograms to extract features with which to compute FID.

4.2.2 View Classification

Handling imbalanced datasets is a critical aspect of view classification evaluation. Imbalanced datasets occur when the number of instances in different classes is significantly unequal. The 4-class dataset used in this project has two minority classes (A2C, PSAX) accounting for only about 10% of the dataset each, while the other two classes (A4C, PLAX) are in the majority. In this scenario, using traditional evaluation metrics like accuracy may not provide an accurate representation of model performance as it would generate accurate predictions for the majority of observations when having minority classes. Instead, metrics like Macro F1-score and *Area under the Precision-Recall Curve* (Area Under the Precision-Recall Curve (AUPRC)), also referred to as *Average Precision*, provide a more transparent and informative measure of performance. Additionally, Confusion Matrices will be used to provide a comprehensive summary of the model's predictions by offering a detailed breakdown of the classification results.

Both Precision and Recall calculations rely on True positives, False positives and False negatives. Precision ($\frac{T_p}{T_p+F_p}$) refers to the ratio of correctly classified positive instances to the total instances predicted as positive. On the other hand, Recall ($\frac{T_p}{T_p+F_n}$) measures the ratio of correctly classified positive instances to the total actual positive instances. AUPRC is a single value summarizing the Precision-Recall Curve (PRC), which itself is a graphical visualization showing the trade-off between Precision and Recall at various threshold values. Plotted on the x-axis is Recall, and Precision on the y-axis, both bounded between 0 and 1. Higher scores, closer to 1, are generally preferred indicating that the classifier performs well for the given task.

AUPRC is calculated by summarizing the PRC as a weighted mean of Precision for every threshold, and uses a weight calculated on the difference in Recall from the previous threshold. This is performed for each class before being average across the total number of classes. As shown in Equation 4.1, P_n and R_n is the corresponding Precision and Recall at threshold index n:

$$AUPRC = \sum_n (R_n - R_{n-1}) P_n \tag{4.1}$$

F1-score is a single metric that combines Precision and Recall into a harmonic mean, providing a balanced global measure of the model's performance. It is especially useful when both Precision and Recall are equally important in view classification tasks. Macro F1-score is calculated per class and is then averaged across classes. Any class without true or predicted images are ignored to ensure only relevant classes are included in the calculation and equally weighted to not prioritize majority classes over minority ones. F1-score calculation is outlined in Equation 4.2 below:

$$F_1 = 2\frac{\text{Precision} * \text{Recall}}{(\text{Precision}) + \text{Recall}} \tag{4.2}$$

4.3 Experiments

Experiments are divided into two main components, one building on top of the other. Starting with experiments to help us land with a diffusion model of choice that can be used for sampling synthetic images. In turn, these synthetic images will be further used in experiments related to echocardiogram view classification, along with real data subsets.

The original 4-class real data (LVH, Dynamic, TMED2) was split into 80% training, 10% validation and 10% testing, while 100% of the 3-class real data (CAMUS-Unity) was used for final testing. All code was written in Python (version 3.9.12) and PyTorch (version 2.0.1) was our main framework utilised for deep learning experiments throughout the project. Computational resources needed to train and evaluate diffusion and classification models were supplied via Alvis with NVIDIA Tesla A100 and A40 GPUs.

4.3.1 Diffusion

4.3.1.1 Initial Repository Exploration

After careful consideration, we identified two GitHub repositories, Medfusion [13] and Improved Diffusion [11], that we could use to generate synthetic images. We conducted tests on both repositories, making minimal modifications, to train the models on our custom datasets.

At this early stage of the process we created 2 custom datasets on which to train on and benchmark the candidate diffusion repositories. These datasets were early

iterations of the full training set detailed in our Data summary section (3.8).

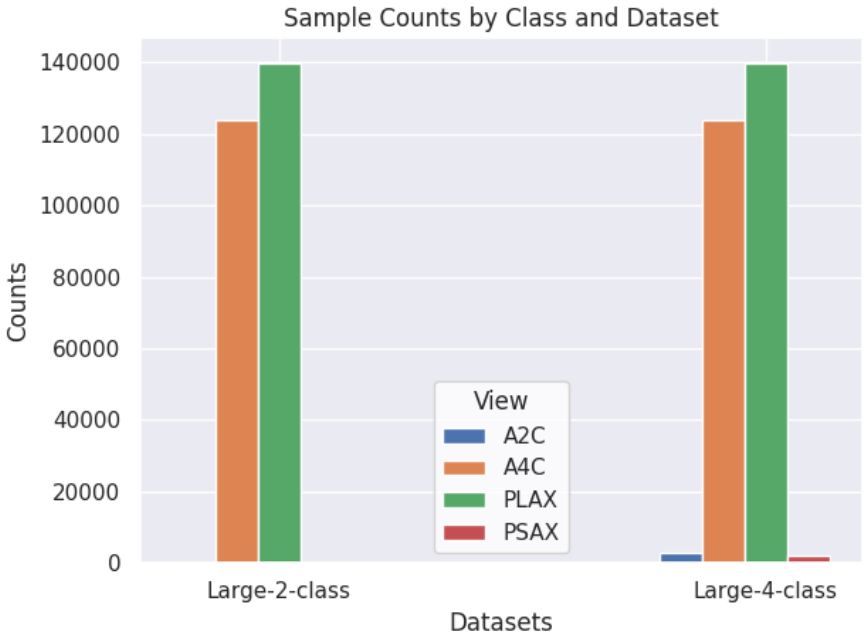


Figure 4.3: Datasets used for initial repository exploration.

It is worth noting that these custom datasets were only used for the purpose of deciding on a repository, and were not used throughout the rest of this thesis. The reason for this is that, as our work progressed, and as our understanding of the capabilities of diffusion models increased, we made further changes to the datasets we worked with, to be able to arrive at more clear and more interesting conclusions. These changes ranged from committing to utilizing 4 classes, to sampling fewer frames from LVH and Dynamic videos for balance purposes. Regarding the results from the analysis conducted, the difference in performance between the 2 repositories was large enough across both datasets that we felt confident in moving forward with the selected one. Our underlying assumption here was that the large performance difference must be due primarily to one repository being better for our use case than the other, and that this pattern would hold even when tested with different datasets like the ones we would later end up using.

In each repository, we made the necessary modifications for the models to accept inputs of the correct shape (112x112x1) and initiated the training process. Additionally, we dedicated a small amount of time to tuning the hyperparameters of each repository at this stage, such as learning rate and layers of the noise estimator (U-Net), but ultimately reverted to the default settings since they produced superior results.

Once training was complete, we sampled 1,000 images from each model (Medfusion and Improved Diffusion). As expected, sampling was a bottleneck, taking considerable time. Generating 1,000 synthetic images required approximately 30 minutes, while generating 10,000 images using four GPUs in parallel took 5-6 hours. The

final step in this stage of the process was evaluating the sampled images against the real echocardiograms. For this purpose we used the FID metric. As a metric sanity check we evaluated a set of 1,000 synthetic samples against itself, and obtained the expected results (FID=0). Then we proceeded to examine the results of the 1,000 synthetic images against the training dataset. These results are presented in Section 5 and based on them we decided to move forward with the Improved Diffusion repository for the rest of this project.

We identified two critical hyperparameters to fine-tune in our Improved Diffusion model. Typically, diffusion models are trained to learn only the mean of the noise at each time step in the reverse diffusion process. However, research suggests that in some cases, learning the variances in addition to the means can improve performance. Likewise, the original linear noise schedule used in the forward diffusion process can be replaced with a cosine noise schedule to yield better results [11]. To investigate these possibilities, we trained four models using different combinations of these hyperparameter choices two synthetic samples were generated. The first set comprised 1,000 images, while the second set consisted of 10,000. We explored the relationship between the number of sampled images and the resulting FID. Specifically, we compared each set of 1,000 synthetic images against 1,000 random real images, and each set of 10,000 synthetic images against 10,000 random real images. At this point we were only interested in comparing the hyperparameters themselves and not the size of synthetic samples. calculated between the two synthetic samples and a set of 10,000 real images.

4.3.1.2 Ideal Sample Size

A great volume of images can be generated once a diffusion model is trained, however the number of samples to be generated requires significant computational resources. Given that sampling can be time-consuming, we aimed to strike a balance between achieving convergence in FID and minimizing sampling time. To gain more knowledge about the ideal sample size we proceeded to generate multiple synthetic samples (1,000, 2,000, 4,000, 6,000, 8,000, 10,000, 15,000, 20,000, 30,000, 40,000, and 50,000). The evaluation was performed by calculating FID between all eleven sets of synthetic samples and a set of 10,000 real images to find a convergence point of FID.

4.3.1.3 Training Time

The Improved Diffusion model can be trained indefinitely unless stopped at certain time, loss or number of update steps. We trained a diffusion model on the full training set of almost 32,000 real images for 72,000 update steps, saving checkpoints at evenly spaced intervals of 9,000 steps. We then sampled a set of 10,000 synthetic images from each of these model checkpoints, and computed their respective FID when compared against a fixed validation set. Thus, we obtain FIDs of images sampled at different points throughout the diffusion model training process. This provides a quantitative way to compare the quality of the generated images directly throughout the diffusion model training process, rather than simply going by the loss metric which proved an unreliable proxy for image quality.

4.3.1.4 Qualitative Evaluation

Visualizations of real and synthetic images will be presented in the results section to allow for some human-eye comparison of real and synthetic images. The purpose is to demonstrate the quality of generated images, and give the opportunity to try to spot synthetic images when mixed with real ones. The idea is also to facilitate comprehension of the challenge presented in the survey administered to individuals with medical experience. Only allowing participant with medical expertise was based on the assumption that medically trained professionals would at some point be exposed to ultrasound images and echocardiograms, and therefore have a better chance at distinguishing real echocardiograms from synthetic ones. Their task was to distinguish real echocardiograms from synthetic ones when presented with pairs of images including one real and one synthetic echocardiogram. The findings from the survey is presented in Section 5.1.5.

4.3.2 View Classification

4.3.2.1 Initial Architecture Exploration

As a first step in choosing a classification model for view classification, multiple state-of-the-art deep neural networks were selected based on the latest echocardiogram view classification research outlined in Section 2. The selected models were VGG16, DenseNet121, ResNet18 and InceptionV3, loaded with PyTorch deep learning framework.

Two training approaches were utilised for each model to evaluate a to decide on a preferred approach out of the two. For the first approach, model architectures were used without loading or freezing any pretrained weights, essentially training models from scratch. In each architecture, the last fully connected layer was replaced by one of output size equal to number of classes. For the second approach, both model architectures and their corresponding pretrained weights were loaded, all pretrained on ImageNet-1K. The convolutional layers were frozen and the last fully connected layer was replaced by one of output size equal to number of classes. We intended to try out different transfer learning approaches, but after reviewing our initial classification results (Table 5.3) we deemed this unnecessary.

All models were trained on the largest subset of real data (32k), while evaluation was done on the validation set. Training for all models was carried out using 25 epochs, stochastic gradient decent with a 0.001 learning rate and 0.9 momentum, a learning rate scheduler with a step size of 5 and 0.1 gamma, a batch size of 32, and cross-entropy loss.

4.3.2.2 Experiments with Real and Synthetic Subsets

We continue to analyse the performance of three classifier architectures under different settings. Our main point of interest is to study how synthetic data can help improve classification performance when real data is limited to varying degrees. In order to do so, and as outlined in Section 3.8, we create training subsets of real and

synthetic data based on the training set from our 4-class dataset:

- Real subsets: We employ the full training set, consisting of almost 32,000 images covering 4 views. In addition, we take increasingly smaller subsets of the full training set, keeping the proportions among views constant, to arrive at 5 additional training sets of sizes 1,000, 2,000, 4,000, 8,000 and 16,000 images, all covering the same 4 views. This results in a total of 6 different real subsets used to train the classifiers.

- Synthetic subsets: We train one diffusion model on each of the real training subsets described above. Once the diffusion models are trained, we sample from each a balanced dataset of 32,000 synthetic images. This results in 6 synthetic subsets of the same size, coming from diffusion models trained on the real subsets of varying sizes. The intention behind this is to explore how the amount of real data available impacts the performance of the resulting synthetic data on a downstream classification task, keeping the size of the sampled synthetic data itself constant. Table A.2 uses column "Size" to refer to the size of the real subset that was used to train the diffusion model from which the synthetic images were sampled from in each case.

The experiment flow for real subsets is summarized in Figure 4.4 while the corresponding experimental flow for synthetic subsets is visualized in Figure 4.5.

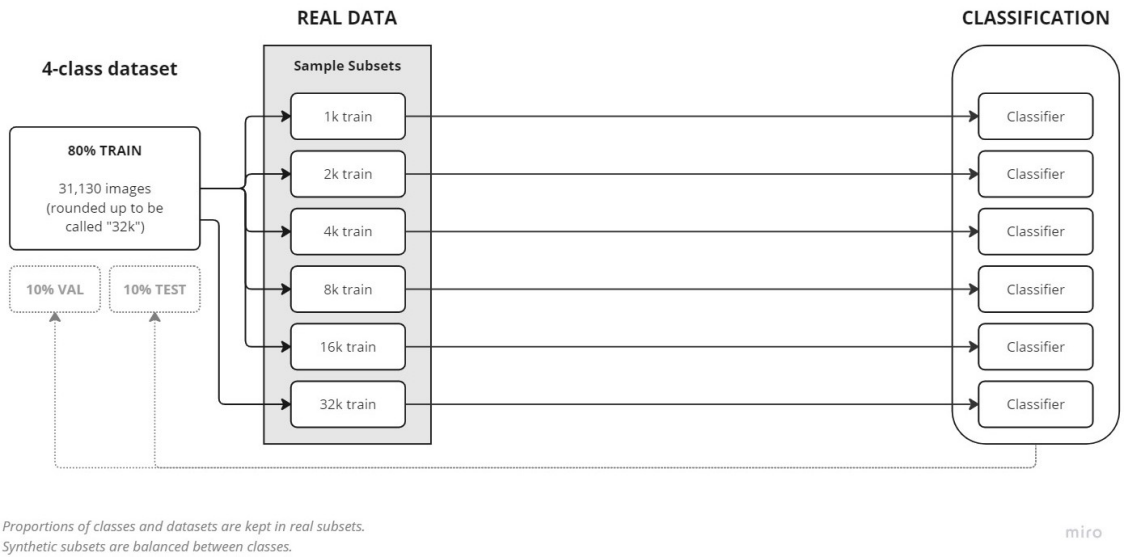


Figure 4.4: Overview of experiments with real subsets of varying sizes.

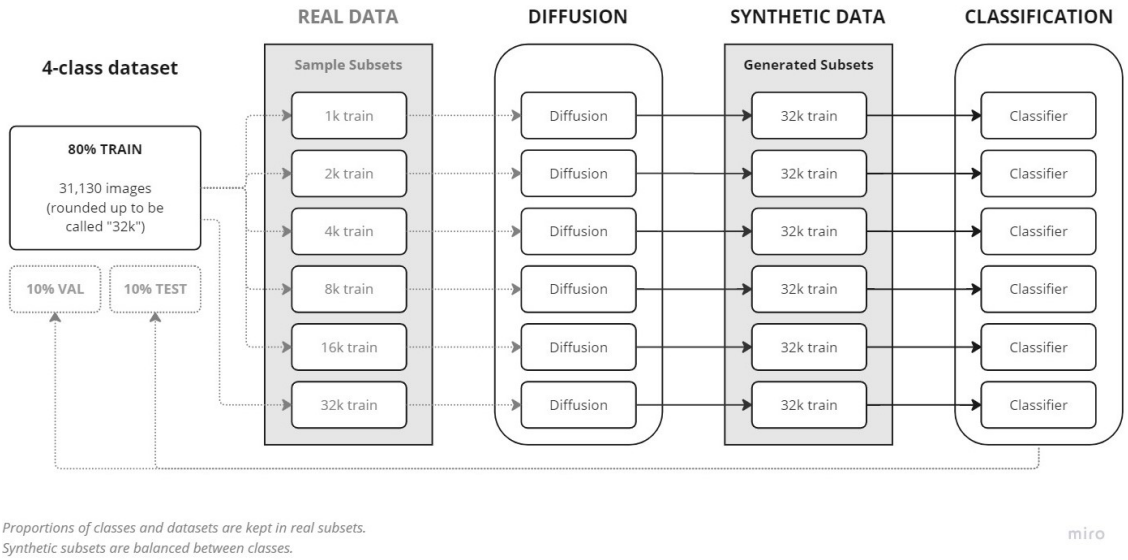


Figure 4.5: Overview of experiments with synthetic subsets of fixed size.

4.3.2.3 Cluster Visualization of Image Features

Real and synthetic images are grayscale, each consisting of 12,544 pixels (112x112x1) with values (intensities) between 0 and 1. These 12,544 numbers for one image can be formatted into a 1x12544 vector, also referred to as the pixel space.

In order to generate an interpretable visualization, we run images through our VGG16 view classifier trained on the full subset of real images, extract the activations of the second to last layer to be used as image features, and perform dimensionality reduction on these. This process transforms the high-dimensional pixel space (12,544-dimensions) first into a lower-dimensional feature space (4,096 dimensions), and finally into a planar representation (2 dimensions).

Principal Component Analysis (PCA) and t-Distributed Stochastic Neighbor Embedding (t-SNE) are common reduction methods converting n-dimensions into k-dimensions while preserving as much information as possible. The main difference is that PCA preserves large pairwise similarities to optimize variances while t-SNE preserves local similarities. t-SNE is a non-linear unsupervised method that allows more flexibility compared to PCA when it comes to more complex non-linear data.

t-SNE uses a student t-distribution having a long tail to fit points so that the neighbourhood of points are maintained. The number of neighbours to preserve is defined by a perplexity parameter (30 by default). Similarities between points (conditional probability that x_i would pick x_j as its neighbor) are converted to joint probabilities by the algorithm, which then aim to minimize Kullback-Liebler divergence between the joint probabilities of high-dimensional data and low-dimensional embeddings. It is a stochastic non-deterministic algorithm, meaning it is random by nature and can produce different results at every initialization (random_state=42 used for reproducibility). By default, the algorithm runs for 1,000 iterations, continuously moving data points, which can be computationally heavy when working on very

high-dimensional data. Default values for perplexity and iterations were used after testing 20,30,50,70 in perplexity with both 1000 and 2000 iterations with very similar results.

4.3.2.4 Impact of Synthetic Data on Minority Classes

The use of synthetic data can serve as a means of mitigating data imbalance issues. To illustrate this point, we present a closer look at the results obtained from evaluating our ResNet18 models trained on real and synthetic data in the scenario where only 1,000 authentic images are available. We have selected ResNet18 for this purpose since it is the architecture that demonstrates the smallest increase in performance when transitioning from training on authentic data to training on synthetic data. Therefore, the results presented in this section can be considered a conservative illustration of the potential benefits that can be attained by utilizing synthetic data to address class imbalance issues on small real datasets.

4.3.2.5 Synthetic Augmentation of Real Data

This section focuses on examining the effects of incorporating synthetic images into real data. Our objective is to assess the impact of gradually introducing synthetic images to each of our real subsets. Specifically, we incrementally add a certain number of synthetic images to the real subsets such that, at each step, 10% of the size of the real subset is added as synthetic images. The resulting image set is then utilized to train a VGG16 classifier, and its performance (measured by AUPRC) is evaluated on an unseen validation set.

4.3.2.6 Synthetic data compared to basic upsampling method

In order to evaluate how the use of synthetic data compares to some basic upsampling method, an experiment will be carried out to extend the data belonging to classes by randomly selecting and copying images. With this technique two additional datasets will be created:

1. With the 1,000 real subset, classes will be upsampled to the same amount of images available in the class with highest frequency. This result in that A2C, A4C and PSAX will be upsampled to contain 468 images each, same as the majority PLAX class. This makes up a total of 1,872 images across all four classes.

2. With the 1,000 real subset, classes are upsampled to contain 8,000 images each, leading to a total of 32,000 images across the four classes. This makes the size comparable to the synthetic dataset generated by the diffusion model trained on the 1,000 real subset.

A VGG16 model architecture will then be used to train on these two upsampled datasets separately. They will then be compared to the AUPRC performance of the VGG16 classifier trained on the 1,000 real subset and the 32,000 synthetic subset (originating from the 1000 real subset). All four classifiers are evaluated on the same validation set.

4.3.2.7 Final Testing

Finally we evaluate our final choice of trained classifier on the holdout test set, which contains images coming from the same 4-class datasets (i.e. same distribution) as the training and validation sets: LVH, Dynamic, and TMED2.

Additionally, we use a secondary holdout test set, which contains images from entirely different sources: CAMUS and Unity. These contain only 3 views in total (A2C, A4C, & PLAX), with potentially large differences in the images compared to the training set, due to factors such as machines used for echocardiogram recordings, among others. This secondary test set is therefore used to analyse how well the final model performs under domain shift.

5

Results

This section is structured into two distinct subsections, namely Diffusion and Classification results, which are inherently connected. The findings obtained from the Diffusion section forms the base for the experiments conducted in the subsequent Classification section. This latter section offers our insights into the influence of synthetic echocardiograms on the performance of view classification based on our own experiments. Notably, each result subheading is closely linked to the description of how each experiment was designed in Section 4.3.

5.1 Diffusion

Results related to Diffusion, presents an exploration of two distinct repositories, which culminated in the selection of a suitable diffusion model and settings. The subsequent discussion sheds light on the rationale behind the sample size used for generating synthetic images and the associated time and resources required. Furthermore, an evaluation is provided assessing the quality of the synthetic samples generated.

5.1.1 Initial repository exploration

Below we present the first comparison between the two tested repositories after training them with two different datasets: one 2-class dataset (A4C and PLAX) and one 4-class dataset (A2C, A4C, PLAX, PSAX). These were both of very large sizes (>300,000 images) since multiple frames were being samples from the LVH and Dynamic video datasets. These datasets later proved irrelevant due to their unrealistically large volume of labeled data in medical scenarios, and being largely imbalanced in relation to dataset representation, where TMED2 became a minority. Hence, these datasets were solely used for initial testing of repositories.

Table 5.1 shows the results of this evaluation. In both cases Improved Diffusion outperformed Medfusion by a fairly large margin, so we decided to move forward with the Improved Diffusion.

Table 5.1: FID comparison between the Medfusion and Improved Diffusion repositories where low FID is better.

Dataset/ FID	Medfusion	Improved Diffusion
Large-2-class	95.89	30.48
Large-4-class	78.16	47.46

We then moved on to hyperparameter tuning. With limited time, we chose to focus on those deemed likely to have the most effect: learning variances and noise schedulers [11]. In order to conduct this test, we sampled 2 sets of synthetic images from our trained diffusion models, one set with 1,000 and another with 10,000 synthetic images. The results are presented in Table 5.2 below:

Table 5.2: Results when testing different hyperparameter combinations for the Improved Diffusion repository.

Learn Variance	Noise Scheduler	FID (1k samples)	FID (10k samples)
True	Linear	16.03	4.56
True	Cosine	20.87	9.07
False	Linear	21.33	9.20
False	Cosine	17.47	6.54

The Improved Diffusion paper [11] shows that learning variances with a cosine noise scheduler usually yields the better results on lower resolution images (32x32 and 64x64). We found that learning variances with a linear noise scheduler generated better results in our case, exhibiting the lowest FID of 4.56. Since neither noise scheduler performed strictly better, but depended on the choice of whether or not to learn variances, it is possible that our resolution (112x112) is close to the point where linear and cosine schedules perform equally well.

Nevertheless, we do see that one combination of hyperparameters performs marginally better than the rest. We therefore choose to move forward training our different diffusion models using the Improved Diffusion repository as a base, with a linear noise scheduler and learning the variances of the reverse diffusion process.

5.1.2 Ideal sample size

Figure 5.1 shows convergence of FID starting at around 10,000 synthetic samples for our echocardiogram dataset. Generating samples of sizes larger than 10,000 images should be safe with little FID improvements as the sample grows larger. Larger samples require considerably more time and computational power to generate. However, we want to make sure we are not on the starting point of converge but have some safe margins. It is also reasonable to sample a set of size comparable to our largest real subset of about 32,000 images. Therefore, 32,000 strikes a good

balance between FID convergence and is a reasonable size for further experiments without resulting in unreasonable computation time required.

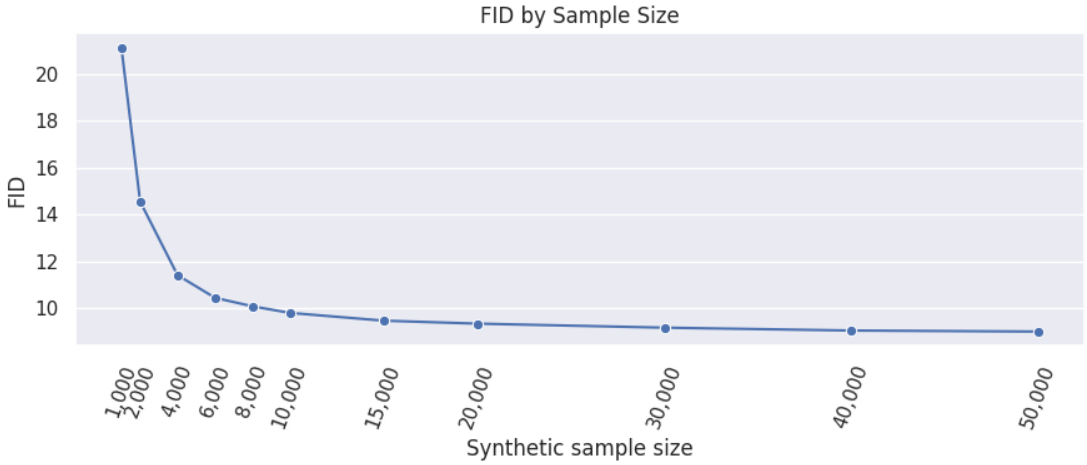


Figure 5.1: Evolution of FID as the number of evaluated synthetic images increases.

A key assumption here is that, independent of how long a diffusion model is trained for and on how many real images it was trained with (within reason), the convergence performance of different synthetic sample sizes holds when evaluated against a fixed set of real images. In other words, 32,000 should be a safe minimum number of images to generate in order to compute a reliable and representative FID.

5.1.3 Time needed to train Diffusion model

Our results, comparing FID and training update steps are displayed in Figure 5.2. These show that, for our chosen training configurations, FID seems to converge somewhere between 63,000 steps (FID 12.79) and 72,000 steps (FID 12.20). This corresponds to just under 10 hours of training on four A100 GPUs. It is worth noting that upon visual inspection the images generated after only 9,000 and 18,000 steps were hardly distinguishable from noise, so we leave their corresponding FIDs (379.36 and 400.78, respectively) out of the plot to allow for a more informative scale. The evolution of synthetic images during diffusion training at increase update steps is seen in Figure 5.3.

Figure 5.2: Evolution of FID throughout diffusion training.

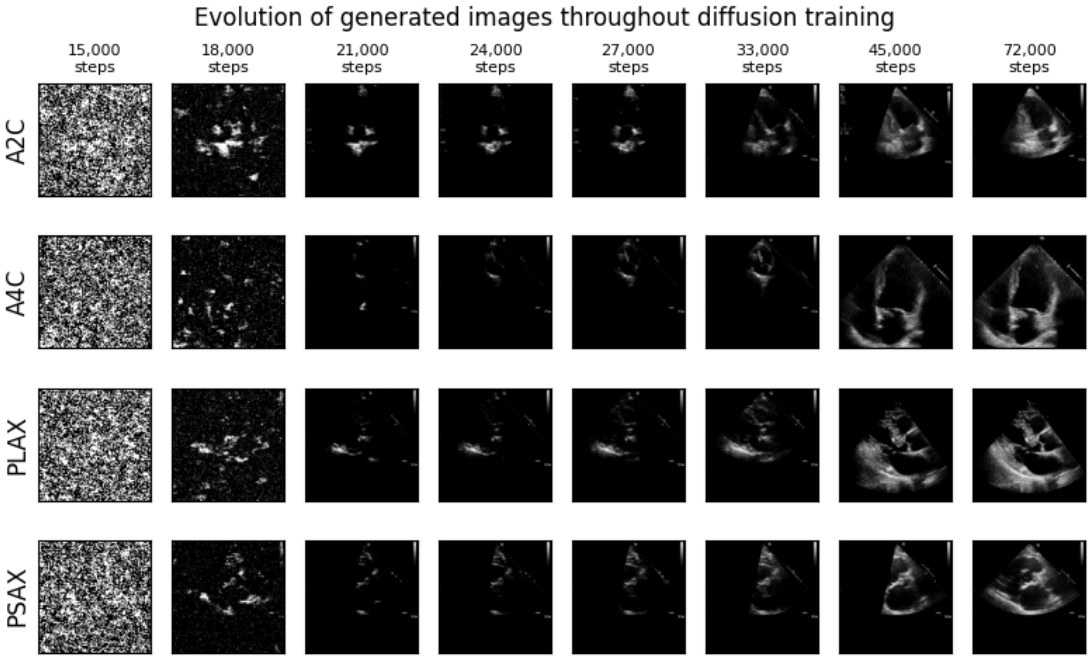


Figure 5.3: Evolution of generated images throughout diffusion training.

5.1.4 Qualitative Evaluation

This section showcases some randomly sampled real images from the training dataset and synthetic images generated with the Improved Diffusion model having the lowest FID performance (learning variances with a linear noise scheduler). Two sets of twelve images each will be shown:

1. Figure 5.4: Only real images.

2. Figure 5.5: An even mix of unlabeled real and synthetic images.

The even mix of real and synthetic images is also available in the Appendix (Figure A.1) where it includes the real/synthetic label along with the class label (A4C, PLAX).

Figure 5.4: Real echocardiogram images.

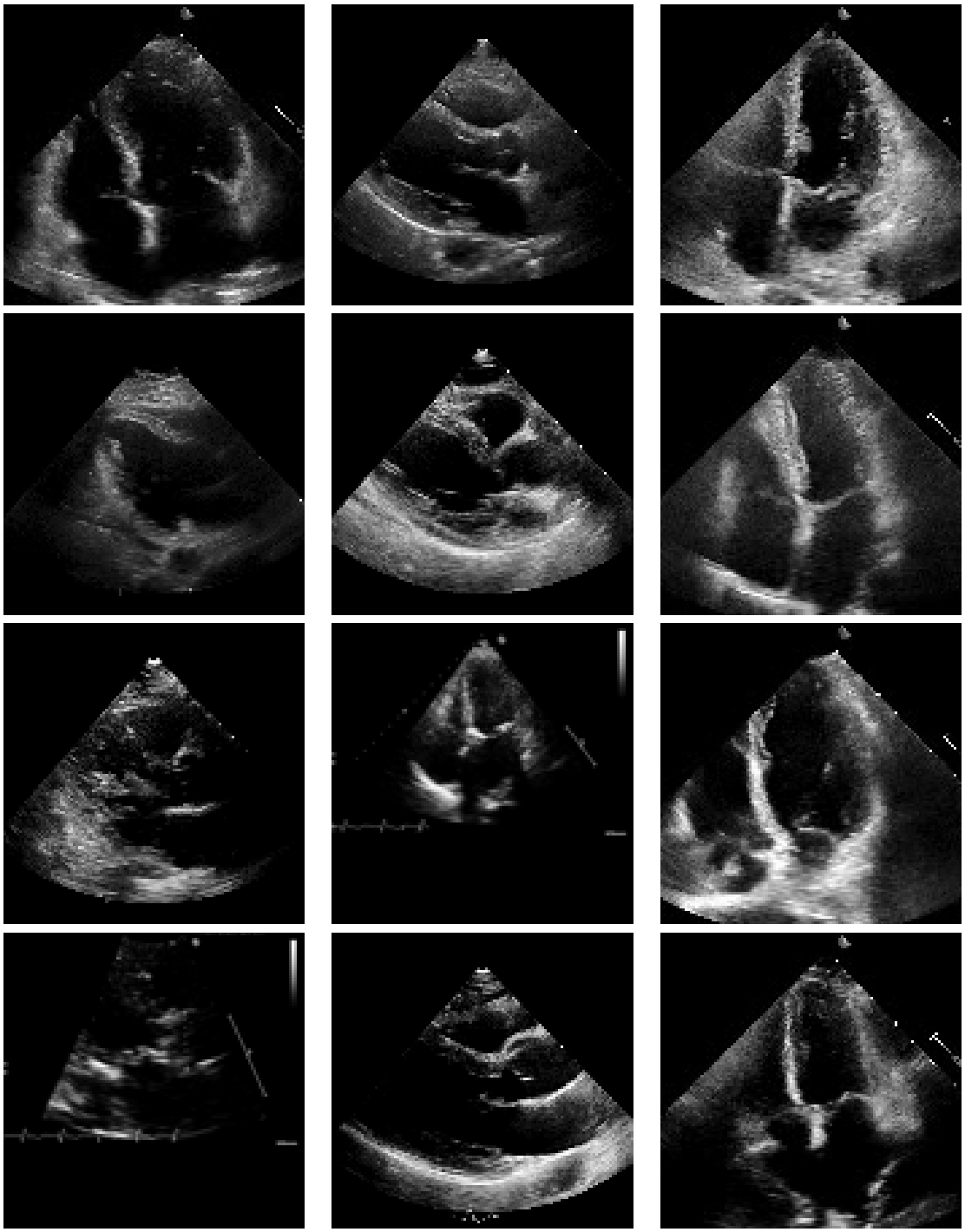


Figure 5.5: Assessment task: try to distinguish the 6 synthetic from the 6 real echocardiograms.

5.1.5 Survey: Can you spot the real echocardiogram?

A total of 15 individuals participated in the survey, while this sample size is not large enough to draw statistical conclusions, it is valuable from a qualitative perspective to start to understand how medical experts perceive the synthetic images generated.

Medical specialism of participants spanned Cardiology (2), Anesthesiology (9), Primary care (1), Infectious diseases (1), Surgery (1), and Ear, Nose and Throat (1) (Figure 5.6). Participants were all either currently, working to get their medical or specialist licence, working as specialist, or retired from the medical profession. The years of work experience post medical school among participants varied from 1 to 35 years (Figure 5.7). As expected, most had some experience with echocardiograms, but greater experience with ultrasound images in general. A breakdown is visible in Figure 5.7 and 5.8.

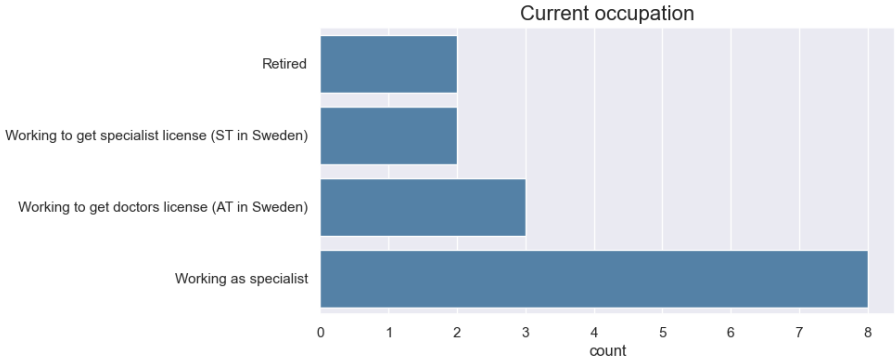


Figure 5.6: Current occupation.

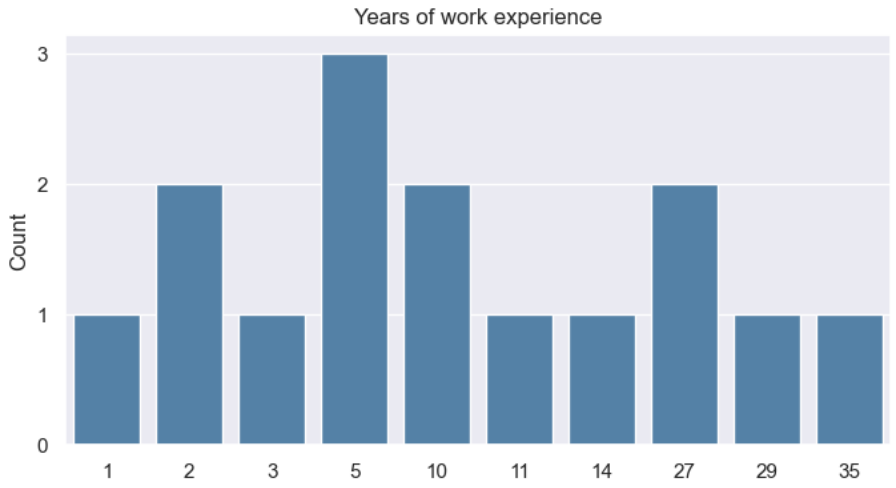
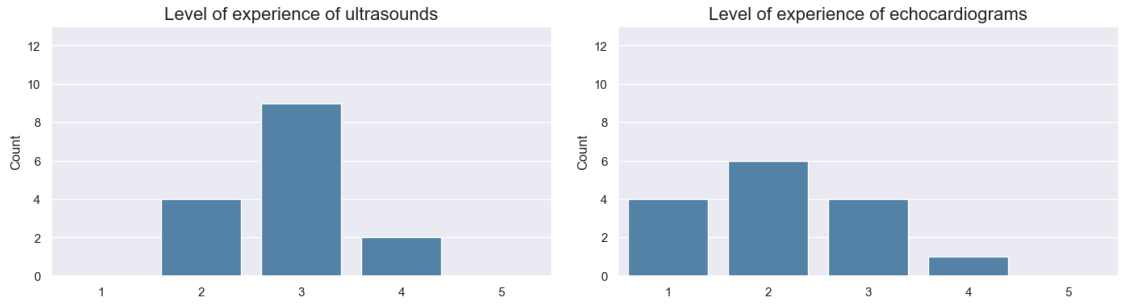


Figure 5.7: Work experience post medical school.



(a) Ultrasounds (b) Echocardiograms

Figure 5.8: Survey questions about the level of experience with ultrasounds and echocardiograms.

Since real and synthetic images were presented in pairs there was a 50% chance of answering each question correctly when guessing at random. The expectation was that overall performance would be higher than 25/50. Both average and median of our small sample was 21. Figure 5.9 depicts a break down of participant's scores. It is noteworthy to mention that low resolution images (112x12 pixels) used in this project, and presented in the survey, are not what medical experts would be exposed to and have experience of during medical practices.

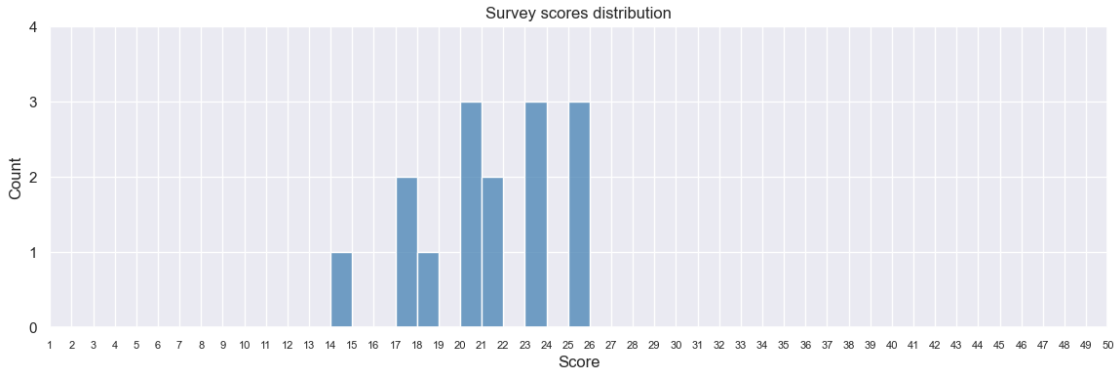
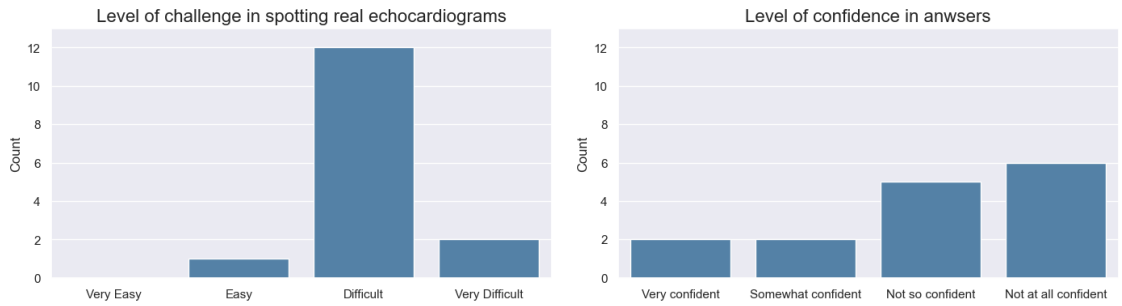


Figure 5.9: Survey performance breakdown.

When asked prior to seeing any image pairs, a majority of participants anticipated that it would be difficult to distinguish real from synthetic echocardiograms. Post answering the 50 image pair questions, participants were asked to rate their general level of confidence in their answers. Participating cardiologists were both "Not so confident", along with the majority as seen in Figure 5.10.



(a) Before (b) After

Figure 5.10: Survey questions asked before and after looking at the 50 image pairs

A question about what the general strategy was to distinguish real echocardiograms from fake ones was asked as part of the survey to gain some intuition behind features that may have stood out as identifiers. Examples are given here:

- "I tried do see which one was most anatomically correct."
- "Difficult to distinguish since it depends on depth on screen, the cut through the heart, the movement of the heart is important to understand the picture and the contrast can be made high or low manually and hence affecting the picture."
- "Difficult to tell without movement"
- "Anatomy relative to angle and resolution/grayscale relative to angle/anatomy. But really hard to decide.."
- "Image quality in the periphery, tissue smoothness"

A full outline of all questions asked in the survey is found in the Appendix A.3.

5.2 View Classification

This section provides a comprehensive account and analysis of the results obtained from our different experiments on view classification. The evaluation encompasses the examination of multiple state-of-the-art architectures, utilizing subsets of both real and synthetic data, followed by feature visualizations of these datasets. Moreover, an assessment regarding the impact of synthetic data on minority class performance is provided, along with the performance of real data augmented to varying degrees with synthetic data. Furthermore, an exploration is undertaken to assess the performance of the selected model on final test sets derived from known and unknown distributions (seen and unseen during training). Finally, certain limitations of the study and results are acknowledged.

5.2.1 Initial architecture exploration

Initial classification results range from 0.843 to 0.988 AUPRC with VGG16, ResNet18, DenseNet121 and InceptionV3 architectures. These were all trained on the full 32k training dataset of real images (including A2C, A4C, PLAX, & PSAX views) and evaluated on a validation set coming from the same distribution.

Table 5.3 details the results per model. Training times across the different architectures vary considerably. When trained from scratch (Pretrained=False), both DenseNet121 and InceptionV3 take significantly longer to train than VGG16 and ResNet18.

Table 5.3: Initial analysis of classification architectures when trained on the largest subset of real data (32k).

Architecture	Pretrained	AUPRC	F1	Loss	Time
VGG16	True	0.892	0.829	0.250	18m
DenseNet121	True	0.895	0.840	0.251	20m
ResNet18	True	0.890	0.818	0.267	14m
InceptionV3	True	0.843	0.787	0.325	36m
VGG16	False	0.988	0.960	0.091	28m
DenseNet121	False	0.982	0.958	0.077	51m
ResNet18	False	0.983	0.951	0.086	23m
InceptionV3	False	0.978	0.940	0.092	57m

Based on these results, it seems more promising to train networks from scratch as opposed to using pretrained weights, with greater performance across AUPRC and F1-Score. Due to the additional resources needed to train, along with the lack of performance increase, InceptionV3 is ruled out from further analysis.

Performance across the remaining model architectures was similar when trained on the full training set. We conduct further experiments on the different subsets of real data, as well as on their corresponding synthetic datasets (see Figure 4.5), in order to decide on a model of choice for final testing.

5.2.2 Experiments with real and synthetic subsets

Shown in Table 5.4 are the results of training the different classifiers from scratch on the real and synthetic subsets. All models were evaluated on the same fixed validation set, and the table reports the resulting validation AUPRC, F1-score, and training time (measured in minutes). It is worth noting that the validation set was not seen during training of either diffusion or classifier models.

Table 5.4: Results of initial classification architecture exploration when trained from scratch. For synthetic data, the "Size" column refers to the size of the real subset used for training the diffusion model from which the synthetic data was sampled.

Data	Size	Architecture	AUPRC	F1	Time
Real	1k	VGG16	0.6409	0.3894	4m
		DenseNet121	0.7411	0.5478	3m
		ResNet18	0.7784	0.6514	2m
	2k	VGG16	0.8590	0.7802	5m
		DenseNet121	0.8469	0.7517	5m
		ResNet18	0.8807	0.8013	4m
	4k	VGG16	0.9278	0.8717	6m
		DenseNet121	0.9139	0.8522	8m
		ResNet18	0.9298	0.8723	4m
	8k	VGG16	0.9682	0.9302	9m
		DenseNet121	0.9507	0.9094	16m
		ResNet18	0.9584	0.9194	6m
	16k	VGG16	0.9835	0.9449	15m
		DenseNet121	0.9716	0.9367	29m
		ResNet18	0.9742	0.9423	14m
	32k	VGG16	0.9881	0.9596	28m
		DenseNet121	0.9817	0.9576	51m
		ResNet18	0.9829	0.9510	23m
Synth	1k	VGG16	0.9307	0.8731	29m
		DenseNet121	0.9120	0.8666	51m
		ResNet18	0.9046	0.8399	23m
	2k	VGG16	0.9528	0.8987	29m
		DenseNet121	0.9525	0.9005	54m
		ResNet18	0.9333	0.8712	26m
	4k	VGG16	0.9600	0.9080	29m
		DenseNet121	0.9553	0.9127	52m
		ResNet18	0.9549	0.8986	23m
	8k	VGG16	0.9746	0.9231	29m
		DenseNet121	0.9574	0.9089	53m
		ResNet18	0.9671	0.9190	21m
	16k	VGG16	0.9840	0.9484	29m
		DenseNet121	0.9703	0.9314	54m
		ResNet18	0.9791	0.9534	20m
	32k	VGG16	0.9860	0.9532	29m
		DenseNet121	0.9784	0.9531	51m
		ResNet18	0.9805	0.9520	23m

Results shown in Table 5.4 are further visualized in Figure 5.11, showing classifier performance as we increase the size of the real data available. Dashed lines correspond to classifiers trained on the different sizes of real subsets. Solid lines correspond to the classifiers trained on synthetic subsets (all of size 32k images), which themselves are sampled from diffusion models trained on the different sizes of real subsets.

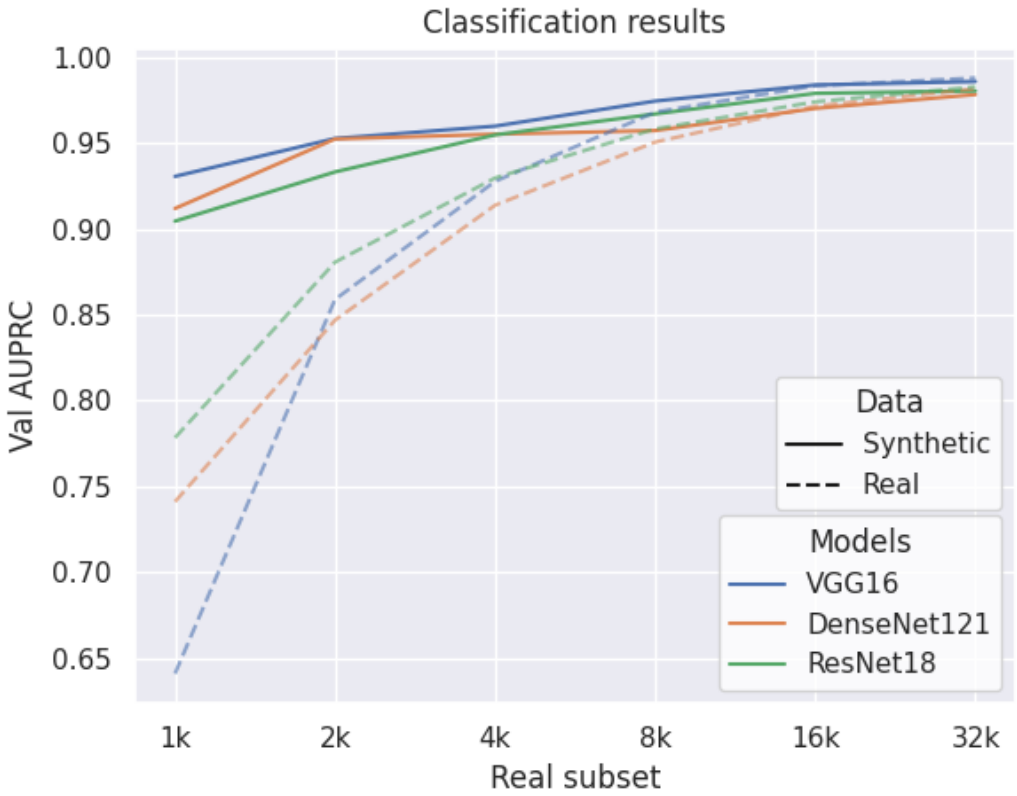


Figure 5.11: Classification results: on real (dashed lines) and synthetic (solid lines) data, for varying sizes of real data available.

As expected, classifiers trained on real data see significant performance gains as the size of the real data available grows larger. While this is also true for classifiers trained on synthetic data, these start off already performing quite well when the size of available real data is limited, exhibiting less dramatic performance gains from increasing the quantity of real data available.

A crucial observation that emerges from this experiment is that the use of synthetic data for training purposes appears to offer the greatest advantages when the amount of real data is severely limited. In cases where sufficient real data is available, it is more reasonable to train on it directly rather than resorting to the generation of synthetic data for training purposes. On the other hand, our findings suggest that when real data is scarce a viable strategy would be to employ the limited available real data to train a diffusion model, which can then be leveraged to generate a

comparatively large dataset of synthetic images for subsequent classifier training. This approach has the potential to yield substantial gains in classifier performance.

Another noteworthy observation we can draw from these results is how the relative performance of different architectures on small datasets may not be indicative of their performance on larger datasets. Specifically, when examining the models trained on real data, the graph indicates that ResNet18 performs the best on the smallest dataset, while VGG16 performs considerably worse. However, at the right end of the graph, VGG16 delivers the best performance when trained on the largest real subset. In contrast, when looking at the models trained on synthetic data, we see that even under the smallest real subset, the corresponding synthetic data is sufficient to demonstrate that VGG16 outperforms the other architectures. This pattern persists across the various increases in real subset sizes. One potential application of synthetic data might thus be as an early form of architecture selection in situations where very limited real data is initially available, but more is expected to be obtained in the future.

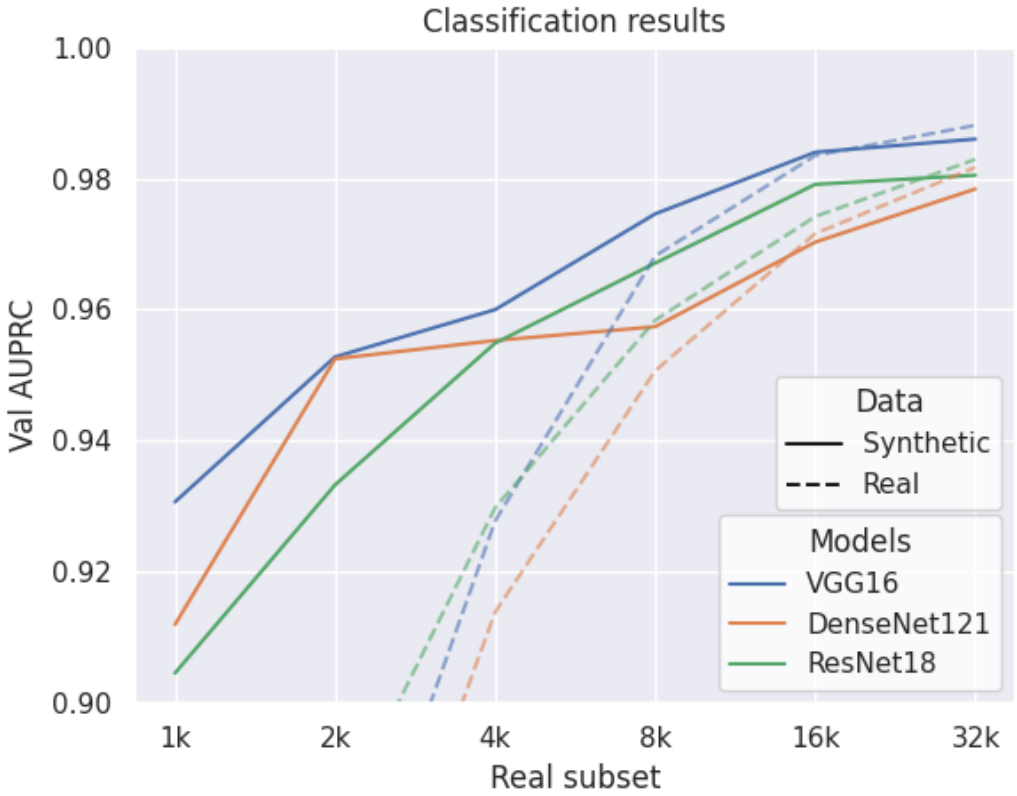


Figure 5.12: Classification results: on real (dashed lines) and synthetic (solid lines) data, for varying sizes of real data available.

Lastly, it is worth noting that for all three evaluated classifier architectures, their respective performances on real versus synthetic data are strikingly similar once we reach the larger real subsets. In the scenario where there are 32,000 real images available, the performances of any given architecture on real and synthetic data

are almost identical, with real data consistently displaying only a slight advantage. Figure 5.18 provides a zoomed-in view of the results under these conditions. While it starts off with the lowest AUPRC on the smallest real subsets, VGG16 surpasses the other architectures on real subsets above 4k. Most importantly, it outperforms the other architectures on all synthetic sets. Based on these results, the preferred architecture of choice is VGG16. This is the model we move forward with for subsequent experiments, unless otherwise stated.

5.2.3 Cluster visualization of image features

In this section we display how images of different views and coming from different datasets cluster together. To that end, we extract features from both real and synthetic images by running them through our VGG16 view classifier, trained on the full training set of real images. We extract the activations of the second to last layer, and perform t-SNE dimensionality reduction on them to reduce them to two components. Throughout this process, each input image is reduced from 12,544 dimensions (pixel values: 112x112 resolution, 1 color channel) to 4,096 dimensions (features extracted from our trained classifier), and subsequently to 2 dimensions (output of t-SNE algorithm).

Figure 5.13: Image clustering by dataset and view. Real and synthetic images are passed through our VGG16 model trained on the full training set of real images. The activations of the second to last layer are extracted and reduced to 2 dimensions by running t-SNE. Full sized versions of these plots can be found in Appendix A.4.

Figure 5.13 displays the results for real and synthetic images, color-coded by dataset (for real images only) and by view. The left-most plot shows that real images can be well separated by their dataset of origin (Dynamic, LVH, TMED2). In the center plot, we further see that real images also form well defined clusters according to view. The right-most plot shows that the generated synthetic images mimic the view clusters evidenced in real images quite well. This indicates the synthetic images do indeed exhibit view-specific characteristics present in the real data. This finding goes in line with our survey results, where participants had a difficult time telling real from synthetic echocardiogram images apart.

From Figure 5.13, it is evident that real images are clustered not only on a view basis, but actually on a dataset-view basis. For instance, PLAX images do not form

one unique cluster, but instead form separate clusters for PLAX images belonging to the LVH dataset and PLAX images belonging to the TMED2. This is evidence that the classifier, and more importantly also the diffusion model, are both learning not only view-specific features but additionally dataset-specific features as well. This raises the question of whether by conditioning the diffusion model on both view and dataset it could learn to separate view-features from dataset-features. We return to this point when we discuss interesting avenues for future research.

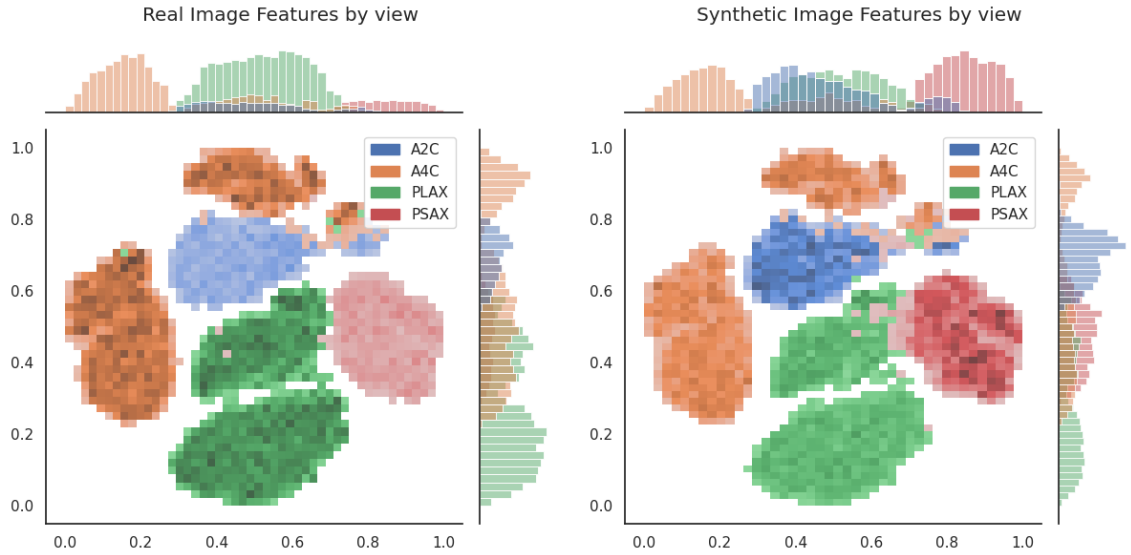
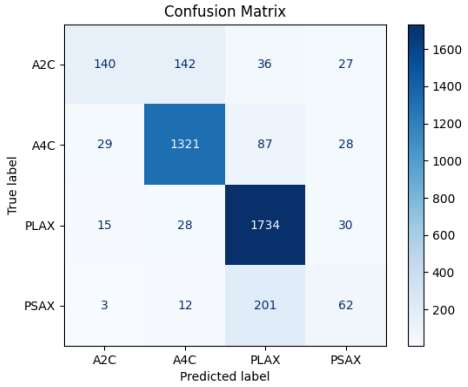


Figure 5.14: 2D-Histograms showing the distribution of real and synthetic images per view. Full sized versions of these plots can be found in Appendix A.4.

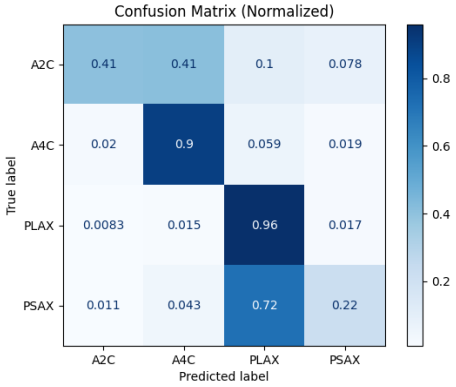
Figure 5.14 evidences the rebalancing effect achieved through the use of our synthetic data. The minority classes (A2C and PSAX), coming only from one source dataset (TMED-2), each have a single cluster. Looking at synthetic images (right plot), since we generate a balanced synthetic dataset, 50% of all synthetic images belong either in the blue or red clusters. Having a smaller area to cover, the distribution density in those minority clusters is higher than in the majority clusters (seen as darker blue and red spots). We see a similar, but opposite situation in the real data (left plot). A vast majority of real images are either A4C or PLAX view, and therefore their respective clusters exhibit a higher density than that in the minority clusters.

5.2.4 Impact of synthetic data on minority classes

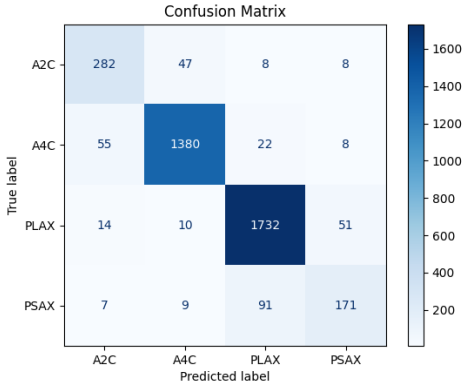
Figure 5.15 shows the confusion matrices (unnormalized & normalized) of our ResNet18 model trained on real and synthetic data. Both models were evaluated on the same validation set. It is worth noting that both the real training set and the validation set exhibit the same degree of class imbalance. One of the key advantages of using a diffusion model to generate synthetic data is that it enables us to construct a balanced dataset of synthetic images. This helps address the imbalance in the original dataset, as the classifier is now allowed to train on a balanced dataset, which results in significantly improved performance across the minority classes (A2C & PSAX).



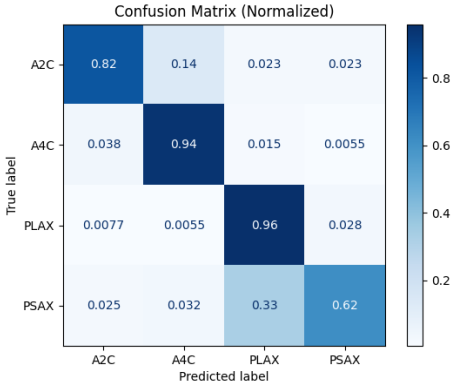
(a) Confusion Matrix: ResNet18 trained on real data.



(b) Normalized Confusion Matrix: ResNet18 trained on real data.



(c) Confusion Matrix: ResNet18 trained on synthetic data.

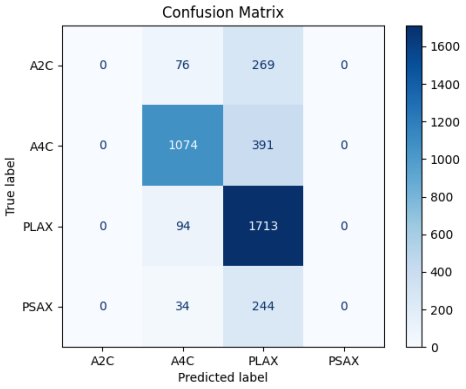


(d) Normalized Confusion Matrix: ResNet18 trained on synthetic data.

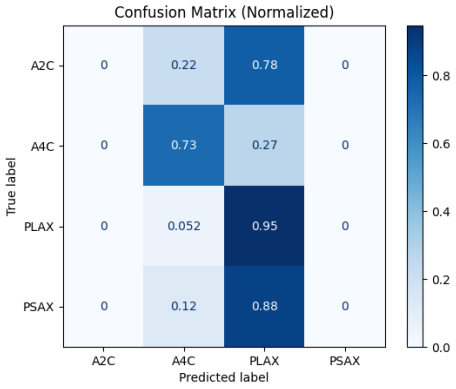
Figure 5.15: ResNet18 confusion matrices under real (top row) and synthetic (bottom row) training data, evaluated on the same validation set. A darker blue color indicate greater performance where minority classes A2C and PSAX are much darker on the diagonal for synthetic training data.

When trained solely on real data, a substantial number of examples from the minority classes are often misclassified, resulting in poor overall performance. However, by training on a balanced dataset of synthetic images, we can alleviate this problem to a considerable extent. Although the results are not perfect, they demonstrate a significant improvement in the accurate classification of the minority classes, as evidenced by the confusion matrices.

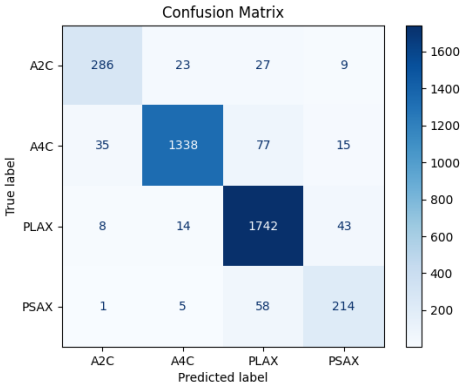
For completeness, we present the corresponding confusion matrices for VGG16 in Figure 5.16. In line with overall classification results displayed in Figure 5.11, VGG16 shows an even larger boost in classifier performance than ResNet18 when moving from training on real data to training on synthetic data.



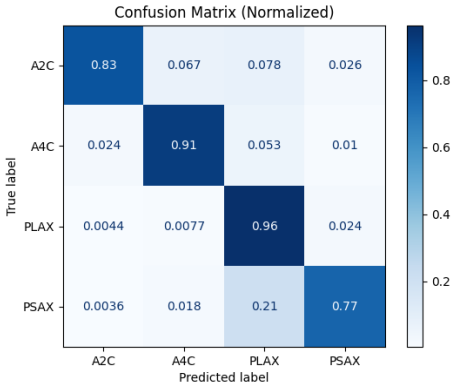
(a) Confusion Matrix: VGG16 trained on real data.



(b) Normalized Confusion Matrix: VGG16 trained on real data.



(c) Confusion Matrix: VGG16 trained on synthetic data.



(d) Normalized Confusion Matrix: VGG16 trained on synthetic data.

Figure 5.16: VGG16 confusion matrices under real (top row) and synthetic (bottom row) training data, evaluated on the same validation set.

5.2.5 Synthetic augmentation of real data

Obtained results when augmenting real data with synthetic samples are illustrated in Figure 5.17. The x-axis corresponds to the fraction of synthetic images added to the real subset at each step, relative to the size of the real subset.

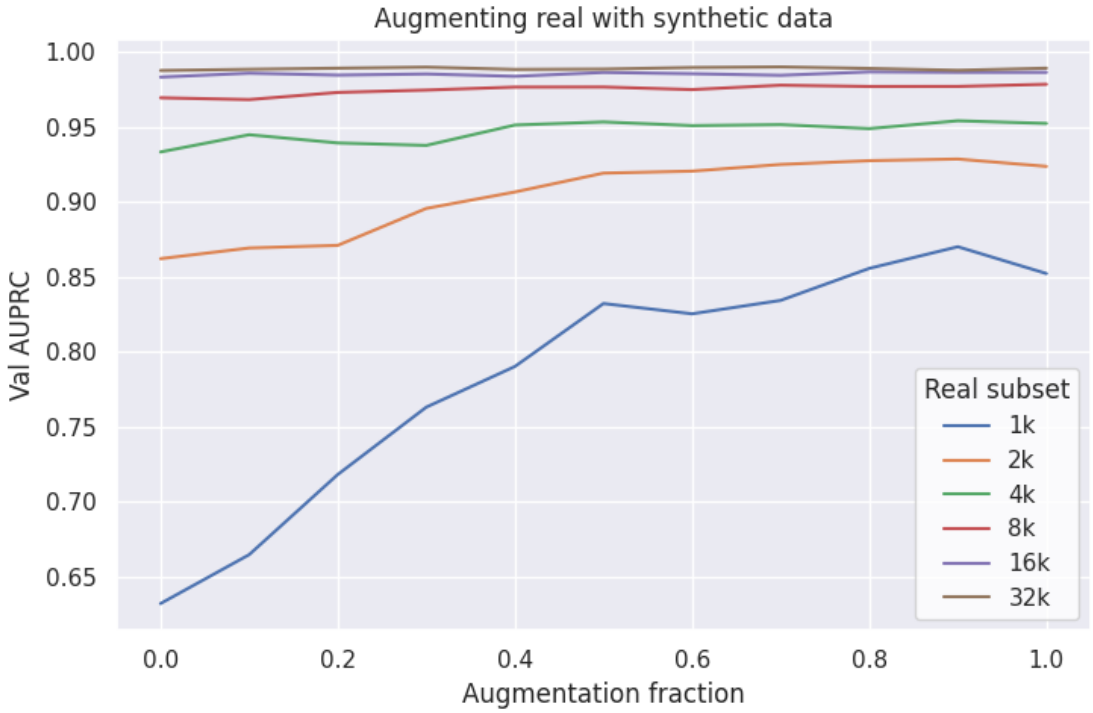


Figure 5.17: Classifier performance trained on real subsets augmented with increasing fractions of synthetic images.

The figure once again highlights how synthetic data has the most impact under conditions where real data is the most limited. Augmenting the smallest real subset shows the highest increase in performance as it is augmented by synthetic data. Conversely, the performance on the largest subsets is relatively unaffected by the addition of synthetic data, at least up to the point of doubling the size of the training set (augmentation fraction of 1.0).

For any given subset, once we reach an augmentation fraction of 1.0 the training set has doubled in size. For example, the 1k real subset augmented with synthetic data with an augmentation fraction of 1.0 results in a training set of 2,000 images. We can then compare the performance of this mixed dataset of 2,000 images (50% real and 50% synthetic) to the real subset of the same size (2k, augmentation fraction 0.0). Val AUPRC for the real subset is still above that for the mixed dataset, even if only slightly. Figure 5.17 shows that this pattern holds for all pairs of the same size (e.g. 2k with 1.0 vs 4k with 0.0, both of size 4,000 images). The conclusion we can draw from this observation is that, while synthetic data holds the potential for significant improvements in classifier performance under severe real data constraints, real data seems to consistently outperform a mix of real and synthetic data of the

same size. In other words, while synthetic data is a valid and especially cost-effective alternative when real data is limited, acquiring more real data is preferable whenever it is a realistic option.

Finally, out of all classifiers trained and evaluated, the one trained on a mixed dataset of 32,000 real and 22,400 synthetic images (32k with 0.7 augmentation fraction) performs the best, achieving a validation AUPRC of 0.990284. We therefore select this particular trained classifier as our final model, and the one we will test on two different holdout test sets in the next section.

5.2.6 Synthetic data compared to basic upsampling method

The comparisons of AUPRC performance for the four classifiers trained on the 1,000 real subset, the two upsampled real subsets containing 1,872 and 32,000 real images each, and the 32,000 synthetic image dataset is visible in Figure 5.18. The real subset of 1000 images is clearly underperforming, while a small effort in upsampling classes to the same level as the most frequent class (PLAX) generated a great performance boost from 0.64 to 0.86 AUPRC. Surprisingly, extending the dataset to 32,000 images with the simple upsampling technique generates just as good performance as with the synthetic dataset, both having 0.93 AUPRC.

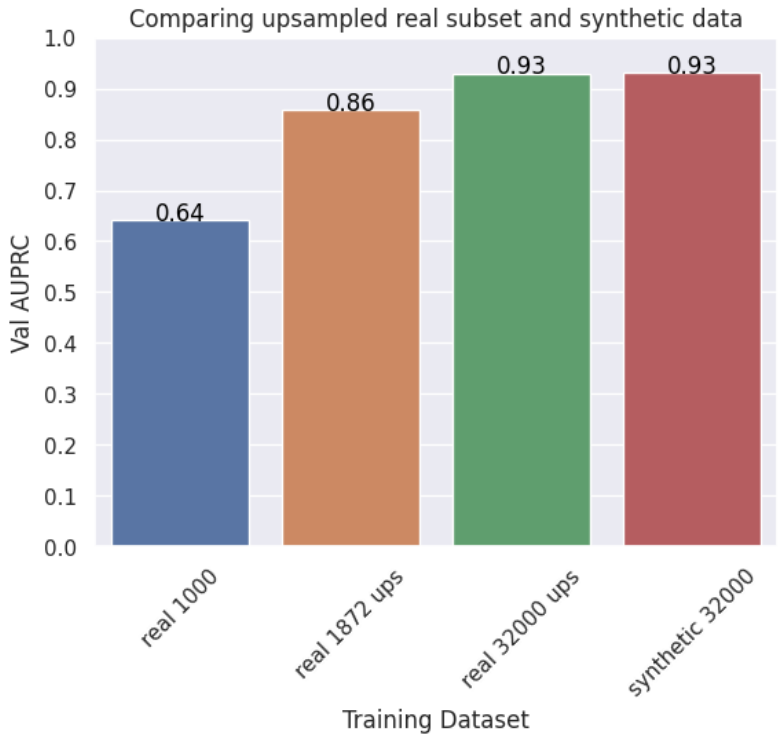
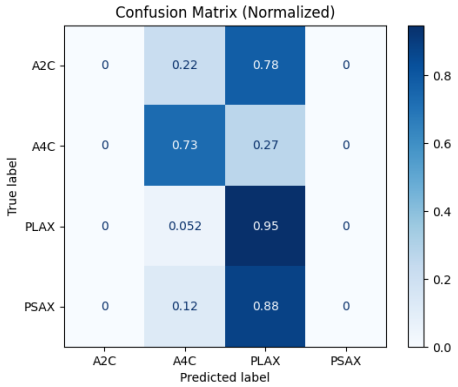


Figure 5.18: Classification results when evaluated on the same validation set for real, real upsampled and synthetic datasets.

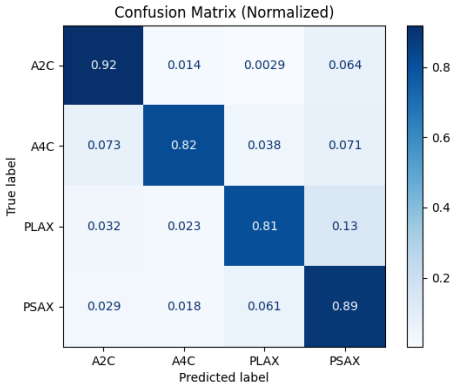
Analysing the confusion matrices (Figure 5.19) support that a minimal effort in upsampling the 1,000 real subset to 1,872 images result in a significant increase in

performance driven by minority classes A2C and PSAX. These move from having no correct predictions to 0.92 (A2C) and 0.89 (PSAX) percent predicted correctly. Having multiple copies of images present in minority classes, as opposed to none (PLAX), or few (A4C) seem to benefit the model predictions for minority classes more than for majority classes. As seen in Figure 5.19 the percentages of correct predictions are greater for minority classes than for majority classes.

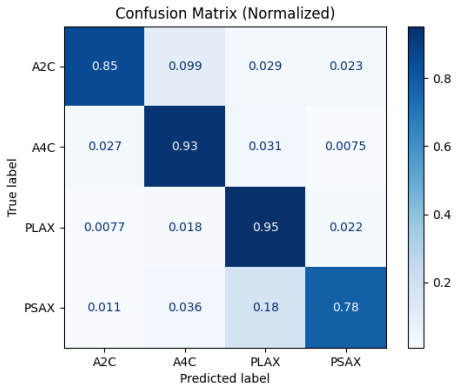
On the contrary, when all classes were substantially upsampled to create a dataset of 32,000 images, majority classes (A4C and PLAX), again had the most items predicted correctly (0.93 and 0.95 respectively) as opposed to minority classes A2C and PSAX. Finally, comparing the large upsampled dataset to the synthetic dataset shows similar performance overall and across classes.



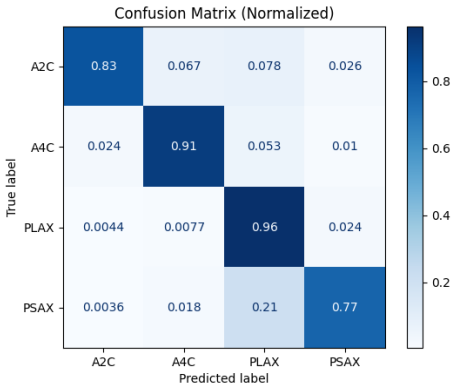
(a) Normalized Confusion Matrix: 1,000 real subset.



(b) Normalized Confusion Matrix: 1,000 real subset upsampled to a total of 1,872 images evenly balanced between classes.



(c) Normalized Confusion Matrix: 1,000 real subset upsampled to a total of 32,000 images evenly balanced between classes.



(d) Normalized Confusion Matrix: 32,000 synthetic dataset (evely balanced) generated from a diffusion model trained on the 1,000 real subset.

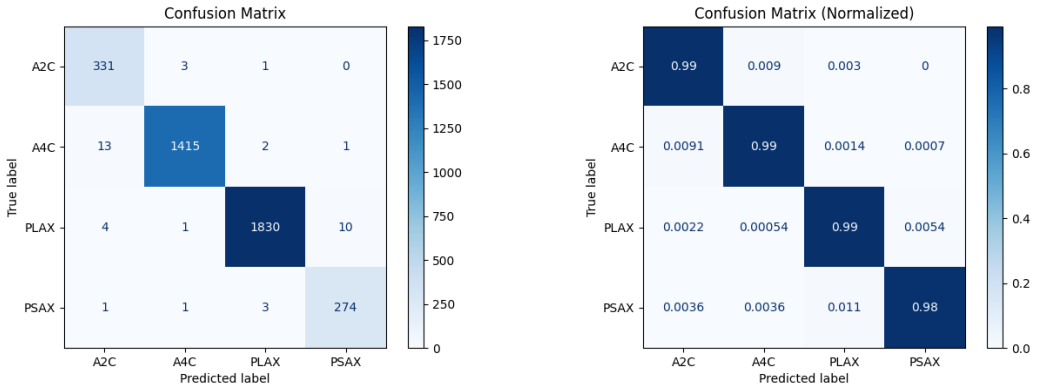
Figure 5.19: Normalized confusion matrices showing the performance per dataset train with VGG16 architecture and evaluated on the validation set. A darker blue color indicate greater performance.

These results highlight that a larger volume of images and balanced data per class are beneficial to the classifier, despite seeing images copied multiple times. With no difference between the larger 32,0000 upsampled and synthetic dataset, results may also be indicative of that the diffusion model generate synthetic images with small variations from the real images seen during training, meaning there is a risk of seeing images that could be similar to the original images or even close copies.

5.2.7 Final Testing

5.2.7.1 Evaluation on Test Set

The model is evaluated on the primary test set, achieving an AUPRC of 0.9982. Figure 5.20 below displays the corresponding unnormalized (left) and normalized (right) confusion matrix. We see that the model performs well on each and every class.



(a) Confusion Matrix

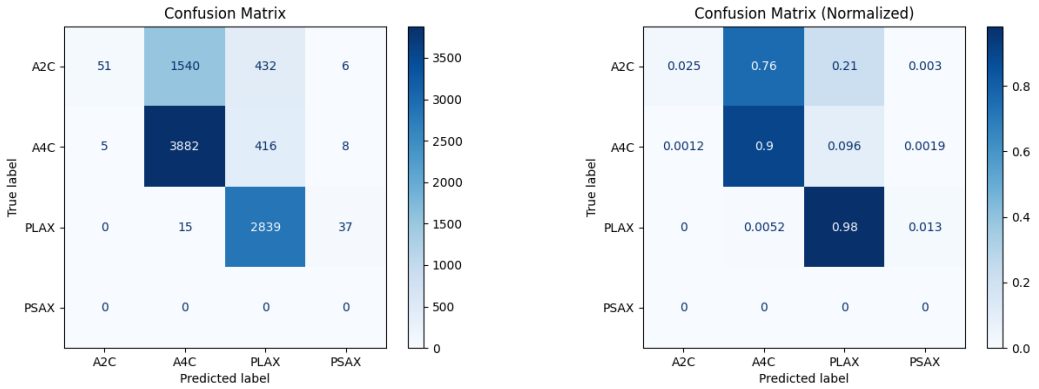
(b) Normalized Confusion Matrix

Figure 5.20: Confusion matrices for final model as evaluated on the holdout test set.

5.2.7.2 Domain Shift Analysis

Figure 5.21 below shows the unnormalized (left) and normalized (right) confusion matrix after evaluating our final model on the secondary holdout test set (CAMUS-Unity images), coming from a different distribution than the training data (LVH, Dynamic, TMED2).

Although the classifier was trained on a dataset consisting of 4 classes, the test set only contains 3. This means it is possible for the classifier to predict that an image belongs to PSAX-class that is not present in the test set. While this indeed happens for a few examples, it is not a major issue overall. Naturally, it would be ideal to use a test set containing the same number of classes as the model was trained on to evaluate its performance under domain shift. Due to the constrained availability of source datasets, we are forced to utilize a domain shift test set that encompasses solely three out of the four classes on which the classifier was trained.



(a) Confusion Matrix

(b) Normalized Confusion Matrix

Figure 5.21: Domain shift analysis: Confusion matrices for final model as evaluated on the holdout CAMUS-Unity test set.

Figure 5.21 shows that the classifier does not generalize well to data from distributions different than the one it was trained on. We can analyze the performance on each of the three classes present in this test set:

- A2C: An overwhelming majority of A2C images were misclassified as A4C, and PLAX. We can hypothesize that this issue traces back to limitations in the training set, wherein all A2C images were sourced exclusively from the TMED2 dataset. Consequently, the model struggled to effectively differentiate between genuine A2C features and features specific to the TMED2 dataset. As a result, when the model encounters A2C images from other datasets during evaluation, it does not recognize the features that it learned to relate to A2C, as these were likely features more related to the TMED2 dataset itself, rather than the A2C view. Some potential examples could be zoom level, cone width, cone angle, brightness, etc.
- A4C: This view seems to perform best at first glance, a large majority predicted correctly. While it is true that the model seems to correctly identify A4C images, it also produces a great many false positives for this class. A large amount of A2C images are misclassified as A4C.
- PLAX: These images are in majority predicted to be PLAX, with few false negatives. Interestingly, we can draw a parallel to the A2C case. The training set contained PLAX images coming from 2 datasets, as opposed to just 1 for A2C. Consequently, the issue of false negatives is low in PLAX than in A2C.

These results highlight the importance of having as varied a training set as possible, ideally having an even distribution of classes sourced from different datasets. This should allow the model to differentiate between view-specific and dataset-specific features, and therefore generalize better. It would be interesting for future work to explore how synthetic data generation might help address this point when an even balance is not naturally present in the available real data. Potentially, a diffusion model could be trained conditioned on both view and dataset, so it could then

generate images of views in the overall style of a dataset that does not include them. In order to be able to properly learn to differentiate between view and dataset features, it is likely that the model would need to be trained on datasets containing at least two views each, and view coming from at least two datasets each. If any view comes from only a single dataset, or if any dataset contains only a single view, then it would likely be impossible for the model to learn the distinction between view-specific and dataset-specific features.

5.3 Limitations

The results from this study may not be applicable beyond the scope of the case outlined in this report. This means that the model performance documented may not be robust to generalize well to data collected in a setting different to the datasets used. Additionally, the data used has been limited to still images coming from image datasets and frames selected from video sequences. In routine clinical care, video data is often used to go back and forth over a sequence to help in deciding the accurate view.

LVH and Dynamic datasets used have been limited to one view only, namely PLAX and A4C respectively. This phenomenon suggests that a classifier may learn features that are specific to a particular dataset, rather than capturing inherent characteristics of the view itself. Consequently, this can have a detrimental effect on the model's robustness when confronted with the same view originating from a different dataset.

It would be interesting to generate synthetic datasets that maintain the class imbalance of the original data, and evaluate the impact on classifier performance, if any. Leveraging synthetic data has at least two separate benefits: 1) increasing training set size and 2) improving balance between classes. We have not studied these two effects in isolation from each other.

Finally, the synthetic data that was generated was balanced by classes while the data it was created from, when training a diffusion model, was imbalanced. This is likely to result in a variability similar to the minority classes found in the original data. This is partially seen in the clustering of synthetic images minority classes being more dense within the same area as the fewer original images coming from the minority classes.

6

Conclusion

Based on the results gained from the experiments carried out and presented in Section 4 and 5 we are now returning to answer our two defined research questions:

1. Can diffusion models generate synthetic echocardiograms that look realistic enough to become indistinguishable from real echocardiograms to a human expert?

   The majority of medical experts who participated in the survey struggled to distinguish between real and synthetic echocardiograms. Furthermore, this difficulty in telling real and synthetic echocardiograms apart does not seem to correlate with the years of experience among the experts. This finding suggests that the inability to distinguish between real and synthetic images is not solely influenced by the level of experience but rather by the fidelity of the synthetic echocardiograms themselves. It is important to acknowledge, however, that the survey responses were few, and predominantly obtained from experts with specializations other than cardiology. Consequently, the overall findings and conclusions drawn from the survey may not be fully representative of cardiologists as a whole.

2. Is it possible to improve view classification performance of real echocardiogram images with synthetic data?

   Our work shows that synthetic data can in fact improve classification performance over real data. In particular, we can draw two main conclusions on the subject based on the results of our work:

   - The extent of classification performance gained through the use of synthetic data is inversely correlated with the size of the available real dataset. In scenarios where the real dataset is relatively small, the generation and use of synthetic training data is particularly advantageous, resulting in notable classification performance. This finding underscores the potential of synthetic data as a valuable resource for compensating for data scarcity issues and mitigating the limitations imposed by limited real data availability. That said, further study into the circumstances under which this use of synthetic data is preferable to other existing methods, such as upsampling, is still required.

   - When a large amount of real data is available, it is preferable to train

a classifier directly on it rather than employing a generative model to generate synthetic training data. In such cases, the direct use of the available real data yields superior classification performance compared to the use of synthetic data. This finding highlights the importance of considering the quality and quantity of the available real data when determining the most effective approach for training a classifier, and suggests that reliance on synthetic data generation may be less beneficial, or altogether unnecessary, in situations where vast amounts of real data are readily accessible.

6.1 Contributions

When applying machine learning to medical sciences, the two major concerns that formed the basis for our work were the very common imbalance found in medical datasets and the privacy concerns related to patient data. The former is related to minority classes' general representation of rare conditions that are of utmost importance to be detected. The latter is concerned with protecting private health data that could harm or disclose private individuals if shared carelessly. The contributions of our work lie in the approach used to get around the shortcomings of medical data. These contributions are four-fold:

1. Training a diffusion model: The generative models trained allowed us to generate samples that closely resemble real echocardiograms. Under the appropriate regulatory frameworks, this saved model could be shared to be sampled from based on conditions similar to our application scenario yet with different requirements on volume of images per class to be generated.

2. Development of a synthetic dataset: We introduce a synthetic dataset specifically designed for our study. This dataset, created using recent generative techniques, encompasses a wide range of realistic A2C, A4C, PLAX and PSAX echocardiograms. It has the potential to be made available under appropriate licensing agreements, enabling other researchers to leverage it for further investigations and studies related to the field of echocardiography.

3. Addressing patient privacy concerns: Our proposed approach, novel to echocardiogram view classification, could mitigate patient privacy concerns by utilizing synthetic data, provided that no close copies are ensured. By employing synthetic data, which then ensures anonymity, we would overcome the challenges associated with using real patient data, thus safeguarding privacy while still enabling effective research.

4. Maximizing the utility of available real data: We explore the utilization of imbalanced real datasets to generate synthetic data via diffusion models. By incorporating these synthetic images into the classification training process, we address the issue of imbalanced classes, specifically by enhancing the prediction of minority classes. For underrepresented classes, the approach improved the classification performance for smaller real datasets. Hence, making the most of limited real data available.

By contributing in these areas, our research aims to advance the field of medical sciences where the standardization of Artificial Intelligence assisted tools is becoming more common and natural part of daily life and work. Our work also aims to contribute in the automation of echocardiogram view classification with an improved accuracy and consistency, and especially in regards to minority classes. The approach presented in this thesis could be implemented in clinical trials, such as AstraZeneca's, to enhance efficiency by saving resources and most importantly medical experts' time, allowing them to screen more patients.

6.2 Future Work

The field can further advance and refine the application of synthetic data generation techniques, ultimately by exploring the protection of patient privacy and validating diversity in synthetic data generated. Additionally, the work can be extended to video formats and as a means of early model selection. Suggestions related to future work are listed here:

1. Diffusion conditioned on dataset-view pairs: Our clustering analysis provided some evidence that our diffusion model learned not only view-specific characteristics to recreate, but also dataset-specific characteristics as well. This suggests that it might be possible to train a diffusion model conditioning on both views and datasets independently. If the diffusion model is able to learn these features independently from one another, then it could be possible to sample images with an arbitrary combination of view-dataset characteristics. This could potentially enable generating synthetic images of an echocardiogram view in the style of a dataset that does not originally contain it (e.g. A2C echocardiograms in a style similar to the LVH dataset). Consequently, one could generate a dataset that is balanced across view and dataset styles, likely aiding generalization.

2. Recreating original training data: An important direction for future work is to investigate whether or not the original training data can be reconstructed using the weights of the trained diffusion model, the synthetic data sampled from it, or when combined. This investigation holds significant implications in relation to patient privacy concerns. If successful, such reconstruction could potentially compromise patient confidentiality and privacy.

3. Measuring variability of synthetic datasets: Future work should study the amount of variation in synthetic image datasets, particularly when minority classes are of high importance. This is crucial as minority classes may not exhibit as much diversity as the majority classes. Understanding and quantifying this variation could contribute to further improving the overall effectiveness of the synthetic data generation process.

4. Extending studies to video sequence data: Another avenue for future research is to expand the scope of the studies to include the generation and analysis of video sequence data. By incorporating temporal information, the application of synthetic data in video-based tasks can be explored, opening up new

possibilities for improving classification and prediction results in dynamic environments.

5. Early model selection: Synthetic data holds promising potential as an early-stage tool for architecture selection, particularly in scenarios where the availability of real data is initially scarce but anticipated to increase over time. By leveraging synthetic data, model performance can be estimated with the usage of larger synthetic datasets. This allows researchers to initiate the model development process even with limited real data, accelerating the overall progress of the project. It also mitigates the risk of making sub-optimal architecture choices based solely on limited real data, thereby potentially avoiding costly redesigns

6. Upsampling through synthetic data: We have shown that diffusion models can be a powerful tool when working with imbalanced datasets. It would be useful to conduct a more in-depth analysis on how synthetic data generation through diffusion models compares to other existing approaches, such as Synthetic Minority Oversampling Technique (SMOTE), and compare their respective impacts on downstream classification tasks.

Bibliography

[1] J. H. Park, S. K. Zhou, C. Simopoulos, J. Otsuki, and D. Comaniciu, "Automatic cardiac view classification of echocardiogram," in *2007 IEEE 11th international conference on computer vision*, IEEE, 2007, pp. 1–8.

[2] J. Zhang, S. Gajjala, P. Agrawal, *et al.*, "A computer vision pipeline for automated determination of cardiac structure and function and detection of disease by two-dimensional echocardiography," *arXiv preprint arXiv:1706.07342*, 2017.

[3] A. Madani, R. Arnaout, M. Mofrad, and R. Arnaout, "Fast and accurate view classification of echocardiograms using deep learning," *NPJ digital medicine*, vol. 1, no. 1, pp. 1–8, 2018.

[4] S. K. Zhou, J. H. Park, B. Georgescu, D. Comaniciu, C. Simopoulos, and J. Otsuki, "Image-based multiclass boosting and echocardiographic view classification," in *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)*, IEEE, vol. 2, 2006, pp. 1559–1565.

[5] N. Azarmehr, X. Ye, J. P. Howard, *et al.*, "Neural architecture search of echocardiography view classifiers," *Journal of Medical Imaging*, vol. 8, no. 3, pp. 034 002–034 002, 2021.

[6] D. Saxena and J. Cao, "Generative adversarial networks (gans) challenges, solutions, and future directions," *ACM Computing Surveys (CSUR)*, vol. 54, no. 3, pp. 1–42, 2021.

[7] L. Wang, W. Chen, W. Yang, F. Bi, and F. R. Yu, "A state-of-the-art review on image synthesis with generative adversarial networks," *IEEE Access*, vol. 8, pp. 63 514–63 537, 2020.

[8] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, "Deep unsupervised learning using nonequilibrium thermodynamics," in *International Conference on Machine Learning*, PMLR, 2015, pp. 2256–2265.

[9] Y. Song and S. Ermon, "Generative modeling by estimating gradients of the data distribution," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[10] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in Neural Information Processing Systems*, vol. 33, pp. 6840–6851, 2020.

[11] A. Q. Nichol and P. Dhariwal, "Improved denoising diffusion probabilistic models," in *International Conference on Machine Learning*, PMLR, 2021, pp. 8162–8171.

[12] P. Dhariwal and A. Nichol, "Diffusion models beat gans on image synthesis," *Advances in Neural Information Processing Systems*, vol. 34, pp. 8780–8794, 2021.

[13] G. Müller-Franzes, J. M. Niehues, F. Khader, *et al.*, "Diffusion probabilistic models beat gans on medical images," *arXiv preprint arXiv:2212.07501*, 2022.

[14] W. H. Pinaya, P.-D. Tudosiu, J. Dafflon, *et al.*, "Brain imaging generation with latent diffusion models," in *MICCAI Workshop on Deep Generative Models*, Springer, 2022, pp. 117–126.

[15] Z. Dorjsembe, S. Odonchimed, and F. Xiao, "Three-dimensional medical image synthesis with denoising diffusion probabilistic models," in *Medical Imaging with Deep Learning*, 2022.

[16] G. Zamzmi, L.-Y. Hsu, W. Li, V. Sachdev, and S. Antani, "Harnessing machine intelligence in automatic echocardiogram analysis: Current status, limitations, and future directions," *IEEE reviews in biomedical engineering*, vol. 14, pp. 181–203, 2020.

[17] G. Zamzmi, S. Rajaraman, L.-Y. Hsu, V. Sachdev, and S. Antani, "Real-time echocardiography image analysis and quantification of cardiac indices," *Medical image analysis*, vol. 80, p. 102 438, 2022.

[18] J. P. Howard, J. Tan, M. J. Shun-Shin, *et al.*, "Improving ultrasound video classification: An evaluation of novel deep learning methods in echocardiography," *Journal of medical artificial intelligence*, vol. 3, 2020.

[19] H. Vaseli, Z. Liao, A. H. Abdi, *et al.*, "Designing lightweight deep learning models for echocardiography view classification," in *Medical Imaging 2019: Image-Guided Procedures, Robotic Interventions, and Modeling*, SPIE, vol. 10951, 2019, pp. 93–99.

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

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

[22] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.

[23] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[24] V. Besnier, H. Jain, A. Bursuc, M. Cord, and P. Pérez, "This dataset does not exist: Training models from generated images," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2020, pp. 1–5.

[25] R. J. Chen, M. Y. Lu, T. Y. Chen, D. F. Williamson, and F. Mahmood, "Synthetic data in machine learning for medicine and healthcare," *Nature Biomedical Engineering*, vol. 5, no. 6, pp. 493–497, 2021.

[26] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," *arXiv preprint arXiv:2010.02502*, 2020.

[27] S. Leclerc, E. Smistad, J. Pedrosa, *et al.*, "Deep learning for segmentation using an open large-scale dataset in 2d echocardiography," *IEEE transactions on medical imaging*, vol. 38, no. 9, pp. 2198–2210, 2019.

[28] J. P. Howard, C. C. Stowell, G. D. Cole, *et al.*, "Automated left ventricular dimension assessment using artificial intelligence developed and validated by a uk-wide collaborative," *Circulation: Cardiovascular Imaging*, vol. 14, no. 5, e011951, 2021.

[29] Z. Huang, G. Long, B. S. Wessler, and M. C. Hughes, "Tmed 2: A dataset for semi-supervised classification of echocardiograms," in *In DataPerf: Benchmarking Data for Data-Centric AI Workshop*, 2022.

[30] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "GANs trained by a two time-scale update rule converge to a local Nash equilibrium," *Advances in neural information processing systems*, vol. 30, 2017.

A

Appendix

A.1 View Classification Results

Feature Extractor	Size	Architecture	AUPRC	F1	Time
False	1k	VGG16	0.6409	0.3894	4m
		DenseNet121	0.7411	0.5478	3m
		ResNet18	0.7784	0.6514	2m
	2k	VGG16	0.8590	0.7802	5m
		DenseNet121	0.8469	0.7517	5m
		ResNet18	0.8807	0.8013	4m
	4k	VGG16	0.9278	0.8717	6m
		DenseNet121	0.9139	0.8522	8m
		ResNet18	0.9298	0.8723	4m
	8k	VGG16	0.9682	0.9302	9m
		DenseNet121	0.9507	0.9094	16m
		ResNet18	0.9584	0.9194	6m
	16k	VGG16	0.9835	0.9449	15m
		DenseNet121	0.9716	0.9367	29m
		ResNet18	0.9742	0.9423	14m
	32k	VGG16	0.9881	0.9596	28m
		DenseNet121	0.9817	0.9576	51m
		ResNet18	0.9829	0.9510	23m
True	1k	VGG16	0.8013	0.7358	2m
		DenseNet121	0.7733	0.7043	2m
		ResNet18	0.7728	0.6747	1m
	2k	VGG16	0.8265	0.7601	3m
		DenseNet121	0.8204	0.7560	2m
		ResNet18	0.8008	0.7246	2m
	4k	VGG16	0.8503	0.7844	4m
		DenseNet121	0.8554	0.7966	3m
		ResNet18	0.8395	0.7638	3m
	8k	VGG16	0.8692	0.8060	6m
		DenseNet121	0.8742	0.8194	6m
		ResNet18	0.8620	0.7860	4m
	16k	VGG16	0.8845	0.8222	10m
		DenseNet121	0.8924	0.8343	11m
		ResNet18	0.8797	0.8157	6m
	32k	VGG16	0.8918	0.8288	18m
		DenseNet121	0.8949	0.8404	20m
		ResNet18	0.8898	0.8182	14m

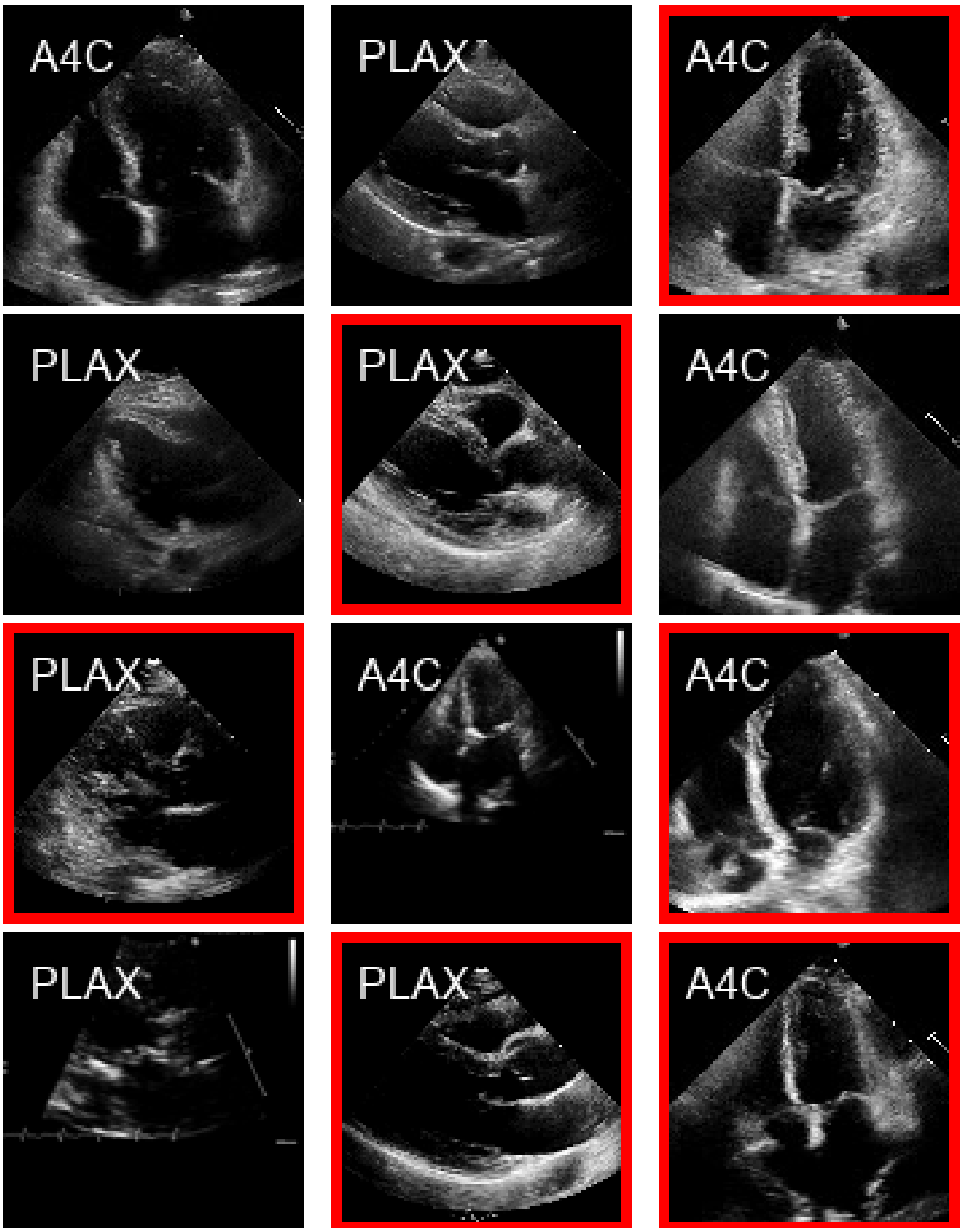
Table A.1: Results of classification training exploration on real data.

Feature Extractor	Size	Architecture	AUPRC	F1	Time
False	1k	VGG16	0.9307	0.8731	29m
		DenseNet121	0.9120	0.8666	51m
		ResNet18	0.9046	0.8399	23m
	2k	VGG16	0.9528	0.8987	29m
		DenseNet121	0.9525	0.9005	54m
		ResNet18	0.9333	0.8712	26m
	4k	VGG16	0.9600	0.9080	29m
		DenseNet121	0.9553	0.9127	52m
		ResNet18	0.9549	0.8986	23m
	8k	VGG16	0.9746	0.9231	29m
		DenseNet121	0.9574	0.9089	53m
		ResNet18	0.9671	0.9190	21m
	16k	VGG16	0.9840	0.9484	29m
		DenseNet121	0.9703	0.9314	54m
		ResNet18	0.9791	0.9534	20m
	32k	VGG16	0.9860	0.9532	29m
		DenseNet121	0.9784	0.9531	51m
		ResNet18	0.9805	0.9520	23m
True	1k	VGG16	0.8296	0.7785	17m
		DenseNet121	0.8118	0.7666	16m
		ResNet18	0.8074	0.7545	13m
	2k	VGG16	0.8442	0.7802	18m
		DenseNet121	0.8317	0.7762	18m
		ResNet18	0.8035	0.7440	15m
	4k	VGG16	0.8414	0.7559	19m
		DenseNet121	0.8177	0.7279	18m
		ResNet18	0.8001	0.7359	16m
	8k	VGG16	0.8542	0.7853	19m
		DenseNet121	0.8325	0.7582	19m
		ResNet18	0.8186	0.7530	13m
	16k	VGG16	0.8736	0.8005	18m
		DenseNet121	0.8591	0.7924	18m
		ResNet18	0.8428	0.7869	13m
	32k	VGG16	0.8893	0.8095	19m
		DenseNet121	0.8829	0.8123	17m
		ResNet18	0.8667	0.7972	15m

Table A.2: Results of classification training exploration on synthetic data.

A.2 Assessment Task

Figure A.1: Assessment task solution: synthetic echocardiograms enclosed in red square boxes (view labels included).



A.3 Survey Questions

Outlined here are the questions included in the survey, in the same order as they were asked to participants:

- Current Occupation
- Medical Specialty
- How many years of experience do you have from working in the medical field post medical school?
- What is your level of experience with ultrasound images in general?
- What is your level of experience with echocardiograms specifically?
- Generally, how challenging do you think it will be to spot the real echocardiograms
- 50 Questions with image pairs asking "Which image is real?"
- Generally, how confident were you with you answers about which images were the real ones?
- Generally, how did you distinguish real echocardiograms from fake ones?

A.4 Clustering plots

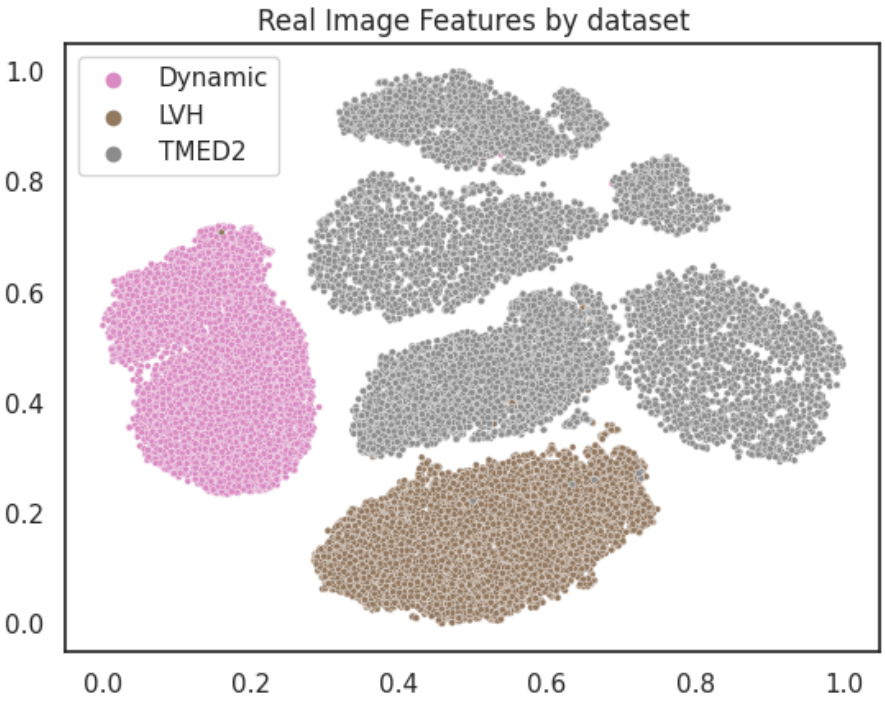


Figure A.2: Real image features by dataset. Real images are passed through our VGG16 model trained on the full training set of real images. The activations of the second to last layer are extracted and reduced to 2 dimensions by running t-SNE.

Figure A.3: Real image features by view. Real images are passed through our VGG16 model trained on the full training set of real images. The activations of the second to last layer are extracted and reduced to 2 dimensions by running t-SNE.

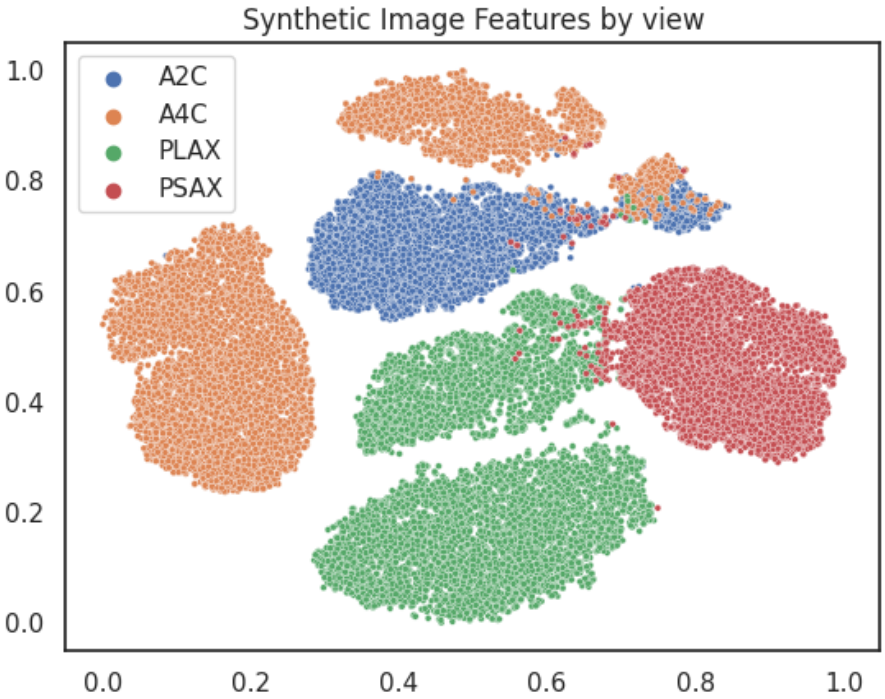


Figure A.4: Synthetic image features by view. Synthetic images are passed through our VGG16 model trained on the full training set of real images. The activations of the second to last layer are extracted and reduced to 2 dimensions by running t-SNE.

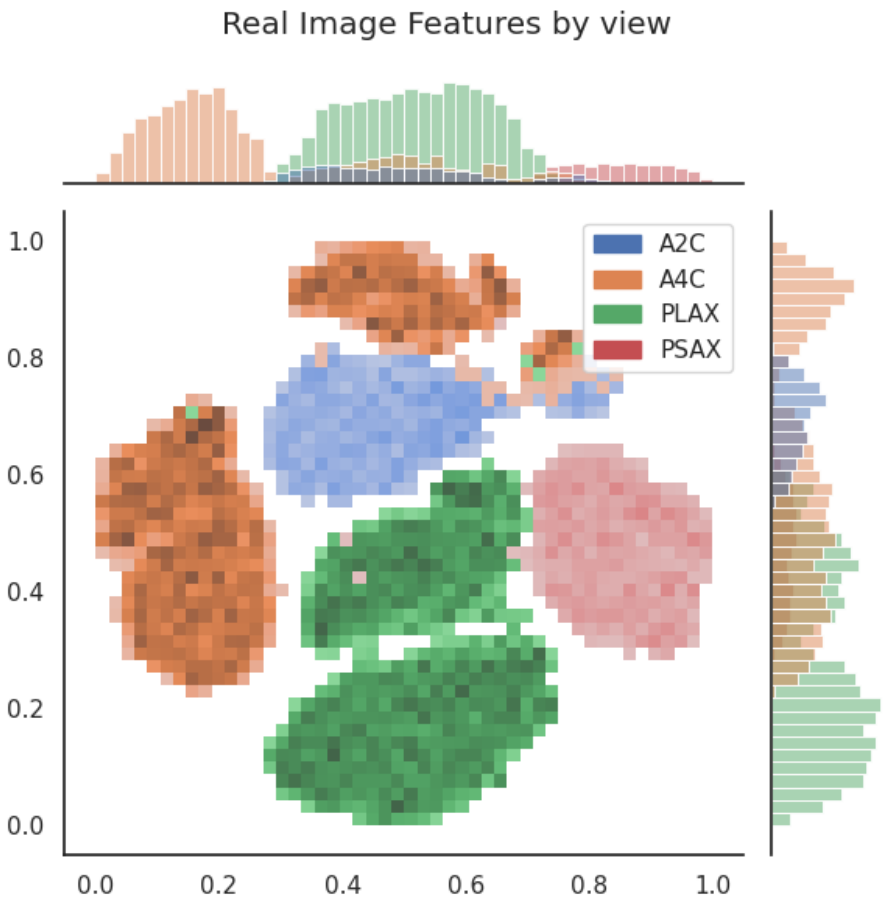


Figure A.5: 2D-Histogram showing the distribution of real images per view. Real images are passed through our VGG16 model trained on the full training set of real images. The activations of the second to last layer are extracted and reduced to 2 dimensions by running t-SNE.

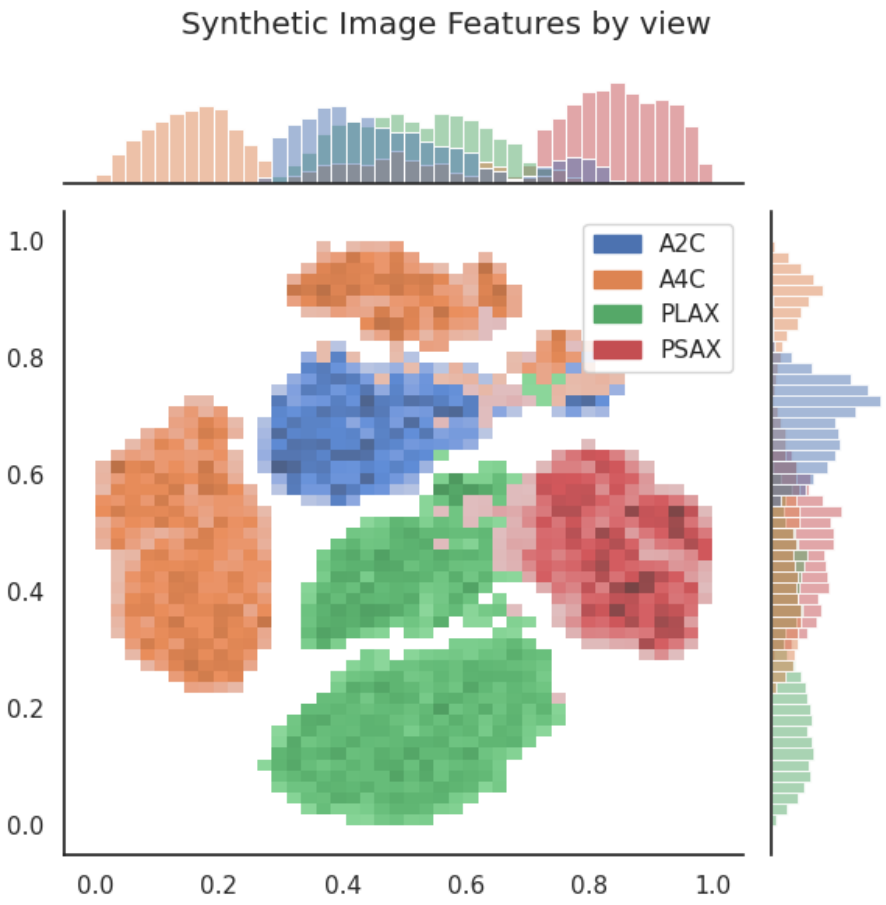


Figure A.6: 2D-Histogram showing the distribution of synthetic images per view. Synthetic images are passed through our VGG16 model trained on the full training set of real images. The activations of the second to last layer are extracted and reduced to 2 dimensions by running t-SNE.