Augmenting Lung Ultrasound with Synthetic Data: A Novel Application of Textual Inversion and Denoising Diffusion Probabilistic Models

Amitay Lev[1]

[1]Faculty of Biomedical Engineering, Technion-IIT, Haifa, Israel

Abstract

In the rapidly evolving field of machine learning, ensuring model reliability, particularly in predicting performance on unseen distributions, poses a significant challenge. This issue is further exacerbated in the absence of labeled data, where minor discrepancies between training and testing distributions can substantially compromise model accuracy. This study delves into the exploration of methods for predicting the accuracy gap that arises when a model, trained on a base distribution, is applied to a different target distribution. We employed various models, including convolutional neural networks for image data and a random forest model for adult income data in the tabular domain, to assess this accuracy gap. Our work offers a comparative analysis of two existing methods - difference of confidence and difference of entropies - and introduces novel methods based on the computation of statistics using the output probabilities of trained classification models on both in-distribution and out-of-distribution test datasets. Our findings suggest that higher statistical moments, such as variance, skewness, and kurtosis, can serve as accurate predictors of accuracy gaps induced by distribution shifts and exhibit less sensitivity to the training split. Additionally, we highlight potential shortcomings of previously proposed methods and propose an extension that leverages multiple statistics to construct a linear regressor for estimating the accuracy gap, demonstrating its advantages.

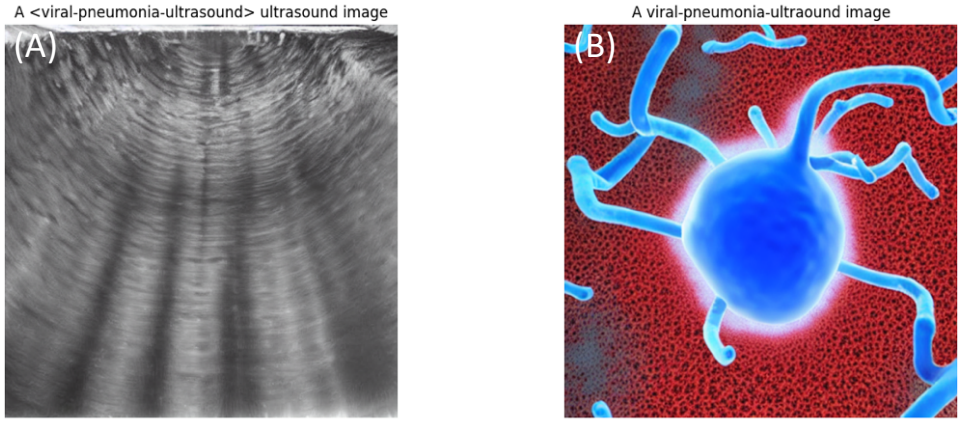

Figure 1: Samples generated using the same prompt from a textual inversion 'object' model trained on all training set images. (A) With the learned token <viral-pneumonia-ultrasound>. (B) Without the learned token.

1. Problem and scientific background

The field of machine learning is known for its rapid advancements and the continuous development of new models and techniques which have revolutionized multiple domains, yet its integration into healthcare has been notably slower. This lag is largely due to the laborious and resource-intensive nature of medical data annotation. The issue becomes even more acute when dealing with partially labeled data creating imbalanced datasets, as models trained under such conditions often demonstrate skewed predictive performance. This is of particular concern in the domain of medical imaging, where diagnostic accuracy is critical.

In this work, I have decided to tackle the specialized domain of lung ultrasound imaging, where the challenges are further compounded. Ultrasound technology offers the advantage of real-time, non-invasive imaging, but it also suffers from a significant lack of publicly available data of normal and abnormal conditions, let alone their labels. Additionally, ultrasound data often contains noisy signals, which can introduce artifacts and further complicate the annotation process. This scarcity and noise are exacerbated by the quality of ultrasound data being highly operator-dependent, adding another layer of complexity to the already challenging task of data annotation.

In recent academic contributions, diffusion models and vision-language models have been explored for their applicability in generating medical imaging data. Chambon et al. [1] fine-tuned the Stable Diffusion model and experimented with textual inversion [2] techniques to generate synthetic medical images, such as chest X-rays. Adhikari et al. [3] demonstrated the utility of Vision-Language Segmentation Models (VLSMs) in echocardiography, showing that pretraining on synthetic datasets led to improved performance metrics. Stojanovski et al. [4] extended the application of diffusion models to ultrasound imaging, specifically generating synthetic ultrasound images that assist in real image segmentation. Gal et al. [2] further explored textual inversion in the context of text-to-image generation, although their focus was not specifically on medical imaging. Collectively, these studies underscore the burgeoning potential of diffusion and vision-language models, along with techniques like textual inversion, in addressing domain-specific challenges such as data scarcity and quality in medical imaging, including ultrasound.

In this research, I propose a novel methodology for data augmentation that specifically addresses these domain-related challenges. Utilizing diffusion models and textual inversion [2], the aim is to augment underrepresented classes within the lung ultrasound dataset. Through this individual research project, the aspiration is to contribute a significant, domain-specific methodology for lung ultrasound data augmentation, thereby enhancing the predictive accuracy of machine learning models in the field of medical imaging.

2. Data description

In this section I will discuss about ultrasound data followed by an elaboration on how I created the dataset distribution for this work.

2.1. Ultrasound data

Ultrasound is a non-invasive, cheap, portable (bedside execution), repeatable and available in almost all medical facilities. All these make ultrasound almost a default choice for emergency situations and rapid response situations. On the other hand, due to the noisy nature of ultrasound images, it might be challenging even for trained professionals to detect patterns of pathologies and abnormalities thus raising the need for automated detection algorithms.

High quality annotated open-source medical imaging datasets are challenging to find, though there are quite a few online challenges and data for modalities such as CT, X-ray, MRI etc. However, when dealing with ultrasound the picture is quite different - there are very few available datasets, of which only a few are accompanied with quality annotations.

At first, I thought it would be best for my purpose of this project to work with abdomen ultrasound data, due to the relatively easy to identify images produced by the ultrasonic waves. In the literature survey I conducted, there were only a few candidate datasets. In [5] there is a GitHub repository holding train and test data of a few hundred abdomen classification images, however they have very low resolution, making it less convenient to handle in terms of generating quality synthetic images.

A promising option was [6], which is an abdomen ultrasound dataset with real ultrasound scans and synthetic images generated with a ray-casting based simulator hosted on Kaggle. It's provided with annotations of several abdominal organs and some of it was created using cycle GANs. This dataset had two main issues, the first was that most of the data was generated by a simulator thus making it less applicable to my task, and secondly, there were very few annotations, making it hard to create a classification dataset.

Another dataset candidate was from the Ultrasound Brachial plexus (BP) nerve segmentation challenge in Kaggle [7], provided with a large training set of images where the nerve has been manually annotated by humans. The annotators were trained by experts and instructed to annotate images where they felt confident about the existence of the BP landmark. This dataset is from over 7 years ago and though it might serve as a relatively fertile ground for segmentation tasks, it fails to be useable for my classification task.

Eventually I used a dataset called the COVID-19 lung ultrasound dataset, as published in [8], on which I will elaborate in the following section.

2.2. COVID-19 dataset

The COVID-19 lung ultrasound dataset is a dataset comprised of few thousand images and videos of lung ultrasound scans. The dataset is the largest public source lung ultrasound dataset, it combines data from collaborating hospitals as well as publicly available resources from the web (e.g. publications and educational websites). Alongside the data are scripts and methodologies for dataset curation (i.e., scraping from online resources, etc.) and creating meaningful data splits in terms of patients and distributions. Eventually I decided to use the data as frames, rather than videos, to simplify the synthetic data creation process. A summary of the final dataset class distribution can be seen in table 1.

Class	Regular	Covid	Pneumonia	Viral Pneumonia
# Frames	1234	826	707	52

Table 1: COVID-19 dataset diistribution

In lung ultrasound imaging, certain key features are commonly observed and analyzed. "A-lines" are horizontal lines that indicate normal air movement in the lungs, while "B-lines" are vertical lines that suggest the presence of fluid, often indicative of inflammation or infection. The "pleural line" is another important feature; it's the bright, white line that represents the interface between

the lung and the chest wall. Irregularities in the pleural line can be indicative of various lung conditions. Visual explanation can be seen in figure 2.

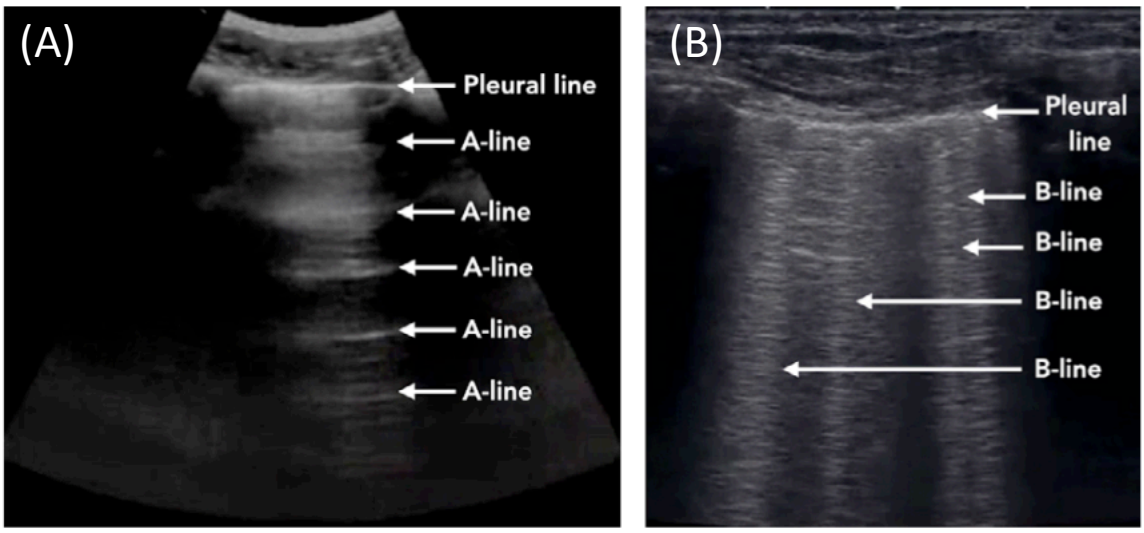

Figure 2: (A) Lung ultrasound showing the pleural line and artifact A-line and (B) Lung ultrasound displaying the artifact B-line [9].

In the "Regular" class of lung ultrasound images, one typically observes a smooth pleural line accompanied by horizontal A-lines, signaling healthy lung tissue. For the "Covid" class, the ultrasound often reveals irregularities in the pleural line and the presence of vertical B-lines, indicative of viral-induced inflammation and infection. In the "Pneumonia" class, localized B-lines and subpleural consolidations are commonly seen, suggesting bacterial infection. Lastly, the "Viral Pneumonia" class is characterized by a diffuse distribution of B-lines and potential irregularities in the pleural line, a pattern that is similar to, yet distinct from, what is commonly observed in Covid-affected lungs.

In this work, I aspired to create a highly unbalanced dataset, such that it will be a good setting for generating synthetic data using methods that require only a few sample images such as textual inversion [2]. To achieve this, I created a meaningful train/test data split by ensuring that the train/validation/test split is not done on a frame-level, but on a video/patient-level. This is important because consecutive lung ultrasound frames are extremely correlated, which could potentially create an obvious patient data leakage. This became even more challenging as the viral pneumonia class is comprised of only three videos, each with very different characteristics as can be seen in figure 3. The challenge is that to avoid data leakage, a single type of examples is available in each of the data splits, making it extra hard for the network to learn and generalize this specific already underrepresented class. A summary of the final data split is described in table 2.

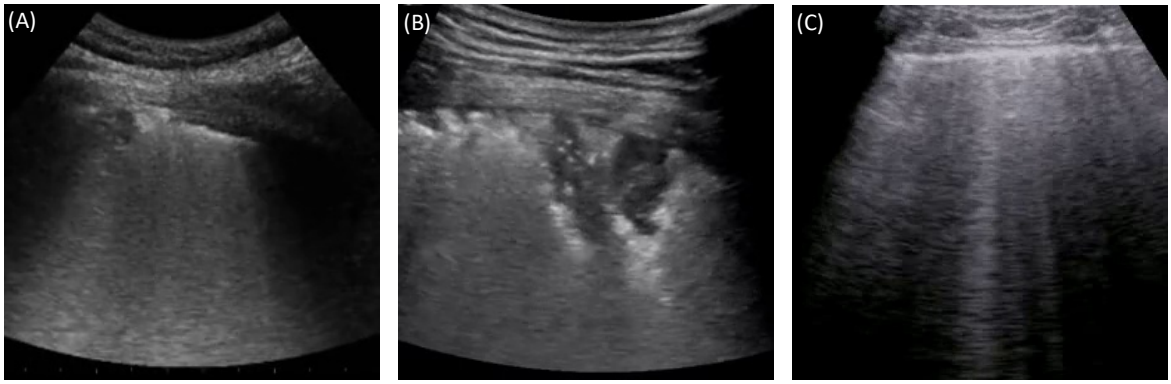

Figure 3: Representations of the three different videos in the "Viral Pneuomonia" class.

	Train	Validation	Test	Total
Regular	814	225	195	**1234**
Covid	571	143	112	**826**
Pneumonia	468	118	121	**707**
Viral Pneumonia	27	9	16	**52**
Total	**1880**	**495**	**444**	**2819**

Table 2: COVID-19 data split

3. Methods

In this project, I set out to generate synthetic images for the 'viral pneumonia' class, which is extremely small relative to the other classes. For this I first set out to achieve a baseline classifier with the current dataset, then try common methods for overcoming the imbalance, followed by generating synthetic 'viral pneumonia' data by training diffusion models in two different ways.

3.1. Baseline classifier

First, I set out to achieve a baseline multi class classifier for the dataset in hand. I employed two standard classifiers from known families of classifier architectures, ResNet50 [10] and a Vision transformer (ViT) [11]. Due to the relatively small amounts of data in hand, I initialized both model weights with ImageNet [12] pretrained weights from Pytorch and tried various fine-tuning settings. However, these led to worse performance compared to fully training the networks, thus in the following research, I will only present results of fully trained models. The classifiers are all trained using a softmax cross-entropy loss function, mathematically represented as:

(eq. 1). $$L(y, \hat{y}) = -\sum_{i=1}^{C} y_i \log(\hat{y}_i)$$

Where y is the true label and $\hat{y}$ is the predicted label for C classes.

3.2. Common class imbalance solutions

In scenarios of extreme class imbalance such as mine, traditional oversampling and undersampling techniques may not be sufficient. Oversampling by duplicating instances from the minority class can exacerbate the risk of overfitting, as the model may memorize these instances

rather than generalize from them. On the other hand, random undersampling of the majority classes can lead to loss of potentially valuable data, further skewing the model's ability to generalize across classes.

Therefore, more sophisticated cost-sensitive learning methods are what I decided to try out. I set out to try the simple method of adding class weights to adjust the loss function to penalize misclassification of the minority class more heavily, thereby directing the model to pay more attention to underrepresented classes. The mathematical equation of the new loss function is presented in equation 2.

(eq. 2). $L_w(y, \hat{y}) = -\sum_{i=1}^{C} w_i y_i \log(\hat{y}_i)$

Where w_i is the weight for class i.

Methods of this sort should be applied cautiously, as they can introduce their own set of challenges, such as making the model too sensitive to noise in the minority class, which in this case is smaller by orders of magnitude.

3.3. Textual Inversion

Textual inversion [2] is an innovative technique that bridges the gap between textual descriptions and image generation, offering a nuanced approach to synthesizing images based on text prompts. Originating from the field of vision-language models, textual inversion employs a dual encoder-decoder architecture to map textual descriptions into a latent space, which is then used to generate corresponding images. Unlike traditional text-to-image synthesis methods, textual inversion allows for the creation of "pseudo-words" or embeddings in the latent space that can guide the image generation process. This enables the model to generate highly specific and context-sensitive images while using very little image samples and compute, making it a potentially powerful tool for data augmentation. The technique has been successfully applied in various applications, demonstrating its versatility and effectiveness in generating high-quality synthetic images. An overview of textual inversion in the context of this work can be seen in figure 4.

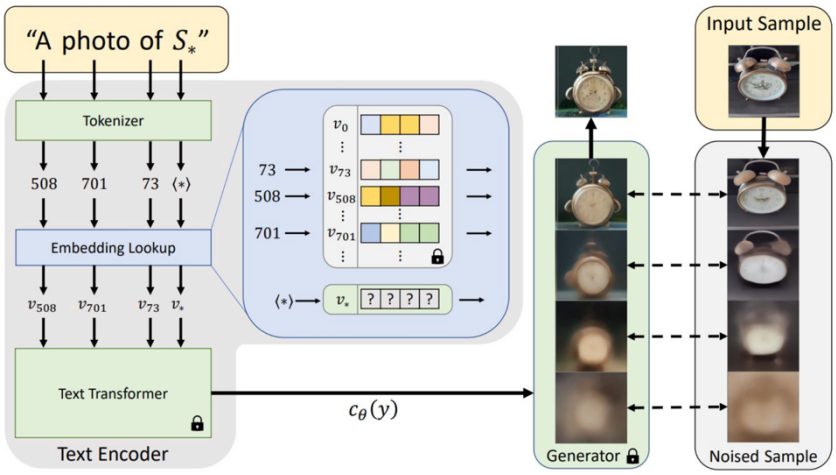

Figure 4: Textual inversion mechanism overview

As explained in [2], the textual inversion mechanism works as following: *"In the text-encoding stage of most text-to-image models, the first stage involves converting the prompt into a numerical representation. This is typically done by converting the words into tokens, each equivalent to an entry in the model's dictionary. These entries are then converted into an "embedding" - a continuous vector representation for the specific token. These embeddings are usually learned as part of the training process. In textual inversion, we find new embeddings that represent specific, user-provided visual concepts. These embeddings are then linked to new pseudo-words, which can be incorporated into new sentences like any other word. In a sense, we are performing inversion into the text-embedding space of the frozen model. This method can be used to represent a wide array of concepts - including visual styles."*

I've seen very few attempts of harnessing textual inversion for medical imaging generation, and after presenting the work done in [1] as part of this course tasks, I've decided to try this out on my own domain of interest, ultrasound. If this works, this could be groundbreaking for specialized domains like medical imaging where the need for high-fidelity, condition-specific images is paramount.

3.4. Denoising Diffusion Probabilistic Model (DDPM)

Denoising Diffusion Probabilistic Models (DDPMs), introduced in [13] are a class of generative models that have gained attention for their ability to produce high-quality synthetic images. Inspired by the principles of nonequilibrium thermodynamics, DDPMs operate by evolving an initial noise distribution through a series of diffusion steps to generate a sample that closely resembles the target data distribution. Unlike traditional generative models like Generative Adversarial Networks (GANs) or Variational Auto Encoders (VAEs), DDPMs do not require an explicit likelihood model but instead rely on a noise schedule and a denoising function to guide the diffusion process. This makes them particularly well-suited for tasks where the data distribution is complex and high-dimensional, such as medical imaging and ultrasound in particular.

DDPMs ability to generate realistic and diverse samples has made it a promising tool for data augmentation, especially in domains where data is scarce or imbalanced. The denoising mechanism of the models, and the noisy nature of ultrasound sparked the idea for this project, and I was excited to try these out. In this work, I used Hugging Face's Diffusers library [14] and pipelines, which is a convenient platform for training and using pretrained DDPMs and other types of diffusion-based models.

4. Results

Experiment	Model	Accuracy	F1 Score	Precision
Baseline	ViT	71.84	70.83	71.02
	ResNet50	76.57	74.59	75.23
Class weights	ViT	68.91	67.45	66.36
	ResNet50	75.90	74.15	74.27
DDPM 3 mixed	ViT	70.72	71.07	74.14
	ResNet50	**78.82**	**76.80**	**76.66**
DDPM all viral	ViT	66.89	65.38	68.19
	ResNet50	70.72	69.06	67.80

Table 3: Experiment results.

4.1. Baseline classifier

In the baseline experiment, the ResNet50 model outperformed the ViT model across all performance metrics as shown in table 3. Specifically, ResNet50 achieved an accuracy of 76.57%, an F1 Score of 74.59%, and a Precision of 75.23%. In comparison, the ViT model showed an accuracy of 71.84%, an F1 Score of 70.83%, and a Precision of 71.02%. This disparity in performance could be attributed to the inherent architectural advantages of ResNet50 for this specific task, suggesting that it serves as a stronger baseline model.

In a more general view, these results are reasonable for this classification task, validating the current research setting.

4.2. Common class imbalance solutions

In addressing the challenge of class imbalance within the dataset, I employed a weighting strategy for the loss function during the model's training phase. Specifically, the weights were computed as the inverse of each class's frequency in the training set, formulated in equation 3. This approach effectively amplifies the contribution of the minority classes to the loss function, thereby mitigating the model's bias towards the majority classes. By doing so, I aimed to enhance the model's sensitivity to all classes, ensuring a more balanced performance. This weighting mechanism is a widely-accepted method for counteracting the effects of class imbalance and was crucial for achieving a more equitable model performance across all classes.

(eq. 3). $$w_i = \frac{Total\ count}{Count\ of\ class\ i}$$

Where "total count" is the sum of the counts of all classes, and "Count of class i" is the number of samples in class i. In this experiment setting, the total count is $Total\ count = 814 + 571 + 468 + 27 = 1880$

$$w_{regular} = \frac{1880}{814} = 2.31\ ;\ w_{covid} = \frac{1880}{571} = 3.29\ ;$$

$$w_{pneumonia} = \frac{1880}{468} = 4.02 \; ; \; w_{viral} = \frac{1880}{27} = 69.63$$

When these class weights were incorporated into the models, a decline in performance was observed compared to the baseline experiment as shown in table 3. ResNet50 yielded an accuracy of 75.90%, an F1 Score of 74.15%, and a Precision of 74.27%. ViT had an accuracy of 68.91%, an F1 Score of 67.45%, and a Precision of 66.36%. The decrement in performance suggests that the models' architectures are not effectively utilizing the weight adjustments. Alternatively, a very reasonable explanation might be that the current weights are not optimal and should be chosen in a more sophisticated manner.

4.3. Textual Inversion

In the scope of this research project, I investigated the capabilities of textual inversion for generating synthetic ultrasound images, specifically targeting the smallest class of 'viral pneumonia'. Textual inversion can be used to learn either an ‘object’ or ‘style’ concept from even a few images and text prompts. The dataset for the experiments was curated from the three distinct videos in the COVID-19 dataset, each capturing unique visual attributes of 'viral pneumonia'. To optimize the training process, each video was designated to a different data split. For this experiment, two data configurations were employed: the first comprised a set of 6 images, with 2 images selected from each video, and the second included all 27 images from the single training set video.

4.3.1. ‘Object’ inversion

My initial experiment was targeted towards learning the 'object' concept, with the first data configuration of 6 mixed images. Utilizing the textual token `<viral-pneumonia-ultrasound>` along with the text prompt "A <viral-pneumonia-ultrasound> ultrasound image", I fine-tuned a stable diffusion model over a span of 3000 epochs. The generated images did exhibit some features that could be associated with viral pneumonia ultrasounds; however, they were not clinically useful. Some of the images were even unsettling, as depicted in figure 5. In figure 6 we can also see a comparative analysis with images generated using a similar but non-explicit text prompt revealed that the model had indeed learned some aspects of 'viral pneumonia', albeit inconsistently.

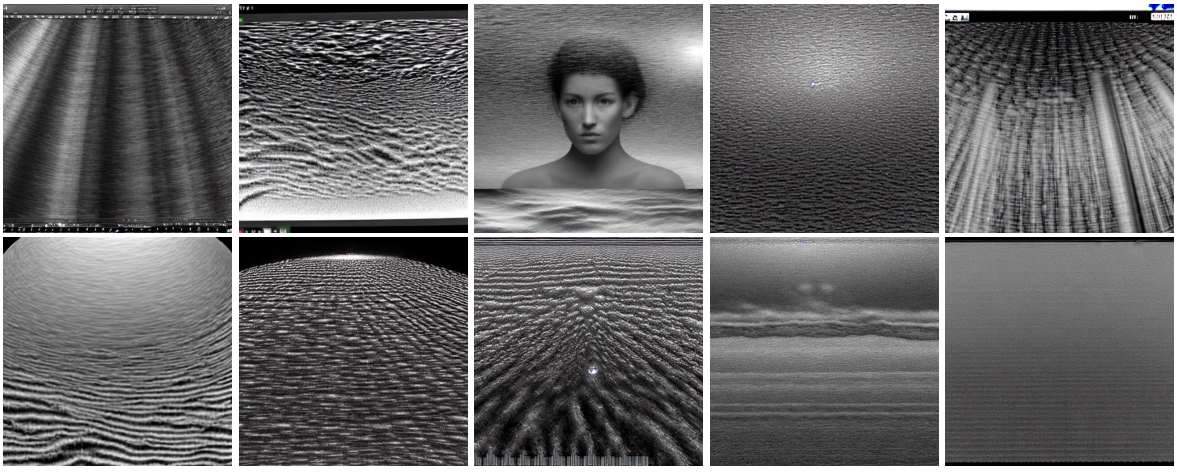

Figure 5: Samples generated from a textual inversion ‘object’ model trained on 6 mixed images with the prompt "A <viral-pneumonia-ultrasound> ultrasound image".

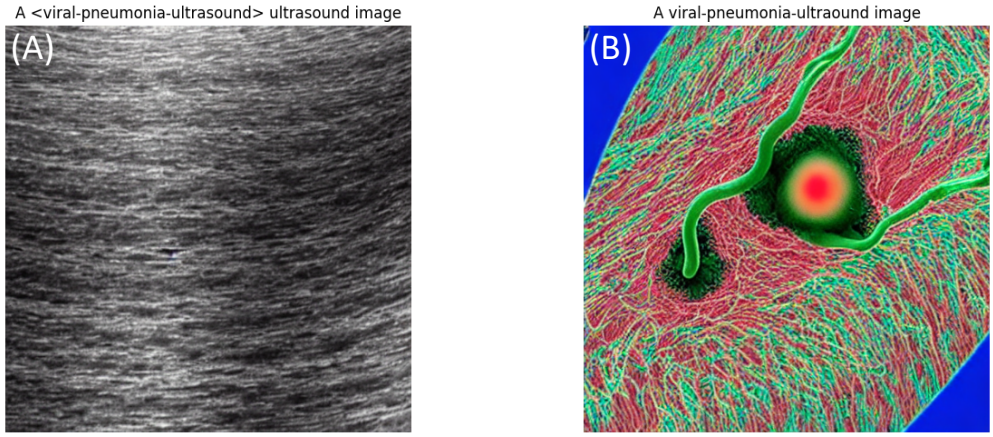

Figure 6: Samples generated using the same prompt from a textual inversion 'object' model trained with 6 mixed images. (A) With the learned token <viral-pneumonia-ultrasound>. (B) Without the learned token.

These results were unsatisfying, and a possible reason might be the lack of focus due to the small and diverse data. To try and target specific characteristics, I expanded the training set to include all 27 images from a single video which predominantly featured vertical B-lines. As seen in figure 7, despite capturing these visual elements, the generated images did not resemble authentic ultrasound images. This limitation can be attributed to the limited variance in this single video. Furthermore, we can see that the results were inconsistent and, in some cases, even unsettling, emphasizing the challenges and unpredictability associated with using stable diffusion models, especially when textual inversion is involved. Figure 1 depicts the same comparative analysis with images generated using a similar but non-explicit text prompt, revealing the same unsatisfying results, but somewhat more ultrasound like.

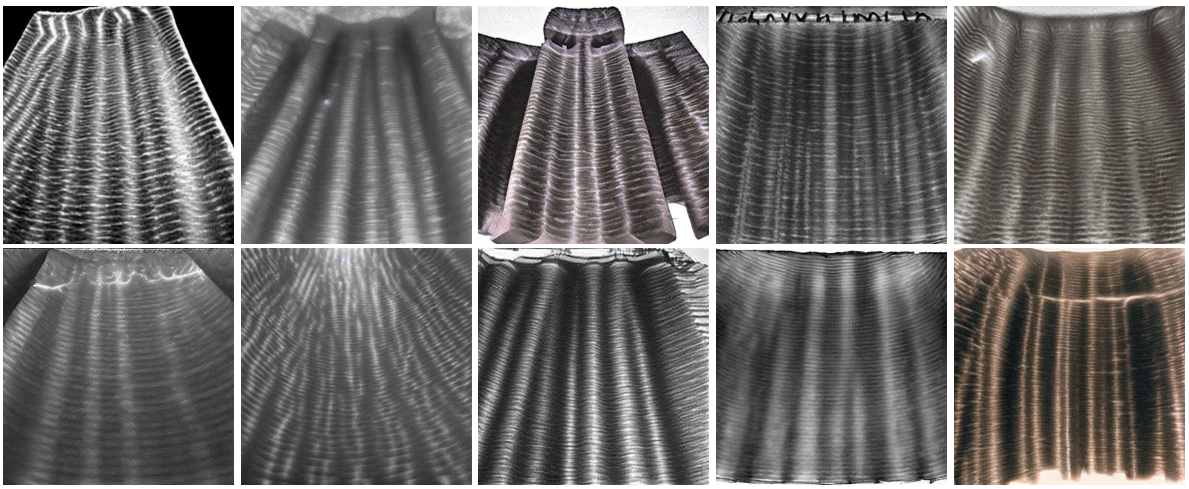

Figure 7: Samples generated from a textual 'object' inversion model trained on all training set images with the prompt "A <viral-pneumonia-ultrasound> ultrasound image".

4.3.2.‘Style’ Inversion

Feeling somewhat disappointed but still optimistic, I shifted my focus to learning the 'style' of the ultrasound images. Using the same 6-images set, the text prompt was modified to "A <viral-pneumonia-ultrasound> style ultrasound image". Initially, the generated images appeared to be more promising; however, a closer examination revealed their lack of clinical utility, as shown in figure 8. The learned token comparative analysis, shown in figure 9, consistently shows a learning of the concept, though it is probably the worst result in terms of details and ultrasound-like characteristics.

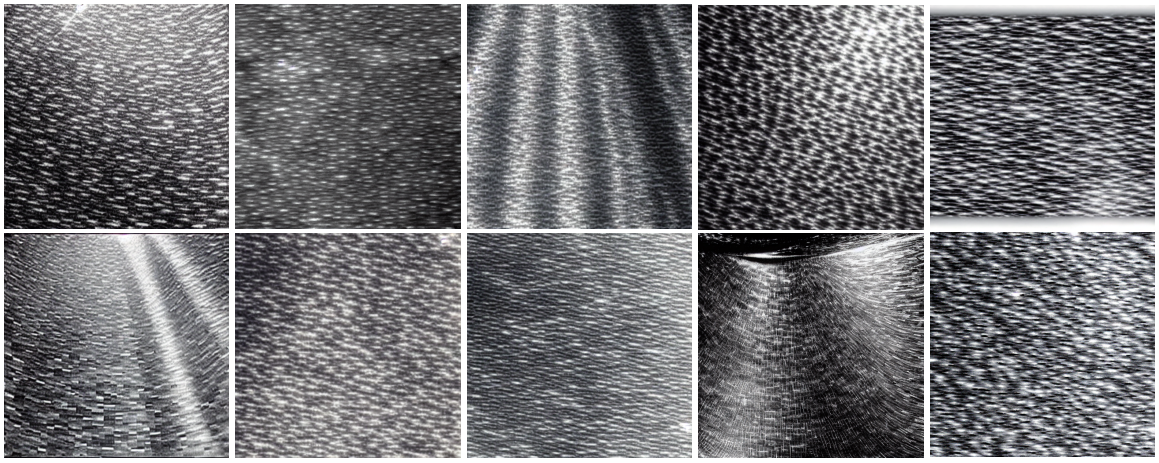

Figure 8: Samples generated from a textual inversion ‘style’ model trained on 6 mixed images with the prompt "A <viral-pneumonia-ultrasound> ultrasound image".

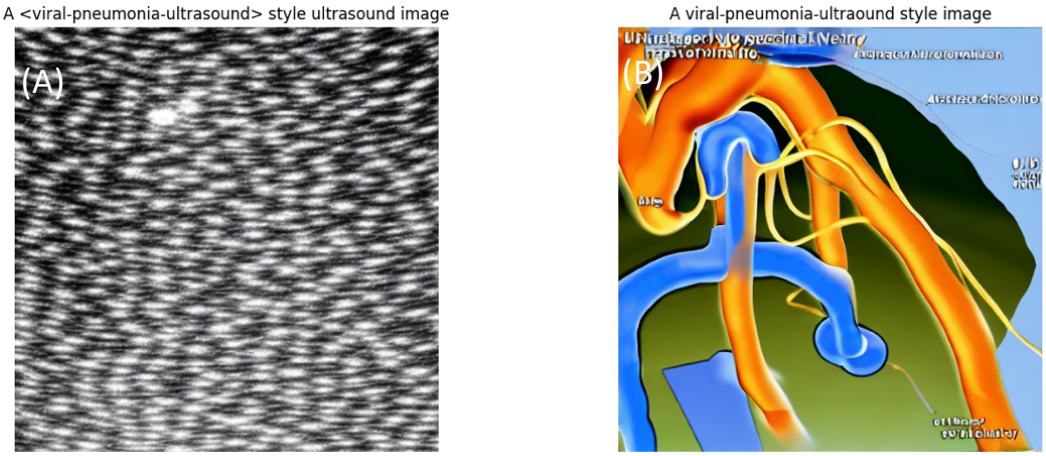

Figure 9: Samples generated using the same prompt from a textual inversion ‘style’ model trained on all training set images. (A) With the learned token <viral-pneumonia-ultrasound>. (B) Without the learned token.

In summary, while the textual inversion technique demonstrated some ability to learn specific tokens and concepts, it fell short in generating clinically meaningful or reliable ultrasound images. This highlights the complexity and challenges of employing this method for synthetic data generation in the medical imaging domain, indicating that further research is required to stabilize and refine these models.

4.4. DDPM

The original purpose for this project was to generate synthetic viral pneumonia ultrasound images using textual inversion. Following the inconsistency and unsatisfying results shown in the previous section, I decided to give diffusion models another chance. The following experiments were targeted at training DDPMs using two data configurations: the first comprised a set of as little as three images, with one image selected from each video (three mixed images), and the second included all 52 images from viral pneumonia (all viral images). To train the DDPM models I used HuggingFace's diffusers library [14] and trained an unconditional image generation diffusion model pipeline. These models are comprised of a pipeline with a noise scheduler and a denoising Unet2D model. These models are fully trained on the dataset; thus, they presented superior performance in terms of ultrasound images generation.

However, being unconditional image generators, the generation process is random and produces a variance of samples, some completely noised and visually unsatisfying. To control the generated samples quality, I investigated the images statistics and empirically defined a threshold for each sampling attempt. This analysis is done on the three images dataset for better generalization. The main attributes I decided upon were the image pixel values standard deviation (equation 4) and entropy (equation 5).

(eq. 4). $$STD(I) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(I_i - \mu)^2}$$

Where I is an image with N pixels, and μ is the mean intensity of the image.

(eq. 5). $$Entropy(I) = -\sum_{i=0}^{L-1} p(i)\log_2 p(i)$$

Where L is the number of gray levels in the image, and p(i) is the probability of gray level I, which is calculated by $p(i) = \frac{n_i}{N}$. A visualization of the empirical thresholding is shown in figure 10, it is noticeable that samples with meaningful attributes have entropy larger than 4.7 and standard deviation of over 30. Therefor these are the thresholds chosen for the generation process, separating noise from high-quality samples.

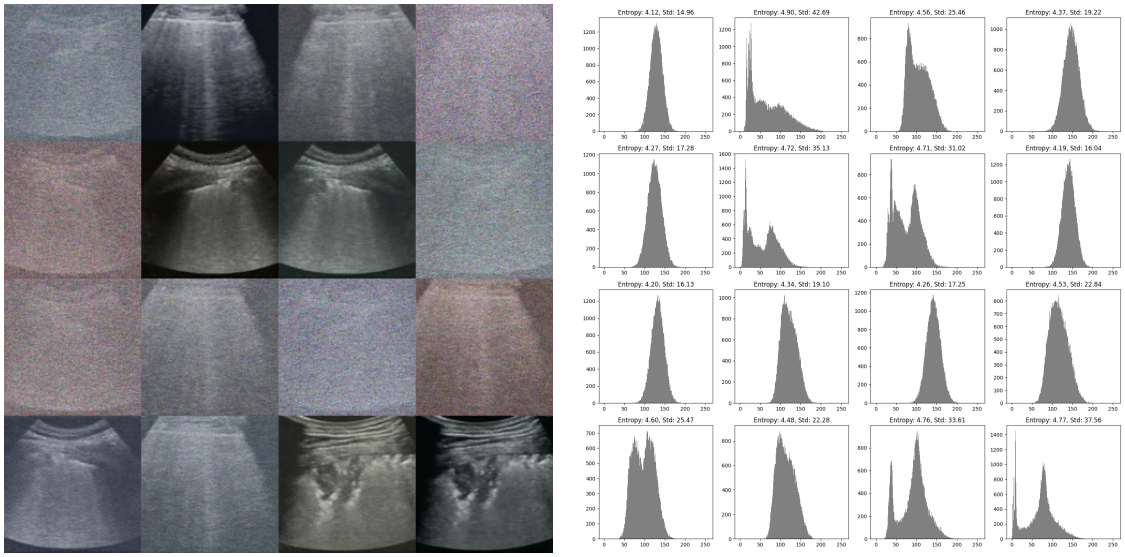

Figure 10: Visualization of the DDPM generation empirical thresholding.

4.4.1. Three mixed images:

In this setting, I have trained a DDPM on 3 mixed images, each from a unique video, for only 500 epochs. After thresholding the generated samples, I have curated a 335 synthetic images dataset, of which a dozen samples is shown in figure 11. Visual analysis of the generated samples shows that that the images are very similar to the original images, however some samples are horizontally flipped, small difference in image colors, blurs, and hopefully new combinations.

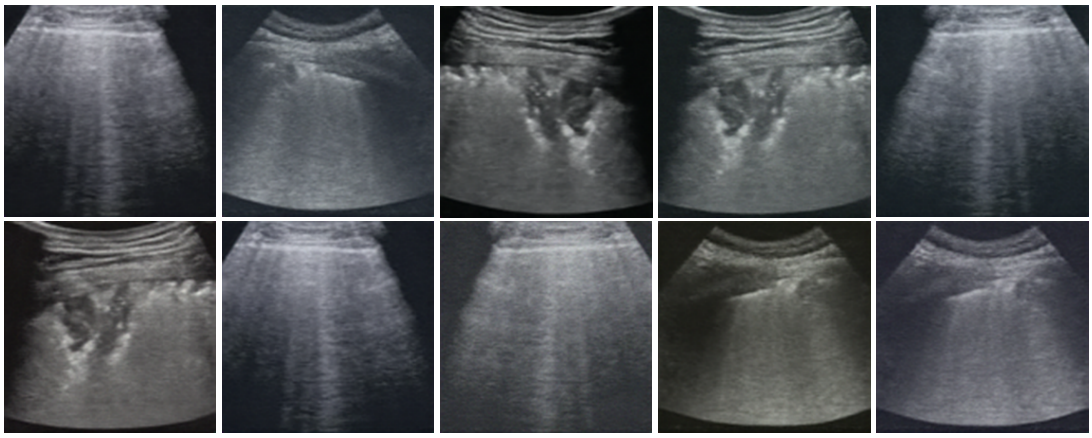

Figure 11: Samples generated using a DDPM model trained on 3 mixed images.

As shown in table 3, the 'DDPM 3 mixed' experiment resulted in a significant performance boost for both models. ResNet50 reached its peak with an accuracy of 78.82%, an F1 Score of 76.80%, and a Precision of 76.66%. ViT also improved, particularly in Precision, which rose to 74.14%, while its accuracy and F1 Score were 70.72% and 71.07%, respectively. This indicates that the 'DDPM 3 mixed' settings might be capturing some underlying features or patterns in the data that are particularly advantageous for the models, especially for ResNet50. These outstanding results are very surprising, given the small amount of training images and epochs usen for training this model. A very interesting result worth noticing is the improved correct detection of the viral pneumonia test set, visible in figure 12, holding all confusion matrices for ViT model experiments.

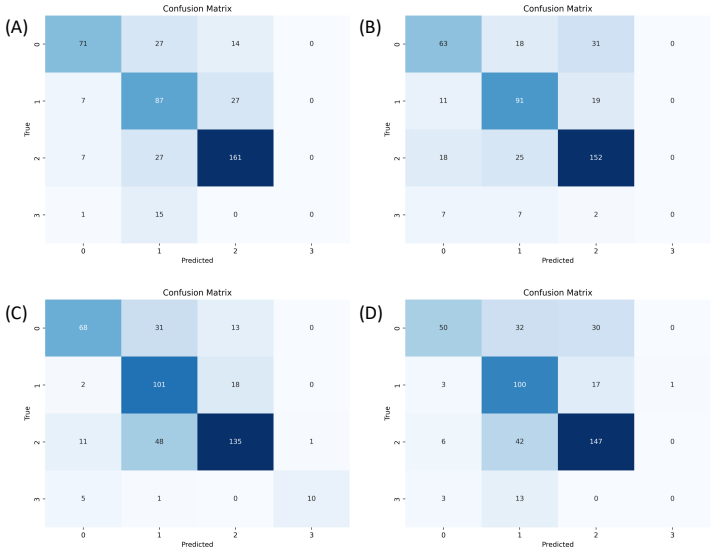

Figure 12: Confusion matrices for the ViT model. (A) Baseline. (B) Class weights. (C) 3 mixed images. (D) All training images.

4.4.2. All viral images:

In the second DDPM experiment, I trained a model on all 52 viral images with expectation that albeit it might not be fully correct in terms of data leakage, but this was for exploring the abilities of ultrasound data generation. The model was trained for 3000 epochs and after quality thresholding, I curated a dataset of 360 synthetic ultrasound images, of which a dozen samples are shown in figure 13. These samples appear to be very similar to the generated samples in the previous section, though the colors and amount of blurriness is with more variance. This result is astonishing, given the amount of training data, and number of training epochs.

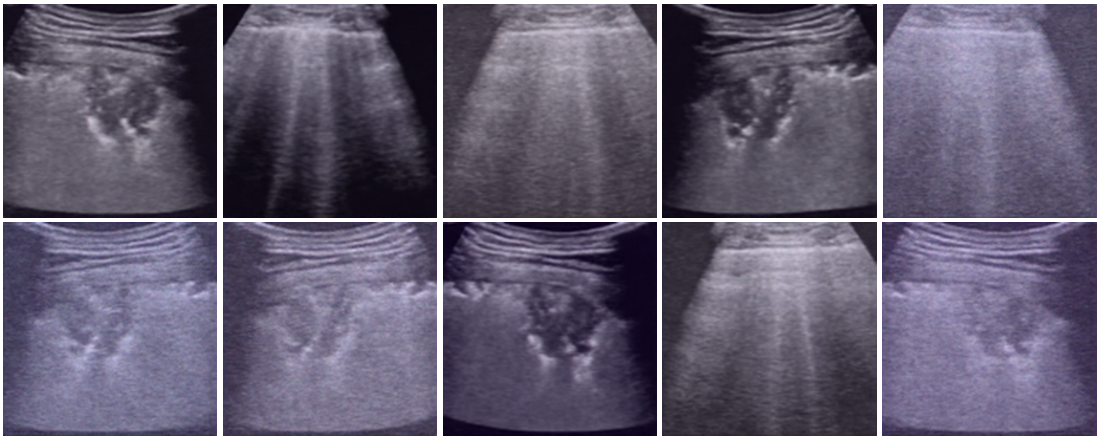

Figure 13: Samples generated using a DDPM model trained on all viral images.

As shown in table 3, the performance metrics dropped for both models in the DDPM all viral experiment. ResNet50 showed an accuracy of 70.72%, an F1 Score of 69.06%, and a Precision of 67.80%. ViT also declined, with an accuracy of 66.89%, an F1 Score of 65.38%, and a Precision of 68.19%. This decline in performance could be due to overfitting or the inability of the models to generalize well under these specific DDPM settings. It suggests a need for further fine-tuning and investigation into the DDPM model parameters and experimental setup. Figure 14 shows the confusion matrices for the ResNet50 model.

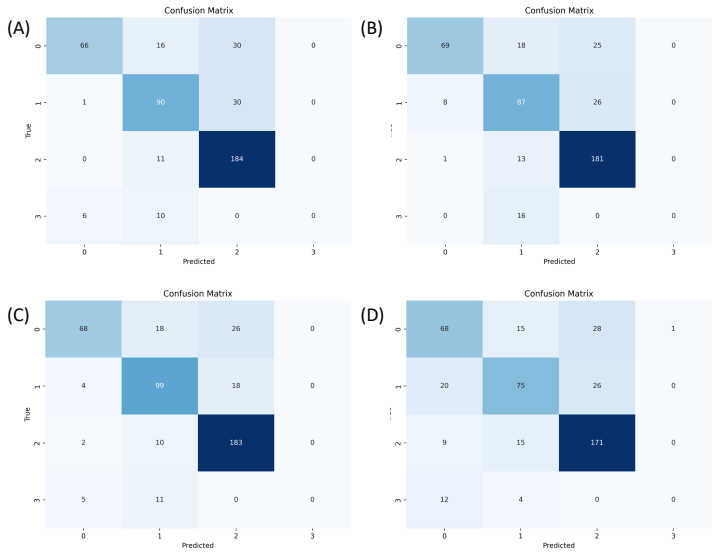

Figure 14: Confusion matrices for the ResNet50 model. (A) Baseline. (B) Class weights. (C) 3 mixed images. (D) All training images.

5. Research limitations

This work was done in the scope of a graduate course, and as such I tried to go as deep as possible under the obvious constrains of time, data, and available compute. The original goal was to use textual inversion and use the generated samples as augmentations for a classification task, exploring the benefits of the method. To my disappointment, textual inversion, and text conditional image generation using stable diffusion and such, is too hard to control – at least when dealing with ultrasound images. The alternative I chose – training a DDPM – showed better performance in terms of ultrasound images generation. However, due to the very small amounts of data used in this setting, these are possibly just memorized variations of the training set, a claim which is supported by recent publications such as [15].

6. Conclusions and future work

In this project, I have undertaken a comprehensive exploration of generating ultrasound images from diffusion models using DDPMs and textual inversion for the goal of improving a classifier. I have experimented and created interesting samples using textual inversion, which resulted in unsatisfying images in terms of ultrasound integrity. The following experiments using DDPMs were not consistent, where unlike 'all training images', the '3 mixed images' experiment showed improvement - backed by the improved classification results shown in table 3. However, as described in the previous section, these samples might be memorized and basically act as oversampling or provide slight augmentations of the original training set.

Future work should include attempts on data distributions with bigger variation in the data, as this research was limited to very small number of the diffusion model training samples. Given a bigger scope, I would have like to create a deeper analysis of the generated images, using various amounts of generated data on the classifier, calculating FID scores, embedding space analysis, and further known methods for generated images integrity.

In conclusion, this work was conducted as the final assignment for the course "ML4Healthcare" in the Technion. Throughout this project, I have acquired both theoretical and practical knowledge regarding various aspects of this crucial AI domain. I had the opportunity to plan, implement, and thoroughly analyze unconditional and text-conditioned diffusion models which significantly expanded my existing knowledge. On a personal level, this course and project has enriched my skills and provided me with a more nuanced perspective on many ML tasks I will encounter in the future. I am grateful for this enlightening experience - thanks!

References

[1] Chambon, Pierre, et al. "Adapting pretrained vision-language foundational models to medical imaging domains." arXiv preprint arXiv:2210.04133 (2022).
[2] Gal, Rinon, et al. "An image is worth one word: Personalizing text-to-image generation using textual inversion." arXiv preprint arXiv:2208.01618 (2022).
[3] Adhikari, Rabin, et al. "Synthetic Boost: Leveraging Synthetic Data for Enhanced Vision-Language Segmentation in Echocardiography." arXiv preprint arXiv:2309.12829 (2023).
[4] Stojanovski, David, et al. "Echo from noise: synthetic ultrasound image generation using diffusion models for real image segmentation." arXiv preprint arXiv:2305.05424 (2023).
[5] https://github.com/ftsvd/USAnotAI
[6] Vitale, S., Orlando, J. I., Iarussi, E., & Larrabide, I. (2019). Improving realism in patient-specific abdominal ultrasound simulation using CycleGANs. International Journal of Computer Assisted Radiology and Surgery, 15(2), 183-192.
[7] https://www.kaggle.com/competitions/ultrasound-nerve-segmentation/data
[8] Gare, Gautam Rajendrakumar, et al. "Weakly Supervised Contrastive Learning for Better Severity Scoring of Lung Ultrasound." arXiv preprint arXiv:2201.07357 (2022).

[9] https://www.scielo.br/j/ramb/a/fsj8QnDqzCK4mjX3d3rZC8L/?lang=en#ModalFigf2
[10] He, Kaiming, et al. "Deep residual learning for image recognition." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.
[11] Dosovitskiy, Alexey, et al. "An image is worth 16x16 words: Transformers for image recognition at scale." arXiv preprint arXiv:2010.11929 (2020).
[12] Deng, Jia, et al. "Imagenet: A large-scale hierarchical image database." 2009 IEEE conference on computer vision and pattern recognition. Ieee, 2009.
[13] Ho, Jonathan, Ajay Jain, and Pieter Abbeel. "Denoising diffusion probabilistic models."Advances in neural information processing systems 33 (2020): 6840-6851.
[14] https://huggingface.co/docs/diffusers/
[15] Carlini, Nicolas, et al. "Extracting training data from diffusion models." 32nd USENIX Security Symposium (USENIX Security 23). 2023.