Partial Eclipse of the Heart: Left Ventricle Segmentation on the EchoNet Dataset

Emma Cruz
Stanford University
Department of Mathematics
emmacruz@stanford.edu

Michael Yu
Stanford University
Department of Electrical Engineering
mjyu@stanford.edu

Abstract

Echocardiograms are commonly used as a cardiac diagnostic tool. One particular area of interest is the left ventricle (LV), the structure of which informs multiple important heart failure metrics. In this project, we evaluated the performance of several deep learning algorithms for LV segmentation of echocardiograms, and compared our results to existing methods.

The two models that we evaluated were Meta's DINOv2 and Segment Anything (SAM), and they were compared against the EchoNet-Dynamic model developed by Ouyang et. al. We found that neither of these models were able to achieve satisfactory segmentation performance as measured by Dice score; however, SAM performed much better than DINOv2. Our average Dice score for SAM with two reference points was 0.584.

Existing segmentation models trained on a general image dataset do not appear to transfer with good accuracy to the task of LV segmentation. Future work including hyperparameter tuning or testing of different models could lead to improved segmentation, with the long-term goal of moving towards fully automated, real-time LV segmentation of echocardiograms.

1. Introduction

Cardiovascular disease (CVD) remains the leading cause of death worldwide, with 17.8 million deaths in 2017 alone attributed to CVD [7]. Simultaneously, CVD remains one of the leading causes of hospital admissions in the United States in recent years [14].

Echocardiography is an important and widely used screening methodology for the diagnosis and assessment of heart failure patients [12, 1]. Over 7 million echocardiograms are performed in the United States annually, each requiring a skilled cardiologist to interpret [3]. Structural areas of interest in cardiac imaging include the left ventricle, right ventricle, and myocardium. However, of these, the structure of the left ventricle (LV) provides important insight into key clinical measures such as ejection fraction and stroke volume, which are oftentimes reduced in patients with heart failure [8]. Thus, it is critical to be able to precisely identify the extent of the left ventricle at multiple timepoints during echocardiogram videos.

Currently, left ventricle traces must be created by a cardiologist manually which is a relatively involved and time-consuming process. Furthermore, only one frame of an echocardiogram can be annotated at a time, limiting the information that can be realistically reported [4]. Development of an accurate and real-time segmentation system would be a significant improvement to the current standard of care.

In this project, we aim to test whether well-known and pretrained image segmentation models could be applied to a novel task of left ventricle segmentation. We furthermore aim to compare the performance of these models with existing models that have been specificially trained for the LV segmentation task. The input to our algorithms will be image frames from echocardiograms and the output will be a mask representing the predicted LV location.

2. Literature Review

There are a variety of previous studies that have used machine learning to analyze echocardiograms. Duffy et al. used a deep learning model with a modified DeepLabV3 backbone in combination with videos of the heart obtained from over 20,000 echocardiograms to measure left ventricular diameter and identify patients with increased left ventricular wall thickness [2]. In a different application, Omar et al. used data from speckle tracking echocardiograms (STE) to create two models: an unsupervised learning model that clustered patients into three groups, which seemed to be based on worsening of cardiac function, as well as a regression to predict several

measures of cardiac pressure such as E/e' using STE data [9].

In 2020, Ouyang et al. developed EchoNet-Dynamic, a model which is able to predict ejection fraction, assess cardiomyopathy, and segment the left ventricle [11]. As part of their publication, they made their dataset of over 10,000 labeled echocardiogram videos publicly available for future research. Similarly to Duffy et al., EchoNet-Dynamic is based on a DeepLabV3 backbone for segmentation tasks. However, instead of only performing segmentation of the LV wall as in Duffy et al., EchoNet-Dynamic segments the entire left ventricle.

Since the publication of the EchoNet-Dynamic database, several groups have attempted to achieve similar or improved segmentation accuracy using different training methodology. One area of interest thus far has been contrastive learning, which is a partially self-supervised algorithm, involving a long unsupervised training portion and then a shorter supervised training portion. This technique could be beneficial in the future not only for LV segmentation but for other cardiac labeling tasks where more limited labeled data is available. Up to now, one group was able to achieve comparable LV segmentation results to the original EchoNet paper while training using only 5% of the labeled data [13]. Another method that has been tried thus far on the EchoNet-Dynamic database involves local attention, allowing the algorithm to better understand pixels in context. One such model, which was called a pyramid local attention neural networks (PLANet), was able to achieve results eclipsing those of the original EchoNet-Dynamic paper while simultaneously achieving real-time segmentation. [5]

3. Methodology

In this report, we evaluate the performance of widely utilized models on the EchoNet-Dynamic dataset and compare their performance to the performance of the model proposed in the original paper. In particular, we were interested in models that are useful for general image segmentation and object detection. We compared the quality of segmentation between these generalized models and a model developed specifically for LV segmentation. A NVIDIA T4 GPU (preconfigured for deep learning on Google Cloud or through Google Colaboratory) was used for all training and evaluation tasks.

3.1. Baseline Method

We first aimed to reproduce similar results to the original EchoNet-Dynamic paper [11] by using the code available in their Github repository. The EchoNet-Dynamic model was trained on the training set for a total of 50 epochs and subsequently evaluated on the validation set. Reproducing the results from the original paper using our available computing resources was important to ensure a fair comparison with future models that we would try.

3.2. Other Models

We also evaluated the ability of two different existing models to perform LV segmentation. These models were developed a number of years after the original EchoNet-Dynamic paper, and as such, we hypothesized that they may do better than the original in either speed or segmentation accuracy.

3.2.1 DINOv2

The first model we tried is Meta's DINOv2 [10]. DINOv2 is designed for not just segmentation but also classification, video understanding, and depth estimation. DINOv2 uses self-supervised learning and claims to be able to learn from any collection of images. We intended to test if this flexibility extended to LV segmentation. One of the main upsides of DINOv2 compared to other models is the theoretical "out of the box" function, as the developers found in the original paper that DINOv2 was able to perform well on new segmentation tasks without any finetuning. Furthermore, they found that finetuning only slightly increased the performance (about a 2% increase in accuracy) compared to only training a linear classifier layer on top of the backbone (linear probing).

Thus, for this project, we evaluated the ability of DINOv2 to perform quality LV segmentation using a pretrained frozen backbone with a linear classifier. Our linear probing was based on a tutorial developed for semantic segmentation with DinoV2 but modified for our specific task, loss, and dataset. The original tutorial used a batch size of 2, but we also tested a batch size of 32. For the linear classifier, we used a convolutional layer with a kernel size of 1×1 and an input channel dimension of 768 (which is the hidden output dimension from the DINOv2 backbone). As we learned in class, convolutional layers can be used to represent a linear classifier.

3.2.2 Segment Anything

The second model we tried is also from Meta: Segment Anything (SAM). Segment Anything can be prompted with graphical or text inputs and has the ability to create masks from them. The authors claim to have the largest segmentation dataset to date (2023). They also claim its design makes it well suited to transfer learning and zero-shot; we wanted to see if this could transfer to the LV segmentation task (where it is unlikely their model has

been trained on such images).

SAM can take as input a set of points and/or bounding boxes. For the point inputs, they can be set as either foreground points or background points; that is, they can tell the model to include a specific region in the mask, or to exclude it. While it would in theory be possible to train a network on top of this to generate these "prompts," we were interested in the model's zero-shot performance. However, since we would still need some point(s) to feed in as input, we decided to use the traces provided in the data set. The idea is that rather than being fully automated, a physician might select a few points and easily generate a segmentation.

We ran the model with two different types of inputs: one-point and two-point inputs. For the one-point inputs, we calculated a midpoint from the provided traces and used only this point as a "foreground" input. This input might correspond to a technician selecting a single point in the center of the left ventricle. For the two-point inputs, we calculated a midpoint as well as two additional co-linear points outside of the left ventricle (with these two additional points being designated as "background" points). This input might correspond to a technician selecting two points on the boundary of the left ventricle.

3.3. Evaluation Metrics

The primary metric of evaluation is Dice score, according with the metric for previous echocardiogram segmentation studies. For two different areas A and B, Dice score is defined as

$$2\frac{|A \cap B|}{|A| + |B|}.$$

In this study, the two areas under comparison will be the model-predicted segmentation area and the "ground truth" area from the labeled data. A Dice score of 0 represents no overlap, while 1 represents perfect agreement.

We will secondarily evaluate training loss curves. The loss functions that we used for this project are all cross-entropy loss functions, taking the following form:

$$\text{Loss} = -\frac{1}{N}\sum_{i=1}^{N} S^{\star}(p_i) \log\left(\frac{\exp S(p_i)}{\sum_{j=1}^{N} \exp S(p_j)}\right)$$

where N is the total number of pixels, $S^{\star}$ is the ground truth segmentation, S is the predicted segmentation, and p_i is the ith pixel.

The original EchoNet-Dynamic paper used pixel-wise binary cross entropy with logits loss (torch.nn.functional.binary_cross_entropy_with_logits),

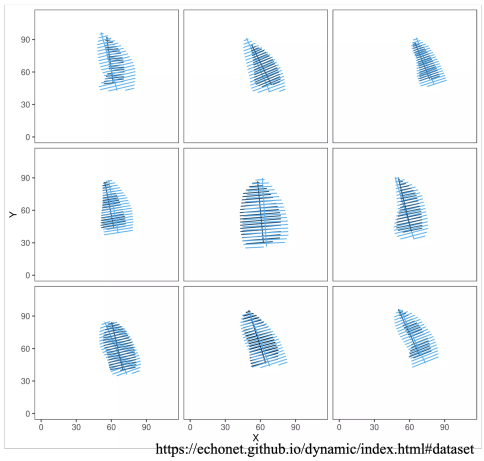

Figure 1. The expert left ventricle segmentations are represented in the EchoNet-Dynamic dataset as paired coordinates, tracing out the area of the ventricle as seen above. (Image from [11])

which is a cross-entropy loss specifically utilized when there are only two possible classes. This loss incorporates a sigmoid function $\sigma(x) = (1 + e^{-x})^{-1}$ before calculating the cross entropy loss to transform logits into probabilities.

For the DINOv2 model, the original tutorial utilized a generic cross entropy loss (torch.nn.CrossEntropyLoss), so we tested both this loss and the binary cross entropy with logits loss to see which one would result in better performance.

In our trials with SAM, we did not calculate loss as we only ran zero-shot prediction. To quantify the quality of our generated masks, we used Dice score with the labeled data.

Lastly, as qualitative verification, we evaluated a sample of segmentation masks by projecting them on top of the echocardiogram videos. This is important to allow us to visualize how our algorithms are performing as well as to identify and understand any frequent areas of confusion.

4. Dataset and Preprocessing

We utilized the publicially available Echonet-Dynamic dataset [11] to train and evaluate all models. The dataset contains a total of 10,030 videos which are split into train, validation, and test splits of 7,465, 1,277 and 1,288 videos.

Each video in the EchoNet dataset has two labeled frames: one at systole (when the heart is most contracted) and one at diastole (when the heart is most relaxed). These frames are annotated with (human) expert left ventricle segmentations. These segmentations are represented in the dataset as a series of coordinates as seen in Figure 1. The

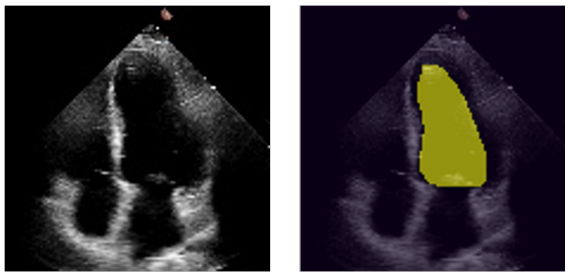

Figure 2. (Representative data). A frame extracted from an echocardiogram (left) with a provided left ventricle segmentation mask overlaid (right) for training.

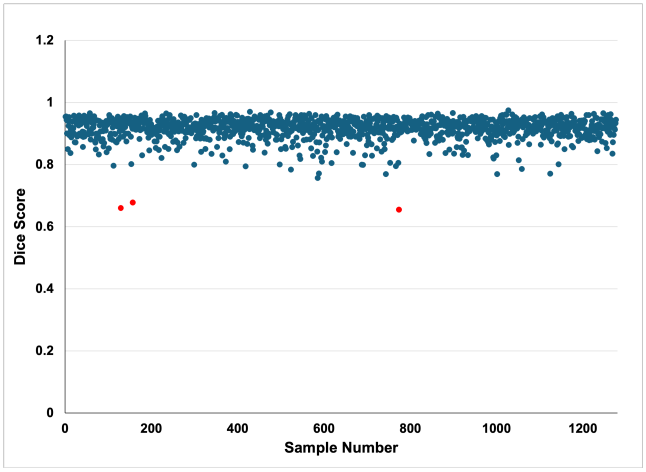

Figure 3. Dice scores for all videos in the test set. Videos with Dice score below 0.7 are highlighted in red.

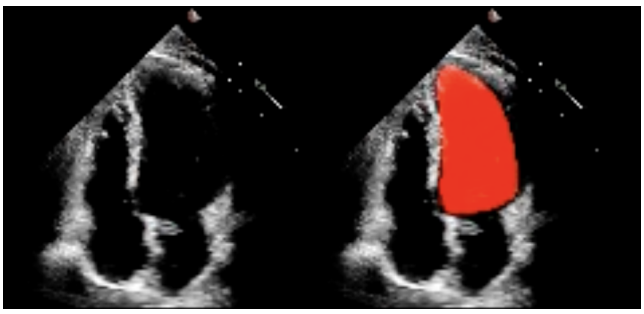

Figure 4. A representative echocardiogram frame with overlaid left ventricle segmentation mask on the right.

first ordered pair of coordinates represents the long axis of the left ventricle, while the other pairs represent evenly spaced estimations of the shorter, more horizontal axis.

We slightly modified a script found in [13] to extract these labeled frames and create LV mask tensors for training purposes, which produced the image and mask pairs shown in Figure 2. The images and masks were each 112×112 pixels and normalization was applied prior to training.

5. Results

5.1. Baseline

Training the EchoNet-Dynamic model for 50 epochs took about 17 hours ($\approx$ 20 minutes per epoch). The best loss during training was 0.035, achieved during epoch 26, computed as a pixel-wise cross entropy loss.

We then evaluated the trained baseline model on the validation and test sets. The average Dice score over all videos was **0.909** for the validation set and **0.914** for the test set. We visualized the Dice scores for the videos in the test set (Figure 3) and noticed that almost all videos were segmented very well with only a few outliers below a Dice score of 0.7.

Lastly, we visualized the segmentations for a random sample of 50 videos from the test set. As seen in the representative example image in Figure 4, the segmentation was very accurate in most cases.

5.2. DINOv2

As a baseline, we first tested how DINOv2 would segment echocardiogram images if given absolutely no additional training beyond the pretrained backbone. To do this, we used the Semantic Segmentation Demo Lab available online. We found that individual cardiac chambers were not recognized as different objects or features and thus segmentation of the left ventricle could not be automated from just the pretrained backbone (Figure 5). That is, zero-shot learning did not seem to be effective for this use case.

We then moved to training a linear classifier on top of the pretrained DINOv2 backbone loaded from the Transformers python package. All experiments were run using the AdamW optimizer [6] and a learning rate of 5×10^{-5}. We began our experiments by using the loss function (cross entropy) and batch size (2) demonstrated in the tutorial, but used Dice score as our evalutation metric as mentioned above instead of intersection over union (IOU)

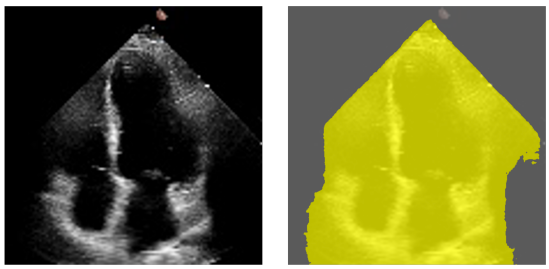

Figure 5. Initial segmentation without linear probing demonstrates that DINOv2 recognizes the basic echocardiogram shape but is unable to segment individual cardiac features.

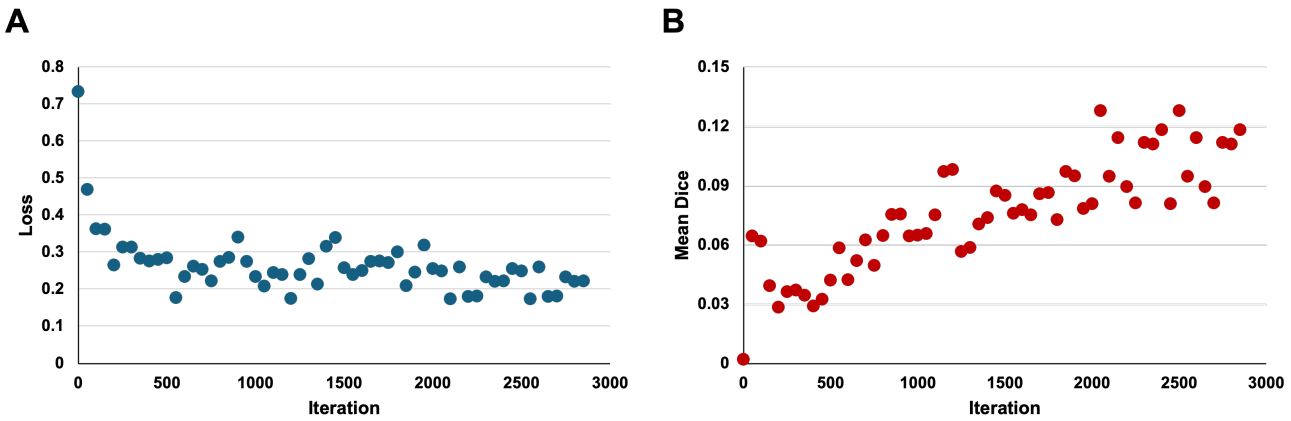

Figure 6. Experimental results for DINOv2 with batch size 2. (A). Loss curve computed as binary cross entropy loss with logits over 3000 training iterations. (B). Mean Dice score (averaged every 50 training iterations).

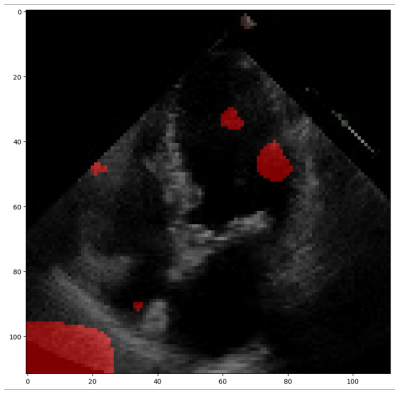

Figure 7. Example segmentation of a validation example using the DINOv2 model + linear classifier trained with batch size 2.

to be consistent with the baseline EchoNet model. With this initial experiment, we found that the loss function originally decreased which seemed to show that the model was learning, but the instead of increasing towards 1, the Dice score also decreased until it approached zero, which led us to hypothesize that our loss function was not properly capturing the information important for learning.

Because of this, we next attempted training the linear classifier with the binary cross entropy with logits loss instead since that is what the original paper used. Our results are visualized in Figure 6. We found that after an initial sharp decrease in loss, the loss was mostly stagnant and the mean Dice score increased very slowly, never reaching above 0.15. Furthermore, visualizing the predicted mask on a test example (Figure 7) showed that the model was unable to properly understand the location of the left ventricle.

With these failed results in mind, we decided to test if a larger batch size would improve the segmentation quality. We retrained the linear classifier on top of the DinoV2 backbone using a batch size 32. Although we did not have enough time to fully train the model with this larger batch size (since each iteration was much slower, taking about 50 seconds for one iteration), we noticed that the Dice score increased much more quickly after only a few iterations and increased above 0.15 after less than 20 iterations, which seemed more promising than the batch size 2 results (Figure 8). However, when we visualized the actual segmentation (Figure 9), we noticed that the LV segmentation was still horrible and that in this case the model was essentially predicting the entire image to be the LV.

Overall, we were unable to acheive any correct segmentations using linear probing with the DINOv2 pretrained backbone. However, we learned that the batch size 32 results may look more promising in terms of Dice score (although still not good), so if we had enough time to train for multiple epochs on the full training set (which we estimate would take >6 hours per epoch), there is a possibility that we would be able to achieve more accurate segmentations.

5.3. Segment Anything

We first tried running SAM with just a single point as input. This one-point input proved to be rather ineffective. In Figure 10, we can see the three masks that SAM outputs; these three masks each have an associated score. While there is a mode which only outputs a single mask, the authors of SAM suggest using this three-output mode and picking the highest score for the best results. In these three masks, we can see that the first one appears to best segment the left ventricle. However, the highest score segmentation is the last mask, which instead segments the entire heart.

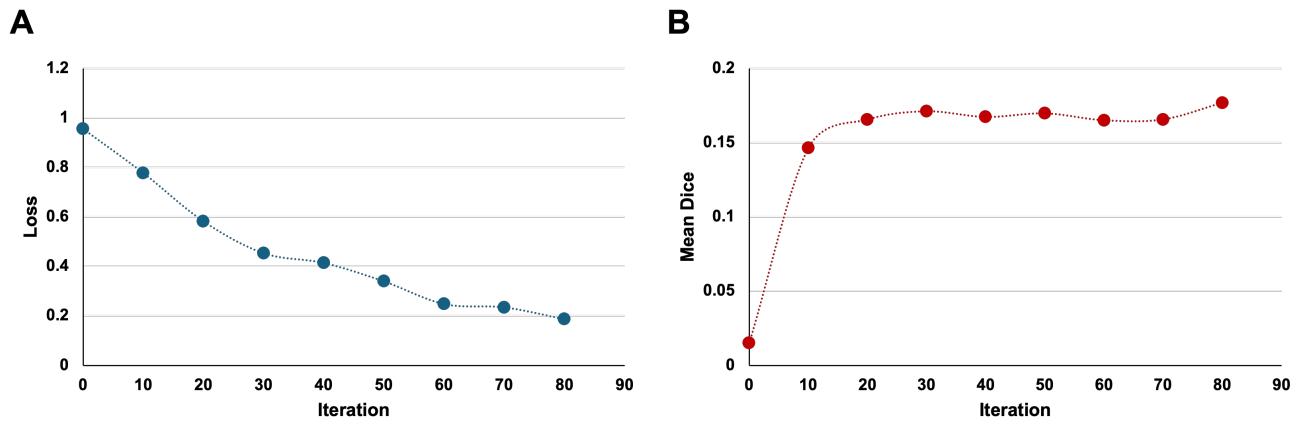

Figure 8. Experimental results for DINOv2 with batch size 32 over 85 iterations. (A). Loss curve computed as binary cross entropy loss with logits over 3000 training iterations. (B). Mean Dice score (averaged every 10 training iterations).

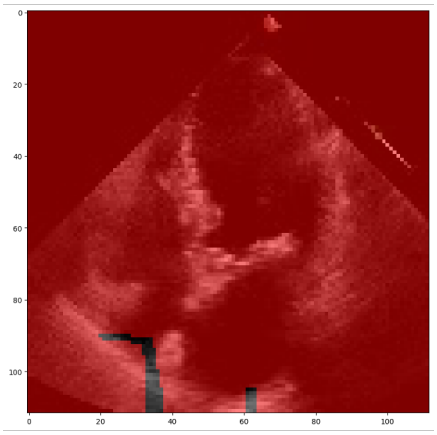

Figure 9. Example segmentation of a validation example using the DINOv2 model + linear classifier trained with batch size 32.

We ran the one-point model on the entire test set and calculated the Dice scores which are shown in Figure 12(A). The mean of these scores was 0.395. We can see from this distribution that while many of the masks have a score above 0.8, the majority are clustered below 0.4. Interestingly, there appears to be a distinct gap between these two regions. This is likely due to the model segmenting either the entire heart (as in the bottom image of Figure 10) resulting in the lower scores, or segmenting the left ventricle (as in one of the top two images of Figure 10) resulting in the higher scores.

We then ran SAM on a "two-point" input. In Figure 11, we can see an example of the three outputted masks. Note that the green star on the images corresponds to the "foreground" point while the two red stars correspond to the "background" points. While the masks appear to be similar to the one-point input masks, if we look at the scores, we notice that the highest score is now in fact the middle image. While this isn't perfect, it is much better than the entire heart.

The Dice scores are plotted in Figure 12(B). The mean is shown in red and has a value of 0.584. Note here that there is a significant number of points clustered towards the top; this suggests that a greater proportion of segmentations result in a mask like one of the first two in Figure 11. There are, however, still many masks that are covering the entire heart.

6. Discussion

Although we were able to replicate the baseline and achieve similar results (Dice > 0.9) compared to the original paper [11], neither of the general image segmentation models that we attempted to apply to this task were able to learn comparable or better segmentations.

For the DINOv2 model, we identify two potential reasons why the segmentation performance was significantly worse than the baseline. One potential reason would be that we were using suboptimal hyperparameters, including batch size, learning rate, and so on, or a suboptimal optimizer. Because running each experiment took several hours of training before we could start to get an understanding of whether it seemed to be working, it was not feasible to find a good combination of hyperparameters in the time that we had for this project. For example, we did test different batch sizes and losses, but we were unable to experiment with different learning rates and optimizers. This hypothesis may be supported by the fact that batch size 32 resulted in more promising loss and Dice score curves compared to batch size 2, even though the predicted segmentation continued to look bad.

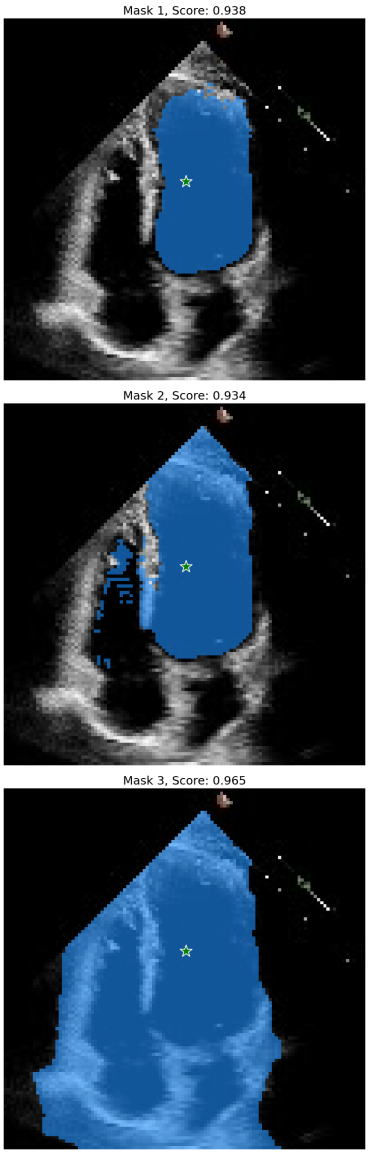

Figure 10. The three generated masks for one-point input. Note that while the first mask appears to be the best, it does not have the highest score.

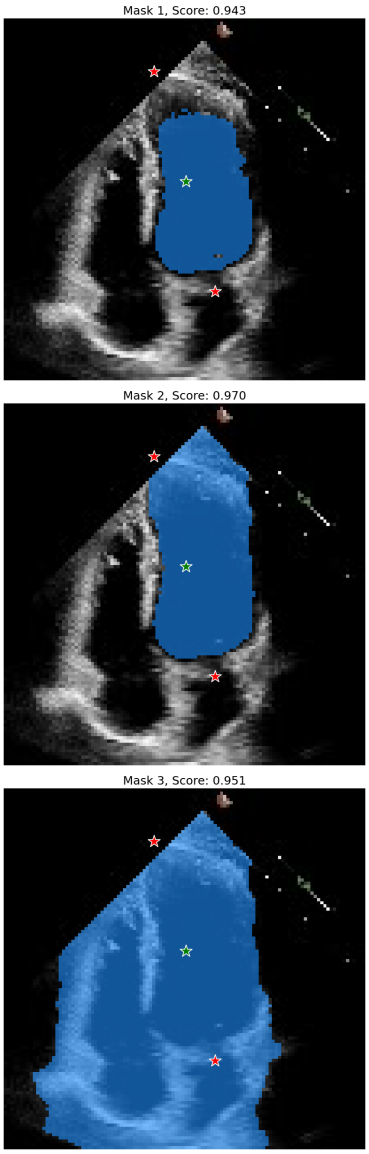

Figure 11. The three generated masks for two-point input. Note that while the first mask appears to be the best, it once again does not have the highest score.

Another potential reason for failure of this model could be that the backbone would need to be finetuned on medical image-specific datasets instead of the general image dataset that it was trained on (ImageNet-22k according to [10]). Although the original DINOv2 paper noted that finetuning did not tend to help the performance significantly, this was for evaluating the performance on unseen datasets such as CityScapes and PascalVOC, neither of which are medical image datasets. However, finetuning the entire model (which contains $>$1 billion parameters) would be extremely computationally expensive so it would probably not be feasible with our available resources.

The difficulty in using SAM lies in the ambiguity of segmentation. With just a single point as a prompt, there is nothing to indicate to the model what we are interesting in segmenting. It is likely the distinct edges that resulted from rotating the echocardiagram videos which result in the full heart mask having the highest score. When we prompted SAM with more points, we see that it can no longer accept the bottom portion of the image as part of the mask. However, the sharp edges at the top are not excluded and thus result in the highest score.

The more points we prompt the model with, the better the

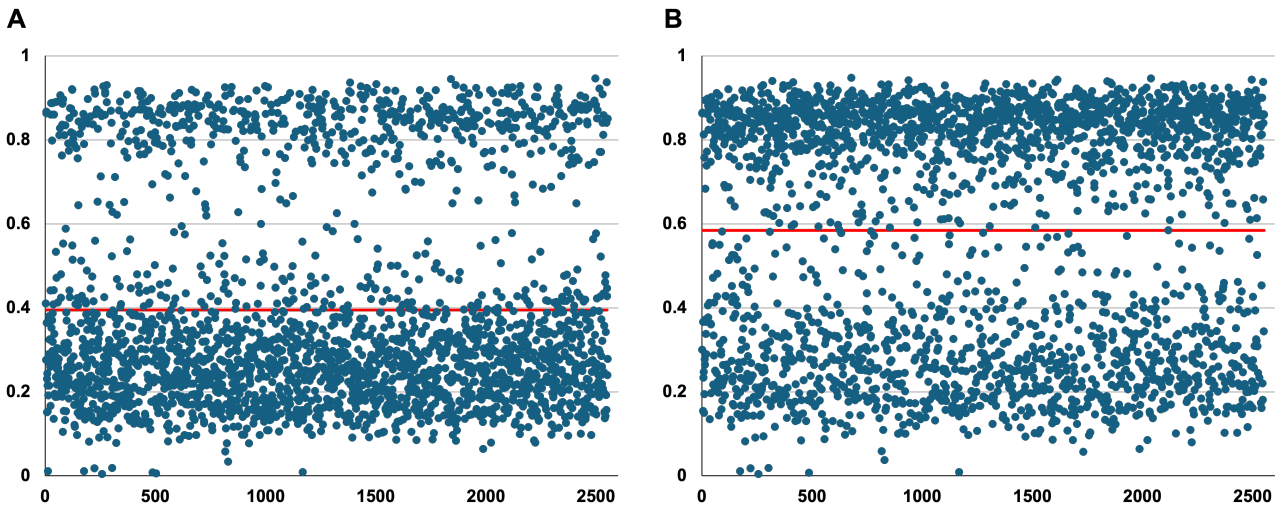

Figure 12. Dice scores of SAM masks vs labeled data for 2552 frame test split (A). Dice scores of one-point inputs with mean line (Dice $= 0.395$) plotted. (B). Dice scores of two-point inputs with mean line (Dice $= 0.584$) plotted.

segmentation would be. However, this would mostly defeat the purpose of using the model since manual segmentation only involves selecting a handful of points anyway. In fact, the method of using points as prompts makes this ill-suited for our left ventricle segmentation task, as we want this to be fast and completely automatic. The task of figuring out which section of the image to choose is the real technical challenge here and is one that SAM does not accomplish; selecting a specific dark region of the image can be done with much less powerful tools (and indeed the superior performance of the baseline model validate this point). SAM is perhaps better suited for identifying several non-trivially distinct objects.

6.1. Future Work

Future work to continue this project of automated or semi-automated left ventricle segmentation could follow three main pathways.

First, if we had additional time and computing resources we could re-attempt some of the experiments that we mentioned above to see if we could achieve improvements. For DINOv2, as mentioned above, this could include either additional hyperparameter tuning, experimenting with different optimizers or loss functions, or finetuning the backbone instead of only training a linear classifier. For SAM, we could attempt to use more abstract inputs as the paper claims is possible (for example, text prompt for "large dark region in the top"). This would allow for the system to become fully automatic.

Furthermore, an interesting and compelling next step could be testing the performance of additional preexisting models. One such model is YOLOv8 from Ultralytics. YOLOv8 is based on the principles of the original YOLO which performed object detection and classification in one network. This iteration of YOLO also has the ability to do image segmentation with good speed, which means that if YOLOv8 was able to segment the LV accurately, it could open up the possibility of real-time automatic LV segmentation.

Lastly, we could explore an entirely different direction in future work. So far, contrastive learning as mentioned in [13] seems particularly promising. Previous models have achieved comparable Dice scores to the original EchoNet paper using supervised training on only 5% of the full training set. If we had additional time, we could attempt to expand along this avenue by investigating the effects of additional augmentations or other improvements to the current contrastive learning models.

6.2. Conclusion

Overall, we evaluated the performance of two pretrained image segmentation models on a novel task of LV segmentation. Although these models were not able to segment as accurately as the baseline, the performance of Meta's Segment Anything (SAM) was relatively promising and could potentially lead to future semi-automated segmentation with only two required input points. Considering the huge number of echocardiograms performed yearly, additional future work to develop accurate and fast LV segmentation models is warranted to improve the quality of cardiac care.

7. Author Contributions

Emma Cruz and Michael Yu contributed equally to this work. E.C. and M.Y. designed and carried out the experiments, analyzed the data, and wrote the report.

References

[1] ACCF/ASE/AHA/ASNC/HFSA/HRS/SCAI/SCCM/SCCT/SCMR 2011 Appropriate Use Criteria for Echocardiography. *Journal of the American Society of Echocardiography*, 24(3):229–267, Mar. 2011. 1

[2] G. Duffy, P. P. Cheng, N. Yuan, B. He, A. C. Kwan, M. J. Shun-Shin, K. M. Alexander, J. Ebinger, M. P. Lungren, F. Rader, D. H. Liang, I. Schnittger, E. A. Ashley, J. Y. Zou, J. Patel, R. Witteles, S. Cheng, and D. Ouyang. High-Throughput Precision Phenotyping of Left Ventricular Hypertrophy With Cardiovascular Deep Learning. *JAMA Cardiology*, 7(4):386–395, Apr. 2022. 1

[3] S. Ferraz, M. Coimbra, and J. Pedrosa. Assisted probe guidance in cardiac ultrasound: A review. *Frontiers in Cardiovascular Medicine*, 10:1056055, Feb. 2023. 1

[4] T. Kim, M. Hedayat, V. V. Vaitkus, M. Belohlavek, V. Krishnamurthy, and I. Borazjani. Automatic segmentation of the left ventricle in echocardiographic images using convolutional neural networks. *Quantitative Imaging in Medicine and Surgery*, 11(5):1763–1781, May 2021. 1

[5] F. Liu, K. Wang, D. Liu, X. Yang, and J. Tian. Deep pyramid local attention neural network for cardiac structure segmentation in two-dimensional echocardiography. *Medical Image Analysis*, 67:101873, Jan. 2021. 2

[6] I. Loshchilov and F. Hutter. Decoupled Weight Decay Regularization, Jan. 2019. arXiv:1711.05101 [cs, math]. 4

[7] G. A. Mensah, G. A. Roth, and V. Fuster. The Global Burden of Cardiovascular Diseases and Risk Factors. *Journal of the American College of Cardiology*, 74(20):2529–2532, Nov. 2019. Publisher: American College of Cardiology Foundation. 1

[8] S. P. Murphy, N. E. Ibrahim, and J. L. Januzzi, Jr. Heart Failure With Reduced Ejection Fraction: A Review. *JAMA*, 324(5):488–504, Aug. 2020. 1

[9] A. M. S. Omar, S. Narula, M. A. Abdel Rahman, G. Pedrizzetti, H. Raslan, O. Rifaie, J. Narula, and P. P. Sengupta. Precision Phenotyping in Heart Failure and Pattern Clustering of Ultrasound Data for the Assessment of Diastolic Dysfunction. *JACC. Cardiovascular imaging*, 10(11):1291–1303, Nov. 2017. 2

[10] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, M. Assran, N. Ballas, W. Galuba, R. Howes, P.-Y. Huang, S.-W. Li, I. Misra, M. Rabbat, V. Sharma, G. Synnaeve, H. Xu, H. Jegou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski. DINOv2: Learning Robust Visual Features without Supervision, Feb. 2024. arXiv:2304.07193 [cs]. 2, 7

[11] D. Ouyang, B. He, A. Ghorbani, N. Yuan, J. Ebinger, C. P. Langlotz, P. A. Heidenreich, R. A. Harrington, D. H. Liang, E. A. Ashley, and J. Y. Zou. Video-based AI for beat-to-beat assessment of cardiac function. *Nature*, 580(7802):252–256, Apr. 2020. Publisher: Nature Publishing Group. 2, 3, 6

[12] M. C. Pastore, G. E. Mandoli, H. S. Aboumarie, C. Santoro, F. Bandera, A. D'Andrea, G. Benfari, R. Esposito, V. Evola, R. Sorrentino, P. Cameli, S. Valente, S. Mondillo, M. Galderisi, M. Cameli, and on behalf of the Working Group of Echocardiography of the Italian Society of Cardiology. Basic and advanced echocardiography in advanced heart failure: an overview. *Heart Failure Reviews*, 25(6):937–948, Nov. 2020. 1

[13] M. Saeed, R. Muhtaseb, and M. Yaqub. Contrastive Pretraining for Echocardiography Segmentation with Limited Data. In G. Yang, A. Aviles-Rivero, M. Roberts, and C.-B. Schönlieb, editors, *Medical Image Understanding and Analysis*, pages 680–691, Cham, 2022. Springer International Publishing. 2, 4, 8

[14] H. M. Salah, A. M. K. Minhas, M. S. Khan, A. Pandey, E. D. Michos, R. J. Mentz, and M. Fudim. Causes of hospitalization in the USA between 2005 and 2018. *European Heart Journal Open*, 1(1):oeab001, Aug. 2021. 1