Medical Image Analysis Of Cardiac Dataset Using Swin Transformer: Hierarchical Vision Transformer using Shifted Windows

Sravanth Potluri UND5UV@VIRGINIA.EDU
Ganesh Nanduru BAE9WK@VIRGINIA.EDU
Swakshar Deb SWD9TC@VIRGINIA.EDU
Rahul Reddy DQB5TY@VIRGINIA.EDU

Abstract

Traditionally, transformers have been used within the Natural Language Processing (NLP) field of Machine Learning. However, there has been a growing trend in using transformers for Computer Vision tasks. This is not without new, domain-specific consequences and drawbacks. The Swin Transformer hopes to alleviate these consequences by introducing a new methodology that allows for success in dense Computer Vision tasks such as objection detection and semantic segmentation. In this paper, we will give an Introduction and Background to the Swin Transformer Architecture, along with its video variant. We also explain the experiment conducted on implementing the Swin Transformer on the Cardiac Dataset for a regression task. We report our results and inferences using the Swin Transformer and it's video variant along with a custom 3D-CNN Model. We discuss these results and provide a conclusion and discussion on future scopes of using Swin Transformer for Medical Image Analysis. Our code to preprocess the dataset and reproduce experiments can be found on Github.

1. Introduction (Sravanth, Rahul)

1.1. Problem Statement And Introduction To SwinTransformer

Medical image analysis, particularly cardiac imaging, presents significant challenges due to the intricate temporal and spatial patterns present in dynamic imaging data. Traditional convolutional neural networks (CNNs), while foundational to computer vision tasks, often struggle to adapt to the complex requirements of such data, particularly in tasks that demand precise temporal regression like mapping video sequences to Time-Of-Systole (TOS) curves. Transformers, originally designed for Natural Language Processing (NLP) tasks, offer a promising alternative due to their ability to model long-range dependencies. However, their direct adaptation to computer vision tasks faces challenges, such as the inability to efficiently handle high-resolution image data and the lack of mechanisms to process hierarchical visual features effectively.

The Swin Transformer emerges as a compelling solution to these challenges, introducing a novel architecture that constructs hierarchical feature maps and reduces computational complexity. Unlike traditional vision transformers, which compute self-attention globally with quadratic complexity, the Swin Transformer achieves linear complexity by computing self-attention within non-overlapping local windows. Furthermore, by gradually merging image patches in deeper layers, it creates a representation that captures both local and

© 2024 S. Potluri, G. Nanduru, S. Deb & R. Reddy.

global features, addressing the multi-scale nature of visual elements. This architecture is particularly suited to dense prediction tasks, as it facilitates integration with advanced techniques like feature pyramid networks (FPN) or U-Net for tasks like object detection, semantic segmentation, and now we try to apply it to the medical image analysis domain.

In this paper, we use the Swin Transformer's video variant for a particular medical image analysis task. The task involves regressing short video sequences, each containing fewer than 25 frames, into TOS curves with 126 points, requiring robust extraction of temporal and spatial features. By adapting the Swin Transformer's hierarchical approach and efficient self-attention mechanism to this domain, we experiment with these models to see if it can bridge the gap between frame-level information and curve-level regression. We compare its performance against a custom 3D-CNN model and evaluate both models' effectiveness on a cardiac dataset.

1.2. Literature Review

Traditionally CNNs have been considered the gold-standard for Computer Vision tasks, beginning with AlexNet in 2011. Other published CNN works have built off this example, and have continued to advance in the space. The original paper highlights past works that have also been commonly used in computer vision tasks, including VGG, GoogleNet, ResNet, DenseNet, HRNet, and EfficientNet. These models have traditionally been used for common Computer Vision tasks such as image classification and object detection. Through years of development, progress has been made to improve the convolutional layers in CNNs. However, there has also been considerable work done to use self-attention mechanisms in the backbone of Machine Learning architectures.

Self attention is a mechanism used in contextualization, and in Computer Vision, self-attention is used to understand the dependencies between parts of an image. The paper mentions that past works have explored replacing some convolutional layers in CNN with self-attention layers as a means to improve accuracy. However, the memory access to perform this task is significant, and results in higher latency. There has also been consideration into augmenting self-attention layers into CNNs rather than replacing them as well in order to capture both localized and broadened contextualization.

Vision Transformers (ViT) have also played an integral role in Computer Vision tasks, and particularly within the medical image analysis space. Vision Transformers make use of the Transformer architecture on image patches for image classification. However, ViT require a significant amount of data in order to perform well on classification tasks. An alternative vision transformer architecture, Data-Efficient Vision Transformers (DeiT), can be used to bolster accuracy with limited datasets, but ViTs typically do not perform well on dense classification tasks such as semantic segmentation and objection detection. When applied to the medical image analysis community, both CNNs and ViTs have been used for a multitude of usecases, to include cardiac imaging, multi-organ segmentation and various tumor segmentation tasks (Fahad Shamshad and Fu, 2022), (D. R. Sarvamangala, 2022). This will be further highlighted in the background.

1.3. Challenge

The primary challenges in this work arise from the nature of the cardiac dataset and the task of regressing video sequences into TOS curves. Cardiac videos exhibit intricate deformations and ring-like structures over time, making it challenging to capture temporal evolution and spatial deformation simultaneously. Additionally, there is a significant disparity between the number of video frames and the corresponding TOS curve points, as each video contains fewer than 25 frames while the curve comprises 126 points, necessitating the extraction of additional information from each frame to predict the entire curve accurately. Adapting the Swin Transformer, which was not natively designed for Medical Image analysis, to effectively handle such dense temporal regression tasks in medical imaging is another key challenge. Moreover, the performance of the Swin Transformer must be evaluated against well-established models like 3D-CNNs to justify its suitability for medical image analysis. These challenges frame the scope of our research and highlight the complexities of applying transformer-based architectures to medical imaging tasks.

1.4. Summary

In this paper, we systematically investigate the application of the Swin Transformer and its video variant for a medical image analysis task involving cardiac video sequences. Following the challenges identified, we delve into the background of CNNs, Transformers, and Video Transformers, and examine how these architectures have been utilized in the medical imaging domain. The methodology section outlines the design and adaptation of the Swin Transformer and Video Swin Transformer, along with a baseline 3D-CNN model, for the task of regressing cardiac video sequences into Time-of-Systole (TOS) curves. This is followed by an experimental analysis that compares the performance of these models on the cardiac dataset. Finally, we present our findings, discuss their implications for the broader domain of medical image analysis, and propose future directions to overcome the current limitations and optimize these models for clinical applications.

2. Background (Rahul)

Understanding the depth of work that has been conducted in the space illuminates the transition in different modeling approaches over time to solving common medical imaging tasks. In this section, we will discuss CNNs, Transformers, and Video Transformers, and how these works have been applied in medical imaging contexts.

2.1. CNNs

CNNs have been long been regarded as the gold-standard for Computer Vision tasks, and show great promise in transfer learning tasks within medical image analysis as well. CNNs usually consist of three main layers: Input Layers, Hidden Layers, and Output Layers. Input layers are typically used to process the original image into the model, and the size of the input layer is usually determined by the sizes of the images that need to be processed. Hidden layers consist of multiple types of layers, which include convolutional layers, pooling layers, and fully-connected layers. Convolutional Layers provide a means of feature extraction from the data, whereas pooling layers seek primarily to reduce the dimensionality of

the data representation, and make the data more computationally efficient. Fully-connected layers seek to connect to each node in the previous layer in the CNN, and are used to process the data so that prediction outputs can be given.

Often times, multiple convolutional layers are used in CNNs in order to understand patterns that exist within images. Medical professionals have also used CNNs for brain tumor segmentation and further classification between benign and malignant tumors (D. R. Sarvamangala, 2022). Furthermore, in some cases, CNNs have outperformed medical experts. CheXNet, a CNN-based model, was used to classify 14 different chest ailments and achieved better results when compared to the average results from medical experts (D. R. Sarvamangala, 2022).

2.2. Transformers

Initially created specifically for machine translation in NLP, Transformers were originally developed in order to capture and contextualize information from entire sentence sequences. This idea was later adapted to create a new Computer Vision model called Vision Transformers (ViT). ViTs are built on top of a vanilla Transformer model by incorporating multiple transformer layers in order to capture global dependencies in an image. An overview of the ViT architecture is shown in Figure 1.

Because the original transformer handles 1-D input, an image $x \in \mathbb{R}^{H\times W\times C}$ must be transformed into a sequences of patches that are defined in the original paper as x_p $\in$R$^{N\times(P^2\times C)}$, where H and W represent the height and width of the original image, C is the number of channels, (P,P) is the dimensionality of each patch, and $N = HW/P^2$ is the number of patches (Dosovitskiy, 2020). These patches are then flattened and mapped to D dimensions through a linear projection. Furthermore, a prepend a learnable embedding and positional embedding are attached to each image patch to capture the representation and position of the patches respectively.

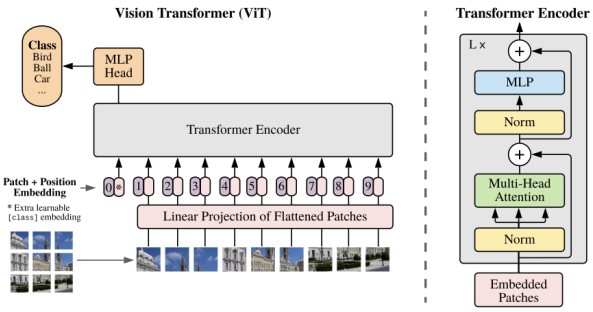

Figure 1: ViT Transformer Architecture

The characteristics of ViTs make this model an excellent choice for medical imaging tasks. ViTs contextualize pixels from the entire image, and can provide modeling insights between two distant pixels in the image. In fact, ViTs have numerous applications in organ-specific segmentation and multi-organ segmentation. Organ-specific segmentation

can be divided into two sections: 2-D Segmentation tasks, and 3-D Segmentation tasks. 2-D Segmentation tasks include Skin Legion Segmentation for Melanoma detection, Cell Segmentation, and Cardiac Imaging. 3-D Segmentation tasks include brain tumor and breast tumor segmentation. Multi-Organ segmentation tasks seek to classify multiple organs at once, and can be difficult due to class imbalances. However, ViT models have shown great promise, especially with the use of hybrid models which draw upon both ViT and CNN architectures (Fahad Shamshad and Fu, 2022).

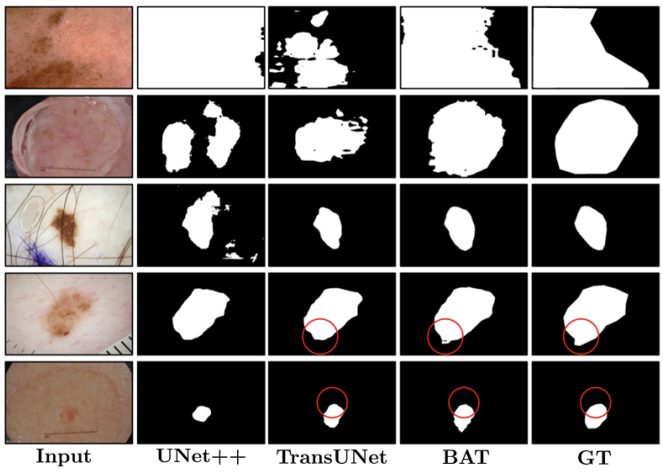

Figure 2: Skin Lesion Image Segmentation across different models, including the ViT-based model TransUNet

An example of a pivotal Transformer-based model in the medical image analysis domain is UNETR, which makes use of Transformers for 3-D Medical Image Segmentation. The UNETR model draws inspiration from the U-Net Deep Learning framework, and the Transformer architecture to not only capture multi-scale information but also to preserve the encoder-decoder structure. The BTCV and MSD datasets were used to assess the validity of the UNETR model, which consist of abdominal CT scans and brain tumor CT scans respectively. Results showed that the UNETR model had an average accuracy of 0.891 on the BTCV dataset, and a dice score accuracy of 0.964 on the MSD dataset (Dosovitskiy, 2021).

2.3. Video Transformer

After discussing extensively about Transformers, this knowledge noted above can be used to discuss the basis of Video Transformers. Video Transformers are simply Transformer-based Machine Learning frameworks that have been designed to process video data. Past works in the background have discussed image processing for one image at a time, however many Video Transformer methods build on top of the ViT structure and incorporate temporal attention mechanisms in order to process each frame of a video. The inspiration behind the video transformer came from the success of the ViT.

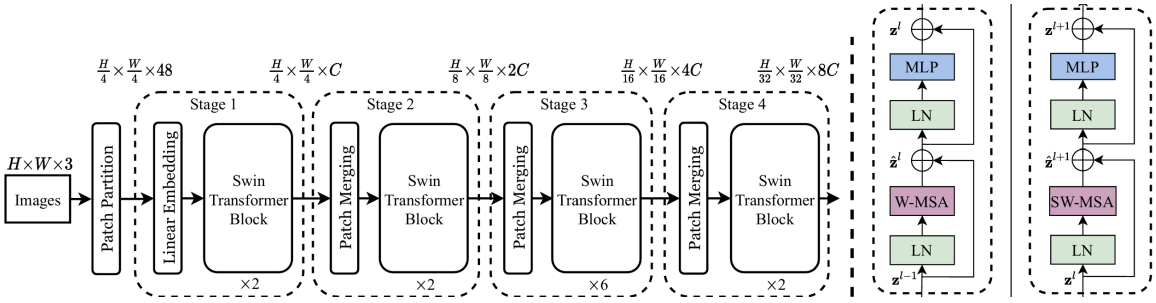

Figure 3: (a) Overview of the Swin Transformer architecture; (b) two subsequent Swin Transformer blocks. W-MSA and SW-MSA denote window (S) and shifted window (SW) based multi-head self-attention modules, respectively. (Liu et al., 2021a)

There are an extensive amount of past works in the Video Transformer space. The Video Transformer Network (VTN) by (Neimark, 2021) was designed for video recognition, and uses a spatial backbone, temporal attention-based encoder, and a multilayer perceptron (MLP) head in order to achieve a high Top-1 and Top-5 accuracy. Furthermore, ViViT, a Video Vision Transformer, utilizes both spatial and temporal dimensions of the input to handle longer sequences of video and results in a high Top-1 and Top-5 accuracy score when compared to the Kinetics 400 and Kinetics 600 dataset (Arnab, 2021).

3. Methodology (Deb)

This section presents three fundamental architectures for video understanding: the Swin Transformer (Liu et al., 2021a), its spatio-temporal extension (Video Swin Transformer (Liu et al., 2021b)), and a more direct approach to 3-D Convolution networks (Tran et al., 2015; Carreira and Zisserman, 2017; Feichtenhofer et al., 2019). We begin by examining the Swin Transformer's (Liu et al., 2021a) innovative approach to vision modeling, which introduces hierarchical processing and shifted window mechanisms for efficient handling of high-resolution image data. We then explore how these concepts are extended into the spatio-temporal domain through the Video Swin Transformer (Liu et al., 2021b). Finally, we discuss 3D-Convolutional Networks(Tran et al., 2015; Carreira and Zisserman, 2017; Feichtenhofer et al., 2019), which represent a different yet equally important approach to video understanding through direct spatiotemporal feature learning. Each of these architectures offers unique advantages in processing video data, with distinct approaches to handling the temporal dimension and computational efficiency. The Swin Transformer's (Liu et al., 2021a) hierarchical design and shifted window-based attention provide a foundation for efficient vision modeling, while its video extension adapts these principles for temporal data. In contrast, 3-D Convolution networks (Tran et al., 2015; Carreira and Zisserman, 2017) take a more direct approach to spatio-temporal feature extraction through volumetric operations.

3.1. Swin Transformer

Overall Architecture: The overall architecture of the Swin Transformer (Liu et al., 2021a) is depicted in Figure 3. Unlike traditional Vision Transformers(Dosovitskiy, 2020) that dominate vision modeling, the Swin Transformer (Liu et al., 2021a) introduces hierarchical processing and shifted window mechanisms to efficiently handle high-resolution image data and varying scales of visual entities. A key feature of the Swin Transformer is its hierarchical design, which processes input images as sequences of patches, similar to tokens in language models. The patches are non-overlapping and of fixed size, typically 4×4, and are embedded into a lower-dimensional space. The hierarchical structure allows for the progressive merging of these patches as the network deepens (See Figure 3(a)). This results in feature maps of varying resolutions, enabling the model to effectively capture information at multiple scales. Such a design is crucial for dense prediction tasks like object detection and semantic segmentation, where fine-grained details play critical roles. Another key innovation of Swin Transformer lies in its shifted window-based self-attention mechanism. Instead of computing attention across the entire image like traditional transformers, the Swin Transformer computes self-attention within each window (i.e., the composition of multiple patches). In the subsequent stage, the window partition is shifted, creating connections between different windows from the previous layer. This shifted window approach serves two crucial purposes: it limits computation to local windows while enabling cross-window interactions, and it maintains linear computational complexity with respect to image size. We discuss the overall architecture of the Swin Transformer (Liu et al., 2021a), into three parts, e.g., shifted window-based attention, effective batch computation, and relative positional bias.

***(i)* Shifted Window-Based Self Attention:** The proposed Swin Transformer block (See Figure 3(a)) is divided into two attention modules. The first attention module is the window self-attention (W-MSA) (See Figure 3(b)). Unlike the traditional vision transformer, where the attention is computed across the entire image tokens to capture the global dependencies, the W-MSA limits the attention within each window. Specifically, each window computes its own key (K), query (Q), and value (V) individually. Although this W-MSA approach achieves linear computational complexity (Liu et al., 2021a), it limits information flow between windows since attention computation is isolated for each window. Hence, this severely hinders the extraction of long-range dependencies present within the image. To overcome this limitation, the following Transformer block employs Shifted Window-based Multihead Self-Attention (SW-MSA) (See Figure 3(b)). The windows are shifted by $([M/2], [M/2])$ pixels from the regular partitioning. This shifting creates new windows that bridge different regions from the previous layer, enabling cross-window connection while maintaining the efficiency of window-based attention computation. The overall self-attention mechanism is described mathematically as:

$$\hat{z}^l = \text{W-MSA}\left(\text{LN}\left(z^{l-1}\right)\right) + z^{l-1},$$
$$z^l = \text{MLP}\left(\text{LN}\left(\hat{z}^l\right)\right) + \hat{z}^l,$$
$$\hat{z}^{l+1} = \text{SW-MSA}\left(\text{LN}\left(z^l\right)\right) + z^l,$$
$$z^{l+1} = \text{MLP}\left(\text{LN}\left(\hat{z}^{l+1}\right)\right) + \hat{z}^{l+1},$$

where $\hat{z}^l$ and z^l denote the final output of the SW-MSA module and the MLP module for block ℓ, and LN is the layer normalization.

***(ii)* Effective Batch Computation:** The disadvantage of the shifting window is that it results in more windows compared to regular partitioning, and some of the windows will be smaller than the regular size. A simple but inefficient solution would be to pad the smaller windows to a fixed size and mask out the padded values during attention computation. However, when the number of regular windows is small (e.g., 2×2), this naive solution significantly increases computation. To solve this issue efficiently, the paper proposes a cyclic-shift approach. In this approach, the feature map is cyclically shifted toward the top-left direction before applying regular window partitioning. After window partitioning in this shifted configuration, a window might contain patches that were originally far apart in the feature map. Therefore, a masking mechanism is employed to prevent attention computation between these non-adjacent patches within each window. The key advantage of this cyclic-shift approach is that it maintains the same number of windows as regular partitioning. This makes the computation more efficient compared to the naive padding solution.

***(iii)* Relative Positional Bias:** To compute self-attention, Swin Transformer includes a relative position bias term $B \in \mathbb{R}^{M^2 \times M^2}$ to each attention head, following similar concepts from previous works (Dosovitskiy, 2020; Vaswani, 2017; Shaw et al., 2018). An important feature of the relative position bias is that it can be transferred between different window sizes. The learned relative position bias in pre-training can be used to initialize a model for fine-tuning with a different window size through bi-cubic interpolation. This makes the model more flexible and adaptable to different configurations. Furthermore, the relative position bias term introduces minimal computational overhead while significantly improving the model's ability to capture spatial relationships between patches within each window.

3.2. Video Swin Transformer

The Video Swin Transformer (Liu et al., 2021b) extends the Swin Transformer (Liu et al., 2021a), initially introduced for images, into the spatio-temporal domain. By leveraging hierarchical feature extraction and shifted windows, this architecture maintains computational efficiency and supports general-purpose video modeling tasks, such as action recognition and temporal modeling.

Overall Architecture: The Video Swin Transformer processes video inputs hierarchically, treating each video as a 3-D tensor with spatial and temporal dimensions. The input video, represented as $T \times H \times W \times 3$, is divided into non-overlapping 3-D patches. These

patches, with dimensions $P \times M \times M$, serve as tokens for further processing. The architecture consists of four hierarchical stages, each progressively down-sampling the spatial dimensions while maintaining the temporal resolution. This design mirrors the original Swin Transformer, enabling multi-scale feature representation. Each stage comprises video swin transformer blocks, which replace traditional self-attention modules with 3-D shifted window-based multi-head self-attention (MSA) modules. This mechanism computes attention within local 3-D windows while alternating between regular and shifted configurations. The shifted windowing strategy introduces connections between neighboring windows, enhancing the model's ability to capture spatio-temporal relationships.

3D Shifted Window: To address the high computational and memory demands of global self-attention, similar to Swin Transformer the Video Swin Transformer employs a 3-D extension of the Swin Transformer's window-based attention. By restricting attention computations to local 3-D windows, the model achieves significant efficiency gains. The shifted window mechanism ensures connectivity across adjacent windows, balancing computational cost and representational power.

3.3. 3-D Convolution

3-D convolution represents a fundamental extension of 2-D CNN operations for processing video data, where the temporal dimension is treated as a third axis alongside spatial dimensions. The operation processes a sequence of frames simultaneously to learn spatiotemporal features from video data. The pioneering work of C3D (Tran et al., 2015) introduced a deep 3-D CNN architecture with 11 layers, demonstrating that 3-D convolutions could effectively capture motion information in videos. This architecture processes video clips directly and learns spatio-temporal features through 3-D kernels that slide along both spatial and temporal dimensions. I3D (Inflated 3D ConvNet) (Carreira and Zisserman, 2017) presented a significant advancement by "inflating" 2-D CNN architectures pre-trained on ImageNet into 3-D CNNs. This approach enabled the model to leverage the strong spatial feature learning capabilities of pre-trained 2-D CNNs while extending them to handle temporal information. The inflation process expands 2-D filters into 3-D by repeating weights along the temporal dimension. Recent work has focused on making 3-D convolutions more efficient and effective. SlowFast Networks (Feichtenhofer et al., 2019) introduced a two-pathway architecture that processes video at different frame rates and temporal resolutions. X3D (Feichtenhofer, 2020) presented a family of efficient video networks through progressive expansion of 2-D architectures along multiple network dimensions.

4. Experiment (Ganesh, Sravanth)

In this section, we describe the design and execution of experiments to evaluate the effectiveness of the Video Swin Transformer and a 3D-CNN baseline for a medical image analysis task involving cardiac video sequences. We begin by detailing the dataset, which includes cardiac videos and their corresponding Time-of-Systole (TOS) curves as continuous ground truth labels. Preprocessing steps, such as data cleanup and upsampling for model compatibility, are outlined to ensure the data aligns with the input requirements of each model.

We then present the experimental setup, including the architecture design of the baseline 3D-CNN and the initialization strategy for the Video Swin Transformer. Training configurations, such as the optimizer, learning rate, and evaluation metrics, are specified to establish a framework for model comparison. Finally, we describe the evaluation methodology, which involves analyzing model performance on the test set using metrics like MSE, R2 scores, and parameter stability to provide an understanding of each model's strengths and limitations.

4.1. Dataset

The cardiac dataset is a list of video sequences (2D x T) that encode heart motions, using TOS curves as the continuous ground truth labels. We include its metadata and some derived statistics about the dataset in Tables 1 and 2.

Attribute	**Value**
Num. videos (raw)	128
Num. videos (processed†)	112
Video length	25 frames
Frame resolution	80x80 pixels
Num. channels	1

Table 1: Metrics and metadata for the cardiac videos.

Attribute	**Value**
Num. curves†	112
Min. value	17.0
Min/max peak	18.3, 118
Avg. peak	70.5
Mean variance	0.0979

Table 2: Metrics and metadata for the TOS curves. All metrics calculated before normalization.

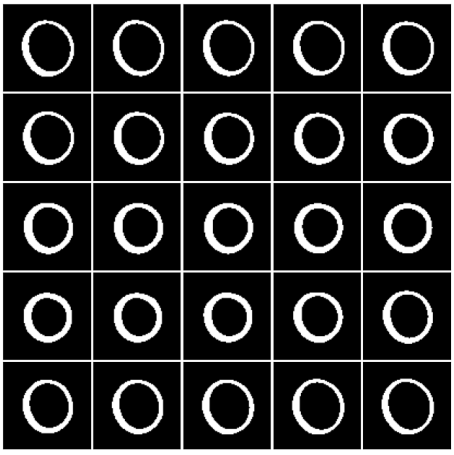

Figure 4: An example video sequence representing an encoding of heart motions.

We show an example video sequence in Fig. 4. From our observations, most videos display a ring-like shape with changing deformations over time. An interesting point to note is that while each video is no longer than 25 frames, each corresponding curve is 126 points. This indicates there is additional information we must extract from each frame to regress a video into a TOS curve.

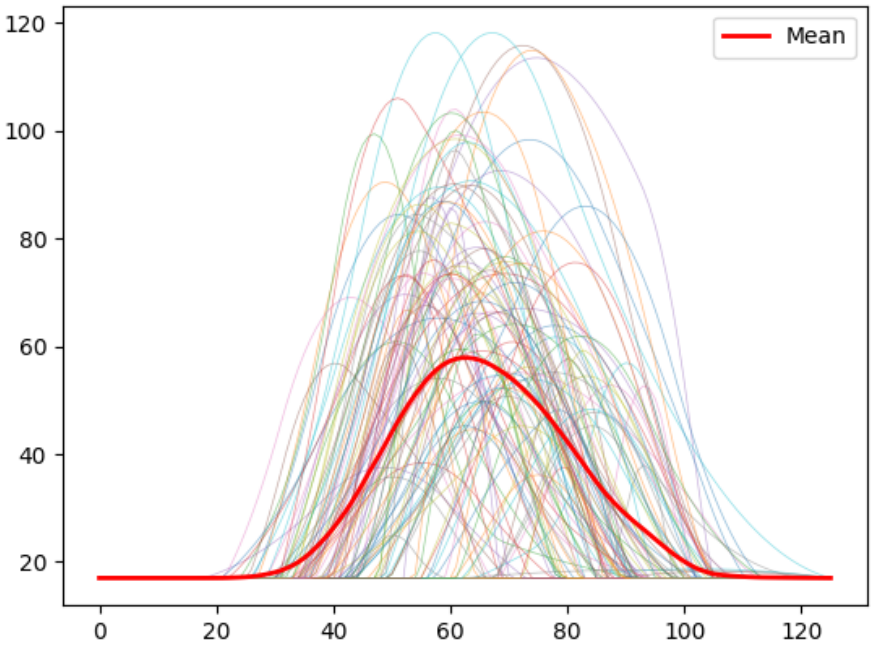

Figure 5: An overlay plot showing all TOS curves, after data cleaning. The mean is highlighted in red.

4.2. Preprocessing

†**Data cleanup:** A minor portion of our dataset has less than 25 frames. We drop these videos to keep the input dimension consistent for our experiment. Additionally, some TOS curves within the dataset are completely flat. We drop these curves as they would spike the training loss of our model and slow down the model's convergence on the rest of the data.

Video Upsampling: The Video Swin Transformer is implemented to process 224x224 images with 3 channels (Liu et al., 2021b). However, the cardiac video dataset has a much lower dimensionality, requiring us to up-sample it for model compatibility. To accomplish this, we repeat the channel dimension and interpolate each frame into a higher resolution. We use the bilinear interpolation strategy to ensure smooth and accurate up-sampling.

4.3. Experimental Setup

We implement a basic 3-D CNN model as a baseline for comparison with the Swin Transformer to evaluate its effectiveness in regression of the cardiac video dataset. We choose a CNN to emphasize the difference between convolutional layers and transformers in capturing features of the cardiac videos. Figure 5 shows the architecture of our 3-D CNN implementation.

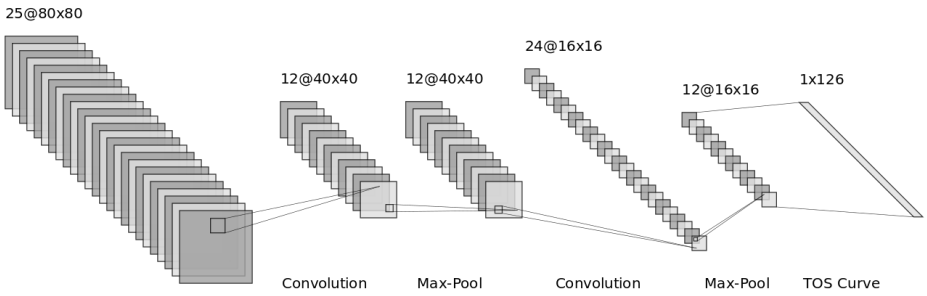

Figure 6: A simplified view of our 3D-CNN implementation, using the LeNet visualization scheme.

We initialize our Video Swin Transformer using the base Swin-3D size. We initialize the attention heads using the weights from its pretraining on the Kinetics-400 dataset, and we use Xavier initialization on its linear layers. We set the head of the Video Swin Transformer to a simple MLP to transform its outputs into a 126-dimension vector. We chose to run the experiments and report the results with only the Video variant and not the Vanilla Swin Transformer because of the obvious drawbacks of the original Swin Transformer in capturing temporal dependencies across frames of the video since it was initially designed only for use in Images.

Training: We split the dataset into an 80-20 train/test split. We use mean squared error (MSE) loss for both models to best match the objective of regression onto continuous labels. We train for up to 500 epochs but implement early stopping with a patience of 15 epochs. We use the Adam optimizer with an initial learning rate of 0.001 for both models. For the Video Swin Transformer, we implement a learning rate scheduler to stabilize training and smoothen the loss curve.

Evaluation: To compare the performance of each model, we calculate their MSE and R2 scores on the test set. We also measure the training time in epochs, and we record training loss curves for each model. To measure the stability of each model during training, we also report the L2 norm of their weights for each training step.

5. Results

Metric	VideoSwinTransformer	3D-CNN
Num. epochs	147	**54**
Final Train MSE	243.	**11.2**
Final Train R2	-0.137	**0.899**
Test MSE	**167.**	179.
Test R2	**0.220**	-0.0945
Final Weight Norm	143	79.3

Table 3: Training and evaluation metrics for both experimental models.

In Table 3, we observe almost 3x faster convergence in the 3D-CNN model. The CNN also achieves a significantly better training MSE and R2 score than the Video Swin Transformer. However, despite its better training performance, the 3D-CNN does not perform as well as the Video Swin Transformer during test set evaluation. The Video Swin Transformer provides a slight improvement from 3D-CNN's test set MSE and R2 score and even an improvement from its own metrics during training. The Video Swin Transformer also ends with a higher L2 norm of all model parameters, indicating greater model complexity.

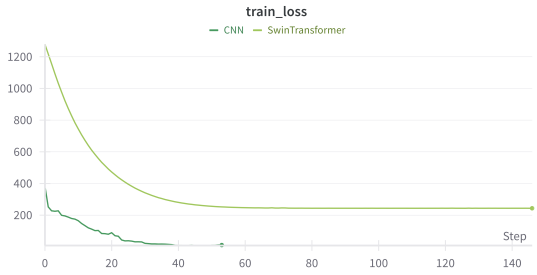

Figure 7: Training loss convergence

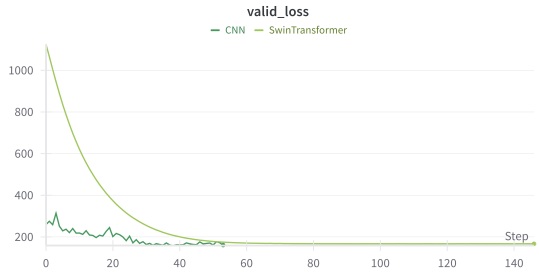

Figure 8: Validation loss convergence

Figure 9: Training and validation loss convergence over the epochs of CNN and Swin Transformer model training. In both graphs, the 3D-CNN curve ends first due to early stopping.

Figure 9 contains two graphs showing loss convergence over training. The loss curves of the 3D-CNN are markedly shorter, indicating a faster convergence over the data. However, the graphs also show the 3D-CNN had a significantly lower loss at the start of training, and the Video Swin Transformer actually had a steeper decrease in loss over the same number of steps. Figure 7 shows each model's training loss over time, which are generally smooth curves. Figure 8 shows each model's validation loss over time, which appears smooth for the Swin Transformer, but is more erratic for the 3D-CNN.

The L2 norms of all parameters of each model over time are shown in Figure 10. We can observe the Swin Transformer model weights decrease in magnitude over time, but the

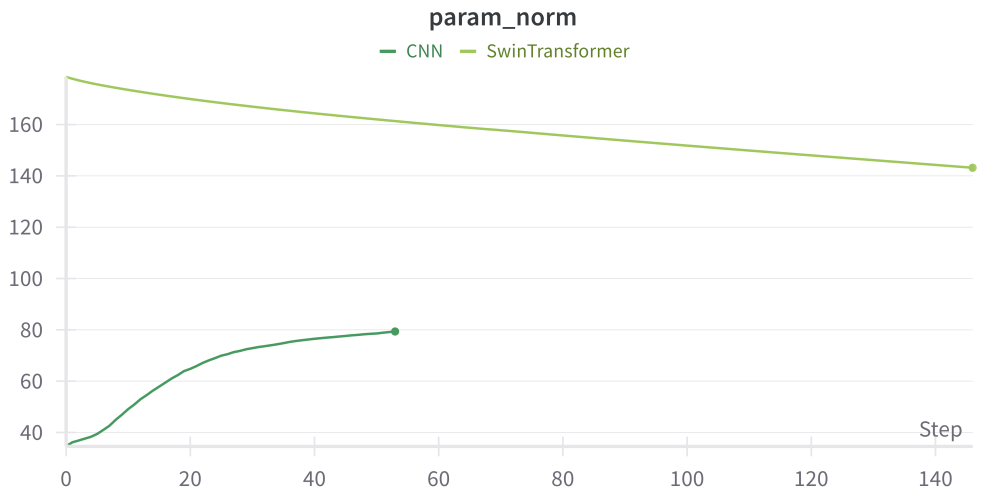

Figure 10: L2 norm of model parameters over each training step.

3D-CNN model weights increase over time. The 3D-CNN model weights L2 norm also shows a sharper increase at the beginning of training, then plateaus until the model reaches its early stopping point.

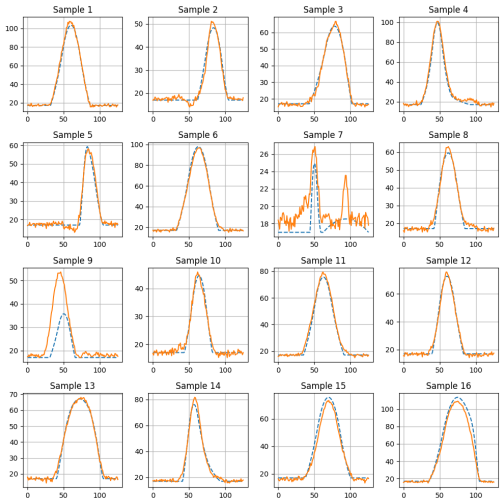

Figure 11: CNN train set predictions

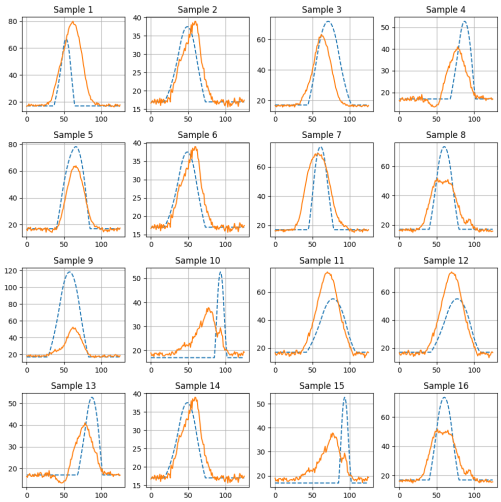

Figure 12: CNN test set predictions

Figure 13: Predicted (orange) vs ground truth (blue) TOS curves for the 3D-CNN model after training. The figure on the left shows 16 curves from the training set and the figure on the right shows 16 curves from the test set.

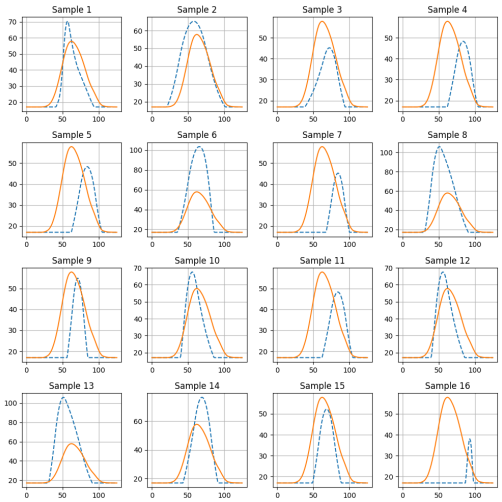

Figure 14: Swin train set predictions

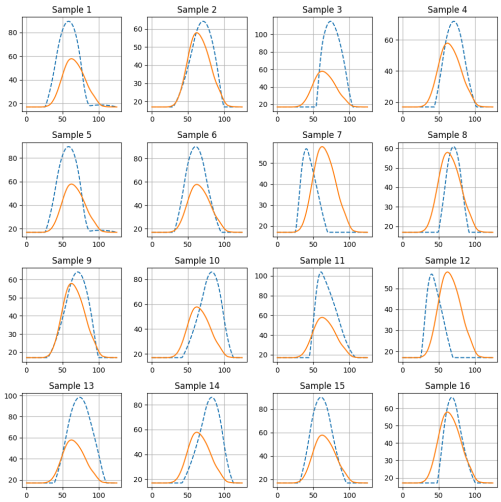

Figure 15: Swin test set predictions

Figure 16: Predicted (orange) vs ground truth (blue) TOS curves for the Video Swin Transformer model after training. The figure on the left shows 16 curves from the training set and the figure on the right shows 16 curves from the test set.

Figures 13 and 16 show examples of model predictions vs. the ground-truth TOS curves. We observe the best matches in Fig. 11, where the 3D-CNN model almost exactly matches each training label, corroborating the model's high R2 score on the training set. However, the 3D-CNN model's predictions are less accurate in Fig. 12, demonstrating a difficulty in generalizability. The Swin Transformer model, on the other hand, does not predict the training set labels nearly as well as the 3D-CNN model as shown in Fig. 14. However, Fig. 15 demonstrates the accuracy of training set predictions generalize equally to the test set labels.

6. Discussion (Ganesh, Sravanth)

Our results suggest that the Video Swin Transformer was not the best model to fit the data, evidenced by its high training error, low training R2 score, and high amount of epochs needed to converge to a relative minima in training loss when compared to the 3D-CNN. This may be explained by the Swin Transformer having a greater model complexity than the CNN: with a more complex architecture and a higher parameter count, the L2 norm of the Swin Transformer parameters was much higher than the norm of the 3D-CNN parameters. Furthermore, the norm of the Swin Transformer parameters decreased over time while the norm of 3D-CNN parameters increased, indicating the Swin Transformer may have been dropping certain weights as the model was too complex for the problem. This is supported by the fact that the cardiac video images are in a lower dimensional space than the larger, 3-channel images that Swin Transformer was designed to process.

In general, transformers require a large amount of training data to properly learn dataset features and create meaningful and precise representations of the data. We theorize that the relatively small cardiac video dataset did not provide enough variety and volume of data for the Swin Transformer to adequately learn the spatial and temporal features of the videos. For comparison, the Kinetics-400 dataset that Swin Transformer researchers trained it on has over 300,000 videos (Kay et al., 2017).

We observe greater stability in model outputs from the Swin Transformer when compared to the 3D-CNN, however. The Swin Transformer has a more consistent translation of results during training to testing, and even marginally outperforms the 3D-CNN during testing despite the CNN's much stronger training performance. We believe the simplicity of the CNN implementation in conjunction with the very small cardiac video dataset size led the model to overfit on the training data.

7. Conclusion (Sravanth, Ganesh)

This study explored the application of the Swin Transformer and its video variant in comparison with a traditional 3D-CNN for the task of medical image analysis using a cardiac dataset. The experimental results revealed that while the 3D-CNN outperformed the Swin Transformer in terms of training efficiency and accuracy on the training set, the Swin Transformer demonstrated better generalization on the test set. This highlights a tradeoff between simplicity and complexity in model architectures, particularly when dealing with small datasets. The Swin Transformer's hierarchical feature extraction and shifted window attention mechanisms, while advantageous for handling high-resolution and large-scale data, struggled to adapt to the low-dimensional and limited cardiac dataset. On the other hand, the 3D-CNN, with its simpler architecture, adapted better to the dataset but showed signs of overfitting due to the lack of sufficient data diversity.

The findings highlight the importance of aligning model architecture with dataset characteristics. Transformers, particularly the Swin Transformer, excel in capturing long-range dependencies and achieving robust generalization but require substantial data volume and variety to fully leverage their capabilities. The 3D-CNN, while effective for small datasets, may require additional regularization techniques or hybrid approaches to overcome its limitations in generalization. However, while augmenting data in a precision critical field such as medical image analysis, we must ensure that the augmented data does not bias the model in a dangerous manner which might have harmful real-life consequences.

7.1. Future work

Future work could focus on expanding the dataset through data augmentation or sourcing additional cardiac imaging data. Synthetic augmentation methods, such as temporal interpolation or generative adversarial networks (GANs), could provide the Swin Transformer with a better training environment, enabling better adaptation to medical imaging tasks. Additionally, pretraining the Swin Transformer on large-scale, domain-specific datasets could improve its initialization and enhance its performance on smaller datasets like the cardiac video dataset used in this study.

Exploring hybrid architectures that combine the strengths of both CNNs and transformers offers another promising direction. For example, integrating CNN-based feature

extraction with the Swin Transformer's global attention mechanisms could provide a balance between local detail capture and global contextual understanding. Moreover, fine-tuning pretrained models on smaller datasets could allow relevantly pre-trained transformers to adapt more effectively while leveraging the advantages of large-scale training. an example of the successful integration of CNN and Swin Transformers include Swin UNETR (Hatamizadeh et al., 2021) for semantic segmentation of brain images which uses the Swin Transformer as a backbone to achieve State Of The Art performance.

This study highlights the potential of transformer architectures like the Swin Transformer in medical image analysis while drawing attention to the challenges posed by low-data scenarios. Addressing these challenges through innovative approaches to data augmentation, pretraining and hybrid modeling could unlock the full potential of these models in the healthcare domain.

References

Heigold Sun Lucic Schmid Arnab, Dehghani. Vivit: A video vision transformer. *arXiv preprint arXiv: 2103.15691*, 2021.

Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017.

Raghavendra V. Kulkarni D. R. Sarvamangala. Convolutional neural networks in medical image understanding: a survey. *arXiv preprint arXiv: 2103.10504*, 2022. URL https://pmc.ncbi.nlm.nih.gov/articles/PMC7778711/pdf/12065_2020_Article_540.pdf.

Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Kolesnikov Weissenborn Zhai Unterthiner Dehghani Minderer Heigold Gelly Uszkoreit Houlsby Dosovitskiy, Beyer. Unetr: Transformers for 3d medical image segmentation. *arXiv preprint arXiv: 2010.11929*, 2021.

Syed Waqas Zamir Muhammad Haris Khan Munawar Hayat Fahad Shahbaz Khan Fahad Shamshad, Salman Khan and Huazhu Fu. Transformers in medical imaging: A survey. *arXiv preprint arXiv:2201.09873*, 2022.

Christoph Feichtenhofer. X3d: Expanding architectures for efficient video recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 203–213, 2020.

Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6202–6211, 2019.

Ali Hatamizadeh, Vishwesh Nath, Yucheng Tang, Dong Yang, Holger R Roth, and Daguang Xu. Swin unetr: Swin transformers for semantic segmentation of brain tumors in mri images. In *International MICCAI brainlesion workshop*, pages 272–284. Springer, 2021.

Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. The kinetics human action video dataset, 2017. URL https://arxiv.org/abs/1705.06950.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021a.

Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer, 2021b. URL https://arxiv.org/abs/2106.13230.

Zohar Asselmann Neimark, Bar. Video transformer network. *arXiv preprint arXiv: 2102.00719*, 2021.

Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. Self-attention with relative position representations. *arXiv preprint arXiv:1803.02155*, 2018.

Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497, 2015.

A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.