A Transformer-based video analysis framework for Estimating Ejection Fraction from Echocardiograms

Seyedeh Neda Ahmadi Amiri, Sana Ayromlou, Zahra Gholami , Mohammad Mahdi Kazemi Esfeh
Department of Electrical and Computer Engineering
University of Britich Columbia
571F project
nedaahmadi77@gmail.com, ayromlous@gmail.com, zahra.gh995@gmail.com
mohammadmahdikazemiesfeh@gmail.com

Abstract

Real-time estimation of ejection fraction and measuring left ventricle volume from echocardiography are crucial for some clinical applications in cardiology. Manual processing and expert labelling of ultrasound video frames suffer from both intra- and interobserver variability which increases the need for automating cardiac measurements. Also, the corruption of frames with noise and the sophisticated anatomical structure of the heart makes it difficult to efficiently train networks from scratch. We present a novel transformer-based architecture inspired by Timesformer, a video transformer that has achieved state-of-the-art performance on many action recognition datasets. We propose a convolution-free, fully attention-based Spatio-temporal architecture to predict the EF from Echo videos in an end-to-end approach. We build on top of this architecture for cardiac ultrasound video analysis and study its feasibility for echocardiography video regression tasks by adding knowledge distillation techniques to be able to train with fewer data. We apply self-attention using a divided space-time scheme which calculates attention over both spatial and temporal dimensions. We evaluate model's performance on learning temporal content. Our experiment shows this architecture significantly outperforms the previously proposed transformer-based network, the ultrasound transformer. Our end-to-end EF estimation approach can estimate the ejection fraction with an RMSE of 6.48. We also visualized the explainability of the proposed model by drawing attention maps and its T-SNE. Our code can be found on: https://github.com/MohammadMahdiKazemi/571MFinalProject

1 Introduction

The human cardiac cycle consists of two periods: diastole and systole. The heart ventricles relax and expand during diastole, filling the chambers with blood. Next, systole is characterized by the heart muscles contracting and pushing blood out of the ventricles, through arteries and veins, to the lungs and other organs. Ejection Fraction (EF) is the ratio of the blood pumped out of the ventricle (stroke volume) to the maximum amount of blood in the ventricle (end-diastolic volume). EF is a commonly used metric for determining functional cardiac health and is used for various clinical evaluations and diagnoses. By accurately measuring EF in an accessible manner, clinicians have easy access to critical information that can help diagnose and treat cardiac patients. A normal EF is typically within the range $EF = 65\% \pm 10\%$ [7].

In recent years, artificial intelligence (AI) technology has become a research hot-spot in cardiovascular imaging, diagnosis, and treatment of heart diseases. Many deep convolutional neural networks were previously applied to cardiac ultrasound videos for measuring ejection fraction by either segmenting the left ventricle or minimizing a regression loss [14, 13, 6, 7].

The features relevant to the estimation of the EF are embedded in End-Systolic (Es) and End-Diastolic (ED) frames, which makes learning long-term temporal dependencies crucial for the estimation of EF. Recent works show transformers significantly improve long-term temporal feature extraction and have recently been used for many Language processing and vision tasks[3, 19, 12, 2, 8].

In this work, we leverage transformers' superior performance for the task of EF estimation from echocardiograms by proposing a novel architecture based on TimeSformer architecture [3], a video transformer that has achieved the state-of-the-art performance on many action recognition datasets. Action recognition tasks in computer vision are closely related to EF estimation as capturing long-term Spatio-temporal dependencies is their crucial ingredient.

Our contribution can be summarized as (a) proposing a convolution-free, fully attention-based, Spatio-temporal architecture to predict the EF from Echo videos in an end-to-end approach. We also validate this network on EchoNet-Dynamic (Echo-Net)[9] dataset. (b) We leverage information captured by deep convolutional networks that has been shown to perform well on this dataset. We specifically deploy the state-of-the-art knowledge distillation techniques for regression tasks [15, 16]. (c) We show the explainablity of our proposed model by visualizing its attention maps in both spatial and temporal dimensions.

2 Related Work

Methods for fully automatic volume and EF measurements Fast, accurate and explainable echo measurements as the main point-of-care imaging modality are crucial in clinical workflows. Early methods extensively used different deep convolutional neural networks (DCNNs) for various tasks. As left ventricular ejection fraction (LVEF) is usually calculated from the AP4 view of the heart, some applications have used CNNs [10] for view classification before assessing further tasks. In [14], the authors segmented the left ventricle from US videos using UNET architecture to predict the ES and ED frames. Also, a few works have calculated LVEF without segmentation using Conv3D [13], ResNet(2+1)D [17] and CNN+LSTM architectures [6] by directly minimizing a regression loss. Other segmentation techniques were also used, including [21] which used co-learning from appearance and shape to increase both temporal and spatial accuracy. Previous works that have used RNN and LSTM-based architectures to extract temporal features suffer from forgetting initial elements, and most of them only accept a fixed number of frames as input [20]. Transformers show great success on video data, and they overcome the mentioned flaws. To the best of our knowledge [11] is the first and only work that uses a transformer-based architecture to predict ejection fraction. They first apply a Residual Auto-Encoder Network to input video to reduce its dimensionality. Then, a BERT model is adapted for token classification, which provides a reasoning ability in the Spatio-temporal domain. This work estimates LVEF by straightly using a regression network rather than segmenting the left ventricle.

Vision Transformers Vision Transformers were originally proposed for image data. One proposed transformer-based network architecture explicitly designed for medical image segmentation tasks is Gated Axial-Attention for Medical Image Segmentation [19]. This work proposes a Medical-Transformer (MedT) built upon a gated position-sensitive axial attention mechanism with a control mechanism added to the self-attention module, which adapts a Local-Global training strategy (LoGo). This work was mostly used for CT and MRI images but not videos. Video Transformers (VTs) [12] mainly are derived from previous transformer designs, especially the ones applied to image domains. However, the inherent structure of videos causes them to have large dimensionality, which exacerbates Transformer limitations. Still, at the same time, it increases the ability to define embeddings, tokenization strategies, and architectures. [2] is a new vision transformer also known as Vivit. They proposed different variations of their transformer, which are different in factorizing spatial and temporal dimensions. This model first applies a linear operation on frames of the input video and rasterizes them into a 1D Spatio-temporal token. A series of transformer layers then encode generated tokens. They propose three different factorizing methods to increase efficiency and scalability, including Spatio-temporal attention, Factorised encoder, and Factorised self-attention. A lot of novel video transformers were used for action detection, such as [8]. The swin transformer is one of the benchmarks that have comparable results to the state-of-the-art methods for multiple datasets. In this work, we want to make use of TimeSformer. In [3] they argue that replacing convolutional layers in transformer with self-attention the network has the potential of overcoming a

few inherent limitations observed in CNN models and gives the architecture the ability to capture both local and global dependencies by operating on spatial and temporal domain. Our work plans to use the suggested platform and propose a novel convolution-free transformer-based architecture for cardiac video analysis. Also, as it finds the relation among different patches, we can visualize its explainability by drawing attention maps which is essential for medical usage.

3 Method

In this work, we propose a novel video transformer architecture based on [3] (see Figure 1), which to the best of our knowledge, is the current state-of-the-art for Spatio-temporal video classification. We construct our model by introducing architectural changes to adapt it for the task of predicting EF from echo cines.

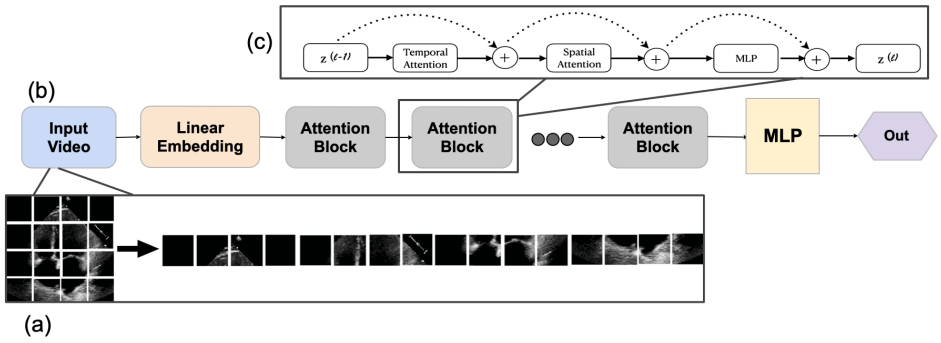


Figure 1: Architecture of proposed model:(a)Decomposing input video into patches, (b) Overall network architecture, (c)Architecture of each attention block

3.1 Input video

The input to our model will be a sequence of video frames $X \in \Re^{H,W,3,F}$ sampled from an echo cine where $H,\ W, 3,\ F$ represent the height, width, RGB channels and number of sampled frames, respectively. Each frame is then decomposed into P partitions of size $N \times N$ (non-overlapping patches that cover the entire frame) i.e. $PN^2 = HW$. In each frame, patches are stacked to form a vector of size $3P^2$. Finally, the input to the timeSformer is defined as $x_0^{p,f} \in \Re^h$ with $p = 1, 2, \cdots, P$ where p specifies each positional embedding and f defines the frame number. A sample of input transformation has been shown in Figure 1,a.

3.2 TimeSformer Model

The Transformer consists of linear embedding and L encoding blocks. Each block is divided into five sections which have been described below. The overall architecture has been shown in Figure 1,c.

Linear Embedding Each feature vector is linearly mapped to an embedding vector $l_0^{p,f} \in \Re^h$ with $p = 1, 2, \cdots, P$ and $f = 1, 2, \cdots, F$. Here, the number of dimensions of the embedded space is denoted by h. Formally we can write

$$l_0^{p,f} = Ex^{p,f} + e^{p,f}, \tag{1}$$

where E is the embedding matrix with the appropriate dimensions and e denotes the corresponding positional embedding similar to the same operation for text positional embedding. As in [5], in the first position of the embedded sequence, we enter a learnable vector that represents the token of classification $l_{0,0}^0 \in \Re^h$. $x^{p,f}$ also represents $p^{\texttt{th}}$ patch from input X corresponding to the $f^{\texttt{th}}$ frame.

Query-Key-Value computation For each attention head input of each attention, the block is fed to a Layer Norm which creates and embeds for query, key, and value for each patch. The input to each

block is the representation $l_{p,f}^{(l-1)}$ encoded by the preceding block.

$$q_{p,f}^{l,a} = W_Q^{l,a} LinearNorm(z_{p,f}^{l-1}) \in \Re^h, \tag{2}$$

$$k_{p,f}^{l,a} = W_K^{l,a} LinearNorm(z_{p,f}^{l-1}) \in \Re^h, \tag{3}$$

$$v_{p,f}^{l,a} = W_V^{l,a} LinearNorm(z_{p,f}^{l-1}) \in \Re^h, \tag{4}$$

Where a denotes the index of attention head.

Self-attention computation In this work, attention is computed over two dimensions (spatial and temporal). For divided space-time attention, NF + 1 query-key comparisons are made. The self-attention weights are computed using a dot product between query patch (p, f) with keys of other patches in the same frame and keys of the same spatial position in other frames. Self-attention weights $a_{p,f}^{(l,a)} \in \Re^{NF+1}$ are calculated as below:

$$a_{p,f}^{(l,a)} = softmax(q_{p,f}^{(l,a)} / \sqrt{h}.(k_{0,0}^{(l,a)}[k_{p=1,\cdots,P,f=1,\cdots,F}^{(p',f')}])) \tag{5}$$

Encoding Having attention weights, weighted sum of value vectors is computed using self-attention coefficients from each attention head. Then, the vectors from all heads are concatenation and passed through an MLP, using residual connections after each operation. Together, these two layers create the encoding $l_{p,f}^{(l)}$ which will be the input to the next block.

$$s_{p,f}^{(l,a)} = a_{p,f,(0,0)}^{(l,a)} v_{0,0}^{(l,a)} + \sum_{p'=1}^{N} \sum_{f'=1}^{F} a_{p,f,(p',f')}^{(l,a)} v_{p',f'}^{(l,a)} \tag{6}$$

$$l_{p,f}^{\prime(l)} = W_0[s_{p,f}^{(l,1)} \cdots s_{p,f}^{(l,A)}].T + l_{p,f}^{(l-1)} \tag{7}$$

$$l_{p,f}^{(l)} = MLP(LinearNorm(l_{p,f}^{\prime(l)})) + l_{p,f}^{\prime(l)} \tag{8}$$

Regression Embedding The initial token of encoding computed from the final block is fed into a linear layer with a fixed bias of 55 to calculate a single value which is the predicted ejection fraction.

$$EF = LinearNorm(l_{0,0}^{L}) \tag{9}$$

Divided Space-Time Attention In this work, computing attention uses the "Divided Space-Time Attention" architecture proposed in [3], where temporal attention and spatial attention are separately applied one after the other. This architecture saves a lot of computation compared to Joint Space-Time attention and performs better than calculating space attention alone. For Divided Attention, within each block, first temporal attention is computed by comparing each patch with all the patches at the same spatial location in the other frames:

$$a_{p,f}^{(l,a)} = softmax(q_{p,f}^{(l,a)} / \sqrt{h}.(k_{0,0}^{(l,a)}[k_{f=1,\cdots,F}^{(p,f')}])) \tag{10}$$

Then temporal attention is fed to spatial attention computation. We will use divided Spatio-temporal attention since it is algorithmically aligned with how humans compute EF. Then, we feed the output of the self-attention units to a Multi-Layer Perceptron to get the output embedding. Therefore distinct query, key, and value matrices should be learned for Time and Space dimensions. Overall, per each patch, $NF + 1$ comparisons should be calculated. Figure 2 gives a representation of this mechanism.

In this work, we predict the volume of the left ventricle at ES and ED frames along with the EF as a percentage. We train the model using Mean squared loss for the EF and volume estimation.

Knowledge Distillation High-performance vision transformers are pre-trained with millions of images. TimeSformer has shown to perform well when pre-trained on the ImageNet dataset [4]. The Echo-Net dataset contains about 10000 videos which are not enough to train the model from the scratch without over-fitting. Knowledge distillation is a training strategy to recover accuracy drop that uses a model that performs well on a dataset as a teacher network and transfers its knowledge to a student model that requires a big amount of data to be appropriately trained. This helps the student model to mimic soft labels coming from a strong teacher network instead of being trained on hard labels.

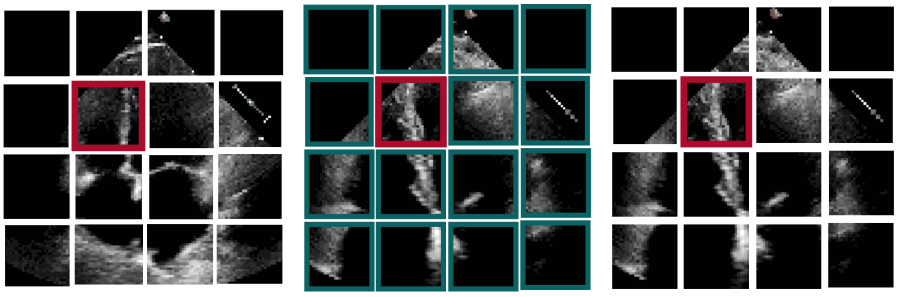


Figure 2: Visualization of the divided space-time self-attention schemes, Each video clip is viewed as a sequence of frame-level patches with a size of 16 × 16 pixels. Temporal attention is illustrated as comparison of red patches and spatial attention is shown as comparison of green patches. Note that self-attention is computed for every single patch in the video clip. We also note that although the attention pattern is shown for only two adjacent frames, it extends in the same fashion to all frames of the clip

Distillation through attention Due to the different structure of timeSformer compared to convolutional networks, knowledge distillation should be defined differently. This is done by adding a new token [16], the distillation token, to the patches and class token. The distillation token goes through self-supervised attention, similar to other tokens. In the final layer, the distillation embeddings learn from the last layer embedding of the teacher network while remaining complementary to the class embedding. Figure 3 However, knowledge distillation is usually adapted in classification problems since it has the advantage of "dark knowledge" and logits outputs in teacher can provide more information for student model compared to the one-hot encoding of the class label. For regression, on the other hand, distillation does not pass any distribution over classes by the teacher to aid learning. The network predicts a continuous value with the same characteristics as the ground truth, with an unknown error distribution. Therefore, we used a regression error as our distillation error to help the distillation token help adjust model weights through backpropagation.

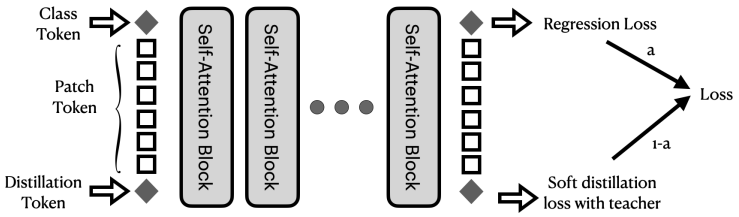


Figure 3: Knowledge distillation process

Distilling Knowledge From Regression Networks Adding a distillation token changes the network structure. Therefore the pre-trained weights on the Imagenet cannot be used, which has a considerable impact on final accuracy that the knowledge distilled to the network from the teacher cannot compensate. Without access to any dark knowledge, we used another method to blend the loss of student network prediction and the teacher's prediction. We define imitation loss as the MSE loss between teacher and student network's outputs, and student loss as the MSE loss of student network outputs with respect to the ground truth. Similar to work proposed by [15] an attentive imitation loss can be used to model the uncertainty in the imitation loss with a weight based on how reliable the teacher prediction is. To model a parametric distribution of teacher's reliability, the teacher loss can be used as below:

$$loss = \alpha MSE(o_T, o_{gt}) + (1 - \alpha) * \phi * MSE(o_T, o_{gt}) \tag{11}$$

$$\phi_i = (1 - \frac{MSE(o_T, o_{gt})_i}{\eta}) \tag{12}$$

$$\eta = max(e_T) - min(e_T) \tag{13}$$

$$e_T = \{||o_T - o_{gt}||_j^2 : j = 1, ..., N\} \tag{14}$$

where ϕ_i is the scale the student network learns from the teacher for each training batch, e_T is a set of teacher loss from entire training data, and η is a normalization parameter that we can retrieve from subtracting the maximum and the minimum of e_T . Here, i is a representation of each batch index (i = 1, ..., n) and j represents the index of each training sample (j = 1, ..., N).

4 Experiments

We adopt the timesformer architecture pretrained on the ImageNet dataset for each experiment. Clips are used with a size of 3 × 32 × 112 × 112, with frames sampled with a period of 2 from a random start-point to make sure we have at least one heart cycle in the input video. The patch size is 16 × 16 pixels(49 (7 × 7) tokens represent the whole frame) and 12 attention blocks are used in the structure.

Comparison to CNNs We chose transformers due to their higher explainability compared to CNNs and their ability to apply temporal and spatial attention. We performed an empirical study and demonstrated Table 3 the preliminary results of ejection fraction prediction for different architectures by reporting the root mean squared error. The results show that our proposed architecture outperforms the previously proposed ultrasound transformer by a large margin. However, convolutional architectures generally have better performance in comparison to both transformers. The reason behind this is that the size of our dataset is relatively small to train transformers, and it can't find the complex relations among all patches. However, the inductive bias in convolutional neural networks helps them to be able to train with a lower amount of data. our architecture has 121.4 M parameters, which leads to a large learning capacity. However, in contrast to CNN architectures, this model does not use any convolutional layers, which decreases its inference cost (0.59 TFLOPs) and increases its capacity while maintaining its efficiency.

	Architectures	RMSE Loss	Params
CNN	Resnet (2+1)D-18	5.8	33.3M
	Tiny VideoNet	6.28	11.2M
Transformer	Ultrasound Transformer	8.38	346.8M
	TimeSformer(ours)	6.48	121.4M

Table 1: Video-level ejection fraction MSE loss on different video architectures

Varying the Number of Tokens The structure of TimeSformer allows the model to operate on any number of video frames and any spatial resolution with a size of 16k *16k, where k is an integer. Studies show that increasing spatial resolution and the number of frames can help the overall accuracy of the model. However, higher resolution and more frames both result in a higher number of patches, increasing the number of tokens. This can make the computational cost very expensive.

4.1 Knowledge distillation

We experimented with both explained knowledge distillation techniques and compared their impact on the final performance of the network. As stated before, the problem with adding a distillation token is that we cannot use pre-trained weights, and therefore, we cannot see a good overall RMSE loss for it. The second method helps the network throughout the training, and the RMSE loss improves.

Knowledge Distillation Method	RMSE Loss
Distillation through Attention	7.65
Distillation through Regression	6.34

Table 2: Using Knowledge distillation to help transformer learn from state of the art network

4.2 Multi head output

In addition to EF, we trained the model in a multi-output regression heads framework to predict the LV volume at ES and ED frames. To prevent the outputs from bouncing around and making the optimization unstable, we preset the bias weights of each output head to a fixed value equal to the

mean of the specific label computed from the training set and don't apply gradient updates to these weights. In other words, the bias weights of the last fully connected layer are set to constant values that don't change during the training. This way, we managed to achieve the best RMSE reported in Table 3.

4.3 Additional Ablations

To find the right size for timeSformer, we performed an ablation study. We investigated the different sizes of depth, path, and various optimizers and used the best hyperparameters for the final model. **Smaller and Larger Transformers** We experimented the timeSformer with a different number of blocks. As the size of the input and output embedding at each block does not change throughout the model, adding or reducing blocks has no impact on the weights and structure of the rest of the model. However, reducing the number of blocks resulted in a higher loss.

Larger Patch Size We also experimented with different patch sizes, P = 32. As choosing a larger patch size reduces spatial granularity, this variant of our model also produced worse RMSE loss than the default variant with a patch size of 16. Another reason behind this is that we were unable to use pre-trained model weights trained on the Imagenet dataset in this variation, which can significantly impact the final result. We did not train any models with P values lower than 16 as those models have a much higher computational cost.

The Order of Space and Time Self-Attention Our proposed "Divided Space-Time Attention" scheme applies temporal attention and spatial attention one after the other. In our preliminary results, we saw that using time attention before spatial attention resulted in a better performance compared to the other way around.

Depth	Patch size	Loss Function	Optimizer	training RMSE	Validation RMSE
12	16	MSE	Adam	3.18	8.12
12	28	MSE	Adam	4.28	8.41
8	16	MSE	Adam	4.81	8.72
12	**16**	**MSE**	**SGD**	**2.02**	**6.48**
12	28	MSE	SGD	2.23	7.93
8	16	MSE	SGD	2.91	8.11

Table 3: Video-level ejection fraction MSE loss on different video architectures

4.4 Explainability and Visualization

Visualizing Learned Space-Time Attention In order to visualize the learned attention, the Attention Rollout scheme was used [1]. Attention maps in transformers give a good visualization of what individual activations look like, but they don't show us how attention flows within different layers. At the end of each block, an attention matrix(A) will be obtained. Element i,j in this matrix defines how much attention is going to flow from token j in the previous layer to token i in the next layer, and multiplying 2 attentions from subsequent layers will define the flow. We took the minimum among attention heads, discarded low attention values, and calculated the attention rollout as below:

$$AttentionRollout_L = (A_L + I)AttentionRollout_{L-1} \tag{15}$$

This value is normalized after each layer and at the final layer the classification token is discarded and

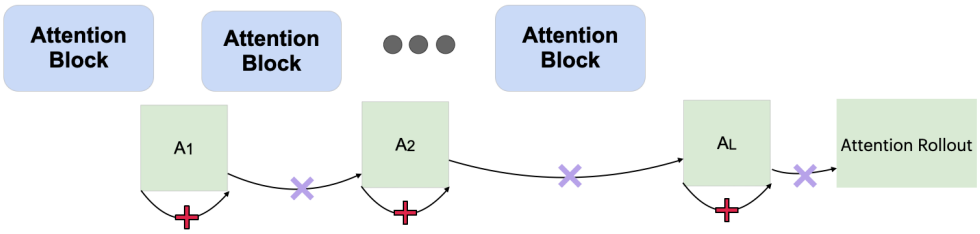
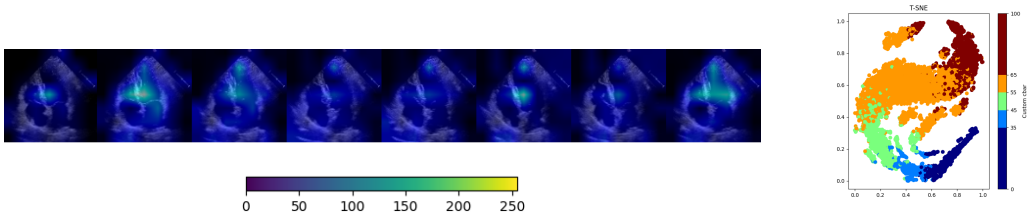


Figure 4: Attention Rollout computation



(a) (b)

Figure 5: Visualizations: (a)EF values have been divided into bounds based on clinical purposes and the figure shows how output embeddings are able to learn separable features for different EF values. (b) Visualization of attention maps from the output token to the input space on Echo-Net dataset. Our model learns to focus on the relevant parts in the video to perform spatiotemporal reasoning

49 remaining tokens for each frame are reshaped to a 7*7 image. Figure 4 shows the resized attention map fed onto the initial frame. Our results suggest that the model focuses on relevant regions in the video and also the attention value increases in frames with more information. This shows that the model is learning both temporal and spatial reasoning.

Visualizing Learned Feature Embeddings

This visualization shows how feature embeddings vary for different EF values. However, because showing a 768 dimensional embedding is not possible, first we decrease the embedding size to 2 using t-SNE algorithm. In the visualization, each point represents a single video, and different colors depict different bounds of EF based on medical definitions. Based on this illustration, we observe that TimeSformer with divided space-time attention learns semantically separable features Figure 5,b.

EchoNet-Dynamic Dataset The EchoNet-Dynamic dataset [9] was created by Standford University. It is a large echocardiography dataset for studying cardiac motions and changes in Left ventrical volume and shape in cardiac cycles. It consists of 10,036 videos of apical four-chamber (A4C) view for patients who had echocardiography between 2016 and 2018 at Stanford Health Care. Each video is labeled with the corresponding left ventricle border tracing, EF, ED and ES frame indexes and volume of the left ventricle at end-systole and end-diastole by expert sonographers. For each video, two frames (ES and ED) are annotated with manual segmentation.

Computational Resources We used four NVIDIA Tesla V100 GPUs with 32GB of memory from UBC ARC Sockeye [18] for training and evaluation.

5 Conclusions

In this work, we proposed a fully attention-based transformer to predict ejection fraction and validated it on the Echonet-Dynamic dataset. We built our model based on timesformer, which applies separate spatial and temporal attentions sequentially. This would lead to a less complex network with fewer parameters, given that bias of attention should be only in one dimension, training the network is more manageable with fewer data. The results have shown that although we outperform the state-of-the-art transformer-based models, convolutional neural networks perform better. We decided to leverage information captured by SOTA convolutional neural network to our proposed timesformer using knowledge distillation. We applied two different approaches: distillation through attention and distilling knowledge from a regression network. Finally, to show the explainability of our proposed network, we visualized attention maps and did an ablation study over different hyper-parameters. Although there is still a gap between our results and CNN-based SOTA methods, improvements of the proposed network compared to SOTA transformers-based networks and its high explainability potential proves that using transformers can be a good approach for learning medical imaging task. As a future work, we suggest trying different distillation techniques and optimization regularizations specifically proposed to train transformers.

References

[1] Samira Abnar and Willem Zuidema. *Quantifying Attention Flow in Transformers*. 2020. DOI: 10.48550/ARXIV.2005.00928. URL: https://arxiv.org/abs/2005.00928.

[2] Anurag Arnab et al. *ViViT: A Video Vision Transformer*. 2021. arXiv: 2103.15691 [cs.CV].

[3] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. "Is space-time attention all you need for video understanding". In: *arXiv preprint arXiv:2102.05095* 2.3 (2021), p. 4.

[4] Jia Deng et al. "Imagenet: A large-scale hierarchical image database". In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee. 2009, pp. 248–255.

[5] Jacob Devlin et al. "Bert: Pre-training of deep bidirectional transformers for language understanding". In: *arXiv preprint arXiv:1810.04805* (2018).

[6] Adrian Meidell Fiorito et al. "Detection of Cardiac Events in Echocardiography Using 3D Convolutional Recurrent Neural Networks". In: *2018 IEEE International Ultrasonics Symposium (IUS)*. 2018, pp. 1–4. DOI: 10.1109/ULTSYM.2018.8580137.

[7] Mohammad Mahdi Kazemi Esfeh et al. "A Deep Bayesian Video Analysis Framework: Towards a More Robust Estimation of Ejection Fraction". In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2020*. Ed. by Anne L. Martel et al. Cham: Springer International Publishing, 2020, pp. 582–590. ISBN: 978-3-030-59713-9.

[8] Ze Liu et al. "Swin Transformer: Hierarchical Vision Transformer using Shifted Windows". In: *arXiv preprint arXiv:2103.14030* (2021).

[9] David Ouyang et al. "EchoNet-Dynamic: a Large New Cardiac Motion Video Data Resource for Medical Machine Learning". In: 2019.

[10] Jin Park et al. "Automatic Cardiac View Classification of Echocardiogram". In: vol. 0. Nov. 2007, pp. 1–8. ISBN: 978-1-4244-1631-8. DOI: 10.1109/ICCV.2007.4408867.

[11] Hadrien Reynaud et al. "Ultrasound Video Transformers for Cardiac Ejection Fraction Estimation". In: *CoRR* abs/2107.00977 (2021). arXiv: 2107.00977. URL: https://arxiv.org/abs/2107.00977.

[12] Javier Selva et al. "Video Transformers: A Survey". In: *CoRR* abs/2201.05991 (2022). arXiv: 2201.05991. URL: https://arxiv.org/abs/2201.05991.

[13] Ahmad Shalbaf et al. "Automatic detection of end systole and end diastole within a sequence of 2-D echocardiographic images using modified Isomap algorithm". In: *2011 1st Middle East Conference on Biomedical Engineering*. 2011, pp. 217–220. DOI: 10.1109/MECBME.2011.5752104.

[14] Erik Smistad et al. "Fully Automatic Real-Time Ejection Fraction and MAPSE Measurements in 2D Echocardiography Using Deep Neural Networks". In: *2018 IEEE International Ultrasonics Symposium (IUS)*. 2018, pp. 1–4. DOI: 10.1109/ULTSYM.2018.8579886.

[15] "Distilling knowledge from a deep pose regressor network". English. In: *Proceedings - 2019 International Conference on Computer Vision, ICCV 2019*. Ed. by In So Kweon et al. Proceedings of the IEEE International Conference on Computer Vision. United States of America: IEEE, Institute of Electrical and Electronics Engineers, 2019, pp. 263–272. ISBN: 9781728148045. DOI: 10.1109/ICCV.2019.00035. URL: http://iccv2019.thecvf.com/,%20https://ieeexplore.ieee.org/xpl/conhome/8972782/proceeding.

[16] Hugo Touvron et al. *Training data-efficient image transformers amp; distillation through attention*. 2020. DOI: 10.48550/ARXIV.2012.12877. URL: https://arxiv.org/abs/2012.12877.

[17] Du Tran et al. "A closer look at spatiotemporal convolutions for action recognition". In: *Proceedings of the IEEE CVPR*. 2018, pp. 6450–6459.

[18] UBC Advanced Research Computing. *UBC ARC Sockeye*. en. 2019. DOI: 10.14288/SOCKEYE. URL: https://arc.ubc.ca/ubc-arc-sockeye.

[19] Jeya Maria Jose Valanarasu et al. "Medical Transformer: Gated Axial-Attention for Medical Image Segmentation". In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2021*. Cham: Springer International Publishing, 2021, pp. 36–46. ISBN: 978-3-030-87193-2.

[20] Ashish Vaswani et al. "Attention is all you need". In: *Advances in neural information processing systems* 30 (2017).

[21] Hongrong Wei et al. “Temporal-Consistent Segmentation of Echocardiography with Co-learning from Appearance and Shape”. In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2020*. Ed. by Anne L. Martel et al. Cham: Springer International Publishing, 2020, pp. 623–632.

A Appendix