# Random Token Fusion for Multi-View Medical Diagnosis

**Jingyu Guo** [1,2] * **Christos Matsoukas** [1,2] **Fredrik Strand** [3,4] **Kevin Smith** [1,2]

[1] KTH Royal Institute of Technology, Stockholm, Sweden
[2] Science for Life Laboratory, Stockholm, Sweden
[3] Karolinska Institutet, Stockholm, Sweden
[4] Karolinska University Hospital, Stockholm, Sweden

## Abstract

In multi-view medical diagnosis, deep learning-based models often fuse information from different imaging perspectives to improve diagnostic performance. However, existing approaches are prone to overfitting and rely heavily on view-specific features, which can lead to trivial solutions. In this work, we introduce Random Token Fusion (RTF), a novel technique designed to enhance multi-view medical image analysis using vision transformers. By integrating randomness into the feature fusion process during training, RTF addresses the issue of overfitting and enhances the robustness and accuracy of diagnostic models without incurring any additional cost at inference. We validate our approach on standard mammography and chest X-ray benchmark datasets. Through extensive experiments, we demonstrate that RTF consistently improves the performance of existing fusion methods, paving the way for a new generation of multi-view medical foundation models.

## 1 Introduction

Physicians routinely employ multi-view analysis in diagnostic procedures. Images gathered at various angles can unveil details that may be obscured in a single view, enhancing the precision of the diagnosis. It has been shown in clinical trials that a large percentage of breast cancers can only be detected when both craniocaudal (CC) and mediolateral oblique (MLO) views are analyzed [13]. Similarly, the frontal and lateral views of chest X-rays can provide unique information that is valuable to the accurate diagnosis of diseases [11, 15, 31, 18]. This is because complementary information from different views helps physicians tackle the challenges posed by superimposed tissues and complex anatomy. Each additional view provides context and offers unique insights, enriching the overall understanding of a patient's condition [18].

Given the clear benefits of multi-view analysis in clinical practice, it stands to reason that foundation models for medical image analysis could similarly improve their diagnostic accuracy by integrating information from multiple views. Information from different views of anatomical structures may reduce ambiguity, better explain spatial relationships, and provide additional context. Although only a handful of works have addressed the topic of multi-view fusion in neural networks, evidence suggests that there is a tangible benefit from fusing information from multiple views [4, 1, 42, 48, 38].

The question of how and where to mix information from different views is an open area of research, with multiple strategies having been proposed in recent years. This is not our main focus. Rather, we propose a solution *that can enhance existing multi-view fusion strategies*, which we call Random

*Corresponding author <jingyug@kth.se>

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

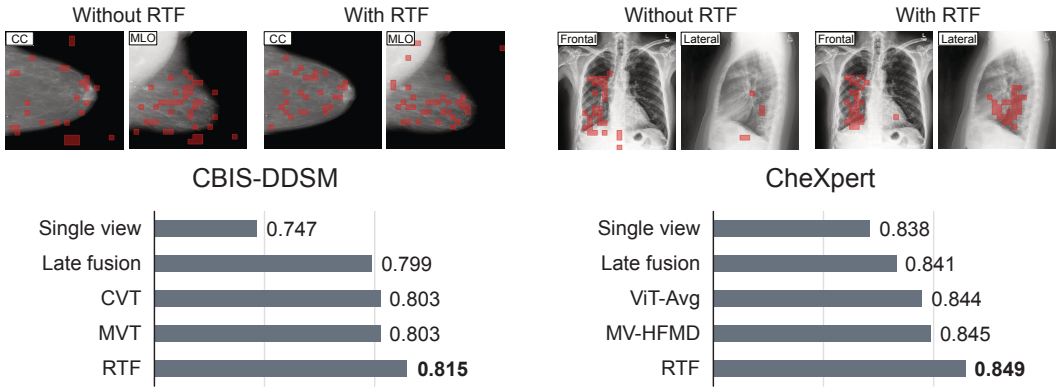


Figure 1: *Illustration of the overfitting problem in multi-view medical diagnosis.* The model's attention becomes overly focused on one of the two available views, resulting in an incomplete interpretation of the case. In this example (**top**), model attention in the MLO view dominates over the CC view in CBIS-DDSM (**top left**), and the frontal view over the lateral view in CheXpert (**top right**). Random Token Fusion (RTF) encourages the model to better utilize information from both views, resulting in balanced attention between both views and increased performance (**bottom**).

Token Fusion (RTF). The motivation for RTF is the observation that models often overfit to the more informative or dominant view, neglecting complementary information in other views [42, 40], as illustrated in Figure 1. This is particularly problematic in medical image analysis, where each view can provide unique diagnostic insights [11, 15, 31, 18]. Integrating information occlusion and mixing into the training process has been a proven method to combat overfitting and improve robustness [7, 22, 46, 45]. RTF randomly fuses tokens from different views, introducing variability in the fused representation, which acts as a regularizer. This forces the model to consider diverse combinations of tokens from both views, and reduces overfitting to any single view.

RTF can be seamlessly integrated with existing multi-view fusion strategies for vision transformers (ViTs), enriching an existing model's feature space without requiring any modification to the inference process. Through extensive experiments on two public medical benchmark datasets, CBIS-DDSM [20] and CheXpert [18], we demonstrate that RTF consistently improves performance of multi-view models, achieving state-of-the-art performance on both mammograms and chest X-rays. The source code used in this work can be found at https://jimyug.github.io/RandomTokenFusion.

## 2 Related Work

### 2.1 Multi-view Fusion in Medical Imaging

Multi-view fusion is a technique for integrating multiple input signals that typically represent the same object or class [43]. It has been used in 3D scene comprehension in the natural domain [35, 30, 10, 12], and has recently gained attention in the medical field for improving diagnostics by exploiting complementary information from multiple views taken from the same exam [26, 42].

For chest X-rays [15, 38] and mammography screenings [39, 36, 28, 42, 4], studies have demonstrated the utility of merging different views. Although it has been established that registration is not necessary for fusing multiple views [2, 4], simply combining modalities does not guarantee enhanced performance [15, 42]. It is observed that in some cases, one view may overshadow the other in terms of relevance, dominating the learning signal [15, 42]. As such, various approaches to multi-view fusion have been investigated.

Wang *et al.* [39] adopt an attention mechanism coupled with a recurrent network to merge the two views. Similarly, Iftikhar *et al.* [17] propose a sensitivity-based weighting mechanism to fuse the predictions on individual views. Lopez *et al.* [24] treat the two views as separate channels, constructing a channel-wise input, and employ convolutions to learn the correlations between them. Liu *et al.* [23] employ graph convolutional networks to discern geometric and semantic relationships

between two mammography views before merging them. MommiNet-v2 [44] emulates radiologists' practice by integrating symmetry and geometry constraints for improved performance. Wu *et al.* [42] randomly omit one of the views during training, forcing the network to utilize both views for improved performance. With the emerging trend of vision transformers, their application in medical imaging has great potential with the ability to model complex dependencies across diverse perspectives of subjects. Van Tulder *et al.* [38] introduce cross-view attention at feature level to transfer information between unregistered views, an approach further proven effective by [37] Moreover, Chen *et al.* [4] demonstrate that vision transformers can effectively analyze unregistered multi-view medical images. Kim [19] and Black *et al.* [1] also show that the self-attention mechanism can help aggregate multi-view features efficiently.

### 2.2 Information Mixing as Regularization

Information occlusion and mixing have been used as a regularization technique in various other contexts. Techniques such as CutOut [7] and PatchDropout [22] have shown their efficacy as regularizers by denying partial information. Methods such as [46, 45, 9] have demonstrated useful regularization effects when mixing information of different inputs. Mixup [46] and CutMix [45] mix samples at image level during training, while PatchUp [9] works in the hidden space of CNNs. For ViTs, Liu *et al.* [21] adapt Mixup by blending two images at the token level, guided by contextual activation maps. Similarly, TokenMixup [5] employs self-attention-based saliency to guide the mixing process towards the most significant tokens. Zhao *et al.* [47] further advance this approach by optimizing both token and label spaces for ViTs.

The aforementioned methods provide effective strategies for increasing the diversity of data. They have been applied to random image pairs of different objects and labels, although the pairs are intentionally constructed and not intended to be mixed for the specific tasks, *e.g.*. classification. Naturally, we believe that the idea could benefit multi-view medical diagnosis where image pairs exist naturally: They originate from the same object and serve the exact same diagnostic goals, and we want to encourage models to fully utilize both inputs. To this end, the proposed RTF randomly selects features from both views before fusion, effectively functioning as a regularizer. By augmenting the feature space, RTF encourages models to analyze cases more comprehensively, leading to improved performance.

## 3 Methods

Combining information from multiple views in medical image analysis enhances diagnostic accuracy, but existing methods often overfit by relying too much on the most informative view [42, 40]. To address this, we introduce Random Token Fusion (RTF). RTF is designed to augment feature learning in multi-view vision transformers by randomly selecting tokens from both views during the fusion phase of training, forcing the model to utilize information from both views effectively (See Figure 2).

Given two views $x_{1,2} \in \mathbb{R}^{H \times W \times C}$, where $H \times W$ is the spatial dimension and $C$ the number of image channels, a local encoder $f_{\theta_{local}}$ processes each input independently, and generates representations for both views:

$$z_{1,2} = f_{\theta_{local}}(x_{1,2}) \tag{1}$$

where $z_{1,2} \in \mathbb{R}^{(N+1) \times D}$, are the local representations of $x_1$ and $x_2$, consisting of $N$ patch tokens and a class (CLS) token [8] of size $D$. Then, $z_{1,2}$ are processed by the fusion module that divides them into two different fusion branches: the random token fusion module $RTF_F$ (described in 3.1) and the global fusion module $Global_F$ (described in 3.2).

Subsequently, the fused outputs are independently forwarded to a global encoder $f_{\theta_{global}}$ that produces the final predictions for both the global and RTF tokens. The overall process for the two branches is described as follows:

$$\hat{y} = f_{\theta_{global}}(Global_F(f_{\theta_{local}}(x_{1,2}))) \tag{2}$$

for the global branch, and

$$\hat{y}_{RTF} = f_{\theta_{global}}(RTF_F(f_{\theta_{local}}(x_{1,2}))) \tag{3}$$

for the RTF branch, where $\hat{y}$ and $\hat{y}_{RTF}$ represent the model's predictions for the global and RTF branch, respectively. In this work, we construct the local and the global encoder from the original ViT

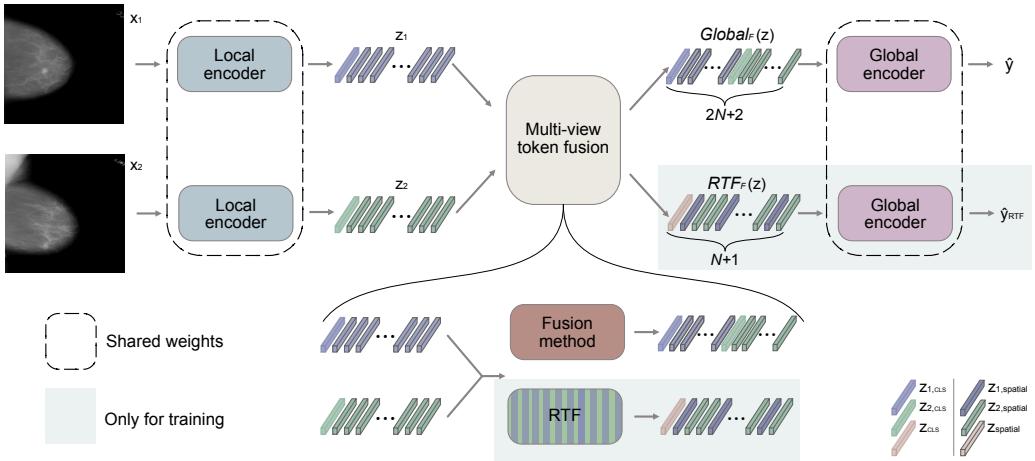


Figure 2: *Multi-view ViTs with Random Token Fusion (RTF).* RTF utilizes a local encoder to generate representations of different views, followed by a token fusion module. This module divides the feature fusion into two distinct branches. One branch uses some strategy to merge all tokens from both views, while the other one randomly drops spatial tokens from each view before mixing them. The fused tokens are processed by a global encoder, which produces two types of predictions: one for the global tokens and one for the RTF tokens. During training, the loss for both branches is minimized towards the same task. After training, RTF tokens are not generated, they are merged using the model's fusion method and passed to the global encoder for inference.

variants [8] by allocating a certain proportion of the initial *Encoder* blocks to form the local encoder, while the remaining blocks constitute the global encoder.

During training the objective is to minimize a combined loss function, considering both the global and RTF branch

$$\mathcal{L} = \ell_{CE}\left(\hat{y}, y\right) + \ell_{CE}\left(\hat{y}_{RTF}, y\right) \tag{4}$$

where $\ell_{CE}$ is the cross-entropy loss between the model's predictions and the target $y$. At inference time, only the process described by Equation 2 is used.

### 3.1 Random Token Fusion

To better instruct the model to utilize information from both views, we introduce the Random Token Fusion (RTF). RTF maximizes the mutual information $I(Z;Y)$ between the fused representation $Z$ and the target $Y$, mathematically defined as:

$$I(Z;Y) = H(Z) - H(Z|Y) \tag{5}$$

where $H(Z)$ is the entropy of the fused representation $Z$, and $H(Z|Y)$ is the conditional entropy given the target $Y$. By incorporating randomness into the token fusion process, RTF encourages the model to learn robust and generalized features from all views, ensuring that the fused representation captures the most informative features.

RTF randomly selects part of the tokens from each view prior to fusion (see Figure 3 Right). Specifically, given two features $z_{1,2} \in \mathbb{R}^{(N+1)\times D}$, RTF randomly selects tokens from either view to form a mixed representation

$$z_{\text{spatial}} = \mathbb{1}_M \odot z_{1,\text{spatial}} + (\mathbb{1} - \mathbb{1}_M) \odot z_{2,\text{spatial}}, \tag{6}$$

for the spatial tokens, where $\mathbb{1}_M \in \mathbb{R}^N$ is a binary mask, whose elements take the value of 1 with a probability that follows a uniform distribution $p \sim \mathcal{U}(0,1)$. To preserve high-level features used for classification from both views, we average the CLS tokens $z_{\text{cls}} = \frac{z_{1,\text{cls}} + z_{2,\text{cls}}}{2}$. We concatenate $z_{\text{cls}}$ and $z_{\text{spatial}}$ to form the final fused feature $z = \{z_{\text{cls}}, z_{\text{spatial}}\}$. By doing so, it compels the network to capture dependencies between patches originating from different views, preventing the model from overfitting to view-specific features.

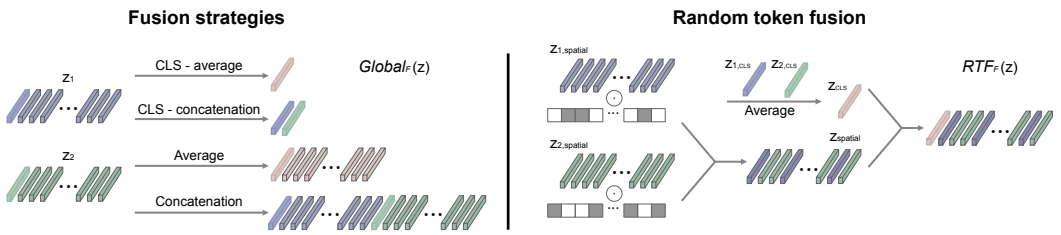


Figure 3: *Illustration of different fusion strategies.* **(Left)** Common fusion strategies to fuse the features (tokens) of different views in ViTs. **(Right)** The proposed random token fusion (RTF) strategy. In RTF, we randomly drop spatial tokens from both images and combine the remaining ones, augmenting the representations during training.

RTF increases both $H(Z)$ and $H(Z|Y)$ by introducing randomness in the token selection process. Despite the increase in $H(Z|Y)$, the overall mutual information $I(Z;Y)$ still benefits because the added randomness enhances the diversity and robustness of the representations, helping the model capture informative complementary features from other views.

### 3.2 Fusion Strategies

RTF is designed to enhance common fusion strategies without bringing additional cost to inference. There are a number of ways to mix representations of different views. In this work, we consider the fusion strategies from the literature.

Given the representations of two views $z_1$ and $z_2$, where $z = \{z_{\text{cls}}, z_{\text{spatial}_1}, \ldots, z_{\text{spatial}_N}\}$, and $z_{\text{cls}}$ represents the class token, $z_{\text{spatial}_i}$ represents the $i$-th spatial token, and $i$ ranges from $1$ to $N$, one strategy is to perform some operation on the features to produce a fused feature $z$. The operation could be token-wise, such as concatenation or selection, or element-wise, such as addition, subtraction, multiplication, max, etc.

In this work, we consider token-wise fusion with concatenation, a choice widely adopted in current research [4, 38], implemented for ViTs by concatenating all tokens from two views. As such, $z$ is defined as $z = \{z_{1,\text{cls}}, z_{1,\text{spatial}_1}, \ldots, z_{1,\text{spatial}_n}, z_{2,\text{cls}}, z_{2,\text{spatial}_1}, \ldots, z_{2,\text{spatial}_m}\}$, as depicted in Figure 3. Since ViTs are able to process an arbitrary number of tokens [38, 4], this is a trivial operation. While this approach has the benefit of preserving information, it adds a considerable cost to the memory and compute. A compromised solution would be keeping and fusing only the CLS tokens $z = \{z_{1_{\text{cls}}}, z_{2_{\text{cls}}}\}$ for efficiency.

An alternative approach is element-wise fusion by averaging [42], implemented by averaging all tokens from two views $z = \frac{z_1+z_2}{2}$ or by using only the CLS tokens, $z = \frac{z_{1,cls}+z_{2,cls}}{2}$ (see Figure 3). While this approach is straightforward and computationally efficient, it may lead to loss of spatial information, particularly when input images are not registered [38].

## 4 Experimental Setup

By design, RTF can be applied to most transformer-based multi-view fusion models. In this work, we employ the standard ViT family [8] for both the local and global encoders. Using ViT Tiny, Small, and Base, we conduct experiments on two standard benchmark medical datasets, CBIS-DDSM [20, 33] and CheXpert [18] to evaluate RTF performance on top of different fusion strategies, described in Section 3.2. We also compare against other multi-view fusion methods designed for ViTs and, for reference, against CNN-based multi-view fusion methods. For both datasets, we use the area under the receiver operating characteristic curve (AUC) to evaluate model performance. All experiments are repeated $4$ times, and we report the mean value and standard deviation.

### 4.1 Datasets

**CBIS-DDSM [20, 33]** is a well-known public mammography dataset, containing $10,239$ samples from $1,566$ unique patients, including both craniocaudal (CC) and mediolateral oblique (MLO) views.

Table 1: The effect of using only a single view, multiple views with late fusion, and multiple views with RTF on CBIS-DDSM (**left**) and CheXpert (**right**).

| View | **DDSM**, AUC ↑ |
|---|---|
| Only CC | $0.730 \pm 0.004$ |
| Only MLO | $0.747 \pm 0.022$ |
| Late fusion | $0.799 \pm 0.008$ |
| Fusion w/ RTF | $\mathbf{0.815 \pm 0.001}$ |

| View | **CheXpert**, AUC ↑ |
|---|---|
| Only Frontal | $0.838 \pm 0.003$ |
| Only Lateral | $0.832 \pm 0.002$ |
| Late fusion | $0.841 \pm 0.001$ |
| Fusion w/ RTF | $\mathbf{0.849 \pm 0.001}$ |

The objective is to classify mass abnormalities as either benign or malignant. We only select images from exams for which both CC and MLO views are available, similar to [38]. This results in a subset of $1,416$ images from $708$ exams, corresponding to $636$ patients.

**CheXpert [18]** is one of the largest publicly available datasets for chest radiograph interpretation, consisting of frontal and lateral chest X-ray scans. The original dataset contains $224,316$ chest radiographs of $65,240$ patients, annotated for 13 diagnostic observations, each labelled as negative, positive, uncertain, or unknown (missing). The uncertain and unknown samples are handled following the suggestion of [18]. For the purposes of this work, we only use exams that include both frontal and lateral views, totaling $62,826$ images from $31,413$ exams ($22,414$ patients).

### 4.2 Implementation Details

We resize all images to $384 \times 384$ and initialize all models with weights pretrained on ImageNet-21K [32, 41], as randomly initialized ViTs cannot effectively be trained on small medical datasets [27]. By default, we use ViT Small [8] as the backbone, with concatenation as the fusion strategy. $75\%$ of the total encoder blocks are used as the local encoder $f_{\theta_{local}}$, and the remaining $25\%$ are applied to the fused feature. This partitioning is used by default throughout the rest of our experiments. We use the AdamW optimizer [25] to train the models for $300$ epochs on CBIS-DDSM and for $60$ epochs on CheXpert. The learning rate is selected based on a grid search. For both datasets, we use spatial scaling, flipping, rotation, and color jittering for data augmentation. After training, the checkpoint with the highest AUC validation score is selected for testing. We employ the same train/validation/test split as in [38, 1] for both datasets. All experiments are conducted on NVIDIA GeForce RTX 4090 Ti GPUs with 24 GB of memory.

## 5 Results and Discussion

### 5.1 Ablation Study

We begin by confirming the advantages of multi-view over single-view ViTs across various fusion strategies. We then show the benefits of incorporating RTF with standard fusion techniques. We then delve into identifying the most compatible fusion strategy for RTF, and conduct ablation studies for different model sizes.

**Are there benefits to using multiple views compared to a single view?** To assess the benefits of multi-view ViTs, we conducted experiments training models on single views and with late fusions and report results in Table 1. Specifically, single-view models only consider one view throughout the task, while late-fusion models combine representations of different views at the end of the model [2, 1, 38]. We verify that, for both CBIS-DDSM and CheXpert, using two views results in improved

Table 2: Performance (AUC) depending on where the encoder is split for fusion, w/(w/o) RTF.

| **Local/ Global** | **25%/ 75%** | **50%/ 50%** | **75%/ 25%** |
|---|---|---|---|
| CBIS-DDSM | 0.802 (0.799) | 0.810 (0.799) | 0.815 (0.803) |
| CheXpert | 0.845 (0.842) | 0.849 (0.844) | 0.849 (0.843) |

Table 3: AUC performance on CBIS-DDSM, showing the effect of using multiple views with and without RTF for different model sizes and fusion strategies.

| Method | RTF Used | ViT Tiny | ViT Small | ViT Base |
|---|---|---|---|---|
| Average | No | $0.798 \pm 0.003$ | $0.803 \pm 0.008$ | $0.813 \pm 0.004$ |
| | Yes | $\mathbf{0.802 \pm 0.001}$ | $\mathbf{0.809 \pm 0.002}$ | $\mathbf{0.825 \pm 0.005}$ |
| $CLS_{cat}$ | No | $0.796 \pm 0.002$ | $0.802 \pm 0.006$ | $0.814 \pm 0.007$ |
| | Yes | $\mathbf{0.801 \pm 0.001}$ | $\mathbf{0.811 \pm 0.008}$ | $\mathbf{0.826 \pm 0.004}$ |
| Concat | No | $0.798 \pm 0.003$ | $0.803 \pm 0.003$ | $0.814 \pm 0.004$ |
| | Yes | $\mathbf{0.802 \pm 0.003}$ | $\mathbf{0.815 \pm 0.001}$ | $\mathbf{0.830 \pm 0.002}$ |

Table 4: AUC performance on CheXpert, showing the effect of using multiple views with and without RTF for different model sizes and fusion strategies.

| Method | RTF Used | ViT Tiny | ViT Small | ViT Base |
|---|---|---|---|---|
| Average | No | $0.835 \pm 0.003$ | $0.844 \pm 0.004$ | $0.847 \pm 0.003$ |
| | Yes | $\mathbf{0.838 \pm 0.003}$ | $\mathbf{0.848 \pm 0.002}$ | $\mathbf{0.850 \pm 0.002}$ |
| $CLS_{cat}$ | No | $0.833 \pm 0.003$ | $0.842 \pm 0.003$ | $0.846 \pm 0.002$ |
| | Yes | $\mathbf{0.837 \pm 0.003}$ | $\mathbf{0.846 \pm 0.001}$ | $\mathbf{0.850 \pm 0.001}$ |
| Concat | No | $0.833 \pm 0.002$ | $0.843 \pm 0.004$ | $0.847 \pm 0.004$ |
| | Yes | $\mathbf{0.839 \pm 0.002}$ | $\mathbf{0.849 \pm 0.001}$ | $\mathbf{0.852 \pm 0.002}$ |

AUC performance compared to single-view models with the same capacity, and fusing information at feature level is more effective than at a late stage, which is consistent with prior findings [15, 42]. This also highlights the importance of encouraging models to utilize both perspectives in medical diagnosis, which is a key motivation for our research.

To determine the optimal point for fusing local features, we conducted experiments to assess its impact on performance. We derived two key observations from the results [2] in Table 2. First, the advantage of integrating multiple views within a model, as opposed to late fusion, generally persists. On both datasets, effective intermediate feature fusion outperforms late fusion, as shown in Tables 1 and 2. Second, RTF consistently enhances performance across all settings. The best results are achieved when the first $75\%$ of the blocks are used for the local encoder, and the remaining $25\%$ blocks for the global encoder. We adopt this as the default setting in other experiments.

**What are the benefits from using RTF to enhance multi-view fusion?** To understand the impact of integrating RTF with common fusion strategies, we train models of different capacities employing the three token fusion strategies described in Section 3.2: averaging all tokens (Average), concatenating only the CLS tokens ($CLS_{cat}$), and concatenating all tokens (Concat) from two views, with and without RTF. We present the results on CBIS-DDSM and CheXpert in Table 3 and Table 4, respectively.

The results show that training with RTF consistently improves performance across all configurations. And the extent of improvement varies with the dataset and model size. CBIS-DDSM appears to gain more from RTF, particularly for larger ViT variants. We hypothesize that this is due to the regularization effects of RTF and the smaller size of the dataset, as higher-capacity models are more prone to overfitting [16]. Overall, integrating RTF with concatenation fusion for the global encoder generally yields better performance compared to other fusion methods, as it preserves the tokens and retains the most information. Therefore, we adopt this combination as the default setting for later comparisons.

Inspired by recent advancements in hybrid architectures [19, 1], we aim to examine the efficacy of RTF with different backbone choices. Compared to the plain ViT [8], hybrid architectures adopt more complex and powerful designs, *e.g.*, the ResNet family or other CNNs, to project input images to tokens before feeding them into a transformer-based network. Following [1], we employ three variants

[2] Extended tables can be found in the Supplement.

Table 5: Comparison vs. SOTA methods on CBIS-DDSM (**left**) and CheXpert (**right**).

| Method | CBIS-DDSM |
|---|---|
| ResNet50 [24] | $0.724 \pm 0.007$ |
| Shared ResNet [42] | $0.735 \pm 0.014$ |
| PHResNet50 [24] | $0.739 \pm 0.004$ |
| MVT [3, 4] | $0.803 \pm 0.003$ |
| CVT [38] | $0.803 \pm 0.007$ |
| ViT-Average [29] | $0.803 \pm 0.008$ |
| **RTF** | $\mathbf{0.815 \pm 0.001}$ |

| Method | CheXpert |
|---|---|
| MVC-NET [48] | $0.813 \pm 0.005$ |
| MVCNN [34] | $0.815 \pm 0.004$ |
| CVT [38] | $0.834 \pm 0.002$ |
| MVT [3, 4] | $0.843 \pm 0.004$ |
| ViT-Average [29] | $0.844 \pm 0.004$ |
| MV-HFMD [1] | $0.845 \pm 0.002$ |
| **RTF** | $\mathbf{0.849 \pm 0.001}$ |

of the "ResNet + ViT" design and apply RTF between the ResNet and ViT Encoder. We apply RTF to the tokens generated by the local CNN encoders and follow the same procedure outlined in Figure 2. As shown in Table 6, RTF consistently benefits the hybrid models. Although hybrid architectures generally achieve better performance than plain ViTs, they are computationally expensive and require larger memory [8, 41]. As such, we opt for standard ViTs as the main backbone in this work.

### 5.2 Comparison with State of the Art

We benchmark RTF against state-of-the-art (SOTA) methods from the literature. On CBIS-DDSM, we include ResNet50 [24], Shared ResNet [42], PHResNet50 [24], Multi-view Transformer (MVT) [3, 4], ViT with average fusion (ViT-Average) [29], and Cross-view Transformer (CVT) [38]. On CheXpert, we compare RTF with MVC-NET [48], MVCNN [34], MVT [3, 4], ViT-Average [29], CVT [38], and MV-HFMD [1]. We choose RTF Small, *i.e.*, ViT Small with RTF, which has a comparable capacity with other models.

As shown in Table 5, RTF outperforms all methods on both datasets, showing its efficacy in multi-view medical diagnosis. Notably, RTF can be used in conjunction with transformer-based methods, such as ViT-Average [29] and MVT [3, 4], for further enhanced performance, as demonstrated in Section 5.1. By design, RTF during inference is essentially a standard ViT [8] with concatenation for feature fusion. The fact that RTF achieves better performance on CheXpert than MV-HFMD [1] – a hybrid architecture – highlights the potential of ViTs in this application when empowered with RTF. It is important to note that comparing RTF against CNN-based methods may place the latter at a disadvantage, as evidence suggests that transformers are more adept at fusion tasks [4, 14]. Nevertheless, these comparisons are presented for completeness.

### 5.3 Qualitative Results

The intuition behind RTF is that by withholding information during training, the model is encouraged to focus its attention on features from both views. To better understand the impact of RTF on model attention, we provide qualitative examples from CBIS-DDSM and CheXpert. In Figure 4, we compare attention maps within the last block of the global encoder from models trained with and without RTF, with each row showing the results of an image pair with two views. On CBIS-DDSM, the baseline model sometimes focuses on uninformative background, a phenomenon observed where patch information is redundant, as discussed in [6]. This may lead to poor generalization. With RTF, the quality of the attention maps generally improves, with more attention focused on relevant areas. RTF also appears to encourage a more balanced focus between both views in many cases. On CheXpert, the model with RTF tends to consider both views, while the baseline model often ignores the lateral view, which may provide valuable information for diagnosis [15, 18]. More examples are included in the Supplement.

Table 6: Performance (AUC) on CheXpert of hybrid models, w/ and w/o RTF.

| Fusion | R+ViT-Ti/16 | R26+ViT-S/16 | R50+ViT-B/16 |
|---|---|---|---|
| w/o RTF | $0.834 \pm 0.002$ | $0.846 \pm 0.002$ | $0.848 \pm 0.001$ |
| w/ RTF | $0.839 \pm 0.002$ | $0.852 \pm 0.002$ | $0.854 \pm 0.002$ |

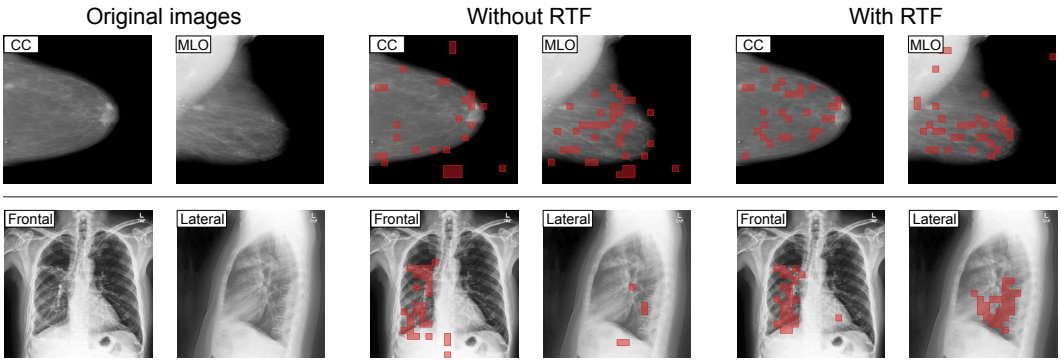


Figure 4: *RTF encourages the model to learn from both views, resulting in better quality of attention maps.* Given two views (**left**), standard fusion methods overfit on a specific view (MLO in CBIS-DDSM and Frontal in CheXpert, in this example), under-utilizing information from the other (**middle**). RTF encourages the model to use information for both views to make better predictions (**right**). For mammograms, the model sometimes focuses on uninformative background, a known issue raised when information of the patches is redundant, and may lead to poor generalization. With RTF, the quality of the attention maps is better in general, with more attention focused on relevant areas. On CheXpert, the baseline model ignores the lateral view that also provide valuable information for diagnosis. RTF also seems to encourage more equal focus between both views.

### 5.4 Further Discussion

**Can RTF work as a standalone, efficient training method?** In this study, we propose RTF as a solution to enhance existing multi-view fusion strategies. RTF is designed to function with an additional branch in the network and loss function, making it compatible with various manipulations of intermediate features (tokens). Some approaches corrupt spatial information, while others retain all tokens as-is or completely discard the spatial ones. It would be intriguing to explore whether RTF alone could serve as an effective fusion strategy, potentially reducing computational requirements through random token selection compared to simple concatenation, and further improving performance. The potential positive outcomes could contribute to both efficient training and robust medical diagnosis. We plan to investigate this further in future work.

## 6 Conclusion

In this work, we focus on multi-view vision transformers for medical image analysis. We address shortcomings of current fusion methods that tend to overfit on view-specific features, not fully leveraging information from all views. Random token fusion (RTF) randomly drops tokens from both views during the fusion phase of training, encouraging the model to learn more robust representations across all views. RTF directly impacts the model's attention and enhances its performance without any additional cost at inference. Our experiments show that RTF exceeds the performance of other fusion methods and seamlessly boosts performance when combined with them – the degree of improvement is influenced by the dataset and model size. While our work focuses on multi-view ViTs for medical diagnosis with chest X-rays and mammograms, we believe that our findings can be extended to multiple medical modalities, as well as other tasks and domains, which we leave for future work.

**Acknowledgements.** This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation.

## References


[1] Samuel Black and Richard Souvenir. "Multi-view Classification Using Hybrid Fusion and Mutual Distillation". In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2024, pp. 270–280.

[2] Gustavo Carneiro, Jacinto Nascimento, and Andrew P Bradley. "Deep learning models for classifying mammogram exams containing unregistered multi-view images and segmentation maps of lesions". In: *Deep learning for medical image analysis* (2017), pp. 321–339.

[3] Shuo Chen, Tan Yu, and Ping Li. "MVT: Multi-view Vision Transformer for 3D Object Recognition". In: *BMVC*. 2021.

[4] Xuxin Chen et al. "Transformers improve breast cancer diagnosis from unregistered multi-view mammograms". In: *Diagnostics* 12.7 (2022), p. 1549.

[5] Hyeong Kyu Choi, Joonmyung Choi, and Hyunwoo J Kim. "Tokenmixup: Efficient attention-guided token-level data augmentation for transformers". In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 14224–14235.

[6] Timothée Darcet et al. "Vision transformers need registers". In: *arXiv preprint arXiv:2309.16588* (2023).

[7] Terrance DeVries and Graham W Taylor. "Improved regularization of convolutional neural networks with cutout". In: *arXiv preprint arXiv:1708.04552* (2017).

[8] Alexey Dosovitskiy et al. "An image is worth 16x16 words: Transformers for image recognition at scale". In: *arXiv preprint arXiv:2010.11929* (2020).

[9] Mojtaba Faramarzi et al. "Patchup: A feature-space block-level regularization technique for convolutional neural networks". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 36. 2022, pp. 589–597.

[10] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. "Convolutional two-stream network fusion for video action recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 1933–1941.

[11] David S. Feigin. "Lateral Chest Radiograph". In: *Academic Radiology* 17.12 (2010), pp. 1560–1566. ISSN: 1076-6332. DOI: `10.1016/j.acra.2010.07.004`.

[12] Konrad Gadzicki, Razieh Khamsehashari, and Christoph Zetzsche. "Early vs late fusion in multimodal convolutional neural networks". In: *2020 IEEE 23rd international conference on information fusion (FUSION)*. IEEE. 2020, pp. 1–6.

[13] A.K. Hackshaw et al. "An Investigation Into Why Two-View Mammography is Better than One-View in Breast Cancer Screening". In: *Clinical Radiology* 55.6 (2000), pp. 454–458. ISSN: 0009-9260.

[14] Khalid Al-Hammuri et al. "Vision transformer architecture and applications in digital health: a tutorial and survey". In: *Visual Computing for Industry, Biomedicine, and Art* 6.1 (2023), p. 14.

[15] Mohammad Hashir, Hadrien Bertrand, and Joseph Paul Cohen. "Quantifying the Value of Lateral Views in Deep Learning for Chest X-rays". In: *Proceedings of the Third Conference on Medical Imaging with Deep Learning*. Ed. by Tal Arbel et al. Vol. 121. Proceedings of Machine Learning Research. PMLR, 2020, pp. 288–303.

[16] Geoffrey E Hinton et al. "Improving neural networks by preventing co-adaptation of feature detectors". In: *arXiv preprint arXiv:1207.0580* (2012).

[17] Hina Iftikhar et al. "Multi-View Attention-based Late Fusion (MVALF) CADx system for breast cancer using deep learning". In: *Machine Graphics and Vision* (2020). URL: `https://api.semanticscholar.org/CorpusID:239001683`.

[18] Jeremy Irvin et al. "CheXpert: A Large Chest Radiograph Dataset with Uncertainty Labels and Expert Comparison". In: *Proceedings of the AAAI Conference on Artificial Intelligence* 33 (July 2019), pp. 590–597.

[19] Dongkyun Kim. "CheXFusion: Effective Fusion of Multi-View Features using Transformers for Long-Tailed Chest X-Ray Classification". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 2702–2710.

[20] Rebecca Sawyer Lee et al. "A curated mammography data set for use in computer-aided detection and diagnosis research". In: *Scientific Data* 4.1 (2017), p. 170177. ISSN: 2052-4463.

[21] Jihao Liu et al. "Tokenmix: Rethinking image mixing for data augmentation in vision transformers". In: *European Conference on Computer Vision*. Springer. 2022, pp. 455–471.

[22] Yue Liu et al. "Patchdropout: Economizing vision transformers using patch dropout". In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2023, pp. 3953–3962.

[23] Yuhang Liu et al. "Act like a radiologist: towards reliable multi-view correspondence reasoning for mammogram mass detection". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44.10 (2021), pp. 5947–5961.

[24] Eleonora Lopez et al. "Hypercomplex Neural Architectures for Multi-View Breast Cancer Classification". In: *arXiv e-prints*, arXiv:2204.05798 (Apr. 2022), arXiv:2204.05798. arXiv: 2204.05798 [cs.CV].

[25] Ilya Loshchilov and Frank Hutter. "Decoupled weight decay regularization". In: *arXiv preprint arXiv:1711.05101* (2017).

[26] Ravikiran Mane et al. "A multi-view CNN with novel variance layer for motor imagery brain computer interface". In: *2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*. IEEE. 2020, pp. 2950–2953.

[27] Christos Matsoukas et al. "Is it time to replace cnns with transformers for medical images?" In: *arXiv preprint arXiv:2108.09038* (2021).

[28] Hasan Nasir Khan et al. "Multi-View Feature Fusion Based Four Views Model for Mammogram Classification Using Convolutional Neural Network". In: *IEEE Access* 7 (2019), pp. 165724–165733.

[29] Huyen TX Nguyen et al. "A novel multi-view deep learning approach for BI-RADS and density assessment of mammograms". In: *2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*. IEEE. 2022, pp. 2144–2148.

[30] Charles R Qi et al. "Volumetric and multi-view cnns for object classification on 3d data". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 5648–5656.

[31] Suhail Raoof et al. "Interpretation of Plain Chest Roentgenogram". In: *CHEST* 141.2 (2012), pp. 545–558. ISSN: 0012-3692. DOI: 10.1378/chest.10-1302.

[32] Olga Russakovsky et al. "ImageNet Large Scale Visual Recognition Challenge". In: *International Journal of Computer Vision (IJCV)* 115.3 (2015), pp. 211–252.

[33] Rebecca Sawyer-Lee et al. *Curated Breast Imaging Subset of DDSM*. 2016. DOI: 10.7937/k9/tcia.2016.7o02s9cy. URL: https://wiki.cancerimagingarchive.net/x/lZNXAQ.

[34] Hang Su et al. "Multi-view Convolutional Neural Networks for 3D Shape Recognition". In: *2015 IEEE International Conference on Computer Vision (ICCV)* (2015), pp. 945–953. URL: https://api.semanticscholar.org/CorpusID:2407217.

[35] Hang Su et al. "Multi-view convolutional neural networks for 3d shape recognition". In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 945–953.

[36] Lilei Sun et al. "Multi-View Convolutional Neural Networks for Mammographic Image Classification". In: *IEEE Access* 7 (2019), pp. 126273–126282.

[37] Yen Nhi Truong Vu et al. "M&M: Tackling False Positives in Mammography with a Multi-view and Multi-instance Learning Sparse Detector". In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2023*. Ed. by Hayit Greenspan et al. Cham: Springer Nature Switzerland, 2023, pp. 778–788. ISBN: 978-3-031-43904-9.

[38] Gijs van Tulder, Yao Tong, and Elena Marchiori. "Multi-view analysis of unregistered medical images using cross-view transformers". In: *Medical Image Computing and Computer Assisted Intervention–MICCAI 2021: 24th International Conference, Strasbourg, France, September 27–October 1, 2021, Proceedings, Part III 24*. Springer. 2021, pp. 104–113.

[39] Hongyu Wang et al. "Breast mass classification via deeply integrating the contextual information from multi-view data". In: *Pattern Recognition* 80 (2018), pp. 42–52. ISSN: 0031-3203.

[40] Weiyao Wang, Du Tran, and Matt Feiszli. "What makes training multi-modal classification networks hard?" In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 12695–12705.

[41] Ross Wightman. *PyTorch Image Models*. https://github.com/rwightman/pytorch-image-models. 2019.

[42] Nan Wu et al. "Improving the ability of deep neural networks to use information from multiple views in breast cancer screening". In: *Medical Imaging with Deep Learning*. PMLR. 2020, pp. 827–842.

[43] Xiaoqiang Yan et al. “Deep multi-view learning methods: A review”. In: *Neurocomputing* 448 (2021), pp. 106–129.

[44] Zhicheng Yang et al. “MommiNet-v2: Mammographic multi-view mass identification networks”. In: *Medical Image Analysis* 73 (2021), p. 102204. ISSN: 1361-8415.

[45] Sangdoo Yun et al. “Cutmix: Regularization strategy to train strong classifiers with localizable features”. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2019, pp. 6023–6032.

[46] Hongyi Zhang et al. “mixup: Beyond empirical risk minimization”. In: *arXiv preprint arXiv:1710.09412* (2017).

[47] Qihao Zhao et al. “MixPro: Data Augmentation with MaskMix and Progressive Attention Labeling for Vision Transformer”. In: *arXiv preprint arXiv:2304.12043* (2023).

[48] Xiongfeng Zhu and Qianjin Feng. “MVC-NET: Multi-View Chest Radiograph Classification Network With Deep Fusion”. In: *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*. 2021, pp. 554–558. DOI: 10.1109/ISBI48211.2021.9434000.

# Appendix

## Appendix A Extended Results

We report additional experimental results included in the main paper. Table 7 and Table 8 extend Table 2 of the main paper. We investigate how many blocks of a standard ViT are used as the local encoder and the remaining as the global encoder for the fused tokens with all mentioned fusion strategies. Two key findings emerge from the results:

First, concatenation proves more robust to the choice of local/global ratio compared to the other fusion strategies. This robustness is expected, as concatenation preserves the information to the most extent and can fully utilize the global transformer blocks. Based on these results, we select concatenation as the default fusion strategy. Second, RTF generally enhances performance across all settings. The only exception occurs when using $25\%$ blocks as the local encoder with $\text{CLS}_{\text{cat}}$. In this scenario, all spatial tokens are discarded at a very early stage, and only the two CLS tokens are sent to the global encoder, resulting in extremely low model capacity. Applying RTF in this situation harms performance, similar to the effects of aggressive regularization techniques on an already under-fitting model.

Table 7: Extended results on CBIS-DDSM, showing AUC performance depending on where the encoder is split for fusion.

| Fusion | RTF Used | 25% local | 50% local | 75% local |
|---|---|---|---|---|
| Average | No | $0.753 \pm 0.007$ | $0.789 \pm 0.014$ | $0.803 \pm 0.008$ |
| | Yes | $\mathbf{0.756 \pm 0.011}$ | $\mathbf{0.793 \pm 0.006}$ | $\mathbf{0.809 \pm 0.002}$ |
| $\text{CLS}_{\text{cat}}$ | No | $\mathbf{0.711 \pm 0.012}$ | $0.782 \pm 0.007$ | $0.802 \pm 0.006$ |
| | Yes | $0.709 \pm 0.005$ | $\mathbf{0.796 \pm 0.001}$ | $\mathbf{0.811 \pm 0.008}$ |
| Concat | No | $0.799 \pm 0.002$ | $0.799 \pm 0.009$ | $0.803 \pm 0.003$ |
| | Yes | $\mathbf{0.802 \pm 0.001}$ | $\mathbf{0.810 \pm 0.003}$ | $\mathbf{0.815 \pm 0.001}$ |

Table 8: Extended results on CheXpert, showing AUC performance depending on where the encoder is split for fusion.

| Fusion | RTF Used | 25% local | 50% local | 75% local |
|---|---|---|---|---|
| Average | No | $0.834 \pm 0.004$ | $0.845 \pm 0.002$ | $0.844 \pm 0.004$ |
| | Yes | $\mathbf{0.835 \pm 0.003}$ | $\mathbf{0.849 \pm 0.001}$ | $\mathbf{0.848 \pm 0.002}$ |
| $\text{CLS}_{\text{cat}}$ | No | $\mathbf{0.815 \pm 0.003}$ | $0.841 \pm 0.003$ | $0.842 \pm 0.006$ |
| | Yes | $0.814 \pm 0.003$ | $\mathbf{0.844 \pm 0.001}$ | $\mathbf{0.846 \pm 0.001}$ |
| Concat | No | $0.842 \pm 0.003$ | $0.844 \pm 0.003$ | $0.843 \pm 0.004$ |
| | Yes | $\mathbf{0.845 \pm 0.002}$ | $\mathbf{0.849 \pm 0.001}$ | $\mathbf{0.849 \pm 0.001}$ |

## Appendix B Additional Saliency Results

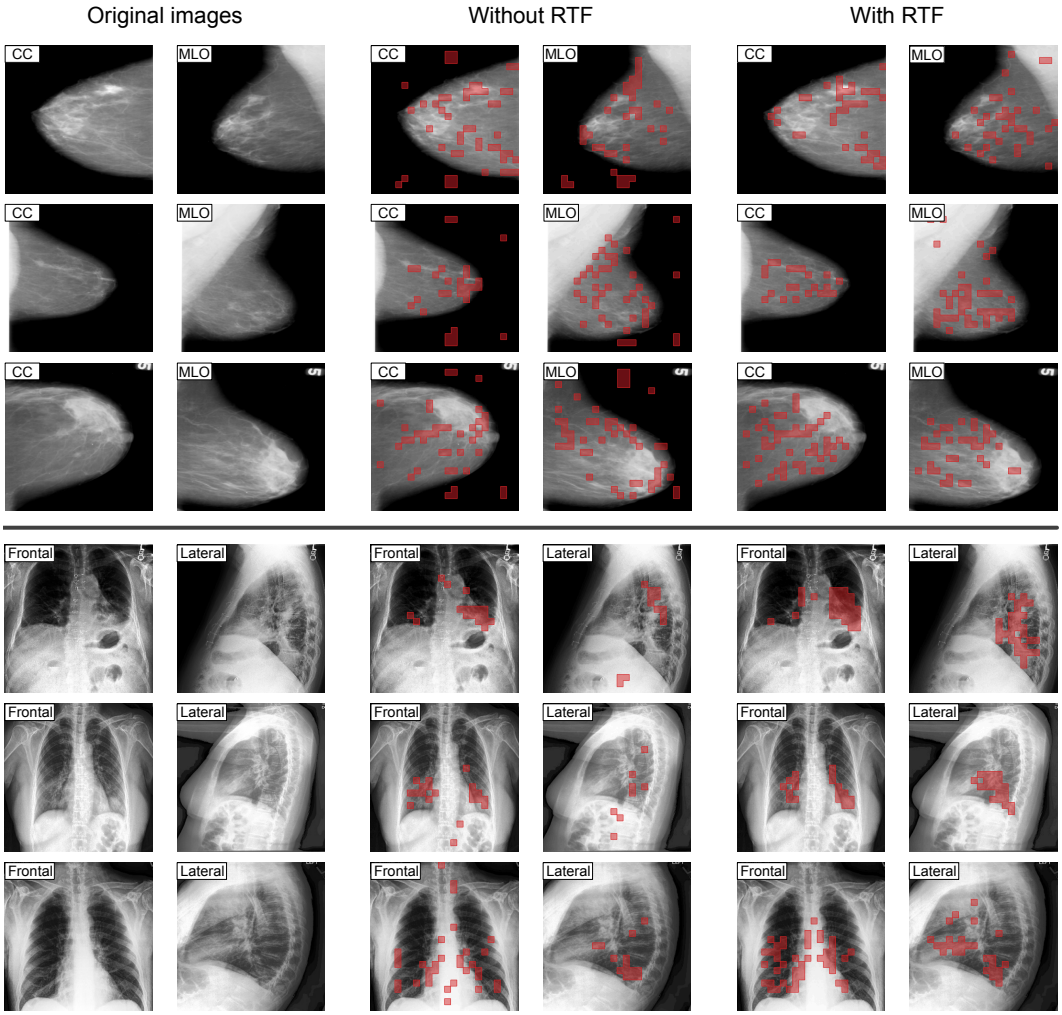


Figure 5: Extended results on CBIS-DDSM (**top**) and CheXpert (**bottom**), showing the model's attention maps within the last block of the global encoder. RTF seems to address the issue of attention being allocated to uninformative areas, a common phenomenon observed in ViTs. It also encourages the model to focus on both views in many cases.

## NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We discuss in detail the contributions and scope of this work in Section 1 Introduction.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We have listed the limitations of our proposed method and under which assumptions it works as described (Section 5.4).

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: This paper does not include any theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provided detailed information on the architecture and hyperparameter selection.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: This paper uses public data, and the code will be published upon acceptance.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: We report in detail in experimental settings used in this work (Section 4) and we will also release the code after publication..

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: We report the mean results and the standard deviations, following the standard protocol in this field.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: While this paper does not introduce methods that alter the common training compute demands, we include sufficient information in the main paper and the appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: No human subjects or crowdsourcing were used during the study. We adhered to the Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discussed potential impacts on AI-based medical diagnosis.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: Our study is about token fusion in ViTs and it does not pose a risk for misuse.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We properly cite papers, data and models we used in this paper.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: We will release the code after publication. No other assets will be released.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: This paper does not involve research with human subjects. All data used is public.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: This paper does not involve research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.