Using SSRL Models to Classify T-cell Autofluorescent Images

Gabe Seir
Stanford University
gseir@stanford.edu

Jennifer Xu
Stanford University
jennxu23@stanford.edu

Trevor Carrell
Stanford University
carrtre@stanford.edu

Abstract

T-cell based immunotherapies hold great promise for treating cancer. However, their efficacy is often limited by issues such as specificity, toxicity, and longevity. A crucial aspect of developing these therapies is the accurate characterization of T-cells. While supervised computer vision models have shown success in analyzing and classifying autofluorescent T-cell images as active or quiescent, they require large, labeled training sets, which are labor-intensive to produce. Self-supervised models present a viable alternative, leveraging recent advancements in computer vision to reduce the dependency on labeled data. In this project, we evaluate the performance of DINOv2, a popular self-supervised model, in classifying T-cell activation states. Based on our experiments, we see that DINOv2 can achieve comparable to, and possibly even better performance than, traditional supervised models. However, we also find that further fine-tuning does not improve the performance of DINOv2 models, and that overall a supervised, fine-tuned CNN is better at determining the activation state of a T-cell.

1. Introduction

Biological image modalities are critical to the understanding and diagnoses of diseases and are rich in data. Developments in both biological imaging and machine learning, particularly in the field of computer vision, have led to new ways of analyzing and recovering data in biological images. However, the curation of high-quality, large biological image datasets is uniquely challenging as generating these datasets is time, cost and labor intensive [8]. Moreover, these datasets often have issues of class-imbalance, high dimensionality, and large datasize size, which hinder performance of deep learning models and require significant compute resources to train such models, respectively [5, 11].

However, recent advancements in deep learning models may remedy these issues with minimal concessions in model performance. Specifically, advancements in large-scale pretrained models provide solid foundations for fine-tuning and allow deep learning problems to be solved with reduced data requirements [7]. Moreover, with recent increases in research efforts focused on self-supervised models, groundbreaking model architectures, such as DINO, self-supervised GANs, and self-predictive vision transformer models, have reduced the performance gap between supervised and self-supervised learning approaches [8, 14, 16]. Ultimately, these advancements have increased the viability of using large, pretrained, self-supervised models for deep learning tasks as they require less data, less compute, and have been shown to be more robust against data imbalances [10]. However, these models are generally not trained on biological images, rather training on many general images, and thus are not thought to be easily transferable to biological applications due to a lack of domain knowledge. However, since self-supervised models learn general embeddings about their inputs, these large, pretrained, self-supervised models offer a prime opportunity to employ and validate on biological imaging tasks, even without specific domain knowledge.

1.1. Approach

In this project, we focus on T-cell images. T-cells help the body fight off infection by recognizing and killing infected cells. More recently, T-cell based immunotherapies have made a meteoric rise within the space of genetically engineered T-cells to fight cancer [17]. One challenge with these immunotherapies is that it's difficult to detect "active" T-cells, T-cells capable of killing infected cells, without expensive and destructive techniques. Thus, methods of detecting T-cell activation state via imaging alone are necessary to avoid difficult and expensive assays as well as avoid damaging or exhausting the T-cells.

Our goal is to leverage the strong foundational knowledge of large, pretrained, self-supervised models to develop a self-supervised computer vision model which classifies T-cells as active or quiescent (non-active). While existing supervised models have shown promising results, a self-supervised model could cut down the labor of generating annotated data and also leverage the power of existing self-supervised foundation models. In particular, we will leverage several of Meta AI's pretrained DINOv2 models, which are implemented with state-of-the-art vision transformers using a knowledge distillation approach [14]. We then evaluate the performance of the models on the task of binary T-cell classification.

Based on our experiments, we find that DINOv2 models can achieve comparable performance to supervised CNN models when tasked with classifying the activation state of T-cells, even without further fine-tuning. Within the different DINOv2 architectures, we see that models with less parameters and that use additional register tokens best capture the features of T-cell autofluorescent images. However, we ultimately determine that, given the size of our training set, the best model for classifying T-cell activation states is a fine-tuned supervised CNN model.

2. Related Works

2.1. T-cell Immunotherapies

T-cells are integral for identifying cancer-specific or overexpressed antigens, consequently destroying cancer cells. T-cell therapies have shown promising results for some cancers [17]. However, T-cell treatments still have several limitations, including toxicity, limited specificity, and longevity [20]. Additionally, characterizing and cultivating T-cells for immunotherapies is labor intensive and destructive. Often, characterizing T-cells requires fluorescent labeling, which can take several weeks, requires expensive antibodies and reagents, and requires destruction of the sample tissue. Finally, the quality of the final image is not guaranteed, as issues in the labeling procedure or in the imaging can affect the quality of the signal from the fluorescent labeling. This could make analysis of the images more difficult downstream, limiting the usability of the data. [18].

Recent research by Walsh et. al shows that T-cells have a natural fluorescence resulting from reduced nicotinamide adenine dinucleotide (NAD(P)H). They also demonstrate that NAD(P)H fluorescence is indicative of the T-cells' metabolic activity. Therefore, T-cells can be imaged without fluorescent labeling [18]. Autofluorescent imaging is an attractive alternative to traditional methods of fluorescent imaging, as it doesn't require tissue fixation or external agents and offers higher contrast. Using autofluorescent imaging to characterize T-cells is drastically less intensive than traditional methods, and potentially streamlines the T-cell immunotherapy research process.

2.2. Self-Supervised learning

Computer vision techniques have shown great promise in extracting relevant features from autofluorescent T-cell images. Using the same dataset used in Walsh et. al [18], Wang et. al demonstrated the robust performance of Convolutional Neural Networks (CNNs) on classifying the activation state of T-cells in autofluorescent images [19]. However, the models used in the study are supervised, and therefore require labeled data for training.

To generate the data, blood samples were collected from six patients. From these samples, equal amounts of activated and quiescent T-cells were cultivated [18]. This process highlights a significant challenge in using computer vision models for biological data: the labor-intensive nature of generating labeled data. Expert intervention is crucial to ensure accurate labeling, which is essential not only for the reliability of the research but also for the potential downstream therapeutic applications. Self-supervised models can mitigate the issues posed by supervised models.

Self-supervised representation learning (SSRL) models are pre-trained on pretext tasks to learn robust feature representations, which are then fine-tuned for specific tasks, such as classification, or tailored to particular domains, such as biological imaging. SSRL models have been successful in traditional image and speech tasks, and are the foundations of popular models such as Google's BERT and OpenAI's GPT-3. SSRL models have also seen success in the biological domain. For example, GAN-DL is a Generative Adversarial Network (GAN) that could distinguish between untreated SARS-Cov2 infected cells, treated SARS-Cov2 infected cells, and uninfected cells with unlabeled data [12]. Similarly, CytoGAN is a GAN that could generate realistic synthetic fluorescent images of cells and also learn representations of cells comparable to CellProfiler features, the current field norm [6]. More recent models leverage vision transformer models (ViT). For example, ChannelViT builds an explicit channel embedding, making it more apt to represent the distinct information found in each channel of fluorescent images[1]. Similarly, scDINO builds on Meta's DINO architecture by fine-tuning on non-RGB multichannel images, which are more characteristic of fluorescent images [15].

In this project, we expand on research done by Wang and assess whether SSRL models can achieve similar or better performance than supervised models in characterizing and classifying T-cells as active or quiescent. We specifically assess the performance of Meta's DINOv2 model, a self-supervised model trained on ImageNet [2]. We hypothesize that self-supervised learning can further improve the efficiency and effectiveness of working with autofluorescence data.

3. Data

Our dataset, generated by Walsh et. al[18], consists of labeled autofluorescence microscopy images of individual T-cells from 6 individual patients.

3.1. Data Preparation

To prepare the images for training and testing models, we followed the image processing pipeline developed by Wang et al.[19], as described below.

First the data is made uniform via padding – we pad smaller images with black border pixels as the T-cells tend to appear towards the center of each image. Then, we filter out images that are too dim or have no cell visible. To do so, we employ an entropy based filtering method where we calculate the entropy for each image, create a normal distribution of entropies in activated and non-activated images, define an entropy threshold for images that are too dim based on these distributions, then filter out all images below these thresholds. Additionally, we filter out any images containing more than one cell using a binary threshold and connected components to detect multiple cells in a single image. Finally, we augment our dataset by creating images that are rotated by 90, 180, or 270 degrees, as well as images that are flipped over a horizontal or vertical mid-line. Through this image augmentation and filtering pipeline, we end up with 4986 $134 \times 134 \times 1$ images – 1758 activated and 3228 quiescent.

For images run through the DINOv2 models, we additionally add further zero padding to each input image such that each image is 140 x 140 x 1 and converted the images from grayscale to RGB as these models require requires the height and width of input images to be divisible by 14 and for images to be RGB.

4. Methods

In this section, we define our baseline methods and report their performance in Table 1. We then go on to introduce our approach of evaluating various pretrained DINOv2 models, which are self-supervised vision tranformers (ViT), on our dataset and report their performance in Table 2 and Table 3.

4.1. Baseline

Following research from Wang et. al [19], we use the following classifiers as baselines: a frequency classifier, a logistic regression classifier, a one-layer fully-connected neural network, LeNet CNN, an out-of-the-box pre-trained CNN, and a fine-tuned CNN, whose accuracy, average precision, and AUC can be seen in Table 1. The features were derived from information grayscale pixel values in the raw images [19]. More information about the baselines and other models evaluated in [19] can be found below:

Frequency Classifier: a classifier which uses the frequency of positive samples in the training set as a predictive probability that a sample in the testing set is positive.

Logistic Regression Classifier: a logistic regression classifier using the features derived from the raw images. Trained using L_1 loss and hyperparameters are tuned using nested-cross validation.

One-layer Fully-connected Neural Network: A simple neural network with one fully-connected hidden layer to learn a non-linear relationship between images and labels. Hyperparameters are tuned similar to the logistic regression classifier.

LeNet CNN: A convolutional neural network (CNN) which consists of a `Conv2D`, followed by a `MaxPool2D`, repeated twice, and followed up by two `Dense` layers. Hyperparameters are tuned similar to the logistic regression classifier.

Pre-trained CNN: An off-the-shelf Inception v3 CNN architecture that has been pretrained on (non-biological) images with an additional fully-connected layer to map the features extracted from the CNN to our classes. Hyperparameters are tuned similar to logisitc regression classifier.

Fine-tuned CNN: The same off-the-shelf Inception v3 CNN model from above, but instead of a single additional fully-connected layer, multiple layers of the off-the-shelf model are finetuned on biological images. Note that the number of layers finetuned is a hyperparameter. Other hyperparameters are tuned similar to logisitc regression classifier.

The use of these specific baselines will offer us a broad reference to be able to compare our model results to random, general machine learning, and convolutional techniques offering a wide range of different approaches to solve this problem.

4.2. Model

4.2.1 DINOv2

DINOv2 was developed in 2024 by Meta AI and extends from their previous DINO approach, while also taking inspiration from iBOT (Image BERT Pretraining with Online Tokenization) loss and the centering of SwAV (Swapping Assignments between multiple Views of the same image)[14]. Like DINO, DINOv2 uses vision transformers for self-supervised learning instead of CNNs [3]. However, DINOv2 is trained on a significantly larger curated dataset; DINO was originally trained on 1.2M images from Google Landmarks v2 (GLDv2) and DINOv2 is trained on a curated 142M image dataset (LVD-142M) [3, 14].

Similar to DINO, DINOv2 uses knowledge distillation between a student and teacher network in which the student network is trained to match the output of the teacher network [3]. For a given image, the student network receives local ”views”, or differing crops of a given image, which contain $< 50\%$ of the original area of the image and the teacher network receives global views, which contain $> 50\%$ of the original area of the image. The output for each network is a probability distribution, p_t and p_s for teacher and student, respectively, and the DINO loss term

(1) is minimized to learn the parameters of the student – note that the DINO loss term is the cross entropy between the teacher and student probability distributions, measuring the difference between the two distributions. The teacher's parameters are learned by an exponential moving average of the student parameters.

$$\mathcal{L}_{\text{DINO}} = -\sum p_t \log p_s. \tag{1}$$

Where DINOv2 differs from DINO is the inclusion of the ”patch-level objective”, where the input image is divided into patches, and some patches are randomly masked in the student, but not in the teacher [14], which improves patch-level tasks. The iBOT loss term (2) is minimized similar to the above approach, except we are now calculating the cross entropy of the probability distributions per patch. The inclusion of the patch-level objective ultimately improves patch-level tasks since it trains the model to reconstruct a set of input tokens through knowledge distillation [14, 21].

$$\mathcal{L}_{\text{iBOT}} = -\sum_i p_{ti} \log p_{si} \mid \text{where } i \text{ is a patch.} \tag{2}$$

Moreover, DINOv2 introduces Sinkhorn Snopp (SV) centering, a KoLeo regularizer (to improve nearest-neighbor search tasks), adaptation of image resolution (to improve segmentation and detection on small objects), and numerous efficient adaptations to improve the performance of the DINOv2 architecture over the original DINO architecture, while simultaneously reducing the computational requirements [14].

4.2.2 Approach

For our approach, we ran images through either the small (S, 21M parameters) or big (B, 86M parameters) DINOv2 models, with or without registers (details provided in later sections). Then, we took the concatenated class and patch tokens (and additionally the register tokens if the model had registers), and ran it through a single linear layer from an embedding dimension of 384 (S) or 768 (B) to a single logit followed by a sigmoid function. As our task was that of binary classification, we used Binary Cross Entropy as our loss function. For test time prediction, scores that were $<= 0.5$ were classified as 0 and all others were classified as 1. For training, we tried both freezing (preventing the pre-trained model weights from updating) and fine-tuning the DINOv2 layer. We attempted some hyperparameter tuning (i.e. learning rate, batch size) but due to the training time of our model we were not able to do a full cross-validation scheme.

This approach is both suitable and the most applicable for our problem as we want to test the ability for general self-supervised models to be applied to biological image applications. As DINOv2 is one of the most novel and broadly trained self-supervised models, we believe our results will be both relevant and applicable. We considered alternative approaches such as developing our own self-supervised model for this task, but decided against doing so due to our relatively small dataset.

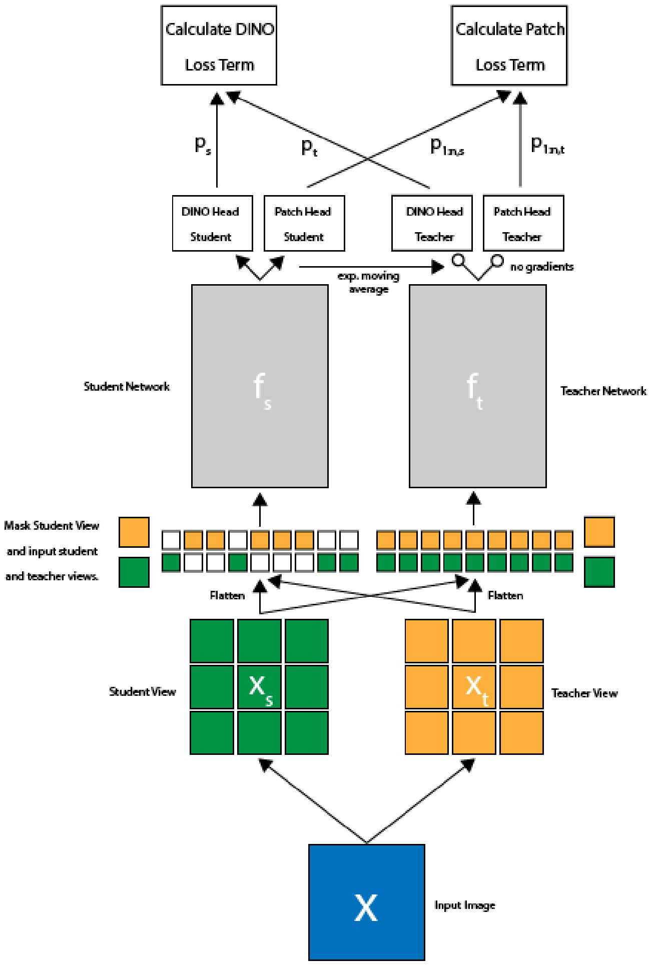


Figure 1: A simplified DINO Architecture based off descriptions of DINOv2 in [14] and iBOT in [21]. First, an input image is sampled and divided into a student (local) and teacher (global) view. Both views are split into patches – with some patches of the student view being masked – and the patches and views themselves are passed into their respective networks. Once passed through, the DINO head uses Equation 1 to update the parameters of the student and teacher network, and the Patch head uses Equation 2 to update the parameters of the student and teacher network.

5. Results

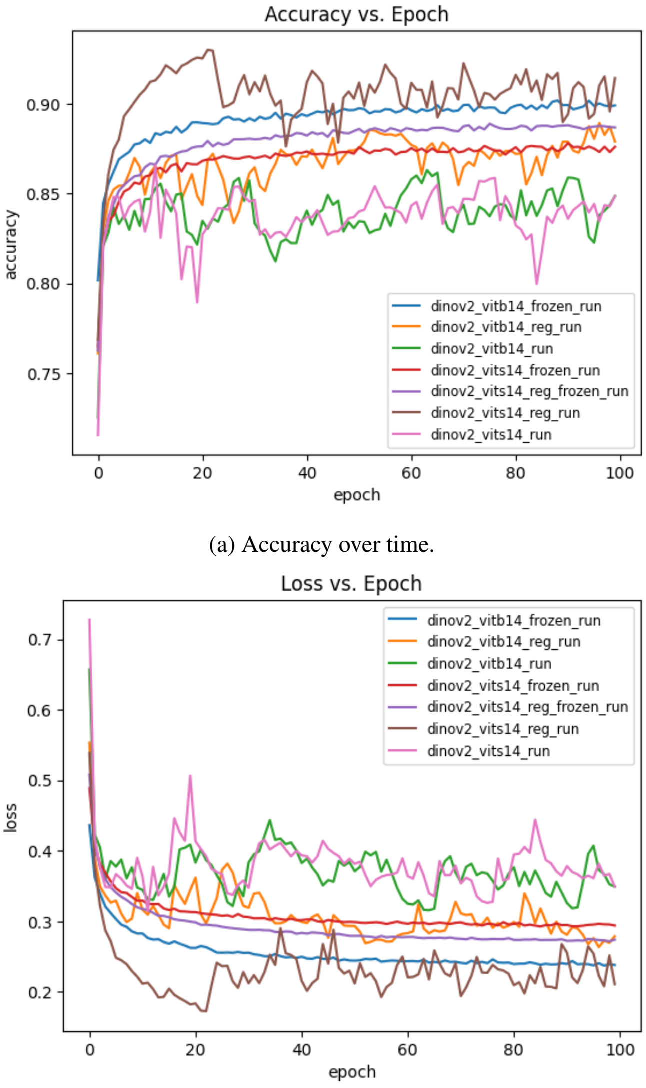
In this section, we present the evaluation of our combined model of various available pretrained DINOv2 models and a classification head on validation and testing sets – located in Table 2 and Table 3, respectively. We evaluate both small (-S, 21M parameters) and big (-B, 86M parameters) models. Each model is either labeled as ”Frozen,” meaning the model weights tuned during pretraining have been prevented from updating, ”Reg,” meaning the models have been trained with registers, ”Reg Frozen,” a combination of the two above, or unlabeled, meaning that the model was allowed to update its weights as it trained on the autofluorescent T-cell images as described in the Methods section.

	Accuracy	Precision	AUC
Frequency Classifier	.5256	.4744	.5000
Logistic Regression	.7808	.8039	.7183
Simple Neural Net	.8006	.8150	.5000
LeNet CNN	.8853	.8842	.9334
Pre-trained CNN	.9036	.9480	.9596
Fine-tuned CNN	**.9356**	**.9638**	**.9667**

Table 1: Accuracy, Average Precision, and AUC accross the baseline models and evaluated models mentioned in Wang et. al [19], averaged across patients.

	Accuracy	Precision	AUC
ViT-S Frozen	0.8718	0.8703	0.8544
ViT-S Reg Frozen	0.8735	0.9019	0.8682
ViT-B Frozen	0.8776	0.9073	0.8732
ViT-S	0.8640	0.8659	0.8468
ViT-B	0.8577	0.8523	0.8362
ViT-S Reg	**0.9188**	**0.9341**	**0.9143**
ViT-B Reg	0.8864	0.8855	0.8758

Table 2: Validation accuracy, precision, and AUC across the DINOv2 models, with best performing models bolded. Metrics pulled from best performing training epoch.



(a) Accuracy over time.

(b) Loss over time.

Figure 2: Training Performance over time for each DINOv2 model.

	Accuracy	Precision	AUC
ViT-S Frozen	0.8597	0.8687	0.8496
ViT-S Reg Frozen	**0.9216**	**0.9251**	**0.9158**
ViT-B Frozen	0.8542	0.8542	0.8403
ViT-S	0.8572	0.8494	0.8408
ViT-B	0.8542	0.8542	0.8403
ViT-S Reg	**0.9216**	**0.9251**	**0.9158**
ViT-B Reg	0.8864	0.8855	0.8758

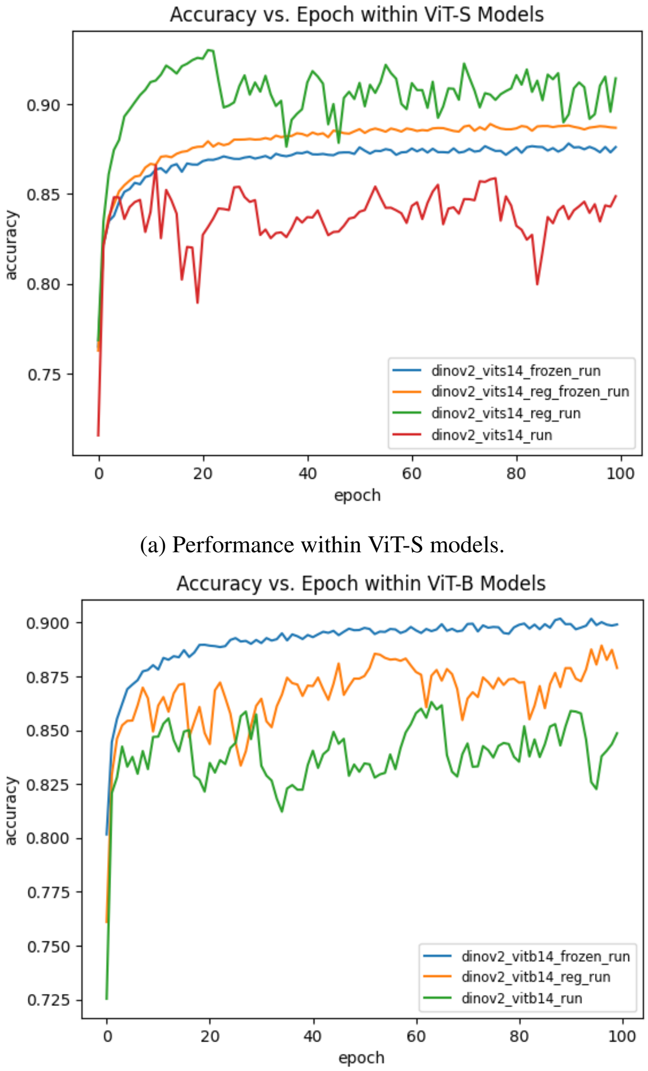
Table 3: Test accuracy, precision, and AUC across the DINOv2 models, with best performing models bolded.

5.1. Evaluation of Model Size

In Table 2 and Table 3, we compare the test performance ViT-S and ViT-B models with the various model setting described above. First, we observe that ViT-S models seem to perform better than their ViT-B counterparts, or models with a lower number of parameters are observed to perform better than those with a higher number of parameters, with the exception of ViT-B Frozen compared to ViT-S Frozen and ViT-S Reg Frozen in Table 2. This is consistent with the training performance as seen in Figure 2a and Figure 2b– the ViT-S models outperform their ViT-B counterparts during training. Here, in the case of Table 3, we find the difference between the best ViT-S models – ViT-S Reg Frozen and ViT-S Reg – and the best ViT-B model is 0.0352. This pattern is indicative of overfitting and consistent with existing literature given the combination of the ViT-B model's larger parameter size and the relatively small size of our training dataset [13]

5.2. Evaluation of Model Weight Freezing

As show in Figure 3, freezing model weights generally leads to better performance than fine-tuning the models. Additionally, the performance of the frozen models were more stable as indicated by the smoother learning curves seen in both model sizes. In other words, fine-tuning DINOv2 models on our specific training set did not necessarily lead to better performance. This is especially clear within the ViT-B models – the fine-tuned weights likely overfit on the data and led to worse performance than the frozen weights. Interestingly, this pattern is not as clear with the ViT-S models. The fine-tuned ViT-S model with registers outperformed its frozen counterpart during training. This is likely because the ViT-S model is less expressive than the ViT-B model due to its smaller parameter size and therefore less prone to overfitting. This, combined with the benefits of including registers, might actually cause the ViT-S model to benefit from fine-tuning. However, in validation (Table 2), the fine-tuned model outperformed the frozen model, and both models performed equivalently during testing (Table 3).



(a) Performance within ViT-S models.

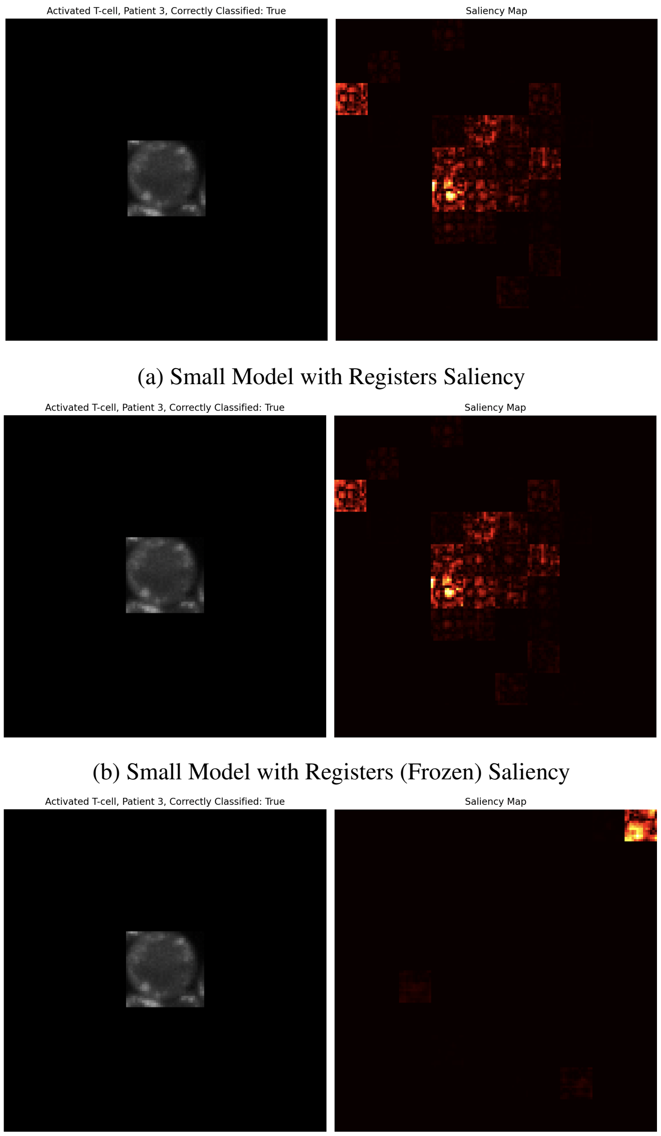
(b) Performance within ViT-B models.

Figure 3: Training Performance split by Model Size.

5.3. Evaluation of Model Saliency

To understand the regions of input images that our models were giving weight to, we generated saliency maps for correctly and incorrectly classified images for each model. Saliency maps are generated by running images through the model forward pass, and then using a backward pass to generate heat maps of important/sensitive regions from the input image.

In Figure 4, we see three saliency maps for the same correctly classified activated T-cell from the small model with registers, small model with registers frozen, and the big model. In (a) and (b) we see that the models correctly focus on the most important region of the image (where the cell is located) and that freezing the DINOv2 layer of the model does not impact the region of the input image that it gives the most weight to indicating that the additional fine-tuning may not have contributed significantly to model performance. In (c), we see that the big model correctly classified the same input image but its focus is nowhere near the cell. This trend was shared in the saliency maps of for many of the images for the big model which could help explain its decreased performance relative to the small model. However, as its performance was still fairly good, this leads us to wonder if there is additional information being conveyed in the images in regions where there is no cell visible (i.e. some sort of bias in the images that's allowing for inflated


Activated T-cell, Patient 3, Correctly Classified: True
Saliency Map
Activated T-cell, Patient 3, Correctly Classified: True
Saliency Map
Activated T-cell, Patient 3, Correctly Classified: True
Saliency Map

(a) Small Model with Registers Saliency

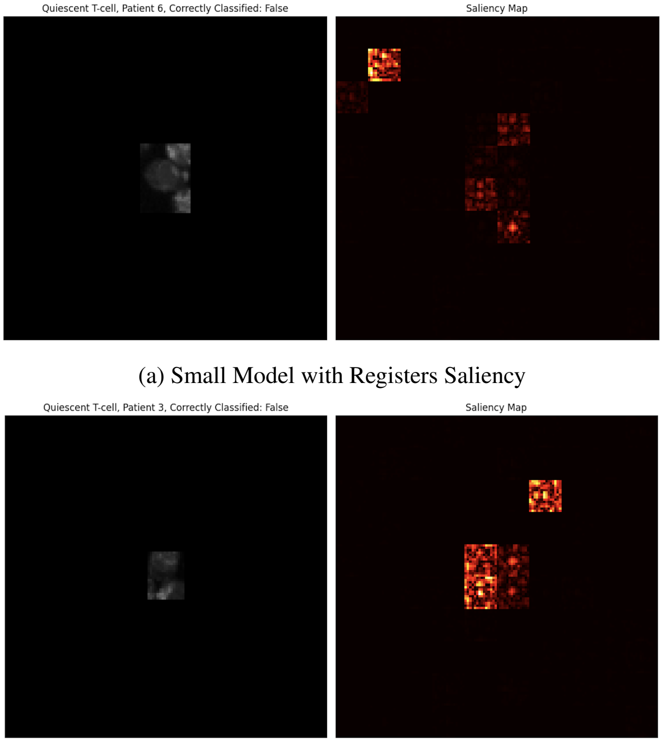
(b) Small Model with Registers (Frozen) Saliency

(c) Big Model Saliency

Figure 4: Saliency Maps for a correctly classified cell across three models.

performance by the model).

In Figure 5, we see two saliency maps for two different incorrectly classified T-cells from the small model with registers. In (a) we see that the model was not able to correctly identify the most important region from the input image which likely contributed to the incorrect classification. However, this observation was not uniform across the incorrect classifications as in (b) the model was still able to focus on the correct part of the input image despite its incorrect classification.


Quiescent T-cell, Patient 6, Correctly Classified: False
Saliency Map
Quiescent T-cell, Patient 3, Correctly Classified: False
Saliency Map

(a) Small Model with Registers Saliency

(b) Small Model with Registers Saliency

Figure 5: Saliency Maps for an incorrectly classified cell by the small model with registers showing correct focus (a) and less focused (b).

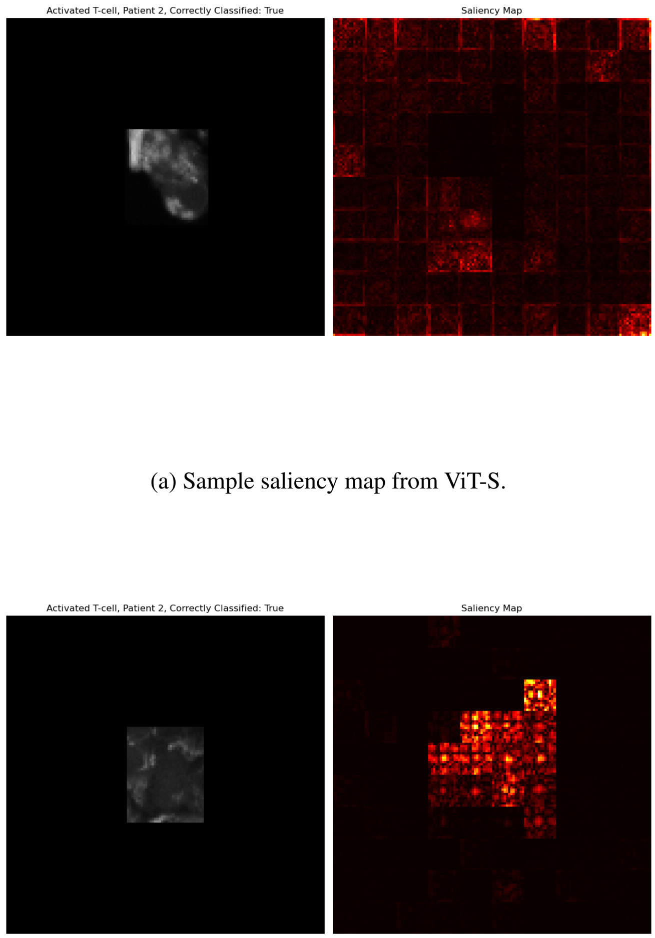
5.4. Evaluation of Registers

Across both model sizes, models with registers outperform models without registers during training(Figure 3), validation (Table 2) and testing Table 3. The models without registers seem to be assigning importance to low-signal patches as seen Figure 6a. This is indicative of a known problem with DINOv2 – the model attempts to use tokens associated with the low-signal patches to store global information, consequently losing the local information associated with these patches [4].

By introducing additional tokens that are discarded at the end of the vision transformer, called registers, the model can learn to aggregate global information in the registers and preserve the local information in the low-signal patches. This leads to better learning and empirically better performance [4]. The benefits of including registers can be seen in Figure 6b, which show that the model is correctly focusing more on T-Cell.

5.5. Comparison with Baselines

Based on Table 1 and Table 3, the best performing SSRL model (ViT-S Reg) outperforms the Pre-trained CNN and


Activated T-cell, Patient 2, Correctly Classified: True
Saliency Map
Activated T-cell, Patient 2, Correctly Classified: True
Saliency Map

(a) Sample saliency map from ViT-S.

(b) Sample saliency map from ViT-S with registers.

Figure 6: Comparison of saliency between small model with registers and small model without registers.

achieves comparable performance to the fine-tuned CNN. However, ultimately, the Fine-tuned CNN is the best performing model among both the baseline models and the SSRL models. This is likely due to the small size of the dataset. Transformers, the foundation of the DINOv2 models, inherently require a lot of data to learn robust representations. While the DinoV2 models already have rich representations of image features from the pre-training process, there likely wasn't enough data for the models to learn features specific to T-cell images.

Interestingly, the best performing frozen SSRL model (ViT-S Reg Frozen) outperforms the Pre-trained CNN. This indicates that, when comparing out-of-the-box performance, DINOv2 is better at predicting T-cell activation state than a pre-trained CNN. More specifically, the weights learned in the pre-training procedure of the DINOv2 model are better representing the T-Cell image features than that of the pre-trained CNN. Overall, with the limited data, fine-tuning a CNN is a better choice than fine-tuning an SSRL ViT model, because it is both more accurate and less computationally intensive [9]. Additionally, using DINOv2 out of the box would lead to better performance than attempting to fine-tune it. However, further research is needed to determine if this trend holds with a larger training set.

5.6. Evaluation of Effect of Class Imbalance

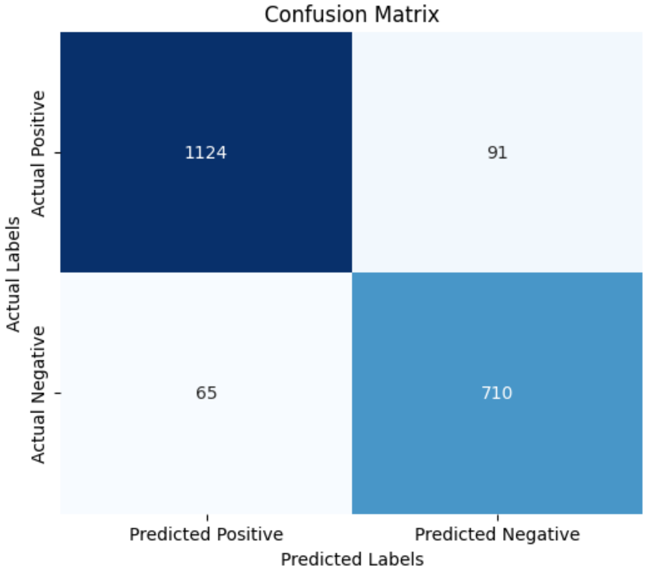


Figure 7: Confusion matrix based on test performance of VIT-S Reg model.

As mentioned in the description of our dataset, we have almost twice as many quiescent T-cell images as we do activated T-cell images. Despite this class imbalance, the best performing model (ViT-S) is able to distinguish between activated and quiescent T-cells – this is demonstrated by the relatively low number of false positives and false negatives in Figure 7. This indicates that, despite performing less well than the baselines, the DINOv2 model is still good at learning and recognizing the distinguishing features of active and quiescent cells.

6. Conclusion

In this study, we evaluate the performance of an SSRL ViT model architecture, DINOv2, with traditional supervised CNNs. We specifically focus on the task of classifying

T-Cell activation based on T-cell autofluorescence images, which have been shown to be an important task for immunotherapy development. We build upon previous work that demonstrates that supervised CNNs do well at classifying T-Cell activation states.

Based on our experiments, we find that DINOv2 can achieve similar performance as compared CNNs. We also see that, if comparing out-of-the-box performance of pre-trained models, DINOv2 performs better than supervised CNN at classifying the activation state of T-cells. However, ultimately, we see that a supervised, fine-tuned CNN is a preferable choice to a fine-tuned DINOv2 model due to both having better performance and requiring less computational resources. This is likely due to the amount of data needed to fine-tune a DINOv2 model, and the comparably small size of our training set.

Outside of direct comparisons between the DINOv2 architecture and the supervised CNN architecture, we also observed other interesting patterns. Consistent with other DINOv2 experiments, we find that using register tokens leads to improved performance and a better understanding of T-cell images. We also find that, with a smaller training set, a smaller model is more impervious to overfitting than a larger model. Overall, we demonstrate that DINOv2 can achieve good performance with biological images without further fine-tuning, despite the fact that it is trained on non-biological images. Using an a SSRL ViT model, DINOv2, out of the box allows for a reduced dependence on data – as we wouldn't need to fine-tune – without sacrificing performance. However, it is not necessarily the best possible model for classifying T-Cell activation based on T-cell auotfluorescent images.

6.1. Further Research

Given more time and resources, there are several other further research topics we could pursue. In this project, we decided to compare a singular ViT SSRL architecture, DINOv2, to a supervised CNN. It would be intersting to evaluate the performance of different SSRL architectures. Specifically, we could compare the performance of a GAN model, a ViT model, and a self-supervised CNN model.

Another interesting research direction could be to compare the performance of a pre-trained SSRL model that has seen biological images in its pre-training process. We mention several examples in the introduction, such as CytoGAN and scDINO – these models may be better at analyzing T-cell images than the out-of-the box DINOv2 architecture.

Finally, we can try different ways of pre-processing our data before feeding it into the models. In this project, we relied on the same image-processing pipeline used for the baseline models. However, we can try using a custom image processing and image augmentation pipeline, and possibly find better ways to artificially enlarge the training set.

7. Supplementary Materials

The code for our project can be found here. The original code for the baseline models and for image processing can be found here.

References

[1] Y. Bao, S. Sivanandan, and T. Karaletsos. Channel vision transformers: An image is worth c x 16 x 16 words. *arXiv preprint arXiv:2309.16108*, 2023.

[2] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin. Emerging properties in self-supervised vision transformers. 2021.

[3] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin. Emerging properties in self-supervised vision transformers, 2021.

[4] T. Darcet, M. Oquab, J. Mairal, and P. Bojanowski. Vision transformers need registers, 2024.

[5] L. Gao, L. Zhang, C. Liu, and S. Wu. Handling imbalanced medical image data: A deep-learning-based one-class classification approach. *Artificial Intelligence in Medicine*, 108:101935, 2020.

[6] P. Goldsborough, N. Pawlowski, J. C. Caicedo, S. Singh, and A. E. Carpenter. Cytogan: Generative modeling of cell images. *bioRxiv*, 2017.

[7] X. Han, Z. Zhang, N. Ding, Y. Gu, X. Liu, Y. Huo, J. Qiu, Y. Yao, A. Zhang, L. Zhang, W. Han, M. Huang, Q. Jin, Y. Lan, Y. Liu, Z. Liu, Z. Lu, X. Qiu, R. Song, J. Tang, J.-R. Wen, J. Yuan, W. X. Zhao, and J. Zhu. Pre-trained models: Past, present and future. *AI Open*, 2:225–250, 2021.

[8] S. Huang, A. Pareek, and M. Jensen. Self-supervised learning for medical image classification: a systematic review and implementation guidelines, 2023.

[9] T. Huang, T. Chen, Z. Wang, and S. Liu. The counterattack of cnns in self-supervised learning: Larger kernel size might be all you need, 2023.

[10] H. Liu, J. Z. HaoChen, A. Gaidon, and T. Ma. Self-supervised learning is more robust to dataset imbalance, 2022.

[11] V. Marx. The big challenges of big data. *Nature*, 2013.

[12] A. Mascolini, D. Cardamone, F. Ponzio, S. Di Cataldo, and E. Ficarra. Exploiting generative self-supervised learning for the assessment of biological images with lack of annotations. *BMC Bioinformatics*, 23(1):295, 2022.

[13] O. A. Montesinos López, A. Montesinos López, and J. Crossa. *Overfitting, Model Tuning, and Evaluation of Prediction Performance*, pages 109–139. Springer International Publishing, Cham, 2022.

[14] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, M. Assran, N. Ballas, W. Galuba, R. Howes, P.-Y. Huang, S.-W. Li, I. Misra, M. Rabbat, V. Sharma, G. Synnaeve, H. Xu, H. Jegou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski. Dinov2: Learning robust visual features without supervision, 2024.

[15] R. Pfaendler, J. Hanimann, S. Lee, and B. Snijder. *Self-supervised vision transformers accurately decode cellular state heterogeneity*, Jan 2023.

[16] N.-T. Tran, V.-H. Tran, N.-B. Nguyen, L. Yang, and N.-M. Cheung. Self-supervised gan: Analysis and improvement with multi-class minimax game, 2020.

[17] A. D. Waldman, J. M. Fritz, and M. J. Lenardo. A guide to cancer immunotherapy: From t cell basic science to clinical practice. *Nature Reviews Immunology*, 20(11):651–668, May 2020.

[18] A. J. Walsh, K. P. Mueller, K. Tweed, I. Jones, C. M. Walsh, N. J. Piscopo, N. M. Niemi, D. J. Pagliarini, K. Saha, and M. C. Skala. Classification of t-cell activation via autofluorescence lifetime imaging. *Nature Biomedical Engineering*, 5(1):77–88, Jul 2020.

[19] Z. J. Wang, A. J. Walsh, M. C. Skala, and A. Gitter. Classifying t cell activity in autofluorescence intensity images with convolutional neural networks. *Journal of Biophotonics*, 13(3), Dec 2019.

[20] M. Y. Want, Z. Bashir, and R. A. Najar. T Cell Based Immunotherapy for Cancer: Approaches and Strategies. *Vaccines*, 11(4):835, Apr. 2023.

[21] J. Zhou, C. Wei, H. Wang, W. Shen, C. Xie, A. Yuille, and T. Kong. ibot: Image bert pre-training with online tokenizer, 2022.