Visualizing Attention in Vision Transformers

Aayush Agrawal, Finn Dayton, Elliot Dauber

Computer Science, Stanford University

Introduction

- Vision Transformers (ViT) have become comparable to Convolutional Neural Networks (CNNs) in performance.
- The central component of a ViT is **Multi-headed Self-Attention**, but the inner workings of attention are intricate and not fully understood.
- We present a **novel approach** for **visualizing attention** using "saliency maps" to identify patterns and correlations that can **improve ViT design choices** and enhance the **understanding of ViTs**.
- We operate in a **data sparse environment** to contrast the robustness of different model architectures.

Problem Statement

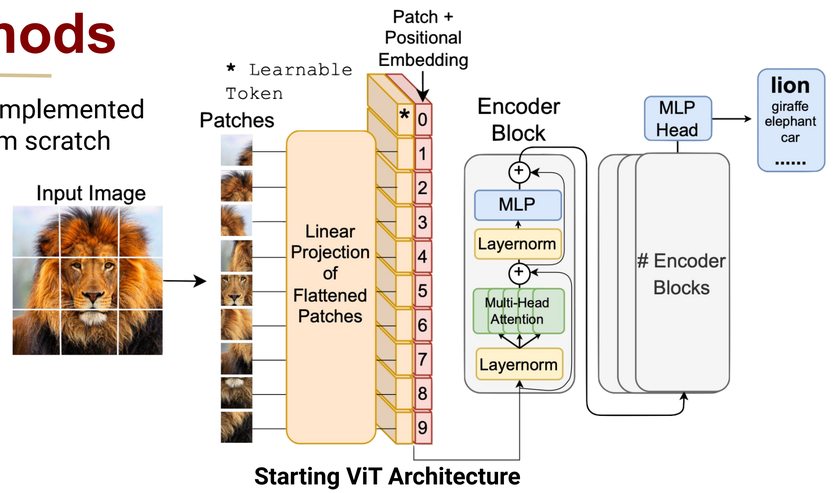
Our problem is **image classification**. The inputs are 64x64 images and the outputs are class labels. We use the conventional quantitative metrics of accuracy, recall, precision, F1 Score and AUC ROC. Additionally, we introduce a novel quantitative metric, "Saliency-map Mask Metric" (**SMM**), explained in "Methods."

Dataset

- We use the **Tiny Imagenet dataset** with 10 classes sampled out of 1000.
- **Only 500 images per class** for training, and 100 per class for testing.
- The dataset contains **challenging classes** that are visually similar, making classification and visualizations more interesting.

Methods

We first implemented a ViT from scratch



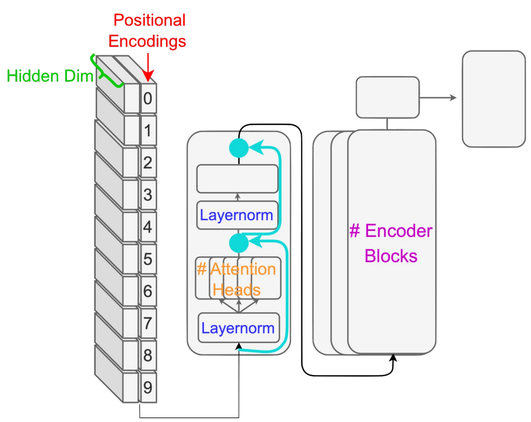
Starting ViT Architecture

Architecture Modifications

We then trained a *series* of models on our dataset, making one architectural change from a baseline for each model. The **six architectural changes** are as follows:

1. Positional Encoding ∈ {**Sin/Cos**, Learned, Integer}
2. # Encoder Blocks ∈ {1, 2, **4**, 8}
3. # Attention Heads ∈ {1, 2, **4**, 8}
4. Size of Hidden Dim ∈ {4, **8**, 16, 32}
5. Presence of Residual Connection ∈ {**Yes** / No}
6. Presence of Layernorm ∈ {**Yes** / No}

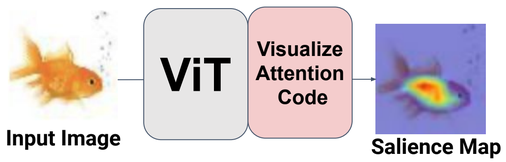
The baseline model choices are **bolded*



ViT Architectural Changes

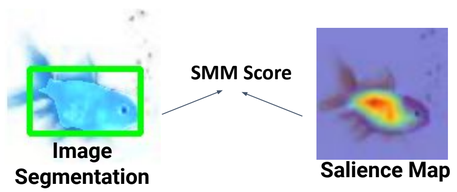
Attention Visualization

To **visualize attention,** we combine the self-attention matrices then multiply by the gradient of the intended class. We then produce a heat map of the resulting matrix:



SMM, a Novel Metric!

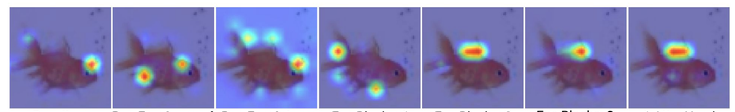
Saliency-map Mask Metric (SMM) quantitatively analyzes a saliency map by comparing it with an **object instance segmentation mask** (Fast R-CNN) to measure the **overlap** of the expected object and attended areas. The result is 0.0 - 1.0.

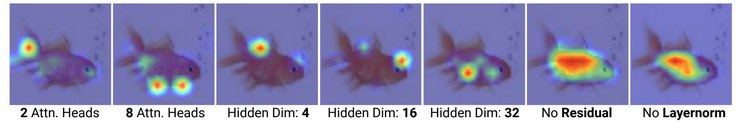


Results and Analysis

Architecture	Train Acc	Test Acc	Recall	Prec	F1 Score	ROC AUC	SMM
Baseline	0.476	0.328	0.317	0.367	0.312	0.766	0.025
Pos-Enc: Learned	**0.536**	**0.390**	0.365	**0.384**	**0.376**	**0.802**	0.065
Pos-Enc: Integer	0.476	0.318	0.302	0.325	0.304	0.769	0.026
Encoder Blocks: 1	0.429	0.336	0.332	0.348	0.325	0.747	0.009
Encoder Blocks: 2	0.475	0.320	0.309	0.323	0.306	0.751	0.039
Encoder Blocks: 8	0.374	0.310	0.309	0.279	0.280	0.739	0.017
Attention Heads: 1	0.475	0.352	0.341	0.337	0.324	0.771	0.041
Attention Heads: 2	0.460	0.372	**0.370**	0.348	0.335	0.771	0.001
Attention Heads: 8	0.523	0.340	0.321	0.373	0.337	0.777	0.029
Hid-Dem: 4	0.388	0.346	0.342	0.311	0.310	0.735	0.021
Hid-Dem: 16	0.510	0.374	0.369	0.356	0.348	0.789	0.026
Hid-Dem: 32	0.100	0.100	0.100	0.010	0.018	0.502	0.017
No Residuals	0.100	0.100	0.100	0.010	0.018	0.500	0.297
No Layernorm	0.100	0.100	0.100	0.010	0.018	0.500	**0.482**

Experiment Results





Attention Visualizations

- Learned Positional Encodings created the **biggest gain**.
- The number of encoding blocks and attention heads had **marginal effect** on performance.
- Changing the hidden dim size drastically affected performance.
- Residuals + Layernorm are absolutely critical!
- The SMM metric tends to be higher for the **worse models** (successful models focus on small, differentiating details in images).

Conclusion + Future Work

- Better models focused on **fewer, smaller details** of the fish – easily quantified with our novel SMM metric.
- In the future, we hope to run **more ablations** on different parts of the ViT.
- We are also interested in collecting 10-100x more training images and re-running all experiments in this non-sparse environment. Do the optimal model configurations change in this new data regime?