Intriguing Properties of Vision Transformers

Muzammal Naseer  Kanchana Ranasinghe  Salman Khan  Munawar Hayat
Fahad Shahbaz Khan  Ming-Hsuan Yang

Kishore Alajangi  James Dwyer  Michael Fielder  Joseph Rivera
Introduction

- Compare Vision Transformers to CNN
  - ViT
  - DeiT
  - T2T

- Experiments
  - Image occlusions - robust
  - Modeling shape - performance comparable to humans
  - Positional Encoding and Preservation of Global Image Context - positional encoding not as important
  - Adversarial and Natural Perturbations - robust in most cases
  - Effective Off-the-Shelf Tokens - generalizable for downstream tasks
Are Vision Transformers Robust to Occlusions?
Are Vision Transformers Robust to Occlusions?

- Occlusion Modeling - PatchDrop
  - Random PatchDrop - randomly selected subset of patches is dropped
  - Salient PatchDrop
    - DINO used to select foreground/important patches
    - Patches above a threshold are dropped
  - Non-Salient PatchDrop
    - Patches below a threshold are dropped
Are Vision Transformers Robust to Occlusions?

- All models pretrained on ImageNet
- Use ImageNet validation set (50k images)
- Random PatchDrop - mean accuracy of 5 runs
- Non-salient/salient - accuracy of 1 run
Are Vision Transformers Robust to Occlusions?
Are Vision Transformers Robust to Occlusions?

\[ \text{corr}(u, v) = \frac{\sum_i \hat{u}_i \hat{v}_i}{n} \]

\[ \hat{u}_i = \frac{u_i - E[u_i]}{\sigma(u_i)} \]

<table>
<thead>
<tr>
<th>Model</th>
<th>Correlation Coefficient: Random PatchDrop</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>25% Dropped</td>
</tr>
<tr>
<td>ResNet50</td>
<td>0.32±0.16</td>
</tr>
<tr>
<td>TnT-S</td>
<td>0.83±0.08</td>
</tr>
<tr>
<td>ViT-L</td>
<td>0.92±0.06</td>
</tr>
<tr>
<td>DeiT-B</td>
<td>0.90±0.06</td>
</tr>
<tr>
<td>T2T-24</td>
<td>0.80±0.10</td>
</tr>
</tbody>
</table>
Shape vs Texture: Can Transformer Model Both Characteristics?
Shape vs Texture

- ImageNet converted to Stylized ImageNet (SIN)
- Data augmentations not used

![Texture image](image1)
- Texture image
  - 81.4% Indian elephant
  - 10.3% indri
  - 8.2% black swan

![Content image](image2)
- Content image
  - 71.1% tabby cat
  - 17.3% grey fox
  - 3.3% Siamese cat

![Texture-shape cue conflict](image3)
- Texture-shape cue conflict
  - 63.9% Indian elephant
  - 26.4% indri
  - 9.6% black swan
Shape vs Texture
Shape vs Texture

![Graph showing Shape vs Texture](image)
Shape vs Texture

- Transformers biased towards shape
- Shape Distillation
  - ResNet50-SIN → teacher

<table>
<thead>
<tr>
<th>Model</th>
<th>Distilled</th>
<th>Token Type</th>
<th>ImageNet top-1 (%)</th>
<th>Shape Bias</th>
</tr>
</thead>
<tbody>
<tr>
<td>DeiT-T-SIN</td>
<td></td>
<td>cls</td>
<td>40.5</td>
<td>0.87</td>
</tr>
<tr>
<td>DeiT-T-SIN</td>
<td>✔️</td>
<td>cls</td>
<td>71.8</td>
<td>0.35</td>
</tr>
<tr>
<td>DeiT-T-SIN</td>
<td>✔️</td>
<td>shape</td>
<td>63.4</td>
<td>0.44</td>
</tr>
<tr>
<td>DeiT-S-SIN</td>
<td></td>
<td>cls</td>
<td>52.5</td>
<td>0.93</td>
</tr>
<tr>
<td>DeiT-S-SIN</td>
<td>✔️</td>
<td>cls</td>
<td>75.3</td>
<td>0.39</td>
</tr>
<tr>
<td>DeiT-S-SIN</td>
<td>✔️</td>
<td>shape</td>
<td>67.7</td>
<td>0.47</td>
</tr>
</tbody>
</table>
Shape vs Texture

<table>
<thead>
<tr>
<th>Model</th>
<th>Distilled</th>
<th>Token Type</th>
<th>Jaccard Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>DeiT-T-Random</td>
<td>✓</td>
<td>cls</td>
<td>19.6</td>
</tr>
<tr>
<td>DeiT-T</td>
<td>✓</td>
<td>cls</td>
<td>32.2</td>
</tr>
<tr>
<td>DeiT-T-SIN</td>
<td>✓</td>
<td>cls</td>
<td>29.4</td>
</tr>
<tr>
<td>DeiT-T-SIN</td>
<td>✓</td>
<td>cls</td>
<td>40.0</td>
</tr>
<tr>
<td>DeiT-T-SIN (Distilled)</td>
<td>✓</td>
<td>shape</td>
<td>42.1</td>
</tr>
<tr>
<td>DeiT-S-Random</td>
<td>✓</td>
<td>cls</td>
<td>22.0</td>
</tr>
<tr>
<td>DeiT-S</td>
<td>✓</td>
<td>cls</td>
<td>29.2</td>
</tr>
<tr>
<td>DeiT-S-SIN</td>
<td>✓</td>
<td>cls</td>
<td>37.5</td>
</tr>
<tr>
<td>DeiT-S-SIN</td>
<td>✓</td>
<td>cls</td>
<td>42.0</td>
</tr>
<tr>
<td>DeiT-S-SIN (Distilled)</td>
<td>✓</td>
<td>shape</td>
<td>42.4</td>
</tr>
</tbody>
</table>
Does Positional Encoding Preserve the Global Image Context?
Background

Patch + Position Embedding
* Extra learnable [class] embedding

Linear Projection of Flattened Patches
Effect of No. of Patches

Accuracy Top-1 (%)

Shuffle Grid Size

196
100
49
64
25
Conclusions

- ViT’s show high permutation invariance to patch positions
- Pos. encodings for ViT do not preserve global image context
- Dependence on positional encodings to perform well under occlusions is also incorrect
- ViTs robustness to occlusion a result of dynamic receptive field & patch size
Robustness of ViT to Adversarial and Natural Perturbations
Robustness to Natural Perturbations

- Are ViTs robust to common corruptions, like fog, rain, snow, and noise?
- ImageNet-C (Corruptions) is used as a baseline
- 15 common augmentation methods (seen right)
  - Noise, Blur, compression, etc.
- 5 levels of aggression in augmentation
Scoring Robustness to Natural Perturbations

Compute test error of a model $f$ on corruption method $C$ with severity $S$ e.g. ViT, $C=$Snow, $S=5$

$$E_{s,c}^f$$

Compute Corruption Error (CE) from all severities normalized

$$CE_c^f = \left( \frac{\sum_{s=1}^{5} E_{s,c}^f}{\sum_{s=1}^{5} E_{s,c}^{\text{AlexNet}}} \right)$$

Average scores across all corruptions to get mean ($mCE$)

$$CE_{\text{Gaussian Noise}}^f, CE_{\text{Shot Noise}}^f, \ldots, CE_{\text{JPEG}}^f$$
Robustness to Natural Perturbations

- mCE is a relative metric (lower is better)
- Most ViTs with data augmentations performed better than ResNet50 (Augmix)
- ViT-SIN models are more susceptible to natural distribution shifts, even compared to ResNet50
  - This supports a shape-bias vs robustness trade-off

<table>
<thead>
<tr>
<th>Trained with Augmentations</th>
<th>Trained without Augmentation</th>
</tr>
</thead>
<tbody>
<tr>
<td>DeiT-B</td>
<td>48.5</td>
</tr>
<tr>
<td>DeiT-S</td>
<td>54.6</td>
</tr>
<tr>
<td>DeiT-T</td>
<td>71.1</td>
</tr>
<tr>
<td>T2T-24</td>
<td>49.1</td>
</tr>
<tr>
<td>TnT-S</td>
<td>53.1</td>
</tr>
<tr>
<td>Augmix</td>
<td>65.3</td>
</tr>
<tr>
<td>ResNet50</td>
<td></td>
</tr>
<tr>
<td>ResNet50-SIN</td>
<td></td>
</tr>
<tr>
<td>DeiT-T-SIN</td>
<td></td>
</tr>
<tr>
<td>DeiT-S-SIN</td>
<td></td>
</tr>
</tbody>
</table>
Robustness to Adversarial Patch Attacks

- Adversarial Patches are taught to confidently confuse a classifier.
- Real-World Use Cases:
  - Highly dangerous to autonomous vehicles when road signs are used to understand speed limits, stop signs, etc.
  - Trick facial recognition software
Training an Adversarial Patch

Input

Dog: 1%
Cat: 99%

Dog: 99%
Cat: 1%
Robustness to Adversarial Attacks

- Adversarial Patch Attack shown to be less effective on ViT models than the baseline ResNet50
  - The change in pixel values is unbounded for this attack
- ResNet models underperform when trained on ImageNet
- ViTs-SIN underperform ResNet50-SIN however, supporting shape-bias vs robustness trade-off
Robustness to Adversarial Attacks - FGSM

- The change in pixel values is bounded for these attacks, and white-box (access to model gradients)
- Fast Gradient Sign Method (FGSM) uses the model to generate an image that maximizes loss with respect to the true class
- Epsilon is used to budget how much each pixel can be changed

![Graph showing accuracy top-1 (%) vs perturbation budget (ε) for different models using FGSM.]
Effective Off-the-shelf Tokens for Vision Transformer
<table>
<thead>
<tr>
<th>Blocks</th>
<th>Class Tokens</th>
<th>Patch Tokens</th>
<th>CUB [37]</th>
<th>Flowers [38]</th>
<th>iNaturalist [39]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Only $12^{th}$ (last block)</td>
<td>✔️</td>
<td>✗</td>
<td>68.16</td>
<td>82.58</td>
<td>38.28</td>
</tr>
<tr>
<td></td>
<td>✔️</td>
<td>✔️</td>
<td>70.66</td>
<td>86.58</td>
<td>41.22</td>
</tr>
<tr>
<td>From $1^{st}$ to $12^{th}$</td>
<td>✔️</td>
<td>✗</td>
<td>72.90</td>
<td>91.38</td>
<td>44.03</td>
</tr>
<tr>
<td></td>
<td>✔️</td>
<td>✔️</td>
<td>73.16</td>
<td>91.27</td>
<td>43.33</td>
</tr>
<tr>
<td>From $9^{th}$ to $12^{th}$</td>
<td>✔️</td>
<td>✗</td>
<td>73.58</td>
<td>90.00</td>
<td>45.15</td>
</tr>
<tr>
<td></td>
<td>✔️</td>
<td>✔️</td>
<td>73.37</td>
<td>90.33</td>
<td>45.12</td>
</tr>
</tbody>
</table>
Conclusion

- Vision Transformers perform well
  - Occlusion
  - Distributional shifts and patch permutations
  - Automatic segmentation
  - Adversarial patches and common corruptions
  - Transferability to downstream tasks
References