Literature Survey

on

" Face Transformer, Swin Transformer & Neighborhood Attention Transformer"

Paper Code : PCC-CS681

Submitted by

Debargha Mitra Roy
Roll No. - 20101104010
Registration No. - 201010100110029

with

Bikash Shaw
Roll No. - 20101104059
Registration No. - 201010100110044

and

Suprio Kundu
Roll No. - 20101104062
Registration No. - 201010100110005

Submitted to

Prof. Srinibas Rana

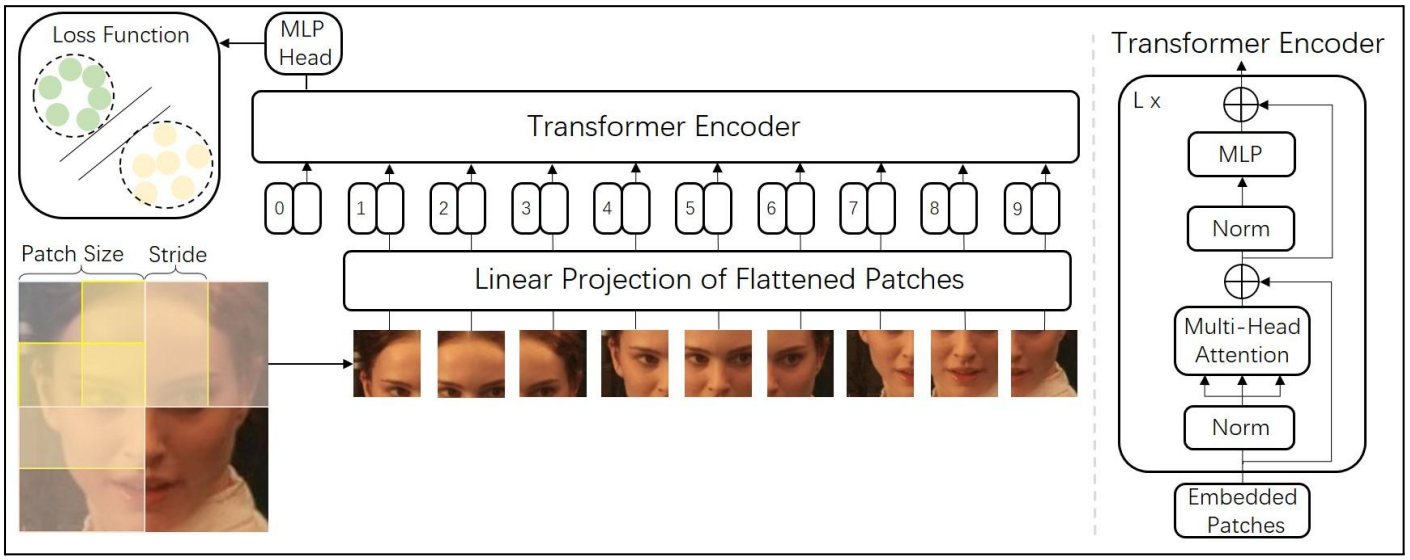
Department of Computer Science and Engineering
Jalpaiguri Government Engineering College
Jalpaiguri, West Bengal 735102

Index

Sl. No.	Topic	Content	Page No.
1.	Literature Survey on Face Transformer	Introduction	2
		Abstract	2
		Objective	2 - 3
		Problem Formation	3
		Proposed Methodology	3
2.	Literature Survey on Swin Transformer	Introduction	3 - 4
		Abstract	4 - 5
		Objective	5
		Problem Formation	5 - 6
		Proposed Methodology	6 - 7
3.	Literature Survey on Neighborhood Attention Transformer	Introduction	7
		Abstract	8
		Objective	8 - 9
		Problem Formation	9
		Proposed Methodology	9 - 10
4.	References		10

Literature Survey on Face Transformer

A face transformer is a type of deep learning model that is used for face recognition and other face-related tasks. It is based on the transformer architecture, which was originally developed for natural language processing tasks. Transformer models are known for their ability to learn long-range dependencies, which makes them well-suited for tasks such as face recognition, where the identity of a person can be determined by their facial features, even if the features are partially obscured or distorted.



Architecture of Face Transformer

- **Abstract :-**

Recently there has been a growing interest in Transformer not only in NLP but also in computer vision. We wonder if transformers can be used in face recognition and whether it is better than CNNs. Therefore, we investigate the performance of Transformer models in face recognition. Considering the original Transformer may neglect the interpatch information, we modify the patch generation process and make the tokens with sliding patches which overlap with each other. The models are trained on *CASIA-WebFace* and *MSCeleb-1M* databases, and evaluated on several mainstream benchmarks, including *LFW, SLLFW, CALFW, CPLFW, TALFW, CFP-FP, AGEDB* and *IJB-C* databases. We demonstrate that Face Transformer models trained on a large-scale database, *MS-Celeb-1M*, achieve comparable performance as CNN with similar number of parameters and MACs.

- **Objective :-**

Here are some of the specific objectives of Face Transformer —

- To learn a representation of face images that is invariant to variations in lighting, pose, and expression.
- To achieve state-of-the-art results on face recognition benchmarks.
- To be robust to variations in the quality of the input images.

- To be efficient in terms of computation and memory.

- **Problem Formation :-**

  Here are some of the specific problems that need to be addressed in order to improve the performance of Face Transformer models —

  - **Data augmentation:** Data augmentation can be used to increase the amount of data available for training face transformer models. This can be done by artificially creating new face images by applying transformations such as cropping, flipping, and rotating.

  - **Robustness to variations in lighting, pose, and expression:** Face transformer models can be made more robust to variations in lighting, pose, and expression by using techniques such as adversarial training and data augmentation.

  - **Computational efficiency:** Face transformer models can be made more computationally efficient by using techniques such as pruning and quantization.

- **Proposed Methodology :-**

  Here are some proposed methodologies on the problems of Face Transformer —

  - **Using more powerful hardware:** Face transformer models can be made more computationally efficient by using more powerful hardware, such as GPUs and TPUs.

  - **Using more advanced techniques:** There are a number of more advanced techniques that could be used to improve the performance of face transformer models, such as distillation and transfer learning.

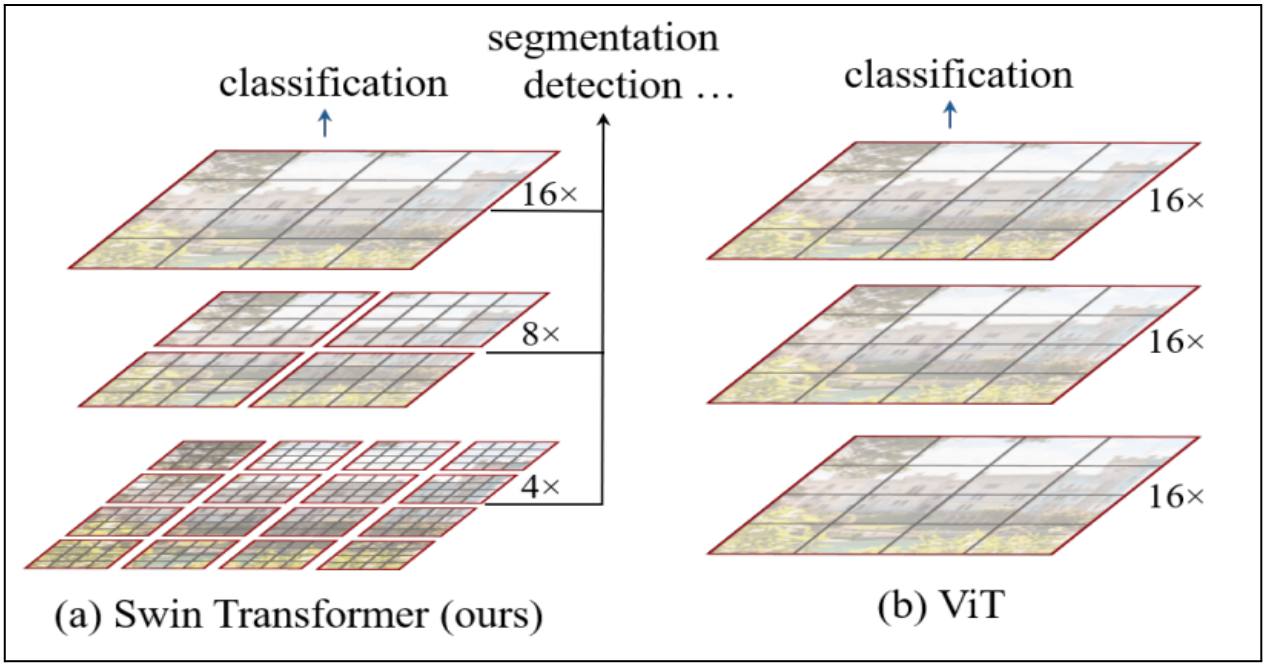
  - **Collecting more data:** The performance of face transformer models could also be improved by collecting more data. This could be done by collecting data from a wider variety of sources, such as social media and surveillance cameras.

Literature Survey on Swin Transformer

Swin Transformer is a hierarchical vision transformer that was proposed by Liu et al. in 2021. It is a powerful model that has achieved state-of-the-art results on a number of image recognition tasks, including image classification, object detection, and semantic segmentation.

The Swin Transformer architecture is based on the **Vision Transformer (ViT)** architecture, which was proposed by Dosovitskiy et al. in 2020. ViT is a powerful model that can learn to

represent images as a sequence of tokens, and then use self-attention to learn long-range dependencies between the tokens.



Architecture of Swin Transformer

- **Abstract :-**

Swin Transformer is a hierarchical vision transformer that uses shifted windows to compute self-attention. This allows the model to learn features at different scales, while also being more efficient than traditional self-attention models.

Swin Transformer consists of a stack of Swin blocks, each of which consists of the following layers:

- A patch embedding layer that converts the input image into a sequence of patches.
- A multi-scale grouping layer that groups the patches into different scales.
- A shifted window self-attention layer that computes self-attention over the patches in each scale.
- A feed-forward layer that applies a linear transformation to the output of the self-attention layer.

The Swin blocks are stacked in a hierarchical fashion, with each subsequent block learning features at a finer scale. This allows the model to learn a hierarchy of features, from coarse to fine.

The shifted window self-attention layer in Swin Transformer uses a sliding window to compute self-attention over the patches in each scale. This allows the model to learn features at different scales, while also being more efficient than traditional self-attention models.

The multi-scale grouping layer in Swin Transformer groups the patches into different scales. This allows the model to learn more complex features, which can help it to achieve better performance on downstream vision tasks.

The feed-forward layer in Swin Transformer applies a linear transformation to the output of the self-attention layer. This allows the model to learn more complex features, and to improve the performance of the model on downstream vision tasks.

Overall, Swin Transformer is a powerful hierarchical vision transformer that can be used for a variety of vision tasks. It is more efficient and scalable than traditional self-attention models, while still being able to achieve competitive performance.

Here are some of the key features of Swin Transformer —

- **Shifted windows:** Swin Transformer uses shifted windows to compute self-attention. This allows the model to learn features at different scales, while also being more efficient than traditional self-attention models.

- **Multi-scale grouping:** Swin Transformer uses multi-scale grouping to learn features at different scales. This allows the model to learn more complex features, which can help it to achieve better performance on downstream vision tasks.

- **Dilated convolution:** Swin Transformer uses dilated convolution to learn long-range dependencies. This allows the model to learn relationships between tokens that are far apart, which can be important for vision tasks such as object detection and segmentation.

Swin Transformer has been shown to achieve state-of-the-art results on a variety of vision tasks, including image classification, object detection, and semantic segmentation. It is a promising new approach to self-attention for vision tasks, and it is likely to be used in a variety of applications in the future.

- **Objective :-**

Here are some objectives behind Swin Transformer —

1. **Shifted windows:** Swin Transformer uses shifted windows to compute self-attention. This allows the model to learn features at different scales, while also being more efficient than traditional self-attention models.

2. **Multi-scale grouping:** Swin Transformer uses multi-scale grouping to learn features at different scales. This allows the model to learn more complex features, which can help it to achieve better performance on downstream vision tasks.

3. **Dilated convolution:** Swin Transformer uses dilated convolution to learn long-range dependencies. This allows the model to learn relationships between tokens that are far apart, which can be important for vision tasks such as object detection and segmentation.

- **Problem Formation :-**

Here are some of the challenges that need to be addressed in order to improve Swin Transformer —

1. **How to reduce the computational complexity of Swin Transformer.** This could be done by using more efficient implementations of the model, or by using a different attention mechanism.

2. **How to make the Swin Transformer easier to train.** This could be done by using better data augmentation techniques, or by using a different optimizer.

3. **How to make the Swin Transformer more well-suited for tasks that require detailed processing for every pixel.** This could be done by using a different feature extraction layer, or by using a different attention mechanism.

- **<u>Proposed Methodology :-</u>**

Here are some proposed methodologies on the problems of Swin Transformer —

1. **How to reduce the computational complexity of Swin Transformer?**

   - **Use more efficient implementations of the model:** This could be done by using a different attention mechanism, or by using a different implementation of the self-attention layer.

   - **Use a different attention mechanism:** There are a number of different attention mechanisms that could be used in place of the self-attention mechanism used in Swin Transformer. These mechanisms may be more efficient, or they may be able to learn better representations.

   - **Use a different feature extraction layer:** The feature extraction layer in Swin Transformer is responsible for converting the input image into a sequence of patches. This layer could be replaced with a different layer that is more efficient, or that can learn better representations.

2. **How to make the Swin Transformer easier to train?**

   - **Use better data augmentation techniques:** Data augmentation can help to improve the performance of Swin Transformer by making the training data more diverse. This can help the model to learn better representations, and it can also help the model to be more robust to noise.

   - **Use a different optimizer:** The optimizer is responsible for updating the weights of the model during training. A different optimizer may be able to help the model to converge more quickly, or it may be able to help the model to learn better representations.

   - **Use a different learning rate schedule:** The learning rate schedule controls how quickly the weights of the model are updated during training. A different learning rate

schedule may be able to help the model to converge more quickly, or it may be able to help the model to learn better representations.

3. **How to make the Swin Transformer more well-suited for tasks that require detailed processing for every pixel?**
   - **Use a different feature extraction layer:** The feature extraction layer in Swin Transformer is responsible for converting the input image into a sequence of patches. This layer could be replaced with a different layer that is more capable of extracting detailed features.

   - **Use a different attention mechanism:** The attention mechanism in Swin Transformer is responsible for learning relationships between different patches. A different attention mechanism may be able to learn better relationships between patches, which could help the model to learn more detailed features.

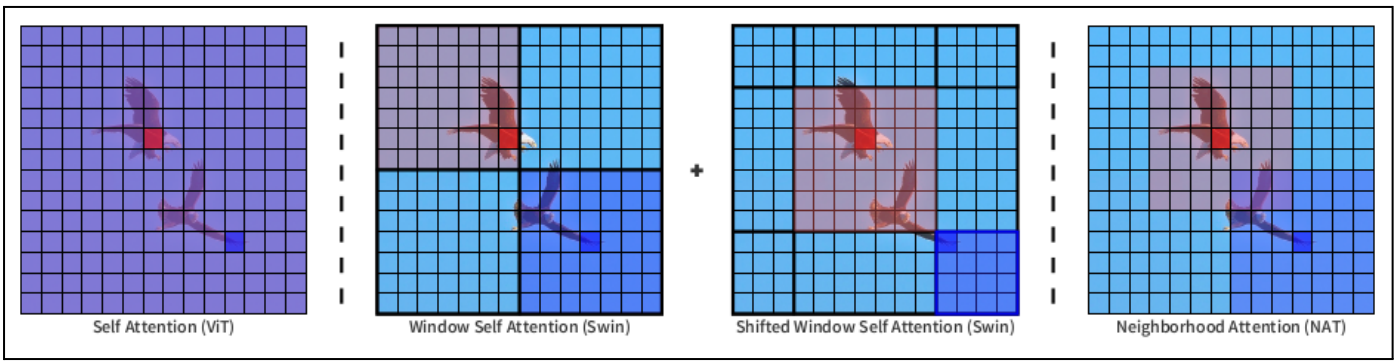
   - **Use a different loss function:** The loss function is used to measure the performance of the model during training. A different loss function may be able to help the model to learn more detailed features.

Literature Survey on Neighborhood Attention Transformer (NAT)

Neighborhood Attention Transformer (NAT) is a recent model architecture that has gained attention in the field of natural language processing (NLP) and computer vision.

Neighborhood Attention Transformer is an extension of the popular Transformer model, which has achieved remarkable success in various NLP tasks such as machine translation, language understanding, and text generation. The Transformer model relies on self-attention mechanisms to capture dependencies between different words or tokens in a sequence.

The idea behind Neighborhood Attention Transformer is to incorporate locality information into the self-attention mechanism. Traditional Transformers operate on a global level, considering all tokens in the sequence. However, in certain tasks, such as image analysis or graph-based data, it is often useful to consider local neighborhoods or patches of tokens instead of the entire sequence. This is particularly important when dealing with data that exhibits strong spatial or structural relationships.



Architecture of Neighborhood Attention Transformer

- **Abstract :-**

Neighborhood Attention Transformer is a self-attention mechanism that localizes attention to a neighborhood around each token. This introduces local inductive biases, maintains translational equivariance, and allows receptive field growth without needing extra operations.

NAT consists of the following layers —

- A patch embedding layer that converts the input image into a sequence of patches.
- A neighborhood attention layer that computes attention over the patches in a local neighborhood around each token.
- A feed-forward layer that applies a linear transformation to the output of the neighborhood attention layer.

The neighborhood attention layer in NAT uses a sliding window to compute attention over the patches in a local neighborhood around each token. This allows the model to learn features at different scales, while also being more efficient than traditional self-attention models.

The feed-forward layer in NAT applies a linear transformation to the output of the neighborhood attention layer. This allows the model to learn more complex features, and to improve the performance of the model on downstream vision tasks.

Overall, Neighborhood Attention Transformer is a powerful self-attention mechanism that can be used for a variety of vision tasks. It is more efficient and scalable than traditional self-attention models, while still being able to achieve competitive performance.

Here are some of the key features of Neighborhood Attention Transformer —

- **Local attention:** Neighborhood Attention localizes attention to a neighborhood around each token. This introduces local inductive biases, maintains translational equivariance, and allows receptive field growth without needing extra operations.
- **Efficiency:** Neighborhood Attention is significantly more efficient than traditional self-attention models, as it reduces the computational complexity from quadratic to linear. This makes it possible to scale up Neighborhood Attention to large images without running into memory or computational constraints.
- **Scalability:** Neighborhood Attention is scalable to large images, as it can be used to learn features at multiple scales. This makes it suitable for a wide range of vision tasks, including image classification, object detection, and semantic segmentation.

Neighborhood Attention Transformer has been shown to achieve state-of-the-art results on a variety of vision tasks, including image classification, object detection, and semantic segmentation. It is a promising new approach to self-attention for vision tasks, and it is likely to be used in a variety of applications in the future.

- **Objective :-**

Here are some objectives behind Neighborhood Attention Transformer (NAT) —

1. **Reduce the computational complexity of self-attention.** The quadratic complexity of self-attention makes it difficult to scale up to large images. NAT addresses this by localizing attention to a neighborhood around each token, which reduces the number of pairwise comparisons that need to be performed.
2. **Preserve translational equivariance.** Translational equivariance is a desirable property of self-attention for vision tasks, as it allows the model to learn invariant features that are independent of the position of the input tokens. NAT preserves translational equivariance by using a sliding-window pattern to compute attention.

3. **Allow receptive field growth without extra operations.** The receptive field of a self-attention model is the size of the region that each token can attend to. NAT allows the receptive field to grow without needing extra operations by using a sliding-window pattern to compute attention.

- **<u>Problem Formation :-</u>**

Here are some of the challenges that need to be addressed in order to improve Neighborhood Attention Transformer (NAT) —

1. **How to learn long-range dependencies with local attention.** This could be addressed by using a combination of local and global attention, or by using a hierarchical attention mechanism.

2. **How to scale up NAT to very large images.** This could be addressed by using a more efficient implementation of NAT, or by using a hierarchical attention mechanism.

3. **How to improve the performance of NAT on downstream vision tasks.** This could be addressed by using a more powerful NAT architecture, or by using better training techniques.

- **<u>Proposed Methodology :-</u>**

Here are some proposed methodologies on the problems of Neighborhood Attention Transformer (NAT) —

1. **How to learn long-range dependencies with local attention**?

    - **Use a combination of local and global attention:** This would allow the model to consider both the immediate neighbors of each token, as well as all tokens in the input. This could be done by using a hierarchical attention mechanism, where the model first attends to local neighborhoods, and then attends to the global context.

    - **Use a hierarchical attention mechanism:** This would allow the model to attend to different scales of the image, which would help it to learn long-range dependencies without becoming too computationally expensive.

    - **Use a more efficient implementation of local attention:** This would allow the model to attend to larger neighborhoods, which would help it to learn longer-range dependencies.

2. **How to scale up NAT to very large images?**

- **Use a more efficient implementation of NAT:** This would reduce the computational complexity of NAT, which would make it possible to scale up to very large images without becoming too computationally expensive.

- **Use a hierarchical attention mechanism:** This would allow the model to attend to different scales of the image, which would help it to scale up to very large images without becoming too computationally expensive.

- **Use a more powerful NAT architecture:** This would allow the model to learn more complex features, which would help it to achieve better performance on very large images.

3. **How to improve the performance of NAT on downstream vision tasks?**

    - **Use a more powerful NAT architecture:** This would allow the model to learn more complex features, which would help it to achieve better performance on downstream vision tasks.

    - **Use better training techniques:** This would help the model to learn better representations, which would help it to achieve better performance on downstream vision tasks.

    - **Use a larger dataset:** This would give the model more data to learn from, which would help it to achieve better performance on downstream vision tasks.

References

[1] Yaoyao Zhong, Weihong Deng. Face Transformer: Face Transformer for Recognition on 27 Mar 2021. (arXiv preprint arXiv:2103.14803v2, 2021 – pdf)

[2] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, Baining Guo, et al. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows (Microsoft Research Asia) on 17 Mar 2021. (arXiv preprint arXiv:2103.14030, 2021 – pdf)

[3] Ali Hassani , Steven Walton , Jiachen Li , Shen Li , Humphrey Shi, et al. Neighborhood Attention Transformer(NAT): Sliding-Window attention mechanism for vision (SHI Labs @ U of Oregon & UIUC, Picsart AI Research (PAIR), Meta/Facebook AI) on 14 Apr 2022. (arXiv preprint arXiv:2204.07143, 2022 – pdf)