Transformer models for FER

Annam Harika, Avula Sai Sruthi, Maddina Syamala, Guide Name: Mrs. Jyostna Devi Bodapati

Department of Computer Science and Engineering,
Vignan's foundation for science Technology and research university Guntur-522213, India

---

1. Abstract:

This paper offers a hybrid version of imaginative and prescient Transformer, referred to as Swin Transformer and CNN (Convolutional Neural Network). The problems in adapting Transformer from language to imaginative and prescient stem from variations among the 2 domains, which includes massive versions withinside the scale of visible entities and the excessive decision of pixels in pictures as compared to phrases in text.Overfitting, exploding gradient, and sophistication imbalance are the fundamental demanding situations at the same time as education the version the use of CNN. These problems can lessen the overall performance of the version.To deal with those distinctions, we advise a hybrid version of Vision Transformer whose illustration is computed with Shifted home windows and CNN.The shifted windowing scheme improves performance via way of means of limiting self-interest computation to non-overlapping neighborhood home windows at the same time as nevertheless taking into account cross-window connections. We show that the fusion of the transformer and CNN-primarily based totally fashions outperforms the respective baseline version. We additionally show the only mannerto mix the transformer and CNN.

2. Introduction:

Convolutional neural networks (CNNs) have long dominated computer vision modeling. The CNN architecture has evolved to become more and more powerful with larger, larger interconnects, and more sophisticated convolutional formats, including AlexNet and its innovative performance in ImageNet image classification challenges. I did. These architectural advances have improved performance and improved the overall field. CNNs serve as the backbone network for various imaging tasks.

On the other hand, the evolution of community architectures in Natural Language Processing (NLP) has taken an exceptional path, with the Transformer now being the dominant architecture. The Transformer, which became designed for series modeling and transduction tasks, is exceptional for its use of interest to version long-time period dependencies in data. Its out of the ordinary achievement withinside the language area has brought on researchers to analyze its version to laptop vision, wherein it has currently established promising effects on particular tasks, consisting of photo category and joint vision-language modeling.

In this paper, we aim to broaden Transformer's applicability so that it can serve as ageneral-purpose backbone for computer vision, as it does for NLP and as CNNs do in vision. We find that differences between the two modalities explain significant challenges in transferring its high performance in the language domain to the visualdomain. Scale is one of these distinctions. Unlike word tokens, which serve as the basic processing elements in language Transformers, visual elements can vary significantly in scale, a problem that is addressed in tasks such as object detection. Tokens in existing Transformer-based models are all fixed scale, which is unsuitable for these vision applications. Another distinction is that pixels in imageshave a much higher resolution than words in text passages. Many vision tasks, such as semantic segmentation, necessitate dense prediction at the pixel level, which would be intractable for Transformer on high-resolution images due to the computational complexity of its self-attention being quadratic to image size. To address these issues, we propose Swin Transformer, a general-purpose Transformerbackbone that constructs hierarchical feature maps and has linear computational complexity to image size.

Swin Transformer builds a hierarchical representation by starting with small patches and gradually merging them in deeper Transformer layers. With these hierarchical feature maps, the Swin Transformer model can easily leverage advanced dense prediction techniques such as feature pyramid networks or U-Net. Self-attention is computed locally within non-overlapping windows that partition an image to achieve linear computational complexity. Because the number of

patches in each window is fixed, the complexity is proportional to image size. Swin Transformer, in contrast to previous Transformer-based architectures that produce feature maps of a single resolution and have quadratic complexity, is suitable as a general-purpose backbone for various vision tasks.

Swin Transformer's window partition shift between consecutive self-attention layers is a key design element. The shifted windows connect the windows of the previous layer, providing connections that significantly increase modeling power. This strategy is also efficient in terms of real-world latency: all query patches within a window share the same key set1, making memory access in hardware easier. Earlier sliding window-based self-attention approaches, on the other hand, suffer from low latency on general hardware due to different key sets for different query pixels.

Our experiments show that the proposed shifted window method has much lower latency than the sliding window method while providing comparable modeling power. The shifted window approach is also advantageous for all-MLP architectures. Our experiments have revealed the best combinations of Swin transformer blocks and CNN. The proposed hybrid architecture of Swin Transformer and CNN performs well on image classification, object detection, Face Emotion Recognition, and semantic segmentation tasks. On all three tasks, it significantly outperforms the ViT / DeiT and ResNet models with comparable latency.

We believe that unified structure for pc imaginative and prescient and herbal language processing might advantage each field due to the fact it might permit for joint modeling of visible and textual alerts and greater deeply shared modeling knowledge. We wish that Swin Transformer`s robust overall performance on numerous imaginative and prescient issues will support this notion and inspire unified modeling of imaginative and prescient and language alerts.

3. Related Work:

CNN is the most commonly used network model in computer vision. CNN has been around for decades, but it wasn't until AlexNet came out that it gained momentum and gained popularity. Since then, deeper and more effective convolutional neural architectures such as VGG, GoogleNet, ResNet, DenseNet, HRNet, and EfficientNet have been proposed to further accelerate the wave of deep learning in computer vision. Apart from architectural advances, much work has been done to improve the individual convolution layers, such as: B. Depth convolution and deformable convolution. CNNs and their variants remain the mainbackbone architecture for computer vision applications, but emphasize the important potential of Transformer-like architectures for integrated modeling of vision and language combined with CNNs. Our work hopes to achieve high performance in a variety of basic visual recognition tasks and contribute to a paradigm shift in modeling.

Some work uses self-aware layers to replace some or all of the popular ResNet spatial convolution layers. Again, it is inspired by the success of the self-awareness layer and transformer architecture in the NLP space. To speed up optimization, these tasks calculate self-awareness within the local window for each pixel, achieving slightly better accuracy / FLOP trade-offs than the corresponding ResNet architecture. However, due to the high cost of memory access, the actual delay is significantly higher than that of a convolutional network. Instead of moving windows, we suggest moving windows between successive layers to allow for a more efficient implementation on common hardware.

Another area of research is to add self-attention layers or Transformers to a standard CNN architecture. The self-attention layers can supplement backbones or head networks by encoding distant dependencies or heterogeneous interactions. Transformer's encoder-decoder design has recently been used for object detection and instance segmentation tasks. Our work investigates the adaptation of Transformers for basic visual feature extraction, which is then combined with CNN.

Vision Transformer (ViT) and its derivatives are most relevant to our work. For image classification, ViT's pioneering work applies the transformer architecture directly to non-overlapping medium-sized image fields. It provides an impressive compromise between the speed and accuracy of image classification compared to convolutional networks. While ViT needs a large training dataset to run properly, DeiT has introduced several training strategies that enable ViT to run well on a small ImageNet 1K dataset. Due to the low resolution functional map and the secondary complexity of image size, ViT's architecture is not suitable for use as a high-density image processing task or a general-purpose backbone network with high input image resolution. There is some work to apply the ViT model to high-density vision tasks such as object detection and

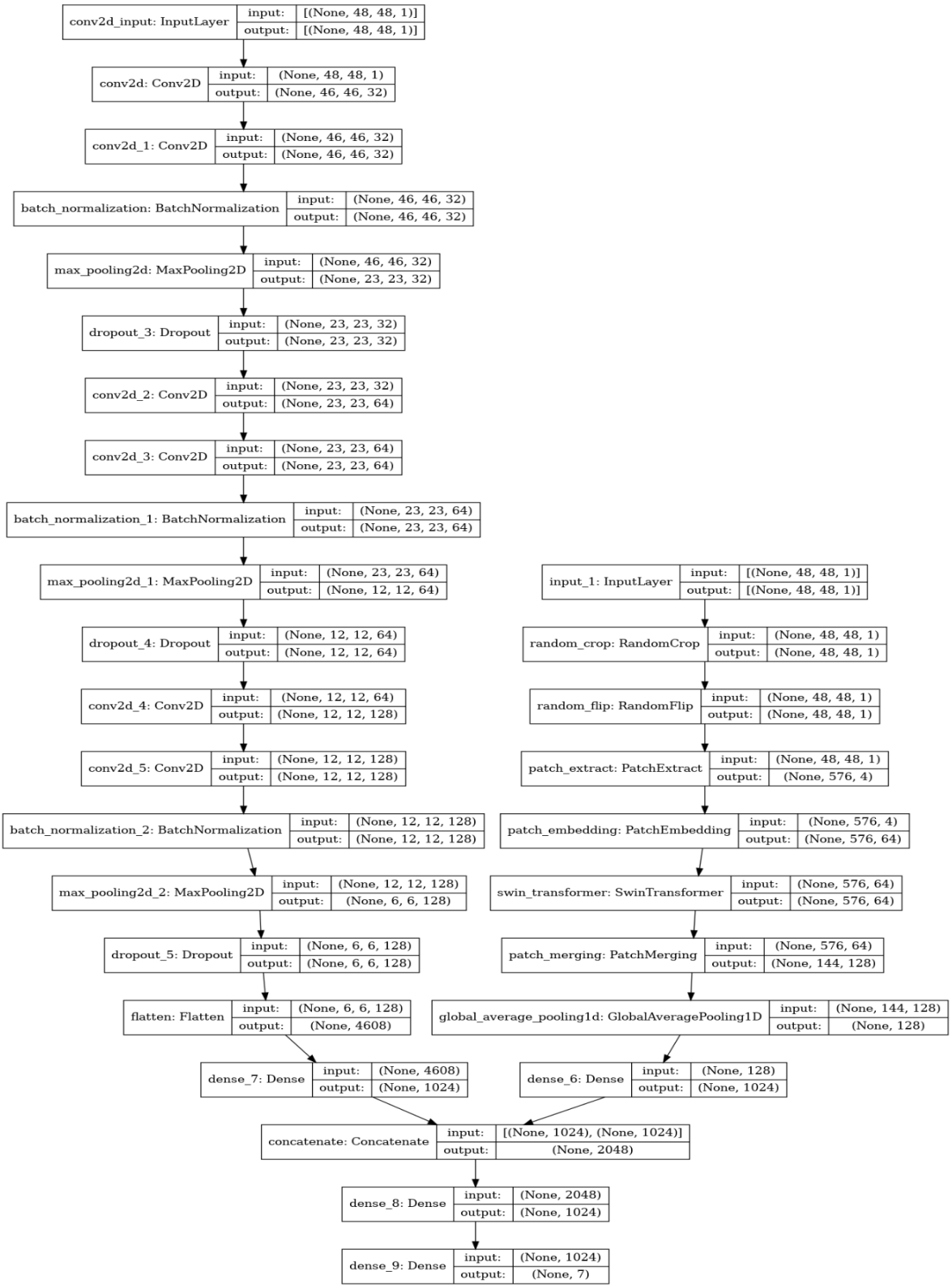
semantic segmentation usingdirect upsampling or deconvolution, but the performance is relatively poor. In parallel with our work, others are changing the ViT architecture to improve image classification. Empirically, our work focuses on general performance rather than specific classification, but our Swin Transformer architecture is the best of these image classification methods between speed and accuracy. You will find that you will reach a compromise. Another parallel work is looking at a similar idea for creating multi-resolution feature maps in Transformers. Its complexity is still quadratic with respect to image size, but our complexity works linearly and locally.This has proven useful in modeling the high correlation of visual signals.

4. Method:

4.1 Architecture:

In this paper we have proposed the hybrid model of CNN and Swin Transformer sothat the advantages of both the methods can be prioritized by reducing the drawbacks in both methods and to improve the accuracy of the model. In ourmodel first the CNN is applied on the dataset. We have used the dataset FER2013. Then the Swin model is applied. We have merged the two models and resulted in the accuracy in various types of merging and making the variations in the Swin transformer blocks.

The architecture of our hybrid model is as shown in the fig



4.2 Swin Transformer:

The architecture of the Swin transformer is having 4 main components:

- 4.2.1 Patch Extraction
- 4.2.2 Linear Embedding
- 4.2.3 Patch Merging
- 4.2.4 Swin transformer Block

4.2.1 Patch Extraction:

A single image of the dataset is partitioned into several small patches. These patches can be overlapping or non-overlapping, it is completely based on our perspective. In this paper we preferred non-overlapping patch partitions. In our implementation, we used a patch size of 2 × 2.

4.2.2 Linear Embedding:

In this step the output of the previous step is taken i.e. the patches are taken as input. This step converts the patch into a c-dimensional vector. For converting the patches into vectors this step is more useful.

4.2.3 Patch Merging:

Every image has 2x2 patches. These patches are merged into a single patch by using a linear layer. The merging is done because by a single patch we can define the size of the output we want. Exploiting this advantage we double c so that now c becomes 2c. Here we have decreased the patches by 2 but increase c by 2 as well.

4.2.4 Swin transformer block:

Problem of Multi-head self-attention(MSA) :

It is a standard attention mechanism and well for language processing tasks, but images are different, in case of images we divide the image into several non-overlapping patches. Even though we divide the input image into different patches we still have to compute the self attention for a given patch with all the other patches in the image. So, this becomes compute intensive even for a reasonably large sized image.

Inorder to overcome this, swin transformer introduces the windows. A window divides the input image into several parts, whenever we compute self attention between patches within that window and we ignore the rest of the patches. In swin transformer, one layer of transformer is replaced by two layers, they are WMSA and SWMSA where W stands for window based SW stands for shifted window

Working of WSMA layer: The input image is divided into four windows and computes attention for patches within the window. The window is straightforward and the first layer of the swin transformer block. The next stage is the shifted window MSA. The window is shifted by two patches and then computes the attention within these windows while the windows shift the empty space created without and within any pixels. A naive solution for this is to zero pad that space. A more sophisticated solution which is called cycle shifting in this paper. This is to copy over the patches from top to bottom and from left to right and also diagonallyacross to makeup for the missing patches.

The first module uses a regular window partitioning strategy which starts from the top-left pixel, and the 8 × 8 feature map is evenly partitioned into 2 × 2 windows ofsize 4 × 4 (M = 4). Then, the next module adopts a windowing configuration that is shifted from that of the preceding layer, by displacing the windows by $\left(\lfloor \frac{M}{2} \rfloor, \lfloor \frac{M}{2} \rfloor\right)$ pixels from the regularly partitioned windows.

With the shifted window partitioning approach, consecutive Swin Transformer blocks are computed as

$$\hat{\mathbf{z}}^{l} = \text{W-MSA}\left(\text{LN}\left(\mathbf{z}^{l-1}\right)\right) + \mathbf{z}^{l-1},$$
$$\mathbf{z}^{l} = \text{MLP}\left(\text{LN}\left(\hat{\mathbf{z}}^{l}\right)\right) + \hat{\mathbf{z}}^{l},$$
$$\hat{\mathbf{z}}^{l+1} = \text{SW-MSA}\left(\text{LN}\left(\mathbf{z}^{l}\right)\right) + \mathbf{z}^{l},$$
$$\mathbf{z}^{l+1} = \text{MLP}\left(\text{LN}\left(\hat{\mathbf{z}}^{l+1}\right)\right) + \hat{\mathbf{z}}^{l+1},$$

where l denote the output features of the (S) W-MSA module and the z and z^{l} MLP module for block l, respectively

Finally the merge layer is established using a dense layer with softmax activation function. The softmax activation function is applied in the dense layers of FER-net architecture. Softmax is used to calculate the probabilities of the predicted classes. The class with the highest probability is considered as an output.

4.3 CNN:

A convolutional neural community (CNN) is a shape of synthetic neural community this is specially meant to investigate pixel enter and is utilized in photograph popularity and processing. A CNN employs a era just like a multilayer perceptron this is optimized for low processing needs. An enter layer, an output layer, and a hidden layer with numerous convolutional layers, pooling layers, completely related layers, and normalizing layers make up CNN`s layers.

The statistical homes of various sub blocks in a herbal photograph are commonly consistent, implying that the capabilities found out from a selected subblock of the photograph may be used as a detector. To attain the activation price with the identical characteristic, all subblocks of the whole photograph are traversed. To carry out convolutional summation over all characteristic maps with admire to the preceding layer and upload offsets, special trainable convolution kernels are used. The activation feature then outputs the neuron withinside the cutting-edge layer. The cutting-edge layer is made from characteristic maps with diverse capabilities.

In general, the convolutional layer calculation expression is:

$$y_j^l = \theta\left(\sum_{i=1}^{N_j^{l-1}} w_{i,j} \otimes x_i^{l-1} + b_j^l\right), j = 1, 2, \ldots, M$$

Where, l represents the current layer

l-1 represents the previous layer

y_j^l represents the jth feature map in the current layer

$w_{i,j}$ represents the convolution kernel between the ith feature map in the previous layer and the jth feature map in the current layer.

x_i^{l-1} represents the bias of the previous layer's ith feature map

b_j^l represents the bias of the current layer's jth feature map.

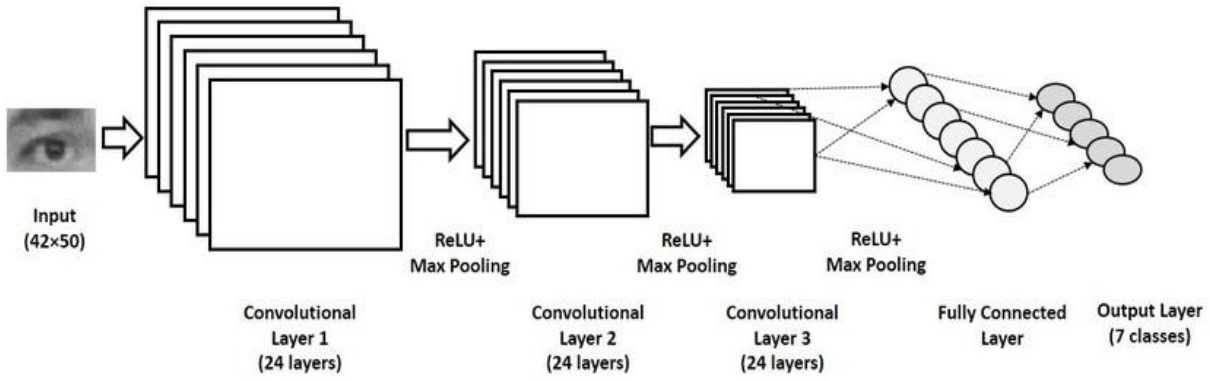
The term b_j^l is set to zero, i.e. $b_j^l = 0$ to train the network quickly and reduce the learning parameters.

N_j^{l-1} is the number of feature maps in the current layer that connect to all feature maps in the previous layer

M is the number of feature maps in the current layer.

It consists of one input layer, five convolution layers, three max-pooling layers and one fully connected layer. Batch-normalization is applied to the outputs of convolutional layers. The first and second convolution layers consist of 32 and 64 filters of kernel_size 3x3 respectively. The input shape is (48,48,1) where 48x48 is the pixel size and 1 indicates that the image is grayscale. The third, fourth and fifthconvolution layers consist of 64, 128 and 128 filters respectively.

The schematic block diagram of the FER-net is shown in fig



The convolution layers perform convolution operations. The convolutional layers produce feature maps, which denote high-level features such as edges, corner points, color from the face region automatically. Once the feature maps are extracted , the next step is to move them to a ReLU layer. This layer introduces non-linearity to the network and performs element wise operation and the output is a rectified feature map. The rectified feature map now goes through a pooling layer. Pooling operation is also called subsampling which reduces the overfitting problem. Pooling is a down-sampling operation that reduces the dimensionality of the feature map. We have done pooling operations with 2x2 filters and stride 2.

$\theta(x)$ represents the activation function. The Rectified Linear Units (ReLU) function is used instead of the usual sigmoid or hyperbolic tangent function because it is more sparse.. The expression for the ReLU function is:

$\theta(x)=\max(0,x)$

The number of feature maps grows in proportion to the number of convolution layers, resulting in a sharp increase in feature dimensions. If all of the features areused to train the Softmax classifier, the dimensions will be enormous. To avoid this issue, a pooled layer is typically used to reduce the feature dimension. The pooling layer performs downsampling. The pooling layer reduces the number of feature maps while increasing their size. That is, the pooling layer can make someoperations such as translation, scaling, and rotation more robust. If the sampling window size is nn, the feature map will be (1/ n)x(1/ n) of the original feature after downsampling. The general expression for the pooling is:

$$y_j^l = \theta(\beta_j^l down(y_j^{l-1}) + b_j^l)$$

Where,

l

y represents jth feature map in the current layer

j

$l-1$

y represents the previous feature map in the current layer

j

down() represents a down-sampling function

l

β represents the multiplicative bias

j

^{l} represents additive bias to the j feature map in the current layer

j

In the experiment, $\beta_j^l = 1$, $b_j^l = 0$

θ(x) represents the activation function

4.4 K-Fold Cross Validation:

Cross-validation is a function which evaluates data and returns the score.K-fold is a class which lets us split your data to k-folds. Here we have used the 10 folds i.e the value of k is 10. K-fold cross validation is a procedure used to estimate the skillof the model on new data.

4.5 Architecture Variants:

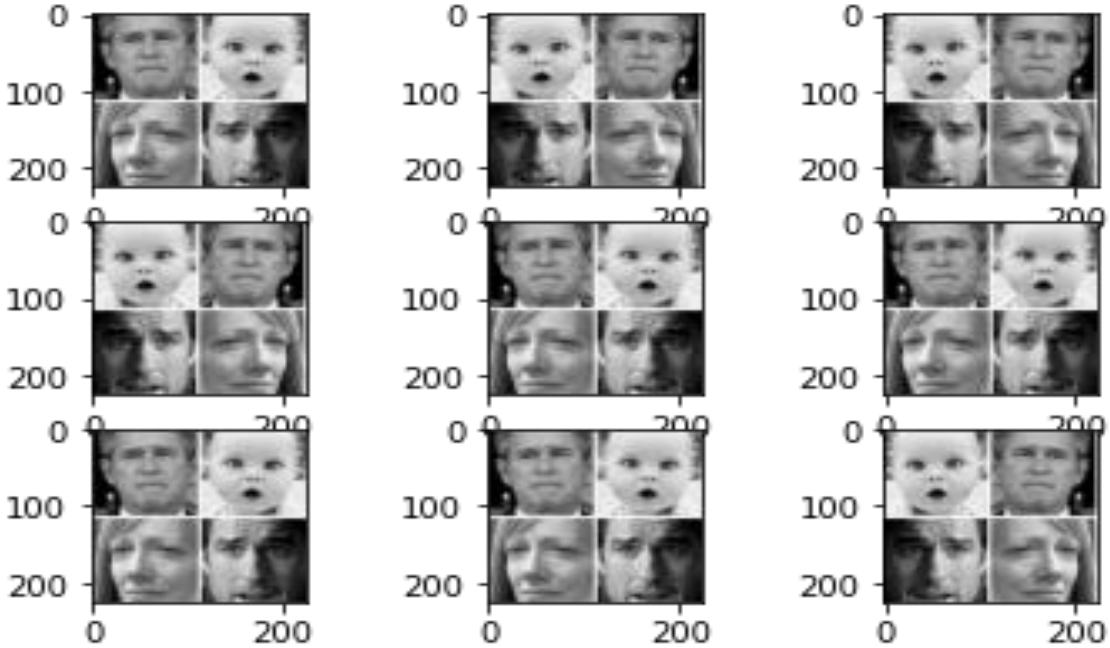
To attain the best model that gives the best accuracy, we have worked with various variations. We have varied the Swin transformer blocks, we have worked out each time by increasing the number of Swin blocks. We worked with one to four Swin blocks. Out of all the best accuracy is produced on using 2 Swin blocks.

As we decided that 2 Swin blocks give the best result, we variegated the type of merging the CNN and Swin transformer. We have merged the layers by concatenating, Dot, Maximum, Multiply and averaging

5. Experiments & Results:

We Used the standard dataset for face emotion recognition i.e. FER2013 which contains 34034 unique values(grayscale images). Each image is of size 48x48 pixel. 80% of the dataset is used for training while the rest for testing.

All the images are normalized to 0-1 scale by normalizing. We proposed 7 categories of emotions : anger, disgust, fear, happy, neutral, sadness and surprise. We also used preprocessing by Randomcrop. This layer will crop all the images in the same batch to the same cropping location.Next we used Randomflip which flipped the images horizontally.



The number of classes are 7, each patch size is 2x2 and the dropout rate is 0.03. The number of attention heads are 8, the MLP layer size is 256 and batch size is 128. The size of the attention window is 2 and the size of the shifting window is 1.

All the experiments in this paper are done on the standard dataset for Face Emotion Recognition FER2013. We have performed the experiments with the Swin blocks one at a time, two at a time, three at a time, four at a time. Out of all the experiments with two Swin blocks at a time resulted in the best accuracy. And by varying the type of merge of layers by Concat, Dot, Maximum, Multiply and Average we got the concatenation of two layers gives the best accuracy.

The results are as shown below:

CNN	**55.7**
CNN + Swish	**57.39**
VGG19 + CNN	**79.9**
VGG16 + CNN	**81.3**
Swin-1+ CNN + KFold	**82.03 (+- 7.29)**
Swin-2 + CNN + KFold	**82.75 (+- 7.85)**
Swin-3 + CNN + KFold	**82.72 (+- 9.51)**
Swin-4 + CNN + KFold	**82.58 (+- 8.68)**
Swin-1 + CNN (Concat)+ KFold	**82.70 (+- 7.75)**
Swin-2 + CNN (Concat)+ KFold	**83.00 (+- 7.63)**
Swin-2 + CNN (Dot) + KFold	**39.65 (+- 12.16)**
Swin-2 + CNN (Max) + KFold	**81.78 (+- 7.35)**
Swin-2 + CNN (Multiply) + KFold	**80.00 (+- 6.39)**

6. Conclusion:

This paper presents Swin Transformer and CNN, a vision Transformer hybrid model (Convolutional Neural Network). The difficulties in adapting Transformer from language to vision stem from differences between the two domains, such as large differences in visual entity scale and the high resolution of pixels in images compared to words in text. The major challenges while training the model with CNN are overfitting, exploding gradients, and class imbalance. These issues may impair the model's performance. To address these distinctions, we propose a VisionTransformer hybrid model whose representation is computed using Shifted windows and CNN. By limiting self-attention computation to non-overlapping local windows while still allowing for cross-window connections, the shifted windowing scheme improves efficiency. We show that combining transformer and CNN-based models outperforms the respective baseline model. We also showed how to combine the transformer and CNN most effectively and found out the best performing model for Face Emotion Recognition on standard FER2013 dataset.

7. References:

[1] B. Ko, "A Brief Review of Facial Emotion Recognition Based on Visual Information,"Sensors, vol. 18, no. 2, p. 401, 2018.

[2] R. C. Chivers, V. V Ramalingam, and A. Pandian, "Facial Emotion Recognition System – A Machine Learning Approach Facial Emotion Recognition System ± A Machine Learning," 2018.

[3] Alvarez, V. M., Velazquez, R., Gutierrez, S., & Enriquez-Zarate, J. (2018). A Method for Facial Emotion Recognition Based on Interest Points. 2018 International Conference on Research in Intelligent and Computing in Engineering (RICE), 1–4.

[4] Y. Zhang et al., "Facial Emotion Recognition Based on Biorthogonal Wavelet Entropy , Fuzzy Support Vector Machine , and Stratified Cross Validation," pp. 8375–8385, 2016.

[5] M.J. Lyons, J. Budynek, S. Akamatsu, "Automatic Classification Of Single Facial Images", IEEE Transactions on Pattern Analysis and Machine Intelligence 21 (12) (1999) 1357–1362.

[6] M. H. Siddiqi et al., "A Brief Review of Facial Emotion Recognition Based on Visual Information," 2018 IEEMA Eng. Infin. Conf. eTechNxT 2018, vol. 5, no. 1, pp. 196–201, 2018.

[7] Y. Tian, "Evaluation of Face Resolution for Expression Analysis", CVPR Workshop on FaceProcessing in Video, 2004.

[8] Y. Gao and K.H. Leung, "Face recognition using line edge map," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 6, June 2002.

[9] Zhao, L., Peng, X., Tian, Y., et al.: Semantic graph convolutional networks for 3d human pose regression. In: Proceedings of the IEEE/CVF Conference on Computer Vision and PatternRecognition, pp. 3425–3435 (2019)

[10] Tian, Y.-I., Kanade, T., Cohn, J.F.: Recognizing action units for facial expression analysis. IEEE Trans. Pattern Anal. Mach. Intell. 23(2), 97–115 (2001)

[11] Meng, Z., Liu, P., Cai, J., et al.: Identity-aware convolutional neural network for facialexpression recognition. In: 2017 12th IEEE International Conference on Automatic Face &Gesture Recognition (FG 2017), pp. 558–565 (2017)

[12] S. Deshmukh, M. Patwardhan, and A. Mahajan, "Survey on RealTime Facial Expression Recognition Techniques," IET Biom., pp. 1- 9, 2015

[13] Yang, H., Ciftci, U., Yin, L.: Facial expression recognition by deexpression residue learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2168–2177 (2018)

[14] Defferrard, M., Bresson, X., Vandergheynst, P.: Convolutional neural networks on graphs with fast localized spectral filtering. Adv. Neural Inf. Process. Syst. 29, 3844–3852 (2016)

[15] Gogi´c, I., Manhart, M., Pandži´c, I.S., et al.: Fast facial expression recognition using local binary features and shallow neural networks. Vis. Comput. 36(1), 97–112 (2020)

[16] T. Kundu and C. Saravanan, "Advancements and recent trends in emotion recognition using facial image analysis and machine learning models," 2017 International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICEECCOT), Mysuru, 2017, pp. 1-6

[17] H. Ebine, Y. Shiga, M. Ikeda and O. Nakamura, "The recognition of facial expressions with automatic detection of the reference face," 2000 Canadian Conference on Electrical and Computer Engineering. Conference Proceedings. Navigating to a New Era (Cat. No.00TH8492),Halifax, NS, 2000, pp. 1091-1099 vol.2.

[18] Lucey P, Cohn JF, Kanade T, Saragih J, Ambadar Z, Matthews I (2010) The extended Cohn–Kanade dataset (ck+): a complete dataset for action unit and emotion-specified expression.
In: 2010 IEEE computer society conference on computer vision and patternrecognition-workshops. IEEE, pp 94–101

[19] M. F. Valstar et al., « FERA 2015 - second Facial Expression Recognition and Analysis challenge », in 2015 11th IEEE International Conference and Workshops on Automatic Face andGesture Recognition (FG)

[20] Nebauer C.Evaluation of convolutional neural networks for visual recognition.IEEE Transactions on Neural Networks, 9 (4) (1998), pp. 685-696

[21] Uçar, A. (2017, July) “Deep Convolutional Neural Networks for facial expression recognition.” In Innovations in Intelligent Systems and Applications (INISTA), 2017 IEEEInternational Conference on (pp. 371-375)..

[22] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,pages 6154–6162, 2018

[23] Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. Deformable convnets v2: More deformable, better results. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 9308–9316, 2019.

[24] Xianzhi Du, Tsung-Yi Lin, Pengchong Jin, Golnaz Ghiasi, Mingxing Tan, Yin Cui, Quoc V Le, and Xiaodan Song. Spinenet: Learning scale-permuted backbone for recognition and localization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 11592–11601, 2020.

[25] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is 11 worth 16x16 words: Transformers for image recognition at scale. In International Conference on Learning Representations, 2021.

[26] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 770–778, 2016.

[27] Vinyals & Kaiser, Koo, Petrov, Sutskever, and Hinton. Grammar as a foreign language. In Advances in Neural Information Processing Systems, 2015.

[28] Kunihiko Fukushima. Cognitron: A self-organizing multilayered neural network. Biological cybernetics, 20(3):121– 136, 1975

[29] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Advances in NeuralInformation Processing Systems, pages 5998–6008, 2017.

[30] Bharat Singh, Mahyar Najibi, and Larry S Davis. Sniper: Efficient multi-scale training. In Advances in Neural Information Processing Systems, volume 31. Curran Associates, Inc., 2018.

[31] https://ieeexplore.ieee.org/document/8606597 L. Xu, M. Fei, W. Zhou and A. Yang, "Face Expression Recognition Based on Convolutional Neural Network," 2018 Australian & New Zealand Control Conference (ANZCC), 2018, pp. 115-118, doi: 10.1109/ANZCC.2018.8606597.

[32] https://ieeexplore.ieee.org/document/9751735 R. Appasaheb Borgalli and S. Surve, "Deep Learning Framework for Facial Emotion Recognition using CNN Architectures," 2022 International Conference on Electronics and Renewable Systems (ICEARS), 2022, pp.1777-1784, doi: 10.1109/ICEARS53579.2022.9751735.

[33] https://www.researchgate.net/publication/336858850_A_Face_Emotion_Recognition_Method_Using_Convolutional_Neural_Network_and_Image_Edge_Computing Zhang, Hongli & Jolfaei, Alireza & Alazab, Mamoun. (2019). A Face Emotion Recognition Method Using Convolutional Neural Network and Image Edge Computing. IEEE Access. PP. 1-1.10.1109/ACCESS.2019.2949741.

[34] https://www.researchgate.net/publication/351056923_Facial_Expression_Recognition_Using _CNN_with_Keras Khopkar, Apeksha & Adholiya, Ashish. (2021). Facial ExpressionRecognition Using CNN with Keras. Bioscience Biotechnology Research Communications. 14. 47-50. 10.21786/bbrc/14.5/10.

[35] https://arxiv.org/abs/2103.14030 Swin Transformer: Hierarchical Vision Transformer usingShifted Windows Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang,Stephen Lin, Baining Guo

[36] https://paperswithcode.com/method/swin-transformer Liu et al. in Swin Transformer: Hierarchical Vision Transformer using Shifted Window