Scene recognition project

ABSTRACT

In the evolving digital landscape, the significance of image classification, a subset of data mining techniques, has developed rapidly. Datasets like COCO[7], ImageNet[2], and ADE20K[15] have been pivotal in propelling research in this domain. Beyond traditional image classification, scene recognition emerges as a specified task, aiming to interpret the broader context of images, from identifying individual elements to discerning entire environments such as offices, bars, or stadiums. Such capabilities hold immense potential for online platforms, enhancing their ability to understand diverse scenes from user-contributed images. This research explores the performance of the Swin Transformer[9] in scene recognition tasks. The Swin_b model[9], trained on the Places365[14] dataset, achieved a notable Top 1 accuracy of approximately 58.679%, outperforming the WaveMix[5] model's 56.45%. The study also benchmarks the Swin Transformer against renowned models like ResNet-50[4] and Vision Transformer (ViT)[3]. The results underscore the Swin Transformer's potential in scene recognition, setting a new standard for future endeavors in this domain.

KEYWORDS

datasets, neural networks, scene recognition, image classification

1 INTRODUCTION

In the contemporary digital landscape, the proliferation of images necessitates a deeper understanding beyond mere object identification. This is where the significance of scene recognition emerges. Unlike traditional image classification that identifies individual components, scene recognition aims to comprehend the overarching context of an image.

For example, in an image showcasing a vibrant city park, traditional image classification might recognize elements such as trees, people, and benches. However, scene recognition thinks differently, providing an overarching analysis that captures the atmosphere, the interactions among individuals, and the prevailing mood of the environment.

As depicted in Figure 1, traditional image classification focuses on discrete entities within an image. In contrast, as demonstrated in Figure 2, scene recognition seeks to interpret the image in its entirety.

Deep learning models[6], with their capacity to discern intricate patterns and hierarchies, have shown promise in scene recognition tasks. The Swin Transformer[9], for instance, is a novel deep learning architecture that leverages shifted windows to capture local and global information, making it particularly suited for scene recognition. On the other hand, ResNet-50[4], a variant of the Residual Network, employs skip connections to mitigate the vanishing gradient problem, enabling deeper networks and improved performance in image classification tasks. In this study, I evaluate the efficacy of both the Swin Transformer[9] and ResNet-50[4] in scene recognition, utilizing the Places365[14] dataset as a benchmark, to determine their respective strengths and potential areas of improvement.

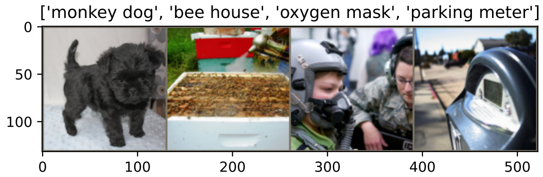


Figure 1: Images from ImageNet emphasizing individual objects

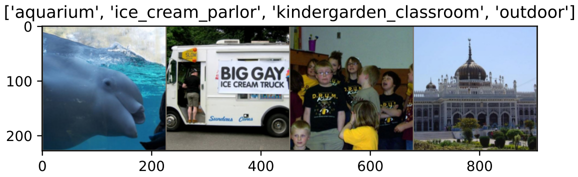

['aquarium', 'ice_cream_parlor', 'kindergarden_classroom', 'outdoor']
0
100
200
0
200
400
600
800

Figure 2: Images from Places365 illustrating comprehensive scenes

2 RELATED WORKS

2.1 ResNet-50

ResNet, short for Residual Networks, was introduced by He et al.[4] to address the degradation problem commonly observed in training deep neural networks. As networks become deeper, they tend to suffer from vanishing and exploding gradient issues, making them harder to optimize. ResNet introduces the concept of "skip connections" or "shortcuts" that allow the gradient to be directly back-propagated to earlier layers. The core idea behind ResNet is the introduction of a "residual block" that adds the output of the convolutional layer to its input.

The ResNet-50 variant consists of 50 layers and uses "bottleneck" blocks to reduce the number of parameters, making the network more efficient. The architecture is illustrated in Figure 3.

2.2 Transformers and Attention Mechanism

Transformers, introduced by Vaswani et al.[12], have revolutionized the field of natural language processing and are now making significant inroads into computer vision. The transformer architecture eschews the recurrent layers used in RNNs and LSTMs and relies entirely on the attention mechanism to draw global dependencies between input and output.

The attention mechanism allows the model to focus on different parts of the input data with varying degrees of attention. The core component of the attention mechanism is the scaled dot-product attention, which is computed as:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{1}$$

Where:

- Q is the query matrix.
- K is the key matrix.
- V is the value matrix.
- d_k is the dimension of the keys.

The softmax function ensures that the weights of the attention mechanism sum up to 1, and the division by $\sqrt{d_k}$ is a scaling factor that leads to more stable gradients.

Transformers utilize multi-head attention, where the model runs through the attention mechanism multiple times in parallel, allowing it to focus on different parts of the input data simultaneously. The application of transformers in image classification tasks has gained traction due to their ability to capture long-range dependencies and global context in images. Unlike traditional CNNs, which operate on local patches of an image, transformers can attend to any part of the image, making them particularly suited for tasks that require understanding the entire context of an image.

Vision Transformer (ViT). The Vision Transformer (ViT)[3] divides an image into fixed-size non-overlapping patches, linearly embeds them, and then processes them with a standard transformer encoder. The output of the final layer's [CLS] token is used as the image representation for classification tasks. ViT has shown that, when pre-trained on large datasets and fine-tuned on smaller ones, it can outperform state-of-the-art CNNs in image classification tasks.

Swin Transformer. The Swin Transformer[9] introduces a shifted window-based self-attention mechanism, allowing for efficient and flexible local and global reasoning. The model hierarchically aggregates information across non-overlapping image patches, making it more computationally efficient. Swin Transformer has demonstrated strong performance across a range of vision tasks, from image classification to dense prediction tasks like object detection and semantic segmentation.

2.3 Related Researches for Places365

InternImage[13] proposed a novel architecture that combines the strengths of vision transformers with deformable convolutions. This hybrid approach aims to capture both local and global contextual information from images, making it particularly effective for scene recognition tasks. The deformable convolutions allow the model to adaptively sample feature points from the input, providing a more flexible way to capture spatial hierarchies in images. When combined with the self-attention mechanism of transformers, InternImage achieves state-of-the-art performance on the Places365 dataset.

WaveMix[5] introduces a new neural architecture tailored for computer vision tasks. It emphasizes resource efficiency while ensuring scalability and generalizability across different datasets and tasks. The architecture employs a mix of wavelet transformations and convolutional layers, allowing it to capture multi-scale features from images effectively. This multi-resolution analysis, combined with its unique mixing mechanism, enables WaveMix to achieve competitive performance on the Places365 dataset, even when compared to more complex and resource-intensive models.

3 PROPOSED WORK

In this study, I aim to explore scene recognition using the Swin Transformer architecture[9]. I believe that the self-attention feature of transformers can capture global patterns in images, making the Swin Transformer a suitable choice for scene recognition.

The Swin Transformer uses two key attention methods: Window-based Multi-head Self Attention (W-MSA) and Shifted Window-based Multi-head Self Attention (SW-MSA), as shown in Figure 4. The W-MSA works within set local windows, which makes attention computation more efficient. On the other hand, SW-MSA shifts these windows, allowing each part of the image to capture both nearby and wider context.

The Swin Transformer v2[8] is an updated version of the original. It changes the attention methods and replaces the Layer Normalization (LN) layer with the v2 Multi-head Self Attention (MSA). These changes aim to improve how the model works. For this study, I choose the Swin base model (Swin_b) as the main focus, with its structure shown in Figure 5.

To improve the model's performance, I use transfer learning. I start the model with weights from the Swin Transformer v2, which was trained on the ImageNet dataset[2]. This method aims to use the features learned from ImageNet to get better results on the Places365 dataset.

For a complete evaluation, I compare different models, including ResNet-50[4], Swin-b[9], Swin-v2-b[8], and ViT[3]. The goal is to see if models like the Swin Transformer, which can capture global context, can do better than traditional models like ResNet in scene recognition. This study will help understand how different models perform in image classification.

4 EVALUATION

To ensure a comprehensive assessment of the model's performance on the Places365 dataset, I have considered a combination of accuracy and loss metrics. This combination offers both a high-level

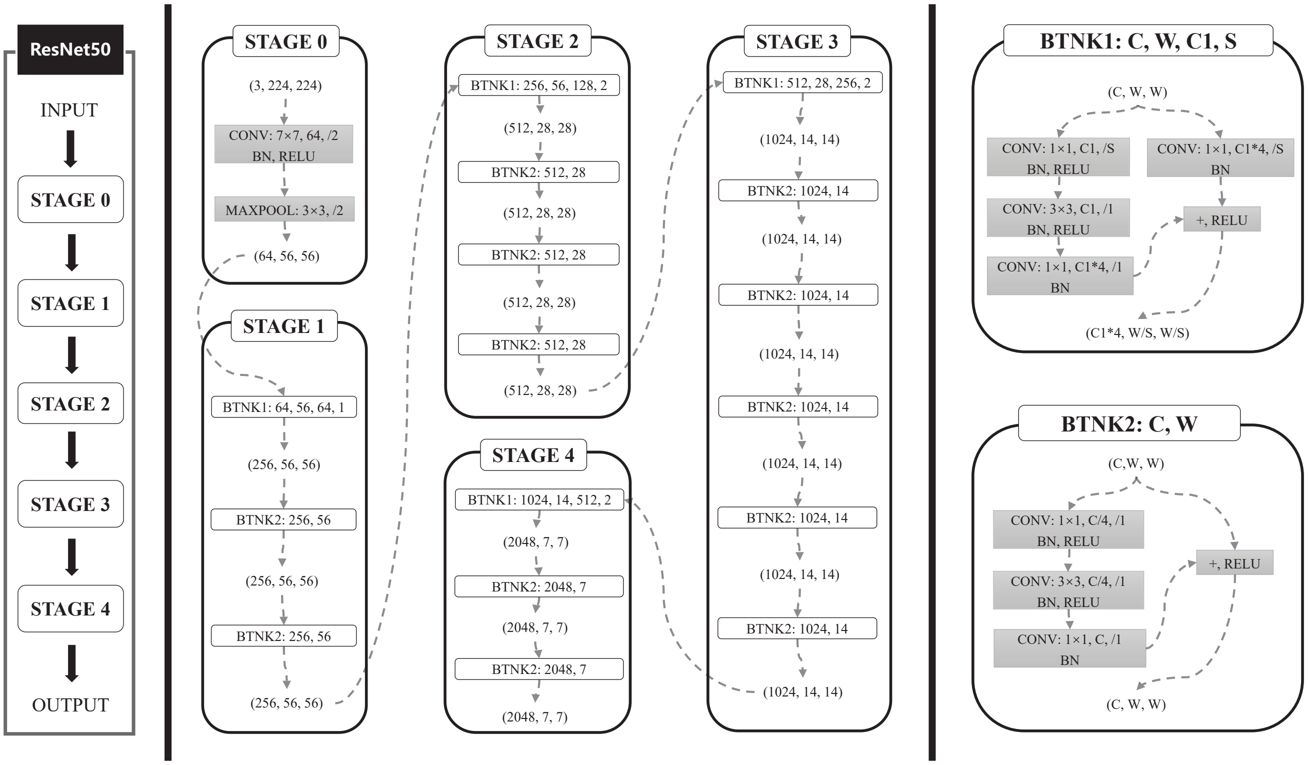
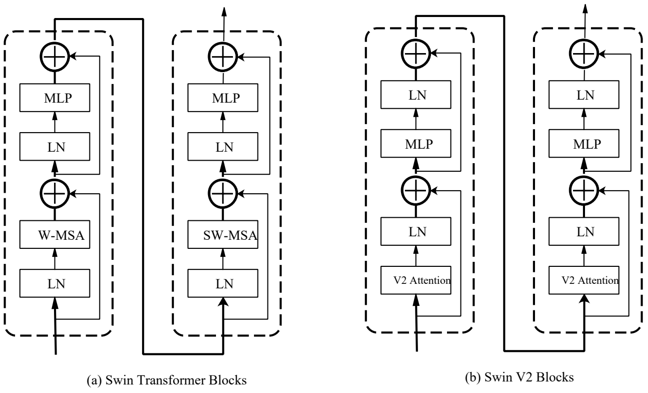


Figure 3: The architecture of ResNet-50[4][1]. 'C' denotes channels, 'W' represents both width and height (assuming a square image), and 'S' signifies stride. The blocks BTNK1 and BTNK2 are bottleneck blocks, with BTNK1 altering the input dimensions while BTNK2 retains them.



(a) Swin Transformer Blocks

(b) Swin V2 Blocks

Figure 4: Structure of Swin Transformer Blocks.

view of the model's classification capabilities (via accuracy) and insight into how well the training process minimized prediction errors (via loss).

4.1 Metrics

Four main metrics are been used in this assessment, include top-1 accuracy, top-5 accuracy, training loss and validation loss. All the losses are computed using cross entropy loss from PyTorch framework[10].

- `Top-1 Accuracy`: This metric measures the proportion of times the model's highest-confidence prediction matched the ground truth. It quantifies the model's capability to correctly predict the scene category directly without considering other high-confidence predictions.
- `Top-5 Accuracy`: more lenient evaluation metric, Top-5 Accuracy measures how often the true label for an image is within the model's top 5 predicted labels. This metric is especially valuable for datasets with a large number of categories, such as Places365, where a prediction might be deemed "close" if it's among the top few predictions, even if it's not the topmost.
- `Training Loss`: This represents the model's prediction error on the training dataset. Monitoring training loss helps us understand how well the model fits the training data over time and iterations. A decreasing training loss typically signifies learning, but an excessively low training loss can also hint at overfitting. The loss function used is cross entropy, the general equation is following:

$$L(y, p) = -\sum_i y_i \log(p_i) \tag{2}$$

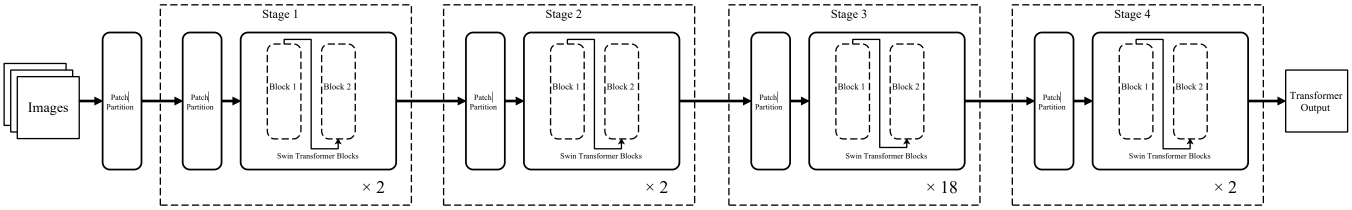


Figure 5: The Swin Transformer (Swin-B) design. The input image is processed through four stages of Swin Transformer Blocks. Each stage has a different number of blocks. The final output captures the main features of the input.

where $L(y, p)$ is the cross entropy loss between the true labels and predicted probabilities. y_i represents the true label of the i^{th} class, typically 0 or 1 in a classification setting. p_i is the predicted probability that an instance belongs to the i^{th} class. The sum $\sum_i$ is over all classes in the classification task. And log is the natural logarithm.

- `Validation Loss`: The prediction error on the validation dataset. It offers insight into how the model generalizes to new, unseen data. A significant gap between training loss and validation loss often indicates overfitting, where the model performs well on the training data but struggles with new data.
- `Training Time`: This metric quantifies the computational efficiency of the model by measuring the time required to complete one training epoch. It serves as an important factor when comparing the feasibility of different models for real-world applications. This information is crucial for understanding the trade-offs between model performance and computational resources.

4.2 Experimental setup

The Places365 dataset was used for this study, which includes 1,803,460 training images and 365,000 validation images. To maintain consistency, all images were processed to match the standards of the ImageNet[2] dataset.

During data preparation, images were randomly cropped and resized to 224x224 pixels. Some images were also flipped horizontally or rotated slightly. Additionally, the brightness, contrast, saturation, and hue of the images were adjusted. The images were then normalized based on set mean and standard deviation values for the Red, Green, and Blue channels. This step is important for stable training.

Training was conducted on an Nvidia RTX 4090 GPU. Two training approaches were explored: one using the full dataset and another using only 10% of the data with adjusted parameters.

The training configurations were:

- **Fully trained:** Used the entire dataset for 20 epochs.
- **Partial trained:** Used 10% of the dataset for up to 20 epochs.
- **Models for full training:** Included Swin_b[9], Swin_v2_b[8], ResNet-50[4], and Vision Transformer (ViT)[3].
- **Models for partial training:** Included Swin_v2_b[8] and ResNet-50[4].

Due to limited computing resources, only the Swin_b model was trained for an extended 266 epochs to achieve the highest possible accuracy.

In the process of hyperparameter tuning, I utilized an innovative method proposed by Smith et al.[11], which is based on cyclical learning rates. This technique offers an alternative to the conventional method of manually determining the best learning rate values and schedules. Instead, it allows the learning rate to oscillate between certain boundary values. This method has been demonstrated to boost classification accuracy without exhaustive tuning and often leads to faster convergence. To establish these boundary values, the learning rate was incrementally raised over several epochs. Implementing this approach is expected to refine the training process and enhance performance on the Places365 dataset.

5 DISCUSSION

5.1 Results

The performance differences among various models are tabulated in Table 1. Notably, ResNet-50[4], a conventional CNN architecture, delivers superior results over models that leverage transformer architecture for this specific task. This outcome implies that, for the given task, transformers may not necessarily offer a distinctive edge. They might necessitate additional epochs to achieve a performance comparable to a CNN architecture.

In Figure 6, a visual representation of the models' accuracy is provided. From the figure, it can be observed that Swin_v2[9] offers performance metrics close to that of ResNet-50[4]. Nonetheless, when evaluating the training time, as illustrated in Figure 7, it becomes evident that the training duration for Swin_v2[8] surpasses that of ResNet-50[4]. Given these results, one can argue that Swin_v2[8], might not be the optimal choice for this task, especially when compared to ResNet approaches.

From the results, it's important to note a possible limitation or characteristic of the Swin[8] model. The transformer-based architecture inherent in Swin may require a more extensive number of epochs for training to achieve its peak performance. While CNNs like ResNet-50 seem to converge to optimal accuracy faster, the Swin model might benefit from extended training cycles. Thus, for applications where extended training durations are permissible, Swin could still be considered a viable choice.

5.2 Training Trend Analysis

The training trends of three fully trained models, Swin_v2, ViT, and ResNet-50, were analyzed to assess their convergence rates and overall performance.

As seen in Figure 8, both Top 1 and Top 5 accuracy trends move in a similar manner, though at different percentages. Due to this

Table 1: Performance of Different Models

Models	Top 1 Accuracy	Top 5 Accuracy
swin_v2_part	0.42493	0.75589
swin_v2	0.51460	0.82625
resnet50_part	0.36951	0.67241
resnet50	0.52499	0.83159
resnet50	0.46860	0.77825
swin	0.42203	0.74022

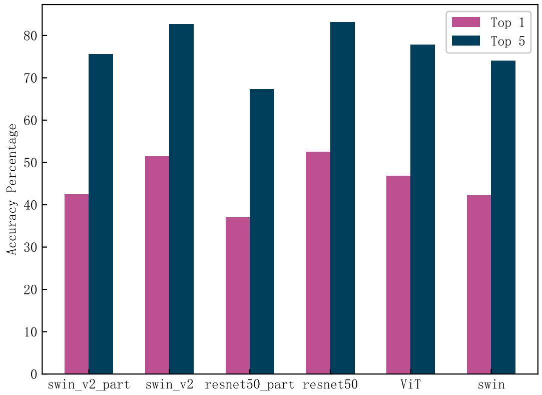


Figure 6: Accuracy of Different Models

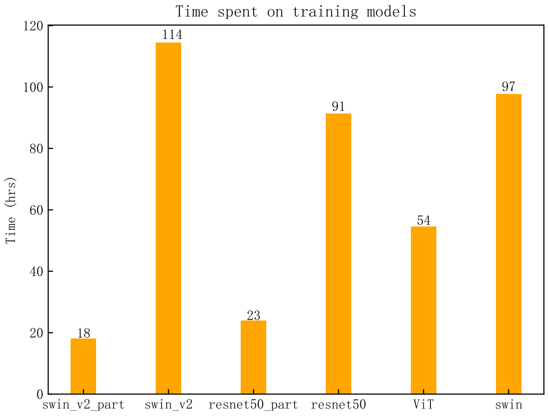


Figure 7: Training Time of Different Models

consistent pattern, further analysis can primarily focus on Top 1 accuracy, which offers a direct measure of the model's performance.

In Figure 9, ResNet-50 displays a faster convergence compared to Swin_v2, which takes more epochs to reach similar accuracy levels. This observation aligns with the earlier discussion that Swin_v2 might require more training epochs to fully realize its potential. On the other hand, ViT's trend is also noteworthy, providing a comparative reference between traditional CNNs and transformer-based models in the context of training dynamics.

To sum up, the analysis suggests that while all models eventually achieve comparable performance, ResNet-50 converges faster than Swin_v2. This reinforces the idea that certain transformer-based models like Swin_v2 might benefit from extended training. Meanwhile, ViT provides an intermediate performance, bridging the gap between CNNs and other transformer models in training speed.

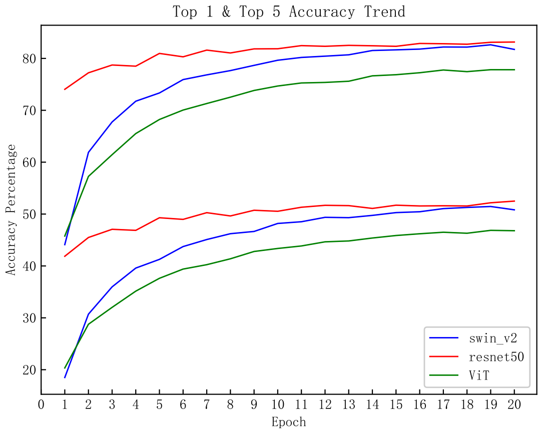


Figure 8: Training Trend for Top 1 and Top 5 Accuracy

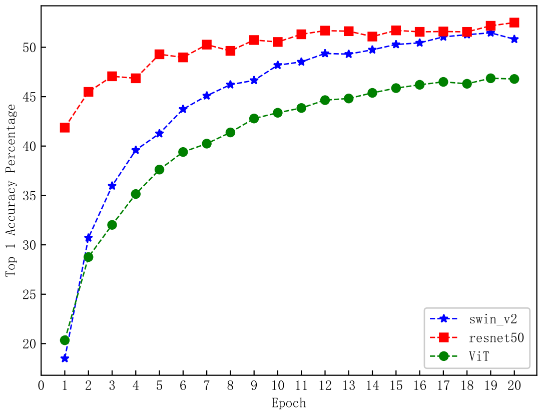


Figure 9: Training Trend for Top 1 Accuracy

5.3 Analysis of Swin Model

The Swin_b model was trained for a period exceeding 14 days. The results, as seen in Figure 10, show that the Top 1 accuracy of the model is approximately 58.679% and Top 5 accuracy of the model is approximately 87.989%. This performance is noteworthy as it is better than the model used in WaveMix[5](56.45%).

The graph in Figure 10 shows how the accuracy of the Swin_b model changes over 250 epochs. There is a quick rise in accuracy in the first 50 epochs. After this, the accuracy goes up slowly and levels off towards the end. This is common for many deep learning models. At first, they learn quickly, and then the improvements slow down as they continue training.

The good performance of the Swin_b model compared to traditional CNN models suggests that transformer models might be very useful for tasks like scene recognition. Transformers can understand patterns in images in a way that other models might not.

In conclusion, the results reaffirm the idea that transformer architectures will perform well in the domain of scene recognition. Their performance makes them a promising method for future research and applications.

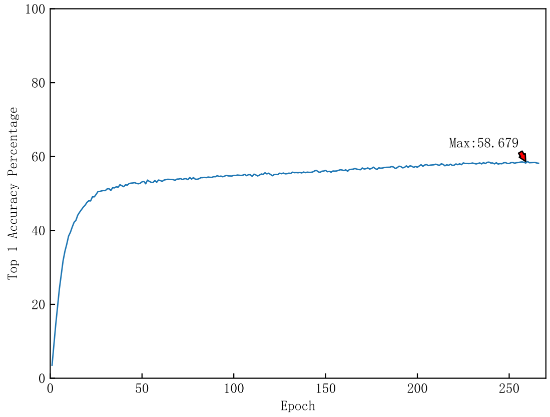


Figure 10: Training Trend for Swin_b Model Over Extended Epochs

5.4 Timeline

To ensure the timely completion of this research, the following one-month timeline needs to be checked and updated:

- **Week 1:** Data Preprocessing and Initial Model Training (Finished)
- **Week 2:** Model Tuning and Optimization (Finished)
- **Week 3:** Evaluation and Comparison with Baseline Models (Finished)
- **Week 3 Update:** Some models need more epochs to train, and the result will be further analyzed after training. (Finished)
- **Week 4:** Final Analysis, Writing, and Submission (Finished)

5.5 Challenges and Lessons

During this research, I identified a few challenges. First, the Swin Transformer needs a lot of computer power because of its detailed design. One solution I thought of was using cloud-based GPUs, which can handle more data. I also looked into ways to simplify the model without losing its quality.

Another challenge was that the Places365 dataset might have more of some classes than others. This can make the model favor those classes. A possible fix for this is data augmentation, where I'd make new versions of current images. I also considered re-sampling to balance the dataset. Another approach is to use other dataset, like COCO[7], ImageNet[2], and ADE20K[15] to pretrain the models, or use private datasets for specific projects.

Lastly, the Swin Transformer might not always give the best results quickly because of its design. To address this, I thought about using different ways to optimize the model and changing the learning rates for faster outcomes.

6 CONCLUSION

6.1 Key Findings

The initial results of this research highlight a few important points. The Swin Transformer showed better accuracy in recognizing scenes compared to other standard models. This suggests that it can detect detailed patterns in images that other models might miss.

A standout feature of the Swin Transformer is its ability to grasp the overall context of an image using its self-attention mechanism. However, a balance between its computational needs and performance was observed. While it provides impressive results, its high computational needs might not be suitable for real-time use or in situations with limited resources.

6.2 Future Works

The results open up several areas for further exploration. It would be interesting to see how different attention mechanisms affect scene recognition. Since attention plays a key role in transformers, understanding its variations could lead to better models.

The potential uses of the Swin Transformer in real-world settings, like surveillance, are also worth exploring. In surveillance, getting a full picture of a scene is vital. Augmented reality is another area where improved scene recognition could be beneficial. As technology advances, combining visual data with other types of information, like sound or text, could lead to even better scene recognition tools.

6.3 Project Summary

This study aimed to see how well the Swin Transformer could recognize scenes, using the Places365 dataset as a test. The early results are encouraging, showing the model's strengths in understanding complex images. But, as with any study, there were challenges, from the need for high computing power to issues with the dataset. As the importance of scene recognition increases in our digital world, models like the Swin Transformer will likely play a big role. This study is just the beginning, pointing to what's possible and hinting at future research in this area.

REFERENCES

[1] Chouxianyu. 2021. ResNet50 Architecture. https://zhuanlan.zhihu.com/p/353235794

[2] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*. 248–255. https://doi.org/10.1109/CVPR.2009.5206848

[3] 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. 2021. An

Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. arXiv:2010.11929 [cs.CV]

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Deep Residual Learning for Image Recognition. arXiv:1512.03385 [cs.CV]

[5] Pranav Jeevan, Kavitha Viswanathan, Anandu A S, and Amit Sethi. 2023. WaveMix: A Resource-efficient Neural Network for Image Analysis. arXiv:2205.14375 [cs.CV]

[6] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. 1998. Gradient-based learning applied to document recognition. *Proc. IEEE* 86, 11 (1998), 2278–2324. https://doi.org/10.1109/5.726791

[7] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Doll'a r, and C. Lawrence Zitnick. 2014. Microsoft COCO: Common Objects in Context. *CoRR* abs/1405.0312 (2014). arXiv:1405.0312 http://arxiv.org/abs/1405.0312

[8] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, Furu Wei, and Baining Guo. 2021. Swin Transformer V2: Scaling Up Capacity and Resolution. *CoRR* abs/2111.09883 (2021). arXiv:2111.09883 https://arxiv.org/abs/2111.09883

[9] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. 2021. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. *CoRR* abs/2103.14030 (2021). arXiv:2103.14030 https://arxiv.org/abs/2103.14030

[10] PyTorch. 2023. CrossEntropyLoss. https://pytorch.org/docs/stable/generated/torch.nn.CrossEntropyLoss.html

[11] Leslie N. Smith. 2015. No More Pesky Learning Rate Guessing Games. *CoRR* abs/1506.01186 (2015). arXiv:1506.01186 http://arxiv.org/abs/1506.01186

[12] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. *CoRR* abs/1706.03762 (2017). arXiv:1706.03762 http://arxiv.org/abs/1706.03762

[13] Wenhai Wang, Jifeng Dai, Zhe Chen, Zhenhang Huang, Zhiqi Li, Xizhou Zhu, Xiaowei Hu, Tong Lu, Lewei Lu, Hongsheng Li, Xiaogang Wang, and Yu Qiao. 2023. InternImage: Exploring Large-Scale Vision Foundation Models with Deformable Convolutions. arXiv:2211.05778 [cs.CV]

[14] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. 2017. Places: A 10 million Image Database for Scene Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2017).

[15] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. 2018. Semantic Understanding of Scenes through the ADE20K Dataset. arXiv:1608.05442 [cs.CV]