Name: Shivam Singh

Roll Number: 22051620

Section: CSE 24

Computer Vision

From Text to Pixels: An Analysis of the Transformer Architecture and its Evolution in Here is the text for your first page. You can copy and paste this into your document:

Abstract: This report analyzes the Transformer architecture, from its 2017 introduction in natural language processing to its adaptation in computer vision. It details the encoder-decoder stacks and the self-attention mechanism. The report then explores the Vision Transformer (ViT), comparing it conceptually and computationally against Convolutional Neural Networks (CNNs), focusing on *inductive bias*. Finally, it examines ViT's limitations—quadratic computational complexity and its "data-hungry" nature—and details solutions from successor models like the Swin Transformer and Data-efficient Image Transformers (DeiT).

The Foundational Transformer Architecture

A. Introduction: A Paradigm Shift "Beyond Recurrence"

Introduced in 2017 by "Attention Is All You Need," the Transformer architecture revolutionized sequence-based tasks by replacing recurrent and convolutional networks with a self-attention mechanism. This shift enabled parallel processing, leading to faster training and the ability to train larger models.

B. The Encoder Stack: Understanding the Input

The Transformer's encoder takes an input sequence and maps it into a "contextualized encoding sequence." It consists of a stack of identical layers, each with two sub-layers:

1. **Multi-Head Self-Attention:** Allows each word (token) to draw context from all other words in the sequence.

2. **Position-wise Feed-Forward Network:** Refines the output of the attention layer.

C. The Decoder Stack: Generating the Output

The decoder generates an output sequence from the encoder's contextualized representations. It also consists of a stack of identical layers, but with three sub-layers:

1. **Masked Multi-Head Self-Attention:** Ensures the model only attends to preceding tokens when generating output.
2. **Encoder-Decoder Attention:** Bridges the input and output, allowing the decoder to consult the encoded input.
3. **Position-wise Feed-Forward Network:** Similar to the encoder's feed-forward network.

D. Inputs: Embeddings and Positional Encoding

To address the attention mechanism's order-agnostic nature, **Positional Encoding** is used. Numerical embedding vectors for each token are augmented with positional encoding vectors, which are generated by fixed sine and cosine functions, providing the model with a sense of sequence order.

The Mechanism of Attention: The Core Engine

A. Scaled Dot-Product Attention: The Q, K, V Analogy

The attention function maps a query and a set of key-value pairs to an output. This can be understood as:

- **Value (V):** The actual content to retrieve.
- **Key (K):** The label or index for its corresponding Value.
- **Query (Q):** The current word looking for relevant information.

Scaled Dot-Product Attention involves four steps:

1. **Score:** Computes the dot product of the Query with every Key to measure relevance.
2. **Scale:** Divides scores by $\sqrt{d_k}$ to prevent large values.
3. **Softmax:** Converts scaled scores into attention weights (probabilities).
4. **Output:** A weighted sum of all Value vectors, weighted by their attention weights.

B. Multi-Head Attention: Learning in Parallel

Multi-Head Attention runs multiple Scaled Dot-Product Attention mechanisms in parallel. Each "head" learns different types of relationships by transforming input embeddings into "sub-queries, sub-keys, and sub-values." The outputs of all heads are concatenated and combined, leading to a richer representation.

Vision Transformers (ViT): Applying Transformers to Images

A. The Core Concept: "An Image Is Worth 16x16 Words"

In 2020, the Vision Transformer (ViT) demonstrated that CNNs were not essential for computer vision by applying a "pure transformer directly to sequences of image patches." ViT treats an image as a 1D sequence of "words."

B. The ViT Pipeline: From Image to Sequence

ViT preprocesses 2D images for the 1D Transformer through:

1. **Image Patching:** Splitting the image into non-overlapping patches (e.g., 16x16 pixels).
2. **Patch Embedding:** Flattening each 2D patch into a 1D vector and projecting it linearly to create a "patch embedding" (token).
3. **Positional Encoding:** Adding learnable positional embeddings to patch embeddings to retain spatial information.
4. **The CLS Token:** A special classification token prepended to the sequence, which aggregates information from all patches to form a holistic image summary for final prediction.

Comparative Analysis: Vision Transformers vs. Convolutional Neural Networks

A. The Conceptual Divide: Inductive Bias

The main difference between ViT and CNNs is their **inductive bias**:

- **CNNs:** Have a *strong* inductive bias, including locality (nearby pixels are related) and translation equivariance (patterns recognized regardless of position).
- **ViTs:** Have a *weak* inductive bias, learning relationships between any two patches and requiring training data to learn translation equivariance. This weak bias makes ViT "data-hungry," requiring massive pre-training.

B. The Computational Divide: Scaling and Efficiency

- **CNNs:** Computational cost scales *linearly* ($O(N)$) with the number of input pixels.
- **ViTs:** Self-attention has a computational complexity of $O(N^2)$, where N is the

number of *patches*. This quadratic scaling makes pure ViT impractical for high-resolution images.

Table 1: Conceptual and Computational Comparison: ViT vs. CNN

Aspect	Convolutional Neural Networks (CNNs)	Vision Transformers (ViTs)
Inductive Bias	Strong (Locality, Translation Equivariance)	Weak (Learns all relationships from data)
Receptive Field	Local (starts small, grows with layers)	Global (Full image from layer 1)
Data Requirement	Can work well with small datasets	"Data-hungry," needs massive pre-training
Computational Complexity	Linear ($O(N)$) w.r.t. pixels	Quadratic ($O(N^2)$) w.r.t. patches
Performance on Small Data	Often better (less overfitting)	Often worse (overfits)
Performance on Large Data	Performance saturates	Scales very well with data and model size

Limitations and Evolutions: Addressing the Challenges of ViT

A. Swin Transformer: Solving Quadratic Complexity

The **Swin Transformer** addresses the $O(N^2)$ computational bottleneck by re-introducing:

1. **Hierarchical Architecture:** Builds a feature pyramid, merging patches to decrease spatial resolution and increase channel depth.
2. **Windowed Attention (W-MSA):** Replaces global attention with attention limited to non-overlapping local windows, reducing complexity to linear ($O(N)$). It uses **Shifted Window MSA (SW-MSA)** to allow cross-window connections.

B. DeiT: Solving the Data-Hungry Problem

The **Data-efficient Image Transformer (DeiT)** tackles the data-hungry problem using **Knowledge Distillation**, a "teacher-student" strategy. It introduces a **distillation token** that learns from a pre-trained "teacher" network (often a ConvNet), effectively injecting convolutional spatial inductive biases into the Transformer.

Conclusion: The Convergent Future of Vision

The Transformer fundamentally changed sequence processing, and its adaptation to computer vision with ViT challenged CNN dominance. However, ViT's quadratic complexity and data-hungriness spurred the development of new architectures. The **Swin Transformer** re-introduces CNN-like principles to solve complexity, while **DeiT** "distills" CNN's inductive bias to solve the data problem. The future of computer vision is a *convergent* one, with state-of-the-art models marrying the efficient feature extraction of CNNs with the global context-modeling of Transformers.

Cited Research Papers

1. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N.,... & Polosukhin, I. (2017). "Attention Is All You Need." *Advances in Neural Information Processing Systems 30 (NIPS)*.
2. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T.,... & Houlsby, N. (2021). "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale." *International Conference on Learning Representations (ICLR)*.
3. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z.,... & Guo, B. (2021). "Swin Transformer: Hierarchical Vision Transformer using Shifted Windows." *arXiv preprint arXiv:2103.14030*.
4. Touvron, H., Cord, M., Douze, M., Massa, F., Le, G., & Jégou, H. (2021). "Training data-efficient image transformers & distillation through attention." *arXiv preprint arXiv:2012.12877*.