CS331. Thị giác máy tính nâng cao

Vision Transformers & Ứng dụng

Phần A – Giới thiệu & Đặt vấn đề

Vision Transformers & Kiến trúc hiện đại

- **Vision Transformer (ViT).**
- CNN: 2012–2019
- Transformer: 2020 - đến nay.

Tại sao cần mô hình mới?

- CNN rất mạnh, nhưng có hạn chế:
- Receptive field chỉ cục bộ, khó nắm quan hệ dài hạn.
- Muốn học toàn cục phải chồng nhiều layer → tốn kém.
- Khó mở rộng.

- Trong khi đó, NLP đã thành công với Transformer. → liệu ta có thể mang Transformer vào thị giác?”

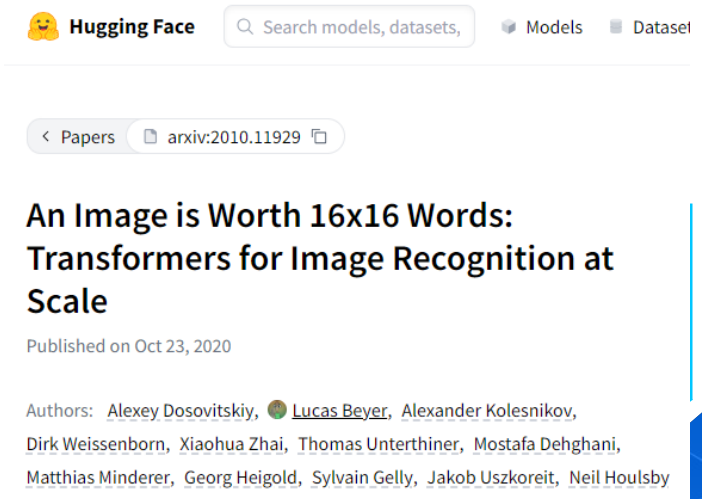
So sánh CNN và Transformer

- CNN: tốt cho local patterns (edge, texture).
- Transformer: tốt cho global dependencies.

- Trong NLP, Transformer thay thế RNN hoàn toàn.
- Trong vision, sự thay thế này đang diễn ra nhanh chóng.

Mốc lịch sử

- 2017: Paper ‘Attention is All You Need’ ra đời.
- 2020: Google giới thiệu Vision Transformer (ViT).
- 2021: Facebook/Meta phát triển DeiT, DINO.
- 2022–2025: hàng loạt biến thể ViT, kết hợp geometry, multimodal, self-supervised.



Ứng dụng của Transformer trong Vision

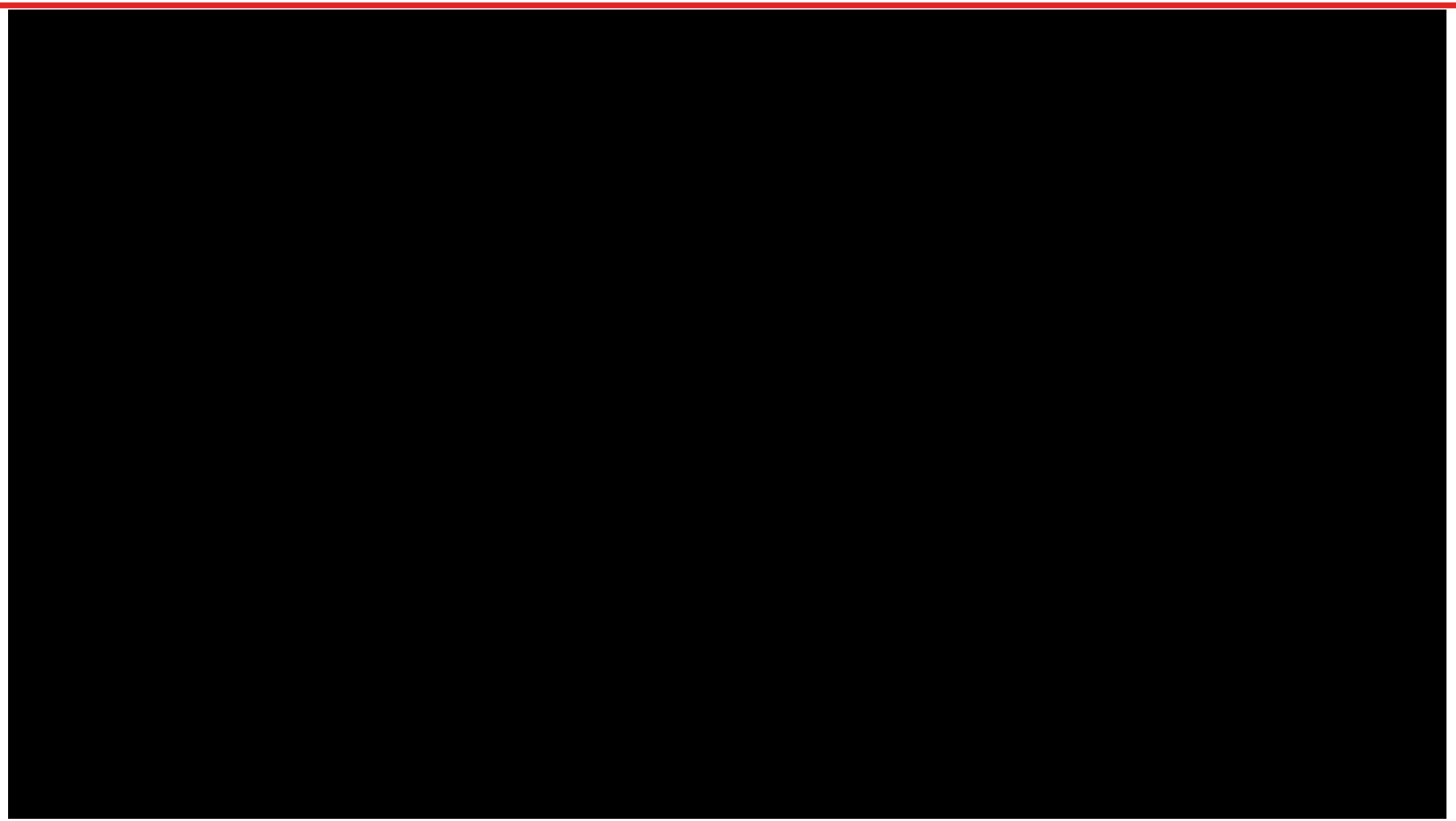
- Classification.
- Detection (DETR).
- Segmentation (Segmenter, MaskFormer).
- Multimodal (CLIP, Flamingo).
- 3D Vision (VGGT).

Cấu trúc bài học

- Kiến trúc ViT cơ bản.
- Các cải tiến (DeiT, Hybrid CNN-Transformer).
- Self-supervised Transformers (DINO, MAE).
- Ứng dụng hiện đại: VGGT (CVPR 2025).
- So sánh CNN vs ViT, thảo luận.

Phần B – Vision Transformer cơ bản

Ý tưởng chính của ViT



Ý tưởng chính của ViT

Ý tưởng chính của ViT

- Thay vì quét ảnh bằng CNN, ViT chia ảnh thành patch (ví dụ 16×16).
- Mỗi patch được biến thành vector, giống như token trong NLP.
- Sau đó, đưa chuỗi token vào Transformer encoder

Quá trình xử lý của ViT

- Chia ảnh thành patch.
- Linear embedding từng patch.
- Thêm positional encoding.
- Transformer encoder (multi-head attention + feed-forward).
- Class token → classification head.

Patch Embedding

- Mỗi patch 16×16×3 được flatten thành vector, rồi nhân với ma trận trọng số để ra embedding. Đây tương tự như từ vựng (word embedding) trong NLP

Positional Encoding

- Transformer không có tính chất không gian tự nhiên, nên phải thêm positional encoding để mô hình biết patch nào ở vị trí nào.

Multi-Head Self Attention (MHSA)

- Phần quan trọng nhất của Transformer là self-attention.
- Công thức:

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

→ Ý nghĩa: mỗi patch nhìn được toàn bộ patch khác.

- Multi-head = nhiều không gian attention song song.

Layer Norm và MLP block

Mỗi layer của ViT gồm:

- Multi-head attention.
- Layer norm.
- Feed-forward MLP.

→ Residual connection giúp gradient ổn định.

Class Token

- ViT thêm một token đặc biệt [CLS].
- Sau khi qua Transformer, vector [CLS] đại diện cho toàn ảnh, dùng cho classification

Ưu điểm của ViT

- Học global context dễ dàng.
- Kiến trúc linh hoạt (dễ kết hợp multimodal).
- Khi dữ liệu đủ lớn → vượt CNN.

Hạn chế của ViT

- Cần dữ liệu cực lớn để huấn luyện từ đầu.
- Tốn tài nguyên tính toán.
- Ít inductive bias hơn CNN → khó học từ dữ liệu nhỏ.

Kết quả ban đầu của ViT

- Trên ImageNet, ViT huấn luyện từ đầu kém hơn ResNet.
- Nhưng khi pretrain trên dataset cực lớn (JFT-300M), ViT vượt trội.

- → xu hướng mô hình lớn

Phần C – Các cải tiến của ViT

DeiT (2021)

- Data-efficient Image Transformer.
- Ý tưởng: dùng distillation từ CNN teacher để huấn luyện ViT với dữ liệu ít hơn.
- DeiT chứng minh ViT có thể cạnh tranh với ResNet chỉ với ImageNet

Hybrid CNN-Transformer

- Kết hợp CNN cho low-level features, Transformer cho high-level.
- Ví dụ: CMT, CvT.
- Ý tưởng:
  - CNN vẫn tốt ở biên, texture;
  - Transformer mạnh ở context.

Hierarchical ViTs

- ViT cơ bản xử lý patch cố định. Nhưng nhiều nghiên cứu đề xuất hierarchical structure, giống CNN pyramid.
- Ví dụ: Swin Transformer.
- Ưu điểm: xử lý ảnh kích thước lớn, hiệu quả hơn.”

Swin Transformer (2021)

- Swin dùng ‘shifted window attention’: chỉ tính attention trong cửa sổ nhỏ, rồi dịch cửa sổ.
- Kết quả: giảm độ phức tạp, tăng hiệu quả cho detection/ segmentation.

DeiT III & các biến thể mới

- DeiT III: training recipe tối ưu.
- ViT-G (Giant): hàng tỷ tham số.
- ConvNeXt: CNN nhưng học từ ViT design.

Ứng dụng ViT trong detection (DETR)

- DETR: Detection Transformer (2020).
- Không cần anchor box, chỉ dùng query + attention.
- Kết quả: đơn giản hơn Faster R-CNN, hiệu quả cao

Ứng dụng ViT trong segmentation

- Segmentation models: Segmenter, MaskFormer, Mask2Former.
- Transformer cho phép segmentation dựa trên attention map, linh hoạt hơn CNN.

ViT trong multimodal (CLIP)

- CLIP dùng ViT để encode ảnh, Transformer để encode text.
- Kết quả: embedding chung cho image-text.
- Đây là nền tảng của vision-language models.

Phần D – Self-Supervised Transformers

DINO: Distillation with No Labels.

- ViT backbone.
- Self-supervised training, không cần nhãn.

→ attention map tự động tập trung vào object, dù không có nhãn.

DINOv2 (2023–2024)

- Huấn luyện trên dataset khổng lồ.
- Representation tốt → dùng được cho downstream task không cần fine-tuning.

MAE (Masked Autoencoder)

- MAE: Self-supervised pretraining cho ViT.
- Che 75% patch.
- Encoder xử lý phần còn lại.
- Decoder tái dựng ảnh.

→ Kết quả: giảm chi phí, học representation tốt

Kết hợp SSL + ViT

- Kết hợp SSL + ViT → backbone: DINOv2, MAE, EVA.

So sánh SSL CNN vs SSL ViT

- SSL với CNN (SimCLR, MoCo) → representation tốt.
- SSL với ViT (DINO, MAE) → representation mạnh hơn, general hơn.

→ Xu hướng hiện tại: ViT + SSL

Bài tập

- Thực hiện phân loại ảnh dùng ViT
- Dataset: HoaVietNam

- https://www.geeksforgeeks.org/computer-vision/vision-transformers-vit-in-image-recognition/
- https://keras.io/examples/vision/image_classification_with_vision_transformer/
- https://github.com/Naveenpandey27/Image_Classification_using_Vision_Transformer

Tài liệu tham khảo

- An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale (Paper Explained)
- Vision Transformer (ViT)
- Vision Transformer Basics
- https://www.geeksforgeeks.org/deep-learning/vision-transformer-vit-architecture/
- https://vinbigdata.com/camera-ai/tong-quan-ve-vision-transformer-vit.html