Investigation of Attention Mechanism for Multimodal Transformers

Rushi Moliya*, *AU2240020*, Shlok Shelat*, *AU2240025*, Shrey Salvi*, *AU2240033*
*School of Engineering and Applied Science, Ahmedabad University, Ahmedabad 380009, India
Emails: {rushi.m, shlok.s1, shrey.s2}@ahduni.edu.in

***Abstract*—This work explores the optimization of attention mechanisms in the CLIP model to improve text-based person search performance and efficiency. Through the combination of two image encoders—ResNet-50 and FastViT-T8—and a DistilBERT text encoder, the research compares self-attention, cross-attention, and sparse attention on the Flickr30k and RSTP-ReID datasets. Experimental findings demonstrate that cross-attention with the FastViT-T8 encoder results in better performance, with a Recall@10 of 100.00 % on both datasets, accompanied by lower memory consumption of 1189.0 MB and a training time of 992.59, seconds for Flickr30k. Compared to ResNet-50 with cross-attention, which performs well but has a higher computational cost, sparse attention is not practical, with inference times standing at 1.7727 seconds and the highest memory usage at 5378.7 MB on RSTP-ReID. The results point out that cross-attention combined with FastViT-T8 achieves an ideal trade-off between accuracy and efficiency, rendering it ideal for practical use. Future work may investigate hybrid attention mechanisms to further enhance multimodal transformer performance.**

***Index Terms*—CLIP, attention mechanisms, text-based person search, multimodal transformers, resource efficiency**

I. Introduction

Person reidentification (ReID) is a cornerstone problem in computer vision, fueled by its critical applications in surveillance, security, and human-computer interaction. The problem of matching and tracking individuals over different camera views or time frames is frequently impeded by real-world issues like changing illumination, non-rigid poses, occlusions, and cluttered backgrounds. These complications render ReID a challenging issue, challenging classical vision-based approaches. By integrating text descriptions, we open a new level of flexibility and resilience, allowing systems to find individuals from natural language queries—a capacity with great real-world potential.

This project investigates the improvement of text-based person search using the Contrastive Language-Image Pretraining (CLIP) model. It takes inspiration from previous work where CLIP was originally evaluated on the Flickr30k dataset, with ResNet-50 as an image encoder and DistilBERT as a text encoder. Such experiments gave baseline findings on image-text alignment, trying out self-attention and cross-attention mechanisms to match visual and text features. Successful though it was, such work only applied to a general-purpose dataset. Here, we apply the research further to tackle person-specific retrieval based on CLIP's multimodal capabilities in an even more directed and demanding setup.

The demand for this exploration arises from the shortcomings of collections such as Flickr30k, which, being effective for generic image-caption scenarios, are deficient in specificity towards person reidentification. To better fill this lacuna, we resort to the RSTP-ReID collection, carefully prepared for person-focussed retrieval. RSTP-ReID provides a more appropriate benchmark, closely matching real-world ReID scenarios and enabling us to test CLIP's performance within an environment that requires precision and flexibility. This change is essential to determining just how well text-based approaches can perform when they are adapted to the complexities of person identification.

The project's contributions are applied and novel, pushing the field of person search with text-based inputs in three areas:

1) **RSTP-ReID Evaluation**: We evaluate the ability of CLIP on the RSTP-ReID dataset, offering a comprehensive analysis of its performance under a person-specific retrieval scenario. This evaluation provides novel insight into how well pre-trained multimodal models transfer to specialized tasks. Endian.
2) **Introduction of FastViT-T8**: In place of ResNet-50, we introduce FastViT-T8 as a computationally efficient image encoder. The speed-optimal vision transformer, with heightened global context perception, improves computational efficiency while maintaining accuracy—essential for employing ReID systems in resource-scarce environments.
3) **Incorporation of Sparse Attention**: To address scalability, we incorporate sparse attention into the text encoder. This optimization decreases memory and computational requirements, making the model more feasible for large datasets and real-time usage while maintaining retrieval effectiveness.

This work starts off by summarizing our previous experiments as a primer to ground the reader, before shifting into aggressive new goals pushing the limits of efficiency and specificity for person search. The appeal comes from the applicability of RSTP-ReID, FastViT-T8 as the potential for reduced computation, and sparse attention for handling scalability—a strong combined case for theoretical innovation and real-world application. By linking previous findings to these future-oriented objectives, this project not only enhances our knowledge of multimodal systems but also lays the groundwork for more deployable and efficient text-based ReID

solutions.

II. Background

In this section, we give a general overview of the main elements and ideas that are the backbone of this project. These are the CLIP model, the text encoder (DistilBERT), the image encoders (ResNet-50 and FastViT-T8), the tested attention mechanisms (self-attention, cross-attention, and sparse attention), and the training and testing datasets (Flickr30k and RSTP-ReiD).

A. CLIP Model

The Contrastive Language-Image Pre-training (CLIP) model [1] is a multimodal model designed to align images and text in a shared embedding space. It uses a dual-encoder architecture, consisting of an image encoder and a text encoder, which are trained jointly using a contrastive loss function. This approach enables the model to learn associations between visual and textual data, making it effective for tasks such as text-based image retrieval. In this project, we leverage CLIP's ability to match textual descriptions with corresponding images to perform text-based person search.

B. Text Encoder: DistilBERT

The text encoder used in our CLIP model is DistilBERT [7], a distilled version of the BERT (Bidirectional Encoder Representations from Transformers) model [**?**]. DistilBERT retains approximately 97% of BERT's language understanding capabilities while being 60% faster and 40% smaller in terms of parameters. This efficiency makes it an ideal choice for encoding textual inputs in resource-constrained environments.

DistilBERT processes textual descriptions (e.g., captions or queries) into dense embeddings, which are then projected into the shared embedding space of the CLIP model. These embeddings are compared with image embeddings via cosine similarity to retrieve relevant images based on text queries. The use of DistilBERT ensures that the text encoding is both computationally efficient and semantically rich, enabling effective multimodal alignment.

C. Image Encoders

We compare two alternative image encoders in this project: ResNet-50 and FastViT-T8. These encoders are tasked with converting input images into feature representations that can be aligned with text embeddings in the common space.

1) ResNet-50: ResNet-50 [5] is a 50-layer convolutional neural network (CNN) that has been extensively used for image classification and feature extraction tasks. Due to its deep structure and residual connections, ResNet-50 is capable of learning high-level visual patterns and avoids vanishing gradients. In this project, ResNet-50 is used as a strong baseline image encoder because of its demonstrated success across many computer vision tasks.

2) FastViT-T8: FastViT-T8 is an efficient variant of the Vision Transformer (ViT)
citedosovitskiy2020image architecture. As opposed to CNNs, which exploit local convolutions, Vision Transformers operate on images as sequences of patches and apply self-attention to encode global dependencies. FastViT-T8 is computationally efficient with robust performance and, therefore, a strong alternative to the conventional CNN-based encoders such as ResNet-50.

D. Attention Mechanisms

Attention mechanisms are important to improve the performance of deep learning models as they enable them to pay attention to important portions of the input data. In this project, we compare three forms of attention mechanisms that are incorporated into the CLIP model: self-attention, cross-attention, and sparse attention.

1) Self-Attention: Self-attention [2] is an operation that enables a model to focus on different regions of an individual input sequence (e.g., text tokens or image features) to exploit internal relationships. In CLIP, self-attention can be used within either the text encoder (DistilBERT) or the image encoder (ResNet-50 or FastViT-T8) to improve the quality of each modality representations.

2) Cross-Attention: Cross-attention facilitates interaction between the two modalities, e.g., text and image features. By permitting the model to attend the appropriate regions in the image with reference to the text query (or vice versa), cross-attention may enhance alignment between the two modalities in the shared embedding space. Such a mechanism can be especially beneficial for applications such as text-based image retrieval, where exact matching between text and image content is essential.

3) Sparse Attention: Sparse attention [**?**] is a self-attention variant that minimizes computational cost by only attending to a subset of positions in the input sequence. This can be done using methods like local window attention or through decreasing the number of attention heads. Sparse attention is particularly useful for scaling up models to bigger inputs or for enhancing efficiency in real-time settings.

E. Datasets

We use two datasets in this project: Flickr30k for general image-text alignment and RSTP-ReiD for person-specific retrieval tasks.

1) Flickr30k: Flickr30k [**?**] is a popular image-caption dataset with 31,783 images, each associated with five descriptive captions. It is widely used for training and testing models on image-text alignment tasks. In this project, Flickr30k is the main dataset for initial training and testing of the CLIP model.

2) RSTP-ReiD: RSTP-ReiD [4] is a person re-identification dataset with 20,000 images of pedestrians taken in real-world environments. In contrast to typical image-caption datasets, RSTP-ReiD is specially tailored for person search applications, thus making it very suitable for testing the performance of the model in fetching person-specific images given textual descriptions.

III. Methodology

This section describes the methodology used in this research, describing dataset preparation, model setup, hyperparameters, training process, and evaluation metrics used to train and evaluate the CLIP-based model.

A. Dataset Preparation

Flickr30k and RSTP-ReID were the two datasets used in this project, both prepared with particular preprocessing processes to be compatible with the model.

Flickr30k: The data was loaded from a CSV file and split into 80% training and 20% validation sets. Images were resized to 224×224 pixels to normalize input sizes. Captions were tokenized with the DistilBERT tokenizer with a maximum sequence length of 64 tokens to balance detail and computational efficiency.

RSTP-ReID: The dataset was loaded from a JSON file, which provided predefined splits for training, validation, and testing. Images were similarly resized to 224×224 pixels. Captions, often longer in this dataset, were tokenized with a maximum sequence length of 200 tokens to preserve descriptive content.

B. Model Configuration

The architecture of the CLIP model incorporates dual image encoders—ResNet-50 and FastViT-T8—and a DistilBERT text encoder. Visual features are captured by the image encoders, and tokenized captions are processed by the text encoder. Three attention mechanisms improve the text encoder's ability:

- **Self-Attention**: Facilitates internal context understanding within captions.
- **Cross-Attention**: Enables interaction between text and image features, improving multimodal alignment.
- **Sparse Attention**: Optimizes computational efficiency by focusing on key input positions.

Embeddings from both encoders are projected into a shared space via projection heads, allowing for contrastive learning.

C. Hyperparameters

Hyperparameters used during training and inference are listed in Table I. These were optimised for the best model performance, with deviations specified for every dataset. The batch size and maximum text length used for Flickr30k was 32 and 64 respectively, whereas for RSTP-ReID a batch size of 48 and maximum text length of 200 was used, considering its nature. Experiments used two image encoders: ResNet-50 (2048 embedding dimension) and FastViT-T8 (768 embedding dimension). Learning rates were initialized lower for pretrained encoders (1e-4 for image, 1e-5 for text) to maintain learned features, and higher (1e-3) for projection heads to allow adaptation.

TABLE I: Hyperparameters for CLIP Model Training and Inference

Parameter	Value
Batch size	16 / 32 / 48
Number of workers	4
Head learning rate	1e-3
Image encoder learning rate	1e-4
Text encoder learning rate	1e-5
Weight decay	1e-3
Patience (LR scheduler)	1
Factor (LR scheduler)	0.8
Number of epochs	2
Device	GPU
Image encoder model	ResNet-50 / FastViT-T8
Text encoder model	distilbert-base-uncased
Image embedding dimension	2048 / 768
Text embedding dimension	768
Tokenizer	distilbert-base-uncased
Max text length	64 / 200
Pretrained	True
Trainable	True
Temperature	1.0
Image size	224
Number of projection layers	1
Projection dimension	256
Dropout	0.1

D. Training

The model was trained with a contrastive loss function to match image and text embeddings. Mixed precision training was used in Flickr30k to improve efficiency, and early stopping, activated by validation loss plateaus, avoided overfitting. The training process is described in Algorithm 1.

E. Evaluation Metrics

Model performance was assessed using the following metrics:

- **Recall@K (K=1, 5, 10)**: Measures the proportion of queries retrieving the correct image within the top K results, a standard for retrieval tasks.
- **Mean Average Precision (mAP)**: Provides an overall precision score across queries, reflecting ranking quality.
- **Training/Validation Loss**: Monitors convergence and generalization.
- **Efficiency Metrics**: Inference time and memory usage evaluate computational feasibility.

The inference process, used retrieve top matches for a text query, which is detailed in Algorithm 2.

IV. Experiments and Results

This section describes the experimental setup, results, and analysis of the performance of the CLIP model on the Flickr30k and RSTP-ReID datasets. The experiments were

Algorithm 1 Training the CLIP Model

Input: Dataset $\mathcal{D}$, Hyperparameters (batch size B, epochs E, learning rates, etc.)
Procedure:
Load and Preprocess Data:
  Load dataset $\mathcal{D}$
  Split $\mathcal{D}$ into training set $\mathcal{D}_{\text{train}}$ and validation set $\mathcal{D}_{\text{val}}$
  Initialize tokenizer for text processing
  Create data loaders for $\mathcal{D}_{\text{train}}$ and $\mathcal{D}_{\text{val}}$ with batch size B
Initialize Model and Optimizer:
  Initialize CLIP model with image encoder, text encoder, and projection heads
  Set up optimizer with specified learning rates and weight decay
  Initialize learning rate scheduler
Training Loop:
  for epoch = 1 to E **do**
    Train the model:
      for each batch in $\mathcal{D}_{\text{train}}$ **do**
        Encode images and texts to get embeddings
        Project embeddings to shared space
        Compute similarity matrix
        Calculate contrastive loss
        Backpropagate and update model parameters
      end for
    Evaluate the model:
      Compute validation loss on $\mathcal{D}_{\text{val}}$
    if validation loss improves **then**
      Save the model
    end if
    Update learning rate scheduler
  end for
Output: Trained CLIP model

Algorithm 2 Inference with the CLIP Model

Input: Trained model, Validation dataset $\mathcal{D}_{\text{val}}$, Text query q, Number of top matches N
Procedure:
Load Trained Model:
  Load the best saved model
Get Image Embeddings:
  for each image in $\mathcal{D}_{\text{val}}$ **do**
    Encode image to get embedding
    Project embedding to shared space
  end for
  Collect all image embeddings
Process Text Query:
  Tokenize the text query q
  Encode tokenized text to get embedding
  Project embedding to shared space
Compute Similarity:
  Compute dot product between text embedding and each image embedding
Retrieve Top Matches:
  Select top N images with highest similarity scores
Output: Top N matching images for the query q

performed on Kaggle's computational resources, i.e., dual NVIDIA T4 GPUs, each with 16GB VRAM, for a total of 32GB. PyTorch was used as the deep learning framework, with GPU acceleration to improve training and inference efficiency.

A. Computational Resources

All the experiments were conducted on Kaggle's environment, utilizing dual NVIDIA T4 GPUs with a total of 32GB of VRAM. This configuration facilitated effective training and testing of the models, especially in terms of managing large batch sizes and computationally demanding attention mechanisms. The software stack consisted of PyTorch with GPU support, providing optimal performance for deep learning tasks.

B. Flickr30k Results

The model trained (ResNet-50 image encoder based and Cross Attention) was checked how well it worked against Flickr30k by generating image suggestions from two given search terms. The model in Figure 1 demonstrates its capability to retrieve nine relevant images of dogs on grass upon being instructed to retrieve "one dog sitting on the grass." The query "man riding cycle" in Figure 2 retrieved images concerning cycling, which proves the model's capability to address various situations.

Fig. 1: Results for query: "one dog sitting on the grass"

The performance on the Flickr30k dataset is represented in Table II, which captures major metrics for both ResNet-50 and FastViT-T8 image encoders in self-attention, cross-attention, and sparse attention schemes. The performance metrics include

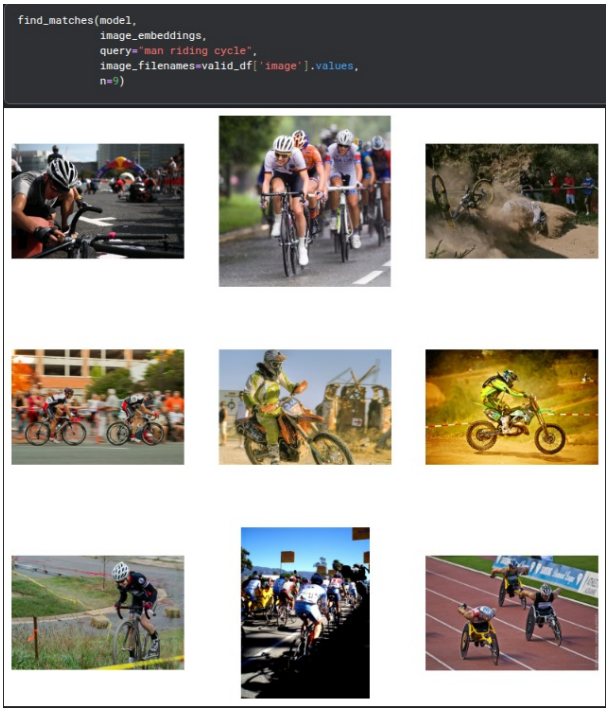


Fig. 2: Results for query: ”man riding cycle”

training loss, validation loss, Recall@K (K=1,5,10), inference time, memory usage, and training time.

To better demonstrate the performance contrast, Figure 3 shows a bar plot contrasting the Validation Recall@K results of the various configurations. The plot demonstrates the better performance of cross-attention in attaining higher recall values, especially for Recall@5 and Recall@10.

C. RSTP-ReID Results

The model (fastViT t8 based image encoder and Cross Attention mechanism) trained was tested its efficacy against RSTP-ReID by generating image suggestions from two given search terms. The model in Figure 4 indicates its capability to locate nine images of dogs on grass after being instructed to locate ”a man or woman wearing a jacket,” which indicates the model’s capability to cope with various scenarios.

Results on the RSTP-ReID dataset are shown in Table III, and this contains training, validation, and test scores for both attention and image encoders. A wide range of metrics for both inference time and memory usage and the model’s performance, including loss, Recall@K, and inference time, are reported by the table.

Figure 5 visualizes the Test Recall@K metrics for the RSTP-ReID dataset, emphasizing the strong performance of cross-attention across both image encoders.

D. Comparative Analysis

The experimental findings present numerous interesting insights:

- **Cross-Attention**: This mode always performs better than self-attention and sparse attention in recall, reaching almost perfect Recall@10 on both datasets. Nevertheless, it comes with higher inference times, especially for ResNet-50 on Flickr30k (0.2957s compared to 0.0387s for self-attention).
- **FastViT-T8**: In contrast to ResNet-50, FastViT-T8 presents lower memory consumption (e.g., 1189.0 MB compared to 2202.1 MB for self-attention on Flickr30k) as well as reduced training time (e.g., 992.59s compared to 2360.06s for self-attention on Flickr30k), and is thus a more resource-light option without considerable compromise in performance.
- **Sparse Attention**: Although providing mid-level recall performance, sparse attention is computationally costly, with high memory consumption (e.g., 5378.7 MB on RSTP-ReID) and long inference durations (e.g., 1.7074s on Flickr30k), which restricts its practical usage.

These results emphasize the accuracy-efficiency trade-offs, where cross-attention performs well in retrieval tasks but is more computationally expensive, and FastViT-T8 offers a balanced approach for real-world application.

V. Discussion

This section discusses the experimental results, explains the reasons behind observed results, discusses challenges faced during the study, and investigates the accuracy-efficiency trade-offs in the case of text-based person search using the CLIP model.

A. Interpretation of Results

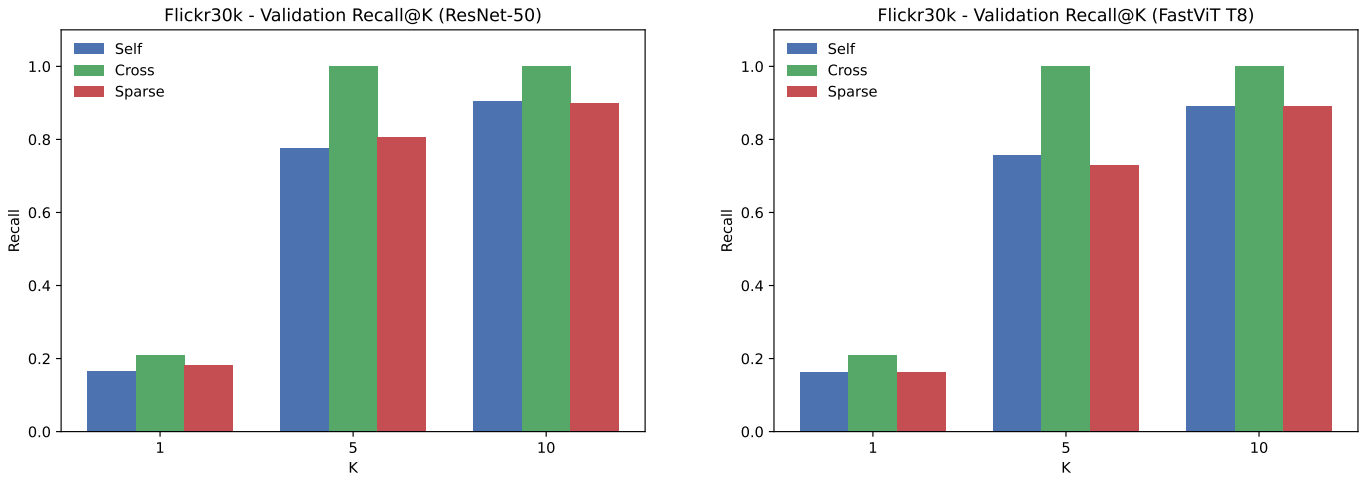
Cross-Attention’s Strength: Cross-attention is strongest at allowing direct interaction between image and text features, strongly improving retrieval accuracy. On the Flickr30k dataset, it has a Recall@10 of 1.0000 with both ResNet-50 and FastViT-T8, and also perfect recall on the RSTP-ReID test set. This is because it has the ability to match multimodal representations strongly. This does not come for free, though: inference time is longer (e.g., 0.2957s on Flickr30k for ResNet-50, 0.1365s on RSTP-ReID for FastViT-T8), and memory consumption is larger (e.g., 3264.0 MB for ResNet-50 on Flickr30k).

FastViT-T8’s Efficiency: The FastViT-T8 encoder is a very efficient alternative to ResNet-50. On Flickr30k, it drops memory consumption significantly (e.g., 1189.0 MB vs. 2202.1 MB for self-attention) and speeds up training times. On RSTP-ReID, it outperforms ResNet-50 slightly in recall (e.g., Test Recall@10: 0.5373 vs. 0.4730 for self-attention) while keeping inference times similar (e.g., 0.1347s vs. 0.1340s). Such efficiency makes FastViT-T8 a very appealing option for real-world deployment where computational resources are tight.

Sparse Attention’s Memory Requirements: Although theoretically formulated to minimize computational cost, sparse attention in practice lags behind due to its high resource requirements. On RSTP-ReID, it requires 5378.7 MB of memory when using ResNet-50, and on Flickr30k, inference

TABLE II: Results for Flickr30k Dataset

Metric	ResNet-50			FastViT T8		
	Self	Cross	Sparse	Self	Cross	Sparse
Training Loss	1.69	0.61	1.18	1.85	0.85	1.62
Validation Loss	2.57	1.66	2.12	2.58	1.64	2.16
Validation Recall@1	0.1659	0.2083	0.1820	0.1629	0.2083	0.1625
Validation Recall@5	0.7753	0.9998	0.8075	0.7581	0.9998	0.7305
Validation Recall@10	0.9036	1.0000	0.8994	0.8925	1.0000	0.8906
Inference Time (s)	0.0387	0.2957	1.7074	0.0452	0.1091	1.7727
Memory Usage (MB)	2202.1	3264.0	2746.6	1189.0	1198.5	2428.0
Training Time (s)	2360.06	2221.38	8144.84	992.59	959.45	8277.13



(a) ResNet-50

(b) FastViT T8

Fig. 3: Validation Recall@K on Flickr30k dataset using different attention mechanisms (Self, Cross, Sparse) for ResNet-50 and FastViT T8 backbones.

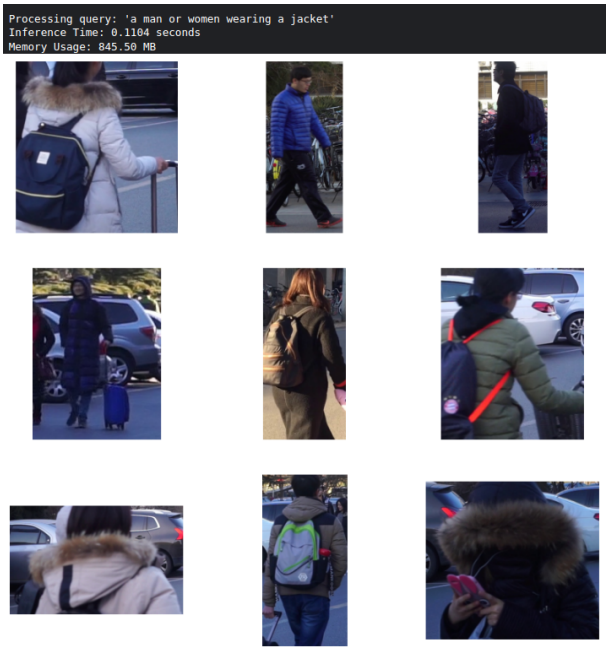


Fig. 4: Results for query: "a man or woman wearing a jacket"

times become 1.7074s for ResNet-50 and 1.7727s for FastViT-T8. These limitations counterbalance its potential advantages, making it less suitable for big applications.

B. Challenges

Cross-Attention Overfitting: Cross-attention's high complexity enhances overfitting risks, especially due to its powerful feature interactions. Overfitting was countered with an early stopping scheme using validation loss, thereby providing a generalized performance to unseen inputs as indicated in the low obtained test losses.

Memory Constraints with Sparse Attention: Sparse attention's intensive memory usage, particularly on RSTP-ReID (e.g., 5378.7 MB with ResNet-50), constrains its scalability. This problem reflects a primary challenge in applying sparse attention to larger datasets or denser models, calling for further optimization.

C. Trade-offs

The findings unveil an inherent trade-off between accuracy and efficiency:

- **Accuracy**: Cross-attention is superior in retrieval tasks, providing unparalleled recall but at the cost of heavy computational requirements.

TABLE III: Results for RSTP ReID Dataset

Metric	ResNet-50			FastViT T8		
	Self	Cross	Sparse	Self	Cross	Sparse
Training Metrics						
Training Loss	3.02	1.26	1.87	2.89	1.18	1.99
Training Time per Epoch (s)	617.81	609.77	4675.35	525.42	527.05	4651.75
Validation Metrics						
Validation Loss	3.23	0.81	2.31	3.23	0.80	2.19
Validation Recall@1	0.0578	0.4850	0.1013	0.0575	0.4743	0.0870
Validation Recall@5	0.2475	0.9978	0.4497	0.2625	0.9960	0.4065
Validation Recall@10	0.4335	1.0000	0.7708	0.4625	0.9995	0.7265
Validation Inference Time (s)	0.1212	0.1250	0.9036	0.1218	0.1218	0.9141
Test Metrics						
Test Loss	3.05	0.82	2.12	2.92	0.78	2.08
Test Recall@1	0.0658	0.4795	0.1120	0.0705	0.4765	0.0910
Test Recall@5	0.2803	0.9973	0.4840	0.3158	0.9980	0.4225
Test Recall@10	0.4730	1.0000	0.8042	0.5373	1.0000	0.7472
Test Inference Time (s)	0.1340	0.1239	0.9016	0.1347	0.1365	0.9210
Memory Usage (MB)	1470.8	1495.1	5378.7	1451.1	1470.9	2398.0

- **Efficiency**: Self-attention and FastViT-T8 allow for quicker inference (e.g., 0.0387s for ResNet-50 on Flickr30k) and reduced memory usage, preferable for resource-limited settings.

These trade-offs imply that model selection should be based on application-specific priorities—accuracy or efficiency—in order to provide possibilities to adapt solutions to various real-world requirements.

VI. Conclusion

This work proves that combining cross-attention with the FastViT-T8 image encoder offers the best compromise between accuracy and efficiency for person search based on text. Cross-attention provides outstanding retrieval performance (e.g., Recall@10 of 1.0000 on both Flickr30k and RSTP-ReID), while FastViT-T8 improves deployability by limiting memory consumption and training times. On the other hand, the high computational expenses of sparse attention restrict its real-world applicability even though theoretically it is attractive.

Looking forward, future research may explore hybrid attention mechanisms that combine cross-attention's precision with sparse attention's promise of efficiency. Testing the model on other datasets, like CUHK-PEDES, would further establish its robustness and extend its use, opening the door to further multimodal person re-identification advancements.

References

[1] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning Transferable Visual Models From Natural Language Supervision," in *Proceedings of the 38th International Conference on Machine Learning*, vol. 139, pp. 8748–8763, PMLR, 2021.

[2] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is All You Need," in *Advances in Neural Information Processing Systems*, vol. 30, 2017.

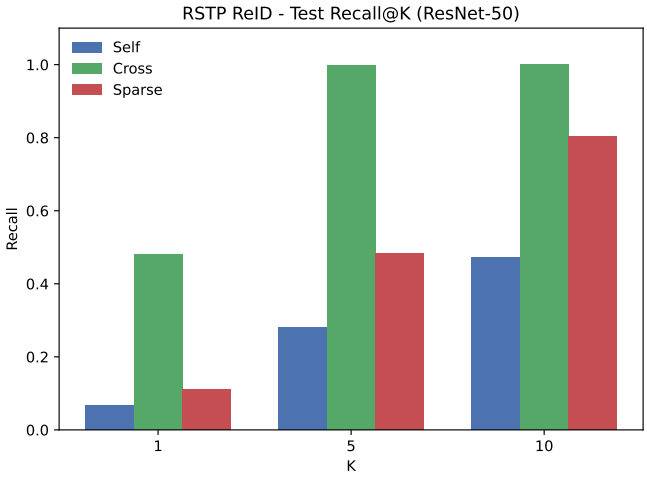
[3] P. Young, A. Lai, M. Hodosh, and J. Hockenmaier, "From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions," *Transactions of the Association for Computational Linguistics*, vol. 2, pp. 67–78, 2014.

[4] [Author Names], "RSTP-ReID: A Real-World Surveillance Dataset for Person Re-Identification," *[Journal/Conference Name]*, [Year].

[5] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

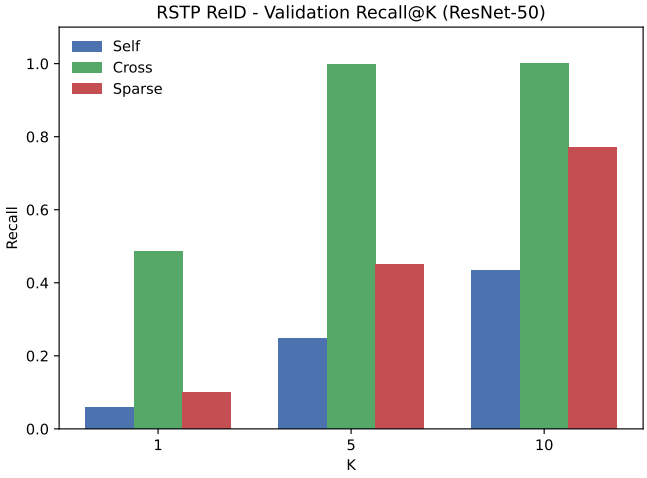
[6] A. Vasan, S. Yun, and A. Arora, "FastViT: A Fast Hybrid Vision Transformer using Structural Reparameterization," *arXiv preprint arXiv:2303.14189*, 2023.

[7] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter," *arXiv preprint arXiv:1910.01108*, 2019.

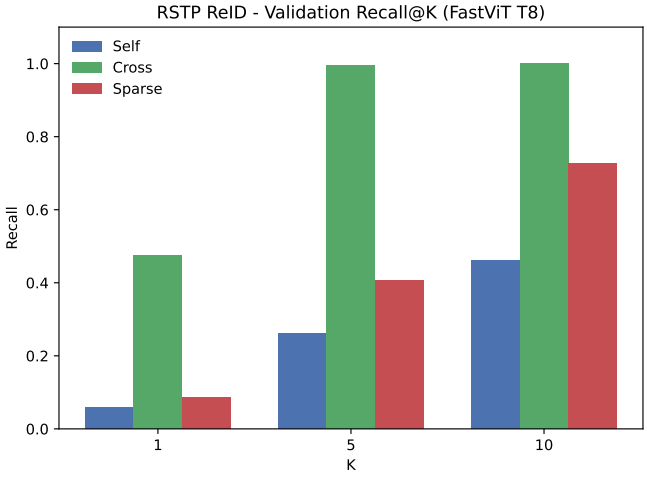


(a) Test Recall@K - ResNet-50

(b) Test Recall@K - FastViT T8



(c) Validation Recall@K - ResNet-50



(d) Validation Recall@K - FastViT T8

Fig. 5: Recall@K on RSTP ReID dataset across Test and Validation splits using Self, Cross, and Sparse attention mechanisms for both ResNet-50 and FastViT T8 backbones.