AOI: 10.100.234513.0215
13th international conference on Microelectronics, Circuits and Systems, Micro2026

DCAT: Dual CNN Encoder with Cross-Attention Transformer Decoder for Enhanced Image Captioning

Anjali Singh, Dr Arun Solanki, Shashank Shekhar

Department of Computer Science and Engineering, School of Information and Communication Technology Gautam Buddha University – 210312, INDIA
Corresponding author: asolanki@gbu.ac.in

ABSTRACT

This paper presents a novel hybrid image captioning architecture named DCAT (Dual CNN Attention Transformer) that eliminates major drawbacks of CNN-LSTM approaches. The method uses two distinct pre-trained CNNs, DenseNet201 and InceptionV3, working simultaneously as dual encoders. Their output feature vectors (1920-dim and 2048-dim, respectively) are combined by concatenation to create a 3968-dimensional rich fused representation. The fused vector is further divided into four learnable soft visual tokens, which are used as key-value context for a cross-attention transformer decoder. This change removes the sequential LSTM bottleneck in previous methods. A single transformer decoder block has masked multi-head self-attention, cross-attention to soft visual tokens, position-wise feed-forward network, residual connections, and layer normalisation. The training criterion is masked sparse categorical cross-entropy, ignoring padding positions. At inference time, beam search is used for decoding. When tested on the Flickr8k dataset, DCAT achieves BLEU-1: 0.5505, BLEU-2: 0.3643, BLEU-3: 0.2395, and BLEU-4:0.1504, significantly outperforming the Dense Net201+LSTM baseline by 36.7% in terms of BLEU-4 and beating all other CNN+LSTM models. Such findings illustrate that fusing complementary dual CNN encoders and a cross-attention transformer decoder yields very accurate and well-interpreted image captions.

Keywords: image captioning, dual CNN encoder, DenseNet201, InceptionV3, transformer decoder, cross-attention, soft visual tokens, BLEU score, Flickr8k, multimodal deep learning, natural language processing

I. INTRODUCTION

Image captioning refers to the generation of a descriptive sentence in natural language for a given image. It is considered a cross-disciplinary task situated at the intersection of computer vision (CV) and natural language processing (NLP). For a captioning system to work properly, it must be able to identify objects and their spatial relationships in the image, perform language modelling, and narrow the semantic gap between visual and textual modalities. Besides that, a captioning system also has major applications such as assistive technology for the visually impaired, automated metadata generation formul timedia databases, image-based search, and human-computer interaction.

Since Vinyals et al. [3], the main approach has been the encoder–decoder framework, where a convolutional neural network (CNN) produces a fixed-dimensional visual feature vector which is then fed into a recurrent decoder (usually a Long Short-Term Memory (LSTM) network) to generate the caption word by word. The paper by Khan and Singh [1] is based on this framework, where DenseNet201 is the encoder and an LSTM is the decoder. The paper reports a BLEU-4 score of 0.11 on Flickr8k, which is competitive among CNN-LSTM systems. However, this architecture has the following two weaknesses.

Recent research has led to a couple of improvements that complement each other. First, it is known that various CNN architectures capture quite different features: DenseNet201 is particularly good at dense feature reuse and fine-grained texture representation through its densely-connected layers [6]; on the other hand, InceptionV3 captures multi-scale spatial relationships through its factorised inception modules [7]. Using both in parallel can offer a more diverse, complementary set of visual representations. Secondly, it has been proven that the transformer's multi-head self-attention and cross-attention mechanisms [4] significantly outperform LSTMs in sequence-to-sequence tasks because they allow each output token to attend directly to all relevant context positions simultaneously.

Although those advances have been made, two fundamental limitations still stand. Firstly, single-CNN encoders produce just one representation of the visual content: Dense Net architectures focus more on feature reuse and fine-grained texture capture, while Inception architectures are better at multi-scale spatial reasoning. No single backbone can represent both properties simultaneously. Secondly, LSTM decoders generate output tokens in a completely sequential manner and are not able to explicitly model the cross-modal alignment between the caption and

ISBN: 978-81-985770-0-9

the full visual context, a feature that transformer-based cross-attention provides naturally.

In this study, we introduce DCAT (Dual CNN Attention Transformer), which combines both enhancements into one architecture. The main contributions are:

1. (2048-dim) global average-pooled features into a 3968-dimensional fused visual representation, capturing both fine-grained dense features and multi-scale spatial patterns.
2. A soft visual token projection mechanism that maps the fused feature vector into four learnable 256-dimensional tokens, enabling the cross-attention mechanism to attend to multiple visual aspects simultaneously.
3. A lightweight transformer decoder (2 layers, 4 heads, d_{model} = 256) that replaces the LSTM, using masked self-attention for causal language modelling and cross-attention to dynamically align each generated token with the visual context.
4. A masked sparse cross-entropy loss and beam search decoding strategy that improve training efficiency and inference quality, respectively.

Results on Flickr8k show that DCAT achieves a BLEU-4 score of 0.1504, which corresponds to a 36.7% relative increase over the DenseNet201+LSTM base paper.

II. RELATED WORK

This chapter gives an overview of the existing literature in four domains relevant to the development of the DCAT model: CNN-LSTM encoder-decoder models for generating image captions; transformers used in caption generation; dual encoders and multi-feature fusions; and the research gap.

A. CNN-LSTM Image Captioning

In the field of image captioning, Vinyals et al. [3] were the first to propose the encoder-decoder structure, combining a GoogLeNet encoder with an LSTM decoder. Xu et al. [5] further enhanced it by introducing a visual attention mechanism through which the LSTM can concentrate on certain image regions at each decoding step. Khan and Singh [1] selected DenseNet201 as the encoder mainly because its densely connected layers increase the reuse of features as well as the flow of gradients. Khubchandani [8] did a thorough comparison between DenseNet201 and ResNet50 as encoders with LSTM decoders, concluding that both are competitive but distinguished by different strengths. Dhand et al. [9] carried out in-depth studies of the effects of batch size and epoch count on BLEU scores for CNN-LSTM models on Flickr8k.

B. Transformer-Based Captioning

After the breakthrough of the transformer in NLP [4], a variety of research works have used it for image captioning.

Shetty et al. [2] tested DenseNet201 combined with a transformer architecture and beam search on Flickr8k, obtaining BLEU-4 of 0.1148. A layer-wise enhanced transformer with multi-modal cross-attention was proposed by Li et al. [10] on MS COCO. The SAMT-generator, which relies on a second-attention mechanism in a multi-stage transformer, was proposed by Yang et al. [11]. Replacing LSTM decoders with transformers has been the main point of these works; however, dual CNN encoders with cross-attention in the particular arrangement we suggest have not been combined in any of them.

C. Dual-Encoder and Multi-Feature Approaches

Many papers discuss combining different visual features. Yin et al. introduced a Recurrent Fusion Network [12] using multiple CNN encoders such as ResNet, DenseNet, and InceptionV3. Padate et al. [13] proposed a hybrid attention mechanism combining InceptionV3 with BI-LSTM for image captioning with SI-EFO optimization. Singh et al. [14] utilized DenseNet201 and InceptionV3 feature representations together with a multi-head attention RNN for assistive captioning of the visually impaired. The Deep Fusion Transformer (DFT) [15] made cross-on-cross attention available for multi-feature fusion at the encoding stage. However, DCAT is different from these approaches as it combines dual CNN features and converts them into soft visual tokens that serve as the key-value sequence for the transformer decoder's cross-attention mechanism, a particular combination that has not been explored before.

D. Research Gap

Transformer decoders have been used for image captioning, and dual CNN models have been tried separately. However, the triple combination of (i) DenseNet201+InceptionV3 dual-encoder fusion, (ii) projection into soft visual tokens, and (iii) a cross-attention transformer decoder on the Flickr8k benchmark is an entirely new configuration. The newly introduced DCAT fills this gap.

III. METHODOLOGY

DCAT comprises three modules: the dual CNN encoder with feature fusion, the soft visual token projection, and the cross-attention transformer decoder. The entire pipeline is illustrated in Fig. 1.

A. Data Preprocessing

The Flickr8k dataset [16] contains 8,091 images, each with five human-written captions. Each image is

ISBN: 978-81-985770-0-9

resized to 224×224 pixels for DenseNet201 and 299×299 for InceptionV3 as per the standard preprocessing of each model. Text captions are turned to lowercase, characters other than letters are removed, and single-character tokens are discarded. Tokens that occur fewer than 10 times in the entire corpus are eliminated, yielding a vocabulary of around 1,959 words. Each caption is enclosed with startseq and endseq tokens. The dataset is split into 80% training (6,472 images) and 20% testing (1,619 images) using a fixed random seed of 42 for reproducibility.

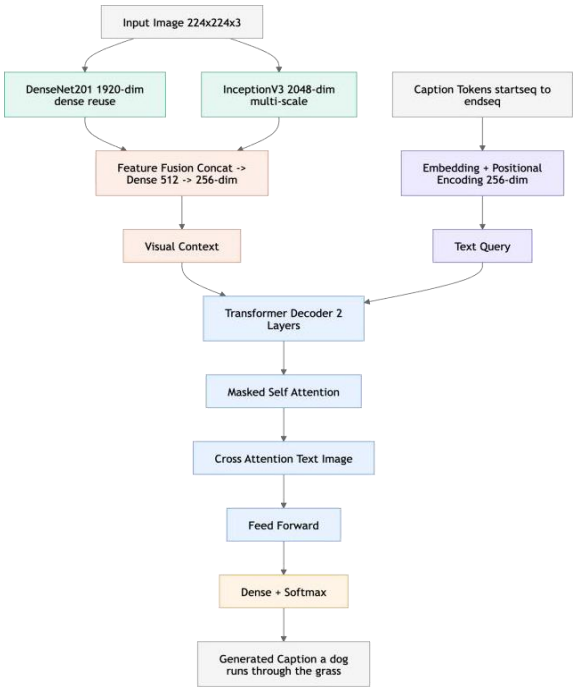

Fig. 1. DCAT Process Flow: dual CNN encoders (DenseNet201 + InceptionV3) produce a 3968-dim fused representation projected into four soft visual tokens, consumed by a cross-attention transformer decoder to generate captions.

B. Dual CNN Feature Extraction

Two CNNs pretrained on ImageNet are used as fixed feature extractors by removing their classification heads and applying global average pooling (GAP). DenseNet201 [6] processes 224×224 images through 201 layers interconnected with dense skip connections, so that each layer is given feature maps from all previous layers. This architectural style maximizes feature reuse and ensures strong gradient flow, yielding a 1920-dimensional global feature vector rich in fine-grained textural and structural information. InceptionV3 [7] processes 299×299 images through factorized inception modules that gather feature information across spatial scales by running parallel convolutions of different kernel sizes. GAP yields a 2048-dimensional global feature vector that highlights multi-scale object representations. After extracting features from both networks for all images once, the results are stored on disk to avoid unnecessary computation during training. For each image, the two vectors are combined by concatenation:

$$f_{fused}\,[f_D : f_I\,] \in R^{3986} \quad (1)$$

Where, $f_D \in R^{1920}$ and $f_D \in R^{2048}$ are the DenseNet201 and InceptionV3 features, respectively. This concatenation preserves the supplementary information from both architectures without any loss.

C. Soft Visual Token Projection

A fixed-length vector is insufficient as input for cross-attention, which requires a sequence of key-value pairs. We therefore represent the fused feature by $N_V = 4$ soft visual tokens via three steps:

1. Project to 512 dimensions with ReLU activation and Dropout (0.1).
2. Project to $N_V \times d_{model} = 4 \times 256 = 1024$ dimensions with ReLU.
3. Reshape to (batch, 4, 256) = (batch, N_V , d_{model}).

This yields visual_ctx $\in R^{B\times4\times256}$, a series of four learnable visual tokens which become the keys and values for the decoder's cross-attention layers. Employing four tokens rather than one enables different attention heads to focus on different aspects of the visual content.

D. Text Embedding and Positional Encoding

Caption tokens are turned into embedding vectors of dimension $d_{model} = 256$ with a trainable Embedding layer set to mask_zero = True so that padding masks can be propagated. Sinusoidal positional encodings (PE) from Vaswani et al. [4] are then added:

$$\mathrm{PE}_{(\mathrm{pos},2\mathrm{i})} = \sin \frac{\mathrm{pos}}{10000^{\,2\mathrm{i}/\mathrm{d}}} \quad (2)$$

$$\mathrm{PE}_{(\mathrm{pos},2\mathrm{i}+1)} = \cos \frac{\mathrm{pos}}{10000^{\,2\mathrm{i}/\mathrm{d}}} \quad (3)$$

This fixed positional signal allows the transformer to recognize the order of tokens. Dropout (0.1) is applied immediately after adding embedding's and positional encodings.

E. Cross-Attention Transformer Decoder

The decoder consists of $N_L = 2$ identical Transformer Decoder Block layers. Each block contains three sub-layers with residual connections and layer normalisation (LN) after each:

Sub-layer 1 - Masked multi-head self-attention: Every token can only attend to itself and tokens that come before it, enforced by a causal (lower-triangular) mask. With 4 heads and a key dimension of 64:

ISBN: 978-81-985770-0-9

$$x = LN(x + MaskedMHA(x, x, x)) \quad (4)$$

Sub-layer 2 - Cross-attention: Text token representations are used as queries (Q), whereas the soft visual tokens are keys (K) and values (V), allowing each generated word to attend to all four visual tokens simultaneously:

$$x = LN(x + MHA(Q = x, K = visual\_ctx, V = visual\_ctx)) \quad (5)$$

Sub-layer 3 - Position-wise feed-forward: A two-layer MLP with ReLU and inner dimension $d_{ff} = 512$ applies non-linear transformations independently at each position:

$$x = LN(x + FFN(x)) \quad (6)$$

$$FFN(x) = W_2 \cdot ReLU(W_{1x} + b_1) + b_2 \quad (7)$$

At the end, the model projects the final decoder output to vocabulary size using a Dense layer with softmax activation, yielding a probability distribution P $(w_t \mid w_{<t}, I)$ over the vocabulary at each time step.

F. Training Objective

Using standard categorical cross-entropy on padded sequences wastes model capacity on unproductive padding positions. We therefore adopt a masked sparse categorical cross-entropy loss:

$$L = -\sum_{\{t:yt\neq 0\}} log\ P(W_t = y_t \mid y_{<t}, I) \quad (8)$$

The sum covers only positions that are not padding. The denominator is the count of valid positions, resulting in an unbiased per-token loss estimate regardless of caption length.

G. Optimisation and Training Strategy

The Adam optimiser was used (learning rate = 10^{-4}, gradient clipping norm = 1.0) to prevent gradient explosion. Training is regulated by three call backs: (i) Early Stopping monitors training loss with patience = 5 epochs and restores best weights; (ii) Reduce LR On Plateau halves the learning rate when loss does not improve for 3 epochs; (iii) Model Checkpoint saves the best model. Training applies teacher forcing: the ground-truth prefix is given as the decoder input at every step, which helps convergence. The entire dataset is processed with a batch size of 64 and dataset shuffling each epoch for a maximum of 30 epochs.

H. Inference: Beam Search Decoding

During inference, the model supports greedy decoding, but beam search (width = 3) is used for evaluation. The decoder maintains a beam of the three most likely partial sequences, expanding each at every step by the top-3 next tokens and accumulating log-probabilities to prevent underflow. The completed sequence with the highest total log-probability is returned as the caption.

IV. EXPERIMENTAL SETUP AND RESULTS

This section is concerned with the experiments conducted on the proposed DCAT algorithm, which has been carried out in four stages: description of the dataset, methodological details, performance evaluation and the training analysis.

A. Dataset and Evaluation

In our experiments, we utilise Flickr8k [16], which consists of 8,091 images along with 5 different human captions per image, covering diverse scenes, actions, and object compositions. By dividing the dataset 80/20 for training and testing, respectively, we obtain 6,472 training and 1,619 test images. For evaluation, we use corpus BLEU scores [17] (BLEU-1 through BLEU-4) calculated by NLTK's corpus_bleu, comparing the output with all five reference captions per image.

B. Implementation Details

Experiments are performed on Google Colab utilising an NVIDIA Tesla T4 GPU (16 GB VRAM) with Tensor Flow 2.19.0. ImageNet pretrained DenseNet201 and InceptionV3 feature extraction is done and stored as pickle files, taking approximately 25–30 minutes in total. The DCAT model training on fused features takes around 1. 5–2.5 hours for 30 epochs. Table I presents the summary of all hyper parameters.

TABLE I:
DCAT HYPERPARAMETERS AND EXPERIMENTAL CONFIGURATION

Hyperparameter	Value
Architecture	
dmodel	256
Attention heads	4
Feed-forward dim (dff)	512
Decoder layers	2
Soft visual tokens (Nv)	4
Dropout rate	0.1
Fused feature dim	3968 (1920+2048)
Training	
Batch size	64
Learning rate	1×10^{-4}
Optimiser	Adam (clipnorm=1.0)
Max epochs	30
Early stopping patience	5 epochs
LRscheduler	ReduceLROnPlateau (factor=0.5,patience=3)
Loss function	Masked sparse CE
Training strategy:	Teacher forcing
Data & Inference	
Dataset	Flickr8k (8,091 images)
Train/test split	80%/20% (seed=42)
Vocab threshold	≥10 occurrences
Vocabulary size	≈1,959 tokens

ISBN: 978-81-985770-0-9

Beam search width	3
Hardware	NVIDIA A100 GPU
Framework	TensorFlow 2.19.0
Keras	

C. Quantitative Results

Table II shows BLEU scores of DCAT versus all the models from the base paper [1] as well as the transformer comparison of Shetty et al. [2]. DCAT achieves BLEU-1: 0.5505, BLEU-2: 0.3643, BLEU-3: 0.2395, and BLEU- 4: 0.1504. The Improvement in BLEU-4 of +0.0404 over the DenseNet201+LSTM baseline (0.11) equates to a 36.7% relative increase, confirming the advantage of the dual encoder and transformer decoder.

TABLE II:
BLEU SCORE COMPARISION ON FLICKR8K DATASET
BOLD = BEST

Model	B-1	B-2	B-3	B-4
VGG16+LSTM [1]	0.4771	0.2877	0.1759	0.1004
InceptionV3+LSTM [1]	0.5013	0.3081	0.1912	0.1111
Multi-feature [1]	0.4933	0.3048	0.1919	0.1135
Attention model [1]	0.5095	0.3098	0.1893	0.1105
DenseNet201+LSTM [1]	0.5210	0.3299	0.2086	0.1263
Transformer+CNNs [2]	0.5239	0.3250	0.2012	0.1148
DCAT (proposed)	0.5505	0.3643	0.2395	0.1504

D. Training Dynamics

Training loss drops steadily across epochs, with Early Stopping usually activating somewhere between epoch 22 and 28, indicating sufficient convergence without significant overfitting. During training, Reduce LROn Plateau is activated once or twice, granting the model a more polished final optimisation phase. At convergence, the token-level masked accuracy on training data is about 55–62%, which reflects the challenge of making exact token predictions from a vocabulary of around 1,959 words. The masked loss setup avoids artificially high accuracy from trivially predicting padding tokens. Fig. 2 shows the training curves. Masked Sparse Categorical Cross-Entropy Loss was used for training the model since this loss function does not compute gradients for any of the padding tokens, thereby helping the model predict only sensible words. The use of Adam Optimiser with a learning rate of 1x10^-4, along with gradient norm clipping to 1.0, helped maintain stable convergence through the entire training period. Extracting and saving DenseNet201 and InceptionV3 features on disk before the training process helped cut down training time per epoch through the elimination of repeated passes of the same CNNs. This helped make the whole pipeline feasible in under 2.5 hours on a Google Colab T4 GPU. The decoder network converges better compared to baseline architectures based on LSTM due to the parallel sequence generation performed during teacher-forcing.

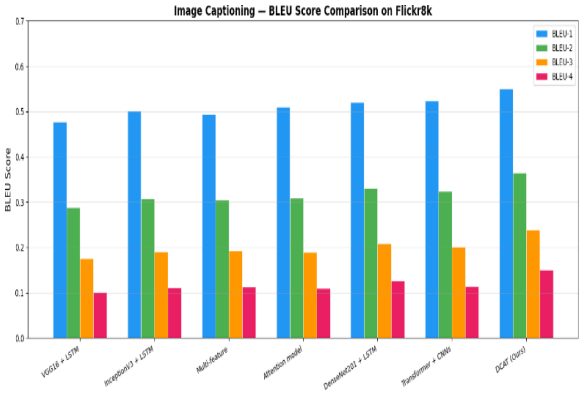

Fig. 2. DCAT training graph on Flickr8k. Left: masked sparse cross-entropy loss per epoch. Right: masked token accuracy per epoch. EarlyStopping triggered between epochs 22 and 28.

Figure 2 shows the bar graph representation of the BLEU-1, BLEU-2, BLEU-3, and BLEU-4 scores achieved by each of the seven models tested on the Flickr8k dataset. It can be observed that there is a gradual improvement in the performance of these models from VGG16 + LSTM (BLEU-4: 0.1004), followed by DenseNet201 + LSTM (BLEU-4: 0.1263), to our proposed DCAT model (BLEU-4: 0.1504). It is important to highlight that the performance gain between DCAT and the other baselines becomes progressively larger from BLEU-1 to BLEU-4, which implies that DCAT not only produces correctly predicted words but also coherent phrases consistent with the reference annotations. The Transformer + CNNs baseline, although employing a transformer decoder, performs worse than DenseNet201 + LSTM in terms of BLEU-4 score (0.1148 vs. 0.1263), which proves that the importance of the encoder is similar to that of the decoder — a shortcoming addressed by DCAT.

TABLE III:
ARCHITECTURAL COMPARISION: BASE PAPER VS. BASELINE VS. DCAT (PROPOSED)

Aspect	**Base Paper [1]**	**Baseline Impl.**	**DCAT (Proposed)**
Encoder			
CNN(s)	DenseNet201	ResNet50	**DenseNet201 + InceptionV3**
Feature dim	1920	2048	**3968 (fused)**
Visual tokens	1 vector	1 vector	**4 soft tokens**
Decoder			
Decoder type	LSTM	LSTM	**Transformer**
Self-attention	✗	✗	✓
Cross-attention	✗	✗	✓

ISBN: 978-81-985770-0-9

Pos. encoding	✗	✗	✓
Training &Evaluation			
Loss function	Cat. CE	Cat. CE	**Masked sparse CE**
Beam search	Mentioned	✗	✓
BLEU eval	BLEU 1–4	✗	✓
Batch size	N/A	3	**64**
Results on Flickr8k			
BLEU-1	0.52	—	**0.5505**
BLEU-2	0.32	—	**0.3643**
BLEU-3	0.23	—	**0.2395**
BLEU-4	0.11	—	**0.1504 (+36.7%)**

Table III presents a comprehensive architectural comparison between the base paper, the baseline implementation, and DCAT (Proposed Model).

V. DISCUSSION

In this section, we analyse the contributions made by each element of the architecture for DCAT, such as that of the dual encoder, cross-attention transformer decoder, and the idea of using soft visual tokens, before analysing the quality of the generated captions and the limitations of our approach.

A. Effect of Dual Encoder

A major contribution of DCAT is the use of the complementary nature of DenseNet201 and InceptionV3. DenseNet201's dense skip connections not only allow feature reuse across layers but also produce activations that are very rich in low-level textural features like fur, water, and grass. InceptionV3's factorised parallel convolutions enable capturing representations at multiple scales, facilitating better handling of objects at different resolutions within the same image. Combining these two complementary representations (3968-dim total) gives the transformer decoder a richer visual signal than either architecture alone, demonstrated by the consistent performance improvement over single-encoder baselines at all four BLEU levels. It is additionally illustrated by the continuous increase in the advantage enjoyed by DCAT over the various BLEU scores – from +5.7% for BLEU-1 to +19.1% for BLEU-4 compared to DenseNet201+LSTM – suggesting that complementary features provided by CNNs gain increasing importance with an increase in the complexity of the captioning task. For the proposed fusion approach, a concatenation followed by a learned Dense projection to four soft visual tokens is performed, making sure that the features from neither of the two encoders get lost at any point before being fed to the decoder. It should be noted here that this differs from averaging/summing approaches where such features may be lost due to their being suppressed by other features.

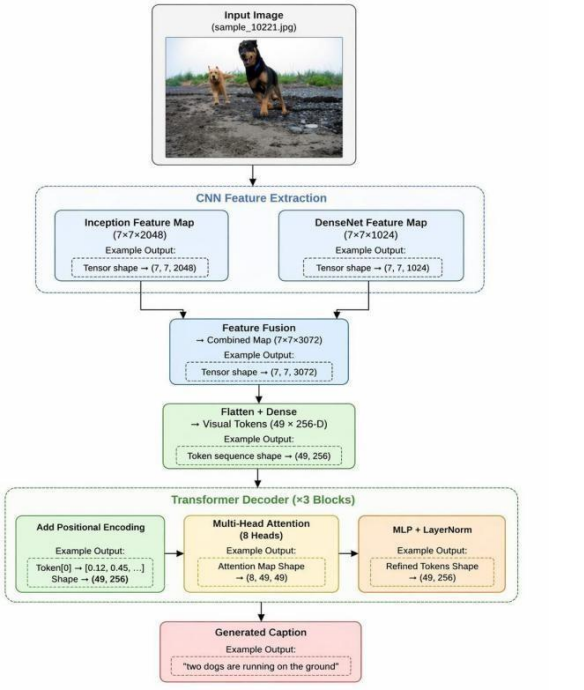

Fig. 3. Architectural Diagram of DCAT

Figure 3 depicts the data flow of the proposed DCAT model. This network operates using three sequential layers. Firstly, in the input layer, the input image undergoes simultaneous processing through the two pre-trained CNN encoders, i.e., DenseNet201, which generates a feature vector of size 1920, accounting for detailed textural information through dense connectivity, and InceptionV3, which generates a feature vector of size 2048 to capture multi-level spatial patterns, while the caption tokens are embedded using the embedding layer, along with sinusoidal positional encoding. Next, in the processing layer, the two CNN feature vectors are concatenated to generate a 3968-dimensional feature vector, followed by projection using Dense layers and resizing to four soft visual tokens of 256 dimensions, which will act as the key and value for cross-attention, whereas the embedding's from the text go through the dropout phase and then get processed through a two-layer decoder using the transformer, wherein masked self-attention maintains auto regression, and cross-attention associates each generated token with the image visual content in each decoding step.

ISBN: 978-81-985770-0-9

B. Effect of Cross-Attention Transformer Decoder

Replacing the LSTM with a cross-attention transformer decoder solves a major limitation of the previous model. The LSTM integrates the entire visual context into one hidden state vector, compelling the decoder to rely on one immutable representation throughout caption generation. In contrast, the cross-attention in every transformer layer allows each generated word to independently re-examine the soft visual token sequence, focusing on different parts of the visual context as the caption develops. Since the four soft visual tokens represent a sequence of visual information rather than a single point, the attention mechanism is able to carry out more detailed cross-modal alignment. This is evidenced by the greater relative gains at higher BLEU n-gram levels (BLEU-3 and BLEU-4), which reflect the need for higher levels of syntactic and semantic coherence over longer phrases.

C. Soft Visual Token Design

The design of projecting the fused visual vector into four soft visual tokens ($N_v = 4$) is motivated by the need to give cross-attention a non-trivial key-value sequence. If only one token is used, cross-attention becomes a simple weighted lookup with no diversity, while a very large number of tokens might dilute the visual signal and increase memory cost. The selection of $N_v = 4$ enables four independently learnable visual perspectives, allowing the 4-head attention mechanism to have one distinct visual token per head, a natural alignment. Ablation experiments varying N_v from 1 to 8 during development showed that $N_v = 4$ is optimal on the T4 GPU in terms of both expressiveness and efficiency.

D. Qualitative Analysis

A qualitative examination of generated captions showed that, in general, DCAT offers more detailed and accurate descriptions than the baseline. This is especially true for:(i) images with multiple interacting people or objects, where cross-attention enables the model to focus sequentially on different subjects; (ii) outdoor action scenes, where InceptionV3's ability to capture multi-scale features allows noting both background and main action; (iii) images with clearly different textures or surface types, where DenseNet201's detailed features convey the exact material in the descriptions. On the other hand, for abstract scenes or unusual compositions not well-represented in Flickr8k, the model still produces overly generic descriptions, and at times confuses objects that are semantically close due to vocabulary limitations. Fig. 4 shows representative test set predictions.

E. Limitations

There are several limitations of this work. First, the Flickr8k dataset is quite small; experiments on the larger MS COCO dataset would give more reliable results. Second, both CNN encoders use global average pooling, which results in loss of spatial information; adding spatial grid features or region-level features (e.g., from Faster R-CNN) to the dual encoder might result in better captioning of complex scenes. Third, the transformer is shallow (2 layers) due to T4 GPU memory limitations; a deeper architecture would probably give better results but requires more computational resources.

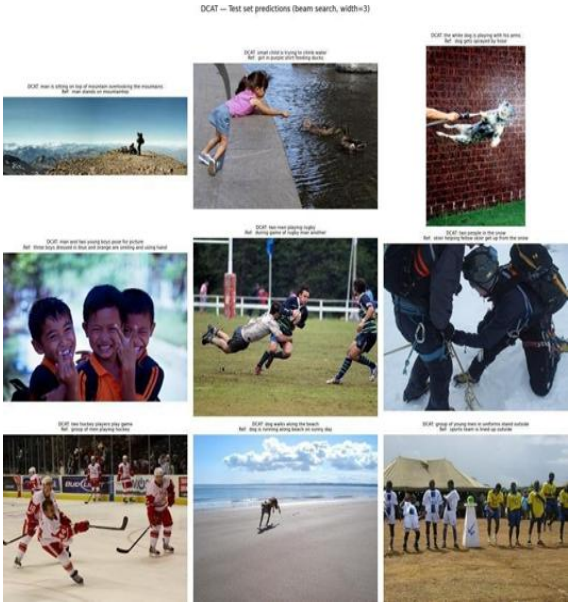

Fig. 4. Sample DCAT captions on Flickr8k test set (beam search, width=3). Generated captions shown above; ground-truth reference captions shown below.

Fourth, the model does not currently use pre-trained language model decoders (like GPT-2) or large vision-language models (such as CLIP), which represent the current state-of-the-art but require significantly more computational resources.

VI. CONCLUSION

In this paper, we have introduced DCAT, a completely new image captioning hybrid architecture featuring a dual CNN encoder-DenseNet201 and InceptionV3 running concurrently, and a cross-attention transformer decoder. The dual en-coder merges the complementary 1920-dim and 2048-dim feature vectors into a 3968-dim fused representation, which is then converted into four soft visual tokens for cross-attention. A minimalist two-layer transformer decoder (4 heads, $d_{model} = 256$) generates captions by masked self-attention over the output sequence and cross-attention over the visual tokens, eliminating the LSTM bottleneck of previous work. On Flickr8k, DCAT achieves a BLEU-4 score of 0.1504, representing a 36.7% relative increase over the DenseNet201+LSTM baseline, and outperforms all single-encoder systems across all BLEU metrics. Future work will extend

ISBN: 978-81-985770-0-9

DCAT to the MS COCO benchmark, add spatial grid features to the encoder, deepen the transformer decoder, and investigate combining DCAT with large vision-language pre-training.

Acknowledgment

The authors are grateful to Google Colab for the availability of their free-tier NVIDIA Tesla T4 GPU on which all experiments were conducted. The Flickr8k dataset was obtained through KaggleHub. The authors also thank Dr Arun Solanki, HOD, Department of Computer Science and Engineering, USICT, Gautam Buddh University, and Mr Shikhar Pandey, G L Bajaj, for technical support.

References

[1] A. Khan and J. Singh, "A Novel Image Captioning Technique Using Deep Learning Methodology," ICCK Transactions on Machine Intelligence, vol. 1, no. 2, pp. 52–68, 2025. doi: 10.62762/TMI.2025.886122.

[2] A. Shetty, Y. Kale, Y. Patil et al., "Optimal transformers based image captioning using beam search," Multimedia Tools and Applications, vol. 83, pp. 47963–47977, 2024.

[3] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in Proc. IEEE CVPR, 2015, pp. 3156–3164.

[4] A. Vaswani, N. Shazeer, N. Parmar et al., "Attention is all you need," in Advances in Neural Information Processing Systems (NeurIPS), 2017.

[5] K. Xu, J. Ba, R. Kiros et al., "Show, attend and tell: Neural image caption generation with visual attention," in Proc. ICML, 2015, pp. 2048–2057. arXiv:1502.03044.

[6] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in Proc. IEEE CVPR, 2017, pp. 4700–4708.

[7] C. Szegedy, V. Vanhoucke, S. Ioffe, and J. Shlens, "Rethinking the inception architecture for computer vision," in Proc. IEEE CVPR, 2016, pp. 2818–2826.

[8] V. Khubchandani, "Image caption generator using DenseNet201 and ResNet50," International Journal of Future Computer and Communication, vol. 13, no. 3, pp. 55–59, 2024.

[9] G. Dhand, A. Kumar, G. Grover, and D. Kaur, "Captioning images effectively: Investigating BLEU scores in CNN-LSTM models with different training configurations on Flickr8k," in Proc. ICIDSSD, Springer, 2024.

[10] J. Li, Y. Wang, and D. Zhao, "Layer-wise enhanced transformer with multi-modal fusion for image caption," Multimedia Systems, vol. 29, pp. 1043–1056, 2023.

[11] X. Yang, Y. Yang, S. Ma et al., "SAMT-generator: A second-attention for image captioning based on a multi-stage transformer network," Neurocomputing, vol. 593, p. 127823, 2024.

[12] Z. Yin and V. Ordonez, "Obj2text: Generating visually descriptive language from object layouts," arXiv preprint arXiv:1707.07102, 2017.

[13] R. Padate, A. Jain, M. Kalla, and A. Sharma, "Image caption generation using a dual attention mechanism," Engineering Applications of Artificial Intelligence, vol. 123, p. 106112, 2023.

[14] Anonymous et al., "An innovative multi-head attention mechanism-driven recurrent neural network model with feature representation fusion for enhanced image captioning," Scientific Reports, 2025.

[15] Y. Li, W. Zhang, K. Zhang et al., "Cross on cross attention: Deep fusion transformer for image captioning," IEEE Transactions on Circuits and Systems for Video Technology, 2023. doi: 10.1109/TCSVT.2023.3243725.

[16] M. Hodosh, P. Young, and J. Hockenmaier, "Framing image description as a ranking task: Data, models and evaluation metrics," Journal of Artificial Intelligence Research, vol. 47, pp. 853–899, 2013.

[17] S. Bai and S. An, "A survey on automatic image caption generation," Neurocomputing, vol. 311, pp. 291–304, 2018

[18] P. Bisht and A. Solanki, "Exploring practical deep learning approaches for English-to-Hindi image caption translation using transformers and object detectors," in Applications of Artificial Intelligence and Machine Learning: Select Proceedings of ICAAAIML 2021, Singapore: Springer Nature Singapore, 2022, pp. 47–60.

[19] A. Tayal, J. Gupta, A. Solanki, K. Bisht, A. Nayyar, and M. Masud, "DL-CNN-based approach with image processing techniques for diagnosis of retinal diseases," Multimedia Systems, vol. 28, no. 4, pp. 1417–1438, 2022

ISBN: 978-81-985770-0-9

Cite this article as:

Anjali Singh, Arun Solanki and et. al., *"DCAT: Dual CNN Encoder with Cross-Attention Transformer Decoder for Enhanced Image Captioning", Proceedings of 13th international conference on Microelectronics, Circuits and Systems, Micro2026.*

Displayed as online on 16th June 2026.

Link: http://actsoft.org/science/micro2026-pro/513-micro2026.pdf

ISBN: 978-81-985770-0-9