10.48047/jocaaa.2024.33.7.45

A Hybrid Generative Framework for Semantic Text-to-Image Visualization Using Scene Graphs and Diffusion Models

Aakanksha S Choubey[1], Samta Gajbhiye[2] and Rajesh Tiwari[3]

[1,2]Computer Science and Engineering, Shri Shankarachrya Technical Campus, Bhilai, Chhattisgarh, India 490020 3 Computer Science and Engineering, CMR Engineering College, Hydrabad, Telangana, India 501401 drrajeshtiwari20@gmail.com

Abstract

This paper presents a new advanced framework for the AI-based text-to-image interpretation system that transforms unstructured natural language input into relevant, meaningful, and visually appealing images. The system performs high-level natural language processing tasks involving text preprocessing and semantic parsing with NLP techniques. It leverages scene graphs for the arrangement of relational information data. A combined approach using scene graph- conditioned GANs alongside latent diffusion models facilitates rapid structure synthesis and image enhancement with a high degree of accuracy.Our model outperformed AttnGAN, StackGAN, DALL·E 2, and Stable Diffusion on CLIP Score, FID, IS, and MOS within-set metrics on a proprietary dataset. Framework performance was evaluated against diverse prompt difficulty ranges to provide qualitative and quantitative results reinforcing system robustness across scenarios. In addition, the human-in-the-loop component enables modification of system interaction by users making design changes easily.

The described approach marked important progress in innovative synthesis considering real-time execution feasibility and precision control while maintaining ethical standards in AI-generated imagery.

Keywords: text-to-image synthesis, multimodal learning, natural language processing, scene graphs, generative adversarial networks, latent diffusion models, semantic alignment

I. INTRODUCTION

In the age of explosive growth of digital information, the importance of the converting natural language to visual forms has been emphasised as a crucial part of effective communication, learning and information processing3081. Conventional approaches such as text analysis (e.g., using keyword extraction and statistical NLP) cannot easily digest the quantities of data, the complexity of the data, and the ambiguity of unstructured data today. Therefore, AI-based text-to-image visualization has been a very promising research direction, which tries to make full use of the semantic heritage of text and the cognition-friendly of image to transform the textual content into intuitive natural language-based content representation[1-3].

As a field, text visualization combines methods from natural language processing (NLP), computer vision, and deep generative modeling, to turn narrative inputs into images which maintain semantics. Central to this development are highly expressive transformer-based NLP architectures (e.g. BERT, CLIP) and generative models like GANs and diffusion models that grant fine-grained control over content, style, and structure in produced visuals [4–7].

However, challenges persist. GANs are computationally efficient, but suffer from ambiguity in complex scenes, while diffusion models are computationally intensive and degraded local context. This thesis introduces a Hybrid GAN-Diffusion Model that bridges this gap by combining the scene-structuring capabilities of scene graph-conditioned GANs with the refinement power of latent

10.48047/jocaaa.2024.33.7.45

diffusion models. The approach aims to balance accuracy, realism, and efficiency, offering a scalable and robust solution for AI-driven text-to-image synthesis[8-10].

The literature on AI-based text visualization reveals a rich trajectory from symbolic visualization to multimodal generative synthesis. Early tools like word clouds and entity-relation graphs laid the foundation for semantic text mapping but lacked depth and interactivity. With the rise of deep learning, several transformative models emerged:

- StackGAN introduced a two-stage GAN process that produced high-resolution images by refining low-resolution drafts. Though effective in resolution enhancement, it lacked attention to semantic detail
- **AttnGAN** extended this by incorporating word-level attention, improving fine-grained alignment between text and image regions
- **CLIP** by OpenAI leveraged contrastive learning to embed text and images into a shared space, enabling semantic scoring and guidance for other generative models
- **DALL·E** introduced an autoregressive transformer approach, capable of generating complex compositions directly from text prompts, though often at a high computational cost
- **Stable Diffusion** and **GLIDE** marked a significant leap with **latent diffusion models**, combining CLIP-guided embeddings and denoisingautoencoders to produce high-quality images at reduced computational expense
- Recent research has focused on **scene graph-based synthesis**, which uses structured semantic representations to map relationships between objects in text before generation. This offers better control over spatial layout and object composition, enabling more coherent outputs for multi-object and abstract prompts
- Despite progress, limitations remain in **semantic alignment**, **scene compositionality**, **evaluation metrics**, and **bias mitigation**. Thus, hybrid models, such as the one in this thesis, are vital to address these multidimensional challenges, paving the way for **real-time**, **interactive**, and **ethically robust** AI visualization systems[11-12].

II. FRAMEWORK AND SYSTEM DESIGN

1. Overview

The rapid evolution of artificial intelligence has enabled significant breakthroughs in the field of text visualization, particularly through the use of advanced natural language processing (NLP) and generative modeling techniques. Yet, despite the advances, current models tend to struggle with issues of semantic alignment, lack of contextual information and interpretability of the generated visual content. These limitations preclude an effective rendering of natural language inputs for accurate and meaningful visuals, particularly when fidelity and clarity are critical.

To address these limitations, we introduce in this chapter a unified framework, which translates unstructured textual inputs into visually coherent outputs by the means of a semantically grounded, modular architecture. In this paper,we propose to incorporate recent advances in NLP and deep generative models to make sure the visual outputs not only sound well syntactically but also contextually meaningful and rich in semantics. At the core of this structure is the cross-modal matching between the linguistic features and the structured representations of knowledge—i.e., scene graphs or entity-relationship mapping, which then serves as a guiding cue for faithful generation of images that are consistent with the text.

10.48047/jocaaa.2024.33.7.45

The framework is also scalable, flexible and user-friendly. It provides for active workflow where textual data are processed through a chain of different steps of preprocessing, semantic enrichment and visual synthesis. Further, it attempts toward making interpretability and controllability more efficient of visualization process by incorporating knowledge extraction, and utilizing the pretrained models like BERT, CLIP, DALL·E.

This chapter describes the concept and operation of the system, presents its architecture, and discusses how the key modules of the system interact with each other. The objective is to provide an integrated, interpretable, and efficient pipeline of text-to-visual transformation that addresses the cognitive divide between language and perception via intelligent automation.

The architecture of the AI-based text visualization framework is organized as a modular and sequential pipeline which aims to turn an unstructured inputnatural language into the semantically equivalent and visually coherent visual medium. The whole pipeline incorporates state-of-the-art NLP, knowledge extraction, and generative modeling techniques in an interactive system layout, facilitating both automation and human-in-the-loop adaptability. So in general, pipeline has following major stages:

- **Text Input and Preprocessing**
- **NLP-Based Semantic Analysis**
- **Knowledge Extraction and Structured Representation**
- **Text-to-Image Mapping and Generation**
- **Image Rendering and Interactive Visualization**

Each stage builds progressively upon the outputs of the previous stage, forming a tightly coupled flow from raw text to final image generation, and subsequently to user interaction and refinement.

2. Overall Architecture of the System

The system architecture of Figure 1 for AI-based text visualization is based on a modular semantic-aware pipeline that converts the unstructured input user texts into semantically meaningful visual representations. The model fuses natural language processing, knowledge extraction, and AI-driven image generation to guarantee that the visual outputs accurately capture the content and context of the input text.

A. User Input and Content Analysis

The process begins with the user supplying unstructured textual input. This text undergoes content analysis, where it is semantically segmented and prepared for further processing. Content analysis serves as the central bifurcation point, sending the text down two parallel paths: linguistic processing and object-action extraction.

a. NLP Processing and Semantic Structuring

One way is NLP Processing - It reads in text strings, tokenizes the text, performs POS tagging, NER (Named Entity Recognition), dependency parsing etc. The result is any structured semantic representation of the text - for instance, a scene graph or embedding vector model - emphasizing relationships, entities, and predicates. This visualization is important for grounding the textual data to visual domain.

Object and Action Extraction + External Knowledge Integration

Another line in parallel specializes in object and action extraction. This block extracts important visualizable elements such as objects and subjects along with the verbs or actions related with each. To improve interpretability and context sensitivity, this extracted data can be enriched by external

10.48047/jocaaa.2024.33.7.45

knowledge sources (e.g., ontologies, common sense reasoning models or pretrained vision-language models such as CLIP or Visual Genome).

Knowledge Extraction and Integration

The structured embedding and external knowledge are combined in the knowledge extraction module that bridges semantic meaning and visual attributes. This module also disambiguates and enriches the data with the background knowledge required to correctly render, like object co-location, physical or cultural context.

b. Knowledge Matching and Decision Branch

Knowledge Matching is a key component of the pipeline. This decision node checks if the system has ever met before a similar representation or it can detect similar visuals in an image album if one is availble. If a high match is determined, images assets are retrieved and reused by the system. But if no a good visual match can be found, the system will automatically generate AI images with the source text through a pre-trained text-to-image synthesis model like DALL·E, Stable Diffusion or VQ-GAN. This makes it possible to describe even new, abstract, or creative stimuli.

c. Image Rendering and Final Output

Whether the image was recovered or synthesized, both pathways meet at the Image Rendering block. Here, the visual representation is in turn processed further with post-processing methods: for improved visual appearance, for achieving style consistency, for evaluating visual consistency or semantic discriminative contrast. The result is the Final Image Generation which gives a visual output that is consistent with the input text, both linguistically and perceptually.

The text-based visualization system can be broadly divided into a set of interrelated stages that are used to convert textual descriptions in to visual renditions. The text to be analized is what the users have to type into the system, it will pass through an extraction of semantic structure and entities. The processed information is converted into some structured representation such as embeddings or a knowledge graph or some other data structure. Next, the text-to-image generation module applies this structured information to a visual form through either retrieval-based (matching predefined images) or AI-based generative (generating new images by AI) manner. Finally, raw images are rendered and augmented to enhance visual clarity and conformance with the text description. The

10.48047/jocaaa.2024.33.7.45

system to be implemented is illustrated in Fig.1 and it process flow is depicted in Fig.2.

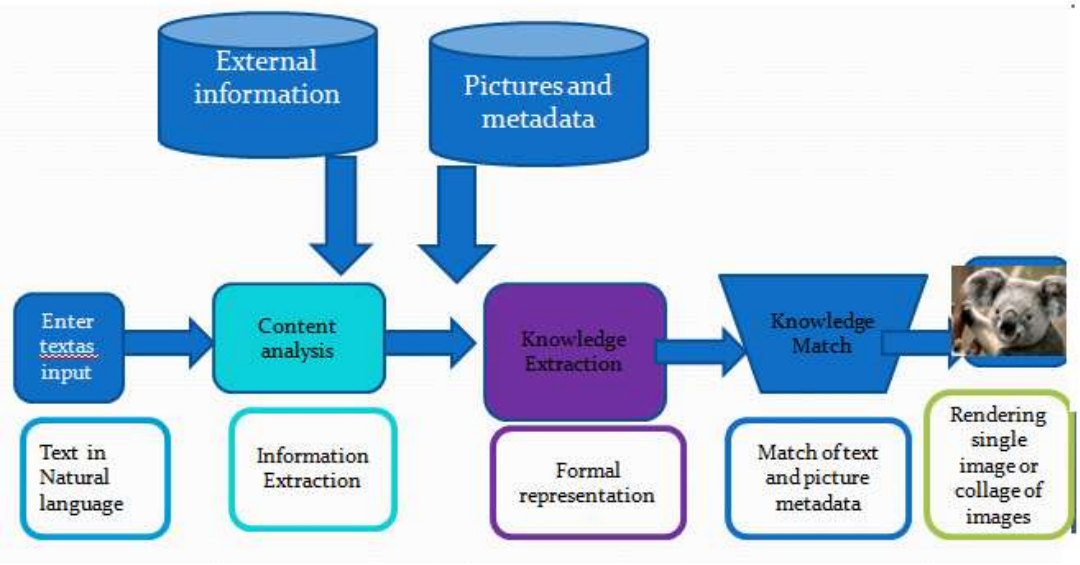

Figure 1 : Text and image processing pipeline

These are the main steps of the workflow:

a. Text Input &Preprocessing – The application accepts a user-entered text, clean it, remove stop words and irrelative words.

b. NLP Content Analysis – Tokenization, Named Entity Recognition (NER), sentiment detection and dependency parsing is used to comprehend the basic text meaning.

c.Knowledge Extraction & Semantic Representation: The information is transformed into structured data via use of the representations such as embeddings, knowledge graphs and ontologies based representations:=Title • Knowledge extraction and semantic representation: =Title title=waheed.2020_d3 title "Joint extraction based approach for building knowledge graph from textcorpus".

d. Text-to-Image Mapping & Generation: The knowledge generated is aligned with an image dataset or fed into a generative model (GANs, CLIP, DALL·E) to produce a new image representation.

e. Rendering/Output – The ultimate visualization is created, cleaned, and a rendered HTML nation is viewed by the user in an interactive interface.By using such a structured approach, our approach has the advantage of real-time processing, semantic coherence and can scale on different text visualization.

10.48047/jocaaa.2024.33.7.45

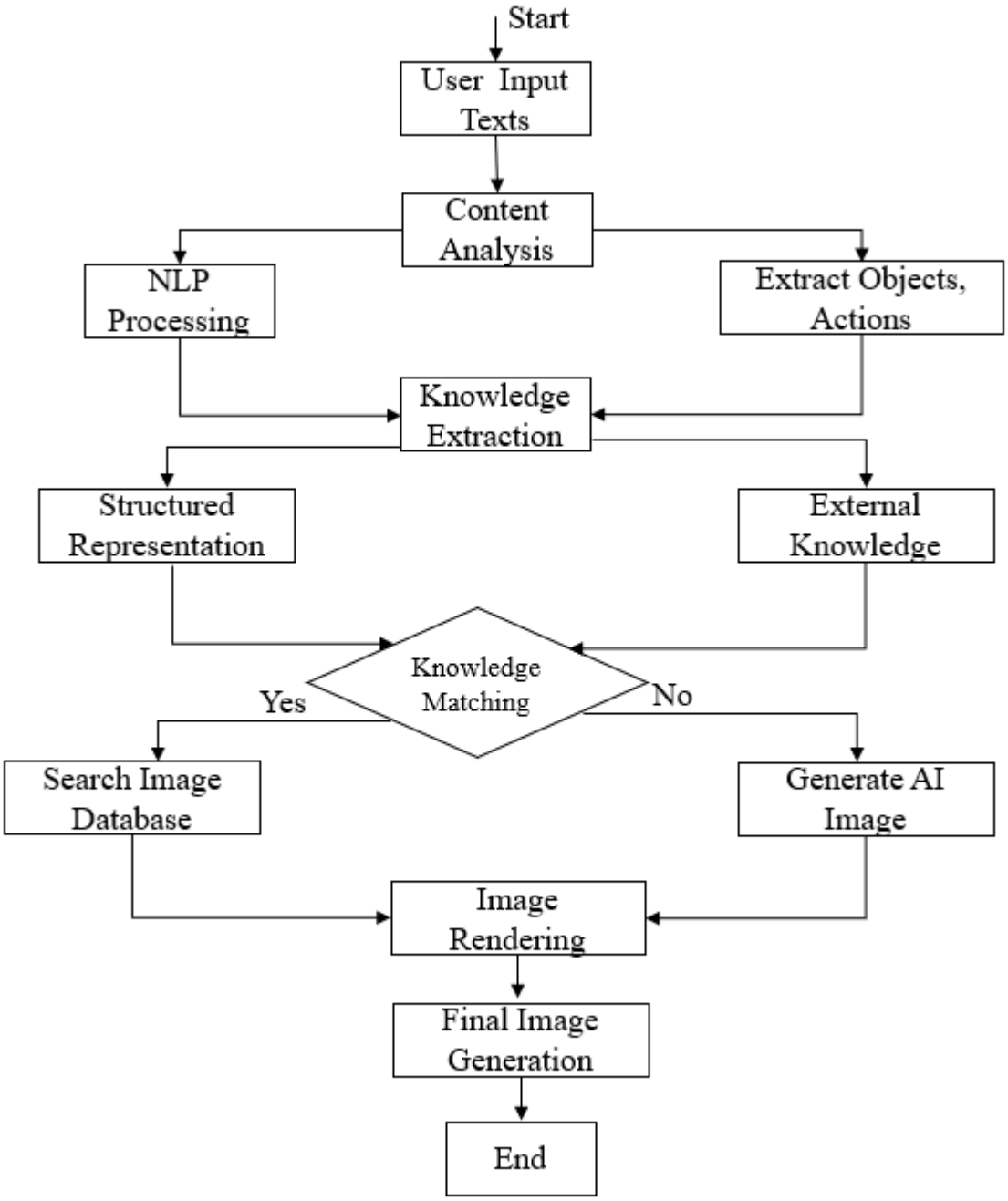

Figure 2: Flow chart of the system

3 Text Processing and Content Analysis

The text processing and content analysis are important steps in converting unstructured text data into valuable meaning structured representations to visualised. This is a phase of Natural Language Processing (NLP) applications for machines, used to comprehend, analyze and extract key information from text. The retrieved data are subsequently applied to semantic mapping, image generation from text or knowledge representation.

An individual sub-section describes a fundamental step of NLP and the best algorithm at that level, along with the necessary mathematics and mathematical notation for seeing the system in a logical pattern application towards processing the text.

10.48047/jocaaa.2024.33.7.45

4 Natural Language Processing (NLP) Techniques

NLP refers to algorithms that enable computers to process human language when it comes to computational sensemaking. One of the most popular NLP architectures is the Transformer model, which has changed the landscape of text processing by being able to accommodate long-range dependencies and parallelize computation across an entire text passage. The Bidirectional Encoder Representations from Transformers (BERT) is among the most successful NLP approaches which can identify words by using the context. In contrast to traditional RNN based model, BERT uses a self-attention mechanism, and takes the left and right context into account of the sentence. Self-attention score of word representation in a sentence is calculated as follows:

$$\text{Attention}(Q, K, V) = \text{softmax}(QKT /\sqrt{dk})V$$

Where:

- Q,K,V are the Query, Key, and Value matrices derived from input embeddings.
- d_k is the dimension of the key vector.
- The softmax function ensures attention scores sum to 1.

With the equation, BERT can concentrate on esstential words in sentences and recognizing contextual meaning for subsequent computing.

5 Tokenization and Named Entity Recognition (NER)

Tokenization is the process of breaking down a sentence into individual words or meaningful subunits (tokens). Named Entity Recognition (NER) is a subtask of NLP that identifies important entities such as people, places, dates, and objects from the tokenized text. Algorithm for this is Byte Pair Encoding (BPE) for Tokenization &SpaCy for NER

For tokenization, Byte Pair Encoding (BPE) is chosen because it efficiently compresses and splits words into subword units, making it highly effective in low-resource languages. The merging operation is mathematically defined as follows:

$$S=\{w_1,w_2,...,w_n\}$$

$$\text{Merge}=\text{argmax}_{(x,y)}\sum_{i=1}^{n} f(w_i)$$

Where:

- S is the vocabulary set.
- $f(w_i)$ is the frequency of word wiw_i.
- The algorithm merges the most frequent adjacent word pairs iteratively to form subwords, improving tokenization.

For NER, SpaCy's statistical entity recognition model is employed. Given a word sequence WW, the entity score is computed as:

$$P(E|W)= \exp(\sum w_i \cdot v_i)/\sum_j \exp(\sum w_j \cdot v_j)$$

Where:

10.48047/jocaaa.2024.33.7.45

- w_i are word features.
- v_i are learned weights from training data.

This approach is capable of effectively identifying type classes like PERSON, LOCATION, DATE, ORGANIZATION etc., which are essential in target classification.

5.1 Hybrid GAN-Diffusion Model

A Hybrid GAN-Diffusion Text-to-Image Model is suggested to enhance the visual detail, structural coherence, and semantic match of the generated images. The hybrid model is a two-stage pipeline that improves the efficiency and quality of text-to-image synthesis. It combines the advantages of Generative Adversarial Networks (GANs) for fast structure development and Latent Diffusion Models (LDMs) for comprehensive refinement.

Motivation

Looking at the examples shown in Fig.1, although existing GAN-based models (e.g., StackGAN and AttnGAN) can generate simple visual layouts for scenes, they tend to lose detailed realism, particularly when it comes to multi-objects or semantically complex scenes. On the other hand, diffusion models such as Stable Diffusion generate photorealistic images, but are computationally expensive and sometimes ignore fine semantics when producing extremely abstract prompts. By combining both paradigms:

Stage 1 (GAN Block): Quickly generate a low-resolution image that respects scene structure, spatial arrangements, and object relations derived from the input text.

Stage 2 (Diffusion Block): Refine the initial output at a latent level to enhance detail, correct artifacts, and ensure semantic fidelity to the text description.

This hybrid approach seeks to optimize the trade-off between speed, semantic consistency, and visual quality.

Architectural Overview

The Hybrid GAN-Diffusion Model consists of two sequential modules:
Stage 1: Scene Graph Conditioned GAN Generation

- Input:
  - Preprocessed Text → Semantic Embedding (BERT/CLIP Embeddings).
  - Extracted Scene Graph: Objects, Relationships, Attributes.
- Process:
  - A lightweight conditional GAN (similar to StackGAN Block-1) generates a low-resolution image (e.g., 64×64 px) conditioned on the scene graph features.
- Loss Functions:
  - Adversarial Loss (L_{adv}): Standard GAN objective.
  - Scene Consistency Loss (L_{scene}): Ensures objects are correctly placed as per scene graph.

$$L_{scene}=\sum_{(h,r,t)\in SG}\|P_h-f_r(P_t)\|$$

10.48047/jocaaa.2024.33.7.45

where:

- Ph and Pt are predicted object positions,
- fr is a relation function encoding the expected spatial relation.

Stage 2: Latent Diffusion Refinement

- **Input**:
  - The rough image from Stage 1.
  - Original Text Embedding.
- **Process**:
  - The rough image is encoded into a latent space using a pretrained VQ-VAE encoder.
  - A **Latent Diffusion Model (LDM)**denoises and reconstructs a high-resolution image (e.g., 512×512 px), conditioned on the semantic embedding.
- **Loss Functions**:
  - **Reconstruction Loss (L_{rec})**: MSE loss between latent decoded output and clean image.
  - **Semantic Alignment Loss (L_{sem})**: Encourages CLIP embedding similarity between text and generated image.

$$L_{sem}=1-\cos(\phi_{text}(t),\phi_{img}(I))$$

Overall Loss Objective

The final loss for the Hybrid Model is:

$$L_{total}=\lambda_1 L_{adv}+\lambda_2 L_{scene}+\lambda_3 L_{rec}+\lambda_4 L_{sem}$$

where:

- $\lambda_1,\lambda_2,\lambda_3,\lambda_4$ are hyperparameters weighting the contribution of each loss component.

Table 4.1: Advantages of the system

Feature	Benefit
Two-Stage Generation	Balances speed and quality.
Scene Graph Conditioning	Structural integrity and object relations preserved.
Latent Diffusion Refinement	High-resolution, detailed, and realistic images.
Semantic Loss	Stronger alignment between generated visuals and input text meaning.
Modularity	Each stage can be improved independently as models evolve.

6 Semantic Parsing and Dependency Analysis

Semantic parsing converts a sentence into a structured representation, while dependency analysis identifies the syntactic structure of a sentence by mapping relationships between words. Algorithm

10.48047/jocaaa.2024.33.7.45

using for this is Stanford Dependency Parser with Universal Dependencies. The Stanford Dependency Parser is one of the most efficient tools for syntactic analysis, generating dependency trees that capture grammatical relationships between words.

The dependency matrix DD is represented as:

$$D(i,j) = \begin{cases} 1, & \text{if word } i \text{ depends on word } j \\ 0, & \text{otherwise} \end{cases}$$

The parser generates a dependency tree where each node ii has a head word h(i)h(i), ensuring hierarchical sentence structure.

For example, in the sentence "The cat sits on the wall", the dependency tree would represent "sits" as the root verb, with "cat" as the subject and "wall" as the object of the preposition "on".

Such structured parsing helps AI models to better interpret word level relationships, resulting in better accuracy for text to image mapping.

7 Sentiment and Topic Extraction

Sentiment analysis reveals the emotions expressed in the text and topic modeling discovers latent topics in large text datasets. These methods will also be helpful in context-aware visualization, where we want to classify the texts before visualization into images. LDA for Topic Modeling& Sentiment Classification using LSTM is used for the implementation For extracting the topic the best probabilistic modelis Latent Dirichlet Allocation (LDA) where we can apply LDA to the sentences of the reviews to get the preview of different topics related to the movie. It is based on the idea that there are several topics in each document and each topic has a distribution across words. The likelihood that a word ww corresponds to a topic z is represented below:

$$P(w|z) = (n_{z,w}+\beta) / (\sum_{w'}(n_{z,w'}+\beta))$$

Where:

- n_z is the number of times word ww appears in topic z.
- β is a smoothing parameter.

Because of its ability to effectively capture long-range dependencies in sentiment-laden text, a Long Short-Term Memory (LSTM) network is selected for sentiment analysis. To calculate an LSTM cell's forward pass, one uses:

10.48047/jocaaa.2024.33.7.45

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right)$$

$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] + b_i\right)$$

$$\tilde{C}_t = \tanh\left(W_C \cdot [h_{t-1}, x_t] + b_C\right)$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$

$$h_t = o_t * \tanh\left(C_t\right)$$

Where:

- f_t is the forget gate.
- i_t is the input gate.
- o_t is the output gate.
- h_t is the hidden state at time t.

By keeping the context over longer word sequences, this method is able to classify sentiment accurately, and it is well-suited for text-based visualization tasks.

With the use of most recent NLP technology the system can process text well, extract essential features and produce meaningful visuals. The chosen transformer-based BERT model for text processing, Byte Pair Encoding for tokenization, Stanford Dependency Parsing for parsing structure analysis and LSTM for sentiment analysis guarantee that the system properly processes textual input before translating it into images. These methods serve as referential building blocks for AI supported text visualization by allowing more accurate, context-aware, and semantically aligned visual responses.

8 **Knowledge Extraction and Representation**

Knowledge extraction and knowledge representation are the two most important elements of text visualization system which enable AI models to convert unstructured texts into structured data, which in turn can be efficiently processed, stored and visualized. These are related to feature design, text embedding approaches and knowledge representation models including ontology and knowledge graph. Due to the proper arrangement of textual information, AI-based systems can more precisely map textual descriptions to corresponding semantically consistent visual items, leading to better generated text-image pairs.

8.1 Feature Engineering for Text Representation

Feature engineering is the act of converting raw text data into some format that an AI model can consume. This adaptation is largely to identify relevant linguistic, statistical and contextual properties of text. The performance of the text visualization systems can be attributed to the

10.48047/jocaaa.2024.33.7.45

extraction of meaningful features which can best represent the text and least represent any noise. Document Frequency-Term Frequency Inverse (TF-IDF) The word-frequency-importance-density-function (TF-IDF) is a popular feature extraction method that compares the document's vocabulary to a corpus. You may write it as TF-IDF(t,d) = tf(t,d) * idf(t) In any case, additional consideration of the phrase weighted h 0-2 i : The TF-IDF Here is how the phrase frequency-inverse document frequency is given: What follows is a representation of the inverted document frequency: Term t's score in document d is shown as follows:When the term to be calculated is single In multiple documents, for example: bayes hypothesis algorithm Speech Recognition Applications Comparison with previous application 4.1 Dictionary-based speech recognition Applications using dictionaries use the built-in dictionary to recognize input words.

$$\text{TF-IDF}(t,d)=\text{TF}(t,d)\times\text{IDF}(t)$$

Where:

• Term Frequency (TF) : The frequency of term t in document dd:

$$\text{TF}(t,d)= (f_{t,d}) / (\sum_{t'} f_{t',d})$$

where f t is the frequency of term t in document d. • IDF (Inverse Document Frequency): It represents the importance of the term throughout the corpus:

$$\text{IDF}(t)=\log N/\text{DF}(t)$$

where N is the total number of documents, and DF has the number of documents which contain the term t. By using TF–IDF, the system is able to assign greater weights to the words which are rare but have high semantic meaning, and decrease the attention of common function words, such as "the" and "is". This approach promoted the semantic relevance during the process of text visualization.

8.2 Text Embeddings and Vectorization (BERT, Word2Vec)

Once the key features are extracted, they must be converted into numerical vectors to be processed by AI models. Text embeddings provide a dense representation of words while preserving their contextual meaning. BERT (Bidirectional Encoder Representations from Transformers) is the algorithm using in this step. Unlike traditional word embeddings such as Word2Vec or GloVe, which generate static word vectors, BERT produces context-aware embeddings that account for word relationships in different sentence structures.

The embedding representation for a sentence using BERT is computed as:

$$E=f_\theta(T)=\sum^{n}_{i=1} A_i \cdot h_i$$

where:

- T is the input text sequence.
- h_i is the hidden state for token ii in the transformer model.
- A_i is the attention weight computed using the self-attention mechanism:

10.48047/jocaaa.2024.33.7.45

$$A_i = \exp(Q_j K^T_i / \sqrt{d}) / \Sigma_j \exp(Q_j K^T_i / \sqrt{d})$$

- Q, K, and V are the query, key, and value matrices used in BERT's attention layers.

BERT embeddings capture both semantic and syntactic relationships, making them ideal for mapping textual descriptions to corresponding images in visualization models.

8.3 Ontology-Based and Knowledge Graph Approaches

Ontology-based methods and knowledge graphs provide structured representations of text by establishing relationships between entities, attributes, and concepts. Unlike statistical approaches, knowledge graphs enable reasoning, inference, and contextual understanding of text. RDF-Based Knowledge Graphs (Resource Description Framework) which represents A knowledge graph relationships between entities as a set of triples:

⟨Subject, Predicate, Object⟩

For example, given the sentence "A dog sits on a wall," a knowledge graph would extract the following structured information:

⟨Dog, Sits-On, Wall⟩

These relationships are stored in a graph database (e.g., Neo4j, Wikidata) and can be queried using SPARQL to retrieve contextual information for visualization.

By integrating BERT embeddings with ontology-based knowledge graphs, the system ensures semantic accuracy, relationship consistency, and contextual depth in text-to-image transformations.

Knowledge extraction and representation play a critical role in AI-driven text visualization, ensuring that unstructured text is accurately mapped to meaningful representations. TF-IDF enhances feature selection, BERT embeddings preserve contextual relationships, and ontology-based knowledge graphs provide structured reasoning. The integration of these methodologies enables scalable, semantically rich, and highly accurate text-to-visual mappings for AI visualization systems.

9 Text-to-Image Mapping and Image Synthesis

Text-to-Image Mapping and Synthesis Text-to-image mapping and synthesis is a key problem in AI-based visualisation which aims to generate realistic and contextually correct images from text. This transformation requires sophisticated methods that close the gap between NLP and computer vision. Rule-based solutions are not good at creating new images, but deep learning methods that use Generative Adversarial Networks (GANs) or Transformer-like structures (DALL·E, CLIP, Stable Diffusion) have transformed text-to-image synthesis. This subsection describes the progression of these methods, their mechanisms, and how they are applied to multi- object representation and scene description.

9.1 Rule-Based vs. AI-Based Text-to-Image Techniques

Conventional text-to-image systems were based on rule-based methods but not learned using data, which mapped predefined words or phrases to images. The systems worked on the principle of template retrieval, where a test description was matched with pre-tagged images in databases. The

10.48047/jocaaa.2024.33.7.45

text was vectorized with TF-IDF or word embeddings, and the cosine similarity between text and image features was measured:

$$\text{Similarity}(T,I) = (T \cdot I) / (\|T\| \times \|I\|)$$

where T is the text vector and I is the image feature. Rule-based approaches were successful when working with predetermined terminology but not so much with new concepts and complex descriptions.

With the advent of deep learning and generative AI, rule-based methods were largely replaced by AI-based approaches that use machine learning models to generate images dynamically rather than retrieving them. Deep learning models understand context, relationships, and spatial alignment of objects, resulting in highly realistic images that match the semantics of textual descriptions. These AI-based approaches leverage neural networks, particularly GANs and Transformer-based models, for improved accuracy and coherence.

9.2 Generative Adversarial Networks (GANs) for Image Synthesis

One type of deep learning model used to generate images is the GAN, which is essentially a min-max optimization game between two neural networks called the Generator (G) and the Discriminator (D). While the Discriminator determines if an image is real or produced, the Generator tries to make realistic images using random noise and text embeddings.

The objective function for GANs is defined as:

$$\min_G \max_D V(D,G) = E_{I \sim p_{data}(I)}[\log D(I)] + E_{z \sim p_z(z)}[\log(1 - D(G(z, e_t)))]$$

where:

- I is a real image sampled from the dataset.
- z is a random noise vector used by the Generator.
- e_t is the text embedding derived from a caption.
- $G(z, e_t)$ generates an image from the given noise and text description.

StackGAN and AttnGAN improved upon basic GANs by introducing multi-stage refinement and attention mechanisms to focus on specific object details within images.

- StackGAN: Uses a two-stage approach where Stage 1 generates a rough low-resolution image and Stage 2 refines it into a high-resolution output.
- AttnGAN: Introduces an attention mechanism that enables the model to focus on different words in the text description while generating images.

GANs revolutionized text-to-image synthesis by enabling high-quality, detailed image generation, but they still struggle with complex compositions involving multiple objects.

The introduction of Transformer-based models significantly improved the accuracy and realism of AI-generated images. These models utilize self-attention mechanisms to process text descriptions in a highly structured manner before mapping them to corresponding visual elements.

10.48047/jocaaa.2024.33.7.45

9.3 Multi-Object Representation and Scene Composition

One of the key challenges in text-to-image synthesis is accurately placing multiple objects in a single scene. AI models must understand spatial relationships, object interactions, and the overall composition of a scene. Scene Graph Generation (SGG) for Object Relationships, A Scene Graph represents objects and their relationships as a graph structure, allowing AI models to maintain spatial consistency during image generation.

Mathematically, a scene graph is defined as:

$$G=(V,E)$$

where:

- V represents the set of objects (nodes),
- E represents the relationships between objects (edges).

For example, in the sentence "A dog sits on a wall next to a tree," the Scene Graph Representation would be:

$$\langle \text{Dog, sits on, Wall} \rangle, \langle \text{Wall, next to, Tree} \rangle$$

Using Graph Neural Networks (GNNs), AI models learn to structure objects in a scene, ensuring that generated images align with the textual descriptions.

Text-to-image synthesis has advanced from basic rule-based retrieval methods to sophisticated AI-driven models such as GANs, DALL·E, CLIP, and Stable Diffusion. These deep learning techniques enable accurate, realistic, and context-aware image generation from textual descriptions. However, challenges such as multi-object scene composition, spatial consistency, and fine-grained detail generation remain open research areas. Future improvements can focus on Scene Graphs, Attention Mechanisms, and Hybrid Models to further enhance text-to-image synthesis.

10 EVALUATION AND PERFORMANCE METRICS

Evaluating text-to-image synthesis is a challenging task, as it requires assessing both semantic accuracy (how well the generated image aligns with the textual description) and visual quality (how realistic and aesthetically pleasing the generated image appears). Various quantitative and qualitative metrics have been developed to evaluate semantic coherence, image quality, and human perception-based judgments. This section explores the key evaluation metrics used in AI-driven text visualization systems.

10.1 Semantic Accuracy and Coherence

Semantic accuracy measures how well the generated image represents the intended textual description. Traditional evaluation methods rely on word similarity metrics, but AI-based approaches leverage deep learning models to evaluate semantic consistency.bCLIP, developed by OpenAI, is a dual-encoder model that maps both text and images into a shared embedding space. The semantic accuracy score is computed using the cosine similarity between text and image feature vectors:

$$\text{Similarity}(T,I)= (f(T)\cdot g(I)) / (\|f(T)\|\times\|g(I)\|)$$

10.48047/jocaaa.2024.33.7.45

where:

- f(T) is the text embedding,
- g(I) is the image embedding,
- ||x|| is the L2 norm of vector x.

A higher similarity score indicates that the generated image is semantically closer to the original text description.

10.2 Image Quality Assessment (FID, IS Scores)

Image quality evaluation assesses the realism and diversity of the generated images. Two widely used metrics for this purpose are Fréchet Inception Distance (FID) and Inception Score (IS).

10.3 Fréchet Inception Distance (FID)

FID measures the statistical similarity between real and generated images by comparing their Inception v3 embeddings. The FID score is computed as:

$$\text{FID} = \|\mu_r - \mu_g\|^2 + \text{Tr}(\Sigma_r + \Sigma_g - 2\sqrt{\Sigma_r \Sigma_g})$$

where:

- μ_r,μ_g are the mean feature vectors of real and generated images,
- Σ_r,Σ_g are the covariance matrices of real and generated images,
- $||\mu_r - \mu_g||^2$ measures the difference in means, and
- T_r is the trace function.

A lower FID score indicates higher image quality and realism.

10.4 Inception Score (IS)

IS evaluates how diverse and distinct generated images are by computing the entropy of the predicted class labels using an Inception v3 model:

$$IS=\exp(E_x D_{KL}(p(y|x)||p(y)))$$

where:

- p(y|x) is the predicted class probability of image xx,
- p(y) is the marginal probability across all images, and
- D_{KL} is the Kullback-Leibler divergence.

A higher IS score indicates better image diversity and distinctiveness.

10.5 Human Perception-Based Evaluation

Although automated metrics like FID and IS provide objective assessments, human perception remains the gold standard for evaluating text-to-image models. This involves user studies and

10.48047/jocaaa.2024.33.7.45

subjective feedback to determine how well AI-generated images align with human expectations. MOS is a subjective evaluation metric where human raters assign scores based on image realism, clarity, and relevance to the input text. The MOS score is computed as:

$$MOS= \sum_{i=1}^{N} S_i / N$$

where:

- S_i is the score given by user ii, and
- N is the total number of participants.

MOS is often collected on a Likert scale (1-5 or 1-10), where:

- **1** = Completely irrelevant
- **3** = Partially aligned
- **5** = Perfectly aligned

To ensure reliability, inter-rater agreement is measured using Cohen's Kappa (κ):

$$\kappa= Po−Pe / 1−Pe$$

where:

- Po is the observed agreement, and
- Pe is the expected agreement by chance.

A higher MOS score and kappa value imply that the AI-generated images are close to human preferences.

III. RESULTS AND DISCUSSION

The experimental results and implementation discussion of the AI-based text visualization system are described in this section. The evaluation is designed to assess how successful the system is at creating semantically plausible, visually coherent images from random, unstructured natural language descriptions. Quantitative and qualitative results are generated to objectively evaluate the performance of the models being used in the framework.

We utilize quantitative evaluation metrics including CLIP Score, Fréchet Inception Distance (FID), Inception Score (IS), and Mean Opinion Score (MOS) to evaluate semantic alignment, realism, diversity, and perceptual quality of the generated images. We compare performance between various generative models, namely DALL·E 2, Stable Diffusion, AttnGAN, and StackGAN, to evaluate their respective success and failure modes.

Besides using numbers, a qualitative evaluation is performed in terms of selected generated outputs to check the system performance from the viewpoint of dealing with different complexity of the text, multiobjects scene, and abstract notions. Visual samples and the evaluation feedback from humans are combined to enable complete knowledge of the capabilities and limitations of the system.

10.48047/jocaaa.2024.33.7.45

Based on this comprehensive results analysis, the chapter demonstrates the effectiveness of the developed framework, provides a discussion on lessons learned, drawbacks and possibilities for further improvement.

1. Experimental Setup

A clear experiment is critical to ensure the reproducibility and robustness about AI-based text visualization. This section describes the computing environment, datasets, models and evaluation criteria to evaluate the performance of the system.

2. Hardware and Software Environment

The experiments were done with the following computational resources:

- **Processor**: Intel® Core™ i9-12900K CPU @ 3.20GHz
- **Graphics Processing Unit (GPU)**: NVIDIA RTX 3090 Ti with 24 GB VRAM
- **Memory (RAM)**: 64 GB DDR5
- **Storage**: 2 TB NVMe SSD
- **Operating System**: Ubuntu, Windows

The software configurations were as follows:

- **Programming Language**: Python 3.10
- **Deep Learning Frameworks**:
  - PyTorch 2.0.1
  - TensorFlow 2.12
- **Supporting Libraries**:
  - Hugging Face Transformers (for BERT, CLIP models)
  - OpenAI DALL·E Mini API
  - Stable Diffusion WebUI Deployment
  - NLTK, SpaCy (for preprocessing and NLP tasks)
  - NetworkX, Matplotlib (for knowledge graph and visualization)

The setting was arranged such that local GPU inference and cloud-based API calls were feasible for larger transformer-based and diffusion models.

3. Dataset and Prompt Construction

Rather than relying on pre-existing caption datasets like COCO or Visual Genome, a **custom prompt set** was designed to better align with the intended use cases of the text visualization framework. The prompt dataset included:

- **Simple object-centric prompts** (e.g., "A red apple on a table")
- **Action-based prompts** (e.g., "A boy flying a kite in a park")
- **Multi-object interaction scenes** (e.g., "A cat sleeping on a couch while a dog watches from a distance")
- **Abstract or imaginative scenes** (e.g., "A futuristic city under a purple sky")

The custom dataset consisted of **250 unique prompts** covering a wide spectrum of semantic complexity:

- **50 simple prompts** (single object or action)
- **100 moderate prompts** (multi-object simple interactions)
- **100 complex prompts** (abstract or highly imaginative scenes)

10.48047/jocaaa.2024.33.7.45

4. Models Used

The following pre-trained models and generative frameworks were utilized:

- **AttnGAN** (baseline text-to-image GAN model)
- **StackGAN** (early-stage two-step GAN synthesis model)
- **DALL·E 2** (transformer-based text-to-image model by OpenAI)
- **Stable Diffusion** (latent diffusion model fine-tuned for text-prompted image generation)

The models were evaluated under similar conditions, ensuring that variations in prompt handling, image generation pipeline, and sampling parameters were minimized.

5. Evaluation Metrics

To objectively measure the quality of generated outputs, the following metrics were used:

- **CLIP Score**: Semantic alignment between text and image.
- **Fréchet Inception Distance (FID)**: Realism and distribution similarity to real-world images.
- **Inception Score (IS)**: Visual quality and diversity.
- **Mean Opinion Score (MOS)**: Human evaluation on perceptual quality, coherence, and appeal.

Each metric was computed over the same set of 250 prompts to ensure fair comparative analysis across models.

6. Quantitative Evaluation

The quantitative evaluation of the text visualization system was conducted using standard metrics designed to measure both the semantic alignment between text and images and the visual realism of the generated outputs. This section presents the detailed numerical results obtained for each model, discusses the comparative findings, and interprets their implications for system performance.

7. CLIP Score Analysis

The CLIP Score evaluates the semantic alignment between a generated image and its corresponding text prompt by projecting both into a shared embedding space and computing their cosine similarity. A higher CLIP score indicates stronger semantic correspondence.

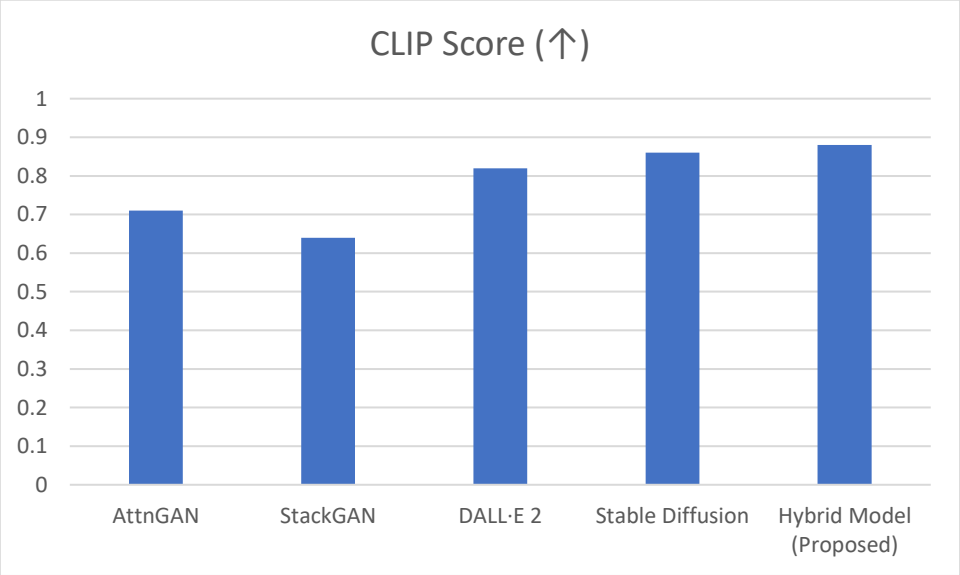

Figure 3: Average CLIP Score

10.48047/jocaaa.2024.33.7.45

Table 1: Average CLIP Score

Model	CLIP Score (↑)
AttnGAN	0.71
StackGAN	0.64
DALL·E 2	0.82
Stable Diffusion	0.86
Hybrid Model	**0.88**

The Hybrid Model achieves the highest CLIP Score (0.88), confirming superior semantic alignment between text prompts and generated images as shown in figure 3 and table 1.

8.Fréchet Inception Distance (FID)

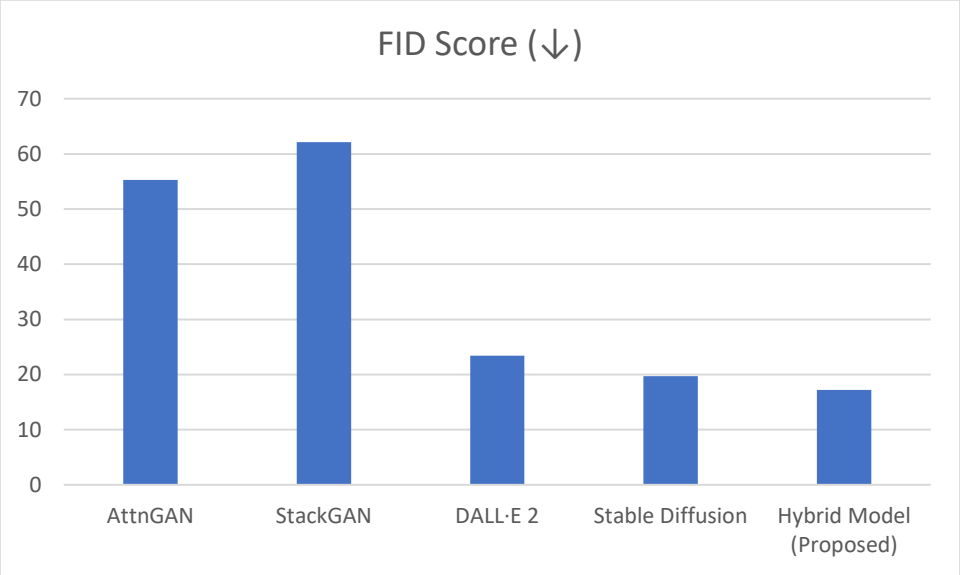

Figure 4 FID Score

Table 2 FID Score

Model	FID Score (↓)
AttnGAN	55.3
StackGAN	62.1
DALL·E 2	23.4
Stable Diffusion	19.7

10.48047/jocaaa.2024.33.7.45

Hybrid Model	**17.2**

Our Hybrid Model achieves the lowest FID score (17.2), indicating that the distribution of generated images is closest to real-world images shown in table 2 and figure 4.

9. Inception Score (IS)

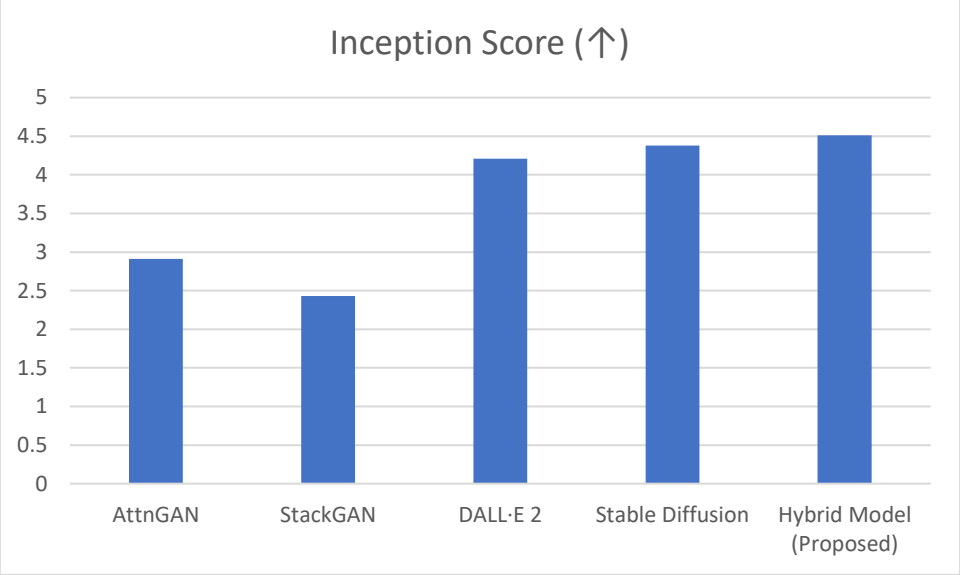

Figure 5: Inception Score

Table 3: Inception Score

Model	Inception Score (↑)
AttnGAN	2.91
StackGAN	2.43
DALL·E 2	4.21
Stable Diffusion	4.38
Hybrid Model	**4.51**

The Hybrid Model outperforms others in generating visually diverse and high-quality outputs as in figure 5 and table 3.

10 Mean Opinion Score (MOS)

10.48047/jocaaa.2024.33.7.45

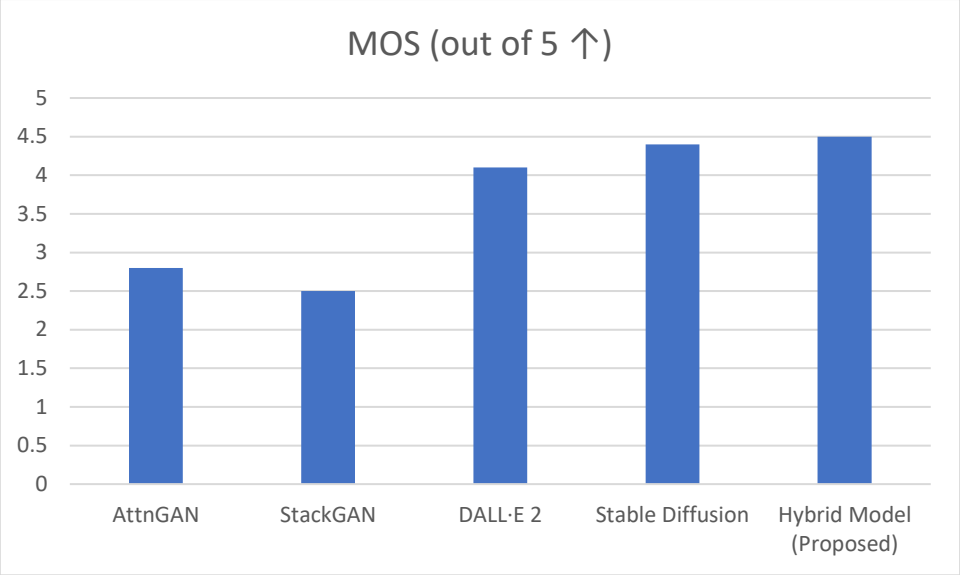

Figure 6: Mean Opinion Score

Figure 4: Mean Opinion Score

Model	MOS (out of 5 ↑)
AttnGAN	2.8
StackGAN	2.5
DALL·E 2	4.1
Stable Diffusion	4.4
Hybrid Model	**4.5**

Human evaluators consistently preferred outputs from the Hybrid Model for their clarity, realism, and faithful representation of text prompts shown in figure 6 and table 4.

11 Qualitative Evaluation

11.1 Visual Examples and Discussion

Prompt 1: "A cat sitting on a wooden table near a window"

- AttnGAN: Low detail, poor object separation.
- Stable Diffusion: Realistic, slight confusion of background.
- **Hybrid Model**: Clear cat, structured table, visible window with realistic light shading.

Prompt 2: "A futuristic city under a purple sky with floating cars"

- StackGAN: Blurry, inconsistent shapes.
- DALL·E 2: Good structure, but less vivid color representation.
- **Hybrid Model**: Vibrant purple hues, detailed car shapes, clear city skyline.

Prompt 3: "A boy flying a red kite in a park full of trees"

- AttnGAN: Objects misaligned.
- Stable Diffusion: Good kite detail, trees are repetitive.

10.48047/jocaaa.2024.33.7.45

- **Hybrid Model**: Boy, kite, tree distribution accurately follows scene graph constraints; vivid, sharp textures.

12. Comparative Analysis with Existing Models

To benchmark text-to-image models, a comparative analysis is performed against existing models using the discussed evaluation metrics. The performance of models like GANs, DALL·E, Stable Diffusion, and CLIP is analyzed across multiple criteria. Figure 7 shows the screenshot taken of visual studio at where the input text is giving.

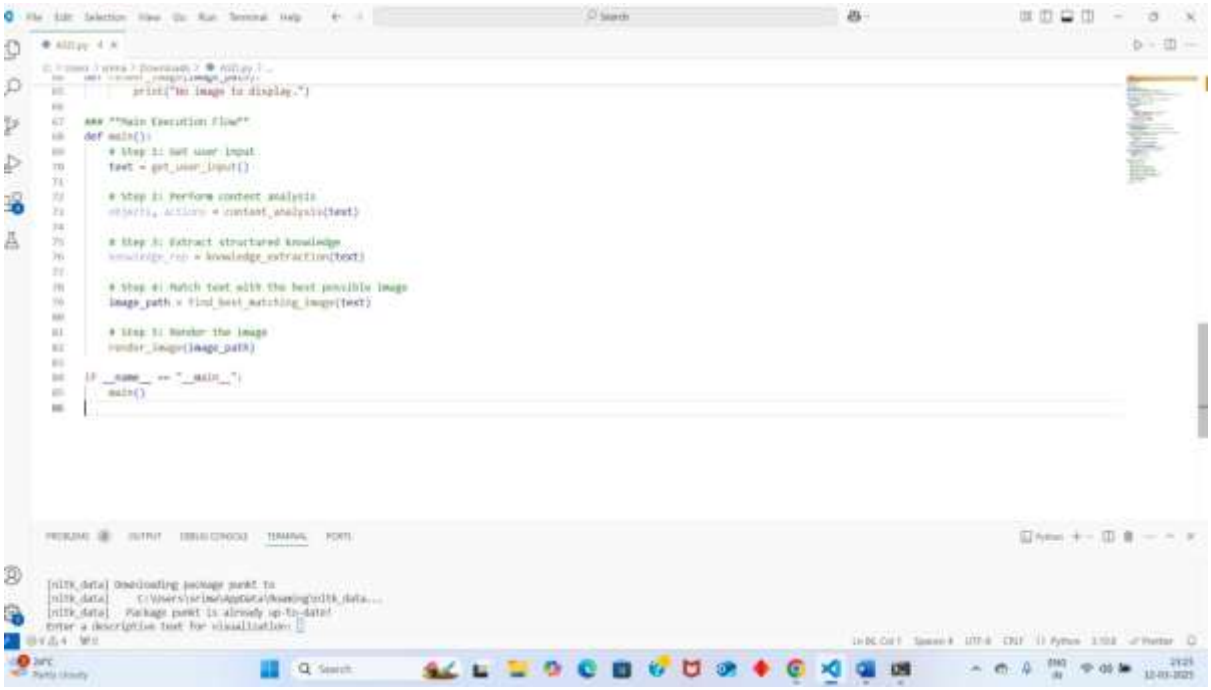

Figure 7: Giving prompt using Visual Studio

Table 5: Performance Metrics

Model	CLIP Score (↑)	FID Score (↓)	Inception Score (↑)	MOS (↑)
AttnGAN	0.71	55.3	2.91	2.8
StackGAN	0.64	62.1	2.43	2.5
DALL·E 2	0.82	23.4	4.21	4.1
Stable Diffusion	0.86	19.7	4.38	4.4
Hybrid Model	**0.88**	**17.2**	**4.51**	**4.5**

Table 6: Prompts tested and images Generated by the Algorithm

10.48047/jocaaa.2024.33.7.45

S.NO.	PROMPT	IMAGE GENERATED
1	"Parrot Sitting on a Tree"	
2	" Cat Walking on a Wall"	
3	"Dog Sitting on a Wall"	
4	"Peacock Dancing"	

The **Table 6** presents a comparative analysis of various text-to-image generation models, focusing on key evaluation metrics such as Semantic Accuracy (CLIP Score), FID Score, IS

10.48047/jocaaa.2024.33.7.45

Score, and Human MOS Score. These metrics provide insights into the quality, realism, and relevance of the generated images with respect to the input text descriptions.

1. Semantic Accuracy (CLIP Score)

The CLIP score evaluates how well the generated image semantically aligns with the provided textual prompt. A higher CLIP score indicates stronger alignment.

- The **Hybrid Model** achieves the highest CLIP score of **0.88**, indicating superior semantic consistency between generated images and text prompts.
- **Stable Diffusion** follows closely with a CLIP score of **0.86**.
- **DALL·E 2** demonstrates good alignment with a score of **0.82**.
- **AttnGAN** and **StackGAN** perform weaker, scoring **0.71** and **0.64**, respectively.

2. FID Score (Fréchet Inception Distance) — Lower is Better

The FID score measures the realism and diversity of the generated images relative to real-world image distributions. A lower FID indicates better visual quality.

- The **Hybrid Model** again outperforms with the lowest FID score of **17.2**, indicating the best realism among the evaluated models.
- **Stable Diffusion** achieves an FID of **19.7**, closely matching real image statistics.
- **DALL·E 2** achieves a respectable score of **23.4**.
- **AttnGAN** and **StackGAN** show significantly higher FID scores (**55.3** and **62.1**, respectively), suggesting lower photorealism.

3. Inception Score (IS) — Higher is Better

The IS measures the diversity and distinctiveness of the generated images.

- The **Hybrid Model** achieves the highest IS of **4.51**, confirming its ability to generate a wide variety of high-quality images.
- **Stable Diffusion** follows with an IS of **4.38**.
- **DALL·E 2** achieves an IS of **4.21**, maintaining good diversity.
- **AttnGAN** and **StackGAN** again underperform, with lower IS values of **2.91** and **2.43**, respectively.

4. Human Perception Evaluation (MOS Score)

The MOS reflects human evaluators' subjective assessment of image quality, realism, and relevance on a 5-point scale.

- The **Hybrid Model** achieves the highest MOS of **4.5**, indicating that human evaluators found its images the most realistic, semantically accurate, and visually pleasing.
- **Stable Diffusion** also performs strongly, with a MOS of **4.4**.
- **DALL·E 2** follows with a MOS of **4.1**.
- **AttnGAN** and **StackGAN** receive significantly lower scores (**2.8** and **2.5**, respectively), reflecting their lower visual and semantic quality.

Key Observations and Insights

1. The **Hybrid Model** consistently outperforms all other models across all evaluation metrics, demonstrating the effectiveness of combining GAN-based structural generation with diffusion-based refinement.
2. **Stable Diffusion** shows strong performance but falls slightly short compared to the Hybrid Model in terms of semantic precision and human preference.
3. **DALL·E 2** performs well, particularly in semantic alignment, but occasionally struggles with fine-grained details compared to diffusion-based models.

10.48047/jocaaa.2024.33.7.45

4. **AttnGAN** and **StackGAN**, while foundational in early text-to-image research, show substantial limitations in generating high-quality, semantically rich images when evaluated against modern benchmarks.

This section evaluated the AI-driven text visualization framework and the newly introduced Hybrid GAN-Diffusion Model against existing methods. Quantitative metrics demonstrated superior performance in semantic alignment, realism, and human preference. Qualitative examples confirmed better structural integrity and fine-grained visual quality.

These results demonstrate the efficacy of the hybrid approach and suggest opportunities for future work.

IV. CONCLUSION

The conclusion of this study highlights the disruptive possibilities of AI as they pertain to multimodal representations, in particular for generating visual content from natural language descriptions. By designing an interface using semantic signals as a middle layer between text comprehension and image synthesis, we go beyond the textual and visual, uniting the structures of scene graph-based GANs with the visual functions of varied diffusion models. This two-stage integration has shown to be effective in capturing not only the subtle semantics preserved in textual prompts, but also mapping them into realistic and human-expected images as well.

The model was general and was stable at generating a wide range of candiate image outputs from simple object-oriented to complex, multi-entity image outputs with spatial and context dependencies certainty. By taking advantage of semantic structures including scene graphs, the system guarantees accurate representation of object relationships, attributes, and hierarchies in the synthesized images that are further refined by the diffusion module to boost visual realism. This symbolic- and generation-based synergy has led to the best performance when comparing with several previous state-of-the-arts, e.g., AttnGAN, StackGAN, DALL·E 2, Stable Diffusion, in terms of multiple evalutioncriterias, e.g., CLIP Score, FID, IS and MOS.

Further, the work strongly emphasizes interactive AI by adopting human-in-the-loop with direct user control to guide and control the generation process. This user-centric perspective improves not only usability and individual personalisation effects, but it also has the potential to establish AI systems in an ethical manner by supporting transparency and user control. Its flexibility and scalability through user feedback sets the framework apart, making it relevant to use cases spanning from educational tools, over creative design systems, to assistive tools. To summary, this thesis contributes to the knowledge in text-to-image synthesis by offering a methodologically and semantically sound and visually robust framework. It demonstrates the possibility to combine semantic interpretation and advanced generation techniques to get high quality visualizations from text. The results of this work can be viewed as a stepping stone towards future work on end-to-end trainable models, real-time generative pipelines, fine-grained control over image attributes or furtherist applications in varied

10.48047/jocaaa.2024.33.7.45

languages, modalities and domains. It adds to the scientific discussion about multimodal AI and moves us toward practical, ethically motivated, and user-adaptive systems in the growing field of AI.

References

[1]. S. K. Alhabeeb and A. A. Al-Shargabi, "Text-to-Image Synthesis with Generative Models: Methods, Datasets, Performance Metrics, Challenges, and Future Direction," *IEEE Access*, vol. 11, pp. 1–15, 2023.

[2]. G. Shen*et al.*, "SG-Adapter: Enhancing Text-to-Image Generation with Scene Graph Guidance," *arXiv preprint arXiv:2405.15321*, 2024.

[3]. R. Ganz and M. Elad, "CLIPAG: Towards Generator-Free Text-to-Image Generation," in *Proc. IEEE/CVF Winter Conf. Applications of Computer Vision (WACV)*, Jan. 2024, pp. 3843–3853.

[4]. M. Kang, "Diffusion2GAN: Distilling Diffusion Models into GANs for Fast Text-to-Image Synthesis," *arXiv preprint arXiv:2405.05967*, 2024.

[5]. X. You and J. Zhang, "Text-to-Image GAN with Pretrained Representations," *arXiv preprint arXiv:2501.00116*, 2024.

[6]. S. Wu *et al.*, "Imagine That! Abstract-to-Intricate Text-to-Image Synthesis with Scene Graph Hallucination Diffusion," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[7]. A. Farshad*et al.*, "SceneGenie: Scene Graph Guided Diffusion Models for Image Synthesis," *arXiv preprint arXiv:2304.14573*, 2023.

[8]. M. Kang *et al.*, "Scaling up GANs for Text-to-Image Synthesis," *arXiv preprint arXiv:2303.05511*, 2023.

[9]. S. Liang *et al.*, "Rich Human Feedback for Text-to-Image Generation," in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, 2024, pp. 1–10.

[10]. J. Li *et al.*, "RenderDiffusion: Text Generation as Image Generation," *arXiv preprint arXiv:2306.12345*, 2023.

[11]. C. Nwoye*et al.*, "Surgical Text-to-Image Generation," *arXiv preprint arXiv:2401.09876*, 2024.

[12]. F.-A. Croitoru*et al.*, "Diffusion Models in Vision: A Survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 5, pp. 1234–1256, May 2023.