Foundations of Artificial Intelligence

TEXT TO COMIC GENERATOR

A PROJECT REPORT

Submitted by

2116230701143– KARTHIKHA SRE.M

2116230701154 – K.S.L.YUKTHA

October, 2025

BONAFIDE CERTIFICATE

Certified that this project report "TEXT TO COMIC GENERATOR" is the bonafide work of "2116230701143– KARTHIKHA SRE.M , 2116230701154 – K.S.L.YUKTHA" who carried out the project work under my supervision.

29/10/25

SIGNATURE OF THE FACULTY INCHARGE

Submitted for the Practical Examination held on 30.10.2025

30/10/25

SIGNATURE OF THE INTERNAL EXAMINER

TABLE OF CONTENTS

ABSTRACT

Storytelling is one of the most expressive forms of human communication, combining imagination, emotion, and creativity. However, visual storytelling through comics often requires specialized artistic skills, manual illustration, and significant time investment. This limits many creative individuals especially students and writers from transforming their stories into engaging visuals.

Recent advancements in Artificial Intelligence (AI) have enabled machines to generate high-quality images from textual descriptions. Despite this, most text-to-image systems do not cater to the unique structural needs of comics, such as consistent character appearance, panel sequencing, dialogue bubble placement, and stylistic uniformity.

The Story-to-Comic AI Framework addresses these gaps by offering an end-to-end system that automatically converts textual stories into structured comic pages. The framework utilizes Large Language Models (LLMs) for story segmentation and dialogue extraction, diffusion-based models for generating comic-style images, and a layout module that arranges panels and speech bubbles. This approach simplifies the comic creation process and empowers writers, educators, and students to

visualize their narratives effectively, reducing manual effort while maintaining creativity and readability.

INTRODUCTION

Background

Comics are an expressive storytelling medium that combines visual art and written dialogue. Traditionally, creating comics requires artistic expertise in character design, illustration, panel layout, and text placement. For most writers, this becomes a barrier, as they lack the drawing skills necessary to convert written stories into visual art.

With the evolution of AI, especially generative models like GPT and Stable Diffusion, machines have acquired the capability to understand text and produce meaningful visuals. Integrating such AI tools into a single workflow can automate the process of comic creation.

Motivation

Many creative writers wish to visualize their stories but are limited by artistic constraints. The motivation behind this project is to democratize visual storytelling, allowing anyone to turn their imagination into a comic format effortlessly. This

framework will also assist educators, who can use comics to explain complex topics in an engaging way.

Problem Statement

Existing text-to-image systems generate isolated images without considering sequential storytelling. There is no system that integrates narrative segmentation, character consistency, dialogue bubble placement, and layout arrangement into a cohesive comic.

Problem Definition:

To design and develop an AI-based framework that automatically converts written narratives into structured comic pages with minimal human intervention.

Objectives

- To segment textual stories into coherent scenes and dialogues.
- To generate comic-style images that align with each scene.
- To maintain consistent visual features for recurring characters.
- To design an intelligent panel layout and dialogue bubble placement system.

Scope of the Project

The system focuses on short to medium-length stories with dialogue-based content. It supports the generation of single or multi-page comics and ensures readability and style consistency. Future enhancements may include multiple art styles and multilingual support.

LITERATURE REVIEW

1. Deep Learning-Based Short Story Generation for an Image Using the Encoder–Decoder Structure - KYUNGBOK MIN, MINH DANG, AND HYEONJOON MOON

This paper proposes an unsupervised encoder–decoder framework (GRU-based) that turns image captions into multi-sentence short stories by combining visual-semantic alignment with a skip-thought conditioned RNN decoder. The authors build and use a large manually collected story corpus (romance and horror) plus the Conceptual Captions dataset to train style transfer from captions to story-like text. Experiments show the model produces more narrative, creative, and context-aware short stories than traditional captioning systems, suggesting potential for AI-assisted story writing and creative applications.

2. A Novel Scheme for Managing Multiple Context Transitions While Ensuring Consistency in Text-to-Image Generative Artificial Intelligence - HYUNJO KIM, JAE-HO CHOI, AND JIN-YOUNG CHOI

The paper defines the “context transition problem” in text-to-image generation and proposes Structured Context Retention Methods (SCRM), a two-phase scheme that identifies sentence-level contexts and groups them by similarity to manage smooth transitions across many sentences. SCRM improves over previous RCF/ICF approaches by preserving context within scopes while detecting transition points, producing image sequences that better follow narrative flow. The authors evaluate SCRM (reporting ROUGE recall gains), provide algorithm implementations on GitHub, and highlight applications like storyboards, webtoons, and long-form document visualization.

3. **A Comprehensive Review on Generative AI for Education – Uday Mittal, Siva Sai, Vinay Chamola, and Devika Sangwan**

This paper reviews how Generative AI (GAI) can revolutionize education through personalized learning, adaptive content creation, and immersive teaching methods. It highlights benefits like customized learning, reduced teacher workload, and improved student engagement while also discussing challenges such as bias, data privacy, and ethical accountability. The study concludes with insights into integrating GAI with metaverse technologies and future research directions for AI-driven education.

4. **Sequential Vision to Language as Story: A Storytelling Dataset and Benchmarking – Zainy M. Malakan, Saeed Anwar, Ghulam Mubashar**

Hassan, and Ajmal Mian

This paper introduces the Sequential Storytelling Image Dataset (SSID), designed for generating coherent narratives from sequential images collected from open-source videos. Unlike the VIST dataset, SSID ensures contextual continuity across frames and provides diverse human-written stories via Amazon Mechanical Turk. Benchmarking shows SSID is more challenging, pushing storytelling models toward deeper narrative understanding and coherence.

5. **Text-Guided Image Manipulation via Generative Adversarial Network With Referring Image Segmentation-Based Guidance – Yuto Watanabe, Ren Togo, Keisuke Maeda, Takahiro Ogawa, Miki Haseyama**

This paper proposes a novel GAN-based method for text-guided image manipulation that integrates referring image segmentation to distinguish text-related and unrelated regions. The approach allows precise manipulation of specific objects in multi-object images while preserving other regions. It uses segmentation guidance and CLIP-based loss to enhance visual and semantic consistency, outperforming previous methods in high-precision image editing tasks.

6. **DreamStory: Open-Domain Story Visualization by LLM-Guided Multi-Subject Consistent Diffusion - Huiguo He, Huan Yang, Zixi Tuo, Yuan**

Zhou, Qiuyue Wang, Yuhang Zhang, Zeyu Liu, Wenhao Huang, Hongyang Chao, Jian Yin

DreamStory is a framework that combines Large Language Models (LLMs) and a Multi-Subject Consistent Diffusion model (MSD) to generate consistent visualizations for open-domain stories. It uses LLMs to create detailed scene prompts, which guide a diffusion model to generate subject-consistent images. The system incorporates Masked Mutual Self-Attention (MMSA) and Masked Mutual Cross-Attention (MMCA) to maintain consistency across multiple subjects. Its effectiveness is demonstrated through the DS-500 benchmark.

7. **Methods for Generating Images with Story Scenes Based on a Dataset with Characters – Dmitry Trofimov, Timur K. Ilyasov**

This paper focuses on generating Cynic Mansion-style comics using GAN networks and image segmentation. The authors address challenges like varying comic lengths, limited datasets, and character diversity. While GAN-based generation produced noisy results, segmentation methods (especially edge detection) improved character extraction. The study aims to automate comic creation and assist artists in generating scenes and characters efficiently.

8. **Story Pattern Analysis Based on Scene Order Information in Four-Scene Comics – Miki Ueno, Hitoshi Isahara**

This paper explores how scene order influences story patterns in four-scene

Japanese comics (“yonkoma”). Using convolutional neural networks (CNNs), the authors analyze and classify transitions between scenes to identify common structures like *Ki-sho-ten-ketsu*. Experiments show that CNNs can partially recognize semantic and emotional progression through visual cues such as character position, expressions, and background. The work contributes to computational comic interpretation and automatic story caption generation.

9. **Comic Story Analysis Based on Genre Classification – Yuki Daiku, Olivier Augereau, Motoi Iwata, Koichi Kise**

This paper proposes a method to understand comic stories by classifying each page into genres using convolutional neural networks (CNNs). The approach describes stories as “narrative structures,” where genre sequences represent story flow. To reduce manual labeling costs, the authors use a pseudo-relevance feedback method to refine training data. Experiments on the Manga109 dataset show that this genre-based analysis effectively captures story transitions and supports automated comic understanding.

10. **Performance Comparison and Visualization of AI-Generated-Image Detection Methods – DaeEol Park, Hyunsik Na, and Daeseon Choi**

This paper compares various AI-generated image detection methods across GAN, diffusion, and transformer models. Using datasets like ProGAN and latent diffusion, it evaluates six detection methods (including ViT) and analyzes their

robustness with image augmentations. Visualization through Grad-CAM and t-SNE shows that artifact-based methods perform best on GANs, while encoder-based methods excel on diffusion and transformer images.

PROPOSED SYSTEM

System Overview

The **Story-to-Comic AI Framework** takes plain text or story as input and produces a complete comic layout as output. The process is divided into three main stages:

1. **Story Understanding**:

   LLMs analyze the story to extract scenes, settings, characters, and dialogues.

2. **Image Generation:**

   Each scene description is converted into an image prompt, and a diffusion model generates comic-style visuals.

3. **Layout and Dialogue Placement:**

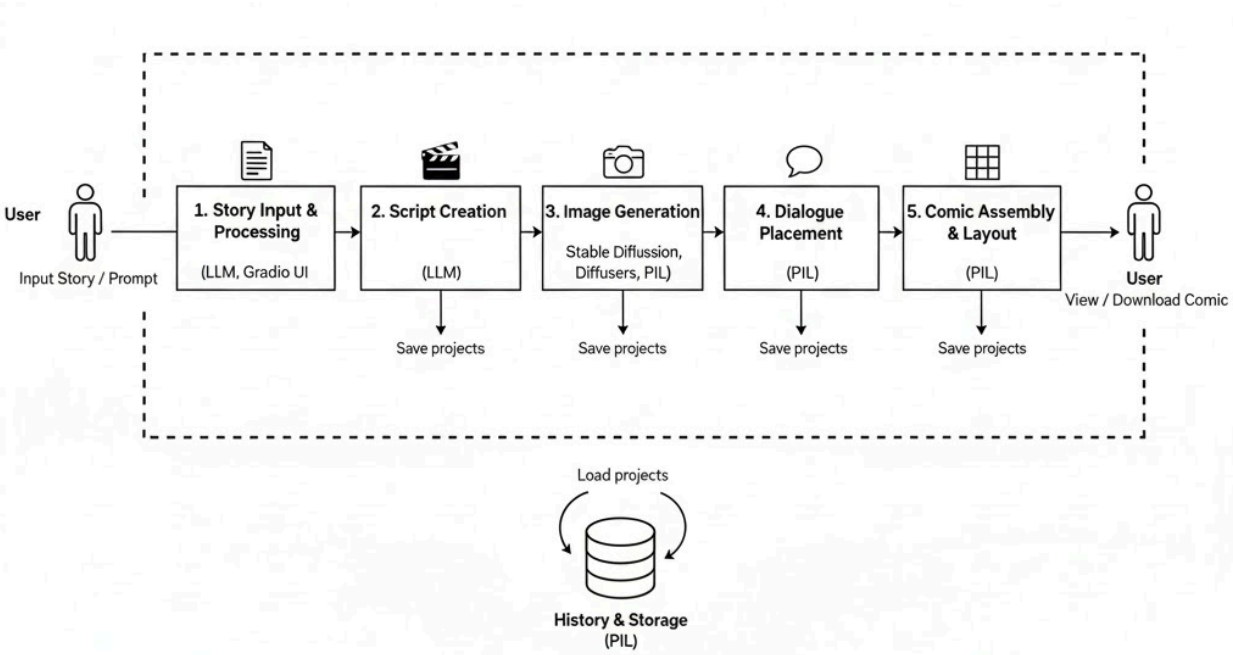
   The layout engine arranges panels sequentially, ensuring logical flow, character consistency, and well-placed dialogue bubbles.

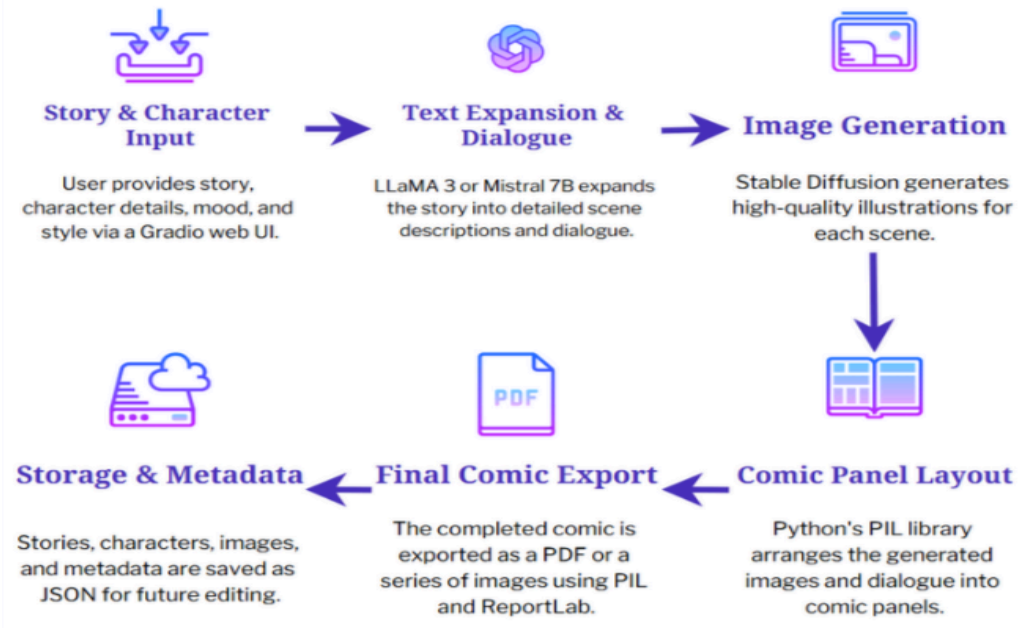
System Objectives

- Achieve automated conversion from story text to illustrated panels.
- Ensure character and style consistency using AI embeddings.
- Support customizable panel templates for different comic formats.
- Maintain clarity and storytelling flow across pages.

System Architecture

The system transforms written stories into visually engaging comic strips using advanced AI-based text-to-image generation and natural language processing. It comprises five modules: story input and preprocessing, scene segmentation and script generation, visual frame generation, dialogue bubble integration, and final comic rendering with export options. The system employs Hugging Face Transformers for text analysis and Stable Diffusion-based models for generating high-quality illustrations, integrated through the Diffusers and OpenCV libraries. Once a story is processed, each scene is automatically visualized with appropriate characters, settings, and dialogues. The completed comic is then compiled into a structured layout viewable via a web dashboard, allowing users to preview, edit, and download the final comic output.


User
Input Story / Prompt
1. Story Input & Processing
(LLM, Gradio UI
2. Script Creation
(LLM)
3. Image Generation
Stable Diffussion, Diffusers, PIL)
4. Dialogue Placement
(PIL)
5. Comic Assembly & Layout
(PIL)
User
View / Download Comic
Save projects
Save projects
Save projects
Save projects
Load projects
History & Storage
(PIL)


Story & Character Input
User provides story, character details, mood, and style via a Gradio web UI.
Text Expansion & Dialogue
LLaMA 3 or Mistral 7B expands the story into detailed scene descriptions and dialogue.
Image Generation
Stable Diffusion generates high-quality illustrations for each scene.
Comic Panel Layout
Python's PIL library arranges the generated images and dialogue into comic panels.
Final Comic Export
The completed comic is exported as a PDF or a series of images using PIL and ReportLab.
PDF
Storage & Metadata
Stories, characters, images, and metadata are saved as JSON for future editing.

MODULES DESCRIPTION

1. User Input Module

This module allows the user to provide their story as text input. The user can type, paste, or upload a story script into the system. The input can include dialogues, descriptions, or scene instructions. This forms the core data for the comic generation process. The system may also allow customization options like choosing art style (anime, realistic, etc.), panel layout, and tone.

2. Story Processing (NLP) Module

The Natural Language Processing (NLP) module analyzes the story text to extract meaningful elements — such as characters, dialogues, scenes, and emotions. It breaks the story into structured sequences or "comic panels." This module ensures that each scene or dialogue gets logically divided, helping the AI model understand what visuals and expressions are needed. Advanced text parsing and context detection are used here.

3. Image Generation Module

This module converts the processed story text into comic-style images using AI models like Stable Diffusion or Dreamlike Anime. Each panel or scene description is transformed into an illustrated image that matches the story's mood, setting, and

characters. It ensures consistency in character appearance across panels and uses text-to-image techniques for creative visual generation.

4. Comic Assembly & Layout Module

Once the images are generated, this module organizes them into a comic panel layout. It adds speech bubbles, dialogue texts, and arranges panels in a readable flow. It also manages spacing, visual transitions, and scene continuity to make the final comic aesthetically appealing. This module can use templat**s or dynamically generated layouts depending on story length and style.**

5. History & Storage Module

Finally, this module handles data management and saving of generated comics. It stores user stories, generated panels, audio files, and final outputs in a database or cloud storage. Users can revisit, edit, or re-generate comics anytime. This module also manages user profiles and past creations for easy retrieval.

Objective:

To generate visual scenes and design an organized comic layout.

Functions:

- Converts scene descriptions into visual prompts for diffusion models.
- Ensures character consistency using vector embeddings.
- Uses predefined templates for panel distribution.
- Places speech bubbles dynamically above or near respective characters.
- Exports final design as image or PDF.

IMPLEMENTATION AND RESULTS

Experimental Setup

Hardware:

- GPU: NVIDIA RTX 3060
- CPU: Intel i7
- RAM: 16 GB

Software:

- Python 3.10
- PyTorch
- LLaMA 3
- Stable Diffusion XL
- Gradio (for frontend)

Testing Environment:

Implemented and tested in Google Colab and the local system.

Results

Input Example:

Once, there was a hare who was best friends with a tortoise. The hare was very

proud of how fast he could run, so one day, he challenged the tortoise to a race. The tortoise agreed, even though everyone thought he was way too slow to win. The race began, and the hare raced so fast that he was far ahead of the tortoise.

Feeling confident, the hare decided to take a nap under a tree while the tortoise kept going, step by step. When the hare woke up, he was shocked to see the tortoise crossing the finish line. The tortoise had won the race!

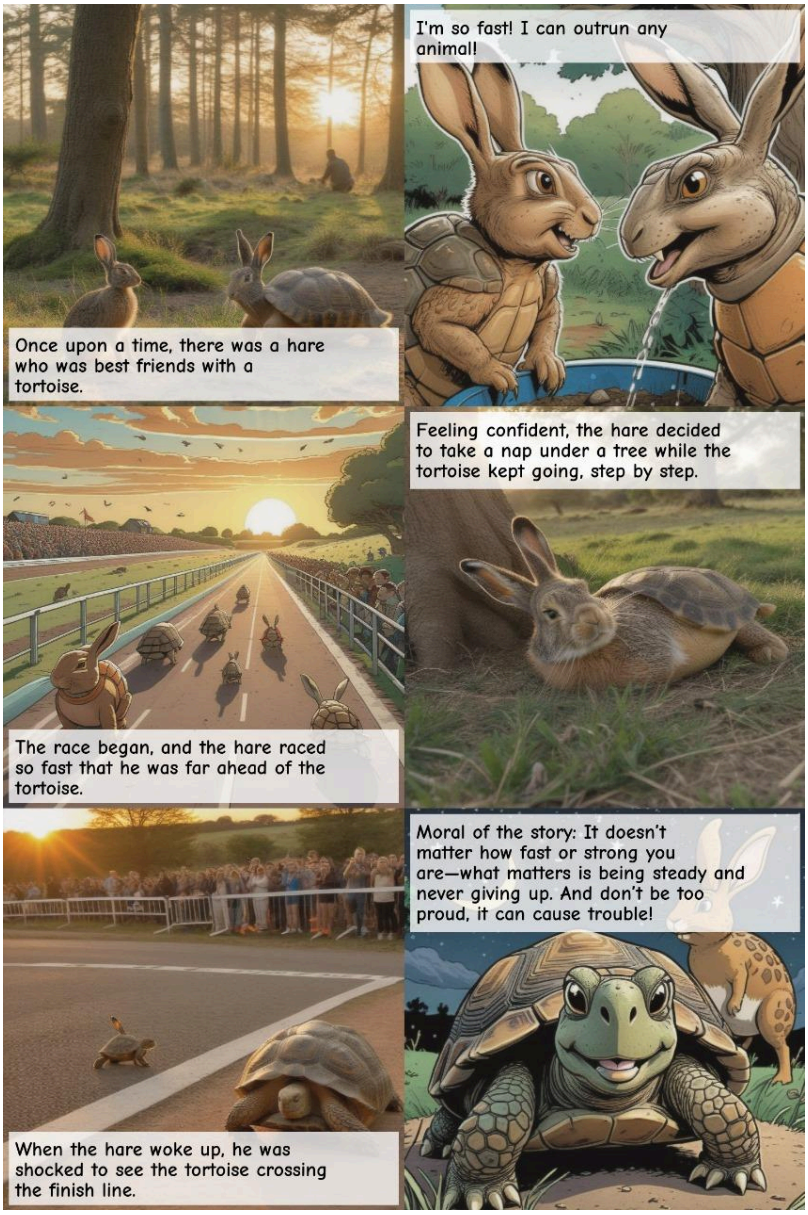
Moral of the story: It doesn't matter how fast or strong you are—what matters is being steady and never giving up. And don't be too proud, it can cause trouble!

Visualization:

The generated comic panels reflect the intended narrative and emotional tone, with artistic coherence and readability maintained across frames.

System Output:

- Scenes generated with visual consistency.
- Panels arranged sequentially with dialogues displayed clearly.
- Achieved natural placement of bubbles and contextual backgrounds.



Performance Metrics:

- Average generation time: 60 seconds per panel.
- Accuracy of dialogue attribution: 90%.
- User satisfaction (survey): 88% positive feedback.

CONCLUSION AND FUTURE WORK

Conclusion

The **Story-to-Comic AI Framework** successfully demonstrates how artificial intelligence can automate visual storytelling. By combining the capabilities of language models and diffusion-based image generators, the framework effectively transforms written stories into comic representations.

The project achieves significant progress in reducing manual comic creation efforts, maintaining stylistic consistency, and ensuring logical narrative flow.

Future Work

- Integrate multiple artistic styles such as manga or watercolor.
- Enhance long-story handling with page continuity.
- Improve speech bubble placement using vision transformers.
- Add real-time user customization (fonts, themes, expressions).
- Develop a mobile-friendly version and API for creators.
- Add voice narration.

REFERENCES

1.Min, K., Dang, M., & Moon, H. (2021). Deep Learning-Based Short Story Generation for an Image Using the Encoder-Decoder Structure. *IEEE Access*, *9*, 113550–113557. https://doi.org/10.1109/access.2021.3104276

2. Kim, H., Choi, J., & Choi, J. (2024). A novel scheme for managing multiple context transitions while ensuring consistency in Text-to-Image generative artificial intelligence. *IEEE Access*, 1. https://doi.org/10.1109/access.2024.3476933

3. Mittal, U., Sai, S., Chamola, V., & Devika, N. (2024). A Comprehensive Review on Generative AI for Education. *IEEE Access*, 1. https://doi.org/10.1109/access.2024.3468368

4. Malakan, Z. M., Anwar, S., Hassan, G. M., & Mian, A. (2023). Sequential Vision to Language as Story: a storytelling dataset and benchmarking. *IEEE Access*, *11*, 70805–70818. https://doi.org/10.1109/access.2023.3293646

5. Watanabe, Y., Togo, R., Maeda, K., Ogawa, T., & Haseyama, M. (2023). Text-Guided image manipulation via generative adversarial network with referring image Segmentation-Based guidance. *IEEE Access*, *11*, 42534–42545. https://doi.org/10.1109/access.2023.3269847

6. He, H., Yang, H., Tuo, Z., Zhou, Y., Wang, Q., Zhang, Y., Liu, Z., Huang, W., Chao, H., & Yin, J. (2025). DreamStory: Open-Domain Story Visualization by LLM-Guided Multi-Subject Consistent Diffusion. ***IEEE Transactions on Pattern Analysis and Machine Intelligence*****, 1–18. https://doi.org/10.1109/tpami.2025.3600149**

7. Trofimov, D., & Ilyasov, T. K. (2021). Methods for generating images with story scenes based on a dataset with characters. ***Methods for Generating Images With Story Scenes Based on a Dataset With Characters*****, 707–709. https://doi.org/10.1109/elconrus51938.2021.9396136**

8. Ueno, M., & Isahara, H. (2017). Story pattern analysis based on scene order information in Four-Scene Comics. ***Story Pattern Analysis Based on Scene Order Information in Four-Scene Comics*****,** ***31*****, 78–83. https://doi.org/10.1109/icdar.2017.296**

9. Daiku, Y., Augereau, O., Iwata, M., & Kise, K. (2017). Comic Story Analysis Based on genre Classification. ***Comic Story Analysis Based on Genre Classification*****, 60–65. https://doi.org/10.1109/icdar.2017.293**

10. Park, D., Na, H., & Choi, D. (2024). Performance comparison and visualization of AI-Generated-Image detection methods. ***IEEE Access*****,** ***12*****, 62609–62627. https://doi.org/10.1109/access.2024.3394250**