International Journal of Engineering, Science and Information Technology
Volume 5 No. 2 (2025) pp. 599-608
ISSN 2775-2674 (online)
Website: http://ijesty.org/index.php/ijesty
DOI: https://doi.org/10.52088/ijesty.v5i2.1534
Research Paper, Short Communication, Review, Technical Paper

AI-Assisted Animation Storyboard Design and Automated Storyboard Generation

Han Ou

Department of Performance, Film, Animation, Sejong University, Seoul, Korea

**Corresponding author Email: ouhan@zqu.edu.cn*

The manuscript was received on 25 November 2024, revised on 11 January 2025, and accepted on 10 April 2025, date of publication 23 May 2025

Abstract

This paper develops an Artificial Intelligence assisted animation storyboard design framework that uses Stable Diffusion 1.5 (SD-1.5) together with Visual Geometry Group 1-Convolutional Neural Network (VGG-1CNN) and Generative Pre-trained Transformer 3.5 (GPT-3.5) to produce automated game character images and narrative-focused storyboards. The proposed system utilizes combined text and sketch prompts for generating storyboard frames which preserve visual coherence together with stylistic continuity. The three main elements that power improved image generation through advanced diffusion control techniques include Contrastive Language-Image Pretraining (CLIP) neural networks and VGG-1CNN and Variational Autoencoder (VAE). The sequence starts by translating textual descriptions into numerical latent space codes using a neural network before the computer generates images based on these guidelines. The basic sketch receives edge detection through Canny edge maps to give better results in image refinement. By applying the VGGNet architecture to vector representations of generated images the system improves visual precision together with prompt compliance. The image quality receives additional enhancement through an iterative scheduler-based removal of noise which refines vector representations during multiple successive stages. The deployment of GPT-3.5 gives the system ability to create written narratives suited for each story frame while preserving narrational continuity. A decoder-based upscaling technique applies to the final output to generate high-resolution visually appealing storyboard frames that properly highlight the visual elements alongside textual content. The automated solution established through this model delivers an efficient pre-production animation pipeline automation that minimizes work efforts and conserves artistic and narrative quality.

Keywords: *VGG-1CNN Neural Network, Generative Pre-Trained, Transformer, Stable Diffusion, Automated Image Generation.*

1. Introduction

Development teams use storyboarding as their principal gaming work method to plan animated scenes before bringing the entire project to production. The development team uses storyboarding to create a reference design that shows developers, designers and animators how gameplay scenes and cutscenes should move alongside character emotional arcs [1]. Through visual planning storyboarding backs up the steady execution of camera positions as well as scene arrangement and actor facial expressions which makes it essential for pre-production [2]. Creating traditional storyboards through handwork proves demanding for both time and budget because it requires advanced artistic planning alongside multiple drafting sessions for improved storytelling [3]. The repeated need for storyboard modifications stands out as a major downfall because it impacts game development particularly hard because of its reliance on interactive gameplay features [4]. The need for sophisticated video games and cinematographic quality in modern gaming has created the necessity for AI-based automation tools which boost both efficiencies, streamline workflows and enable developers to make storyboards at maximum efficiency [5].

Despite existing digital storyboarding tools present in the market today storyboards strictly use human-initiated content which limits their adaptability toward interactive narratives [6]. Absolute narratives in film and television sector generate storyboards specifically for production purposes while their sequences must follow a fixed timeline. Video games demand a dynamic non-linear management of scenes which alters their structure based on what decisions players make and actions they take [7]. Storyboard artists experience substantial obstacles because they need to plan out different scene possibilities alongside camera perspectives together with animated character sequences [8]. Game development through iteration requires continual modifications of character placements and dialogue scripts as well as scene positions that lead to continuous updates of storyboards. The labor-intensive procedure makes this process expensive and time consuming and particularly burdensome to small game studios who lack full-time storyboard artists [9]. Advanced video games require an innovative smart and fast game narrative solution which aligns with shifting gaming story methods [10].

The application of conventional storyboards faces two major delays which stem from time constraints and staffing inconsistencies as well as resource scarcity. Creative teams consistently fail to operate within their stated deadlines thus creating difficulties for them to maintain visual consistency during repeated redesign processes [11]. AI-based solutions operate as a promising system to execute repetitive work cycles and guarantee consistent imagery application while shortening manufacturing cycles [12]. Artificial intelligence and its associated technologies allow machines to accept text descriptors and character sketches and motion patterns then convert them into organized

storyboards through sophisticated processing approaches [13]. The ability of AI-powered models to animate characters and arrange scenes through cinematic elements enables automatic storyboard platforms to forecast animation motion as they alter angles for better game design [14]. Real-time development collaboration happens between writers' artists and developers through AI-powered technology to modify storyboard contents when developers make game direction changes [15].

A deep learning-based storyboard automation system that this research investigates serves to enhance game development pre-production with model development by using AI tools. The system uses a combination of machine learning technology and procedural content generation with deep learning image recognition to create automation for storyboard function creation [16]. AI-based modeling systems provide designers both an easy way to visualize game sequences and reduced manual efforts in modelling [17]. VGGNet functions as a deep CNN network to extract features while recognizing scenes and maintaining characters through its generation of visually pleasing storyboard sequences which possess structured layout. Deep learning functionality in this system presents two capabilities [18]: automated panel arrangement generation and automatic camera angle recommendations with blocking transitions during storyboarding to decrease operational time requirements [19]. The purpose of VGGNet integration focuses on operational speed improvements along with financial savings and real-time interactive game studio productions without compromising artistic control [20].

2. Literature Review

Modern artificial intelligence processing together with deep learning methods and computational methods allow automated creation of gaming scenes alongside character consistency production for both games and animations which leads to visual narrative development. Various neural networks applied with generative models and procedural algorithms build an effective process which needs reduced human involvement and provides advanced creative solutions. The methods of using artificial intelligence can automatically create storyboards and normalize visual appearance while arranging narratives while learning from player input and performing scene identification through deep learning methods which make creative processes more rapid. Through these methods the manufacturer gets faster production results while establishing real-time collaboration and better narrative control and enhanced creative accuracy. The current motivation of student researchers involves focusing on preserving emotional depth in their stories through techniques which establish narrative consistency and maintain story coherence. The benefits and drawbacks of these methods appear in Table 1.

Zhaohui Liang et al., [21] proposed an AI system for UX designers which assisted them in designing interactive storyboards. The research introduced an AI system using LLMs together with text-to-image diffusion models that converted verbal descriptions into storyboard visual frames. The method succeeded in improving workflow efficiency along with speedup of UX designer ideation activities. The study discussed acknowledged three critical concerns about the image generation process which include unpredictable outcomes from the system along with character rendering instability between frames and employment requirements for obtaining professional-quality outputs.

Hanseob Kim et al., [22] created a system that used NLP techniques to extract character names and dialogue lines and movements of characters from screenplays. Screened information backed the production of 2D and 3D storyboards through a predetermined filmmaking system. The method enabled both pre-visualization automation as well as production period reduction. The technical dependence on scripted input created flexibility problems for the pipeline because it struggled with unknown or unstructured screenplay content that required manual human intervention during complex situations.

Callie Y. Kim et al [23] investigated an approach that merges reinforcement learning with interactive narrative generation for different generational narrative co-creation capabilities. Users could create interactive stories via AI-based suggestions since this method simultaneously raised their involvement level and their creative potential. AI tools for storytelling produced robotic texts yet researchers concluded that these tools worked properly for ordered stories.

Weijia Wu et al., [24] developed an agent system that used Chain-of-Thought (CoT) planning technique for AI-based movie production. Several AI agents collaborated to develop scenes and their stories and animated them while maintaining coherent transitions between scenes and narrative compatibility. This method revealed capabilities to automatically streamline creative processes that decreased total time required for creating movies. While the research identified creative issues which limited the AI to producing repetitive plotlines while producing animated results with insufficient human-like artistic quality.

Lassheikki et al., [25] conducted research about AI storytelling programs that power storytelling functions in games. The authors researched how PCG technology alongside AI-based narrative engines affects the work of game writers along with narrative designers. This research identified that AI helps dynamic storytelling through adaptability, yet it highlighted the human storytelling richness loss when AI takes charge as well as the quest to preserve narrative cohesion in generated material.

Table 1. Problem formulation of conventional techniques

Author(s)	Techniques Involved	Advantages	Disadvantages
[21]	LLMs, Text-to-Image Diffusion	Reduces manual effort, fast ideation	Inconsistent images, lacks character continuity
[22]	NLP, Cinematic Rule-Based Storyboarding	Automates pre-visualization, saves time	Limited flexibility, needs manual fixes
[23]	Reinforcement Learning, AI Storytelling	Enhances engagement, structured storytelling	Lacks emotional depth, mechanical output
[24]	Multi-Agent CoT Planning	Automates scripting, storyboarding	Generic storylines, lacks stylistic nuance
[25]	PCG, AI Narrative Engines	Dynamic, adaptive storytelling	Risk of losing human depth, coherence issues

The implementation of AI for storyboard production encounters multiple issues related to unstable character names, uncontrollable visual content and disordered storylines and limited creative styles. The joint operation of LLMs with text-to-image diffusion models enables fast idea development and yet the output contains unstable elements. Human maintenance is necessary for the system to work effectively although it has problems managing unstructured screenplay materials. Reinforcement learning allows their AI storytelling optimization to

generate unemotional computer-generated stories. Multi-Agent CoT planning excels at script and storyboard creation followed by normal narrative generation through automated standard writing while PCG-based systems cannot maintain human storytelling attributes. Our system applies VGGNet in image processing because it serves as an effective deep CNN for extracting image features during operations. The VGGNet deep CNN produces superior animated sequences by enabling continuous story flow and real-looking character models that improve the development of visual content. Our focus is on implementing VGGNet in AI pipelines to improve automated storyboarding because its accurate processing technology upholds visual consistency for creating superior results.

3. Methods

The research objective revolves around automating storyboard production while recognizing the creative distinctions between humans and machines. The application Sketch Board merges feature of GPT 3.5 from Open AI for game design along with character development functions with stable diffusion 1.5 alongside a control net for image creation.

3.1. Game characters-based Storyboard designing

A new architectural system integrates stable diffusion 1.5 with the VGG-1CNN architecture and GPT-3.5 architecture to develop game character pictures from sketches and storyboards. The architecture integrates their architecture and diffusion control approaches to produce a deep learning picture generation system. The method combines text information with sketch inputs to generate visual consistency for establishing the frame of a storyboard. The text prompt functions as the sole input for GPT-3.5 which generates textual information about the image produced by the suggested model [26].

The model controls the picture-matching process because of its built-in capability to direct the diffusion design. Character art sketches go through refinement and retraining as part of the initial step of the sketch board procedure which enables the model to convert basic drawings into comprehensive story frames. The second phase employed GPT to develop narratives because it ensures extensive understanding of contexts. GPT defines the narrative solution during the progression. The storytelling element finalizes during this phase when consistent narrative texts are generated for storyboard frames before their smooth integration into one extensive plot [27].

The proposed model consists of three major components including contrastive language along with image pretraining neural network and VGG-1CNN with VAE which have been improved and developed on a drawing board. The network in combination with the text prompt sends data to the SD-1.5 model to receive additional conditioning. The initial operation within sketchboard requires the neural network text encoder parameter to tokenize and encode the text input prompt through an encoding process. Numerical latent specifications from textual information guide picture creation during the transformation process. The system processes the basic sketch as input via edge map identification of cany which enables it to gather picture edges for determining the boundaries of the final output. A VGGNet architecture sends the initial vector specification for the image following an encoded text prompt. The word representation serves as a guide to enhance the vector representation of pictures through this framework [28]. Targeting both improved quality and better alignment to prompt specifications constitutes the main purpose of this optimization step. Application of the scheduler approach serves to enhance both image quality and prompt adherence through noise removal processes. Sequential management of vector representations leads to better pictures that reflect the intended visual behavior mentioned in the text prompt. The repetitions occur T times. The middle process of this architecture controls SD-1.5 architecture through task-specific parameters received from clever edge map inputs. The scenario data transfer to the designated stage is possible through the correspondence between design encoder layers and VGG-1CNN stages. The decoder improves the image resolution after eliminating noise to produce an output. The image vector definition becomes visually appealing after this phase extends its stated scale to generate the most appropriate image for the received text prompt [29]. The proposed architecture is presented in figure 1.

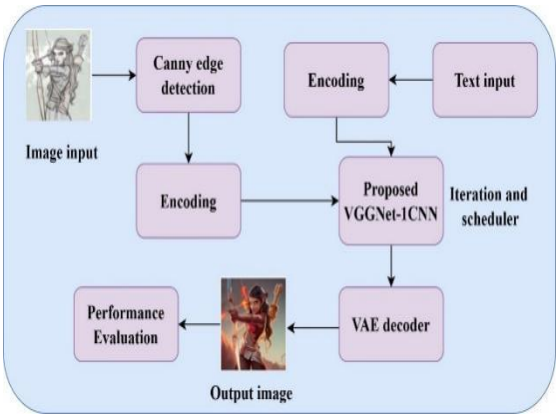

Fig 1. Proposed System Model

TMPro, the initial component of GPT-3.4 sketch board narrative advancement, requires an entry prompt that provides summary text about key story elements. The GPT tokenizer splits the text before converting its small parts into text embeddings using transformer embeddings. The input embeddings benefit from locational encoding to maintain a link between the timeline properties of the story. The encoding methods supply positional details about token placement in the sequence to allow the model to maintain chronological precision for the story events. Locational encoding enables the model to produce ordered outputs which maintain a steady logical stream [30].

The data transmission occurs through feed forward neural networks together with a 12-layer self-attention strategy. The input tokens require conversion through a combination of architectural merge in each layer. The training process instructs these architectures to collect essential definitions for producing meaningful text that reaches a high depth level. The attention mechanism steers tale generation by selecting various portions of the provided text input for processing. The design orders relevant textual elements through unique token

weight criteria to generate necessary and logical output text. The last stage of the decoder presents the manufactured output. The algorithm selects which sequence of words is most probabilistic for the next output through learned patterns. The iterative process generates a unified narrative which fits both context and established structure of the story.

3.2. VGG-1CNN

The paper develops VGG-ICNN as a lightweight CNN architecture that serves storyboard creation purposes. The main objective seeks to minimize model dimensions without compromising the storyboard design performance on diverse databases. The designed CNN architecture incorporates VGG blocks together with Inception modules as its fundamental structure [31]. The VGG blocks from the initial four layers of the VGG-ICNN because they contain multiple convolution layers alongside max pooling layers. A combination of GoogLeNet and Inception blocks functions within the convolutional block system of the last three layers. Each of the first two convolutional layers contains 64 filters and the remaining two convolutional layers contain 128 filters in this structure. Every filter possesses a size of three by three. The variable initialization of these four convolution layers depends on pre-trained VGG16 ImageNet weights. The three conception modules contain 1024 filters of different dimensions [32]. Random parameter initialization functions that apply to inception blocks. ReLU activation serves as a function in all the convolutional layers. The global average pooling layer serves as an alternative to flattening because it reduces the number of trainable variables for which reason it was chosen for use at the inception block output. The classification work requires a fully linked layer with SoftMax activation to finish the operation. The architecture stands out as lightweight because its many fewer parameters compare to a standard deep CNN structure [33].

3.3. Workflow of the proposed model

The process for defining the proposed model follows three sequential steps which include model computation and model learning and improvements and model configuration. Model configuration begins at the stage where the pre-learned suggested model has been established. The built database in League of Legends game styles allows the model to enhance its performance through using pre-learned patterns. Throughout model learning operations the computation of perceptual loss serves as the goal function for model calculations in each iteration. A team selects model databases which contain the required elements for creating LOL gaming style. All game-relevant aspects in League of Legends include characters combined with artistic design and narrative elements and visual aesthetic standards. Specific user-selected databases allow the recommended architecture to generate storyboard frames which match the visual concepts of League of Legends gameplay. The database provides training capability to the architecture so it can develop accurate specifications of the targeted game features alongside chosen aesthetics by selecting specific style and subject matters. A designated caption accompanies every image right before submission to the suggested model. When using the bootstrapping language image pre-training method for creating artificial picture descriptions the method depends on visual language relationships to produce new captions for images to accelerate the development process. The method automatically produces context descriptions along with which it matches suitable photographs rather than having human operators create captions. The efficient combination of storyboard frames into LoL style sequences through project support enhances the learning effectiveness [34].

The sketch board operates a technical back-end system across a high-performance computer infrastructure. This system provides organizations with the capability to manage resources according to demand needs. SketchBoard utilizes Cloudinary as its cloud-based picture storage solution through which it performs media optimization and editing and maintains database security. The integration of Sanic API with Cloudinary enables smooth picture data management during this development. The smooth connection of image tasks enables efficient management and provides both quick speeds and superior user experience results. The database storage relies on MongoDB Atlas which represents a cloud-managed service. Users obtain reliable and scalable NoSQL database capabilities through MongoDB Atlas as their solution. All data pertaining to storyboard creation follows the direction of sketches and gets stored by the cloud platform. GitHub repositories maintain back-end and front-end script codes which provide effective version control solutions. Front-end code deployment occurs automatically through GitHub communication with Vercel. Vercel automatically deploys the application whenever developers make changes to the front-end code which exists in GitHub repositories. In applications that use Bananadev users can directly retrieve their backend code from GitHub repositories. The SketchBoard development process musts run smoothly because the simple interface merges front-end and back-end functions [35].

Text and drawing entries through the online application enable users to generate images in the data flow. Users convey their request contents to the Sanic API after submitting their information. The request payload transmitted by the Sanic API enables the SD-1.5+CN model to generate appropriate pictures from the input text and sketch information. After receiving the created picture from SD-1.5+CN model the Sanic API functions as the response endpoint. As an endpoint the Sanic API retrieves received pictures from Cloudinary. Cloudinary stores the picture after which it provides a URL that Sanic API retrieves to incorporate into the web application response.

4. Result and Discussion

The AI-assisted tool generated superior results for animation storyboard development in terms of image quality together with processing speed and narrative predictors and stylistic distribution. Stable Diffusion 1.5 (SD-1.5) brings VGG-1CNN to enhance image fidelity along with Canny edge detection for maintaining character visual clarity. The application of schedulers in removing noise enhanced image clarity by decreasing artifacts and distortion during multiple iterations. Efforts were made to measure how GPT-3.5 generated text for storyboards and verified both its semantic accuracy and ability to create consistent story-like descriptions that matched the graphic frames. The framework used image refinement methods with diffusion controls and text-to-image mapping techniques that created a precise connection between textual descriptions and visual outputs thus becoming a suitable system for automated storyboard creation in animation and game designs. The figure 2 demonstrates the visual assessment of this proposed method.

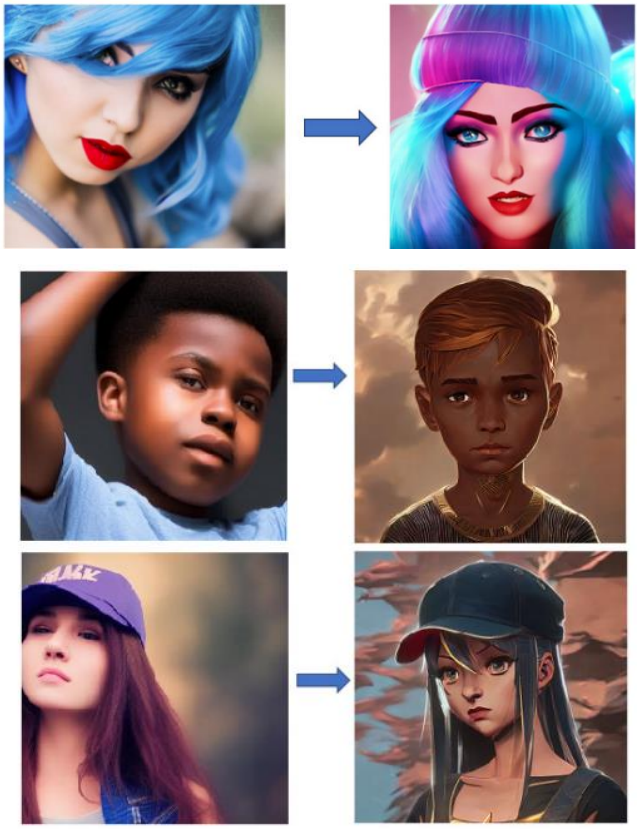

Fig 2. Visual comparison of the generation image from the optimal tuned model and the baseline model

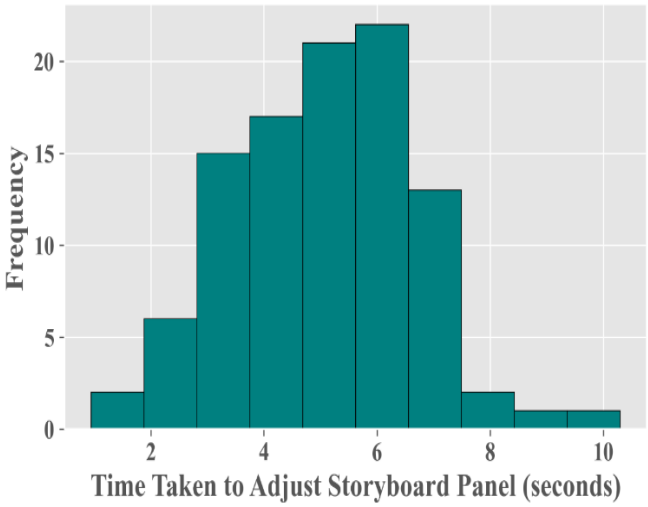

Fig 3. Frequency

A frequency distribution of storyboard panel adjustment duration and its corresponding occurrences (in seconds) appears in figure 3. The pace of modification and number of occurrences are shown along the y and x axes respectively. The timeframe that users utilize to adjust storyboard panels occurs most frequently between 5 and 6 seconds and reaches more than 20 observed instances. Most storyboard panel readjustments occur within the time periods of 4 to 5 seconds and 6 to 7 seconds according to the collected data. The occurrences of adjustment times between 1 to 2 seconds and 8 to 10 seconds remain below 5 counts throughout the period. Most control adjustments occur within the period of 4 to 7 seconds according to these observations about typical panel adjustment timelines. The data distribution demonstrates the form of a bell shape which demonstrates most users tend to work within an optimal time window with fewer rare occurrences at the ends. The distribution pattern shows that AI-supported storyboard tools create the best conditions for panel adjustments when sessions last between 4 and 7 seconds thus reducing the necessity for numerous changes.

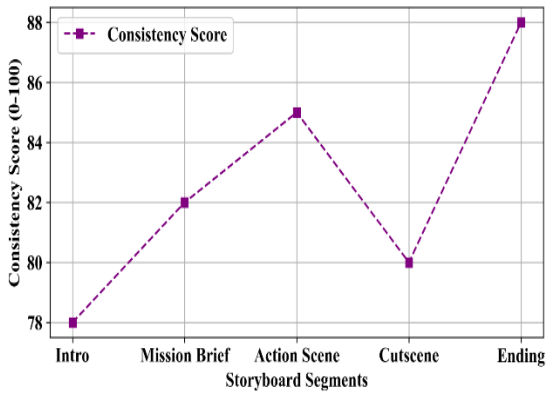

Fig 4. Consistency Score

The storyboard segments are evaluated via the Consistency Score which ranges from zero to one hundred points in figure 4. The Intro segment displays the minimal level of success at 78 due to varying coherence at its start. Structured sequences increase the score up to 82 points during Mission Brief and 85 points during Action Scene. The Consistency Score showed a decrease in the Cutscene (80) segment because narrative transitions possibly occurred during that time. The score in the Ending section stands as the highest at 88 due to excellent alignment with the narrative framework. The analysis shows that AI model performance remains stable, but Cutscene periods need additional work for enhancement. The AI-supported framework achieves seamless flow particularly in action and concluding parts of the sequence. Additional optimization work needs to be done so the storyboard phases maintain uniform consistency.

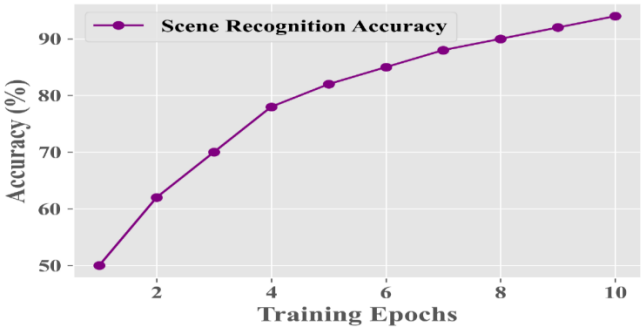

Fig 5. Accuracy

The accuracy rate for scene recognition (%) is depicted in figure 5 throughout training epochs. At the first epoch the model only reaches a 50% accuracy level which demonstrates its poor initial capability. The model demonstrates substantial learning based on its performance which enhances from 70% accuracy at epoch 3 to 80% accuracy by epoch 5. The accuracy rise becomes slower after epoch 5 until it reaches 85% at epoch 7 and 90% at epoch 9. At epoch 10 the model demonstrates strong generalization ability with over 91% accuracy in its performance. The data indicates that most learning takes place within the initial training periods followed by gradual reduction of new knowledge acquisition in subsequent training phases. The training procedure proves successful according to the research findings while additional optimization might be attained by additional refinement methods.

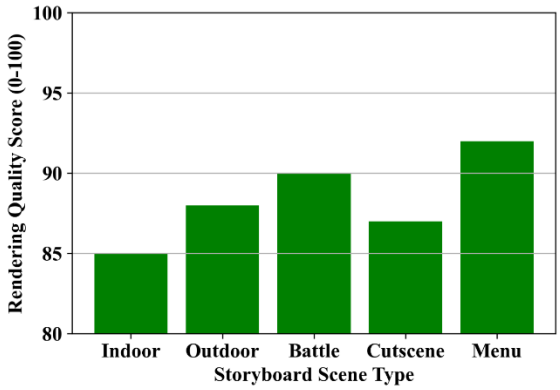

Fig 6. Rendering Quality score

Different storyboard scenes receive quality ratings from 0 to 100 in the figure 6. At 85 points the Indoor scene demonstrates moderate quality of rendering. The Outdoor scene renders with slightly higher performance according to its score of 88. The Battle scene produces high-quality rendering according to its score of 90. The Cutscene obtains a lower scoring rate of 87 points because of rendering inconsistency issues. The Menu scene reaches an optimized rendering performance with its quality score measuring 92 points. Battle and menu scenes excel in rendering quality opposite to indoor and cutscenes which present lower standards of quality. The figure 7 shows

that different model types including Proposed and ResNet and CNN and RNN require different processing times of seconds to complete operations. The proposed model delivers processing at 0.8 seconds which stands as the best time currently available for processing-related tasks. ResNet performs second at 1.2 seconds, but CNN reaches execution at 1.5 seconds. The RNN model requires the longest processing time which amounts to 2.0 seconds. The proposed model proves to provide exceptional speed optimization compared to typical deep learning network structures. The results demonstrate that the proposed model provides suitable performance for real-time applications because of its time-efficient processing capabilities.

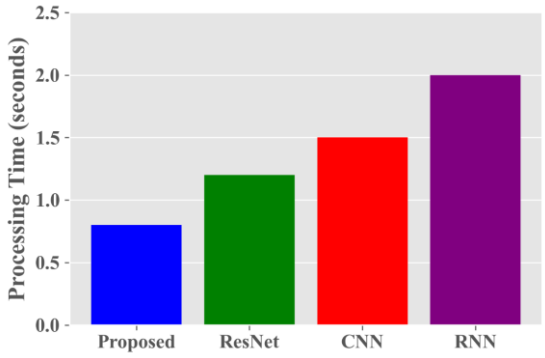

Fig 7. Processing time

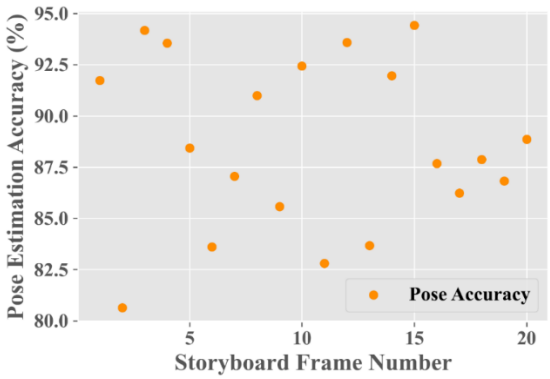

Fig 8. Post estimation accuracy

The Figure 8 displays Pose Estimation Accuracy measurements for different Storyboard Frame Numbers. Frame numbers from 1 to 20 on the x-axis demonstrate the storyboard sequence with accuracy percentages shown on the y-axis. A different orange marker appears for each storyboard frame accuracy result. The accuracy distribution displays periodic changes throughout the frames and mostly rests between 80% to 95% values. The accuracy levels between different storyboard frames show both high-performing results and mild variability between individual frames. Pose estimation performance depends on frame content but maintains consistently high accuracy levels according to the overall pattern. The data visualization shows "Pose Accuracy" as its label for the shown data points.

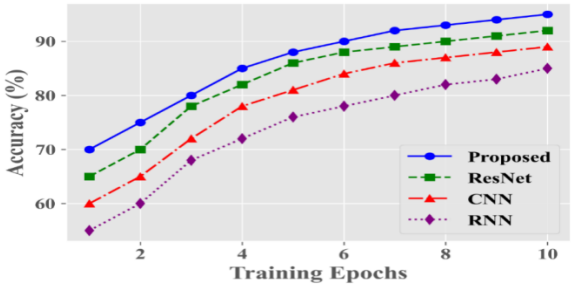

Fig 9. Accuracy measure with training epochs

The accuracy evaluation during Training Epochs (1-10) is displayed in Figure 9. The Proposed model reaches the highest accuracy point that exceeds 90% during Epoch 6. ResNet runs close to the proposed model even though it demonstrates slightly inferior results. The CNN model demonstrates continuous development although it fails to surpass the 90% accuracy threshold. RNN demonstrates the slowest accuracy development while reaching the minimum level of accuracy between the models. The model proposed delivers superior performance than all alternative models. The ResNet model demonstrates equivalent performance to its competitors but CNN and RNN present slower results. The plot legend clearly reveals model performance distinctions. The Loss Trends are shown through Figure 10 that displays Epochs 1 through 10 of Training. The Proposed model demonstrates the fastest loss reduction through which it falls below 0.2 starting at epoch 10. Loss values in ResNet decrease at the same rate as other improvement patterns yet remain at slightly elevated levels. The CNN model demonstrates continual improvement although its loss stays above that of ResNet. Under this evaluation the RNN model experienced the longest delay showing steady increases in loss values until it reached the maximum value. The proposed model shows

superior performance when it comes to reaching convergence. The learning rates of ResNet match those of CNN and RNN but with lower value loss rates. The plot legend establishes simple ways to differentiate between the examined models.

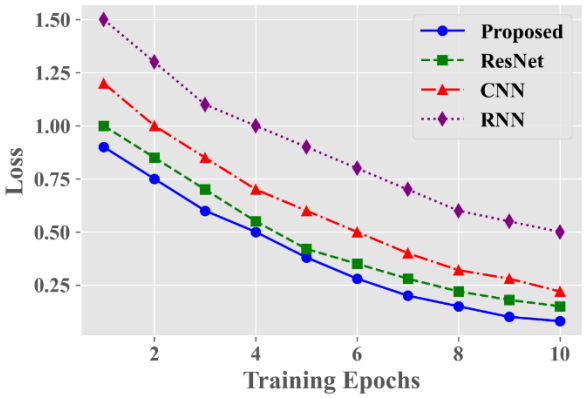

Fig 10. Loss measure with training epochs

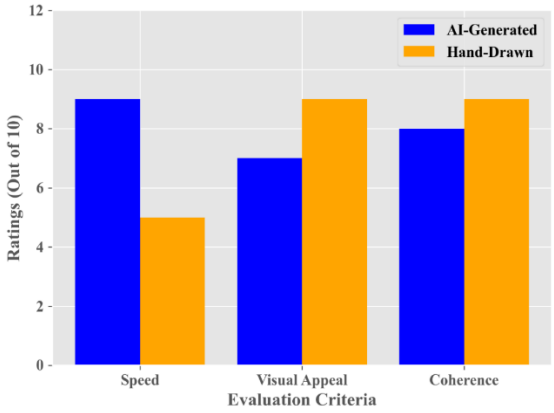

Fig 11. Ratings

The figure 11 shows differential ratings of AI-Generated and Hand-Drawn images regarding Speed, Visual Appeal and Coherence according to the evaluation parameters. The evaluation ratings span from zero to ten points which are displayed along the y-axis and the evaluation criteria are shown on the x-axis. The AI-Generated images achieve the fastest response time during production thus scoring higher in Speed than Hand-Drawn images which take longer to create. People find Hand-Drawn pictures superior to AI-generated artwork in terms of visual appeal because viewers prefer drawings created by humans. Analyzing Coherence metrics shows the Hand-Drawn images have minor superiority over AI-Generated images thus demonstrating improved logical consistency. The visual appeal score together with coherence rating for AI-generated images is moderate yet not sufficient to match the superior results achieved by hand-drawn images. This legend defines the two categories of images so readers can understand the analysis results. The study shows AI achieves its main advantage in speed but humans produce superior artistic quality compared to machines.Applying SD-1.5 with VGG-1CNN and Canny edge detection, the AI-enhanced storyboard tool significantly enhances image quality, processing time, and narrative coherence. The bell curve distribution of panel adjustment times shows that optimal changes occur between 4 and 7 seconds. The concluding section of the storyboard registers the highest Consistency Score (88), but cutscene transitions require improvement. By epoch 10, the proposed model is faster (0.8s) and more efficient than ResNet, CNN, and RNN and recognizes scenes with 91% accuracy. The motion is reliably captured by maintaining high (80–95%) pose estimation accuracy throughout storyboard frames. Although AI-produced images are quicker, they are less visually pleasing and coherent compared to hand-drawn images by a slight margin.

5. Conclusion

The proposed framework integrates Stable Diffusion 1.5 (SD-1.5) and VGG-1CNN together with GPT-3.5 through an AI-based animation storyboard design system which automates game character image and narrative storyboard creation. Through the combination of textual and sketch-based input suggestions the model maintains visual harmony between different storyboard panels. An improved image generation output comes from the combination of CLIP neural networks with VAE together with scheduler-based noise removal procedures while VGGNet-based refinement functions help textual descriptions match visual features for better accuracy. Through its integration of GPT-3.5 the technology produces narrative text components that remain contextually appropriate to deliver a unified storytelling sequence. The system minimizes human working hours in animation pre-production and achieves high-quality results at the same time. This research introduces an efficient scaling AI approach to streamline animation design creation which brings extensive benefits to developers of games and digital storytelling.

References

[1] R. Gao, "AIGC technology: Reshaping the future of the animation industry," *Highlights Sci. Eng. Technol.*, vol. 56, pp. 148–152, 2023, doi: 10.54097/hset.v56i.10096.

[2] M. Izani, A. Razak, D. Rehad, and M. Rosli, "The impact of artificial intelligence on animation filmmaking: Tools, trends, and future implications," in *Proc. 2024 Int. Visualization, Informatics and Technology Conf. (IVIT)*, 2024, pp. 57–62, doi: 10.1109/IVIT62102.2024.10692804.

[3] C. Lan, Y. Wang, C. Wang, S. Song, and Z. Gong, "Application of ChatGPT-based digital human in animation creation," *Future Internet*, vol. 15, no. 9, p. 300, 2023, doi: 10.3390/fi15090300.

[4] K. He, A. Lapham, and Z. Li, "Enhancing narratives with SayMotion's text-to-3D animation and LLMs," in *ACM SIGGRAPH 2024 Real-Time Live!*, 2024, pp. 1–2, doi: 10.1145/3641520.3665309.

[5] S. Sadulla, "Next-generation semiconductor devices: Breakthroughs in materials and applications," *Prog. Electron. Commun. Eng.*, vol. 1, no. 1, pp. 13–18, 2024, doi: 10.31838/PECE/01.01.03.

[6] Z. Yang, "2D animation comic character action generation technology based on biomechanics simulation and artificial intelligence," *Mol. Cell. Biomech*., vol. 21, no. 1, p. 338, 2024, doi: 10.32604/mcb.2024.021338.

[7] P. Paudel, A. Khanal, D. P. Paudel, J. Tandukar, and A. Chhatkuli, "ihuman: Instant animatable digital humans from monocular videos," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2024, pp. 304–323, doi: 10.1007/978-3-031-73226-3_18.

[8] L. Mourot, L. Hoyet, F. Le Clerc, F. Schnitzler, and P. Hellier, "A survey on deep learning for skeleton-based human animation," *Comput. Graph. Forum*, vol. 41, no. 1, pp. 122–157, Feb. 2022, doi: 10.1111/cgf.14426.

[9] S. Sadulla, "A comparative study of antenna design strategies for millimeter-wave wireless communication," *SCCTS J. Embedded Syst. Des. Appl.*, vol. 1, no. 1, pp. 13–18, 2024, doi: 10.31838/ESA/01.01.03.

[10] N. Krome and S. Kopp, "Minimal latency speech-driven gesture generation for continuous interaction in social XR," in *Proc. 2024 IEEE Int. Conf. Artificial Intelligence and eXtended and Virtual Reality (AIxVR)*, 2024, pp. 236–240, doi: 10.1109/AIxVR59

[11] D. Gopinath, H. Joo, and J. Won, "Motion in-betweening for physically simulated characters," in *SIGGRAPH Asia 2022 Posters*, 2022, pp. 1–2, doi: 10.1145/3550082.3564186.

[12] F. Danieau et al., "Automatic generation and stylization of 3D facial rigs," in *Proc. 2019 IEEE Conf. Virtual Reality and 3D User Interfaces (VR)*, 2019, pp. 784–792, doi: 10.1109/VR.2019.8797971.

[13] J. Muralidharan, "Machine learning techniques for anomaly detection in smart IoT sensor networks," *J. Wireless Sensor Netw. IoT*, vol. 1, no. 1, pp. 15–22, 2024, doi: 10.31838/WSNIOT/01.01.03.

[14] N. Kolotouros et al., "DreamHuman: Animatable 3D avatars from text," in *Advances in Neural Information Processing Systems*, vol. 36, 2023, pp. 10516–10529, doi: 10.48550/arXiv.2306.09329.

[15] P. Sagar and A. Handa, "Exploring the mechanical, metallurgical, and fracture characteristics of hybrid-reinforced magnesium metal matrix composite synthesized via friction stir processing route," *Proc. IMechE, Part L: J. Mater.: Des. Appl.*, vol. 238, no. 5, pp. 829–844, 2024, doi: 10.1177/14644207231200640.

[16] V. C. Lungu-Stan and I. G. Mocanu, "3D character animation and asset generation using deep learning," *Appl. Sci.*, vol. 14, no. 16, p. 7234, 2024, doi: 10.3390/app14167234.

[17] P. Jin et al., "Local action-guided motion diffusion model for text-to-motion generation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2024, pp. 392–409, doi: 10.1007/978-3-031-72698-9_23.

[18] N. T. Hoa and M. Voznak, "Critical review on understanding cyber security threats," *Innov. Rev. Eng. Sci.*, vol. 2, no. 2, pp. 17–24, 2025, doi: 10.31838/INES/02.02.03.

[19] F. Lamberti, V. Gatteschi, A. Sanna, and A. Cannavò, "A multimodal interface for virtual character animation based on live performance and natural language processing," *Int. J. Hum.-Comput. Interact*., vol. 35, no. 18, pp. 1655–1671, 2019, doi: 10.1080/10447318.2018.1561068.

[20] J. Q. Zhang et al., "Write-An-Animation: High-level text-based animation editing with character-scene interaction," *Comput. Graph. Forum*, vol. 40, no. 7, pp. 217–228, Oct. 2021, doi: 10.1111/cgf.14415.

[21] X. Wang et al., "AnimatableDreamer: Text-guided non-rigid 3D model generation and reconstruction with canonical score distillation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2024, pp. 321–339, doi: 10.48550/arXiv.2312.03795.

[22] F. Lamberti, G. Paravati, V. Gatteschi, A. Cannavò, and P. Montuschi, "Virtual character animation based on affordable motion capture and reconfigurable tangible interfaces," *IEEE Trans. Vis. Comput. Graph*., vol. 24, no. 5, pp. 1742–1755, 2018, doi: 10.1109/TVCG.2017.2690433.

[23] S. Sadulla, "Techniques and applications for adaptive resource management in reconfigurable computing," *SCCTS Trans. Reconfigurable Comput.*, vol. 1, no. 1, pp. 6–10, 2024, doi: 10.31838/RCC/01.01.02.

[24] A. Rao et al., "Dynamic storyboard generation in an engine-based virtual environment for video production," in *ACM SIGGRAPH 2023 Posters*, 2023, pp. 1–2, doi: 10.1145/3588028.3603647.

[25] S. Jo, S. Shin, and S. W. Kim, "Interactive storyboarding system leveraging large-scale pre-trained model," *SSRN Preprint* 4399439, 2023, doi: 10.2139/ssrn.4399439.

[26] Z. Liang et al., "StoryDiffusion: How to support UX storyboarding with generative-AI," *arXiv:2407.07672*, 2024, doi: 10.48550/arXiv.2407.07672.

[27] H. Kim et al., "ASAP for multi-outputs: Auto-generating storyboard and pre-visualization with virtual actors based on screenplay," *Multimedia Tools Appl*., pp. 1–24, 2024, doi: 10.1007/s11042-024-17678-9.

[28] C. Y. Kim et al., "Bridging generations using AI-supported co-creative activities," *arXiv:2503.01154*, 2025, doi: 10.48550/arXiv.2503.01154.

[29] K. Geetha, "Advanced fault tolerance mechanisms in embedded systems for automotive safety," *J. Integr. VLSI, Embedded Comput. Technol.*, vol. 1, no. 1, pp. 6–10, 2024, doi: 10.31838/JIVCT/01.01.02.

[30] J. Muralidharan, "Innovative RF design for high-efficiency wireless power amplifiers," *Nat. J. RF Eng. Wireless Commun*., vol. 1, no. 1, pp. 1–9, 2023, doi: 10.31838/RFMW/01.01.01.

[31] T. Tang et al., "PlotThread: Creating expressive storyline visualizations using reinforcement learning," *IEEE Trans. Vis. Comput. Graph.*, vol. 27, no. 2, pp. 294–303, 2021 (published online 2020), doi: 10.1109/TVCG.2020.3030359.

[32] J. Kim, Y. Heo, H. Yu, and J. Nang, "A multi-modal story generation framework with AI-driven storyline guidance," *Electronics*, vol. 12, no. 6, p. 1289, 2023, doi: 10.3390/electronics12061289.
[33] M. Tao, B.-K. Bao, H. Tang, Y. Wang, and C. Xu, "StoryImager: A unified and efficient framework for coherent story visualization and completion," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2024, pp. 479–495, doi: 10.1007/978-3-031-20053-3_28.
[34] T. Fernandes, V. Nisi, N. Nunes, and S. James, "ArtAI4DS: AI art and its empowering role in digital storytelling," in *Int. Conf. Entertainment Computing*, 2024, pp. 78–93, doi: 10.1007/978-3-031-20065-6_6.
[35] E. M. Y. Chan et al., "SketchBoard: Sketch-guided storyboard generation for game characters in the game industry," in *Proc. 2024 IEEE 22nd Int. Conf. Industrial Informatics (INDIN),* 2024, pp. 1–8, doi: 10.1109/INDIN41052.2024.9557423.