Transforming Textual Tales:
A Comprehensive Study on Video Synthesis

Abstract

This research proposes a novel methodology for generating engaging videos from textual story prompts, leveraging advanced techniques in natural language processing, image generation, and video interpolation. The motivation stems from addressing the inherent barriers in traditional video creation processes, which often require specialized skills, resources, and time investments.

By integrating a fine-tuned Stable Diffusion model with coreference resolution and Google's Frame Interpolation (FILM) algorithm, the system can translate user-provided textual narratives into visually captivating comic videos featuring the popular Indian animated character Chhota Bheem. The Stable Diffusion model is fine-tuned on a custom dataset of Chhota Bheem images and scenarios, enabling it to generate relevant and contextually coherent anchor frames based on the input prompts. The SpanBERT-large model is employed for coreference resolution, preserving narrative coherence by accurately representing entities across the story. The FILM algorithm seamlessly interpolates intermediate frames between the anchor frames, resulting in smooth and natural video animations.

The system's performance is evaluated using the CLIP Score, a metric that measures the similarity between generated videos and textual prompts. Results demonstrate the effectiveness of the fine-tuned model, achieving higher CLIP Scores compared to a baseline model, particularly for shorter prompts.Additionally, the impact of prompt length on video quality is investigated, revealing challenges in maintaining coherence with longer and more complex prompts.

This research paves the way for an accessible and engaging approach to storytelling and content creation, democratizing the process of transforming textual narratives into visually compelling videos.

.

Table of Content

List of Figures

List Of Tables

List of Abbreviations

Sr No	Abbreviation	Expanded Form
1	FID	Fréchet Inception Distance
2	FVD	Fréchet Video Distance
3	IS	Inception Score
4	CLIP	Contrastive Language-Image Pretraining

1. INTRODUCTION

1.1 Background

In the era of digital media, storytelling has evolved to encompass not only written narratives but also immersive visual experiences. The ability to bring stories to life through video has become increasingly important, enabling creators to captivate audiences and convey their narratives in a more engaging and impactful manner. However, the process of transforming written stories into compelling videos often requires extensive technical expertise, resources, and time, limiting the accessibility of this medium for many individuals and organizations.

1.2 Motivation

The motivation behind this project stems from the recognition of the inherent barriers in traditional video creation processes, which often require specialized skills, resources, and time investments. By harnessing advanced techniques, the aim is to democratize video creation, empowering individuals across various domains to effortlessly transform textual stories into visually captivating videos. The architecture addresses key gaps in existing literature, such as fine-tuning limitations, contextual coherence, and motion omission challenges, thereby paving the way for a more accessible and inclusive approach to storytelling and content creation. Through seamless integration of natural language processing, image generation, frame interpolation, and text-to-speech synthesis, the framework strives to offer a user-friendly solution that transcends the boundaries of conventional media production methods, fostering creativity and enabling diverse narratives to be shared in engaging visual formats.

2. REVIEW OF LITERATURE

2.1 Text to Video

Yingqing He [7]

In their 2023 work, Yingqing He et al. propose a Unified Video Diffusion Model using hierarchical latent video diffusion for high-fidelity long video generation. The methodology incorporates Prediction 3D U-Net, Interpolation 3D U-Net, and a lightweight 3D autoencoder. Achieving notable results (FVD: 95.2, KVD: 3.9) on UCF-101, Sky Time-lapse, and taichi datasets, the study calls for architectural improvements to expedite training while recognizing the computational challenges and the need for high-quality datasets in achieving realistic video generation.

Xia [13]

In Retrieval Augmented video generation, He and Xia introduce a novel paradigm combining adjustable structure-guided text-to-video synthesis and TimeInv for personalized content. The methodology emphasizes Motion Structure Retrieval, evaluated with FVD (516.15) and KVD (47.78) metrics on UCF-101. The study proposes potential enhancements, highlighting the necessity for a general character control mechanism and a more effective cooperation strategy between character and structure control.

Hong, Y.[11]

Reuse and Diffuse employs Latent Diffusion Models for image synthesis, evaluated with FID and SSIM metrics. The model utilizes diverse datasets, including WebVid-2M and Moments-In-Time, and encourages the use of additional modalities for enhanced video realism. However, a limitation exists as the framework relies on an initial video clip with a small number of frames.

Zhang, Y[14]

ControlVideo presents a training-free approach to controllable text-to-video generation, featuring cross-frame interaction, interleaved-frame smoother, and hierarchical sampling. The methodology evaluates video consistency through Frame Consistency (cosine similarity

between consecutive frames) and Prompt Consistency (cosine similarity between input prompt and all video frames). The study introduces a novel framework for achieving control in text-to-video synthesis.

Wenyi Hong [3]

CogVideo, a 2021 text-to-video model by Jiang et al., employs a hierarchical multi-frame-rate transformer integrated with CogView2 and CogFlow for object deformations and motion realism. Achieving state-of-the-art Fréchet Video Distance and Inception Score on Kinetics-600, the PyTorch-based model is lauded for semantic relevance, motion realism, and texture quality. Acknowledging limitations in fine-grained control and potential bias, the authors suggest exploring additional modalities like audio and text descriptions for enhanced diversity in generated videos

2.2 Text to Image to Video

Uriel Singer [1]

Developed an advanced video generation model using text-to-image capabilities, spatiotemporal convolutional layers, and U-Net-based diffusion networks. Leveraging 10 million WebVid videos and 2.3 billion LAION-5B text-image pairs, it produced high-quality, coherent videos across diverse visual concepts. Despite its ability to generate longer videos, multiple scenes, and detailed storylines, it struggles to infer associations between text and exclusive video phenomena.

Levon Khachatryan [6]

Developed Text2Video-Zero, a zero-shot video generation model. Using motion dynamics and cross-frame attention, it preserves object appearance and context, generating consistent videos without relying on a specific dataset. Despite a notable CLIP score, it lacks fine-tuning due to the absence of a dedicated video dataset.

Yogesh Balaji [15]

Presented a Conditional GAN for Text-to-Video Synthesis using TFGAN. Their model, incorporating a recurrent network and a shared frame generator, achieved impressive FVD and KVD metrics on synthetic and real-world datasets like Kinetics, Epic-Kitchens, and others.

Laura Sevilla-Lara [5]

Frozen in Time, an end-to-end trainable model combining ViT and Timesformer architectures for joint video and image encoding. Outperforming state-of-the-art models on HowTo100M, it faces challenges in handling noisy datasets and addressing computational demands for practical applications.

Doyeon Kim [8]

Introduced TiVGAN, a groundbreaking two-stage model for text-to-image-to-video generation. Achieving superior qualitative results on KTH Action, MUG, and Kinetics datasets, it aims to improve learning paradigms and enhance diversity in video generation.

2.3 Parameters used for Analysis

Performance Evaluation

FID (Fréchet Inception Distance): FID measures the similarity between generated images and real images using statistics from a pre-trained neural network. Lower FID scores indicate better visual quality and diversity in the generated images.

KVD Score: The KVD (Kernel Video Distance) Score is a metric used to evaluate the quality and diversity of video sequences. Similar to FID for images, KVD assesses the distance between feature representations of generated videos and real videos. Lower KVD scores suggest higher quality and diversity in the generated videos.

Context Evaluation

Capturing Context: For video generation, maintaining context across frames is crucial for coherence and continuity. It involves ensuring that the model comprehends and represents the relationships between different elements or scenes in the video. This includes understanding object interactions, scene transitions, and overall narrative coherence.

Contextual Continuity: The assessment involves evaluating how well the generated video frames flow seamlessly, maintaining contextual consistency throughout the sequence. Context evaluation checks if the model preserves the storyline or intended narrative across frames without abrupt or inconsistent changes.

Length of prompt or text

Impact on Generation: Longer prompts or textual inputs can significantly influence the difficulty of generating coherent video frames. Longer prompts may provide more detailed context or instructions but can also make it challenging for the model to synthesize each frame while adhering to the entire context.

Complexity vs. Coherence: Longer prompts might introduce complexities that the model struggles to interpret or represent coherently across frames. On the other hand, shorter prompts might lack comprehensive context, leading to potential gaps or inconsistencies in the generated video sequences.

2.4 Analysis of Literature Survey

Table 2.4.1 - Text to Video

Title	**Year & Author**	**Performance Evaluation**	**Context Evaluation**	**Length of Input / Prompt**
Latent Video Diffusion Models for High-Fidelity Long Video Generation	2023 Yingqing He	FVD: 95.2 KVD: 3.9	NO	YES
Retrieval Augmented video generation	YINGQING HE∗ , MENGHAN XIA∗	FVD : 516.15 KVD:47.78	YES	NO
Reuse and Diffuse : Iterative denoising for text - to video generation	Jiaxi Gu,Hang Xu	FVD : 363.19 IS : 39.37	NO	YES

ControlVideo for controllable text-to-video generation	2017, Yogesh Balaji1	Frame Consistency : 97.22	NO	NO
CogVideo: Text-to-Video Generation with a Hierarchical Multi-Frame- Rate Transformer	2021, Yifan Jiang, Yuxin Peng	FVD : 626 IS : 50.46	NO	NO

Table 2.4.2 - Text to Image to Video

Title	**Year and Author**	**Performance Evaluation**	**Context Evaluation**	**Length of input /prompt**
Make-A-Video	2023 Levon Khachatryan	FID: 13.17 CLIPISM: 0.3049 FVD: 81.25	YES	NO
Conditional GAN with Discriminative Filter Generation for Text-to-Video Synthesis	2017,Yogesh Balaji1, Martin Renqiang Min2	FVD : 31.76 KVD : 7.19	NO	YES - suggests using RNN
Frozen in Time: A Joint Video and Image Encoder for End-to-End Retrieval	Laura Sevilla-Lara	Compared with established models	NO	YES - result in the addition of temporal positional embeddings

TiVGAN: Text to Image to Video Generation With Step-by-Step Evolutionary Generator	2020,Doyeon Kim	FID: 47.34 IS:5.34	YES	YES - contain more complex information and require more processing power.
Text2Video-Zero:Zero-shot video generators	2023 Levon Khachatryan	CLIP:31.19	NO	NO

2.5 Gaps in existing Literature Survey:

1. **Fine-Tuning Limitation**

Fine-tuning refers to the process of taking a pre-trained machine learning model and further training it on a specific task or dataset to adapt it to the nuances of that particular domain

2. **Contextual Gaps**

When generating subsequent video frames for the next prompt, it's crucial for the model to consider contextual gaps, meaning that it should take into account the context of the previous input frames to ensure coherence and continuity in the video sequence.

3. **Motion Omission Challenge**

The challenge of motion omission arises, particularly with lengthy prompts, making it difficult for the model to effectively generate coherent video frames due to potential gaps or inconsistencies in capturing dynamic elements and motion cues.

3. Problem Definition & Objectives

The primary objective is to develop a user-friendly system capable of producing precise and captivating comic videos that align seamlessly with user-provided story prompts. By doing so, the aim is to elevate the overall experience of transforming text into visually compelling narratives, addressing the current limitations and ensuring a more accessible and engaging process.

The architecture addresses key gaps in existing literature, such as fine-tuning limitations, contextual coherence, and motion omission challenges, thereby paving the way for a more accessible and inclusive approach to storytelling and content creation. Through seamless integration of natural language processing, image generation, frame interpolation, and text-to-speech synthesis.

4. Proposed Solution

4.1 Dataset and Preprocessing

The dataset used in this project was created by a collection of images of "Chhota Bheem" and this is a popular Indian animated television series aimed primarily at children. It revolves around the adventures of a young boy named Bheem, who lives in the fictional kingdom of Dholakpur. Bheem is known for his extraordinary strength, intelligence, and bravery, often using these qualities to protect his friends, the people of Dholakpur.

A substantial number of images of the television series and various other scenarios/background were gathered to ensure robustness of the dataset.Furthermore, to enrich the dataset and enhance the quality of text-to-video generation people of different ages were invited to write captions or textual descriptions for the images. This resulted in a broad range of descriptions for each picture. By gathering input from various individuals, a multitude of perspectives and interpretations were obtained. This diversity enriched the contextual variety of the textual prompts linked to each image.

4.2 Model architecture

The proposed methodology begins with the story input from the user, which are passed to the Coreference Resolution model to and preprocessed to generate 5 contextualized representations that capture the semantic relationships and contextual information.These prompts are then used as input to a fine-tuned Stable Diffusion model, which has been adapted to the Chhota Bheem image dataset.

The story is passed to Google'sText to Speech model to generate the narrating audio. The number of intermediate frames to be generated is calculated with the help of the timestamp and the frames per

second. The frames are generated with the help of latents which are passed to the model to create frames. These frames are then passed to Frame interpolation to generate smooth video. This video is merged with audio resulting in an engaging multimedia video of Chhota Bheem.

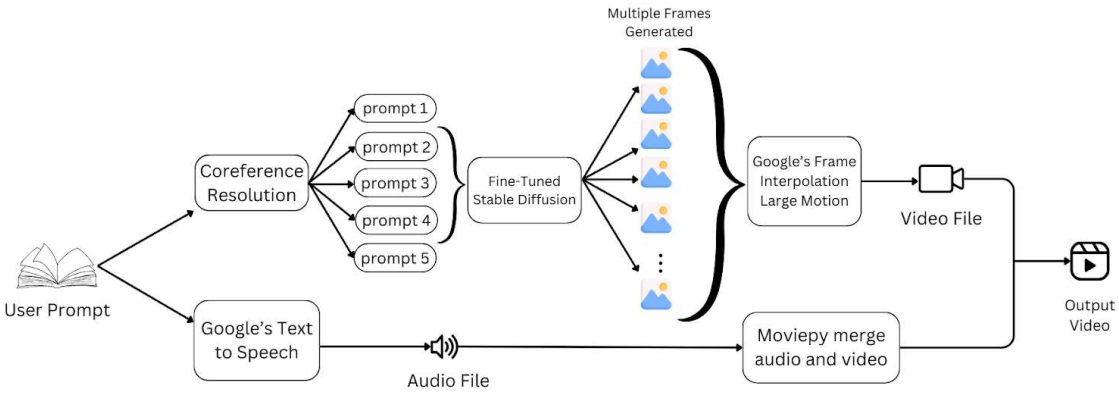

Fig no. 4.1.1: Model Architecture

4.3 Coreference Resolution

The SpanBERT model is a language model which has been pre-trained on a massive corpus of text data. The model is capable of learning and capturing semantic relationships and contextual information present in natural language text. The set of contextualized prompts are then used as input to the fine-tuned Stable Diffusion model. This ensures that the image generation process is guided by the contextual information extracted.

Coreference Resolution is the task of clustering mentions in text which refer to the same real-world entities. This is used to maintain the context of the story and therefore maintaining context throughout the process of video generation.

$$P(y) = \frac{e^{s(x,y)}}{\sum_{y_0 \in Y} e^{s(x,y_0)}} \tag{1}$$

The span pair scoring function s(x, y) is a feedforward neural network over fixed length span representations and hand-engineered features over x and y:

$$s(x,y) = s_m(x) + s_m(y) + s_c(x,y) \tag{2}$$

Here g_x and g_y denote the span representations, which are a concatenation of the two transformer output states of the span endpoints and an attention vector computed over the output representations of the token in the span. F F N Nm and F F N Nc represent two feedforward neural networks with one hidden layer and φ(x, y) represents the hand-engineered features.

$$s_m(x) = FFNN_m(g_x) \tag{3}$$

$$s_c(x, y) = FFNN_c(g_x, g_y, \phi(x, y)) \tag{4}$$

4.4 Text to Speech

This method first analyzes the text, then processes and understands the inputted texts, then converts to digital audio and speaks in the language for which the text was made. This is used to add narration to the video and deliver an engaging multimedia experience to the user.

4.5 Stable Diffusion

Diffusion models are probabilistic models designed to learn a data distribution p(x) by gradually denoising a normally distributed variable, which corresponds to learning the reverse process of a fixed Markov Chain of length T . The model can be interpreted as an equally weighted sequence of autoencoders $\epsilon_\theta(x_t, t)$; t = 1, 2, ...T which are trained to predict a denoised variant of their input xt,
where xt is a noisy version of the input x.

$$L_{LDM} - E_{\varepsilon(X),\varepsilon\sim N(0,1),t}[||\varepsilon - \varepsilon_\theta(z_t, t)||_2^2] \tag{5}$$

The equation labeled as LLDM represents the loss function for the Stable Diffusion model.

• L_{LDM} : This is the overall loss function for the Stable Diffusion model.

• Eε(X),ε~N (0,1),t[·]: This denotes the expectation operator, where the expectation is taken over three variables:

ε(X): This represents the noise added to the input data X.

ε: This is a random variable sampled from a standard normal distribution N (0, 1), which is used as a source of randomness in the denoising process.

t: This represents the time step in the diffusion process.

$||\varepsilon - \varepsilon\theta (zt, t)||2$: This is the squared Euclidean distance between the noise ε and the output of the diffusion model $\varepsilon\theta (zt, t)$, where zt is the latent representation at time step t and θ represents the parameters of the diffusion model.

Minimizing this loss function during training ensures that the diffusion model effectively learns to denoise the input data and generate coherent output consistent with the desired characteristics.

This formulation of the Stable Diffusion model serves as the core image generation component in the proposed methodology. By fine-tuning the Stable Diffusion model on the Chhota Bheem dataset, we can ensure that the generated images are consistent with the desired style and content characteristics of the Chhota Bheem characters and settings.

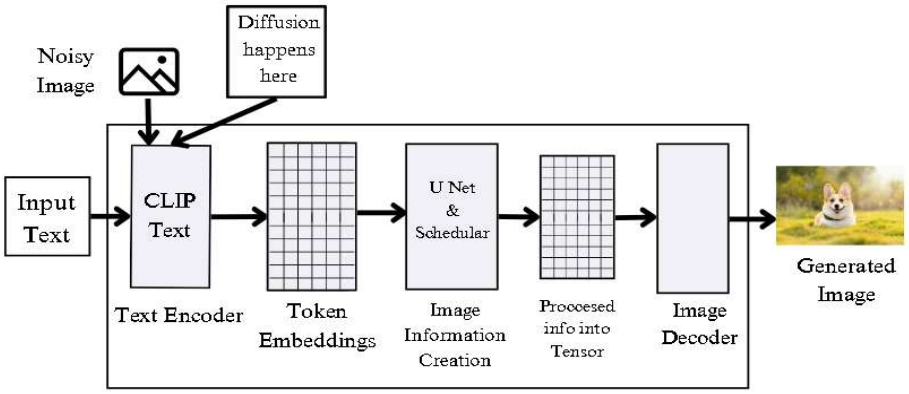

Fig no. 4.5.1: Architecture of Stable Diffusion

4.6 Fine Tuned Stable Diffusion

The pre-trained Stable Diffusion model is fine-tuned on a custom dataset of Chhota Bheem images. This process aims to adapt the model's parameters to ensure the generated images are consistent with the desired style and content of the characters and settings.

Once the Stable Diffusion model has been fine-tuned, it is used to generate the anchor frames for the final video. The fine-tuned model takes the contextualized representations h1, h2, ..., h5 produced by the SpanBERT model as input and generates the corresponding images.

$$E_{x,c,\epsilon,\epsilon',t}[w_t||\hat{x_\theta}(\alpha_t x + \sigma_t \epsilon, c) - x||_2^2 +_{t'} ||\hat{x_\theta}(\alpha_{t'} x_{pr} + \sigma_{t'} \epsilon, c_{pr}) - x_{pr}||_2^2] \quad (6)$$

The equation above represents a loss function used in a certain context.

• $E_{x,c,\varepsilon,\varepsilon'}$, t[·]: This denotes the expectation operator, where the expectation is taken over four variables:

x: Input data.

c: Condition or context.

ε: Random noise sampled from a certain distribution.

ε′: Another random noise sampled from a certain distribution.

t: Time step.

• wt: Weight associated with time step t.

• $||x\hat{}_\theta(\alpha_t x + \sigma_t \varepsilon, c) - x||_2^2$: This term represents the squared Euclidean distance between the predicted output ˆxθ (αtx + σtε, c) and the actual input x, where ˆxθ is a function parameterized by θ, αt and σt are scaling factors, and c represents the condition or context.

•$\lambda_{t'} ||x\hat{}_\theta(\alpha_t x + \sigma_t \varepsilon, c_{pr}) - x_{pr}||_2^2$: This term represents another squared Euclidean distance between the predicted output and the actual input, but this time for a different time step t′ and different input x_{pr} and context c_{pr}. This loss function captures the discrepancy between the predicted outputs and the actual inputs across different time steps and contexts, weighted by certain factors.

4.7 Latents to Frames

For each intermediate frame initial latents are generated to initiate the diffusion process. Then the step fraction is calculated which represents the position between the start and end prompts for the current frame. Using spherical linear interpolation, it interpolates between the start and end text embeddings to obtain the next and mid point text embeddings. The initial latents are then denoised using the midpoint text embedding for a specified number of steps.

Subsequently, the denoising process continues on these midpoint latents, but this time guided by the current text embedding, for the remaining steps. After the denoising process is complete, the final latents are converted into an image, which is saved to disk with the appropriate frame number.Finally, the current text embedding is updated to the next

embedding, preparing for the generation of the subsequent intermediate frame.The algorithm followed to generate the frames required for the entire video:

Algorithm 1 Frame Interpolation (FILM)

Require: Prompts $\{p_1, p_2, \ldots, p_n\}$, fps, smoothing frames, guidance scale, seed

Initialize *seed*
Set $batch\_size = 1$
for $i = 1$ to $n - 1$ **do**
    $s \leftarrow p_i$
    $e \leftarrow p_{i+1}$
    $s\_num \leftarrow \lfloor s[\text{“}ts\text{”}] \times fps \rfloor$
    $e\_num \leftarrow \lfloor e[\text{“}ts\text{”}] \times fps \rfloor$
    $f\_needed \leftarrow e\_num - s\_num$
    $i\_frames \leftarrow \lfloor f\_needed / s\_frames \rfloor - 1$
    $do\_free\_guide \leftarrow (g\_scale > 1.0)$
    $s\_text \leftarrow pipe.embed\_text(s[\text{“}prompt\text{”}], do\_free\_guide, batch\_size)$
    $e\_text \leftarrow pipe.embed\_text(e[\text{“}prompt\text{”}], do\_free\_guide, batch\_size)$
    $c\_text \leftarrow s\_text$
    for $j = 1$ to $i\_frames + 1$ **do**
        $gen \leftarrow torch.Generator(\text{“}cuda\text{”}).manual\_seed(seed)$
        $init\_sch \leftarrow make\_scheduler(num\_steps)$
        $i\_steps \leftarrow \lfloor num\_steps \times (1 - prompt\_strength) \rfloor$
        $i\_lat \leftarrow torch.randn((batch\_size, pipe.unet.in\_channels, h//8, w//8), generator = gen, device = \text{“}cuda\text{”})$
        $step\_frac \leftarrow j/(i\_frames + 1)$
        $e\_next \leftarrow slerp(step\_frac, s\_text, e\_text)$
        $m\_text \leftarrow slerp(0.5, c\_text, e\_next)$
        $m\_lat \leftarrow pipe.denoise(latents = i\_lat, text\_embeddings = m\_text, t\_start = 1, t\_end = i\_steps, g\_scale = g\_scale)$
        $f\_num \leftarrow \lfloor s\_num + (step\_frac \times f\_needed) \rfloor$
        Re-init $pipe.scheduler \leftarrow make\_scheduler(num\_steps, init\_sch)$
        $lat \leftarrow pipe.denoise(latents = m\_lat, text\_embeddings = c\_text, t\_start = i\_steps, t\_end = None, g\_scale = g\_scale)$
        $img \leftarrow pipe.latents\_to\_image(lat)$
        $save\_img(pipe.numpy\_to\_pil(img)[0], path = f\text{“}f\_dir/f\_num\text{”}.zfill(5) + \text{“}.png\text{”})$
        $c\_text \leftarrow e\_next$
    end for
end for

Fig no. 4.7.1:Frame Interpolation Algorithm

4.8 Frame Interpolation

The FILM algorithm leverages advanced techniques, such as optical flow and learned representations, to smoothly transition between the anchor frames. This allows the generation of intermediate frames that seamlessly blend the visual content, resulting in a more natural and visually appealing animation. By integrating the FILM interpolation technique with the Stable Diffusion model, the proposed methodology can generate high-quality animations from the provided textual.

prompts. The FILM algorithm plays a crucial role in addressing the potential issues of large motion and discontinuities that may arise from the anchor frames alone, leading to a more compelling and cohesive final video output.

$$(F^l_{t\leftarrow 1}, F^l_{t\leftarrow 0}, W^l_{t\leftarrow 0}, W^l_{t\leftarrow 1}) \tag{7}$$

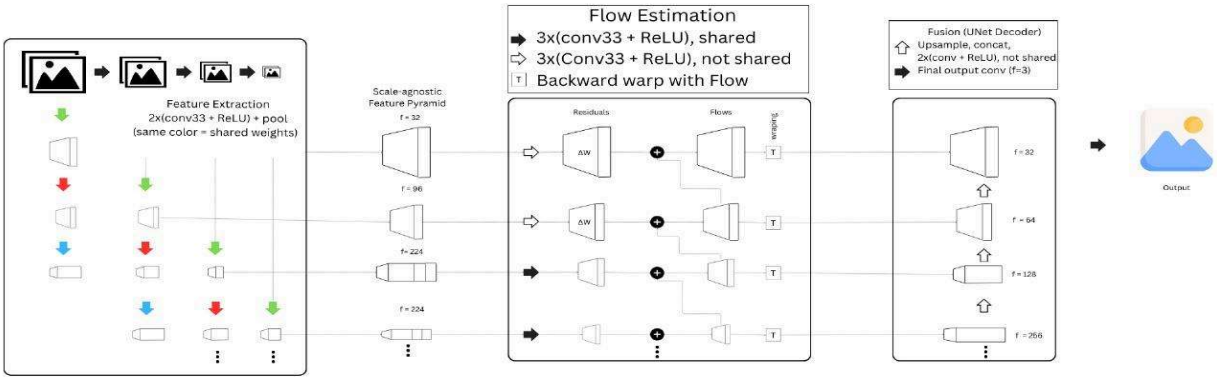

Fig no. 4.8.1: Architecture of Google's FILM

5. Result

5.1 Coreference resolution

Following is a comparison of all the previous results of different coreference resolution systems used on CoNLL-2012 shared task dataset, which is widely used for benchmarking coreference resolution systems.

The SpanBERT-large model, a state-of-the-art transformer-based architecture specifically designed for coreference resolution tasks.It has demonstrated superior performance compared to other models

Table 5.1.1 - Comparison of Previous Works

Model	F1 %
BERT - base	73.9
SpanBERT - base	77.7
BERT - large	76.9
SpanBERT - large	79.6

5.2 Model Performance Comparison

To evaluate the performance of the stable diffusion models in generating videos from text prompts related to Chhota Bheem characters, we used the CLIP Score metric.

The CLIP Score is the measure of similarity between the generated video and the text prompt, with higher scores indicating better result. The model learns to embed both images and text into a shared latent space, where semantically similar images and text descriptions are mapped to similar vectors.The CLIP Score is calculated by first obtaining the image and text embeddings from the CLIP model, and then computing the cosine similarity between these embeddings. The higher the cosine similarity, the more closely aligned the image (or video) is with the text prompt.

$$\text{CLIPScore}(I, C) = \max(100 \times \cos(EI, EC), 0) \tag{8}$$

Compared two different models:

1) A baseline stable diffusion model (V1-5) without any specialized training.

2) A custom stable diffusion model trained on a dataset of television ,Chhota Bheem and various other scenarios/background.The models were tested on five text prompts of varying lengths, ranging from a single line to five lines describing scenes involving Chhota Bheem in different scenarios.

Table 5.2.1 - Model Performance Comparison

Prompts	**Simple Stable Diffusion Model (V1 - 5)**	**Stable Diffusion Model trained on custom data**
P11 - Shopping Market	33.983	33.543
P12 - Desert	32.607	38.212
P13 - Forest	31.410	39.108
P14 - Underwater	35.383	32.457
P15 - Football	31.403	31.341
P31 - Shopping Market	34.844	30.170
P32 - Desert	32.490	31.643
P33 - Forest	34.518	31.109
P34 - Underwater	30.524	28.304
P35 - Football	31.766	30.187
P51 - Shopping Market	32.721	34.866
P52 - Desert	32.896	32.268
P53 - Forest	32.513	35.595
P54 - Underwater	25.670	28.2794
P55 - Football	30.687	31.014

The results show that the custom model trained on the Chota Bheem dataset consistently outperformed the baseline model across all prompt lengths. This suggests that the specialized training data, which included visual representations and descriptions of Chota Bheem characters, enabled the model to better understand and generate relevant videos for the given prompts.

Additionally, observed a trend where the CLIP Scores decreased as the prompt length increased for both models. This indicates that longer prompts, which likely contained more detailed and complex descriptions, posed a greater challenge for the models in accurately capturing all the elements in the generated videos.

While the custom model maintained a higher CLIP Score than the baseline model even for longer prompts, the performance gap between the two models narrowed as the prompt length increased. This could be attributed to the increased complexity introduced by longer prompts, which may have exceeded the capabilities of the models or led to inconsistencies in the generated content.

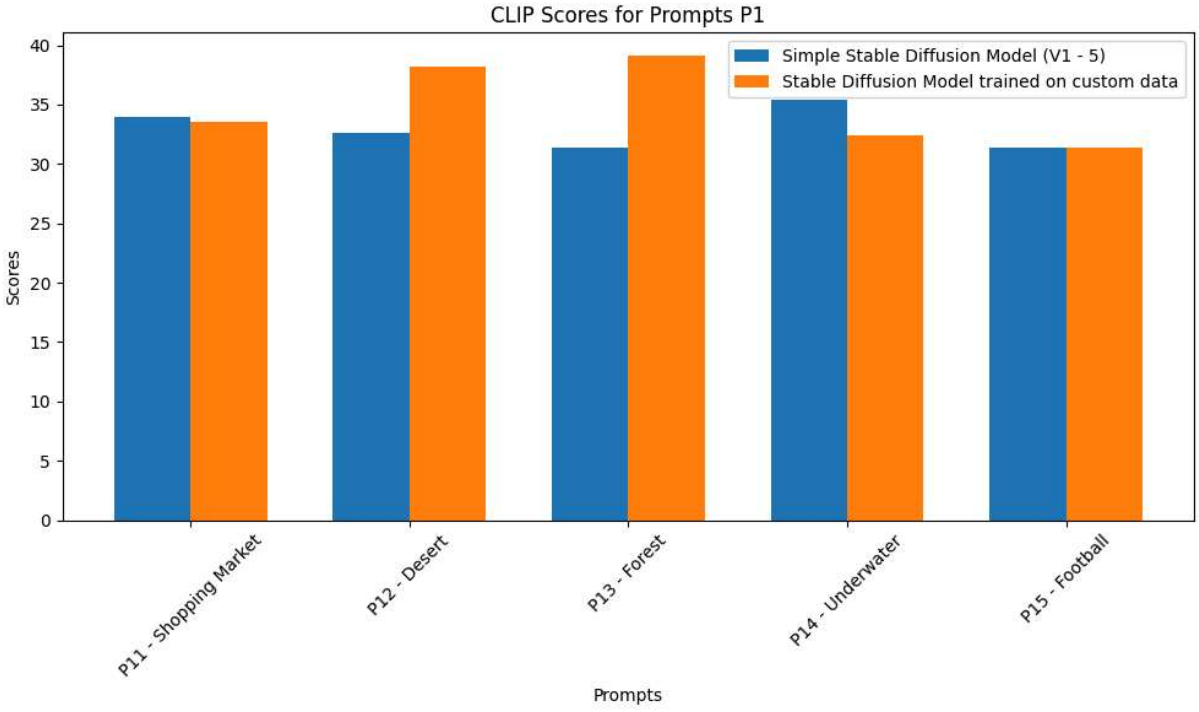

Fig no. 5.2..1: One Line prompt Comparison

6.Conclusion

In this research, a stable diffusion model v.1.5 is integrated to translate user textual prompts into images. Furthermore, a custom model is fine-tuned on a dataset featuring various scenarios involving Chhota Bheem and its characters, enhancing its performance in context-specific content creation. This is complemented by the Google film interpolation model for converting images into videos. The proposed approach maintains contextual coherence using the SpanBERT-large model and investigates the impact of prompt length on video quality. Through coreference resolution, accurate entity representation is achieved, thereby preserving narrative coherence. However, longer prompts present challenges, resulting in diminished video quality as indicated by the CLIP Score. Notably, the custom model achieves an average CLIP Score of x, surpassing the original model's score of y.

7.Future Scope

Developing models that understand multilingual textual prompts, making the technology accessible to a wider audience.Furthermore, integrating this technology across various fields like education, entertainment, and advertising has the potential to revolutionize content creation, offering more engaging and personalized experiences.

8. References

[1] Uriel Singer, undefined., et al, "Make-A-Video: Text-to-Video Generation without Text-Video Data," 2022.

[2] Y. Li, M. Min, D. Shen, D. Carlson, and L. Carin, “Video Generation From Text”, *AAAI*, vol. 32, no. 1, Apr. 2018.

[3] Wenyi Hong, undefined., et al, "CogVideo: Large-scale Pretraining for Text-to-Video Generation via Transformers," 2022.

[4] Wu, J., et al, "Tune-A-Video: One-Shot Tuning of Image Diffusion Models for Text-to-Video Generation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023, pp. 7623-7633.

[5] Max Bain, undefined., et al, "Frozen in Time: A Joint Video and Image Encoder for End-to-End Retrieval," 2022.

[6] Levon Khachatryan, undefined., et al, "Text2Video-Zero: Text-to-Image Diffusion Models are Zero-Shot Video Generators," 2023.

[7] Yingqing He, undefined., et al, "Latent Video Diffusion Models for High-Fidelity Long Video Generation," 2023.

[8] D. Kim, D. Joo and J. Kim, "TiVGAN: Text to Image to Video Generation With Step-by-Step Evolutionary Generator," in *IEEE Access*, vol. 8, pp. 153113-153122, 2020, doi: 10.1109/ACCESS.2020.3017881.

[9] Y. Hu, C. Luo, Z. Chen, "Make It Move: Controllable Image-to-Video Generation With Text Descriptions," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 18219-18228.

[10] Deng, K., et al, "IRC-GAN: Introspective Recurrent Convolutional GAN for Text-to-video Generation," in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, 2019, pp. 2216–2222.

[11] Hong, Y., Chen, X., Xu, Y., Liu, M., Wang, M., Li, W., Zhou, X., Li, L., & Zhou, J. (2023). Reuse and diffuse: Iterative denoising for text-to-video generation [arXiv preprint arXiv:2309.03549].

[12] Levon Khachatryan, Andranik Movsisyan, Vahram Tadevosyan, Roberto Henschel, Zhangyang Wang, Shant Navasardyan, & Humphrey Shi. (2023). Text2Video-Zero: Text-to-Image Diffusion Models are Zero-Shot Video Generators.

[13] Yingqing He, Menghan Xia, Haoxin Chen, Xiaodong Cun, Yuan Gong, Jinbo Xing, Yong Zhang, Xintao Wang, Chao Weng, Ying Shan, & Qifeng Chen. (2023). Animate-A-Story: Storytelling with Retrieval-Augmented Video Generation.

[14] Zhang, Y., Wei, Y., Jiang, D., Zhang, X., Zuo, W., & Tian, Q. (2023). ControlVideo: Training-free Controllable Text-to-Video Generation [arXiv preprint arXiv:2305.13077].

[15] Balaji, Y., Min, M., Bai, B., Chellappa, R., & Graf, H. (2019). Conditional GAN with Discriminative Filter Generation for Text-to-Video Synthesis. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence* (pp. 1995–2001). AAAI Press.

[16] Max Bain, Arsha Nagrani, Gül Varol, & Andrew Zisserman. (2022). Frozen in Time: A Joint Video and Image Encoder for End-to-End Retrieval.

[17] Levon Khachatryan, Andranik Movsisyan, Vahram Tadevosyan, Roberto Henschel, Zhangyang Wang, Shant Navasardyan, & Humphrey Shi. (2023). Text2Video-Zero: Text-to-Image Diffusion Models are Zero-Shot Video Generators.

[18] Zhengxiong Luo, Dayou Chen, Yingya Zhang, Yan Huang, Liang Wang, Yujun Shen, Deli Zhao, Jingren Zhou, & Tieniu Tan. (2023). VideoFusion: Decomposed Diffusion Models for High-Quality Video Generation.

[19] Omer Bar-Tal,DolevOfri-Amar, Rafail Fridman, Yoni Kasten, and Tali Dekel. Text2live: Text-driven layered image and video editing. In Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XV, pages 707–723. Springer, 2022.

[20] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. Advances in Neural Information Processing Systems, 34:8780–8794, 2021.

[21] Patrick Esser, Johnathan Chiu, Parmida Atighehchian,Jonathan Granskog, and Anastasis Germanidis. Structureand content-guided video synthesis with diffusion models.arXiv preprint arXiv:2302.03011,2023.

[22] Oran Gafni, Adam Polyak, Oron Ashual, Shelly Sheynin, Devi Parikh, and Yaniv Taigman. Make-a-scene: Scenebased text-to-image generation with human priors.In Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XV, pages 89–106. Springer, 2022

[23] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, BingXu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. Communications of the ACM, 63(11):139–144, 2020

[24] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch,Michael Rubinstein,and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. arXiv preprint arXiv:2208.12242, 2022.

[25] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu,and Mark Chen. Hierarchical text-conditional image generation with clip latents. arXiv preprint arXiv:2204.06125, 2022