Enhancing Sketch Animation: Text-to-Video Diffusion Models with Temporal Consistency and Rigidity Constraints

Gaurav Rai[a] and Ojaswa Sharma[b]

Graphics Research Group, Indraprastha Institute of Information Technology Delhi, India

Keywords: Sketch Animation, Bézier Curve, Diffusion Models, Control Points, Regularization, As-Rigid-As-Possible.

Abstract: Animating hand-drawn sketches using traditional tools is challenging and complex. Sketches provide a visual basis for explanations, and animating these sketches offers an experience of real-time scenarios. We propose an approach for animating a given input sketch based on a descriptive text prompt. Our method utilizes a parametric representation of the sketch's strokes. Unlike previous methods, which struggle to estimate smooth and accurate motion and often fail to preserve the sketch's topology, we leverage a pre-trained text-to-video diffusion model with SDS loss to guide the motion of the sketch's strokes. We introduce length-area (LA) regularization to ensure temporal consistency by accurately estimating the smooth displacement of control points across the frame sequence. Additionally, to preserve shape and avoid topology changes, we apply a shape-preserving As-Rigid-As-Possible (ARAP) loss to maintain sketch rigidity. Our method surpasses state-of-the-art performance in both quantitative and qualitative evaluations. https://graphics-research-group.github.io/ESA/.

1 INTRODUCTION

Sketches serve as a medium for communication and visual representation. Animating 2D sketch illustrations using traditional tools is tedious, cumbersome, and requires significant time and effort. Keyframe-based animation is highly labor-intensive, while video-driven animation methods are often restricted to specific motions. In recent years, sketch animation has emerged as a significant area of research in computer animation, with applications in video editing, entertainment, e-learning, and visual representation. Previous sketch animation methods, such as that in (Xing et al., 2015; Patel et al., 2016), require extensive manual input and artistic skill, presenting challenges for novice users. Traditional methods are limited to specific types of motion, such as facial and biped animation. More recent techniques, like Su et al. (Su et al., 2018), animate sketches based on a video, but still require manual input. Animation-Drawing (Smith et al., 2023) is a sketch animation technique that does not rely on manual input, generating animations using pose mapping, but is limited to biped motion. In contrast, LiveSketch (Gal et al., 2023) is a learning-based approach that takes a

sketch and text prompt to produce an animated sketch. While it generates promising results, it faces challenges with temporal consistency and shape preservation (see Figure 1). To address these issues, we propose a method for animating input sketches based solely on a text description, with no manual input required. Our approach represents each stroke as a Bézier curve, similar to LiveSketch (Gal et al., 2023), and extends LiveSketch's capabilities by a novel Length-Area regularization and rigidity loss. Furthermore, we utilize local and global paths for motion estimation and apply Score Distillation Sampling (SDS) loss (Poole et al., 2022) for optimization. We propose length-area (LA) regularization that maintains temporal consistency, yielding smooth and accurate motion in the animated sketch. Further, the As-Rigid-As-Possible (ARAP) loss (Igarashi et al., 2005) preserves local rigidity in the sketch's shape during animation. Our method outperforms state-of-the-art techniques in both quantitative and qualitative evaluations. We achieve better sketch-to-video consistency and text-to-video alignment compared to previous method. Our main contributions are as follows:

- We propose a Length-Area regularization to maintain temporal consistency across animated sequences. It allows for the generation of a smooth animation sequence.
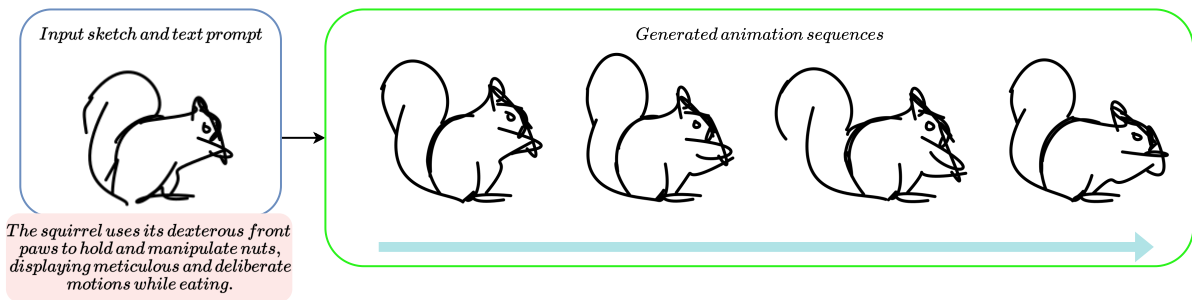
151

Figure 1: Problem with the LiveSketch (Gal et al., 2023) method. In this example, we can observe the lack of temporal consistency and shape distortion during motion.

• A shape-preserving ARAP loss to preserve local rigidity in sketch strokes during animation. The rigidity loss overcomes the shape distortion during animation.

2 RELATED WORK

2.1 Sketch Animation

Traditional sketch animation tools are time-consuming and require a certain level of artistic skills. Agarwala et al. (Agarwala et al., 2004) proposed a rotoscoping approach that estimates motion from contour tracking and animates the sketches. It reduces manual user inputs in the contour-tracking process. Bregler et al. (Bregler et al., 2002) extract motion from cartoon animated characters and re-target these to sketches using the keyframe-based approach, producing more expressive results but requires additional user inputs. Guay et al. (Guay et al., 2015) propose a method that enables timestep shape deformation by sketching a single stroke but is limited to a few animation styles. Autocomplete methods (Wang et al., 2004; Xing et al., 2015) predict the subsequent sketching style by the user using temporal coherency, but these methods require manual user input for sketching operations for each keyframe. Several learning-based and energy optimization-based animation methods have been proposed in recent years. These methods aim to perform animation using video motion, text prompt input, and predefined motion trajectory. Santosa et al. (Santosa et al., 2013) animate a sketch by marking over the video using optical flow. However, this method suffers in the case of structural differences between the sketch and the video object.

Deep learning-based methods (Liu et al., 2019; Jeruzalski et al., 2020; Xu et al., 2020) provide an alternative for animators by demonstrating robust capacity for rig generation. Animation Drawing (Smith et al., 2023) generates a rigged character of children's drawing using the alpha-pose mapping from a pre-defined character motion. SketchAnim (Rai et al., 2024) maps the video skeleton to the sketch skeleton and estimates the skeleton transformation to animate the sketch using skinning weights. It handles self-occlusion and can animate non-living objects but fails to animate stroke-level sketches. Character-GAN (Hinz et al., 2022) generates an animation sequence (containing a single character) by training a generative network with only 8-15 training samples with keypoint annotation defined by the user. Neural puppet (Poursaeed et al., 2020) adapts the animation of hand-drawn characters by providing a few drawings of the characters in defined poses. Video-to-image animation (Siarohin et al., 2019; Siarohin et al., 2021; Wang et al., 2022; Mallya et al., 2022; Tao et al., 2022; Zhao and Zhang, 2022) methods extract the motion of keypoint learning-based optical flow estimated from the driving video and generate the animated images. However, it is limited to the image modality. AnaMoDiff (Tanveer et al., 2024) estimates the optical flow field from a reference video and warps it to the source input. Su et al. (Su et al., 2018) defines control points on the first frame of the video, tracks the control points in the video for all frames, and applies this motion to the control points on the input sketch. Unlike previous methods that require skeletons, control points, or reference videos, our approach generates high-quality, non-rigid, smooth sketch deformations using only text prompts without manual user input.

2.2 Image and Text-to-Video Generation

Text-to-video generation aims to produce the corresponding video using text prompt input automatically. Previous works have discovered the ability of GANs (Tian et al., 2021; Zhu et al., 2023; Li et al., 2018) and auto-regressive transformers (Wu et al.,

2021; Yan et al., 2021) for video generation, but these are restricted to the fixed domain. Recent progress in diffusion models sets an enrichment in the video generation methodology. Recent methods such as (Wang et al., 2023; Chen et al., 2023a; Guo et al., 2023; Chen et al., 2023c; Zhou et al., 2022) utilize Stable Diffusion (Ni et al., 2023) to incorporate temporal information in latent space. Dynamicrafter (Xing et al., 2023) generates videos from input images and text prompts. Despite advancements, open-source video generation faces challenges in maintaining text readability during motion. LiveSketch (Gal et al., 2023) animates vector sketches that do not require extensive training. It uses a pre-trained text-to-video diffusion model to utilize motion and instruct the motion to sketch using SDS (Poole et al., 2022). Similar to text-to-video generation and image-to-video generation, a closed research area aims to generate video from an input image. Latent Motion Diffusion (Hu et al., 2023) estimates the motion by learning the optical flow from video frames and uses the 3D-UNet diffusion model to generate the animated video. Make-It-Move (Hu et al., 2022) uses an encoder-decoder network condition on image and text prompt input to generate the video sequences. VideoCrafter1 (Chen et al., 2023a) and LivePhoto (Chen et al., 2023c) preserve the input image style and structure by training to be conditioned on text and image input. CoDi (Tang et al., 2024) trained on shared latent space with conditioning and output space and aligned modalities such as image, video, text, and audio. However, these approaches struggle to preserve the characteristics of the vectorized input sketch.

DreamFusion (Poole et al., 2022) proposed SDS loss that generates 3D representations from text input using 2D image diffusion. The SDS loss is similar to diffusion model loss. However, it does not include the U-Net jacobian, which helps it overcome the high computation time of backpropagation within the diffusion model and aligns the image per the text condition by guiding the optimization process. SDS loss is also used to optimize the other generative tasks such as sketches (Xing et al., 2024), vector graphics (Jain et al., 2023), and meshes (Chen et al., 2023b). The diffusion network predicts the sketch points' position for each frame and aligns the entire animation with the text prompt using SDS loss.

3 METHODOLOGY

Our methodology extends the framework introduced by Gal et al. (Gal et al., 2023), which produces animations from sketches guided by textual descriptions.

Each sketch consists of a set of strokes, represented as cubic Bézier curves. We represent the set of control points within a frame as $B = \{\mathbf{p}_i\}_{i=1}^{4k}$ where $\mathbf{p}_i \in \mathbb{R}^2$ and k is the total number of strokes. Further, we define a sketch video of n frames by a set of moving control points $Z = \{B_i\}_{i=1}^{n}$, where $Z \in \mathbb{R}^{4k \times n \times 2}$.

Animation of a sketch requires the user to provide a text prompt passed into the network as input along with the sketch. Similar to LiveSketch (Gal et al., 2023), we use a neural network architecture that takes an initial set of control points, Z^{init}, as input and produces the corresponding set of displacements, ΔZ. For each frame, Z^{init} is initialized to the set B. Each control point is first projected onto a latent space using a mapping function, $g_{shared} : \mathbb{R}^2 \to \mathbb{R}^{\mathbb{D}}$. This function takes the initial point set $Z^{init} \in \mathbb{R}^2$ and projects it into a higher-dimensional space enriched with positional encoding, thereby generating point features. These features are processed through two branches: a local motion predictor, $\mathcal{M}_l$, implemented as a multi-layer perceptron (MLP) F_θ, which computes unconstrained local motion offsets, and a global motion predictor, $\mathcal{M}_g$, which estimates transformation matrices $\mathbf{M}_i$ for scaling, shear, rotation, and translation, yielding the global motion offsets T_i. The generated animation sequence suffers from a lack of temporal consistency and degradation of sketch identity during motion. We propose a novel Length-Area (LA) regularization framework to significantly enhance temporal coherence in animated sequences. Our approach estimates the Bézier curve length and the area between consecutive frames. Furthermore, we introduce a shape-preserving As-Rigid-As-Possible (ARAP) loss, leveraging a mesh constructed via Delaunay triangulation (Delaunay, 1934) of control points within each frame. Unlike existing methods, our ARAP loss is explicitly designed to maintain local shape consistency, addressing critical challenges in deformation handling and ensuring robust animation fidelity. Figure 2 provides a detailed illustration of our proposed network architecture, highlighting its key components. To evaluate its performance, we experimented with different learning rate configurations and conducted multiple iterations of the optimization process, systematically refining the model.

3.1 Regularization

The LA regularizer is designed to minimize abrupt changes in stroke lengths between consecutive frames, ensuring smoother transitions and preserving structural consistency by maintaining stable stroke lengths across the animation. To mitigate error propa-
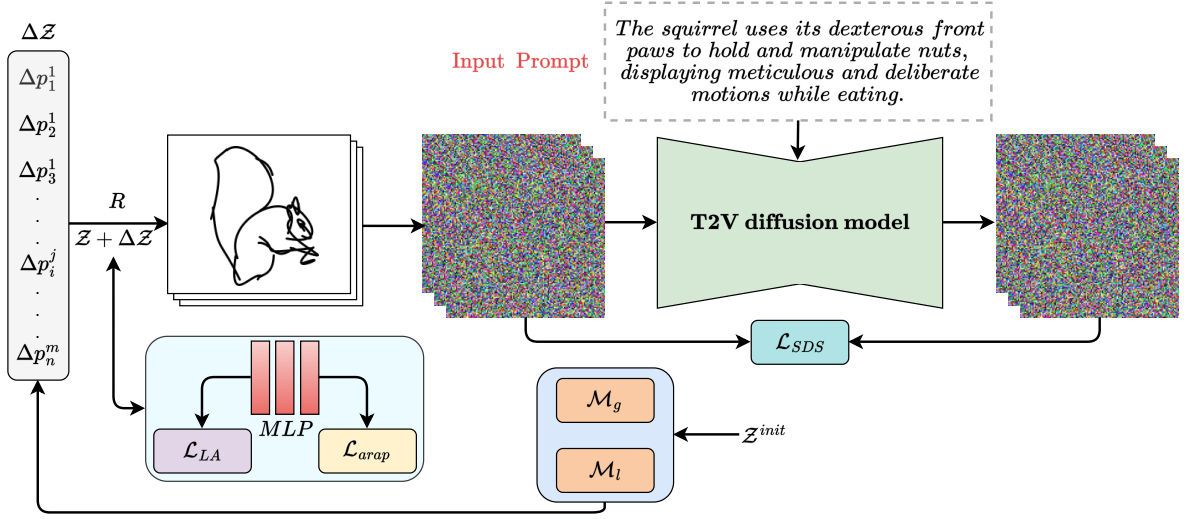
Figure 2: Network architecture of our proposed framework. We use a Length-Area loss to maintain temporal consistency and avoid drastic shape changes. Further ARAP loss maintains the sketch stroke's rigidity and prevents shape distortions during motion.

gation, the length minimization for a stroke in a given frame is computed relative to its length in the initial frame. The stroke length is estimated as the curve length, $L = \int_0^1 |\dot{\mathbf{f}}(u)| \, du$ of the Bézier curve $\mathbf{f}(u)$.

Bézier curves lack local control, meaning that even minor adjustments to control point positions can lead to significant changes in the resulting curve. To mitigate this issue, we introduce an area loss term that minimizes the area spanned by a stroke between consecutive frames, thereby enhancing temporal stability and reducing undesirable deformations. To compute this area, we consider a stroke represented by the Bézier curve $\mathbf{f}_i(u)$ defined by the control points $\mathbf{p}_{i,j}$, where $j = \{0 \ldots 3\}$, for an intermediate frame i. Let the estimated global transformation matrix of control points for frame i be denoted as $\mathbf{M}_i$, and the corresponding local motion offsets as $\Delta \mathbf{p}_{i,j}$. The control points for the next frame are determined as $\mathbf{p}_{i+1,j} = \mathbf{M}_i \mathbf{p}_{i,j} + \Delta \mathbf{p}_{i,j}$. The space-time Bézier surface $\mathbf{f}(u,t)$ for $t \in [t_i, t_{i+1}]$ is defined by time-varying control points $\mathbf{p}_j(t) = \mathbf{M}(t)\mathbf{p}_{i,j} + \Delta \mathbf{p}_{i,j}(t - t_i)/(t_{i+1} - t_i)$, where $\mathbf{M}(t)$ is obtained by interpolating the transformation parameters appropriately over time. The surface area swept by the stroke between frames i and $i+1$ is computed as

$$A_i = \int_{t_i}^{t_{i+1}} \int_0^1 \left\| \frac{\partial \mathbf{f}}{\partial u} \times \frac{\partial \mathbf{f}}{\partial t} \right\| du \, dt. \tag{1}$$

The LA regularization, denoted as $\mathcal{L}_{LA}$ is defined as

$$\mathcal{L}_{LA} = \sum_{i=0}^{n-1} (\lambda_l |L_{i+1} - L_i| + \lambda_a A_i), \tag{2}$$

where $\mathcal{L}_{LA}$ represents the length-area loss function. This formulation aims to minimize both the variation in stroke length and the swept area between consecutive frames, ensuring temporal coherence and stability in animation. We use a multilayer perceptron (MLP) to optimize this loss, with values of hyperparameters λ_l and λ_a set to 0.1 and $1e-5$, respectively.

LiveSketch (Gal et al., 2023) uses the SDS loss to train its model, which has separate blocks for optimizing the global and local motion. The SDS loss is defined as

$$\triangledown_\phi \mathcal{L}_{sds} = \left[w(\gamma)(\varepsilon_\theta(x_\gamma, \gamma, y) - \varepsilon) \frac{\delta x}{\delta \phi} \right], \tag{3}$$

where $\varepsilon_\theta(x_\gamma, \gamma, y)$ is the output of the diffusion model, ε denotes the actual noise, γ represents the timestep, and $w(\gamma)$ is a constant term that depends on the noising schedule. The SDS loss is calculated at every step of the diffusion generation process for all frames, guiding the training of these blocks and the overall generation process. During each generation step, the optimization occurs after completing the SDS loss-based optimization for both blocks of the base model, resulting in updated control points. These control points are used as input for our optimization procedure.
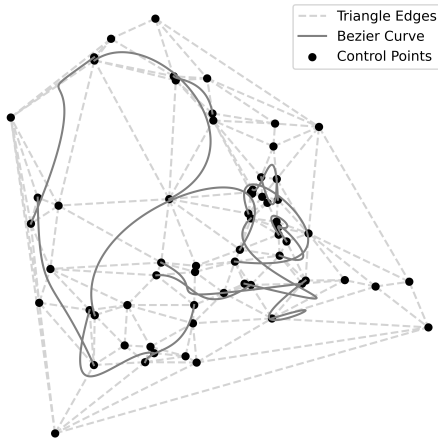
Figure 3: Cubic Bézier curve of each stroke and their corresponding control points. Delaunay triangulation of Bézier control points.

3.2 Shape Preservation

As-rigid-as-possible deformation enables point-driven shape deformation by moving anchor points, which act as constraints within the model. This deformation framework maintains the rigidity of each element of the mesh as closely as possible, ensuring that transformations are smooth and visually coherent. The ARAP method leverages a two-step optimization algorithm. In the first step, an initial rotation is estimated for each triangle in the mesh. This involves computing the optimal rotation matrix that best approximates the transformation required to map the vertices of each triangle from their initial positions to their target positions while minimizing distortion. The second step involves adjusting the scale, ensuring that the transformation adheres to an as-rigid-as-possible model by minimizing the amount of stretch that would distort the original shape. The approach minimizes distortion across the triangular mesh by optimizing each triangle's local transformations while maintaining global consistency across the mesh.

In our proposed approach, we extend the standard ARAP loss (Igarashi et al., 2005) by formulating it as a differentiable function, enabling the use of gradient-based optimization techniques and backpropagation within the network. This differentiable ARAP loss is optimized using a multilayer perceptron (MLP), allowing adaptive and flexible shape deformation.

The ARAP loss is computed based on a global mesh structure formed by triangulating Bézier control points (see Figure 3) within each frame. Calculating the ARAP loss relative to a similar triangulated mesh for the next frame ensures stroke preservation, which

is essential for generating smooth and consistent animations. The ARAP loss $\mathcal{L}_{ARAP}$ is computed by identifying all triangles in the mesh formed by the control points of a given frame, with the triangulation topology $\mathcal{T}$ remaining fixed across all frames. The same is defined as

$$\mathcal{L}_{ARAP} = \sum_{e \in \mathcal{T}} \alpha_e \left\| e' - \mathbf{D}e \right\|^2, \qquad (4)$$

where $\mathbf{D}$ is the ARAP transformation matrix, e represents an edge of a triangle, estimated from the control points of the initial sketch, and e' denotes the corresponding deformed edge of the triangle of the subsequent frames. α_e denotes the weight, usually proportional to the edge length. The ARAP loss in equation 4 is calculated by first identifying the triangles that form the mesh of the given frame. These triangles are used to compute the transformation matrix, which is then optimized using a multi-layer perceptron (MLP).

4 EXPERIMENTS AND RESULTS

4.1 Implementation Details

We use a text-to-video diffusion model (Wang et al., 2023) similar to the approach in LiveSketch (Gal et al., 2023), to generate the required motion in pixel space. Further, we use the generated frames to apply the SDS loss training for a timestep to find the updated control points. These updated control points are then further optimized using our learning procedure. It takes the post-LiveSketch updated control points of the current frame as input and outputs the optimized control points. We train the MLP for 1000 iterations of the LiveSketch model. We have used t=1000 to estimate the Bézier curves and find their length and area. We use the values of λ_l, λ_a and λ_{arap} as 0.1, $1e-5$, and 0.1 respectively. Further, we use similar parameters for the local and global paths given by LiveSketch (Gal et al., 2023). Our method takes approximately 2 hours to generate a sequence of 24 animated sketches.

4.2 Results and Comparison

4.2.1 Quantitative Evaluation

We compare our approach with previous baseline methods VideoCrafter1 (Chen et al., 2023a), and LiveSketch (Gal et al., 2023). We use sketch-to-video consistency and Text-to-video alignment as evaluation matrices similar to LiveSketch (Gal et al.,
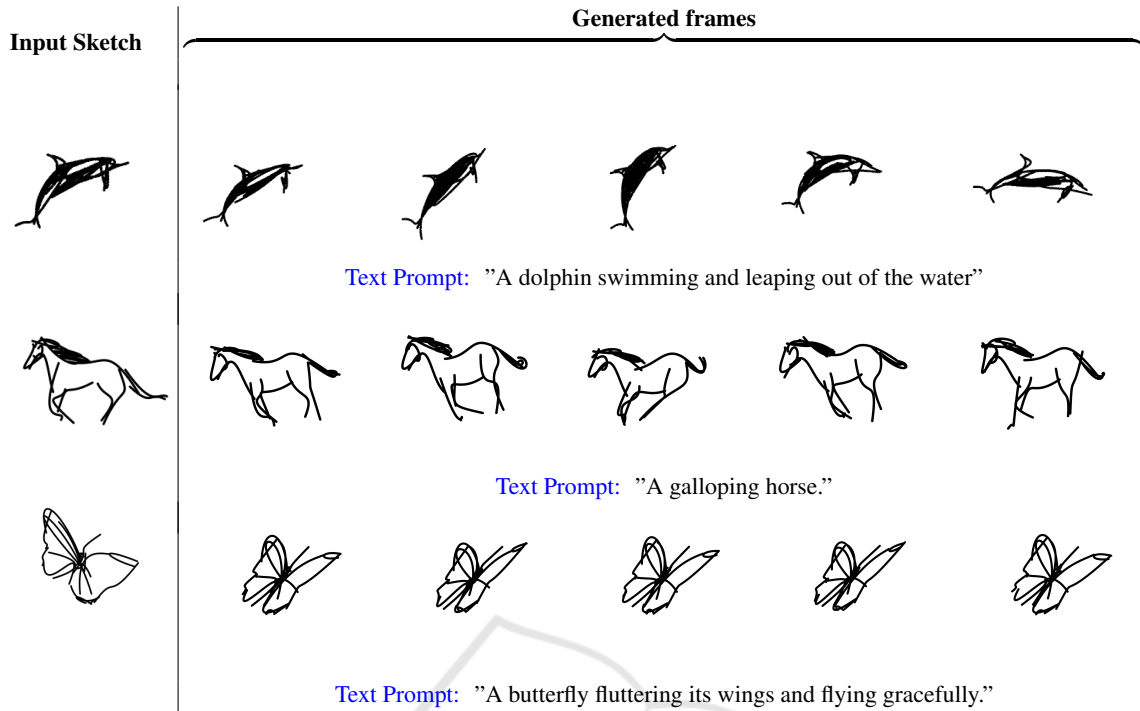
Figure 4: Qualitative results of our proposed method and generated animated sketch sequences from input text prompts.

Table 1: Comparison with state-of-the-art methods.

	Sketch-to-video consistency ($\uparrow$)	Text-to-video alignment ($\uparrow$)
VideoCrafter1 (Chen et al., 2023a)	0.7064	0.0876
LiveSketch (Gal et al., 2023)	0.8287	0.1852
Ours	**0.8561**	**0.1893**

2023) that use CLIP (Radford et al., 2021) to estimate sketch-to-video consistency and X-CLIP (Ni et al., 2022) for text-to-video alignment. We used 20 unique sketch samples and text for the quantitative evaluation. VideoCrafter1 (Chen et al., 2023a) is an image-to-video generation model and the conditions on image and text prompts. Table 1 depicts that our method outperforms the quantitative analysis compared to previous methods. We maintain the text-to-video alignment similar to LiveSketch (Gal et al., 2023), but the sketch-to-video consistency performance is superior to our method.

4.2.2 Qualitative Evaluation

In the qualitative comparison, we measure the sketch-to-video consistency and Text-to-video alignment. In the streamline, we further estimate the improvements such as temporal consistency and shape preservation (see Figure 4). Sketch-to-video consistency describes the temporal consistency of the generated sketch sequences. Figure 5 shows that the bottom of the wine glass and squirrel is temporally consistent in all

the frames compared to VideoCrafter1 (Chen et al., 2023a), and LiveSketch (Gal et al., 2023). On the other hand, we observe that the surfer and squirrel examples preserve the original shape during animation. The mesh-based rigidity loss helps to produce a smooth deformation compared to the baseline methods. Our method maintains temporal consistency, preserves shape during animation, and outperforms baseline methods.

4.3 Ablation Study

4.3.1 With and Without Regularization

We evaluate our method without LA regularization and observe that it fails to maintain temporal consistency. LA regularization helps to address the issue of drastic changes in stroke. In Figure 6, the lizard's tail and legs move rapidly, and the stroke length varies excessively. In our proposed method, we can see the smooth motion and nominal change in stroke length. Table 2 shows the quantitative results w/o the LA regularizer and our method with the LA regularization.
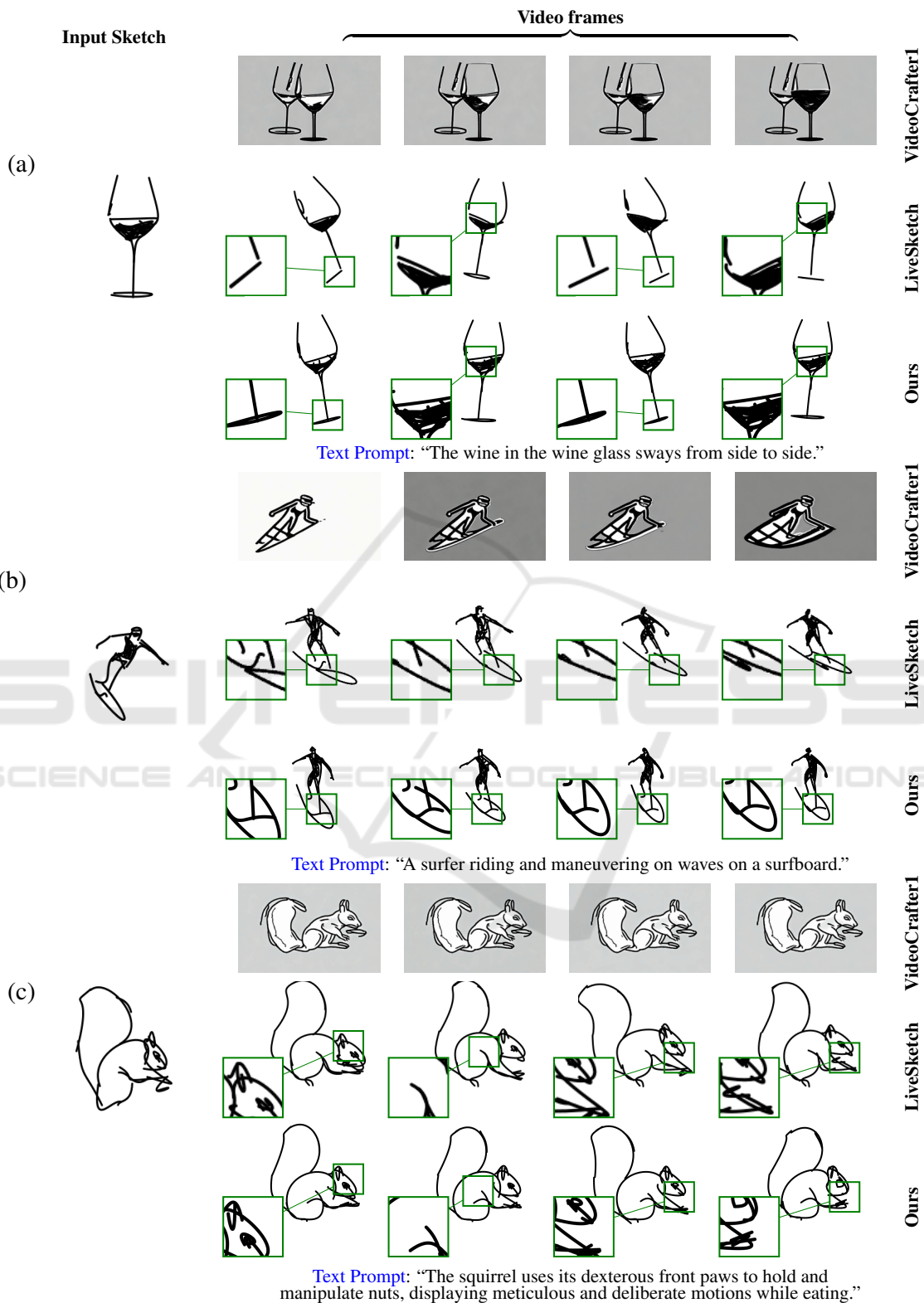
Figure 5: Comparison with state-of-the-art methods (VideoCrafter1 (Chen et al., 2023a) and LiveSketch (Gal et al., 2023)) . In the above figure, the base of the wine glass is distorted in the previous methods, and in the surfer base, the original shape is missing compared to ours. The squirrel tails and body shape contain the original topology in our method.
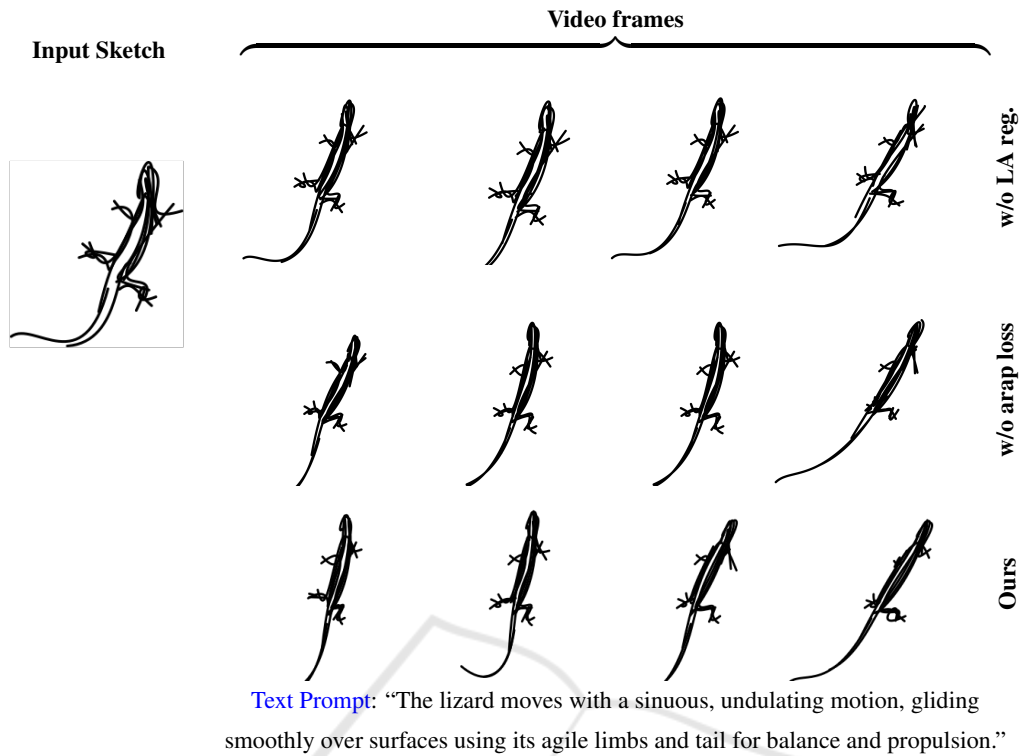
Video frames

Input Sketch

w/o LA reg.

w/o arap loss

Ours

Text Prompt: "The lizard moves with a sinuous, undulating motion, gliding smoothly over surfaces using its agile limbs and tail for balance and propulsion."

Figure 6: Ablation study on different setting such as w/o LA regularization, w/o shape preserving ARAP, and our full method.

4.3.2 With and Without Shape-Preserving ARAP

We evaluate the method without shape preservation and observe shape distortion during animation. The animated sketch shows distortion when local motion increases, as topology is not preserved in the animated sketch video. In Figure 6, the lizard body distorts during the motion, compared to our method with shape-preserving arap loss. Quantitatively (see Table 2), the performance without shape preservation is comparable, but our complete method gives better results.

5 LIMITATIONS

Our method relies on a pre-trained text-to-video prior (Zhu et al., 2023), which may struggle with certain types of motion, leading to errors that propagate and manifest as noticeable artifacts in some cases of the generated animations. Improvements could be made by employing more advanced text-to-video priors capable of handling text-to-video alignment with higher accuracy. Additionally, our approach faces challenges in animating multi-object scenarios, particularly when functional relationships exist between objects. Designed primarily for single-object anima-

Input Sketch　　**Ours**

Text Prompt: "The two dancers are passionately dancing the Cha-Cha, their bodies moving in sync with the infectious Latin rhythm."

Text Prompt: "The biker is pedaling, each leg pumping up and down as the wheels of the bicycle spin rapidly, propelling them forward."

Figure 7: Failure cases.

tions, the method experiences a decline in quality when dealing with such cases. For example, as shown

Table 2: Ablation results—the quantitative evaluation on w/o LA regularization, w/o shape preserving ARAP, and our proposed method.

	Sketch-to-video consistency ($\uparrow$)	Text-to-video alignment ($\uparrow$)
W/o LA reg.	0.8306	0.1864
W/o Shape-preserving ARAP	0.8489	0.1891
Ours	**0.8561**	**0.1893**

in Figure 7, the human and the bicycle are incorrectly separated, resulting in unnatural motion during the animation. Future work could address this limitation by implementing object-specific translations rather than relying on relative motion.

6 CONCLUSION

This work presents a method for generating animated sketches from a combination of sketch inputs and text prompts. To ensure temporal consistency in the animations, we introduce a Length-Area (LA) regularizer, and to preserve the original shape's topology, we propose a shape-preserving ARAP loss. Our approach delivers superior performance both quantitatively and qualitatively, addressing challenges in animation generation. However, the method has certain limitations, including its inability to handle multi-object scenarios and its reliance on a pre-trained text-to-video prior. Future work will focus on addressing these limitations to further enhance the method's capabilities.

ACKNOWLEDGEMENTS

REFERENCES

Agarwala, A., Hertzmann, A., Salesin, D. H., and Seitz, S. M. (2004). Keyframe-based tracking for rotoscoping and animation. *ACM Transactions on Graphics (ToG)*, 23(3):584–591.

Bregler, C., Loeb, L., Chuang, E., and Deshpande, H. (2002). Turning to the masters: Motion capturing cartoons. *ACM Transactions on Graphics (TOG)*, 21(3):399–407.

Chen, H., Xia, M., He, Y., Zhang, Y., Cun, X., Yang, S., Xing, J., Liu, Y., Chen, Q., Wang, X., et al. (2023a). Videocrafter1: Open diffusion models for high-quality video generation. *arXiv preprint arXiv:2310.19512*.

Chen, R., Chen, Y., Jiao, N., and Jia, K. (2023b). Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22246–22256.

Chen, X., Liu, Z., Chen, M., Feng, Y., Liu, Y., Shen, Y., and Zhao, H. (2023c). Livephoto: Real image animation with text-guided motion control. *arXiv preprint arXiv:2312.02928*.

Delaunay, B. (1934). Sur la sphère vide. *Bulletin de l'Académie des Sciences de l'URSS: Classe des Sciences Mathématiques et Naturelles*, 6:793–800.

Gal, R., Vinker, Y., Alaluf, Y., Bermano, A. H., Cohen-Or, D., Shamir, A., and Chechik, G. (2023). Breathing life into sketches using text-to-video priors. *arXiv preprint arXiv:2311.13608*.

Guay, M., Ronfard, R., Gleicher, M., and Cani, M.-P. (2015). Space-time sketching of character animation. *ACM Transactions on Graphics (ToG)*, 34(4):1–10.

Guo, Y., Yang, C., Rao, A., Wang, Y., Qiao, Y., Lin, D., and Dai, B. (2023). Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. *arXiv preprint arXiv:2307.04725*.

Hinz, T., Fisher, M., Wang, O., Shechtman, E., and Wermter, S. (2022). Charactergan: Few-shot keypoint character animation and reposing. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1988–1997.

Hu, Y., Chen, Z., and Luo, C. (2023). Lamd: Latent motion diffusion for video generation. *arXiv preprint arXiv:2304.11603*.

Hu, Y., Luo, C., and Chen, Z. (2022). Make it move: controllable image-to-video generation with text descriptions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18219–18228.

Igarashi, T., Moscovich, T., and Hughes, J. F. (2005). As-rigid-as-possible shape manipulation. *ACM transactions on Graphics (TOG)*, 24(3):1134–1141.

Jain, A., Xie, A., and Abbeel, P. (2023). Vectorfusion: Text-to-svg by abstracting pixel-based diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1911–1920.

Jeruzalski, T., Levin, D. I., Jacobson, A., Lalonde, P., Norouzi, M., and Tagliasacchi, A. (2020). Nilbs: Neural inverse linear blend skinning. *arXiv preprint arXiv:2004.05980*.

Li, Y., Min, M., Shen, D., Carlson, D., and Carin, L. (2018). Video generation from text. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

Liu, L., Zheng, Y., Tang, D., Yuan, Y., Fan, C., and Zhou, K. (2019). Neuroskinning: Automatic skin binding for production characters with deep graph networks. *ACM Transactions on Graphics (ToG)*, 38(4):1–12.

Mallya, A., Wang, T.-C., and Liu, M.-Y. (2022). Implicit warping for animation with image sets. *Advances in Neural Information Processing Systems*, 35:22438–22450.

Ni, B., Peng, H., Chen, M., Zhang, S., Meng, G., Fu, J., Xiang, S., and Ling, H. (2022). Expanding language-image pretrained models for general video recognition. In *European Conference on Computer Vision*, pages 1–18. Springer.

Ni, H., Shi, C., Li, K., Huang, S. X., and Min, M. R. (2023). Conditional image-to-video generation with latent flow diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18444–18455.

Patel, P., Gupta, H., and Chaudhuri, P. (2016). TraceMove: A data-assisted interface for sketching 2D character animation. In *VISIGRAPP (1: GRAPP)*, pages 191–199.

Poole, B., Jain, A., Barron, J. T., and Mildenhall, B. (2022). DreamFusion: Text-to-3D using 2D diffusion. *arXiv*.

Poursaeed, O., Kim, V., Shechtman, E., Saito, J., and Belongie, S. (2020). Neural puppet: Generative layered cartoon characters. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3346–3356.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. (2021). Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR.

Rai, G., Gupta, S., and Sharma, O. (2024). Sketchanim: Real-time sketch animation transfer from videos. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 1–11.

Santosa, S., Chevalier, F., Balakrishnan, R., and Singh, K. (2013). Direct space-time trajectory control for visual media editing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1149–1158.

Siarohin, A., Lathuilière, S., Tulyakov, S., Ricci, E., and Sebe, N. (2019). First order motion model for image animation. *Advances in Neural Information Processing Systems*, 32.

Siarohin, A., Woodford, O. J., Ren, J., Chai, M., and Tulyakov, S. (2021). Motion representations for articulated animation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13653–13662.

Smith, H. J., Zheng, Q., Li, Y., Jain, S., and Hodgins, J. K. (2023). A method for animating children's drawings of the human figure. *ACM Transactions on Graphics*, 42(3):1–15.

Su, Q., Bai, X., Fu, H., Tai, C.-L., and Wang, J. (2018). Live sketch: Video-driven dynamic deformation of static drawings. In *Proceedings of the 2018 chi conference on human factors in computing systems*, pages 1–12.

Tang, Z., Yang, Z., Zhu, C., Zeng, M., and Bansal, M. (2024). Any-to-any generation via composable diffusion. *Advances in Neural Information Processing Systems*, 36.

Tanveer, M., Wang, Y., Wang, R., Zhao, N., Mahdavi-Amiri, A., and Zhang, H. (2024). Anamodiff: 2d analogical motion diffusion via disentangled denoising. *arXiv preprint arXiv:2402.03549*.

Tao, J., Wang, B., Xu, B., Ge, T., Jiang, Y., Li, W., and Duan, L. (2022). Structure-aware motion transfer with deformable anchor model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3637–3646.

Tian, Y., Ren, J., Chai, M., Olszewski, K., Peng, X., Metaxas, D. N., and Tulyakov, S. (2021). A good image generator is what you need for high-resolution video synthesis. *arXiv preprint arXiv:2104.15069*.

Wang, J., Xu, Y., Shum, H.-Y., and Cohen, M. F. (2004). Video tooning. In *ACM SIGGRAPH 2004 Papers*, pages 574–583.

Wang, J., Yuan, H., Chen, D., Zhang, Y., Wang, X., and Zhang, S. (2023). Modelscope text-to-video technical report. *arXiv preprint arXiv:2308.06571*.

Wang, Y., Yang, D., Bremond, F., and Dantcheva, A. (2022). Latent image animator: Learning to animate images via latent space navigation. *arXiv preprint arXiv:2203.09043*.

Wu, C., Huang, L., Zhang, Q., Li, B., Ji, L., Yang, F., Sapiro, G., and Duan, N. (2021). Godiva: Generating open-domain videos from natural descriptions. *arXiv preprint arXiv:2104.14806*.

Xing, J., Wei, L.-Y., Shiratori, T., and Yatani, K. (2015). Autocomplete hand-drawn animations. *ACM Transactions on Graphics (TOG)*, 34(6):1–11.

Xing, J., Xia, M., Zhang, Y., Chen, H., Wang, X., Wong, T.-T., and Shan, Y. (2023). Dynamicrafter: Animating open-domain images with video diffusion priors. *arXiv preprint arXiv:2310.12190*.

Xing, X., Wang, C., Zhou, H., Zhang, J., Yu, Q., and Xu, D. (2024). Diffsketcher: Text guided vector sketch synthesis through latent diffusion models. *Advances in Neural Information Processing Systems*, 36.

Xu, Z., Zhou, Y., Kalogerakis, E., Landreth, C., and Singh, K. (2020). Rignet: Neural rigging for articulated characters. *arXiv preprint arXiv:2005.00559*.

Yan, W., Zhang, Y., Abbeel, P., and Srinivas, A. (2021). Videogpt: Video generation using vq-vae and transformers. *arXiv preprint arXiv:2104.10157*.

Zhao, J. and Zhang, H. (2022). Thin-plate spline motion model for image animation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3657–3666.

Zhou, D., Wang, W., Yan, H., Lv, W., Zhu, Y., and Feng, J. (2022). Magicvideo: Efficient video generation with latent diffusion models. *arXiv preprint arXiv:2211.11018*.

Zhu, J., Ma, H., Chen, J., and Yuan, J. (2023). Motionvideogan: A novel video generator based on the motion space learned from image pairs. *IEEE Transactions on Multimedia*.