

# Breathing Life Into Sketches Using Text-to-Video Priors

## Supplementary Material

### Contents

<b>1. Additional results and videos</b>	<b>1</b>
<b>2. Analysis and ablation</b>	<b>1</b>
2.1. Text prompt effect	1
2.2. Different levels of abstraction	2
2.3. Sketch representation	2
2.4. Learning rate scaling and tradeoffs	2
2.5. Hyperparameter effects	3
2.6. Other text-to-video backbones	3
<b>3. Implementation and technical details</b>	<b>3</b>
3.1. Sketch generation	3
3.2. Additional training details	4
3.3. Evaluation details	4
3.3.1 Baseline implementations	4
3.3.2 Evaluation metrics	4
3.3.3 Evaluation data	4
3.3.4 User Study	4
<b>4. Acknowledgements</b>	<b>5</b>

### 1. Additional results and videos

All videos and a large number of additional results are available in our supplementary project page: <https://livesketch.github.io/>.

These include an array of subjects animated with our method, along with additional comparisons, ablation experiments and visualizations of limitations. Please note that all comparisons and ablation baseline results use our default parameters, while the large video gallery includes results with different parameter settings, chosen according to our aesthetic preferences.

### 2. Analysis and ablation

In this section we present an array of experiments that explore the sensitivity of our method to different hyperparameters of the approach. These include technical changes (such as learning rate adjustments), but also conceptual explorations such as the effect of sketch abstraction on the generated videos.

#### 2.1. Text prompt effect

Our animation process is guided by a user-provided text, based on the prior of a pretrained text-to-video model. This section further examines how the specified prompt affects the animation. We first verify that the text itself influences

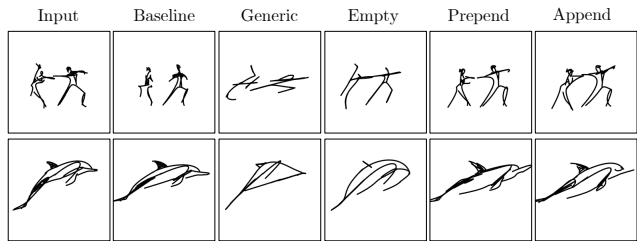


Figure 1. Text prompt effect. We investigate the effects of using a generic prompt (“The object is moving”) for all sketches, the effect of using an empty prompt, or prepending and appending strings that compel the diffusion model go generate sketches. Additional video results are shown in the website.

the results in a meaningful way. To do so, we apply our method to several example sketches, using two alternatives: A “generic” prompt (“the object is moving”), and the empty prompt (“”). The results are shown in Fig. 1 and in the “Text Prompt Effect” section of the website. Using the generic prompt leads to irrelevant animations in which both the motion and the sketch appearance exhibit significant artifacts. Using an empty prompt leads to results with no visible motion, and large shape deviations. We can thus conclude that using prompts tailored for the input sketch is crucial, both to preserve its characteristics and for the ability to generate meaningful motion.

We further examine the impact of modifying the prompt in a way that would motivate the text-to-video to create a sketch. Specifically, we either prepend the string “A sketch of” or append the string “Abstract sketch. Line drawing” to the prompts.

In general, explicitly prompting for a sketch works comparably well to the original prompts. In some cases we observe slight differences in the extent of the motion or in the adherence to slight details in the input sketch (*e.g.* the penguin’s left fin is filled out when using the sketch prompts). However, these can likely be accounted for with learning rate tuning. We thus conclude that the model can reasonably infer the semantics of the object even when the prompt does not directly convey its sketch-based nature.

Finally, we show additional results for applying different prompts to the same input sketch (see “Varying the Prompt” in the provided website). For example, observe how the boxer changes his motion in accordance with the texts provided, demonstrating the actions of jumping, running, and punching. Similarly, a cat can be made to change its pose, or walk towards the camera. However, in some cases the method is not sensitive enough to the changes in the pro-

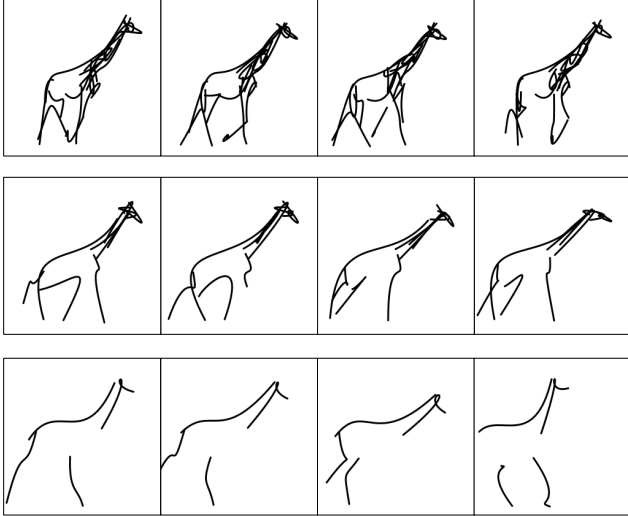


Figure 2. Different levels of abstraction. We show four selected frames for each level of abstraction. The model can successfully synthesize movement even for very abstract representations.

vided text prompt. This is particularly apparent when the prompt requests large changes in the shape of the subject, or when the diffusion model struggles to generate the described motion even in its basic text-to-video setup. In the video website, we demonstrate this on the ballerina sketch, where the specifics of the prompts are largely ignored, leading to similar dancing motions. However, notice that supplying the base diffusion model with those same prompts, also creates videos with dancing that is unrelated to the motion described in the prompt. We hope that this limitation could be overcome as better, more expressive text-to-video models become available.

## 2.2. Different levels of abstraction

We also demonstrate the effect of altering the abstraction level of the input sketches. We show results for three objects with three levels of abstraction. The sketches were generated using 16, 8, and 4 strokes. An example is provided in Fig. 2, and more examples and the full videos are provided in the supplementary website’s “Abstraction Level” section. As can be seen, even for the extreme case of very abstract sketches with only four strokes, our method still manages to produce animations that fit the given prompt. Yet, the abstract animations may appear less smooth, leaving room for future work to tackle such challenging cases.

## 2.3. Sketch representation

As described in the main paper, we represent a sketch as a set of black cubic Bezier curves, and use CLIPasso [7] to automatically generate the sketches shown in the paper. However, our approach can be applied to alternative sketch representations. As highlighted in the limitations section of the

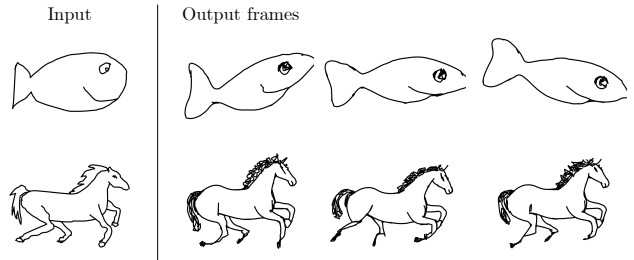


Figure 3. Human-drawn sketches. We applied our method to sketches from the TU-Berlin dataset. With our default parameters, these create reasonable motion but fail to preserve the exact sketch appearance. By tuning the parameters for this input style, shape preservation can be improved. See the website for examples.

main paper, employing different sketch representations may require additional hyperparameter tuning. To illustrate the impact of changing the sketch representation, we applied our method to sketches from the TU-Berlin sketch dataset [1], a human-drawn class-based sketch dataset. We showcase the results of four representative sketches. Our method was directly applied to the provided SVG files. Fig. 3 shows a few representative frames from the videos produced for two sketches. More results are shown in the supplementary website. As can be seen, our method successfully animated the sketches, however their appearance is not fully preserved when using the default hyperparameters. This can be improved by using lower learning rates for the local path.

Next, we selected 100 random animal sketches from the set and animated them using motion prompts generated with ChatGPT. We evaluated the results with the core paper’s quantitative metrics using our method and the leading competitor (VideoCrafter). The results are provided in Tab. 1. Here too our method outperforms the competition by significant margins, showing that we can provide improvements on multiple sketch styles.

Table 1. CLIP-based consistency and text-video alignment comparisons to VideoCrafter on our 100-animal subset of the TU-Berlin sketch dataset [1]

Method	Sketch-to-video consistency ( $\uparrow$ )	Text-to-Video alignment ( $\uparrow$ )
VideoCrafter	0.886 $\pm$ 0.008	0.118 $\pm$ 0.004
Ours	<b>0.949</b> $\pm$ 0.003	<b>0.139</b> $\pm$ 0.003

## 2.4. Learning rate scaling and tradeoffs

As discussed in the main paper, there exists a trade-off between the quality of generated motion and the capacity to retain the appearance of the initial sketch. To illustrate this trade-off, we conducted an experiment wherein we ran-

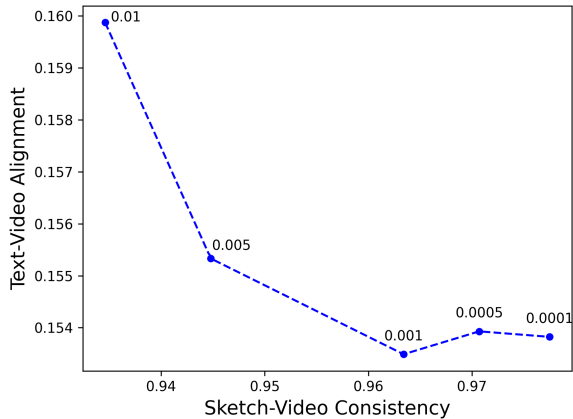


Figure 4. Investigation of the tradeoff between motion quality and sketch preservation. Increasing the local learning rates trades one aspect for another.

domly selected three sketches from each class in our evaluation set (9 sketches in total). We then tested the impact of scaling the local learning rate within the range of 0.01 to 0.0001, keeping all parameters constant except for the local learning rate. Qualitative results are shown in the website, under the “Trade-off” section. Observe that as we move from the left (0.0001) to right (0.1), the motion in the animations increases, better aligning with the text prompt. However, this comes at the cost of preserving the original sketch’s appearance. For example, observe how the fish and the crab undergo complete transformations when using a learning rate greater than 0.001. This trade-off introduces additional control for the user, who may prioritize stronger motion over sketch fidelity.

Furthermore, we assess the results using CLIP-based metrics (Fig. 4). As can be observed, increasing the learning rate leads to a smooth tradeoff between motion quality and sketch preservation. Working with learning rates  $\in [0.001, 0.005]$  generally leads to a good compromise between the two aspects - though a user can choose a different working point according to their preferences.

## 2.5. Hyperparameter effects

We demonstrate how changing different hyperparameters in our method can provide the user with additional control (see “Hyperparameter Effects” in the website). We observe different effects across various sketches, which may be attributed to the video model’s prior or the initial sketch quality. Specifically, in the third column (“+lr local”), we showcase the impact of increasing the learning rate of the local path. As evident, in some cases (biking and butterfly), this improved the generated motion without significantly harming the sketch’s appearance. However, in other cases (cobra

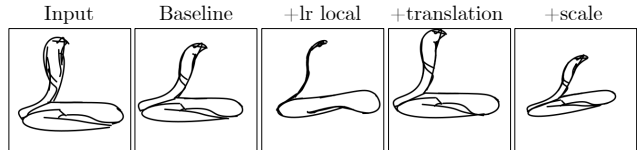


Figure 5. Hyperparameter effect. We show one representative frame from each video (the full videos and additional examples are provided in the website).

and boat), increasing the local path’s learning rate leads to a complete alteration of the original sketch. In the fourth and fifth columns we show the effect of increasing the translation and scale prediction weights. As observed, this indeed causes the objects to move more across the frame or change their scale.

## 2.6. Other text-to-video backbones

We investigate the performance of the model when we swap one text-to-video prior for another. In the main paper, we use ModelScope [8] as our text-to-video diffusion backbone. Here, we qualitatively evaluate the effect of replacing it with other text-to-video models. In particular, we look at a set of ZeroScope models, tuned across a range of resolutions and framerates. The results are shown in the supplementary videos (website section “Comparing Video Models”). Two representative examples are provided in Fig. 6. Our method generalizes to these models with no additional changes. However, note that different models do lead to different motion patterns, and some of them may result in different tradeoffs between the level of motion and the ability to preserve the sketch. For example, observe the cat (second row) which either wags their tail, raises its front legs, or does both, depending on the model. For some models (e.g. zeroscope v1-1 320s) the cat appears more deformed, and a user may prefer to use another working point on the local learning-rate axis in order to restore the shape.

## 3. Implementation and technical details

Here we outline additional details required to reproduce our work and experiments. We will release all code and image sets used for evaluations to facilitate further research and comparisons.

### 3.1. Sketch generation

Unless otherwise noted, all sketches presented in the main paper and the supplementary material were generated using CLIPasso [7]. CLIPasso is a method for automatically generating object sketches represented with cubic Bezier curves. In the majority of examples, we applied CLIPasso with the default settings, using 16 strokes. The sketch’s canvas size is  $256 \times 256$ , and the strokes width is 1.5. It is im-

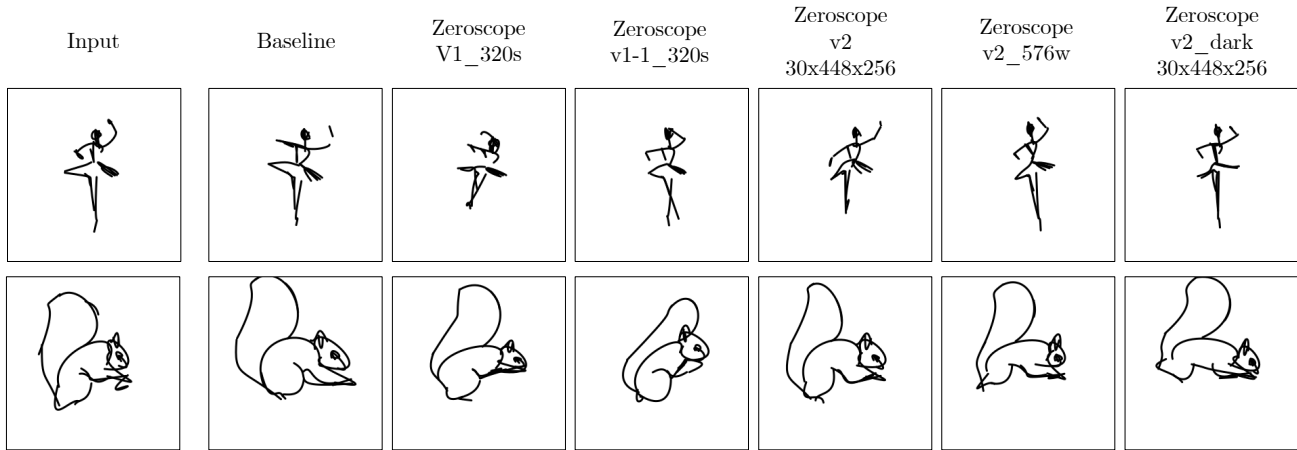


Figure 6. Other text-to-video backbones. We show the first frame from the results of five alternative text-to-video models. The full videos and additional examples are provided in the website. Observe that the choice of backbone model affects the output video in terms of both the sketch’s appearance and the type of generated motion.

portant to note that our method can be employed with vector sketches created through alternative approaches, such as [2–4, 6], or even sketched by hand. For optimal performances, we recommend to represent the input sketch with cubic Bezier curves.

### 3.2. Additional training details

To improve stability in early training steps, we initialize  $\mathcal{M}$  so that the predicted local displacements are small and the global transformations  $\mathcal{T}^j$  are close to the identity matrix.

When sampling timesteps for the SDS loss, we follow DreamFusion [5] and avoid sampling very early or very late steps. In practice we sample the steps uniformly in the range [50, 950]. When rendering the video frames for training we use a canvas size of  $256 \times 256$ , even when using text-to-video models trained with different aspect ratios. This limitation is primarily due to memory constraints. Lifting this restriction may aid in improving visual fidelity at the cost of higher VRAM requirements. We similarly restrict ourselves to 24 frames. Increasing this value can improve smoothness at the cost of additional memory. Our baseline method requires roughly 23GB of VRAM.

## 3.3. Evaluation details

### 3.3.1 Baseline implementations

When comparing to alternative methods, we used the following implementations:

- ModelScope: <https://huggingface.co/spaces/damo-vilab/MS-Image2Video-demo/tree/main>
- ZeroScope: <https://huggingface.co/spaces/fffiloni/zeroscope-img-to-video/tree/main>

- VideoCrafter: <https://huggingface.co/spaces/VideoCrafter/VideoCrafter/tree/main>
- Animated Drawings: <https://sketch.metademolab.com/canvas>
- Gen-2: <https://research.runwayml.com/gen2>

Note that Gen-2 is actively updated. We obtained our results on October 19th, 2023.

### 3.3.2 Evaluation metrics

For our sketch-to-video consistency metric we use OpenAI’s CLIP ViT-B/32. For the text-to-video alignment metric we use Microsoft’s xclip-large-patch14. This X-CLIP model expects 8 input frames, which are sampled uniformly from the generated video.

### 3.3.3 Evaluation data

In Tabs. 2 to 4 we provide the list of sketches used for our quantitative evaluations, along with their associated prompt.


### 3.3.4 User Study

As discussed in section 5.2 of the main paper, we conduct a user study to validate our suggested components. The user study examines the sketch-to-video consistency and text-to-video alignment of the animations produced when disabling different components of our method. An example question is shown in Fig. 7. These questions were repeated for all the targets in the evaluation set, each time comparing our full method to a random choice of the ablation scenarios.


Below are two short animations of the input sketch shown on the left. The sketch is animated according to the prompt: "**A ceiling fan rotating blades to circulate air in a room.**"

Please choose the most suitable answer in the following two questions (if both options look the same to you, just pick a random one).


Input



A



B



A

B

Which of the animations above better fits the text prompt?  A  B

Which of the animations above better preserves the appearance of the input sketch?  A  B

Figure 7. User study example question.

## 4. Acknowledgements

We thank Oren Katzir and Guy Tevet for providing feedback on early versions of our manuscript. This work was partially supported by BSF (grant 2020280) and ISF (grants 2492/20 and 3441/21).

Table 2. Sketches, and prompts used for our quantitative evaluations for the "animal" class.



The penguin is shuffling along the ice terrain, taking deliberate and cautious step with its flippers outstretched to maintain balance.



The goldenfish is gracefully moving through the water, its fins and tail fin gently propelling it forward with effortless agility.



The crab scuttled sideways along the sandy beach, its pincers raised in a defensive stance.



A galloping horse.



The eagle soars majestically, with powerful wing beats and effortless glides.



A hummingbird hovers in mid-air and sucks nectar from a flower.



A dolphin swimming and leaping out of the water.



A butterfly fluttering its wings and flying gracefully.



A gazelle galloping and jumping to escape predators.



The squirrel uses its dexterous front paws to hold and manipulate nuts, displaying meticulous and deliberate motions while eating.

Table 3. Sketches, and prompts used for our quantitative evaluations for the "human" class.



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



The boxer ducking and weaving to avoid his opponent's punches, and to punch him back.



The runner runs with rhythmic leg strides and synchronized arm swing propelling them forward while maintaining balance.



The jazz saxophonist performs on stage, his upper body sways subtly to the rhythm of the music.



The ballerina is dancing.



The biker is pedaling, each leg pumping up and down.



A martial artist executing precise and controlled movements in different forms of martial arts.



A surfer riding and maneuvering on waves on a surfboard.



A figure skater gliding, spinning, and performing jumps on ice skates.



A basketball player dribbling and passing while playing basketball.

Table 4. Sketches, and prompts used for our quantitative evaluations for the "object" class.



A waving flag fluttering and rippling in the wind.



A parachute descending slowly and gracefully after being deployed.



A wind-up toy car, moving forward or backward when wound up and released.



A windmill spinning its blades in the wind to generate energy.



A ceiling fan rotating blades to circulate air in a room.



A clock hands ticking and rotating to indicate time on a clock face.



The wine in the wine glass sways from side to side.



The airplane moves swiftly and steadily through the air.



The spaceship accelerates rapidly during takeoff, utilizing powerful rocket engines.



The flower is moving and growing, swaying gently from side to side.



## References

- [1] Mathias Eitz, James Hays, and Marc Alexa. How do humans sketch objects? *ACM Trans. Graph. (Proc. SIGGRAPH)*, 31(4):44:1–44:10, 2012. [2](#)
- [2] Kevin Frans, Lisa B. Soros, and Olaf Witkowski. Clipdraw: Exploring text-to-drawing synthesis through language-image encoders. *CoRR*, abs/2106.14843, 2021. [4](#)
- [3] David Ha and Douglas Eck. A neural representation of sketch drawings. *CoRR*, abs/1704.03477, 2017.
- [4] Ajay Jain, Amber Xie, and Pieter Abbeel. Vectorfusion: Text-to-svg by abstracting pixel-based diffusion models. *arXiv*, 2022. [4](#)
- [5] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. In *The Eleventh International Conference on Learning Representations*, 2022. [4](#)
- [6] Yael Vinker, Yuval Alaluf, Daniel Cohen-Or, and Ariel Shamir. Clipascene: Scene sketching with different types and levels of abstraction. 2022. [4](#)
- [7] Yael Vinker, Ehsan Pajouheshgar, Jessica Y. Bo, Roman Christian Bachmann, Amit Haim Bermano, Daniel Cohen-Or, Amir Zamir, and Ariel Shamir. Clipasso: Semantically-aware object sketching. *ACM Trans. Graph.*, 41(4), 2022. [2](#), [3](#)
- [8] Jiuniu Wang, Hangjie Yuan, Dayou Chen, Yingya Zhang, Xiang Wang, and Shiwei Zhang. Modelscope text-to-video technical report. *arXiv preprint arXiv:2308.06571*, 2023. [3](#)