Supplementary Materials: Sketch-Aware Interpolation Network

Anonymous Authors

1 STATISTICAL SIGNIFICANCE FOR QUANTITATIVE EVALUATION

Standard deviation. We report the standard deviation (σ) on all quantitative metrics as shown in Table 1. Specifically, our method achieves the lowest standard deviation on metrics SSIM and CD, and remains small on other metrics. It indicates our method fluctuates slightly compared to other existing methods.

Statistical significance. Table 2 lists the statistical significance of our method's improvements over existing algorithms. We further conducted a paired sample t-test. The test is with the null hypotheses that the performance of our method regarding PSNR, SSIM, IE and CD metrics are identical to to the existing methods. We can observe that for all metrics, our model exhibits a statistical significance under a confidence level 0.01 regarding the p-value. Therefore, we reject the null hypothesis and have sufficient evidence to say that our method is improved from the existing interpolation methods.

2 ADDITIONAL QUALITATIVE EXAMPLES

Figure 1 illustrates five additional interpolation examples for the comparison between the proposed SAIN and other state-of-the-art interpolation methods.

To further emphasize the validity of various model components, we provided more detailed view for qualitative ablation study as shown in Fig. 2.

3 ANIMATION DEMO

We also included a video example in this submission, which can be found in the supplemental files or via the Youtube link: https://youtu.be/00-KFxRYvCM. It compares our proposed method with a most recent method [12]. The top left is the input animation with a low frame rate, which contains 5 frames per second. The the top right animation is the ground true frames with a frame rate 10. The bottom left animation is interpolated by DQBC [12], and the animation interpolated using our SAIN is given at the bottom right. The animation interpolated by DQBC contains obvious blurriness, while our method produce a high quality result that is very close to the ground truth.

4 CODE & DATASET

The full implementation of our method can be found in the github repository: https://github.com/none-master/FC-SIN. The link for our dataset STD-12K is also available in this repository.

5 DETAILS OF EVALUATION METRICS

To quantitatively evaluate the results of our experiment, we applied four metrics: Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM), Interpolation Error (IE) and Chamfer Distance (CD). They are commonly used in benchmarking video and animation interpolation methods. We provides the details for their computations below.

PSNR is used to measure the reconstruction quality of lossy image compression codecs, which provides an approximate estimate of the human perception of the reconstruction quality. Given a reference image f and a test image g, with size $M \times N$, the PSNR between f and g is computed as:

$$PSNR(f,g) = 10\ log_{10}\ (255^2/MSE(f,g)), \tag{1}$$

where:

$$MSE(f,g) = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}(f_{ij} - g_{ij})^2. \tag{2}$$

The Mean Square Error (MSE) represents the average error of all pixels between the two images. Therefore, the PSNR value approaches infinity as the MSE approaches zero, *i.e.*, a higher value of PSNR implies lower image differences.

SSIM serves as a tool to measure the structural similarity between two images, which is considered relevant to the quality perception of the human visual system. Unlike conventional error summation methods such as PSNR, SSIM is designed by modelling image distortion as a combination of correlation loss, luminance distortion and contrast distortion. Given a reference image f and a test image g, SSIM is defined as:

$$SSIM(f,g) = l(f,g)\ c(f,g)\ s(f,g), \tag{3}$$

where:

$$\begin{cases} l(f,g) = \dfrac{2\mu_f\mu_g + C_1}{\mu_f^2 + \mu_g^2 + C_1}, \\ c(f,g) = \dfrac{2\sigma_f\sigma_g + C_2}{\sigma_f^2 + \sigma_g^2 + C_2}, \\ s(f,g) = \dfrac{\sigma_{fg} + C_3}{\sigma_f\sigma_g + C_3}. \end{cases} \tag{4}$$

In detail, $l(f,g)$ is the luminance comparison function, which measures the closeness of two images' mean luminance (μ_f and μ_g). This factor is maximal and equal to 1 only if $\mu_f = \mu_g$. $c(f,g)$ represents the contrast comparison function, which measures the closeness of the contrast between two images. The value of contrast is measured by the standard deviation σ_f and σ_g. $s(f,g)$ is for the structure comparison, which evaluates the correlation coefficient between two images. Note that μ_{fg} is the covariance between f and g. C_1, C_2 and C_3 is included for the purose of avoiding null denominator. The SSIM value ranges in $[0,1]$, where higher scores represent the better correlation between two images.

IE measures the pixel-wise difference between a reference image f and a test image g, which is defined as:

$$IE(f,g) = \sqrt{MSE(f,g)}. \tag{5}$$

CD is typically used in 3D scenarios, where the distance between two point clouds is calculated by averaging the shortest distance from each point in a cloud to the other. In the context of 2D sketch interpolation measures, CD is able to measure the distance between

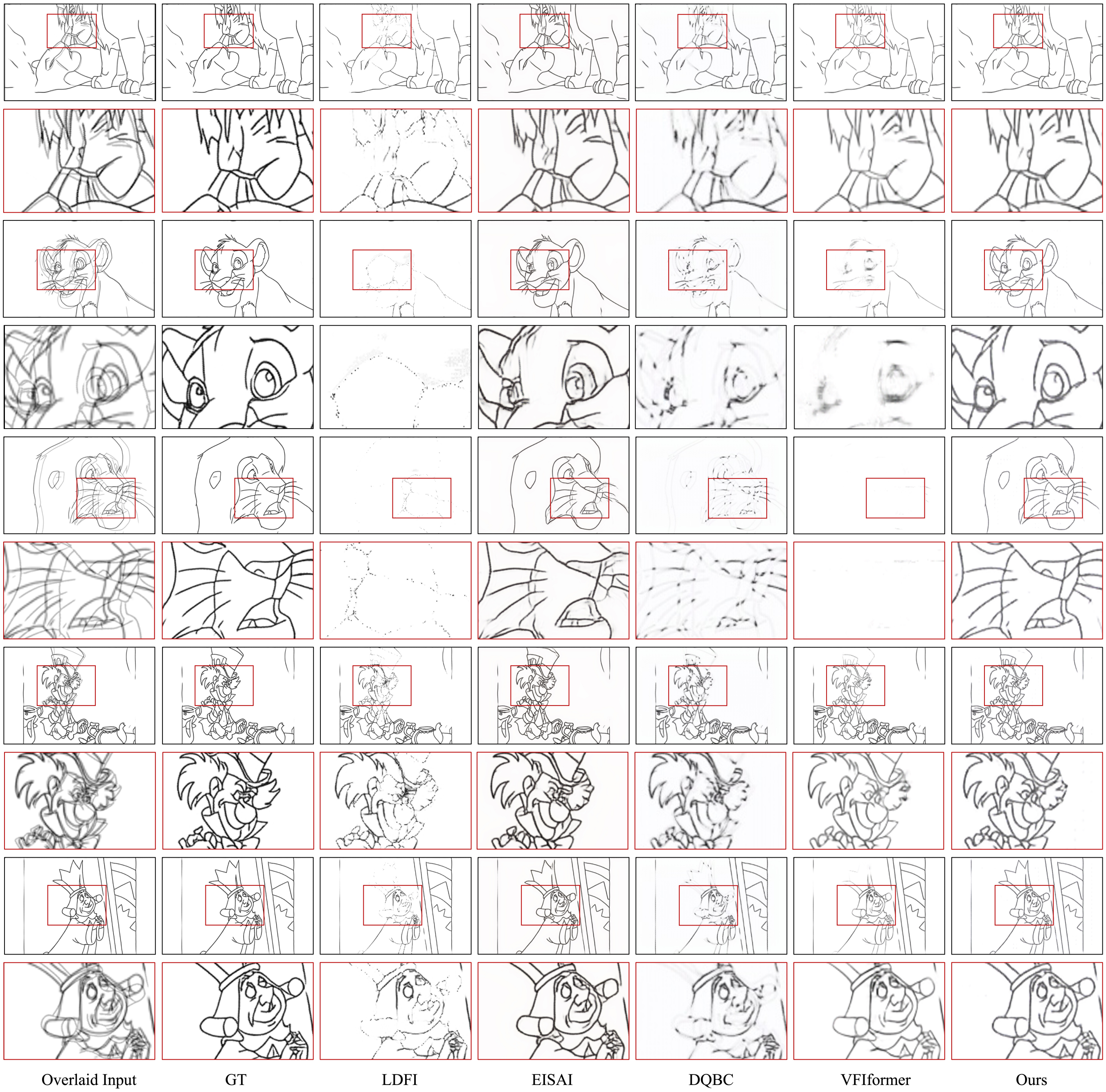

Figure 1: Qualitative comparison between the proposed SAIN (Ours) and the state-of-the-art interpolation methods.

strokes in interpolated images and ground truth images. Given two binary images f and g, CD is defined as:

$$CD(f,g) = \frac{1}{2HW} \sum fDT(g) + gDT(f), \tag{6}$$

where DT denotes the Euclidean distance transform and HW is the product of image height and width.

6 HISTOGRAM OF STD-12K DATASET

We provided the histogram of the stroke intensity as shown in Fig 4.

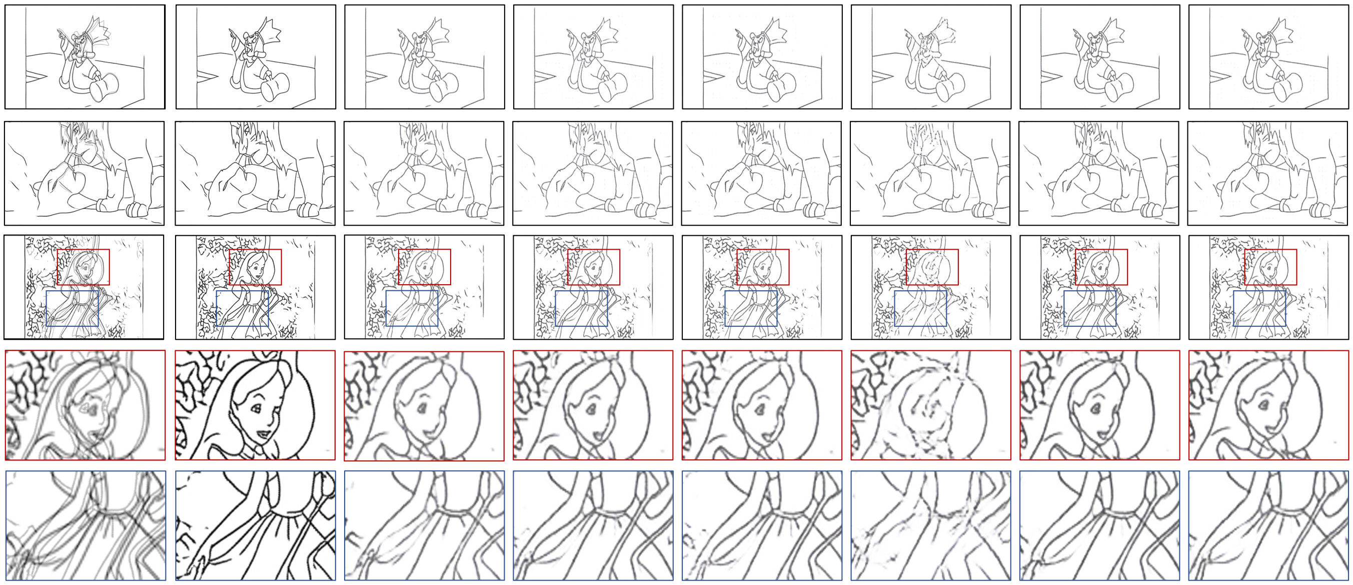
Overlaid Input GT w/o. CSB stream w/o. CCB stream w/o. sketch corr. w/o. pixel dynamics w/o. region corr. SAIN

Figure 2: Further qualitative example of ablation study.

Method (Year)	PSNR ↑	SSIM ↑	IE ↓	CD ↓
AnimeInbet (2023)	12.30 ± 2.19	0.5796 ± 0.16	25.00 ± 0.059	62.20 ± 3.52e-3
Sketchformer (2020)	17.23 ± **0.28**	0.7847 ± **1.60e-5**	14.14 ± 0.030	10.34 ± 0.033
LDFI (2019)	18.18 ± 2.29	0.8048 ± 0.084	12.71 ± 0.030	4.05± 3.29e-4
SGCVI (2021)	17.56 ± 2.03	0.7850 ± 0.077	13.56 ± 0.027	3.68 ± 3.27e-4
EISAI (2022)	<u>19.07</u> ± 2.66	<u>0.8422</u> ± 0.084	11.62 ± 0.033	<u>1.76</u> ± 1.51e-4
Super SloMo (2018)	18.05 ± 2.20	0.7995 ± 0.081	12.86 ± 0.028	3.82 ± 2.52e-4
AdaCoF (2020)	18.08 ± 2.19	0.8027 ± 0.079	12.82 ± 0.028	4.39 ± 3.05e-4
SoftSplat (2020)	17.08 ± 1.40	0.7328 ± 0.073	14.17 ± **0.022**	5.61 ± 2.61e-4
VFIT (2022)	8.45 ± 2.30	0.5622 ± 0.15	39.03 ± 0.091	13.59 ± 5.73e-4
RIFE (2022)	15.11 ± 2.73	0.6258 ± 0.16	18.37 ± 0.054	641.58 ± 0.033
VFIformer (2022)	19.05 ± 2.51	0.8387 ± 0.079	<u>11.59</u> ± 0.031	6.54 ± 7.71e-4
DQBC (2023)	18.60 ± 2.29	0.8015 ± 0.082	12.12 ± 0.029	2.39 ± 1.53e-4
SAIN (Ours)	**20.32** ± 2.71	**0.8727** ± 0.071	**10.09** ± 0.030	**1.54** ± **1.17e-4**

Table 1: Quantitative comparison between SAIN and the state-of-the-art interpolation methods.

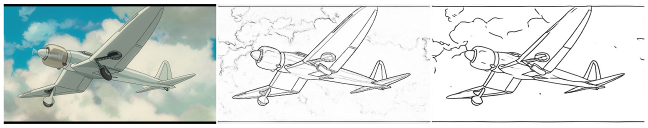

Figure 3: An example of noise refinement.

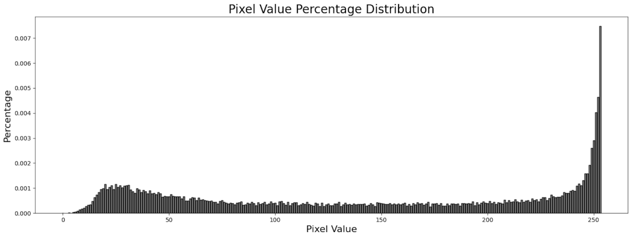

Figure 4: Histogram of the stroke intensity.

7 NOISE REDUCTION OF STD-12K DATASET

As noise can happen during the extraction of lines, we adopted an existing CNN based method - *Sketch Simplify* for refinement. Fig. 3 shows the comparison between the noisy and refined results.

Method (Year)	PSNR		SSIM		IE		CD	
	statistic	*p*	statistic	*p*	statistic	*p*	statistic	*p*
AnimeInbet (2023)	-143.35	<0.01	-84.83	<0.01	124.77	<0.01	64.27	<0.01
Sketchformer (2020)	-314.29	<0.01	-547.65	<0.01	766.07	<0.01	88.05	<0.01
LDFI (2019)	-65.28	<0.01	-58.06	<0.01	66.84	<0.01	40.78	4.60e-265
SGCVI (2021)	-71.12	<0.01	-65.77	<0.01	77.61	<0.01	54.43	<0.01
EISAI (2022)	-58.40	<0.01	-44.43	1.88e-300	57.43	<0.01	15.29	5.24e-50
Super SloMo (2018)	-63.52	<0.01	-56.67	<0.01	66.11	<0.01	53.32	0
AdaCoF (2020)	-61.97	<0.01	-55.25	<0.01	64.67	<0.01	52.06	<0.01
SoftSplat (2020)	-70.32	<0.01	-127.24	<0.01	81.68	<0.01	76.23	<0.01
VFIT (2022)	-208.22	<0.01	-94.99	<0.01	156.80	<0.01	93.94	<0.01
RIFE (2022)	-88.88	<0.01	-69.60	<0.01	75.83	<0.01	88.05	<0.01
VFIformer (2022)	-51.60	<0.01	-37.94	1.09e-237	50.09	<0.01	31.77	1.265e-179
DQBC (2023)	-55.72	<0.01	-63.19	<0.01	55.91	<0.01	42.87	2.62e-285

Table 2: Paired two-sample t-test for the comparison of our method with the existing methods.

8 ADDITIONAL ABLATION STUDY ON VECTOR-BASED ENCODING

To demonstrate the benefit of raster field correspondence compared with the vector-based strategy, we altered the region-correspondence by introducing the vectorized mechanism used in Sketchformer. The results shown in Table 3 demonstrate the advantage of using raster fields.

Method	PSNR ↑	SSIM ↑	IE ↓	CD ↓
SAIN (Ours)	**20.32**	**0.8727**	**10.09**	**1.54**
Raster encoding	**20.32**	**0.8727**	**10.09**	**1.54**
Vector encoding	19.83	0.8512	10.67	1.99

Table 3: Raster encoding vs. vector encoding

9 EVALUATE ON DIFFERENT ANIMATION STYLE SUBSET

We have provided an analysis of our method regarding different anime categories (Disney and Japanese) to help understand the method for different scenarios as listed in Table 4.

Method	PSNR ↑	SSIM ↑	IE ↓	CD ↓
SAIN (Ours)	**20.32**	**0.8727**	**10.09**	**1.54**
Disney	20.49	0.8812	9.82	1.34
Japanese	20.03	0.8588	10.53	1.88

Table 4: Evaluate on different animation style subset

REFERENCES

[1] Shuhong Chen and Matthias Zwicker. 2022. Improving the Perceptual Quality of 2D Animation Interpolation. (2022).

[2] Zhewei Huang, Tianyuan Zhang, Wen Heng, Boxin Shi, and Shuchang Zhou. 2022. Real-time intermediate flow estimation for video frame interpolation. (2022), 624–642.

[3] Huaizu Jiang, Deqing Sun, Varun Jampani, Ming-Hsuan Yang, Erik Learned-Miller, and Jan Kautz. 2018. Super slomo: High quality estimation of multiple intermediate frames for video interpolation. (2018), 9000–9008.

[4] Hyeongmin Lee, Taeoh Kim, Tae-young Chung, Daehyun Pak, Yuseok Ban, and Sangyoun Lee. 2020. Adacof: Adaptive collaboration of flows for video frame interpolation. (2020), 5316–5325.

[5] Xiaoyu Li, Bo Zhang, Jing Liao, and Pedro V Sander. 2021. Deep sketch-guided cartoon video inbetweening. *IEEE Transactions on Visualization and Computer Graphics* 28, 8 (2021), 2938–2952.

[6] Liying Lu, Ruizheng Wu, Huaijia Lin, Jiangbo Lu, and Jiaya Jia. 2022. Video Frame Interpolation with Transformer. (2022), 3532–3542.

[7] Rei Narita, Keigo Hirakawa, and Kiyoharu Aizawa. 2019. Optical flow based line drawing frame interpolation using distance transform to support inbetweenings. (2019), 4200–4204.

[8] Simon Niklaus and Feng Liu. 2020. Softmax splatting for video frame interpolation. (2020), 5437–5446.

[9] Leo Sampaio Ferraz Ribeiro, Tu Bui, John Collomosse, and Moacir Ponti. 2020. Sketchformer: Transformer-based representation for sketched structure. (2020), 14153–14162.

[10] Zhihao Shi, Xiangyu Xu, Xiaohong Liu, Jun Chen, and Ming-Hsuan Yang. 2022. Video frame interpolation transformer. (2022), 17482–17491.

[11] Li Siyao, Tianpei Gu, Weiye Xiao, Henghui Ding, Ziwei Liu, and Chen Change Loy. 2023. Deep Geometrized Cartoon Line Inbetweening. (2023), 7291–7300.

[12] Chang Zhou, Jie Liu, Jie Tang, and Gangshan Wu. 2023. Video Frame Interpolation with Densely Queried Bilateral Correlation. *arXiv preprint arXiv:2304.13596* (2023).