Koala-36M : A Large-scale Video Dataset Improving Consistency between Fine-grained Conditions and Video Content

Supplementary Material

A. Effectiveness of video splitting methods

To validate the accuracy and efficiency of our proposed Color-Struct SVM (CSS) for scene transition detection, we conduct the following experiments. We annotate transitions in 10,000 video clips, creating a test set (approximately half of the videos contain transitions). We then apply our proposed method and open-source methods to detect transitions in the test set, recording the precision and recall of the detections. The open-source method is primarily based on Pyscenedetect [6], and we test two versions: one that detects transitions based solely on HSL (Hue, Saturation, Lightness) and another that uses both HSL and edge detection. As shown in the Tab. 3, it can be observed that our transition detection algorithm outperforms the two pyscenedetect-based methods in terms of both precision and recall. Notably, our algorithm achieves a high recall rate, indicating that it rarely misses transitions in videos.

Table 3. **Transitions Detection Metrics for Different Methods**

Method	Accuracy	Recall	Precision
Pydetect(hsl)	0.4421	0.3096	0.5920
Pydetect(hsl+edge)	0.4574	0.4146	0.5854
Ours	**0.7741**	**0.9395**	**0.7547**

On the other hand, we compare the runtime efficiency of our method with that of the open-source algorithms. We record the CPU runtime of our algorithm and other open-source algorithms at different resolutions, with the experimental results shown in Tab. 4. We find that our method performs comparably to other methods at 256 resolution. However, as the video resolution increases, our method becomes significantly faster than the other methods (Fig. 11).

Table 4. **Time Consumption for Different Resolutions and Methods(ms)**

Resolution	Our Method	Pydetect (hsl)	Pydetect (hsl+edge)
256	1.42	**0.68**	2.50
512	**2.45**	2.63	8.82
720p	**6.15**	10.73	30.57
1080p	**12.26**	26.16	70.11
4k	**41.98**	102.55	267.18

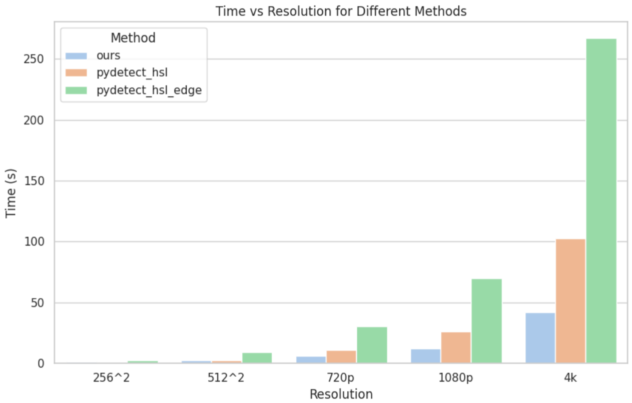


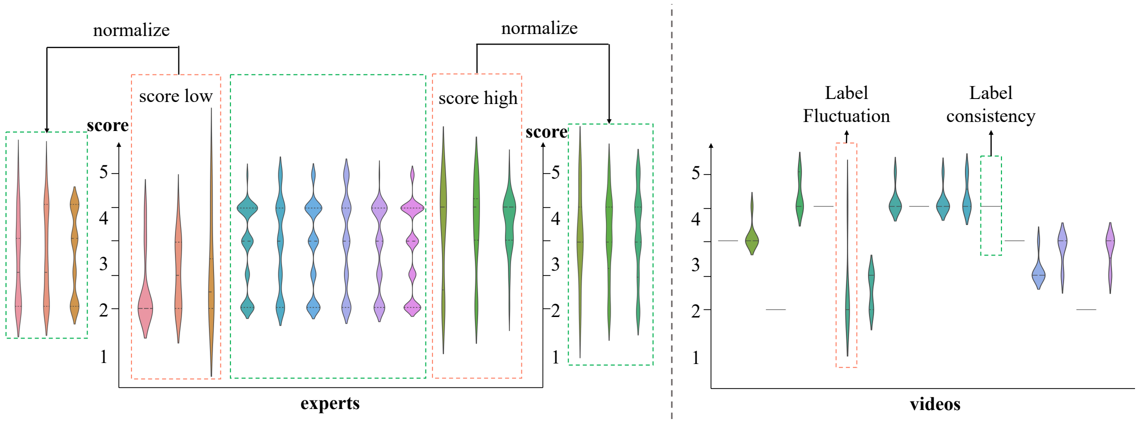
Figure 11. **Time Consumption for Different Resolutions.** Our method is faster than the others at higher video resolutions.

B. Elimination of Deviations between True Scores and Labeled Scores

After establishing the criteria, we randomly sample 200k videos and have it annotated by trained experts, with each video being scored by eight experts on a scale of 1 to 5. To ensure that the annotations closely reflect the true suitability scores, we need to address two types of errors: **Individual Preference Bias**: As shown in the Fig. 12(a), we visualize the violin plots of scores given by different experts. The expert on the left tends to give lower scores, while the expert on the right tends to give higher scores. These individual preferences can cause the final scores of some videos to be lower or higher than their actual values. Therefore, we standardize the scores of each expert and then scaled them using the mean and variance of the overall scores to eliminate the bias introduced by different experts. From the figure, it can be seen that the scores processed through our normalization and rescaling methods align more closely with the overall score distribution. **Label Fluctuation Bias**: As shown in the Fig. 12(b), each video is annotated by eight experts, and different experts may assign different scores due to varying interpretations of the criteria. This leads to label fluctuations. We use the mean score to reduce the error caused by these fluctuations.

C. Ablation Study of Training Suitability Assessment Network

We conduct comprehensive ablation experiments on our *Training Suitability Assessment Network*. The experimental results are shown in Tab. 5. The baseline model utilizes only dynamic features. Adding the static branch en-



(a)

(b)

Figure 12. **Score distribution of different experts and videos.** Fig.(a) visualizes the score distribution of different experts. We eliminate individual preference bias through normalization. Fig.(b) visualizes the score distribution of different videos. We reduce label fluctuation bias with average.

ables the model to capture more static information, thereby improving overall performance. The inclusion of the feature branch allows the model to leverage additional label information, further enhancing its performance. The WCGB module integrates label information with dynamic and static features through a cross-gating mechanism, achieving optimal performance. Each module addition significantly boosts the model's performance. Combining dynamic and static branches allows the model to capture both types of information. The feature branch utilizes label information for further improvement. The WCGB module optimizes feature integration, achieving the best results. In Tab. 6, we further supplement the results of Dover to demonstrate the superiority of our TSA module.

Table 5. Performance of Different Combinations

Dynamic branch	Static branch	Feature branch	WCGB	PLCC↑	SRCC↑	KRCC↑	RMSE↓
✓				0.8684	0.8580	0.7027	0.4644
✓	✓			0.8730	0.8637	0.7111	0.4555
✓	✓	✓		0.8953	0.8864	0.7397	0.4203
✓	✓	✓	✓	**0.8974**	**0.8868**	**0.7406**	**0.4099**

Table 6. **Comparison between Dover and TSA.**

Method	PLCC↑	SRCC↑	KRCC↑	RMSE↓
FastVQA	0.8684	0.8580	0.7027	0.4644
Dover	0.8554	0.8506	0.6788	0.6497
Ours	**0.8974**	**0.8868**	**0.7406**	**0.4099**

D. Further Analysis on Data Filtering

We claim existing methods neglect the joint distribution of sub-metrics, resulting in inaccurate thresholds in Sec. 4.3. We demonstrate our claim with two conclusions. (1) There exists joint distribution between sub-metrics. As shown in Tab. 7, we select three sub-metrics and calculate the pairwise correlation coefficients on the unfiltered data, finding that the different sub-metrics are not independent of each other. (2) Due to the slight deviation of the sub-metrics threshold from the optimal values, the filtering errors accumulate on the errors of each sub-metric, resulting in a larger overall filtering error. As shown in Tab. 8, the amount of incorrectly filtered data increases as the number of sub-metrics with inaccurate threshold increases.

Table 7. **Correlation coefficients between sub-metrics.**

Correlation coefficients	(Clarity, Aesthetic)	(Clarity, Motion)	(Motion, Aesthetic)
Pearson correlation	0.3774	-0.4028	-0.2515
Spearman's rank correlation	0.3732	-0.4324	-0.2347

Table 8. **The incorrectly filtered data with the increasing number of inaccurate sub-metrics thresholds.**

Sub-metrics threshold with deviation (+10%)	(Clarity)	(Clarity, Aesthetic)	(Clarity, Motion, Aesthetic)
Incorrectly filtered data / all data	250K/48M	290K/48M	340K/48M

E. More Quantitative Results

As shown in Tab. 9, we further pretrain the same model on the HD-VG-130M dataset and evaluate its performance on VBench and additional metrics, such as FVD score. Our dataset outperforms Panda-70M and HD-VG across all metrics. The non-training metrics of other datasets are also presented in Tab. 1 of the main paper. Meanwhile, we further conduct experiments on higher resolution (512) and longer duration (4s), demonstrating the superiority of our dataset.

Table 9. **Quantitative results of text-to-video generation**

	Quality Score↑	Semantic Score↑	Total Score↑	FVD↓
Panda 256-2s	0.7343	0.3093	0.6493	570.87
HD-VG-130M 256-2s	0.7696	0.4541	0.7065	590.86
Koala-36M (condition) 256-2s	**0.7846**	**0.5919**	**0.7460**	**549.79**
Panda 256-4s	0.7395	0.4448	0.6806	451.09
Koala-36M (condition) 256-4s	**0.7644**	**0.4646**	**0.7045**	**354.79**
Panda 512-2s	0.7439	0.3954	0.6742	579.57
Koala-36M (condition) 512-2s	**0.7849**	**0.6495**	**0.7578**	**392.26**

F. Comparison of results from different metrics conditions

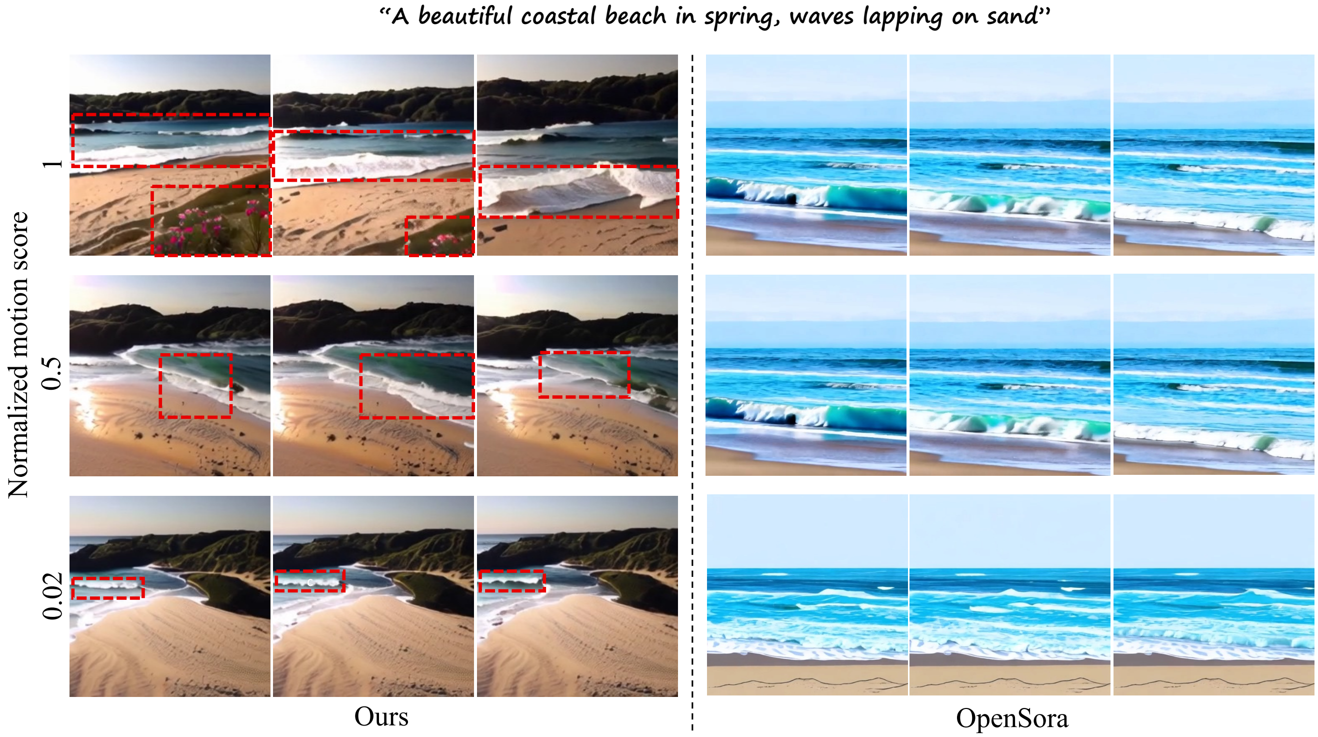

"A beautiful coastal beach in spring, waves lapping on sand"
Normalized motion score
1
0.5
0.02
Ours
OpenSora

Figure 13. **Comparison of results from different metrics conditions.** Our method has more precise control under the same normalized metrics score and stronger ability to decouple control over different metrics, when the style of videos transfer with the motion score.

G. More Qualitative Results of Text-to-video Generation

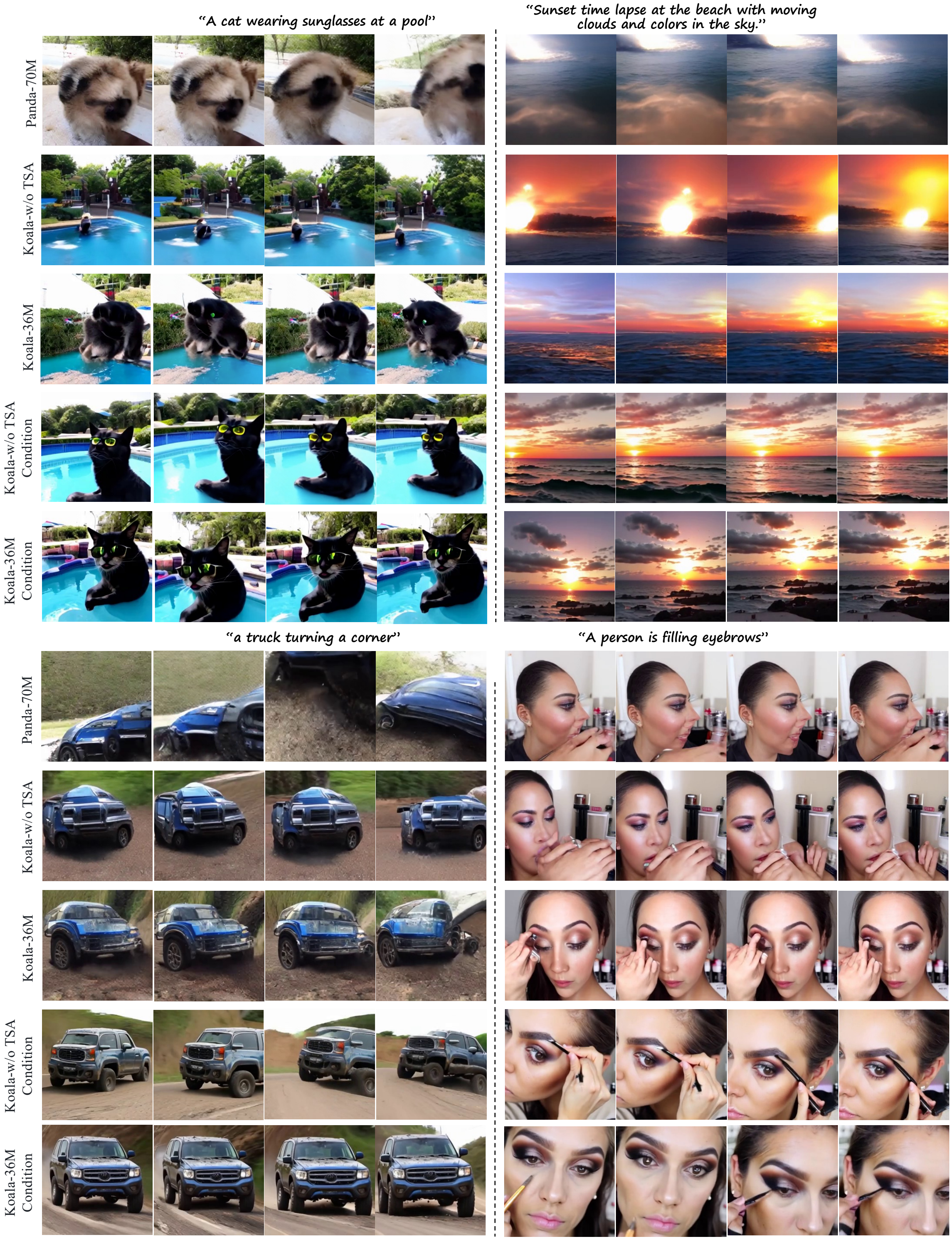

"A cat wearing sunglasses at a pool"
"Sunset time lapse at the beach with moving clouds and colors in the sky."
Panda-70M
Koala-w/o TSA
Koala-36M
Koala-w/o TSA Condition
Koala-36M Condition
"a truck turning a corner"
"A person is filling eyebrows"
Panda-70M
Koala-w/o TSA
Koala-36M
Koala-w/o TSA Condition
Koala-36M Condition

Figure 14. **More qualitative results of text-to-video generation.**