M³L: Language-based Video Editing via Multi-Modal Multi-Level Transformers

Tsu-Jui Fu[†], Xin Eric Wang[‡], Scott T. Grafton[†], Miguel P. Eckstein[†], William Yang Wang[†]

[†]UC Santa Barbara [‡]UC Santa Cruz

tsu-juifu@ucsb.edu, {scott.grafton, miguel.eckstein}@psych.ucsb.edu

william@cs.ucsb.edu, xwang366@ucsc.edu

Abstract

Video editing tools are widely used nowadays for digital design. Although the demand for these tools is high, the prior knowledge required makes it difficult for novices to get started. Systems that could follow natural language instructions to perform automatic editing would significantly improve accessibility. This paper introduces the language-based video editing (LBVE) task, which allows the model to edit, guided by text instruction, a source video into a target video. LBVE contains two features: 1) the scenario of the source video is preserved instead of generating a completely different video; 2) the semantic is presented differently in the target video, and all changes are controlled by the given instruction. We propose a Multi-Modal Multi-Level Transformer (M³L) to carry out LBVE. M³L dynamically learns the correspondence between video perception and language semantic at different levels, which benefits both the video understanding and video frame synthesis. We build three new datasets for evaluation, including two diagnostic and one from natural videos with human-labeled text. Extensive experimental results show that M³L is effective for video editing and that LBVE can lead to a new field toward vision-and-language research.

1. Introduction

Video is one of the most direct ways to convey information, as people are used to interacting with this world via dynamic visual perception. Nowadays, video editing tools like Premiere and Final Cut are widely applied for digital design usages, such as film editing or video effects. However, those applications require prior knowledge and complex operations to utilize successfully, which makes it difficult for novices to get started. For humans, natural language is the most natural way of communication. If a system can follow the given language instructions and automatically perform related editing actions, it will significantly improve accessi-
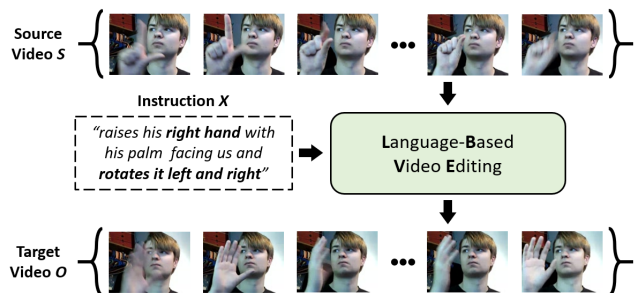

Figure 1. The introduced language-based video editing (LBVE) task. LBVE requires to edit a source video S into the target video T guided by the instruction X.

bility and meet the considerable demand.

In this paper, we introduce language-based video editing (LBVE), a general V2V task, where the target video is controllable directly by language instruction. LBVE treats a video and an instruction as the input, and the target video is edited from the textual description. As illustrated in Fig. 1, the same person performs different hand gestures guided by the instruction. Different from text-to-video (T2V) [3,34,38,43], video editing enjoys two following feature: 1) the scenario (e.g., scene or humans) of the source video is preserved instead of generating all content from scratch; 2) the semantic (e.g., property of the object or moving action) is presented differently in the target video. The main challenge of LBVE is to link the video perception with language understanding and reflect what semantics should be manipulated during the video generation but under a similar scenario. People usually take further editing steps onto a base video rather than create all content from the beginning. We believe that our LBVE is more practical and corresponding to human daily usage.

To tackle the LBVE task, we propose a multi-modal multi-level transformer (M³L) to perform video editing conditioning on the guided text. As shown in Fig. 2, M³L contains a multi-modal multi-level Transformer where the encoder models the moving motion to understand the entire video, and the decoder serves as a global planner to generate

each frame of the target video. For better video perception to link with the given instruction, the incorporated multi-level fusion fuses between these two modalities. During encoding, the local-level fusion is applied with the text tokens for fine-grained visual understanding, and the global-level fusion extracts the key feature of the moving motion. Reversely, during decoding, we first adopt global-level fusion from whole instruction to give a high-level plan for the target video, and then the local-level fusion can further generate each frame in detail with the specific property. With multi-level fusion, M³L learns explicit vision-and-language perception between the video and given instruction, yielding better video synthesis.

For evaluation, we collect three datasets under the brand-new LBVE task. There are E-MNIST and E-CLEVR, where we build from hand-written number recognition MNIST [32] and compositional VQA CLEVR [27], respectively. Both E-MNIST and E-CLEVR are prepared for evaluating the content replacing (different numbers or shapes and colors) and semantic manipulation (different moving directions or related positions). As a new task, diagnostic datasets help analyze the progress and discover the shortcomings. To investigate the capability of LBVE for natural video with open text, E-JESTER is built upon the same person performing different hand gestures with human instruction.

Our experimental results show that the multi-modal multi-level transformer (M³L) can carry out the LBVE task, and the multi-level fusion further helps between video perception and language understanding in both aspects of content replacing and semantic manipulation. In summary, our contributions are four-fold:

- We introduce the LBVE task to manipulate video content controlled by text instructions.
- We present M³L to perform LBVE, where the multi-level fusion further helps between video perception and language understanding.
- For evaluation under LBVE, we prepare three new datasets containing two diagnostic and one natural video with human-labeled text.
- Extensive ablation studies show that our M³L is adequate for video editing, and LBVE can lead to a new field toward vision-and-language research.

2. Related Work

Language-based Image Editing. Different from text-to-image (T2I) [42, 47, 53], which generates an image that matches the given instruction, language-based image editing (LBIE) understands the visual difference and edits between two images based on the guided text description. Image Spirit [11] and PixelTone [31] first propose the LBIE framework but accept only rule-based instruction and pre-defined semantic labels, which limits the practicality of LBIE. Inspired by numerous GAN-based methods [46, 67, 70] in T2I, there are some previous works [10, 51] perform LBIE as image colorization by the conditional GAN. Since humans do not always finish editing all-at-once but will involve several different steps, iterative LBIE (ILBIE) [15, 17] is proposed to imitate the actual process by the multi-turn manipulation and modeling the instructed editing history. Similar to LBIE, language-based video editing (LBVE) is to edit the content in a video by the guided instruction. To perform LBVE, it is required to model the dynamic visual perception instead of just a still image and consider the temporal consistency of each frame during the generation to make a smooth result video.

Language-based Video Generation. Generative video modeling [2–4, 12, 14, 16, 20, 23, 24, 28, 37, 38, 40, 44, 48, 49, 52, 54, 56, 57, 62] is a widely-discussed research topic that looks into the capability of a model to generate a video purely in pixel space. Built upon video generation, text-to-video (T2V) [3, 34, 38, 43] synthesizes a video by the guided text description, which makes the video output controllable by the natural language. In this paper, we investigate the video editing task, which replaces the specific object with different properties or changes the moving motion in the input video. Different from generating video from scratch, video editing requires extracting the dynamic visual perception of the source video and manipulating the semantic inside to generate the target video.

Video-to-Video Synthesis. Video super-resolution [1, 26], segmentation video reconstruction [59, 60], video style transfer [9, 13, 63], or video inpainting [6, 29, 66] can be considered as the particular case of video-to-video synthesis (V2V). Since they all depend on the task themselves, the variability between source-target is still under the problem-specific constraint. Among them, video prediction [19, 33, 45, 58], which predicts future frames conditioning on the given video, is one of the most related to our present LBVE task. Both video prediction and LBVE should understand the hidden semantic of the given video first and then predict the target frames with different content inside. While for video prediction, there are many possibilities of appeared future events, which makes it not deterministic for real-world usage [38]. On the other hand, LBVE is controllable by the given instruction, which involves both content replacing (object changing) and semantic manipulation (moving action changing). With the guided text description, LBVE can perform V2V with content editing and lead to predictable target video.

3. Language-based Video Editing

3.1. Task Definition

We study the language-based video editing (LBVE) task to edit a source video S into a target video O by a given
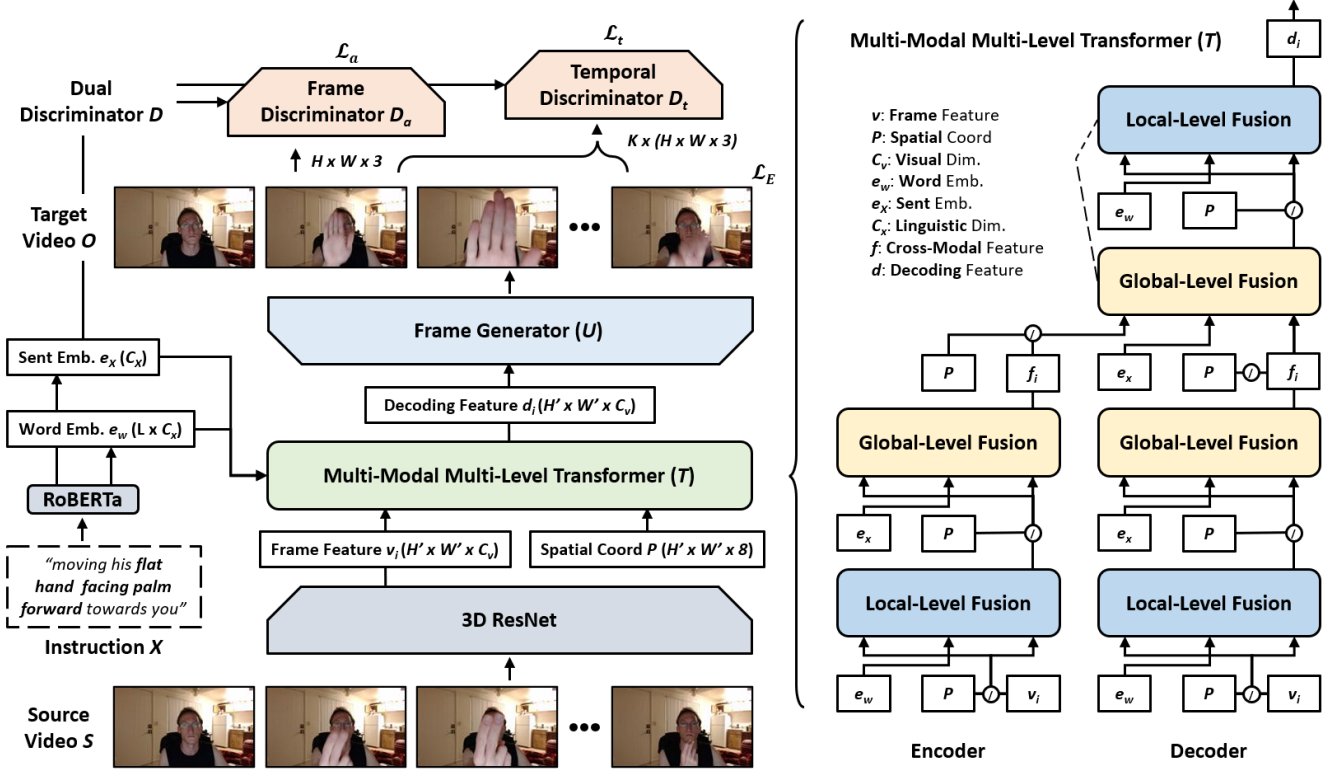
Figure 2. An overview architecture of our multi-modal multi-level transformer (M³L). M³L contains the multi-modal multi-level transformer T to encode the source video S and decode for the target video frame o by the multi-level fusion (MLF).

instruction X, as shown in Fig. 1. Specifically, the source video S contains N frames as $\{s_1, s_2, ..., s_N\}$, and the instruction $X = \{w_1, w_2, ..., w_L\}$ where L is the number of word token in X. The target video O also includes N frames as $\{o_1, o_2, ..., o_N\}$. For LBVE, the model should preserve the scenario from S but change the related semantics in O guided by X. Note that the editing process is at a pixel level where the model has to generate each pixel of each frame and then assemble them as the target video.

3.2. Overview

An overview of our multi-modal multi-level transformer (M³L) for LBVE is illustrated in Fig. 2. M³L first extracts the frame feature v_i for the frame s_i in the source video S; the sentence embedding e_X and each word embedding e_w for the instruction X. Then, the multi-modal multi-level transformer T is proposed to model the sequential information of the source and the target video as the decoding feature d_i. In particular, the multi-level fusion (MLF) performs the cross-modal fusion between video v and instruction $\{e_X, e_w\}$. The local-level fusion (LF) extracts which portion is perceived by token e_w across all words in X. Besides, the global-level fusion (GF) models the interaction between the entire video perception and the semantic motion from the whole instruction e_X. Finally, with d_i, the generator U generates the frame o_i in the target video

O. In addition, we apply the dual discriminator D, where the frame discriminator D_a helps the quality of every single frame, and the temporal discriminator D_t maintains the consistency as a smooth output video.

Frame and Linguistic Feature Extraction. To perform the LBVE task, We first apply 3D ResNet and RoBERTa [36] to extract the frame feature v and linguistic feature $\{e_X, e_w\}$ for the two modalities independently:

$$
\begin{aligned}
\{v_1, v_2, ..., v_N\} &= \text{3D ResNet}(\{s_1, s_2, ..., s_N\}), \\
e_X, \{e_{w_1}, e_{w_2}, ..., e_{w_L}\} &= \text{RoBERTa}(X),
\end{aligned}
\tag{1}
$$

where e_{w_i} is the word embedding of each token w_i, e_X is the entire sentence embedding of X, and L represents the length of the instruction X. In detail, $v \in \mathbb{R}^{H' \times W' \times C_v}$ and each $e \in \mathbb{R}^{C_x}$, where C_v and C_x is the feature dimension of vision and language, respectively.

3.3. Multi-Modal Multi-Level Transformer

As illustrated in Fig. 2, with the frame feature v and linguistic feature $\{e_X, e_w\}$ as the inputs, the multi-modal multi-level transformer T contains an encoder to model the sequential information of the source video S with the given instruction X, and a decoder to acquire the decoding feature d_i for generating the target video frame o_i. Both the encoder and decoder are composed of multi-level fusion
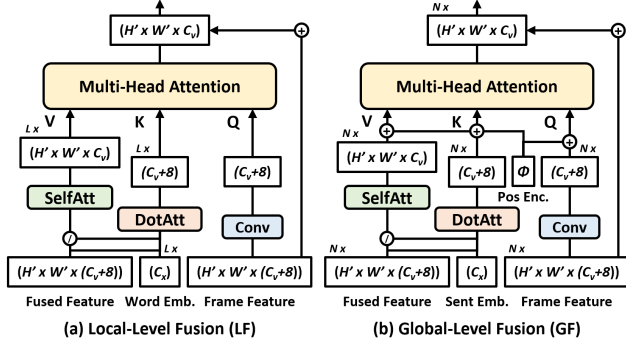
Figure 3. The computing flow of multi-level fusion (MLF), including local-level fusion (LF) and global-level fusion (GF).

(MLF), which is applied to fuse between vision and language with aspects from different levels.

Multi-Level Fusion Both video and language are multi-level conveyed, where video is composed of a series of image frames and language is a set of word tokens with a specific order. The multi-level fusion (MLF) consists of the local-level fusion (LF) to fuse between a single frame and each word token, and the global-level fusion (GF) models the entire video sequence with the whole instruction. The computation flow of MLF is illustrated in Fig. 3. Both LF and GF are computed with the multi-head attention (MHA) [55]. MHA acquires the weighted-sum of the value feature (V) by considering the correlation between the query feature (Q) and the key feature (K):

$$\text{MHA}(\text{Q}, \text{K}, \text{V}) = \text{softmax}(\frac{\text{Q} \cdot \text{K}^T}{\sqrt{C_{\text{K}}}})\text{V}. \qquad (2)$$

For the local-level fusion (LF), it investigates which portion should be focused by each word e_w in a single frame v_i. We provide the relative spatial information by concatenating a 8-D spatial coordinate feature P [35] with v_i as p^{L} . To fuse between vision and language, we apply the self-attention mechanism (SelfAtt) [68, 69] upon the concatenated feature q^{L} to capture the correlation between word expression and visual context into s^{L}. Different from CMSA [68], which concatenates frame feature with all token embedding directly, our LF further considers the importance of each token. We adopt a 1-layer convolutional net (Conv) to extract the context-only visual feature c^{L} along the channel of v_i; and the widely-used dot-product attention (DotAtt) [7, 65] for the word-focused visual feature d_l^{L} with each word e_{w_l}. Therefore, the correlation between c^{L} and d_l^{L} can be considered as the important portion of word w_l for our LF. We treat the context-only visual feature c^{L} as K, the word-focused visual feature d_l^{L} as Q, and the cross-modal feature s^{L} as V to perform LF through MHA. We also utilize the residual connection [21, 55] in LF:

$$\text{LF}(v_i^{\text{L}}) = v_i^{\text{L}} \oplus \text{MHA}(\text{c}^{\text{L}}, \text{d}^{\text{L}}, \text{s}^{\text{L}}), \qquad (3)$$

where

$$\text{p}^{\text{L}} = [v_i^{\text{L}}, P], \text{q}^{\text{L}} = \{[v_i^{\text{L}}, P, e_{w_1}], ..., [v_i^{\text{L}}, P, e_{w_L}]\},$$

$$\text{c}^{\text{L}} = \text{Conv}^{\text{L}}(\text{p}^{\text{L}}),$$

$$\text{d}_l^{\text{L}} = \text{DotAtt}(\text{p}^{\text{L}}, e_{w_l}) = \sum_{(h,w)} \text{softmax}(\text{p}^{\text{L}} \cdot W_{\text{d}}^L \cdot e_{w_l}^T)_{(h,w)} \cdot \text{p}_{(h,w)}^{\text{L}},$$

$$\text{s}_l^{\text{L}} = \text{SelfAtt}(\text{q}_l^{\text{L}}), \text{s}_{l(h,w)}^{\text{L}} = \sum_{(x,y)} \text{softmax}(\text{q}_l^{\text{L}} \cdot \text{q}_{l(h,w)}^{\text{L}}{}^T)_{(x,y)} \cdot \text{q}_{l(x,y)}^{\text{L}},$$

and W_{d}^L is the learnable attention matrix between p^{L} and e_w. In this way, our LF fuses between visual context and word expression from SelfAtt and take the important portion of each token from DotAtt into consideration.

For the global-level fusion (GF), it views the entire frame sequence $\{v_1, ..., v_N\}$ with the whole instruction e_X to extract the global motion of the video. Similar to LF, we acquire the fused cross-modal feature s_n^{G} from SelfAtt, the context-only visual feature c_n^{G} from Conv^{G}, and the sentence-focused visual feature d_n^{G} from DotAtt for v_n^{G}. To model the entire video, we follow [55], where the video-level feature of v_i can be represented as the relative weighted-sum over all frame-level v, and add on the positional encoding ϕ to incorporate the sequential order. We treat $\{\text{s}_n^{\text{G}}\}$ as V, $\{\text{c}_n^{\text{G}}\}$ as Q and $\{\text{d}_n^{\text{G}}\}$ as K for the correlation between a frame pair, to perform GF through MHA:

$$\text{GF}(v^{\text{G}}) = v^{\text{G}} \oplus \text{MHA}(\text{c}^{\text{G}} \oplus \phi, \text{d}^{\text{G}} \oplus \phi, \text{s}^{\text{G}} \oplus \phi), \qquad (4)$$

where

$$\text{p}^{\text{G}} = \{[v_1^{\text{G}}, P], ..., [v_N^{\text{G}}, P]\}, \quad \text{q}^{\text{L}} = \{[v_1^{\text{G}}, P, e_X], ..., [v_N^{\text{G}}, P, e_X]\},$$

$$\text{c}_n^{\text{G}} = \text{Conv}^{\text{G}}(\text{p}^{\text{G}})_n, \text{ d}_n^{\text{G}} = \text{DotAtt}(\text{p}_n^{\text{G}}, e_X), \text{ s}_n^{\text{G}} = \text{SelfAtt}(\text{q}_n^{\text{G}}).$$

By considering the correlation between frame with respect to the whole instruction from DotAtt, our GF models the video sequence as fused cross-modal feature from SelfAtt.

Encoder and Decoder. The encoder (Enc) in the multi-modal multi-level transformer T serves to model the source video sequence S with the given instruction X. Enc first adopts the local-level fusion (LF) to extract important portion from each single frame v^s with each word embedding e_w; then the global-level fusion (GF) extracts the entire video motion with the sentence embedding e_X as the cross-modal feature f_i^s:

$$f_i^s = \text{GF}(\text{LF}(v^s, e_w), e_X)_i. \qquad (5)$$

During decoding, the decoder (Dec) also extracts the cross-modal feature f_i^o as the same way from the previous generated frames $\{o_1, ..., o_{i-1}\}$. To acquire the decoding feature d_i to generate the target frame, GF is first adopted to give the high-level concept of moving motion by the interaction between the cross-modal feature f from source and target, where we treat f^s as the fused feature (V). LF is applied for detailed specific property provided from word tokens e_w:

$$f_i^o = \text{LF}(\text{GF}(\{v_1^o, ..., v_{i-1}^o\}, e_X | f^s)_i, e_w). \qquad (6)$$

In summary, the multi-modal multi-level transformer T models the source video frame v^s and the given instruction $\{e_X, e_w\}$, and considers previous generated target frames $\{o_1, ..., o_{i-1}\}$ to acquire the decoding feature d_i:

$$d_i = T(\{o_1, ..., o_{i-1}\}|v^s, \{e_X, e_w\}). \quad (7)$$

3.4. Video Frame Generation

With the decoding feature d_i from T, we adopt Res-Blocks [41] into the generator U to scale up d_i and synthesize into $\hat{o}_i$:

$$\hat{o}_i = U(d_i), \qquad \hat{O} = \{\hat{o}_1, \hat{o}_2, ..., \hat{o}_N\}. \quad (8)$$

We calculate the editing loss $\mathcal{L}_E$ by mean pixel difference using mean-square loss over each frame between O and $\hat{O}$:

$$\mathcal{L}_E = \frac{1}{N} \sum_{i=1}^{N} \text{MSELoss}(o_i, \hat{o}_i). \quad (9)$$

Dual Discriminator. Apart from the visual difference, we also consider the video quality of our generated $\hat{O}$. Similar to DVD-GAN [12], we apply the dual discriminator D, where the frame discriminator D_a improves the single frame quality and the temporal discriminator D_t constrains the temporal consistency for a smooth output video $\hat{O}$. We treat D_a as a binary classifier, which discriminates a target video frame o is from ground-truth O or our synthesized $\hat{O}$. Simultaneously, D_t judges that if K consecutive frames are smooth and consistent enough to be a real video fragment as the binary discrimination. The video quality loss $\mathcal{L}_G$ is computed for both frame quality and temporal consistency:

$$\begin{aligned}
\mathcal{L}_{\hat{a}} &= \frac{1}{N} \sum_{i=1}^{N} \log(1 - D_a(\hat{o}_i)), \\
\mathcal{L}_{\hat{t}} &= \frac{1}{M} \sum_{i=1}^{M} \log(1 - D_t(\{\hat{o}_i, ..., \hat{o}_{i+K-1}\})), \\
\mathcal{L}_G &= \mathcal{L}_{\hat{a}} + \mathcal{L}_{\hat{t}},
\end{aligned} \quad (10)$$

where $M = N - K + 1$. On the other hand, the dual discriminator D is training to distinguish between O and $\hat{O}$ by the following:

$$\begin{aligned}
\mathcal{L}_a &= \frac{1}{N} \sum_{i=1}^{N} (\log(1 - D_a(\hat{o}_i)) + \log(D_a(o_i))), \\
\mathcal{L}_t &= \frac{1}{M} \sum_{i=1}^{M} (\log(1 - D_t(\{\hat{o}_i, ..., \hat{o}_{i+K-1}\})) \\
&\qquad + \log(D_t(\{o_i, ..., o_{i+K-1}\}))), \\
\mathcal{L}_D &= \mathcal{L}_a + \mathcal{L}_t.
\end{aligned} \quad (11)$$

Therefore, they are optimized through an alternating min-max game:

$$\min_G \max_D \mathcal{L}_G + \mathcal{L}_D. \quad (12)$$

Algorithm 1 Multi-Modal Multi-level Transformer (M³L)

1: T: Multi-Modal Multi-Level Transformer
2: U: Frame Generator
3: D: Dual Discriminator, including D_a and D_t
4: S, X: Source Video, Instruction
5: O: Ground-Truth Target Video
6:
7: Initialize T, U, D
8: **while** TRAINING **do**
9:     $\{v_1, ..., v_N\}$ = 3D ResNet(S)
10:     $e_X, \{e_{w_1}, ..., e_{w_N}\}$ = RoBERTa(X)
11:     **for** $i \leftarrow 1$ to N **do**   ▷ teacher-forcing training
12:         $d_i \leftarrow T(\{o_1, ..., o_{i-1}\}|v, \{e_X, e_w\})$ ▷ Eq. 7
13:         $\hat{o}_i \leftarrow U(d_i)$
14:         $\mathcal{L}_E \leftarrow$ visual difference loss with O ▷ Eq. 9
15:         $\mathcal{L}_G \leftarrow$ video quality loss from D ▷ Eq. 10
16:         Update T and U by minimizing $\mathcal{L}_G + \mathcal{L}_E$
17:         $\mathcal{L}_D \leftarrow$ discrimination loss for D ▷ Eq. 11
18:         Update D by maximizing $\mathcal{L}_D$
19:     **end for**
20: **end while**

3.5. Learning of M³L

Algo. 1 presents the learning process of the proposed multi-modal multi-level transformer (M³L) for LBVE. Since LBVE is also a sequential generation process, we apply the widely used teacher-forcing training trick, where we feed in the ground-truth target frame o_{i-1} instead of the predicted $\hat{o}_{i-1}$ from the previous timestamp to make the training more robust. We adopt the multi-modal multi-level transformer T to model the source video and input instruction, and the frame generator U to generate the target video frame. During training, we minimize the video quality loss $\mathcal{L}_G$ with the visual difference $\mathcal{L}_E$ to optimize M³L. We also update the dual discriminator D, including the frame discriminator D_a and the temporal discriminator D_t, by maximizing $\mathcal{L}_D$. Therefore, the entire optimization object can be summarized as:

$$\min_{G, E} \max_D \mathcal{L}_G + \mathcal{L}_E + \mathcal{L}_D. \quad (13)$$

4. Datasets

To the best of our knowledge, there is no dataset that supports video editing with the guided text. Therefore, we build three new datasets specially designed for LBVE, including two diagnostic datasets (E-MNIST and E-CLEVR) and one human gesture dataset (E-JESTER) for the language-based video editing (LBVE) task. An overview of our built datasets is shown in Table 1, and examples of these three datasets are illustrated in Fig. 4.

E-MNIST. Extended from Moving MNIST [32, 52], the new E-MNIST dataset contains the instruction to describes the difference between two video clips. Hand-written num-
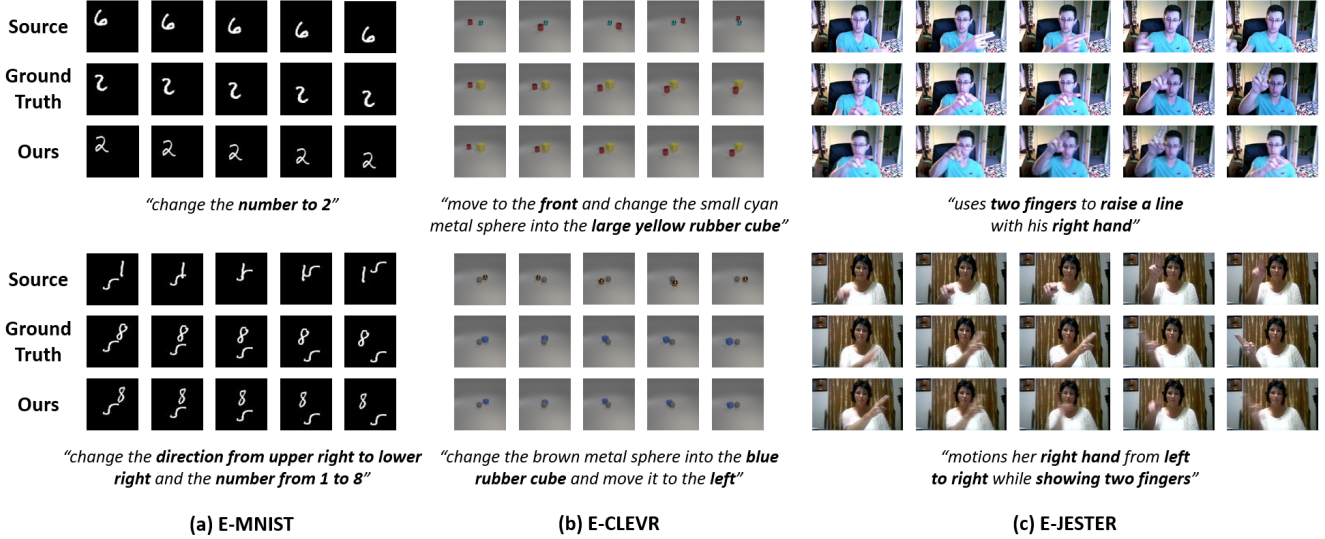
Figure 4. The sampled source videos, the ground-truth target videos, and the generated LBVE videos on all three datasets.

Dataset	#Train	#Test	#Frame	#Word	Resolution
S-MNIST	11,070	738	354,240	5.5	64x64
D-MNIST	11,070	738	354,240	16.0	64x64
E-CLEVR	10,133	729	21,7240	13.4	128x128
E-JESTER	14,022	885	59,508	9.9	100x176

Table 1. The statistics of our collected datsets.

bers are moving along a specific direction and will reverse its direction if bumps into a boundary. The instructions include two kinds of editing actions: *content replacing* is to replace the specific number with the given one, and *semantic manipulation* changes the starting direction for different moving motion. We prepare two levels of E-MNIST, S-MNIST and D-MNIST. S-MNIST is an easier one and includes only a single number, so the model only needs to replace the number or change the moving direction at a time. There are two numbers in the advanced D-MNIST, where the model is required to perceive which number should be replaced and which starting direction should be changed simultaneously. For both S-MNIST and D-MNIST, there are 11,808 pairs of source-target video.

E-CLEVR. Following CATER [18], we create each frame and combine them as the video in our E-CLEVR upon the original CLEVR dataset [27]. Each example consists of a pair of source-target videos with an instruction described the semantic altering. The editing action includes changing the property of the specific object and placing the moving object into a particular given final position. E-CLEVR contains plentiful object properties (e.g., color, shape, size, ...) and different relative positions of the final target. To highlight the importance of visual perception, not all aspects of the property will change; only the mentioned properties like the color and shape should be changed but keeps others the same. We generate 10,862 examples for E-CLEVR.

E-JESTER. Toward human action understanding, 20BN-JESTER [39] builds a large gesture recognition dataset. Each actor performs different kinds of gesture moving in front of the camera, which brings out 27 classes in total. This setting is appropriate to the video editing task where the source-target videos are under the same scenario (same person in the same environment) but with different semantics (different hand gestures). To support our LBVE task, we prepare pairs of clips from the same person as the source-target videos and collect the human-labeled instruction by Amazon Mechanical Turk (AMT)[1]. A person can exist in both training and testing sets but with different gestures. We ensure that there is no overlapping of the same person-gesture pairs between train/test splits. In this way, we can have the natural video whose scenario is preserved, but semantic is changing with natural guided text for our E-JESTER dataset, which can be a sufficient first step for LBVE. There are 14,907 pairs in E-JESTER.

5. Experiments

5.1. Experimental Setup

Evaluation Metrics.
- **VAD**: Inspired by IS [50] and FID [22], we apply 3D CNN and compute the video activation distance (VAD) as the mean L2 distance between video feature. Specifically, ResNeXt [64] is adopted for the diagnostic E-MNIST and E-CLEVR dataset. Besides, we utilize I3D [5] to extract the action video feature for E-JESTER. A lower VAD means that videos are more related to each other.
- **OA**: Apart from the visual-base evaluation, we consider the object accuracy (OA) for E-MNIST and E-CLEVR.

---

[1]Amazon Mechanical Turk: https://www.mturk.com/

	S-MNIST			D-MNIST			E-CLEVR			E-JESTER	
	VAD ↓	OA ↑	mIoU ↑	VAD ↓	OA ↑	mIoU ↑	VAD ↓	OA ↑	mIoU ↑	VAD ↓	GA ↑
pix2pix [25]	2.06	96.6	74.3	3.05	87.7	64.1	2.84	80.4	60.5	2.00	8.6
vid2vid [60]	1.30	97.0	88.6	2.30	87.5	77.9	2.21	80.5	69.3	1.62	82.0
E3D-LSTM [61]	1.29	97.8	92.8	2.10	90.4	81.3	2.11	83.1	72.2	1.55	83.6
M³L (Ours)	**1.28**	**99.7**	**93.6**	**1.90**	**93.2**	**84.7**	**1.96**	**84.5**	**78.4**	**1.44**	**89.3**

Table 2. The overall testing results of the baselines and our M³L under the E-MMIST, E-CLEVR, and E-JESTER datasets.

		E-JESTER	
Instruction	MLF	VAD ↓	GA ↑
✗	✗	1.99	4.7
✓	✗	1.50	85.4
✓	✓	**1.44**	**89.3**

Table 3. The ablation results when without the instruction or MLF.

	D-MNIST			E-CLEVR		
MLF	VAD ↓	OA ↑	mIoU ↑	VAD ↓	OA ↑	mIoU ↑
✗	2.64	82.6	73.6	2.32	70.1	66.6
✓	**2.35**	**87.5**	**79.1**	**2.29**	**76.7**	**71.5**

Table 4. Zero-shot generalization under D-MNIST and E-CLEVR.

OA is calculated by the correctness of the presented objects in the target video from a pre-trained object detector[2]. A higher OA shows that the model can edit specific properties of the mentioned object from the instruction.

- **mIoU**: We also evaluate the position of objects for E-MNIST and E-CLEVR via mean Intersection over Union (mIoU) between generated and ground-truth results. mIoU is averaged from each frame in the video also based on the pre-trained object detector[2]. A higher mIoU indicates that the model is able to manipulate the object into the mentioned relative position.
- **GA**: We report the gesture accuracy (GA) for E-JESTER, which is calculated as the gesture classification accuracy of the edited video by MFFs[3]. Although the generated video may not be the same as the ground truth, a higher GA represents that the model is able to follow the guided text and generate the corresponding type of gesture.

Baselines. Since our LBVE is a brand new task, there is no existing baselines. We consider following methods conditioning on an instruction, by concatenating the languistic feature, to carry out LBVE as the compared baselines.

- **pix2pix** [25]: pix2pix is an image-to-image translation approach. For the sake of video synthesis, we process the source video frame-by-frame to perform pix2pix.
- **vid2vid** [60]: vid2vid applies the temporal discriminator for better video-to-video synthesis, which considers several previous frames to model the translation.
- **E3D-LSTM** [61]: E3D-LSTM incorporates 3D CNN into LSTM for video prediction. We treat the source video as the given video and predict the remaining part as the target video.

Implementation Detail. We apply 3-layer ResBlocks [41] into the 3D ResNet and the generator U with kernel size 3 and stride 1 in the first layer. In particular, we incorporate

---

[2] We have more than 99% OA and 95% mIoU of our pre-trained object detector, which can precisely evaluate E-MNIST and E-CLEVR.

[3]MFFs (https://github.com/okankop/MFF-pytorch) has 96% accuracy on JESTER and serves for evaluating E-JESTER.

1-layer self-attention for better frame generation into U following SAGAN [69]. The visual feature dimension C_v is 256 and the language feature dimension C_x is 1024 from RoBERTa [36]. Adam [30] is adopted to optimize through our multi-modal multi-level transformer (M³L) with learning rate 3e-4 for the visual difference loss $\mathcal{L}_E$, and learning rate 1e-4 for $\mathcal{L}_G$ and $\mathcal{L}_D$ from the dual discriminator D.

5.2. Quantitative Results

Table 2 shows the overall testing results compared between the baselines and ours M³L. pix2pix only adopts image-to-image translation, resulting in insufficient output video (*e.g.,* 64.1 mIoU under D-MNIST and 2.84 VAD under E-CLEVR). Even if vid2vid and E3D-LSTM consider temporal consistency, the lack of explicit cross-modal fusion still makes them difficult to perform LBVE. While, our M³L, which incorporates the multi-level fusion (MLF), can fuse between vision-and-language with different levels and surpass all baselines. In particular, M³L achieves the best results across all metrics under all diagnostic datasets (*e.g.,* 99.7 OA under S-MNIST, 84.7 mIoU under D-MNIST, and 1.96 VAD under E-CLEVR).

Similar trends can be found on the natural E-JESTER dataset. pix2pix only has 8.6% GA, which shows that it cannot produce a video with the correct target gesture. Although vid2vid and E3D-LSTM may have similar visual measurement scores to our approach, M³L achieves the highest 89.3% GA. The significant improvement of GA demonstrates that the proposed MLF benefits not only the visual quality but also the semantic of the predicted video and makes it more corresponding to the given instruction.

5.3. Ablation Study

Ablation Results. Table 3 presents the testing results of the ablation setting under E-JESTER. If without the given instruction, the model lacks the specific editing target and results in poor 1.99 VAD and 4.7% GA. The performance comprehensively improves when incorporating our proposed multi-level fusion (MLF) (*e.g.,* VAD from 1.50

	w/ MLF	w/o MLF	Tie
Video Quality	**67.1%**	27.1%	5.8%
Video-Instruction Alignment	**53.3%**	35.1%	11.6%
Siml. to GT Video	**59.6%**	28.9%	11.6%

Table 5. Human evaluation on E-JESTER with aspects of video quality, video-instruction alignment, and similarity to GT video.

down to 1.44 and GA from 85.4% up to 89.3%). The multi-level modeling from MLF benefits not only the understanding between video and instruction, but also leads to accurate frame generation. The above ablation results show that the instruction is essential under the video editing task, and our MLF further helps to perform LBVE.

Zero-Shot Generalization. To further investigate the generalizability of M^3L, we conduct a zero-shot experiment for both the D-MNIST and E-CLEVR datasets. In D-MNIST, there are 40 different object-semantic combinations[4]. We remove out 10 of them in the training set (*e.g.,* number 1 with upper left or number 3 with lower down) and evaluate under the complete testing set. For E-CLEVR, we filter out 12 kinds (*e.g.,* small gray metal sphere or large purple rubber cube) from the total 96 possibilities[5]. This testing scenario is widely used to evaluate new combinations of object-semantic pairs that are not seen during training [8, 17, 18]. The results are shown in Table 4. Due to the lack of object properties or moving semantics, the model has a significant performance drop under the zero-shot settings. While, our proposed MLF helps the property and moving motion for both video perception and generation by multi-modal multi-level fusion. Therefore, MLF still improves the generalizability (*e.g.,* OA from 82.6 up to 87.5 under D-MNIST and mIoU from 66.6 up to 71.5 under E-CLEVR) even if training with the zero-shot examples.

Inference Efficiency. As a video processing task, not only the performance but also the efficiency is important of the editing framework. When using only the CPU, it carries out the E-JSTER with about 11.9 FPS, where the processed frame is 128x128. With the acceleration from the GPU (TITAN X), the model can further achieve 35.8 FPS, which is faster than the real-time requirement (24 FPS). The results show that our M^3L with the multi-level fusion (MLF) can carry out the LBVE task for practical usage efficiently.

Human Evaluation. Apart from the quantitative results, we also investigate the quality of the generated video from the human aspect. Table 5 demonstrates the comparison between without and with MLF. We randomly sample 75 examples and ask three following questions: (1) Which video has better quality; (2) Which video corresponds more to the given instruction; (3) Which video is more similar to the ground-truth target video. Each example is assigned to 3 different MTurkers to avoid evaluation bias. Firstly,

about 67% think that generated videos from MLF have better quality. Moreover, more than 50% of Mturkers denote that the target videos produced from MLF correspond more to the instruction and are also more similar to the ground truth. The results of the human evaluation indicate that our MLF not only helps improve the generating quality but also makes the target video more related to the guided text.

Qualitative Results. Fig. 4 shows the keyframes of the generated examples of LBVE on all three datasets. For E-MNIST, we have to recognize which number should be replaced and which one will change the moving semantic. Note that the instruction only tells the replacing number, but without the style, thus our model replaces with another kind of number 2 under S-MNIST. Under the advanced D-MNIST dataset, our model can replace with the number 8 and move the number 5 along the lower right with multi-level fusion. The challenge of E-CLEVR is to transform object properties and move to the different target positions related to the fixed object. The visualization examples show that our model can understand the linguistic to change the specific object into the correct properties. Also, it has the spatial concept that can perceive the final related position and maintain the moving motion. The E-JESTER dataset, which contains nature video and human-labeled instruction, requires the link of the complex natural language with the human gesture action. The presented video indicates that our model can not only preserve a similar scenario (the background and the person) but also generate the visual motion of the corresponding gesture.

6. Conclusion

We introduce language-based video editing (LBVE), a novel task that allows the model to edit, guided by a natural text, a source video into a target video. We present multi-modal multi-level transformer (M^3L) to dynamically fuse video perception and language understanding at multiple levels. For the evaluation, we release three new datasets containing two diagnostic and one natural video with human-labeled text. Experimental results show that our M^3L is adequate for video editing, and LBVE can bring out a new field toward vision-and-language research.

---

[4]**D-MNIST**: 10 different numbers and 4 different directions
[5]**E-CLEVR**: 3 shapes, 8 colors, 2 materials, and 2 shapes

References

[1] Ding Liu abd Zhaowen Wang, Yuchen Fan, Xianming Liu, Zhangyang Wang, Shiyu Chang, and Thomas Huang. Robust Video Super-Resolution with Learned Temporal Dynamics. In *ICCV*, 2017. 2

[2] Mohammad Babaeizadeh, Chelsea Finn, Dumitru Erhan, Roy H. Campbell, and Sergey Levine. Stochastic Variational Video Prediction. In *ICLR*, 2017. 2

[3] Yogesh Balaji, Martin Renqiang Min, Bing Bai, Rama Chellappa, and Hans Peter Graf. Conditional GAN with Discriminative Filter Generation for Text-to-Video Synthesis. In *IJCAI*, 2019. 1, 2

[4] Bert De Brabandere, Xu Jia, Tinne Tuytelaars, and Luc Van Gool. Dynamic Filter Networks. In *NeurIPS*, 2016. 2

[5] Joao Carreira and Andrew Zisserman. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. In *CVPR*, 2017. 6

[6] Ya-Liang Chang, Zhe Liu Yu, , and Winston Hsu. VORNet: Spatio-temporally Consistent Video Inpainting for Object Removal. In *CVPR WS*, 2019. 2

[7] Devendra Singh Chaplot, Kanthashree Mysore Sathyendra, Rama Kumar Pasumarthi, Dheeraj Rajagopal, and Ruslan Salakhutdinov. Gated-Attention Architectures for Task-Oriented Language Grounding. In *AAAI*, 2018. 4

[8] Devendra Singh Chaplot, Kanthashree Mysore Sathyendra, Rama Kumar Pasumarthi, Dheeraj Rajagopal, and Ruslan Salakhutdinov. Gated-Attention Architectures for Task-Oriented Language Grounding. In *AAAI*, 2018. 8

[9] Dongdong Chen, Jing Liao, Lu Yuan, Nenghai Yu, and Gang Hua. Coherent Online Video Style Transfer. In *ICCV*, 2017. 2

[10] Jianbo Chen, Yelong Shen, Jianfeng Gao, Jingjing Liu, and Xiaodong Liu. Language-Based Image Editing with Recurrent Attentive Models. In *CVPR*, 2018. 2

[11] Ming-Ming Cheng, Shuai Zheng, Wen-Yan Lin, Jonathan Warrell, Vibhav Vineet, Paul Sturgess, Nigel Crook, Niloy Mitra, and Philip Torr. ImageSpirit: Verbal Guided Image Parsing. In *ACM Transactions on Graphics*, 2013. 2

[12] Aidan Clark, Jeff Donahue, and Karen Simonyan. Adversarial Video Generation on Complex Datasets. In *arXiv:1907.06571*, 2019. 2, 5

[13] Yingying Deng, Fan Tang, Weiming Dong, Haibin Huang, Chongyang Ma, and Changsheng Xu. Arbitrary Video Style Transfer via Multi-Channel Correlation. In *AAAI*, 2021. 2

[14] Emily Denton and Rob Fergus. Stochastic Video Generation with a Learned Prior. In *ICML*, 2018. 2

[15] Alaaeldin El-Nouby, Shikhar Sharma, Hannes Schulz, Devon Hjelm, Layla El Asri, Samira Ebrahimi Kahou, Yoshua Bengio, and Graham W.Taylor. Tell, Draw, and Repeat: Generating and Modifying Images Based on Continual Linguistic Instruction. In *ICCV*, 2019. 2

[16] Ohad Fried, Ayush Tewari, Michael Zollhöfer, Adam Finkelstein, Eli Shechtman, Dan B Goldman, Kyle Genova, Zeyu Jin, Christian Theobalt, and Maneesh Agrawala. Text-based Editing of Talking-head Video. In *SIGGRAPH*, 2019. 2

[17] Tsu-Jui Fu, Xin Eric Wang, Scott Grafton, Miguel Eckstein, and William Yang Wang. SSCR: Iterative Language-Based Image Editing via Self-Supervised Counterfactual Reasoning. In *EMNLP*, 2020. 2, 8

[18] Rohit Girdhar and Deva Ramanan. CATER: A diagnostic dataset for Compositional Actions and TEmporal Reasoning. In *ICLR*, 2020. 6, 8

[19] Vincent Le Guen and Nicolas Thome. Disentangling Physical Dynamics from Unknown Factors for Unsupervised Video Prediction. In *CVPR*, 2020. 2

[20] Zekun Hao, Xun Huang, and Serge Belongie. Controllable Video Generation with Sparse Trajectories. In *CVPR*, 2018. 2

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *ICCV*, 2015. 4

[22] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. In *NeurIPS*, 2017. 6

[23] Jun-Ting Hsieh, Bingbin Liu, De-An Huang, Li Fei-Fei, and Juan Carlos Niebles. Learning to Decompose and Disentangle Representations for Video Prediction. In *NeurIPS*, 2018. 2

[24] Qiyang Hu, Adrian Waelchli, Tiziano Portenier, Matthias Zwicker, and Paolo Favaro. Video Synthesis from a Single Image and Motion Stroke. In *arXiv:1812.01874*, 2018. 2

[25] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-Image Translation with Conditional Adversarial Nets. In *CVPR*, 2017. 7

[26] Younghyun Jo, Seoung Wug Oh, Jaeyeon Kang, and Seon Joo Kim. Deep Video Super-Resolution Network Using Dynamic Upsampling Filters Without Explicit Motion Compensation. In *CVPR*, 2018. 2

[27] Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Fei-Fei Li, Larry Zitnick, and Ross Girshick. CLEVR: A Diagnostic Dataset for Compositional Language and Elementary Visual Reasoning. In *CVPR*, 2017. 2, 6

[28] Nal Kalchbrenner, Aaron van den Oord, Karen Simonyan, Ivo Danihelka, Oriol Vinyals, Alex Graves, and Koray Kavukcuoglu. Video Pixel Networks. In *ICML*, 2017. 2

[29] Dahun Kim, Sanghyun Woo, Joon-Young Lee, and In So Kweon. Deep Video Inpainting. In *CVPR*, 2019. 2

[30] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In *ICLR*, 2015. 7

[31] Gierad Laput, Mira Dontcheva, Gregg Wilensky, Walter Chang, Aseem Agarwala, Jason Linder, and Eytan Adar. PixelTone: A Multimodal Interface for Image Editing. In *CHI*, 2013. 2

[32] Yann LeCun, Corinna Cortes, and CJ Burges. MNIST Handwritten Digit Database. 2010. 2, 5

[33] Xuechen Li, Ting-Kam Leonard Wong, Ricky T. Q. Chen, and David Duvenaud. Scalable Gradients for Stochastic Differential Equations. In *AISTATS*, 2020. 2

[34] Yitong Li, Martin Renqiang Min, Dinghan Shen, David Carlson, and Lawrence Carin. Video Generation from Text. In *AAAI*, 2018. 1, 2

[35] Chenxi Liu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, and Alan L Yuille. Recurrent Multimodal Interaction for Referring Image Segmentation. In *ICCV*, 2017. 4

[36] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A Robustly Optimized BERT Pretraining Approach. In *arXiv:1907.11692*, 2019. 3, 7

[37] Ziwei Liu, Raymond A. Yeh, Xiaoou Tang, and Aseem Agarwala Yiming Liu. Video Frame Synthesis using Deep Voxel Flow. In *ICCV*, 2017. 2

[38] Tanya Marwah, Gaurav Mittal, and Vineeth N. Balasubramanian. Attentive Semantic Video Generation using Captions. In *CVPR*, 2017. 1, 2

[39] Joanna Materzynska, Guillaume Berger, Ingo Bax, and Roland Memisevic. The Jester Dataset: A Large-Scale Video Dataset of Human Gestures. In *ICCV WS*, 2019. 6

[40] Michael Mathieu, Camille Couprie, and Yann LeCun. Deep Multi-Scale Video Prediction Beyond Mean Square Error. In *ICLR*, 2016. 2

[41] Takeru Miyato and Masanori Koyama. cGANs with Projection Discriminator. In *ICLR*, 2018. 5, 7

[42] Anh Nguyen, Jeff Clune, Yoshua Bengio, Alexey Dosovitskiy, and Jason Yosinski. Plug & Play Generative Networks: Conditional Iterative Generation of Images in Latent Space. In *CVPR*, 2017. 2

[43] Yingwei Pan, Zhaofan Qiu, Ting Yao, Houqiang Li, and Tao Mei. To Create What You Tell: Generating Videos from Captions. In *ACMMM*, 2017. 1, 2

[44] Yunchen Pu, Martin Renqiang Min, Zhe Gan, and Lawrence Carin. Adaptive Feature Abstraction for Translating Video to Text. In *AAAI*, 2018. 2

[45] Fitsum A. Reda, Guilin Liu, Kevin J. Shih, Robert Kirby, Jon Barker, David Tarjan, Andrew Tao, and Bryan Catanzaro. SDC-Net: Video Prediction using Spatially-Displaced Convolution. In *ECCV*, 2018. 2

[46] Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. Generative Adversarial Text to Image Synthesis. In *ICML*, 2016. 2

[47] Scott Reed, Aäron van den Oord, Nal Kalchbrenner, Sergio Gómez Colmenarejo, Ziyu Wang, Dan Belov, and Nando de Freitas. Parallel Multiscale Autoregressive Density Estimation. In *ICML*, 2017. 2

[48] Masaki Saito, Eiichi Matsumoto, and Shunta Saito. Temporal Generative Adversarial Nets with Singular Value Clipping. In *ICCV*, 2017. 2

[49] Masaki Saito and Shunta Saito. TGANv2: Efficient Training of Large Models for Video Generation with Multiple Subsampling Layers. In *arXiv:1811.09245*, 2018. 2

[50] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved Techniques for Training GANs. In *NeurIPS*, 2016. 6

[51] Seitaro Shinagawa, Koichiro Yoshino, Sakriani Sakti, Yu Suzuki, and Satoshi Nakamura. Interactive Image Manipulation with Natural Language Instruction Commands. In *NeurIPS WS*, 2017. 2

[52] Nitish Srivastava, Elman Mansimov, and Ruslan Salakhutdinov. Unsupervised Learning of Video Representations using LSTMs. In *ICML*, 2015. 2, 5

[53] Fuwen Tan, Song Feng, and Vicente Ordonez. Text2Scene: Generating Compositional Scenes from Textual Descriptions. In *CVPR*, 2019. 2

[54] Sergey Tulyakov, Ming-Yu Liu, Xiaodong Yang, and Jan Kautz. MoCoGAN: Decomposing Motion and Content for Video Generation. In *CVPR*, 2017. 2

[55] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In *NeurIPS*, 2017. 4

[56] Ruben Villegas, Jimei Yang amd Seunghoon Hong, Xunyu Lin, and Honglak Lee. Decomposing Motion and Content for Natural Video Sequence Prediction. In *ICLR*, 2017. 2

[57] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Generating Videos with Scene Dynamics. In *NeurIPS*, 2016. 2

[58] Jacob Walker, Ali Razavi, and Aäron van den Oord. Predicting Video with VQVAE. In *arXiv:2103.01950*, 2021. 2

[59] Ting-Chun Wang, Ming-Yu Liu, Andrew Tao, Guilin Liu, Jan Kautz, and Bryan Catanzaro. Few-shot Video-to-Video Synthesis. In *NeurIPS*, 2019. 2

[60] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Guilin Liu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. Video-to-Video Synthesis. In *NeurIPS*, 2018. 2, 7

[61] Yunbo Wang, Lu Jiang, Ming-Hsuan Yang, Li-Jia Li, Mingsheng Long, and Li Fei-Fei. Eidetic 3D LSTM: A Model for Video Prediction and Beyond. In *ICLR*, 2019. 7

[62] Dirk Weissenborn, Oscar Täckström, and Jakob Uszkoreit. Scaling Autoregressive Video Models. In *ICLR*, 2020. 2

[63] Xide Xia, Tianfan Xue, Wei-Sheng Lai, Zheng Sun, Abby Chang, Brian Kulis, and Jiawen Chen. Real-Time Localized Photorealistic Video Style Transfer. In *WACV*, 2021. 2

[64] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated Residual Transformations for Deep Neural Networks. In *CVPR*, 2017. 6

[65] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, and Yoshua Bengio. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. In *ICML*, 2015. 4

[66] Rui Xu, Xiaoxiao Li, Bolei Zhou, and Chen Change Loy. Deep Flow-Guided Video Inpainting. In *CVPR*, 2019. 2

[67] Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong Hes. AttnGAN: Fine-Grained Text to Image Generation with Attentional Generative Adversarial Networks. In *CVPR*, 2018. 2

[68] Linwei Ye, Mrigank Rochan, Zhi Liu, and Yang Wang. Cross-Modal Self-Attention Network for Referring Image Segmentation. In *CVPR*, 2019. 4

[69] Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-Attention Generative Adversarial Networks. In *PMLR*, 2019. 4, 7

[70] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris Metaxas. StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networkss. In *ICCV*, 2017. 2