G^{3}AN: This video does not exist Disentangling motion and appearance for video generation

Yaohui Wang[1,2] Piotr Bilinski[3] Francois Bremond[1,2] Antitza Dantcheva[1,2]

[1]Inria [2] Université Côte d'Azur [3]University of Warsaw

Abstract

Creating realistic human videos introduces the challenge of being able to simultaneously generate both appearance, as well as motion. To tackle this challenge, we propose the novel spatio-temporal GAN-architecture G^3AN, which seeks to capture the distribution of high dimensional video data and to model appearance and motion in disentangled manner. The latter is achieved by decomposing appearance and motion in a three-stream Generator, where the main stream aims to model spatio-temporal consistency, whereas the two auxiliary streams augment the main stream with multi-scale appearance and motion features, respectively. An extensive quantitative and qualitative analysis shows that our model systematically and significantly outperforms state-of-the-art methods on the facial expression datasets MUG and UvA-NEMO, as well as the Weizmann and UCF101 datasets on human action. Additional analysis on the learned latent representations confirms the successful decomposition of appearance and motion.

1. Introduction

Generative Adversarial Networks (GANs) [9] have witnessed increased attention due to their ability to model complex data distributions, which allows them to *generate* realistic *images* [5, 16, 17, 20, 23, 24, 38, 40], as well as to translate images [2, 14, 27, 30]. While realistic *video generation* is the natural sequel, it is substantially more challenging w.r.t. complexity and computation related to the simultaneous modeling of appearance, as well as motion.

Specifically, in inferring and modeling the distribution of human videos, generative models face three main challenges: (a) generating uncertain motion, (b) retaining of human appearance throughout the generated video, as well as (c) modeling spatio-temporal consistency. Such challenges have been alleviated by conditioning the generation on potent priors such as input images [39], human keypoints [6] and optical flow [22]. This relates to learning to sample from conditional distributions, assuming access to the marginal distributions instead of learning the joint distributions [26].

Deviating from such approaches, in this work we focus on the highly intricate problem of video generation without prior knowledge w.r.t. either appearance or motion. Specifically, based on noise variables, we generate an appearance, *e.g.* human face and body, which we concurrently animate, by a facial expression or human action.

G^3AN, our new generative model, is streamlined to learn a *disentangled representation* of the video generative factors *appearance* and *motion*, allowing for manipulation of both. A disentangled representation has been defined as one, where single latent units are sensitive to changes in single generative factors, while being relatively invariant to changes in other factors [4]. In this context, our G^3AN is endowed with a three-stream Generator-architecture, where the main stream encodes spatio-temporal video representation, augmented by two auxiliary streams, representing the independent generative factors *appearance* and *motion*. A self-attention mechanism targeted towards high level feature maps ensures video quality.

G^3AN is hence able to generate realistic videos (tackling challenges (a) and (c)) by following a training distribution and without additional input, as well as able to manipulate the appearance and motion disjointly, while placing emphasis on preserving appearance (challenge (b)).

The **main contributions** of our work are the following:

- We propose a new spatio-temporal approach, which seeks to learn disentangled representations of the generative factors *appearance* and *motion*. This is achieved by a Generator-architecture, incorporating hierarchical G^3-modules. Each module contains three streams, where two auxiliary streams augment the main spatio-temporal stream.

- The learned disentangled representation allows for individual *manipulation* of appearance and motion.

- We propose a novel *spatio-temporal fusion scheme*, which fuses the feature maps in each G^3-module. The fusion scheme ensures that the output from each auxiliary stream represents *appearance* and *motion*, individually.

- We propose a *factorized spatio-temporal self-attention*, which improves the quality of generated videos.

- We demonstrate qualitatively and quantitatively that G^3AN systematically and significantly outperforms state-of-the-art baselines on a set of datasets.

2. Related Work

Despite the dynamic progress in image generation, the extension to *video* generation is surprisingly challenging. While videos constitute sequences of temporally coherent images, video generation encompasses a majority of challenges that have to do with generation of plausible and realistic appearance, coherent and realistic motion, as well as spatio-temporal consistency. A further challenge, namely the generation of uncertain local or global motion, associated to future uncertainty, allows for multiple correct, equally probable next frames [35]. Finding suitable representation learning methods, which are able to address these challenges is critical. Existing methods include approaches based on Variational Autoencoders (VAEs) [19], autoregressive models, as well as most prominently GANs [9].

While video generation tasks aim at generating realistic temporal dynamics, such tasks vary with the *level of conditioning*. We have video generation based on additional priors related to motion or appearance, as well as contrarily, video generation following merely the training distribution. We note that the latter is more challenging from a modeling perspective, due to lack of additional input concerning *e.g.* structure of the generated video and therefore the majority of approaches include a conditioning of some kind.

Video generation with additional input. Due to difficulty of modeling high dimensional video data, additional information such as semantic maps [25, 37], human key points [15, 39, 36, 6], 3D face mesh [41] and optical flow [22] can be instrumental as guidance for appearance and motion generation. This additional information is either pre-computed throughout the generated video [15, 41, 6] or predicted based on an initial input image [39]. The additional information guides conditional image translation, which though results in lack of modeling of spatio-temporal correlations.

Video generation from noise. Directly generating videos from noise requires the capturing and modeling of a dataset distribution. Existing works usually reduce the complexity of this task by decomposing either the output [34] or latent representation [28, 33]. VGAN [34] was equipped with a two-stream spatio-temporal Generator, generating foreground and background separately. TGAN [28] decomposed the latent representation of each frame into a *slow part* and a *fast part*. Due to jointly modeling appearance and motion, generated results from VGAN and TGAN might comprise spatial unrealistic artefacts, see Figure 5. The closest work to ours is MoCoGAN [33], which decomposes the latent representation of *each frame* into motion and content, aiming at controlling both. However, there are two crucial differences between MoCoGAN and G^3AN. Firstly, instead of only sampling two noise vectors for each video, MoCoGAN samples a sequence of noise vectors as motion and a fixed noise as content. However, involving random noise for each frame to represent motion increases the learning difficulty, since the model has to map these noise vectors to a consecutive human movement in the generated videos. As a result, MoCoGAN gradually ignores the input noise and tends to produce a similar motion, as we illustrate in Figure 8. Secondly, MoCoGAN incorporates a simple image Generator aiming at generating each frame sequentially, after which content and motion features are jointly generated. This leads to *incomplete disentanglement* of motion and content. Deviating from that, we design a novel Generator architecture, able to entirely decompose appearance and motion in both, latent and feature spaces. We show that such design can generate realistic videos of good quality and ensure factor disentanglement.

Disentangled representation learning. Learning a disentangled representation of data generative factors has been shown beneficial for a large variety of tasks and domains [4]. Disentangling a number of factors in *still images* have been widely explored in recent years [7, 23, 31, 21]. In the context of *video generation*, an early approach for motion and appearance disentanglement was incorporated in MoCoGAN, however experiments, presented later in this work (see Figure 6) suggest that the results are not satisfactory.

3. Approach

In this work, we propose G^3AN, a novel GAN architecture, aiming at generating videos in a disentangled manner from two noise vectors, $z_a \in Z_A$ and $z_m \in Z_M$, which represent appearance and motion, respectively. G^3AN consists of a three-stream Generator G, as well as a two-stream Discriminator D, as illustrated in Figure 1. G aims at generating videos with the ability to modulate appearance and motion disjointly, while D accounts for distinguishing generated samples from real data, in both, videos and frames, respectively.

3.1. Generator

Hierarchical Generator with $\mathbf{G^3}$-modules. We design the Generator G in a hierarchical structure of G^3 modules. Specifically, we have N levels of hierarchy, denoted as $G^3_{n=0...N-1}$. The first G^3 module, G^3_0 accepts as inputs the noise vectors z_a and z_m. The remaining modules $G^3_{n=1...N-1}$, inherit the three feature maps $F_{S_{n-1}}$, $F_{V_{n-1}}$ and $F_{T_{n-1}}$ as their inputs from each previous G^3_{n-1} module, see Figures 1 and 2.

Each G^3_n module consists of three parallel streams: a spatial stream G_{S_n}, a temporal stream G_{T_n}, as well as a video stream G_{V_n} (Figures 1 and 2). These streams are de-

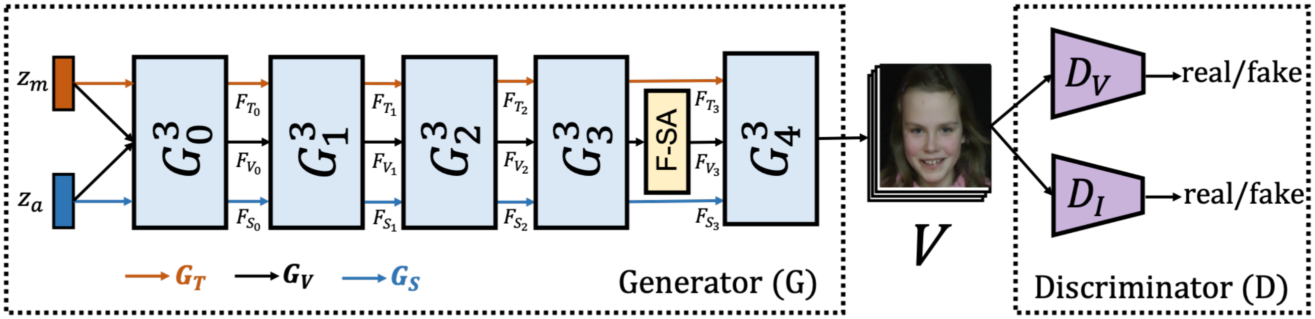


Figure 1: **Overview of our G^{3}AN architecture.** It is a fully convolutional GAN that aims to generate realistic video sequences. It consists of a three-stream Generator and a two-stream Discriminator. The Generator has 5 stacked G^3 modules, a factorized self-attention (F-SA) mechanism, and takes as input two random noise-vectors, z_a and z_m, aiming at controlling appearance and motion, respectively.

signed to generate three different types of features. The spatial stream G_{S_n}, denoted by a blue line in Figures 1 and 2, takes as input z_a for $n = 0$ and $F_{S_{n-1}}$ for $n > 1$, and generates 2D appearance features F_{S_n} by upsampling input features with a transposed 2D convolutional layer. These features evolve in spatial dimension and are shared at all time instances. The temporal stream G_{T_n}, denoted by an orange line, accepts as input z_m for $n = 0$ and $F_{T_{n-1}}$ for $n > 1$, and seeks to generate 1D motion features F_{T_n} by upsampling input features with a transposed 1D convolutional layer. These features evolve in temporal dimension and contain global information of each time step. Then, the video stream G_{V_n}, denoted by a black line, takes as input both z_a and z_m for $n = 0$ and $F_{V_{n-1}}$ for $n > 1$. It models spatio-temporal consistency and produces 3D joint embeddings $F_{V'_n}$ by upsampling input features with a factorized transposed spatio-temporal convolution, see below. Then, F_{S_n} and F_{T_n} are catapulted to the spatio-temporal fusion block, where they are fused with $F_{V'_n}$, resulting in F_{V_n}. Finally, F_{S_n}, F_{T_n} and F_{V_n} serve as inputs of the next hierarchy-layer G^3_{n+1}.

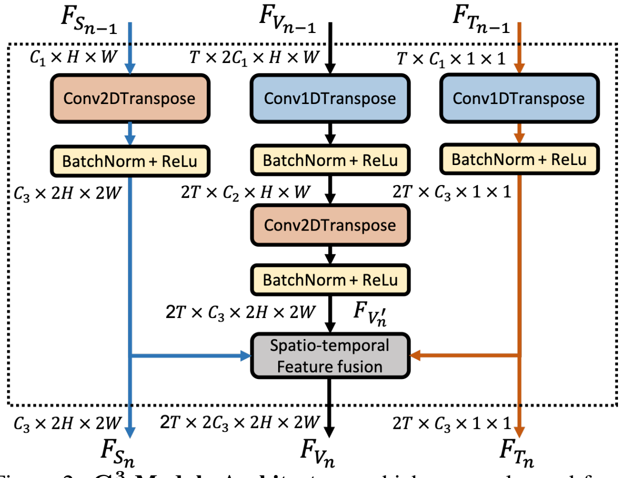

$F_{S_{n-1}}$
$F_{V_{n-1}}$
$F_{T_{n-1}}$
$C_1 \times H \times W$
$T \times 2C_1 \times H \times W$
$T \times C_1 \times 1 \times 1$
Conv2DTranspose
Conv1DTranspose
Conv1DTranspose
BatchNorm + ReLu
BatchNorm + ReLu
BatchNorm + ReLu
$C_3 \times 2H \times 2W$
$2T \times C_2 \times H \times W$
$2T \times C_3 \times 1 \times 1$
Conv2DTranspose
BatchNorm + ReLu
$2T \times C_3 \times 2H \times 2W$
$F_{V'_n}$
Spatio-temporal Feature fusion
$C_3 \times 2H \times 2W$
$2T \times 2C_3 \times 2H \times 2W$
$2T \times C_3 \times 1 \times 1$
F_{S_n}
F_{V_n}
F_{T_n}

Figure 2: **G^3 Module Architecture**, which upsamples and fuse features together.

Factorized transposed spatio-temporal convolution. We propose to explicitly factorize transposed 3D convolutional filters into two separate and successive operations, M transposed 1D temporal convolutional filters followed by a 2D separate spatial components, which we refer to as transposed (1+2)D convolution. Such decomposition brings an additional nonlinear rectification between these two operations and facilitates optimization. Crucially, factorizing the transposed 3D convolutional filters yields significant gains in video quality, see Section 4.

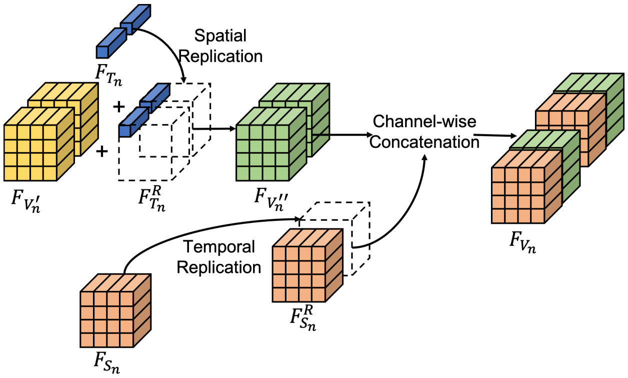


Figure 3: **Spatio-temporal fusion.** The upsampled feature maps F_{S_n}, F_{T_n} and $F_{V'_n}$ are fused in space and time. F_{T_n} is firstly spatially replicated to $F^R_{T_n}$ and added to the $F_{V'_n}$ in a position-wise manner to obtain $F_{V''_n}$. Then, we replicate F_{S_n} to $F^R_{S_n}$ in temporal dimension. Finally, $F_{V''_n}$ is channel-wise concatenated with $F^R_{S_n}$, resulting in the final feature map F_{V_n}.

Spatio-temporal fusion is a key-element in each G^3 module and hence our architecture, as it allows to learn well disentangled features. We propose a simple and effective way to combine F_{S_n}, F_{T_n} and $F_{V'_n}$, denoting feature maps obtained from the transposed convolution layers of G_{S_n}, G_{T_n} and G_{V_n}, respectively. We note that $F_{V'_n}$ has the same temporal dimension as F_{T_n}, as well as the same spatial dimension as F_{S_n}. Therefore, firstly, we perform spatial replication of F_{T_n} and temporal replication of F_{S_n}, in order to obtain two new feature maps $F^R_{T_n}$ and $F^R_{S_n}$, respectively.

These new feature maps are of the same dimension as $F_{V'_n}$. Next, $F^R_{T_n}$ and $F_{V'_n}$ are combined through a position-wise addition, creating a new spatio-temporal embedding $F_{V''_n}$. Finally, we apply channel-wise concatenation of $F^R_{S_n}$ and $F_{V''_n}$, obtaining the final fused feature map F_{V_n}. The features maps F_{S_n}, F_{T_n} and F_{V_n} represent inputs for each following G^3_{n+1} module.

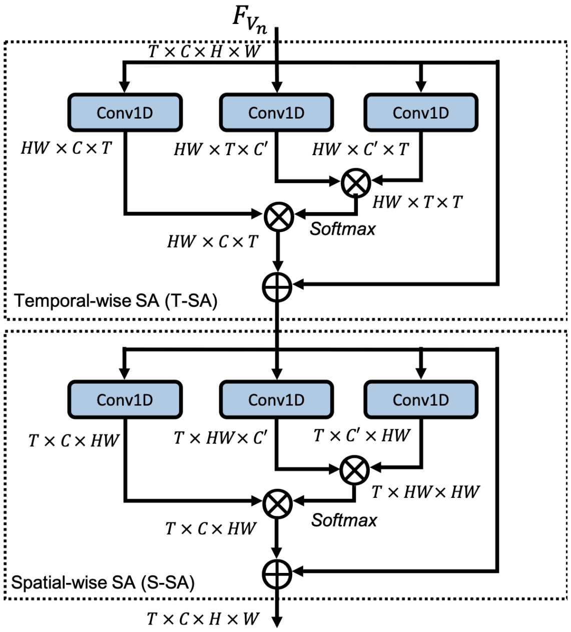


Figure 4: **Factorized spatio-temporal Self-Attention (F-SA).**

Factorized spatio-temporal Self-Attention (F-SA). Self-Attention has been successfully used for image generation. However, it has not been tested in the context of video generation.

In this paper, we study a module of Self-Attention (SA), enabling our G to utilize cues from all spatio-temporal feature positions and efficiently model relationships between widely separated regions. However, such spatio-temporal self-attention requires heavy computation processing, in particular if it is used on larger feature maps of G. Therefore, we propose to factorize spatio-temporal self-attention, as shown in Figure 4. The factorized spatio-temporal SA (F-SA) consists of a Temporal-wise SA (T-SA), followed by a Spatial-wise SA (S-SA) mechanism. Such factorization reduces the complexity, allowing for application of the F-SA on larger feature maps.

In our G^3AN architecture, we apply the F-SA mechanism on the output of the G^3_3 in the G_V stream, which has shown to improve the quality of generated videos. We refer to Section 4 for evaluation of our architecture, where we apply the proposed self-attention mechanism at various hierarchy-layers of the G^3AN.

3.2. Discriminator

Towards improving both video and frame quality, we use a two-stream *Discriminator* architecture, containing a video stream D_V and an image stream D_I, similarly to [33]. While D_V is based on five 3D convolutional layers, D_I contains five 2D convolutions. During training, D_V accepts a full video as input, and D_I randomly samples frames from videos.

3.3. Training

Our objective can be expressed as

$$\mathcal{L}(G, D_I, D_V) = \mathcal{L}_I(G, D_I) + \mathcal{L}_V(G, D_V), \quad (1)$$

where $\mathcal{L}_I$ denotes the loss function related to D_I, $\mathcal{L}_V$ represents the loss function related to D_V,

$$\begin{aligned}\mathcal{L}_I =& \mathbb{E}_{x' \sim p_{data}}[log(D_I(x^{'}))] \\ &+ \mathbb{E}_{z_a \sim p_{z_a}, z_m \sim p_{z_m}}[1 - log(D_I(G(z_a, z_m)^{'}))], \quad (2)\\ \mathcal{L}_V =& \mathbb{E}_{x \sim p_{data}}[log(D_V(x))] \\ &+ \mathbb{E}_{z_a \sim p_{z_a}, z_m \sim p_{z_m}}[1 - log(D_V(G(z_a, z_m)))], \quad (3)\end{aligned}$$

where G attempts to generate videos from z_a and z_m, while D_I and D_V aim to distinguish between generated samples and real samples, *i.e.* $G^* = \min_G \max_{D_I, D_V} \mathcal{L}(G, D_I, D_V)$. $(\cdot)^{'}$ characterizes that T frames are being sampled from real and generated videos.

4. Experiments

Experimental Setup. We use PyTorch to implement our model. The entire network is trained end-to-end with the standard back-propagation algorithm on 4 NVIDIA GTX 1080Ti GPUs. We employ ADAM optimizer [18] with β_1=0.5 and β_2=0.999. We set the learning rate to $2e^{-4}$ for both G and D, and we use the batch size of 128, *i.e.* 32 per GPU, to process more samples during one iteration, as presented by Brock *et al.* [5]. Dimensions of latent representations have been set to 128 for z_a and 10 for z_m. We set $N = 5$ in order to generate videos of 16 frames with scale 64×64. We refer to the Supplementary Material (SM) for more details.

4.1. Datasets

We evaluate our method on following four datasets.

Facial expression datasets. The **MUG** Facial Expression dataset [1] contains 1254 videos of 86 subjects, performing 6 facial expressions, namely *happy*, *sad*, *surprise*, *anger*, *disgust* and *fear*. The **UvA-NEMO** Smile dataset [8] comprises 1240 video sequences of 400 smiling individuals, with 1 or 2 videos per subject. We pre-process videos of the facial expression datasets similarly to previous methods by detecting faces using OpenFace [3] and creating videos around them.

Action recognition datasets. The **Weizmann** Action dataset [10] consists of 90 videos of 9 subjects, performing 10 actions such as *wave* and *bend*. We augment the dataset by horizontally flipping the existing videos. The **UCF101** dataset [32] contains 13,320 videos of 101 human action classes. Similarly to TGAN [28], we scale each frame to 85×64 and crop the central 64×64 regions for learning.

In all our experiments, we sample video frames with a time step ranging between 1 and 4 randomly for data augmentation, and scale each frame to 64×64 pixels.

4.2. Experimental Results

We test our method both quantitatively and qualitatively, providing results on four experiments. Specifically, firstly we evaluate and compare videos generated from G^3AN, VGAN, TGAN and MoCoGAN, quantitatively and qualitatively on all four datasets. Next, we test *conditional* and *unconditional* video generation, where we aim to demonstrate the effectiveness of the proposed decomposition method. Then, we manipulate the latent representation, providing insight into each dimension of the two representations. We proceed to add appearance vectors and study the latent representation. Finally, we conduct an ablation study, verifying the effectiveness of our proposed architecture.

4.2.1 Quantitative Evaluation

We compare G^3AN with three state-of-the-art methods, namely VGAN, TGAN, as well as MoCoGAN. We report two evaluation metrics on the above four datasets. In particular, we extend the two most commonly used metrics in image generation, the Inception Score (IS) [29] and Fréchet Inception Distance (FID) [12], into video level by using a pre-trained 3D CNN [11] as our feature extractor, similar to Wang *et al.* [37].

The **Video FID** grasps both visual quality and temporal consistency of generated videos. It is calculated as $\|\mu - \widetilde{\mu}\|^2 + Tr(\Sigma + \widetilde{\Sigma} - 2\sqrt{\Sigma\widetilde{\Sigma}}))$, where μ and Σ represent the mean and covariance matrix, computed from real feature vectors, respectivelye, and $\widetilde{\mu}$, and $\widetilde{\Sigma}$ are computed from generated data. Lower Video FID scores indicate a superior quality of generated videos.

The **Video Inception Score** captures the quality and diversity of generated videos. It is calculated as $\exp(\mathbb{E}_{x\sim p_g} D_{KL}(p(y|x)\|p(y)))$, where $p(y|x)$ and $p(y)$ denote conditional class distribution and marginal class distribution, respectively. A higher IS indicates better model performance.

We report video FID on MUG, UVA-Nemo and Weizmann datasets, and both video FID and video IS on UCF101. We do not report IS on the other datasets, since we use a feature extractor pre-trained from kinetics-600, finetuned on UCF101; whereas IS can only be reported when GAN-architecture, as well as feature extractor are trained on the same dataset. We generate 5000 videos per dataset for each method for a fair comparison.

Comparison results among different methods are reported in Table 1. Our method consistently achieves the lowest video FID on all datasets, suggesting that videos generated by G^3AN entail both, best temporal consistency and visual quality. At the same time, the obtained highest video IS on UCF101 indicates that our method is able to provide the most diverse samples among all compared methods. Such evaluation results show that proposed decomposition method allows for controlling the generated samples, and additionally facilitates the spatio-temporal learning of generating better quality videos. Generated samples are illustrated in Figure 5.

	MUG	**UvA**	**Weizmann**	**UCF101**	
	FID ↓	FID ↓	FID ↓	FID ↓	IS ↑
VGAN	160.76	235.01	158.04	115.06	2.94
TGAN	97.07	216.41	99.85	110.58	2.74
MoCoGAN	87.11	197.32	92.18	104.14	3.06
G^3AN	**67.12**	**119.22**	**86.01**	**91.21**	**3.62**

Table 1: **Comparison with the state-of-the-art** on four datasets w.r.t. FID and IS.

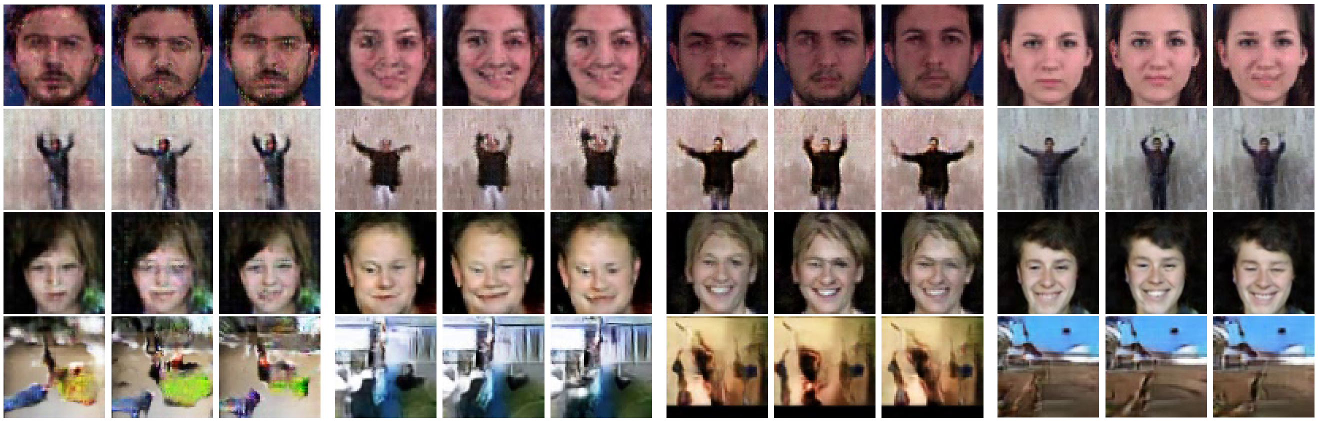
In addition, we conduct a subjective analysis, where we ask 27 human raters to pairwise compare videos generated by our approach with those generated by the state-of-the-art methods. We report the mean user preference in Table 2. We observe that human raters express a strong preference for the proposed framework G^3AN over MoCoGAN (84.26% *vs.* 15.74%), TGAN (87.31% *vs.* 12.69%) and VGAN (90.24% *vs.* 9.76%), which is consistent with the above listed quantitative results. Further, we compare real videos from all datasets with the generated video sequences from our method. The human raters ranked 25.71% of videos from our G^3AN as more realistic than real videos, which we find highly encouraging.

Methods	**Rater preference (%)**
G^3AN / MoCoGAN	**84.26** / 15.74
G^3AN / TGAN	**87.31** / 12.69
G^3AN / VGAN	**90.24** / 9.76
G^3AN / Real videos	25.71 / **74.29**

Table 2: Mean user preference of human raters comparing videos generated by the respective algorithms, originated from all datasets.

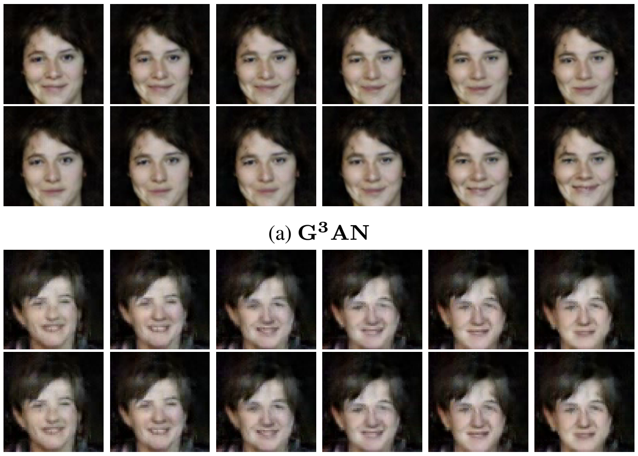
4.2.2 Qualitative Evaluation

We conduct an **unconditional generation**-experiment utilizing the Uva-NEMO dataset, where we fix z_a and proceed to randomly vary motion, z_m. Associated generated samples from G^3AN and MoCoGAN are shown in Figure 6. While we observe the varying motion in the video sequences generated by G^3AN, the appearance remains constant. Hence, our model is able to successfully preserve fa-



(a) **VGAN** (b) **TGAN** (c) **MoCoGAN** (d) **G^{3}AN**

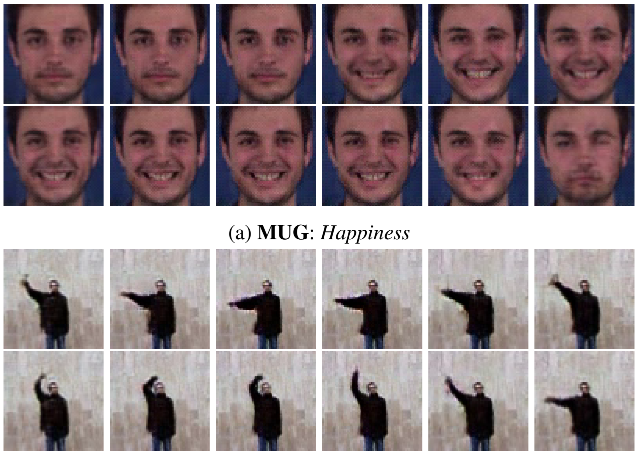
Figure 5: **Comparison of G^{3}AN with the state-of-the-art** on MUG (top-left), Weizmann (top-right), UvA-NEMO (down-left) and UCF101 (down-right). More samples are presented in the Supplementary Material.



(a) **G^{3}AN**

(b) **MoCoGAN**

Figure 6: **Unconditional video generation** of G^3AN and MoCoGAN on Uva-Nemo. For each approach, we fix z_a, while testing two z_m instances (top and bottom lines). See the Supplementary Material (SM) for additional generated samples.



(a) **MUG**: *Happiness*

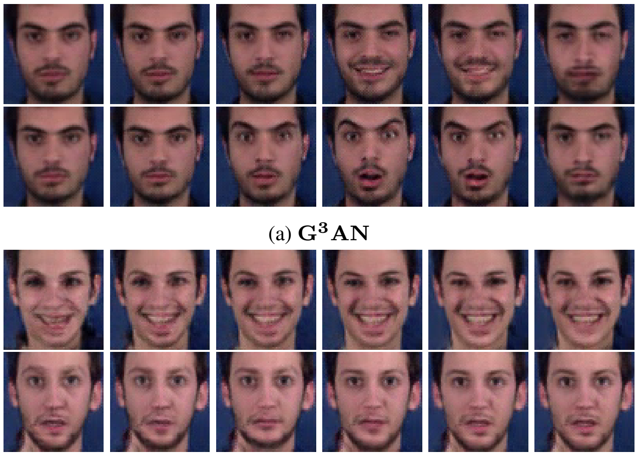
(b) **Weizmann**: *One hand waving*

Figure 7: **Conditional video generation** on MUG and Weizmann. For both datasets, each line is generated with random z_m. We observe that same category (*smile* and *one hand waving*) is performed in a different manner, which indicates that our method is able to produce *intra-class* generation. See **SM** for more samples.

cial appearance, while *altering* the motion. Therefore, this suggests that our three-stream design allows for manipulation of appearance and motion separately. On the contrary, video sequences generated by MoCoGAN experience constant motion, despite of altering z_m.

Further, we leverage on labels of the MUG and Weizmann datasets, in order to analyze **conditional video generation**. Towards this, we here concatenate a one-hot category vector and motion noise z_m, feeding it into G_T. We note that the inputs of G_S and G_V remain the same as in the unconditional generation-setting. Related results show that when varying motion-categories, while having a fixed appearance, G^3AN correctly generates an identical facial appearance, with appropriate category-based motion (facial expressions and body actions), see Figure 7. Further, we note that appearance is very well preserved in different videos and is not affected by category-alterations. In addition, in the same conditional setting, we note that when varying the noise z_m, G^3AN is able to generate *the same category-motion in different ways*. This indicates that z_m enables an intra-class diversity.

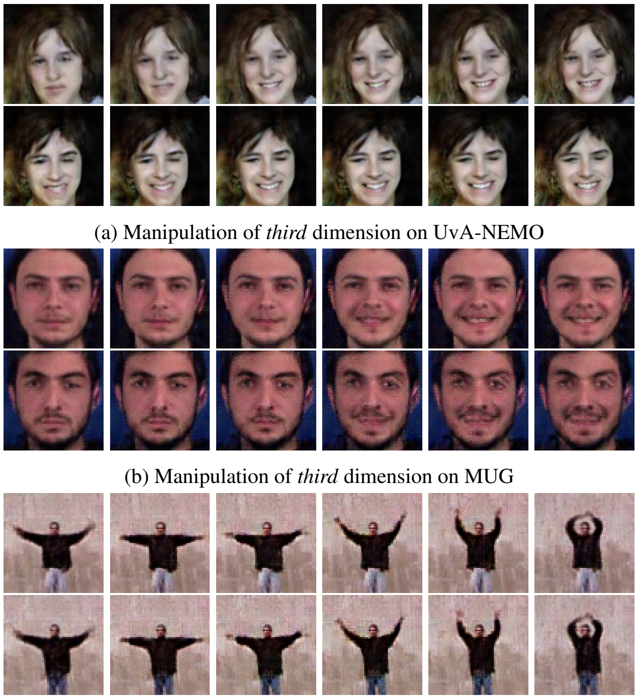
In videos generated by MoCoGAN, we observe a correctly generated motion according to given categories, however we note that the category also affects the appearance. In other words, MoCoGAN lacks a complete disentanglement of appearance and motion in the latent representation, see Figure 8. This might be due to a simple motion and content decomposition in the latent space, which after a set of convolutions can be totally ignored in deeper layers. It is notable that G^3AN effectively prevents such cases, ensured by our decomposition that occurs in both, latent and feature



(a) $\mathbf{G^3AN}$

(b) **MoCoGAN**

Figure 8: **Comparison between G^{3}AN and MoCoGAN.** Given fixed z_a and z_m, as well as two condition-labels *smile* and *surprise*, G^3AN and MoCoGAN generate correct facial expressions. However, while G^3AN preserves the appearance between rows, MoCoGAN alters the subject's appearance.

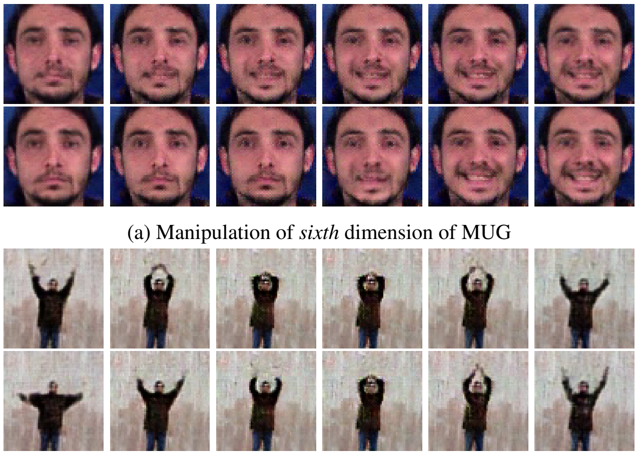


(a) Manipulation of *third* dimension on UvA-NEMO

(b) Manipulation of *third* dimension on MUG

(c) Manipulation of *second* dimension on Weizmann

Figure 9: **Latent appearance representation manipulation.** For each dataset, each row shares the same motion representation, whereas from top to bottom values in one dimension of appearance representation are increased. See the **SM** for more results.



(a) Manipulation of *sixth* dimension of MUG

(b) Manipulation of *second* dimension on Weizmann

Figure 10: **Latent motion representation manipulation.** For each dataset, each row shares the same appearance representation, whereas from top to bottom values in one dimension of the motion representation are increased. See the **SM** for more results.

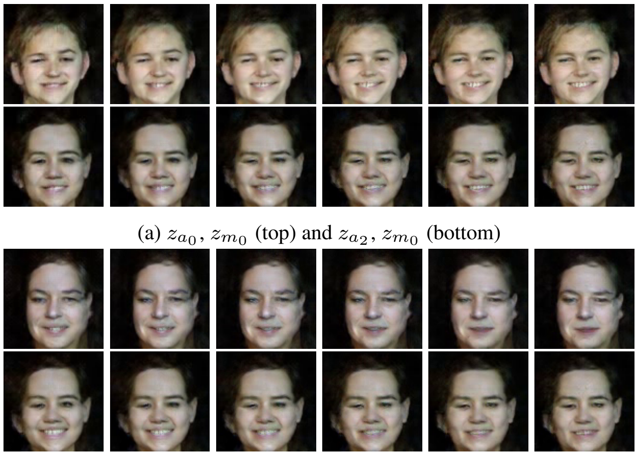
spaces.

Latent representation manipulation. While there is currently no general method for quantifying the degree of learnt disentanglement [13], we proceed to illustrate the ability of our model to learn latent representations by manipulating each dimension in the appearance representation. We show that by changing *dimensions* of the *appearance representation*, we are able to cause a modification of specific appearance factors, see Figure 9. Interestingly such factors can be related to semantics, *e.g.* facial view point in Figure 9a, mustache in Figure 9b, and color of pants in Figure 9c. We note that motion is not affected by altering the appearance-dimensions. Similarly, when altering *dimensions* in the *motion representation*, we observe that factors such as starting position, motion intensity and moving trajectory are being affected, see Figure 10. Such observations show that our method learns to interpolate between different data points in motion- and appearance-latent spaces, respectively.

Addition of appearance representations. We here *add* appearance vectors, aiming to analyze the resulting latent representations. Towards this, we generate two videos V_a and V_b by randomly sampling two sets of noises, (z_{a_0}, z_{m_0}) and (z_{a_1}, z_{m_1}). Next, we add z_{a_0} and z_{a_1}, obtaining a new appearance z_{a_2}. When combining (z_{a_2}, z_{m_0}) and (z_{a_2}, z_{m_1}), we observe in the two new resulting videos a *summary appearance* pertaining to z_{a_0} and z_{a_1}, with identical motion as z_{m_0} and z_{m_1}, see Figure 11.

4.2.3 Ablation Study

We here seek to study the effectiveness of the G^3AN-architecture, as well as the effectiveness related to each component in the proposed Generator. Towards this, we firstly generate videos by removing G_S and G_T, respectively, in order to verify their ability of controlling motion and appearance. We observe that in removing G_T, our method is able generate different subjects, however for

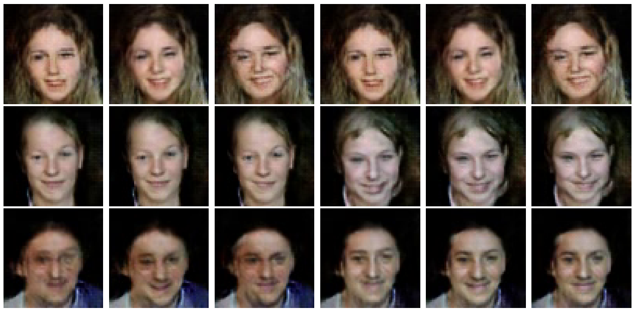


(a) z_{a_0}, z_{m_0} (top) and z_{a_2}, z_{m_0} (bottom)

(b) z_{a_1}, z_{m_1} (top) and z_{a_2}, z_{m_1} (bottom)

Figure 11: **Addition of appearance representations.** We add the appearance vectors of two samples (top rows of (a) and (b)), and obtain the sum-appearance in each bottom row. We inject motion pertained to each top appearance of (a) and (b) and are able to show same motion within lines of (a) and (b).

each person the facial movement is constant, see Figure 12 (*top*). Similarly, when G_S is removed, changing motion will affect subject's identity, whereas the appearance vector loses its efficacy, see Figure 12 (*middle*). When removing both, G_T and G_S, appearance and motion are entangled and they affect each other, see Figure 12 (*bottom*). This demonstrates the effective disentanglement brought to the fore by the streams G_S and G_T.



(a) z_a, z_{m_0}

(b) z_a, z_{m_1}

Figure 12: **Ablation study.** Generated videos obtained by removing G_T (*top row*), removing G_S (*middle*), and both (*bottom row*).

We proceed to demonstrate the contribution of G_S, G_T and F-SA in the Generator w.r.t. video quality. In this context, we remove each component individually and report results on the four datasets in Table 3. The results show that after removing all three components, video quality is the poorest, which proves that all of them contribute to the final results. Individually, G_S plays the most pertinent role, as removing it, decreases FID most profoundly for all datasets. This indicates that generating appearance features separately can be instrumental for good quality videos. Moreover, our results confirm the necessity to use the F-SA module in our approach.

Architecture	**MUG**	**UvA**	**Weizmann**	**UCF101**	
	FID ↓	FID ↓	FID ↓	FID ↓	IS ↑
w/o G_S,G_T,F-SA	117.10	164.04	252.97	127.09	2.78
w/o G_S,G_T	113.44	159.54	176.73	120.17	3.16
w/o G_S	109.87	129.84	141.06	117.19	3.05
w/o F-SA	85.11	128.14	97.54	98.37	3.44
w/o G_T	82.07	121.87	94.64	96.47	3.16
G^3AN	67.12	119.22	86.01	91.21	3.62

Table 3: **Ablation study.** Contribution of main components in G.

Transposed Convolutions. Then, we compare the proposed factorized transposed spatio-temporal convolution ($(1+2)D$), standard transposed 3D convolution, and transposed $(2+1)D$ convolution, when used in G_V w.r.t. video quality. We carefully set the number of kernels, allowing for the three networks to have nearly same training parameters. We report the results of the quantitative evaluation in Table 4. Both convolution types, $(1+2)D$ and $(2+1)D$ outperform standard 3D kernels w.r.t. generated video quality. $(1+2)D$ is slightly better than $(2+1)D$, and the reason might be that the $(1+2)D$ kernel uses more 1×1 kernels to refine temporal information, which we believe to be important in video generation tasks.

	MUG	**UvA**	**Weizmann**	**UCF101**	
	FID ↓	FID ↓	FID ↓	FID ↓	IS ↑
3D	93.51	149.98	154.21	117.61	2.88
$(2+1)D$	73.08	141.35	95.01	98.70	**3.36**
$(1+2)D$	**69.42**	**140.42**	**87.04**	**96.79**	3.07

Table 4: **Comparison of various convolution types in G**.

Where to insert self-attention? Finally, we proceed to explore at which level of the Generator, the self-attention module F-SA is the most effective. We summarize performance rates in Table 5. Inserting the self-attention after the G_4 module provides the best results, which indicates that *high level feature maps* contribute predominantly to video quality. We note that high level feature maps generally correspond to larger spatio-temporal dimensions, which brings to the fore a need for higher computation complexity in the model.

	MUG	**UvA**	**Weizmann**	**UCF101**	
	FID ↓	FID ↓	FID ↓	FID ↓	IS ↑
G_0^3	83.01	188.60	96.38	100.37	3.09
G_1^3	72.54	178.64	99.66	126.12	2.74
G_2^3	69.02	160.12	97.53	112.36	3.03
G_3^3	**67.12**	**119.22**	**86.01**	**91.21**	**3.62**

Table 5: **Attention mechanism F-SA, inserted at different hierarchical levels of $\mathbf{G^3}$AN.** G_n^3 indicates that the F-SA was inserted after the n^{th} G^3 module in the Generator.

5. Conclusions

We have presented the novel video generation architecture G^3AN, which leverages among others on (i) a three-stream Generator modeling appearance and motion in disentangled manner, as well as (ii) a novel spatio-temporal fusion method. We have performed an extensive evaluation of our approach on 4 datasets, outperforming quantitatively and qualitatively the state-of-the-art video generation methods VGAN, TGAN and MoCoGAN. Further, we have shown the ability of G^3AN to disentangle appearance and motion, and hence manipulate them individually.

Public release. We intend to release our *source code*, as well as trained *models*. Moreover, we will release the *dataset* of videos, generated by G^3AN, VGAN, TGAN and MoCoGAN, in order to facilitate research on identifying fake videos.

References

[1] Niki Aifanti, Christos Papachristou, and Anastasios Delopoulos. The mug facial expression database. In *Workshop on Image analysis for multimedia interactive services (WIAMIS)*, pages 1–4. IEEE, 2010. 4

[2] Badour AlBahar and Jia-Bin Huang. Guided image-to-image translation with bi-directional feature transformation. In *ICCV*, 2019. 1

[3] Brandon Amos, Bartosz Ludwiczuk, and Mahadev Satyanarayanan. Openface: A general-purpose face recognition library with mobile applications. Technical report, CMU-CS-16-118, CMU School of Computer Science, 2016. 4

[4] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE TPAMI*, 35(8):1798–1828, 2013. 1, 2

[5] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *ICLR*, 2019. 1, 4

[6] Caroline Chan, Shiry Ginosar, Tinghui Zhou, and Alexei A Efros. Everybody dance now. In *ICCV*, 2019. 1, 2

[7] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *NIPS*, 2016. 2

[8] Hamdi Dibeklioğlu, Albert Ali Salah, and Theo Gevers. Are you really smiling at me? spontaneous versus posed enjoyment smiles. In *ECCV*, 2012. 4

[9] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, 2014. 1, 2

[10] Lena Gorelick, Moshe Blank, Eli Shechtman, Michal Irani, and Ronen Basri. Actions as space-time shapes. *TPAMI*, 29(12):2247–2253, December 2007. 5

[11] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Can Spatiotemporal 3D CNNs Retrace the History of 2D CNNs and ImageNet? In *CVPR*, 2018. 5

[12] 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 *NIPS*. 2017. 5

[13] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. *ICLR*, 2017. 7

[14] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-Image Translation with Conditional Adversarial Networks. In *CVPR*, 2017. 1

[15] Yunseok Jang, Gunhee Kim, and Yale Song. Video Prediction with Appearance and Motion Conditions. In *ICML*, 2018. 2

[16] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017. 1

[17] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, 2019. 1

[18] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 4

[19] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *ICLR*, 2014. 2

[20] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew P Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *CVPR*, 2017. 1

[21] Hsin-Ying Lee, Hung-Yu Tseng, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang. Diverse image-to-image translation via disentangled representations. In *ECCV*, 2018. 2

[22] Yijun Li, Chen Fang, Jimei Yang, Zhaowen Wang, Xin Lu, and Ming-Hsuan Yang. Flow-grounded spatial-temporal video prediction from still images. In *ECCV*, 2018. 1, 2

[23] Liqian Ma, Qianru Sun, Stamatios Georgoulis, Luc Van Gool, Bernt Schiele, and Mario Fritz. Disentangled person image generation. In *CVPR*, 2018. 1, 2

[24] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *ICLR*, 2018. 1

[25] Junting Pan, Chengyu Wang, Xu Jia, Jing Shao, Lu Sheng, Junjie Yan, and Xiaogang Wang. Video generation from single semantic label map. *arXiv preprint arXiv:1903.04480*, 2019. 2

[26] Yunchen Pu, Shuyang Dai, Zhe Gan, Weiyao Wang, Guoyin Wang, Yizhe Zhang, Ricardo Henao, and Lawrence Carin. Jointgan: Multi-domain joint distribution learning with generative adversarial nets. *arXiv preprint arXiv:1806.02978*, 2018. 1

[27] Andrés Romero, Pablo Arbeláez, Luc Van Gool, and Radu Timofte. Smit: Stochastic multi-label image-to-image translation. In *ICCV Workshops*, 2019. 1

[28] Masaki Saito, Eiichi Matsumoto, and Shunta Saito. Temporal generative adversarial nets with singular value clipping. In *ICCV*, 2017. 2, 5

[29] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, Xi Chen, and Xi Chen. Improved techniques for training GANs. In *NIPS*. 2016. 5

[30] Md Mahfuzur Rahman Siddiquee, Zongwei Zhou, Nima Tajbakhsh, Ruibin Feng, Michael B Gotway, Yoshua Bengio, and Jianming Liang. Learning fixed points in generative adversarial networks: From image-to-image translation to disease detection and localization. In *ICCV*, 2019. 1

[31] Krishna Kumar Singh, Utkarsh Ojha, and Yong Jae Lee. Finegan: Unsupervised hierarchical disentanglement for fine-grained object generation and discovery. In *CVPR*, 2019. 2

[32] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A Dataset of 101 Human Action Classes From Videos in The Wild. Technical report, CRCV-TR-12-01, November 2012. 5

[33] Sergey Tulyakov, Ming-Yu Liu, Xiaodong Yang, and Jan Kautz. MoCoGAN: Decomposing motion and content for video generation. In *CVPR*, 2018. 2, 4

[34] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Generating videos with scene dynamics. In *NIPS*, 2016. 2

[35] Jacob Walker, Carl Doersch, Abhinav Gupta, and Martial Hebert. An uncertain future: Forecasting from static images using variational autoencoders. In *ECCV*, 2016. 2

[36] Jacob Walker, Kenneth Marino, Abhinav Gupta, and Martial Hebert. The pose knows: Video forecasting by generating pose futures. In *ICCV*, 2017. 2

[37] 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, 5

[38] Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In *CVPR*, 2018. 1

[39] Ceyuan Yang, Zhe Wang, Xinge Zhu, Chen Huang, Jianping Shi, and Dahua Lin. Pose guided human video generation. In *ECCV*, 2018. 1, 2

[40] Bo Zhao, Lili Meng, Weidong Yin, and Leonid Sigal. Image generation from layout. In *CVPR*, 2019. 1

[41] Long Zhao, Xi Peng, Yu Tian, Mubbasir Kapadia, and Dimitris Metaxas. Learning to forecast and refine residual motion for image-to-video generation. In *ECCV*, 2018. 2