On Evaluating Video-based Generative Adversarial Networks (GANs)

Nancy Ronquillo
University of California, San Diego
nronquil@eng.ucsd.edu

Josh Harguess
SSC Pacific
joshua.harguess@navy.mil

***Abstract*—We study the problem of evaluating video-based Generative Adversarial Networks (GANs) by applying existing image quality assessment methods to the explicit evaluation of videos generated by state-of-the-art frameworks [1]–[3]. Specifically, we provide results and discussion on using quantitative methods such as the Fréchet Inception Distance [4], the Multi-scale Structural Similarity Measure (MS-SSIM) [5], as well as the Birthday Paradox inspired test [6] and compare these to the prevalent performance evaluation methods in the literature. We summarize that current testing methodologies are not sufficient for quality assurance in video-based GAN frameworks, and that methods based on the image-based GAN literature can be useful to consider. The results of our experiments and a discussion on evaluating video-based GANs provide key insight that may be useful in generating new measures of quality assurance in future work.**

***Index Terms*—Generative Adversarial Networks, Video, Evaluation, Metrics**

I. Introduction

Recent literature has closely followed the growing interest in generative adversarial networks (GANs) [7] and their application to a variety of problems, now including video generation [1]–[3]. In the development of these robust GAN frameworks, special care is required in order to improve fidelity and diversity for mitigating the problem of mode collapse, which is a severe form of non-convergence to the full intended distribution [8]. However, the problem of developing a general evaluation method that assesses fidelity and diversity in GANs, even for image-based GANs, is still an open research problem and no one solution, or metric, has been able to capture all elements that can be considered in evaluating GAN performance. This has motivated a lot of work on the evaluation of image-based frameworks ([9] provides a good summary). The difficulty lies in determining what aspects of a GAN are important to evaluate, and finding a method that fits all of these into a single metric or test. On one hand, for generated GAN data it is important to provide *qualitative* assessments of the generated samples, which includes checking for naturalness and realism in colors, shapes, perspective, and structure. For example, in evaluating realism a good method would condemn samples with people or objects that are physically improbable. This has most successfully been accomplished by the use of human mass surveys assessing preference. It is also important to provide *quantitative* evaluation of generated GAN data, which includes detecting non-convergences (such as mode collapse and providing measures of diversity), evaluating fidelity of the generated samples, and measuring the distance of a generated distribution from the intended training distribution using statistical estimates.

Compared to image-based GAN frameworks, video-based GAN frameworks face the additional challenge of incorporating spatio-temporal dynamics which are inherent in video data. Recent solutions, such as those from [1]–[3] have proposed the innovative use of deeper, more complex GANs for capturing aspects of appearance and motion. Among the three video-based GAN frameworks mentioned above the prevalent methods of qualitative and quantitative evaluation are the use of a human preference mass surveys and the inception score [10] for the UCF 101 dataset [11]. While human preference mass surveys provide an excellent way of evaluating generated video quality (naturalness and realism in colors, shapes, movements and appearance) due to the highly trained human eye, they can be biased toward evaluating visual quality, require large amounts of resources, and be unable to detect nuanced instances of non-convergence such as mode-collapse. That is, while key details in appearance may depict non-convergences, such as a lack of diversity implying mode collapse, a human oracle assessing a video sample may not be attuned to look for these subtleties and ignore them all together in their evaluation in mass surveys. The inception score aims to capture the essence of both fidelity and diversity for a batch of generated videos (see section III-B1 for more details), however it depends on a large number of samples and a third party feature extractor used only with the UCF 101 dataset as reported in [1]–[3]. It has recently also become subject of large scrutiny for its many limitations; the authors of [12] summarize this work well. This work highlights a gap in the literature addressing proper and reasonable evaluation methods for video-based GANs.

In this work we apply existing image quality assessment strategies to explicit evaluation of videos generated by the prevalent frameworks in the literature: Video GAN (VGAN) [1], Temporal GAN (TGAN) [2], and Motion and Content GAN (MoCoGAN) [3], to quantitatively evaluate video-based GANs. We apply the image assessment strategies of the Fréchet Inception Distance [4], the Multi-scale Structural Similarity Measure (MS-SSIM) [5], as well as the Birthday Paradox inspired test [6] and compare these to the reported

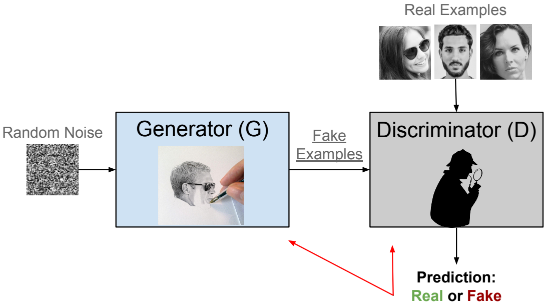

Fig. 1. Diagram of a GAN framework [7]. Using a seed of real data for training, in this adversarial training procedure the generator (G) would like to fool the discriminator (D) by making realistic fakes, and D would like to recognize all fakes.

inception scores of each framework. We show that the testing methodologies meant for quality assurance in image-based GAN literature are not sufficient for the same assurance in video-based GAN frameworks. New methodologies are required in order to capture the spatio-temporal dynamics inherent in video generated data. Nevertheless, in this work, we provide a discussion on evaluating video-based GANs and provide key insights that may be useful in generating new measures of quality assurance.

NOTATIONS: We use boldface letters to represent vectors. The conditional distribution between random variable X and Y is denoted as $p(y|x)$, and the marginal distribution of y over a set of $x \in G(z)$ is $p(y) = \int p(y|x = G(z))dz$. The Gaussian distribution with mean μ, and variance Σ is denoted as $p(x) \sim \mathcal{N}(\mu, \Sigma)$. $\mathbb{E}_x$ denotes the expectation with respect to x. We denote the Kullback-Leibler (KL) divergence between distributions p and q by $\mathrm{KL}(p||q)$. We also use, $Tr(A)$ and $||A||_2$ to denote the trace, and L2-norm of A respectively. Lastly, $|\mathcal{B}|$ is the support size of set $\mathcal{B}$.

II. Preliminaries

A. Brief Review of GANs

Generative Adversarial Networks (GANs) [7] are a class of unsupervised machine learning algorithms that are characteristically comprised of two competing networks playing a zero-sum game. That is, a GAN simultaneously trains a generator and a discriminator which are natural adversaries. An unlabeled dataset, which is desired to be augmented, is used as input to the discriminator along with fake output samples from the generator. The discriminator predicts the authenticity of its inputs and attempts to maximize the probability of correctly distinguishing real examples from the fake examples. Simultaneously, the generator aims to decrease the probability of the discriminators' successes by learning to generate increasingly realistic fake examples. Simplistically, GANs have the objective to learn a generative function that uses a random input to create realistic data that is modeled after the training dataset. This learning framework is illustrated in Fig. 3.

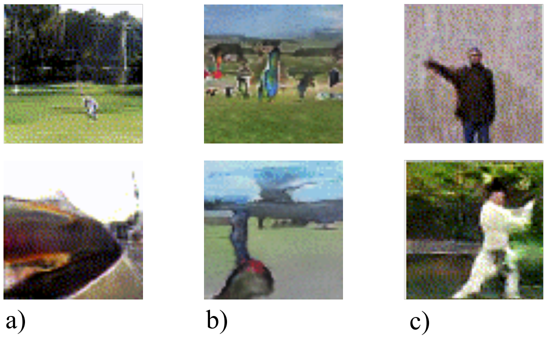

Fig. 2. Examples of videos generated by GAN frameworks: a) VGAN using the Golf and Train Station datasets [1], b) TGAN using the Golf and UCF 101 datasets [2], and c) MoCoGAN using the Actions and Tai-Chi datasets [3].

B. Video GANs

While the applications of a GAN framework are far and wide, GANs that generate high resolution realistic images have harnessed the interest of much of the computer vision community and as a result have proved to be the most prominent application for GAN, so much so that state-of-the-art image-based GANs achieve incredible results of producing extremely realistic, very high resolution images (1024×1024 pixels). Following historical trends for computer vision, where video advances follow successful image advances, the success in video generation with GANs follows and leverages advances in image-based GANs, currently with results that do not yet mimic the incredulity of image-based GANs. In video generation, state-of-the-art GAN frameworks attempt to encode the deeply involved spatio-temporal dynamics inherent to videos [1]–[3]. The ability for videos to capture motion across time adds complexity to any processing of video data, and certainly to these video-based GAN frameworks.

We consider a video GAN to be a framework that generates a video (or sequence of frames) from a single random input drawn from the latent space. In this work, we compare three video GAN frameworks that are prominent baselines in the literature and who uniquely approach the spatio-temporal dynamics of video. Next, we briefly describe each.

1) VGAN [1]: In [1] a video generation framework (VGAN) highlights the scene dynamics present in a video and hinges on modeling these by decomposing a video into a moving foreground (3D-convolutions) and a static background (2D-convolutions). This is implemented using a two-stream model that independently treats the foreground and the background to encourage the network to focus on the motion of the objects in the foreground. With VGAN, given a low dimensional random vector from latent space a high dimensional video is produced.

As the first large scale video generation framework, [1] sets the stage for evaluating video GANs following some of the methods of evaluating image-based GANs. [1] compares VGAN video results to those crafted by an auto-encoder network with similar parameter size, as well as those in a set of real videos. The comparison methods include a qualitative human preference mass survey and quantitative evaluation by reporting classification accuracy achieved with the generated videos. Fig. 2 a) shows stills of videos generated by VGAN. The separation of foreground and background is visually most successful in examples where the content of the video indeed has a natural static background and moving foreground. For example like in the Golf dataset where a sample may contain a golfer swinging in center frame with a green field in the background.

2) TGAN [2]: The temporal GAN (TGAN) framework of [2] is characterized by coupling two sub-networks, a temporal generator and an image generator, that sequentially generate frames for a video and together make up the generator portion of the GAN. This generator exploits a single latent input by first passing it through a temporal generator that outputs multiple latent variables, each representing an image corresponding to a single frame in the video. Next, each latent variable serves as input to the image generator in order to generate a sequence of frames that make up the final generated video. The result is a model that aims to capture the latent space representation of the time dimension in videos by placing emphasis on specifically learning the time-dynamics with its temporal generator.

[2] claims improved qualitative and quantitative results in terms of a human preference mass survey and the inception score for the UCF 101 [11] dataset (discussed in section III-B1) over VGAN. This is the first use of the inception score for video-based GAN evaluation. Fig. 2 b) shows stills of videos generated by TGAN.

3) MoCoGAN [3]: Most recently, a framework aimed at decomposing motion and content in videos (MoCoGAN) [3] has achieved impressive results in generating realistic videos by using four sub-networks. In MoCoGAN, while the content specifies the “who” (for example a particular person, animal, or thing), the motion specifies the “action” (for example a facial expression, a pose, or a movement). The decomposition starts with a latent space input that consists of a motion portion and a content portion. To generate a video with either a fixed content and varying motion, or a fixed motion with varying content, MoCoGAN first fixes the motion or the content portion of the latent variable and realizes the rest as a random latent variable via a recurrent neural network. The full latent space input is therefore a sequence of vectors (corresponding to a sequence of frames) that each has a motion and content portion (one of which is fixed). Each vector (consisting of motion and content) serves as input to the image generator, which produces a corresponding frame. All frames are evaluated individually by and an image discriminator and together as a single video by a video discriminator.

Fig 2 c) shows stills of videos generated by MoCoGAN. At first glance, it is apparent that the generated videos have distinguishable subjects (mostly different humans) conducting mostly distinguishable actions. [3] reports qualitative and quantitative improvements (in terms of a human preference mass survey and inception score for the UCF 101 [11] dataset) for MoCoGAN over TGAN and VGAN. However, results from different datasets are not compared, and the dependency on fixing either motion or content are not explored.

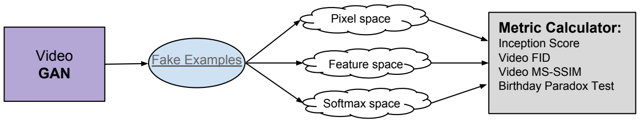

Fig. 3. Experimental procedure. In this work, we used videos generated by GAN frameworks (fake examples), and study these in pixel, feature, and softmax spaces. We compare results from the applications of MS-SSIM and Birthday Paradox test to video-based GAN evaluation.

III. Experimental Setup

In this work, we apply metrics from image-based GAN literature for evaluating videos generated by GANs. First, we obtain videos generated by VGAN, TGAN, and MoCoGAN using the datasets described below. For the purpose of this study, we focus on explicit evaluation methods that utilize a batch of generated videos and process these in either pixel space, feature space, or using the output of the softmax layer. We aim to study the viability of these metrics for video-based GAN evaluation and address their strengths and weakness for this application. The experiment tasks are illustrated in Fig. 3.

In this section, we begin by describing in detail the datasets considered. We provide a brief summary of the currently used inception score and discuss its limitations. Then we summarize three methods used for image-based GAN evaluation that we apply for video-based GAN evaluation: the FID, the MS-SSIM, and the Birthday Paradox based test. We discuss each method, and propose adaptations for these to be implemented for studying GAN generated videos. We discuss the implementation of FID with C3D and provide results for using this to evaluate the TGAN framework with the UCF 101 dataset. We discuss why the FID with C3D method is not reasonable for evaluating the VGAN or MoCoGAN frameworks and maintain further study of these with FID implementation beyond the scope of this paper. In the results Sect. IV, we exhaustively compare our implementation of the MS-SSIM and Birthday Pardox test methodologies and discuss the results we obtain from applying these for evaluating vide-based GANs.

A. Datasets

1) UCF 101 [11]: The UCF 101 dataset consists of 13,320 videos corresponding to 101 action classes [11]. Each video contains a single label belong to one of the action classes. The actions can be sports, small body movements, or group activities that are characterized by any or all of the following: human and object interaction, body-motion only, human-human interaction. UCF 101 has large diversity in terms of actions but also in terms of large variations in camera motion, object appearance and pose, object scale, viewpoint, cluttered background, illumination conditions, etc. Thus, UCF 101, is challenging yet very complete in terms of studying video generation. In experimental results we pay special attention to the performance of a framework trained on UCF 101 compared to more simple datasets.

2) Golf [13]: The Golf dataset [1], [13], is a compilation of 20,268 videos depicting golf scenes which generally include a golf course background and a golfer in the foreground. While every video in the dataset is unique, the videos in the Golf dataset appear very similar to each other, most appearing to be largely green with small moving dark shapes. In the implementation of VGAN, these video are stabilized and scaled to 64×64 pixels. In evaluating diversity for generated videos, we pay special attention to the comparison of using the golf dataset to a more varied dataset, like UCF 101 to study the role played by the dataset in diversity evaluation.

3) Moving MNIST [14]: The Moving MNIST dataset consists of 10,000 short videos depicting two moving digits across a black screen [14]. The motion of each digit is defined by its random initial position, direction, and velocity which are chosen form a Gaussian distribution. The moving MNIST dataset has the simplest content of all datasets used because of the black and white color range, and simple two character content, however, because of its randomly selected motion parameters, the support size of the potential unique videos is still infinite.

4) Tai Chi [3]: The Tai Chi dataset consists of 4,500 videos of various people performing slow body movements of Tai Chi [3]. The subjects of the video are similar in appearance, mostly wearing either white or orange shirt and trousers, in general with a background of greenery. Each video is centered and scaled to 64×64 pixels.

5) Actions [15]: The Human Actions dataset consists of 81 low resolution videos, scaled to 96×96 pixels, of various people performing one of 9 simple body movements such as running, walking, or jumping jacks [15]. Each action is performed by each of 9 total subjects of varying appearance and clothing. Camera motion appears to be negligible and the background of each video is consistently a brown or gray wall. This actions dataset has the smallest amount of samples among the datasets used, thus video generation results with this dataset will yield insight into the role of a training set size on the evaluation methods used and results.

B. Metrics

1) Inception Score [10]: For completeness, we briefly describe the inception score which is the prevalent method in the literature for evaluation of video GANs. The inception score was first proposed for evaluating image-based GANs in [10], and aims to capture both a measure of fidelity and diversity in a single score. It relies on a pre-trained network (the Inception Net [16] for images and the C3D network for videos [17]) which uses the generated GAN samples to obtain estimates of the conditional label (y) distribution of the samples (x), $p(y|x)$ (i.e. the output of the softmax layer), and the marginal label distribution $p(y)$ over all samples. For samples with high fidelity, which are highly discriminable, a conditional distribution $p(y|x)$ is expected to have a low entropy, that is has low ambiguity in assigning a label. On the other hand, the marginal label distribution for samples with high diversity, and therefore less likely to be characterized by mode-collapse, the marginal distribution $p(y)$ is expected to have a high entropy. Under these traits, the inception score captures the qualities of fidelity and diversity by using the expected KL divergence between $p(y|x)$ and $p(y)$, where a higher score implies lower entropy in $p(x|y)$ and higher entropy in $p(y)$. The inception score is calculated using the estimates of $p(y|x)$ and $p(y)$

$$IS(p(y|x), p(y)) = \exp(\mathbb{E}_x \mathrm{KL}(p(y|x)||p(y))). \qquad (1)$$

While the inception score is reported to be correlated with the properties of fidelity and diversity, and is shown to be correlated with results form human mass surveys, it has also been shown to be subject of many limitations. These include: its dependence on a third party architecture, its inconsistent and misleading results with certain distributions, its requirement of a large number of samples, and its inability to detect severe mode-collapse (see [12] for a good summary).

2) Fréchet Inception Distance [4]: The first evaluation method that we apply uses statistical estimates of the generated video distribution similarly to the inception score, it compares generated videos directly to the training dataset. The Fréchet Inception Distance (FID) also relies on a third party architecture (we suggest the C3D network for videos fine-tuned to the UCF 101 dataset [17]) to make statistical estimates about the generated distribution. FID aims to calculate the similarity between the distribution of the generated data and the distribution of the training data. First, a batch of generated data and the entire training dataset are each run through the C3D network to obtain a feature space representation (the output of the 5th pooling layer) of every sample. Secondly, the distribution of each of these data is approximated as a multidimensional Gaussian and characterized by their mean and variance: the training data distribution $p_w(x) \sim \mathcal{N}(\mu_w, \Sigma_w)$, and the generated data distribution $p(x) \sim \mathcal{N}(\mu, \Sigma)$. Lastly, the FID between the two estimated Gaussian distributions is given by:

$$FID(p(x), p_w(x)) = ||\mu - \mu_w||_2^2 + \mathrm{Tr}(\Sigma + \Sigma_w - 2\sqrt{\Sigma\Sigma_2}). \qquad (2)$$

The third party architecture chosen to provide feature space representations of each sample is important to consider for computing reasonable FID scores that take into consideration the methods for encoding spatio-temporal dynamics. For video GAN frameworks, such as VGAN, TGAN, and MoCoGAN, that use differing approaches to encode spatio-temporal dynamics the best feature extractor should naturally mimic the aims of each of the GAN frameworks, otherwise the resulting statistical estimates may be misleading. The C3D architecture uses 3D convolutional layers that aim to capture both appearance and motion. Visualizations of the network over time show that it focuses first on the appearance of frames and on the salient motion exclusively in layers thereafter [17]. The TGAN framework has a similar approach of decomposing the frame appearance and temporal aspects (motion) using

2D and 1D convolutions in image and temporal generators respectively to capture appearance and motion separately. We find this match of TGAN with C3D to be the most reasonable to study, since the other VGAN and MoCoGAN frameworks have distinctly different approaches to modeling motion. For that reason, these are not reasonable to compare using C3D.

3) MS-SSIM [5]: Next, we apply the multi-scale structural similarity measure (MS-SSIM) to evaluate diversity in GAN generated videos. The MS-SSIM was first proposed as a method for evaluating video compression algorithms, and was later adopted for evaluating image-based GANs by [5]. The MS-SSIM aims to assess perceived similarities between two samples and yields a score between 0 (low similarity) and 1 (high similarity). The average MS-SSIM score for a set of generated data, when all samples are compared to each other, is used to capture an essence of diversity for a GAN. The MS-SSIM down-samples inputs at different scales and computes functions of contrast $C(x, y)$ and structure $S(x, y)$ at each scale for pixel neighborhoods in two samples denoted by (x, y), in addition to a function of luminance $L(x, y)$ at the broadest scale [5]. The MS-SSIM score is calculated as a weighted average of these functions at all scales in Eq. 3 below.

$$\text{MS-SSIM}(x, y) = L_M(x, y)^{\alpha_M} \prod_{l=1}^{M} C_l(x, y)^{\beta_l} S_l(x, y)^{\gamma_l}. \quad (3)$$

We use a simple adaptation of the MS-SSIM to evaluate videos, that compares one frame from one sample to all frames from a second sample to capture similarities across time. The average of the MS-SSIM score over all frames is the score for each video, and the average over all videos is reported as the video MS-SSIM in this paper. That is, for any two videos i, j from the generated video set $\mathcal{G}$ the video MS-SSIM (VMS-SSIM) is computed as

$$\text{VMS-SSIM}(i, j) = \frac{1}{F_{i,j}} \sum_{n \in \mathcal{F}^{(i)}} \sum_{m \in \mathcal{F}^{(j)}} \text{MS-SSIM}(x_n, y_m), \quad (4)$$

where $\mathcal{F}^{(i)}$ is the set frames in video i, $\mathcal{F}^{(j)}$ is the set frames in video j, $F_{i,j} = |\mathcal{F}^{(i)}| + |\mathcal{F}^{(j)}|$, M is number of scales used, and α, β, and γ are the weighting parameters.

4) Birthday Paradox [6]: The work in [6] raises questions about the true performance of GANs and their ability to truly learn the training data distribution. The authors stuidy how closely a learned distribution is able to mimic the true training data distribution by comparing the support size of each. The authors propose a simple test, based on properties of discrete probability, in order to calculate an upper bound on the support size of the generated data set. [6] characterizes an upper bound on the support size (SP) of a generated set given that a *near duplicate* is probable in a sample set. First, through an automated procedure, a batch of candidate duplicate pairs is found for a set of s generated samples. A human oracle then picks out true near duplicates and records these for the test. When a near duplicate is found among a set of s samples with probability greater than 50%, the support size is upper bounded as follows:

$$SP(\mathcal{G}) \leq s^2, \quad (5)$$

where $\mathcal{G}$ is the set of generated samples.

We propose implementing the birthday paradox test for videos just as detailed above, with the following procedure for detecting near duplicates. For training sets considered to be simple or with low variety (like the moving MNIST, Golf, Tai Chi, and Actions datasets) we use the euclidean distance between all frames as a measure for finding candidate duplicates automatically. In particular, for a batch of size s, the top 20 videos with the smallest euclidean distance among frames are selected as candidates. For training sets considered to be highly varied (like the UCF 101 dataset) we use the point distance (with points obtained by the feature space representation, e.g. using C3D) between two samples instead. Next, we describe the crucial part in evaluating near duplicates of motion using a human oracle. When using the minimum distances in pixel space or in feature space, the automated selection of near duplicates will have some bias toward selecting samples that are similar in appearance (e.g. similar colors or content). While analyzing diversity in appearance is important, the complexity lies in analyzing diversity of temporal content, or motion, when it comes to videos. Here, the power belongs to the human oracle who may focus on either, or both appearance and motion. In the following experiments, the human oracle (researcher conducting experiments) selects near duplicates when similarities in both appearance and motion are observed. The resulting appearance and motion near duplicates are useful to study in particular for video frameworks which make a priority of capturing these spatio-temporal dynamics. The final upper bound on the support size is found as above. The support size of the generated distribution is compared to the support size of the training dataset to get a sense of how well the GAN has learned this distribution.

IV. Results

Next, we discuss our implementation of each of the proposed evaluation metrics and provide a discussion about each. We use the inception scores (which correlated with results of a human preference survey) reported by [3] as a reference for accepted performance of the frameworks considered. Specifically, [3] reports the highest score of $(12.42 \pm .03)$ for MoCoGAN, followed by a score of $(11.85 \pm .07)$ for TGAN, and a score of $(8.18 \pm .05)$ for VGAN.

A. FID

The implementation of FID for video with C3D requires processing all videos in the training dataset in addition to all the videos in the generated set. This incurs larger computational cost than the inception score, video MS-SSIM, and Birthday Paradox for motion where only the videos in the generated dataset are processed. However, the main limitation of FID with C3D lies in its large dependence on the C3D architecture for providing reasonable statistical estimates.

As a proof of concept, we compute the FID score with C3D method to videos generated by the TGAN framework using the UCF 101 training dataset. Using 100 videos generated by TGAN using the UCF 101 training dataset, a score of ($45 \pm 1 \times 10^3$) is obtained. Computing only a single score for TGAN highlights another limitation of the FID score, which is that the FID score is difficult to decipher unless being directly compared to other frameworks. We note that a variety of spatio-temporal feature extractors may be applied with FID for evaluating specific video GANs separately. However, a single feature extractor is unlikely to capture and assess the spatio-temporal dynamics efficiently in a general manner since these are uniquely encoded by the video GAN frameworks to consider appearance and motion in distinctly different manners. Thus, we maintain exploration of other possible third party feature extractors beyond the scope of this paper, here only discussing the role this selection may play in affecting the FID score for video GANs.

B. Video MS-SSIM

TABLE I
MS-SSIM

Model	Dataset	Video MS-SSIM
VGAN	GOLF	0.09
TGAN	GOLF	0.11
	M-MNIST	0.09
	UCF 101	0.07
MoCo-GAN	Tai Chi	0.19
	Actions	0.40

In Table I we report the video MS-SSIM scores obtained using various datasets with 100 samples of each and using the frameworks of VGAN, TGAN, and MoCoGAN. The first thing that we note, is a large variation on the scores between datasets and between frameworks. This sheds light on the need to carefully consider evaluation over a variety of datasets, since these can vary widely even when using the same framework. Secondly, we note that the accepted conclusions reached by the inception score cannot be corroborated with these video MS-SSIM scores, that is when evaluating diversity alone, it is not clear that one architecture is clearly better than another, and in fact doing so may be misleading. For example, consider the results obtained by VGAN and TGAN for the Golf dataset, where the results suggest that VGAN is better able to produce diverse results, but lower scores are achieved using TGAN and datasets such as UCF 101 or moving MNIST.

More notably, consider the results obtained by MoCoGAN which are the larges and therefore suggest much smaller diversity than for any dataset in VGAN or TGAN. The videos generated by MoCoGAN are a obtained by using a combination of fixing the content and motion portions separately. For any dataset, when using MoCoGAN we expected that when content portion is fixed the appearance should be similar among generated video samples, but the motion should be varied, and vice-versa when instead the motion portion is fixed. Since MoCoGAN achieves very high video MS-SSIM scores, this suggests that as-is video MS-SSIM is not be able to perceive small variations in appearance or motion and instead discounts these as being highly similar. While these limitations disqualify video MS-SSIM from being adopted as general determinant of diversity, this quick and low computational complexity test is very useful at least for assessing diversity of datasets using a single framework.

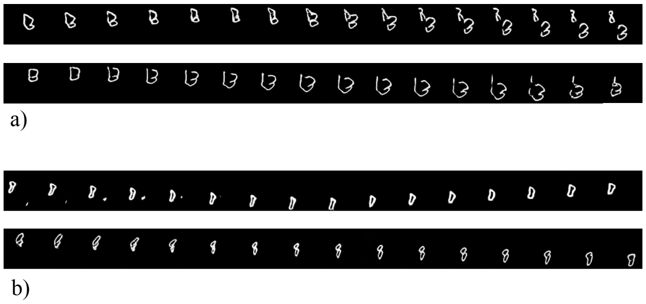

Fig. 4. Examples of automatically selected candidate near duplicates using the Birthday Paradox inspired test on generated videos from the TGAN framework using the moving MNIST dataset. a) Pair selected as a near duplicate due to similar motion. b) Pair not selected as a near duplicate.

C. Birthday Paradox with motion

Lastly, we implement the birthday paradox test with motion for a variety of datasets with the three frameworks: VGAN, TGAN, and MoCoGAN. As described in sect III-B4 we pay special attention to evaluating both appearance in motion when determining the near duplicated for this test. Fig. 4 shows examples of candidate near duplicates selected using the automatic method described in Sect. III-B4. The final selection of near duplicates is focused on both similar appearance and motion. For example, the pair of videos in Fig. 4 a) is selected since the samples have similar appearance and motion. On the other hand, although the main digits floating in the pair in Fig. 4 b) are nearly identical, their motion across the time appears to be quite different, thus this pair is not selected as a near duplicate. The results for all tests are presented in Table II where we compared the upper bound of the support size of the generated video sets to the size of the corresponding training datasets used.

First, we notice that for training datasets that have low variations in appearance, like the golf dataset the generated sets of videos have a significantly smaller support size. For example, using the Golf dataset with VGAN results in a support size around 100 times smaller than the size of the training dataset. A large improvement in diversity is obtained by using the golf dataset with TGAN, where the support size is only around 10 times smaller than the size of the training dataset. For very small datasets, like the Actions dataset with MoCoGAN the resulting support size is very small (only 36), but compared to the size of the training data is half as large, showing that actually, this achieves quite good diversity. We highlight that using the birthday paradox test with motion for evaluating a video GAN provides a quantitative measure of diversity that correlates well with the inspection scores reported in [3] (and therefore also human preference surveys), but in addition also helps to provide intuition and information that is useful for evaluating GAN performance between different datasets and frameworks. For example, using the Birthday Paradox test with motion, can enable an evaluator to form guarantees about a particular dataset being used, or provide useful measures for comparing between frameworks.

TABLE II
BIRTHDAY PARADOX

Model	Dataset	Training Set size	Generated Set support size
VGAN	GOLF	20,268	225
TGAN	GOLF	20,268	2,500
	M-MNIST	10,000	2,500
	UCF 101	13,319	10,000
MoCo-GAN	Tai Chi	4,500	256
	Actions	81	36

V. CONCLUSION

In this work we provide a study on methods for evaluating video-based GAN frameworks. First, we summarize the limitations of the prevalent evaluation methods which are human preference surveys and the inception score, and highlight a gap in the literature for assessing fidelity and diversity in videos generated by GANs. We apply FID with C3D to the TGAN framework with UCF 101 dataset and discuss the viability of the metric and its dependence on a third party architecture which makes it difficult to be generalizable. We also propose methods to implement the MS-SSIM and Birthday Paradox tests for video GANs, and study the performance. We show that these methods are useful in providing more information to the evaluator, enabling intuition and some guarantees about the framework/datasets combinations used. While, none of these methods are shown to be sufficient stand-alone evaluation metrics, they are certainly informative and should be implemented when able for a well rounded evaluation of a framework or dataset. We hope that the results of our experiments and a large discussion on evaluating video-based GANs provide key insight that may be useful in motivating and generating new measures of quality assurance in future work.

VI. ACKNOWLEDGEMENTS

The authors would like to thank Marissa Dotter and Cameron Hilton for their continued support and guidance on generating data, and Benjamin Cichy for help in dataset selection.

REFERENCES

[1] C. Vondrick, H. Pirsiavash, and A. Torralba, "Generating videos with scene dynamics," in *Proceedings of the 30th International Conference on Neural Information Processing Systems*, ser. NIPS'16. USA: Curran Associates Inc., 2016, pp. 613–621. [Online]. Available: http://dl.acm.org/citation.cfm?id=3157096.3157165

[2] M. Saito, E. Matsumoto, and S. Saito, "Temporal generative adversarial nets with singular value clipping," in *ICCV*, 2017.

[3] S. Tulyakov, M.-Y. Liu, X. Yang, and J. Kautz, "Mocogan: Decomposing motion and content for video generation," *CVPR*, 2018.

[4] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update tule converge to a local nash equilibrium," in *Conference on Neural Information Processing Systems 27*. Curran Associates, Inc., 2017.

[5] A. Odena, C. Olah, and J. Shlens, "Conditional image synthesis with auxiliary classifier GANs," in *Proceedings of the 34th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, D. Precup and Y. W. Teh, Eds., vol. 70. International Convention Centre, Sydney, Australia: PMLR, 06–11 Aug 2017, pp. 2642–2651. [Online]. Available: http://proceedings.mlr.press/v70/odena17a.html

[6] S. Arora, A. Risteski, and Y. Zhang, "Do GANs learn the distribution? some theory and empirics," in *International Conference on Learning Representations*, 2018. [Online]. Available: https://openreview.net/forum?id=BJehNfW0-

[7] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2014, pp. 2672–2680. [Online]. Available: http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf

[8] L. Metz, B. Poole, D. Pfau, and J. Sohl-Dickstein, "Unrolled generative adversarial networks," *CoRR*, vol. abs/1611.02163, 2016.

[9] A. Borji, "Pros and cons of gan evaluation measures," *CoRR*, vol. abs/1802.03446, 2018.

[10] T. Salimans, I. J. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," in *NIPS*, 2016.

[11] K. Soomro, A. R. Zamir, and M. Shah, "Ucf101: A dataset of 101 human actions classes from videos in the wild," *CoRR*, p. 2012.

[12] S. Barratt and R. K. Sharma, "A note on the inception score," *CoRR*, vol. abs/1801.01973, 2018.

[13] B. Thomee, D. A. Shamma, G. Friedland, B. Elizalde, K. Ni, D. Poland, D. Borth, and L.-J. Li, "Yfcc100m: The new data in multimedia research," *Commun. ACM*, vol. 59, no. 2, pp. 64–73, Jan. 2016. [Online]. Available: http://doi.acm.org/10.1145/2812802

[14] N. Srivastava, E. Mansimov, and R. Salakhutdinov, "Unsupervised learning of video representations using lstms," in *Proceedings of the 32Nd International Conference on International Conference on Machine Learning - Volume 37*, ser. ICML'15. JMLR.org, 2015, pp. 843–852. [Online]. Available: http://dl.acm.org/citation.cfm?id=3045118.3045209

[15] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri, "Actions as space-time shapes," in *The Tenth IEEE International Conference on Computer Vision (ICCV'05)*, 2005, pp. 1395–1402.

[16] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2818–2826, 2016.

[17] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, ser. ICCV '15. Washington, DC, USA: IEEE Computer Society, 2015, pp. 4489–4497. [Online]. Available: http://dx.doi.org/10.1109/ICCV.2015.510