Towards Bridging the Semantic Spaces of the One-to-Many Mapping in Cross-Modality Text-to-Video Generation

Anonymous authors
Paper under double-blind review

Abstract

Despite recent advances in text-to-video generation, the role of text and video latent spaces in learning a semantically shared representation remains underexplored. In this cross-modality generation task, most methods rely on conditioning the video generation process by injecting the text representation into it, not exploring the implicit shared knowledge between the modalities. Nonetheless, the feature-based alignment of both modalities is not straightforward, especially for the *one-to-many* mapping scenario, in which one text can be mapped to several valid semantically aligned videos, which generally produces a representation collapse in the alignment phase. In this work, we investigate and give insights on how both modalities cope in a shared semantic space where each modality representation is previously learned in an unsupervised way. We explore a perspective from the latent space learning view and analyze a framework proposed in this work with a plug-and-play nature by adopting autoencoder-based models that could be used with other representations. We show that the one-to-many case requires different alignment strategies than the common ones used in the literature, which suffer in aligning both modalities on a semantically shared space.

1 Introduction

Cross-modality video generation has recently received a lot of attention due to the impressive performance of recent video generators, making it more difficult to identify synthetic from real samples. However, in the representation learning aspect of this task, more specifically, in the coupling with joint embedding learning, we have few answers as to how both modalities cope in latent space and how the feature alignment occurs for different approaches. Recent works (Girdhar et al., 2023; Maiorca et al., 2023; Theodoridis et al., 2020) focus on alignment directions in latent space but with a general approach that does not deal explicitly with the *one-to-many* mapping scenario where we have one input from an origin modality that can be mapped to n different and valid outputs of a target modality.

In text-to-video generation, the nature of language enables different forms to describe a singular scene enclosed in a video format, while at the same time it enables different visual interpretations of the enclosed scene in a unique text description. In this context, cross-modality alignment is hindered by a one-to-many case, as a collapse process is unintentionally encouraged in training, where one input is associated to several cross-modality outputs, being prone to collapse to the most frequent association, a mean representation of it, or even a random and closer output in latent space, for example.

Although a challenging task, the analysis of the learned joint latent space under the generative context is underexplored. In representation learning, most methods rely on classification and retrieval tasks when dealing with a joint embedding approach (Fang et al., 2022; Girdhar et al., 2023; Xue et al., 2023) to validate the learned representation. Regarding text-to-video generators, most methods focus on a solution in which the text representation is passed into a fusion process in the video generator model (Ge et al., 2022; He et al., 2022; Ho et al., 2022; Wang et al., 2024). In these approaches, the latent representation is held in the background, as this alignment is learned implicitly in the process.

In this context, some works have been proposed to align and generate data from multiple modalities (Tang et al., 2023) in which each modality model is trained from scratch to regularize and generate the semantically shared representation space. Nevertheless, currently there are available in the computer vision community several pre-trained models for both text and video, and, to the best of our knowledge, few works try to benefit from this pre-learning phase in order to map the modalities in a feature-based alignment approach. Moreover, we also have little information on this alignment in the latent space perspective, which could bring further insight into how the modalities cope in a semantic shared space.

In particular, when dealing with autoencoding approaches that regularize the target-modality latent space, the analysis of this implicit representation could aid in the understanding of the alignment process. For the image modality, some works have explored this relation, from concerns about bias (Gat et al., 2022) when analyzing the latent space, ideal latent distributions for generative models (Hu et al., 2023), better understanding of simpler alignments between the modalities for classification tasks (Maiorca et al., 2023), to approaches to build such joint distribution from autoencoders models (Piening & Chung, 2024; Xu et al., 2019) that enable the generation process.

In this work, we aim for a better understanding and reasoning of the *one-to-many* case in text-to-video generation. We consider a pipeline that benefits from models trained in an unsupervised way with their corresponding modality, such as text and video encoders, and align them in the representational space. We show that approaches that try to align those representations directly (Girdhar et al., 2023) suffer in the one-to-many case. Furthermore, we investigate the impact of self-supervised representation learning alignments originally proposed for the same modality, such as BYOL (Grill et al., 2020), SimSiam (Chen & He, 2021), and VicReg (Bardes et al., 2022), and show their limitations when applied to this case. The main contributions of this work[1] include the following:

- We identify the *one-to-many* mapping scenario as a key challenging case in cross-modality text-to-video generation and demonstrate its impact in feature alignment approaches. We also take the perspective of latent space analysis and explore the relationship between text and video distributions.
- We propose a unidirectional progressive model for reasoning of the mapping between text and video, where text is mapped first to a semantic shared space before being mapped to the target distribution.
- We investigate different mapping functions between the data modalities and show their impact in the semantic shared space and how the individual modality representations can affect the alignment.

2 Related Works

Text-to-video generation can be divided into approaches that inject the text as conditioning information in the video generation process and approaches that try to learn a generation pipeline by aligning the latent representation of both modalities.

2.1 Fusion-based text-to-video generation

In multimodal machine learning, Liang et al. (2024) categorizes fusion into two types: *fusion with abstract modalities* and *with raw modalities*. In text-to-video generation, most methods adopt the former, which considers encoders to represent each modality before applying a fusion method with the two streams of data. The latter considers a fusion process from early representation learning stages, such as the raw modalities themselves, and is less explored in this work context.

Text conditioning in fusion-based techniques can rely on a simple injection of text in the process, such as a concatenation of text and video embeddings (Ge et al., 2022; Wang et al., 2024), to complex fusion methods based on attention modules or Multilayer Perceptron (MLP) layers to process and generate a text-video fused embedding (He et al., 2022; Ho et al., 2022). Both text-only and text-video fused embeddings can be used at different stages or layers of the video generation process. This type of injection attempts to ensure that the conditioning information is maintained in the generation and aligned with the desired video semantics. Ge et al. (2022) prepend the text embedding to the video tokens in their transformer-based video generator. Ho et al. (2022) applied MLP layers to the text embedding before adding it to each residual block of the

[1] The models, checkpoints and data sets generated in this work are publicly available at `http://to.be.shared`.

diffusion process. He et al. (2022) concatenates the conditioning information with the latent input with the option to apply or not cross-attention layers before adding it as input to a Latent Diffusion Model (LDM).

2.2 Cross-modality generation based on feature-alignment

Unlike approaches that inject the conditioning information into the generation process, multi-modal latent alignment is focused on creating a shared latent space between different modalities. The alignment can enable from a unidirectional or one-to-one (Theodoridis et al., 2020) to an any-to-any generation (Tang et al., 2023). In the former, from an input modality M_1 a sample of a second target modality M_2^t is generated but not in the other way around. In the latter, from one or multiple input modalities $M_1, \ldots, M_n$, one can generate one or multiple target modalities $M_1^t, \ldots, M_m^t$.

In the unidirectional context, Theodoridis et al. (2020) proposed a alignment of the latent spaces of two modalities using Variational Auto-Encoders (VAEs) in two separate phases. First, a VAE model for each modality is trained to learn its corresponding latent space. Then, in a second phase, an additional VAE is used to learn a mapping between the two modalities that represents a joint embedding space between them. The alignment is learned with the Fréchet distance (C. Dowson & V. Landau, 1982) between the distributions and validated on food image analysis and 3D hand pose estimation. Similarly, but in an any-to-any context, CoDi (Tang et al., 2023) was also proposed with a two-stage process. The first stage included learning the representation of each modality with a LDM. Then, the second stage included learning a shared latent space between the modalities, where one representation is projected onto another by also injecting the target modality in the representation learning, and an alignment is learned with a contrastive approach.

Although these approaches implicitly support text-to-video generation, they did not explore this scenario or the *one-to-many* case. Nevertheless, they require a joint training framework from scratch, leaving in the background representation-oriented models for text and video that could benefit the generation process as pre-trained models for representation.

Moreover, in video-language pre-training other methods were proposed for cross-modality representation in: video-text retrieval, video question and answering (Xue et al., 2022; 2023). Xue et al. (2022) proposed a method that considered high- and low-resolution frames of a video to be encoded separately and later combined in a fusion approach before feeding the cross-modal method. Beyond text and video, Girdhar et al. (2023) proposed a representation-based alignment focused on images as the main binding modality between other modalities, except video; and Maiorca et al. (2023) presented a similar approach to CoDi (Tang et al., 2023) but in the text and image domain where the decoders to the target modality are pre-trained in their modality of origin.

Regarding the alignment process, a contrastive approach is generally used, such as InfoNCE (van den Oord et al., 2018) adopted in CLIP (Radford et al., 2021). Other works further explore its alignment process (Li et al., 2022; Yeh et al., 2022). DeCLIP (Li et al., 2022) proposed to use a smaller set of 88M pairs with self-supervised (SS) learning on both modalities, a multi-view cross-modality loss extending the multi-crop transformation of Caron et al. (2020), and a nearest-neighbor approach. Yeh et al. (2022) propose the removal of the negative-positive-coupling effect in learning. Although these works propose different modality augmentations, they are not directly applicable to the *one-to-many* case, as changing the text could generate bigger direct mapping sets for the same input text and mix different semantics. Also, InfoNCE reinforces a direct match, i.e., one-to-one case, between the modalities, considering other direct pairs, such as those in the one-to-many case, as "incorrect pairings" by their batch formulation of positive and negative samples.

3 Unidirectional Progressive Learning for Semantic-Shared Latent Space Alignment

We propose a unidirectional approach of the cross-modality text-to-video generation for understanding and reasoning of the one-to-many case. First, we assume that video, v, and text, t, data are in an ideal joint space, $p^*(v, t, z)$, where a latent variable, z, holds the joint semantic meaning of them. Hence, our problem is to learn the marginalized distribution $p^*(z)$ given the observations that come from the other marginals that are available at training time. To address the modeling problem of the semantic space $p^*(z)$, we intend to learn the marginal conditional distributions for each modality. That is, we intend to learn $p(z \mid v) \equiv p^*(z \mid v)$

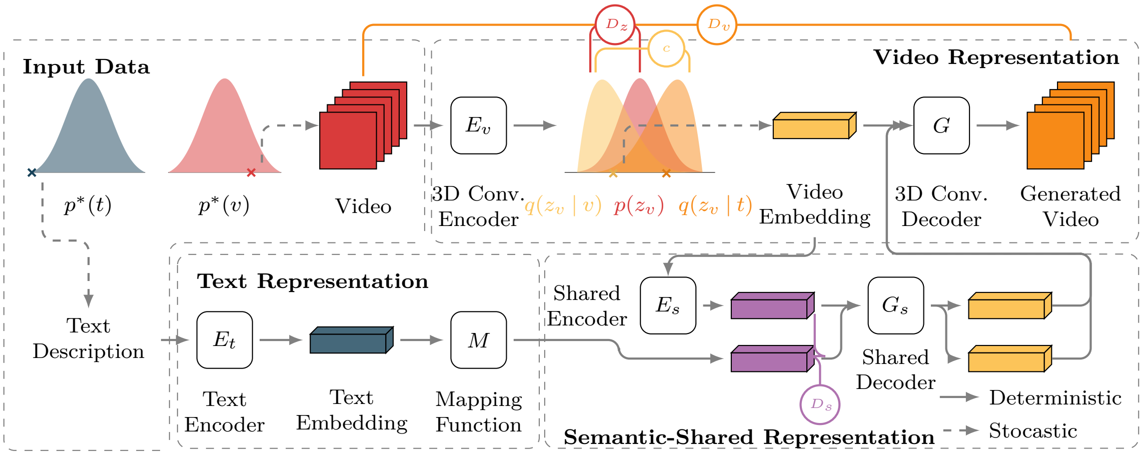


Figure 1: Pipeline to learn a joint semantic space $p(z)$ by bridging the gap between the conditional latent distributions of each data modality, $q(z \,|\, v)$ and $q(z \,|\, t)$, by minimizing the divergences between shared and target video representations, between the available pairs of text and video. We learn each of these posteriors individually as a variational family using the original data, $p^*(v)$ and $p^*(t)$, through a specific encoder, E_v and $M \circ E_t$, respectively. Given that our evaluation task is video generation from text, we evaluate the quality of the generated videos (by decoder G) through a discriminator D_v, and by inspecting the similarity on the latent spaces of the decoupling process through additional discriminators D_z and D_s.

and $p(z \mid t) \equiv p^*(z \mid t)$. However, learning these posteriors is intractable. Thus, we intend to approximate them with a variational family, $q(z \,|\, v)$ and $q(z \,|\, t)$, respectively, parameterized with neural networks. Finally, we need to make them similar so that the semantic information of both is equivalent.

To learn $q(z \,|\, v)$ and $q(z \,|\, t)$, we decouple each approximation into two phases. The first phase is the learning of each modality representation, such as $q(z_v \mid v)$ and $q(z_t \mid t)$, and, the second phase is the learning of the shared representation of $q(z \mid z_v)$ and $q(z \mid z_t)$, respectively. For $q(z_v \mid v)$, we propose a video extension of a Wassertein Autoencoder (WAE), which is trained in an unsupervised way and presented in Appendix A.2. Although a pre-trained model can be incorporated here as well, we select this approach to further evaluate different video architectures in the alignment process. For $q(z \mid t)$, we employ a pre-trained text encoder model E_t, for which we do not impose any restrictions. An overview of this pipeline is presented in Figure 1.

Bridging the Semantic Spaces by Progressive Decoupling. To learn the shared semantic space given text, $p(z \mid t)$, we approximate this posterior with a parameterized variational family, $q(z \,|\, t)$, for which we propose a two-part decoupling process represented by two models in hierarchical form. The first part encodes the string of words into an embedding z_t using a text encoder model E_t. Then, z_t is projected into a shared representation space with a mapping function $M \colon z_t \to t_s$. We map t_s to the video latent space $q(z \mid t)$, through a generator G_s, which is the decoder part of an autoencoder model from the video latent space $q(z_v \mid v)$ to a shared representation between text and video. In this scenario, the encoder E_s generates the shared representation v_s from the video code sampled from $q(z_v \mid v)$.

We found empirically that trying to approximate the target distribution in a one-step approach, i.e. without a hierarchical form, led to the collapse of $q(z \,|\, t)$ in the *one-to-many* case. The decoupling by a hierarchical latent space is similar to Xu et al.'s (2019), but we apply to a modality that is different from the origin modality. Moreover, instead of applying a regularization in the latent space in the second stage, our model considers regularization in both intermediary (shared) and target (video) latent spaces.

To link the information between the learned semantic spaces from text, $q(z|t)$, and video, $q(z|v)$, we consider a WAE-based approach similar to the video semantic space, where we define:

$$D_{\mathrm{W}}\left(p^*(z), p(z \mid z_t)\right) = \inf_{q(z \mid z_v), q(z \mid z_t) \in \mathcal{Q}'} \left\{ \mathop{\mathbb{E}}_{z_1 \sim p^*(z)} \; \mathop{\mathbb{E}}_{z_2 \sim q(z \mid z_t)} \left[c\left(z_1, z_2\right) \right] + \lambda_{z_s} \mathcal{D}_s(q(z), p(z)) \right\}, \tag{1}$$

such that $\mathcal{Q}'$ is a non-parametric set of deterministic encoders, $z_1 \sim p^*(z)$ is a latent code representing the 'real' distribution, $q(z \mid z_v)$, $z_2 \sim q(z \mid z_t)$ is a generated latent code (through M) that depends on text embedding $z_t \sim q(z_t \,|\, t)$, and $\lambda_{z_s} > 0$ is weight for the divergence measure $\mathcal{D}_s$ between $q(z) = \mathbb{E}_{z \sim p^*(z)}\left[q(z \mid z_t)\right]$ and $p(z)$ representing our shared semantic space. For this phase, we consider cost similarity $c(z_1, z_2)$ as:

$$\begin{aligned} c(z_1, z_2) = & \lambda_s \left\|z_1 - z_2\right\|_1 + \lambda_{feat}(\left\|G_s(z_1) - G_s(z_2)\right\|_1 + \left\|G_s(z_1) - z_v\right\|_1 + \left\|G_s(z_2) - z_v\right\|_1) \\ & \lambda^s_{pixel}(\left\|G(G_s(z_1)) - v\right\|_1 + \left\|G(G_s(z_2)) - v\right\|_1), \end{aligned} \tag{2}$$

where $G_s(z_1)$ is the video semantic code from shared code z_1; $G(G_s(z_1))$ is the video generated from code $G_s(z_1)$; and λ_s, λ_{feat}, and λ^s_{pixel} are weights for shared semantic codes, video semantic space, and reconstructed videos terms, respectively.

Shared Latent Space Divergency. The divergence measure $\mathcal{D}_s$ is defined as:

$$\mathcal{D}_s(q(z), p(z)) = \mathcal{L}_{D^s_z} + \mathcal{L}_{bucket}, \tag{3}$$

where $\mathcal{L}_{D^s_z}$ is defined considering a shared semantic space discriminator D^s_z between distribution samples z_1 and z_2 (similarly to Equation A.5), and $\mathcal{L}_{bucket}$ is a divergence loss based on a bucket approach.

In the *one-to-many* case, the text is represented by the same conditioning information, which is mapped to several semantic-related output videos, named a *bucket*. A bucket $\mathcal{B}_i$ is composed of videos with the same semantics of $t_i \in T$, such as $1 \leq i \leq N$ and N is the number of different text samples. The loss $\mathcal{L}_{bucket}$ is defined with the buckets available in a training batch and follows a contrastive approach between the similarities of intra- (same semantics) and inter-bucket (different semantics) samples. Given $z^s_v \sim\; p(z)$ and $z^s_t \sim q(z)$, we define the loss as:

$$\mathcal{L}_{bucket}\left(z^s_t, z^s_v\right) = \frac{\lambda_{neg}}{N_t}\left(\sum_{i=1}^{N_t} \sum_{j=1, j \notin \mathcal{B}_i}^{N_v} \frac{S_{ij}}{|\mathcal{B}_i|}\right) + \frac{\lambda_{pos}}{N_t}\left(\alpha - \sum_{i=1}^{N_t} \sum_{j=1, j \in \mathcal{B}_i}^{N_v} \lambda_{ij} \frac{S_{ij}}{|\mathcal{B}_i|}\right), \tag{4}$$

where $S_{ij} = \frac{1}{2}(\cos(z^s_t(i), z^s_v(j)) + 1)$ is the cosine similarity between embeddings $z^s_t(i)$ and $z^s_v(j)$ of the batch, λ_{ij} is a weight for the intra-bucket pair, which is set $\lambda_{ij} = 1$ if $i \neq j$ (the sample belongs to the bucket but is not the direct match in the batch) and $\lambda_{ij} = \alpha$ if $i = j$ (direct match of the batch). The left term of Equation 4 maintains inter-bucket samples far from each other, while the right term encourages intra-bucket samples to be closer to each other, where its direct match is reinforced to prevent collapse of mapping t_i to the same z^s_t and maintain sample diversity.

Note that, different from InfoNCE loss (van den Oord et al., 2018) or some extensions (Li et al., 2022; Yeh et al., 2022), we do not pose as positive samples only direct matches in the pairwise cosine similarity phase (or diagonal match). In fact, we consider all samples from a bucket as positive samples instead of negative samples by the masking approach of Equation 4.

4 Experiments

In this section, we first describe the main components of our evaluation protocol (Section 4.1). Then, we explore by a latent space perspective the *one-to-many* case showing its general structure (Section 4.2). This initiates our decoupling and understanding of this scenario, which starts by isolating the video autoencoder model that will form the target modality representation and how different architectures generate different target distributions (Section 4.2.1). Then, we analyze the learning of the semantic shared space between text and video in Section 4.2.2, in which we better understand how different models impact the alignment of the modalities. We finish our assessment with ablation experiments for the alignment process. To analyze the latent space, we adopt the dimensionality reduction methods: Principal Component Analysis (PCA), t-SNE (van der Maaten & Hinton, 2008), and UMAP (McInnes et al., 2018).

4.1 Implementation Details

Architectures. We used 3D convolutional deep neural networks for our probabilistic encoder E_v, deterministic decoder G, and discriminator D_v. For the discriminator D_z, mapping function M, video shared encoder E_s, and video semantic space generator G_s we used fully connected networks. For video representation, we considered latent spaces with dimension d_z and isotropic Gaussian prior distributions $p_z = \mathcal{N}(z; 0, \sigma^2 I_{d_z})$. We used different d_z depending on the video architecture, but maintained these values in all corresponding experiments and for all data sets. We did not optimize our model on the choice of d_z for any set.

We consider CLIP (Radford et al., 2021) text encoder with its pre-trained model from the ViT-B/32 version. For the video autoencoder (AE), we consider three architectures for comparison. The first is a 3D convolutional network extended from the 2D DCGAN (Radford et al., 2016) guidelines (3DConv-Base). This network does not use attention modules and residual blocks, although we added skip connections to improve its convergence. The second architecture (UNetLDM) is adapted from Rombach et al. (2022) which is based on latent diffusion. This network is extended with 3D convolutional and transposed operations and includes residual blocks and attention mechanisms. Beyond that, we also include the VDM (Ho et al., 2022) model based on diffusion to have a baseline comparison for video quality only, as this model was not proposed for representational learning with a posterior reconstruction decoder step. More details of the training setup can be found in Appendix A.

Data Sets and Metrics. We consider three data sets of increasing complexity that present the one-to-may case: Moving MNIST (Mittal et al., 2017) (SyncDraw-MM), KTH Human Action (Schuldt et al., 2004), and TACoS Multi-level Corpus (Rohrbach et al., 2014). For all sets, we sampled 16-frame videos with 64×64 pixels. Also, we consider objective full-reference measures to evaluate the quality of the video, which includes: Peak-Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM) (Wang et al., 2004), and the perceptual metrics LPIPS (Zhang et al., 2018) and DISTS (Ding et al., 2022). Furthermore, to evaluate the generated video distributions, we consider both Fréchet Video Distance (FVD) and Kernel Video Distance (KVD) (Unterthiner et al., 2018) metrics.

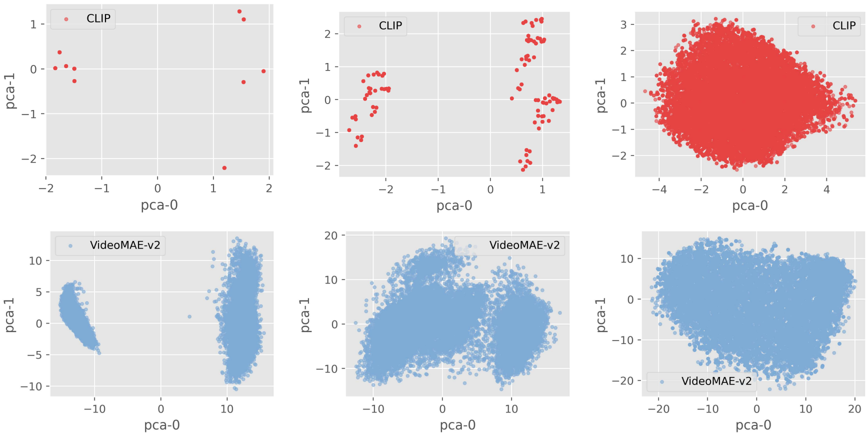
4.2 Latent Space Understanding of the One-to-Many Scenario

In this section, we analyze the alignment of text and video modalities through a latent space perspective. We start by presenting its general structure. Then, we analyze the learning of the target modality in an unsupervised way and follow to the alignment analysis between text and video. We first isolate each component of the cross-modality task to understand their impact in the overall result, as multiple factors could contribute to a satisfactory alignment between the modalities. Since the text modality is represented by a pre-trained model, we only consider its structure in this initial understanding of the mapping case.

The general visual structure of the one-to-many scenario is presented in Figure 2, with the latent spaces produced by CLIP text encoder (first row) and a video model (second row). We can notice that for the text spaces, the SyncDraw-MM set has fewer small clusters compared to KTH and TACoS, which present an increasing variety in the distributions. For the video modality, we investigate different AE approaches in Section 4.2.1, but to have an unbiased reference of the video latent space outside that domain, we considered the latent space generated with ViT-based embeddings obtained with VideoMAE-v2 (Wang et al., 2023), which was previously demonstrated by Ge et al. (2024) its effectiveness in representing video compared to other models commonly used in the FVD calculation. It can be seen in Figure 2, that the origin (text) modality is sparser and concentrated in different regions, indicating the same or close semantics while its corresponding target (video) modality presents a dense distribution for the video part of the same text-video pairs. In this regard, the SyncDraw-MM, KTH, and TACoS sets present a decreasing level of the one-to-many case, where TACoS presents a lower difficulty level of the task.

4.2.1 Learning a Video Representation for the One-to-Many Scenario

The video modality can be represented with an AE pre-trained model, but to understand the role of this part and the impact of different generated distributions on the alignment, we trained and evaluated different architectures. In Table 1, we present the quantitative results on video generation and the qualitative results are presented in Appendix A. Overall, the UNetLDM model obtained the best results for all data sets, except for TACoS, for which VDM (Ho et al., 2022) obtained the best ones considering FVD and KVD metrics. Nevertheless, the UNetLDM model, which considers a backbone adapted from a LDM model, generated satisfactory results without the diffusion process. Conversely, while the 3DConv-Base model does



(a) SyncDraw-MM **(b)** KTH **(c)** TACoS

Figure 2: Visualization of the text latent spaces generated with CLIP (Radford et al., 2021) (first row) along with the corresponding video latent spaces generated with VideoMAE-v2 (Wang et al., 2023) (second row), for each data set with their training split.

Table 1: Quantitative results of video autoencoder (AE) models on three data sets: SyncDraw-MM, KTH, and TACos. The best results between the sets are highlighted with a gray cell color in a column-wise comparison. Columns represent the corresponding metric and its values the model results over the test set. Notation: mean over the images (± standard deviation), ↑ indicates that higher is better and ↓ that lower values are better. PSNR is in decibel scale (db); SSIM results in $[0, 1]$; LPIPS, FVD and KVD in $[0, \infty]$.

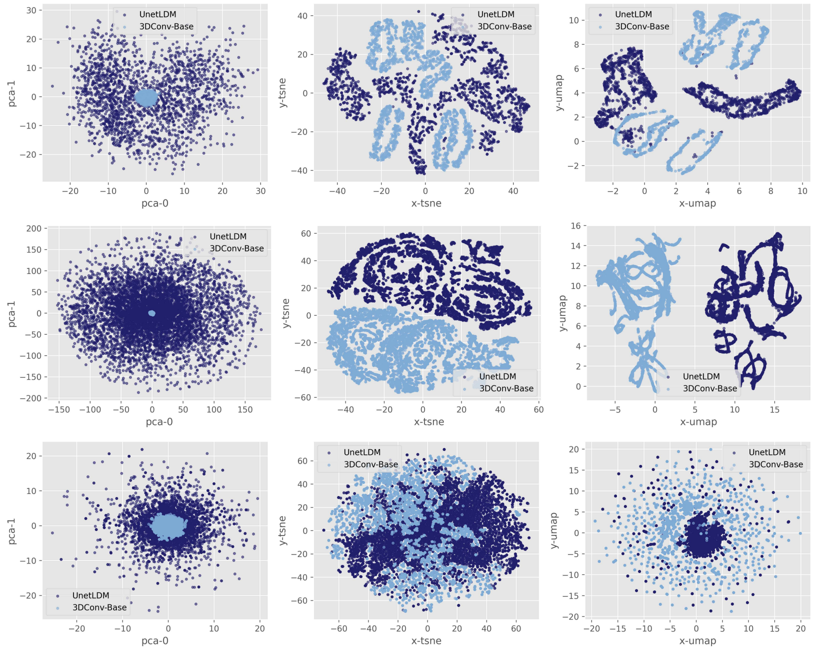
Data Set	Model \ Metrics	PSNR↑	SSIM↑	LPIPS↓	DISTS↓	FVD↓	KVD↓
SyncDraw-MM	3DConv-Base	19.1 ± 1.9	0.89 ± 0.03	0.08 ± 0.02	0.09 ± 0.02	2.62	0.003
	UNetLDM	27.8 ± 2.2	0.97 ± 0.01	0.02 ± 0.01	0.03 ± 0.01	0.27	0.0001
	VDM (Ho et al., 2022)	–	–	–	–	4.15	0.006
KTH	3DConv-Base	18.8 ± 2.7	0.41 ± 0.16	0.14 ± 0.07	0.24 ± 0.05	7.77	0.007
	UNetLDM	21.7 ± 2.6	0.52 ± 0.19	0.10 ± 0.07	0.20 ± 0.06	5.88	0.005
	VDM (Ho et al., 2022)	–	–	–	–	7.70	0.009
TACoS	3DConv-Base	18.6 ± 2.3	0.54 ± 0.07	0.08 ± 0.03	0.13 ± 0.02	26.18	0.047
	UNetLDM	18.7 ± 2.1	0.52 ± 0.07	0.06 ± 0.03	0.13 ± 0.02	19.67	0.039
	VDM (Ho et al., 2022)	–	–	–	–	10.79	0.013

not achieve optimal video quality, it produces satisfactory results with occasional reconstruction errors, such as confusing digit 1 with digit 7.

In Figure 3, the latent spaces obtained with both 3DConv-Base and UNetLDM are presented. Although both of them generate good reconstructions, they present different distributions regarding their latent spaces, where the latter generates a sparser space compared to 3DConv-Base. By analyzing each space alone, we can also observe that they present different clustering structures, indicating a different representational learning for the same task with the same training parameters and sets. How this can impact the alignment is explored in the next section.

4.2.2 Learning a Semantic-Shared Representation for the One-to-Many Scenario

In this section, we bridge the text and video modalities with our alignment method described in Section 3. To compose the video representation, we consider the models evaluated in the previous Section 4.2.1, and CLIP for the text representation. We also consider some alignment baselines for comparison, including: ImageBind (Girdhar et al., 2023), CoDi (Tang et al., 2023), and CLIP (Radford et al., 2021). Since we focus on the alignment aspect, we only consider this corresponding part from these approaches. This is done to isolate the impact of the alignment method from the modality representation learning, which could be learned in different ways and with different data sets.



(a) SyncDraw-MM

(b) KTH

(c) TACoS

Figure 3: Visualization of the joined video semantic spaces obtained with the 3DConv-Base and UnetLDM models for Syncdraw-MM, KTH, and TACoS data sets. For UnetLDM, a dimensionality reduction is done first to the same latent space dimension considered in 3DConv-Base with PCA.

For ImageBind-based alignment, we consider its approach that uses a projection layer from the video and text representation space to the semantic shared space. The CLIP-based alignment model considers our default mapping function architecture instead of a projection layer. Since CoDi (Tang et al., 2023) follows a different paradigm in representation learning, we defined an alternative inspired by their model. Instead of using the representation injection through cross-attention layers in all the modality autoencoders, which are based on LDMs, we considered the UnetLDM video AE model with our mapping function architecture, which is based on a non-sharing representational framework, i.e., the video embeddings are not passed through the text encoder or vice versa. For all baseline methods, except for the latter, we use the 3DConv-Base video AE. Also, for all baselines, InfoNCE (van den Oord et al., 2018) is used as alignment loss.

Note that in the text-to-video evaluation, different from distribution-based metrics (e.g., FVD and KVD), the full-reference metrics are compared to all elements in a *bucket* instead of just one direct match, since one input text can have different semantically aligned videos. Hence, a *bucket*-based metric compares a generated video to all videos in its corresponding bucket, the final result being the one with the best result. In Table 2, we present the quantitative results for these methods and, in Figures 4 and 5, we present the target video and semantic shared latent spaces obtained with them. We focus on this section on the latent space understanding but we also provide in Appendix A.4.2 qualitative results for these models.

Although the alignment approaches used different architectures and losses, they achieved similar quantitative results. It does not seem that a specific loss or architecture (projection layer versus progressive mapping) has a major impact on the result. Moreover, each data set that has a different complexity of the *one-to-many* mapping case suggests a different best model. However, when we also consider the latent spaces generated from each approach, this perspective changes.

Video Architectures on Alignment. Although the UnetLDM model provides one of the best results in video reconstruction, it is not the one from which the alignment generates the best text-to-video results. Note that one version of our method and the CoDi-based alternative rely on UnetLDM. Quantitatively, the CoDi-based alternative seems to produce slightly better or equivalent results to our approach, but producing different distributions in the video latent spaces. We expect a different alignment pattern since the corresponding latent space of this video AE is sparser than the one obtained with 3DConv-Base, as

Table 2: Quantitative results of the feature alignment between text and video modalities on three data sets: SyncDraw-MM, KTH, and TACoS. The best results between the sets are highlighted with a gray cell color in a column-wise comparison. Notation: "B" indicates bucket approach for the metric.

Data Set	Model \ Metrics	B-PSNR↑	B-SSIM↑	B-LPIPS↓	B-DISTS↓	FVD↓	KVD↓
SyncDraw-MM	CLIP-based	20.1 ± 1.6	0.858 ± 0.025	0.25 ± 0.06	0.14 ± 0.02	7.29	0.007
	CoDi-based	19.9 ± 1.3	0.868 ± 0.013	0.21 ± 0.06	0.13 ± 0.02	7.19	0.008
	ImageBind-based	20.0 ± 1.4	0.858 ± 0.033	0.23 ± 0.06	0.13 ± 0.02	7.78	0.009
	Our method - 3DConvBase	19.6 ± 1.4	0.855 ± 0.019	0.21 ± 0.05	0.14 ± 0.02	6.77	0.006
	Our method - UnetLDM	21.2 ± 1.6	0.889 ± 0.014	0.37 ± 0.03	0.15 ± 0.02	8.18	0.010
KTH	CLIP-based	21.2 ± 1.8	0.367 ± 0.094	0.26 ± 0.05	0.29 ± 0.04	31.86	0.037
	CoDi-based	20.7 ± 1.3	0.390 ± 0.067	0.24 ± 0.06	0.29 ± 0.05	23.45	0.021
	ImageBind-based	19.9 ± 2.5	0.34 ± 0.14	0.29 ± 0.08	0.31 ± 0.06	35.78	0.047
	Our method - 3DConvBase	20.7 ± 2.2	0.358 ± 0.088	0.26 ± 0.05	0.31 ± 0.04	27.51	0.028
	Our method - UnetLDM	19.8 ± 4.2	0.28 ± 0.15	0.24 ± 0.09	0.30 ± 0.08	23.58	0.015
TACoS	CLIP-based	17.9 ± 2.3	0.508 ± 0.081	0.14 ± 0.08	0.16 ± 0.04	27.22	0.049
	CoDi-based	17.3 ± 2.2	0.471 ± 0.079	0.20 ± 0.08	0.19 ± 0.04	26.30	0.049
	ImageBind-based	17.3 ± 2.1	0.478 ± 0.081	0.18 ± 0.07	0.18 ± 0.04	29.11	0.051
	Our method - 3DConvBase	17.9 ± 2.3	0.505 ± 0.083	0.14 ± 0.08	0.16 ± 0.04	27.57	0.050
	Our method - UnetLDM	17.3 ± 2.1	0.504 ± 0.079	0.29 ± 0.10	0.22 ± 0.05	58.88	0.103

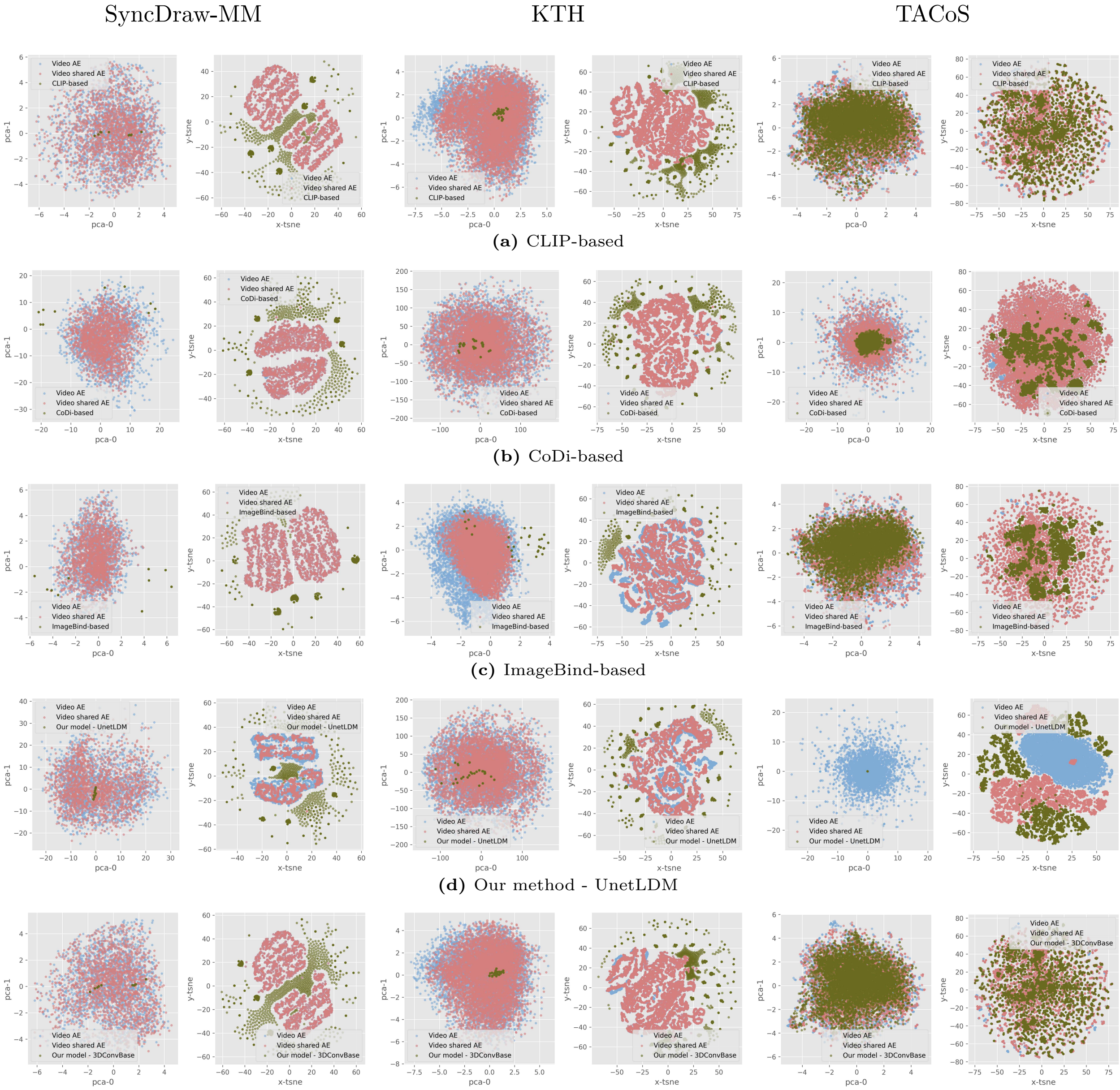
noticed in Section 4.2.1, but the model seems to partially affect this result, as in CoDi a better alignment is observed.

Furthermore, the video spaces produced by both models seem to indicate a concentration of the generated video codes in particular regions that do not align with the expected distribution. For our method, even the distribution obtained with the video shared AE, which maps the video latent codes to the semantic shared space between text and video, does not align with the "true" distribution (blue color). This result suggests that UnetLDM has difficulties in the alignment, which is partially associated with its architecture and the resulting sparser video space, as different losses are considered for this model, and the results demonstrate poor alignment and a level of collapse in the TACoS set—Figures 4(b) and 4(d).

Projection versus Progressive Approach. Regarding the use of projection layers versus our progressive architecture, we found that the latter generates more free collapse latent spaces, although it alone does not avoid it. From the latent spaces in Figure 4, we can observe more occurrences of small clusters using the ImageBind-based method. Beyond that, this method appears to present a slight misalignment of the video shared codes with the expected ones for the KTH set—Figure 4(c). This also occurs to some degree in other methods. But when we analyze the semantic shared space in Figure 5(c), we can observe this behavior with the projection-based model as well and also with the TACoS set, which has a lower level complexity of the *one-to-many* mapping problem. Hence, a progressive architecture seems to favor alignment compared to the use of a projection layer.

Progressive Alignment. Although we can see that the video codes originating from the video shared AE are correctly mapped to the video latent space, we can notice that this does not occur entirely for the embeddings coming from the text modality, especially at the higher difficulty levels of the task. This indicates a more straightforward task for the video shared AE to align the video latent space and the semantic shared space than the mapping from the text modality to the target video space. Note that in the video context, we still consider a *one-to-one* mapping, and only from the text modality this becomes our target problem.

By analyzing the resulting semantic shared spaces in Figure 5, we can notice poor alignments that propagate to the video space phase. For instance, the ImageBind-based method has well-separated distributions, even for TACoS. This set is also one for which the CoDi-based alternative generates poor alignment. The ones producing the best alignments, in a latent space perspective, seem to be the CLIP-based and our method, both considering 3DConv-Base. However, they still present a cluster concentration of the generated distribution, which could also indicate a certain level of misalignment between the generated and target distributions (e.g., SyncDraw-MM and KTH columns in Figure 4).



(a) CLIP-based

(b) CoDi-based

(c) ImageBind-based

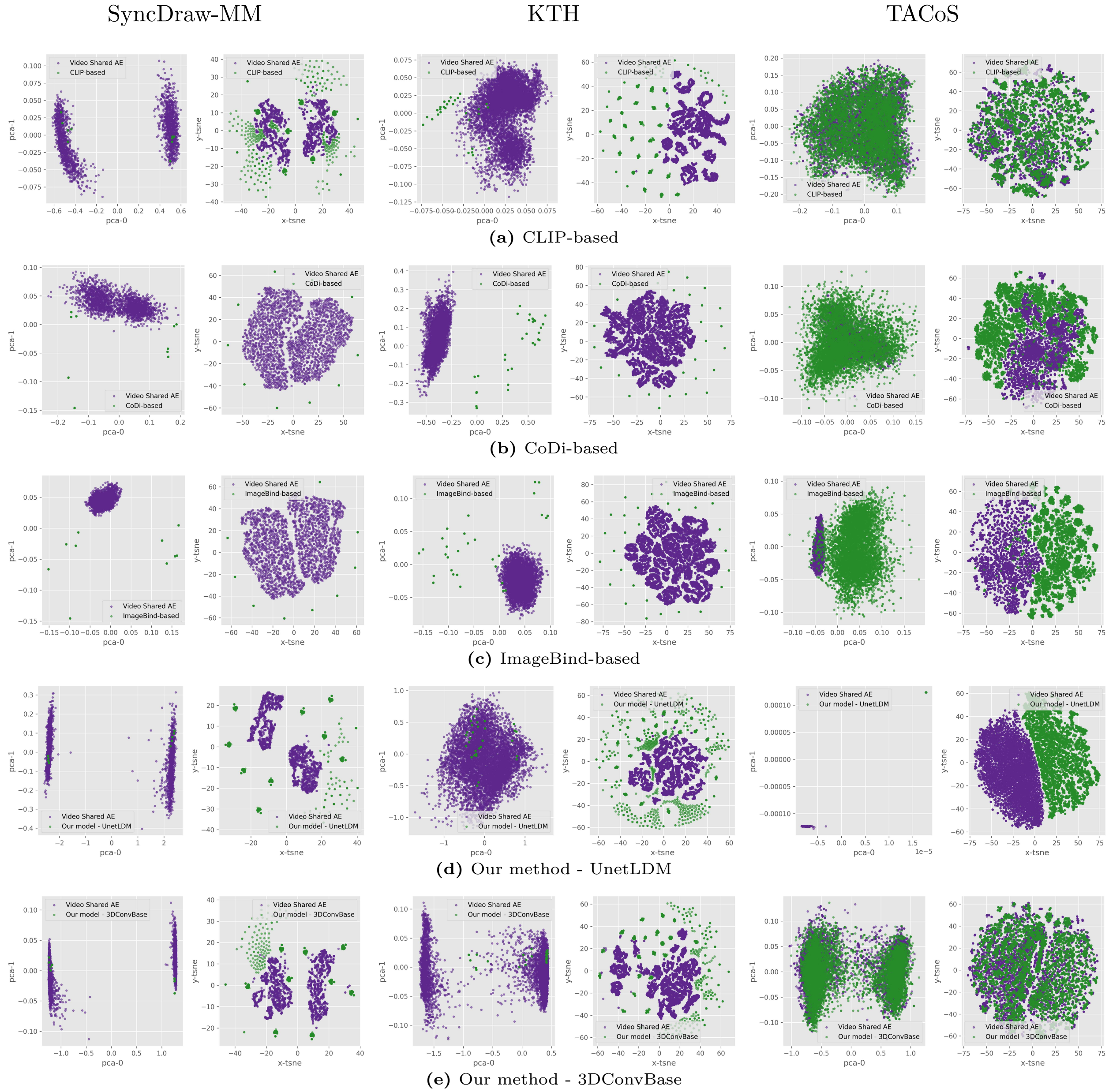
(d) Our method - UnetLDM

(e) Our method - 3DConvBase

Figure 4: Visualization of the video latent space produced by the feature alignment methods, with the embeddings predicted from the corresponding mapping function and the ones obtained from the video AE used in each experiment. Results for SyncDraw-MM (columns 1-2), KTH (3-4), and TACoS (5-6) sets using PCA (odd columns) and t-SNE (even columns).

4.2.3 Ablation experiments

We perform ablation experiments to understand the impact of our progressive approach and loss in the alignment process. Other experiments for video representation learning are included in Appendix A.4.1. In this section, we considered the SyncDraw-MM set as the one-to-many scenario is more prominent here, and the 3DConv-Base video AE baseline. We evaluate a non-progressive method by directly aligning the text into the video representation without an intermediary step, considering both our adapted loss and InfoNCE (van den Oord et al., 2018). Also, we evaluate our progressive approach adapted with self-supervised techniques, such as: VicReg (Bardes et al., 2022) and BYOL (Grill et al., 2020). We adapt these techniques by considering each stream of augmentation data as a data modality stream (e.g., text and video streams instead



(a) CLIP-based

(b) CoDi-based

(c) ImageBind-based

(d) Our method - UnetLDM

(e) Our method - 3DConvBase

Figure 5: Visualization of the semantic shared latent space produced by the feature alignment methods, with the embeddings predicted from the corresponding mapping function and the ones from the video shared AE used in each experiment. Results for SyncDraw-MM (columns 1-2), KTH (3-4), and TACoS (5-6) sets using PCA (odd columns) and t-SNE (even columns).

of two views of the same modality). We empirically found that SimSiam (Chen & He, 2021) obtained similar results to BYOL, so we only report BYOL. In Table 3, we present the quantitative results. The qualitative results and latent space exploration are presented in Appendix A.4.2.

Regarding the non-progressive versus progressive approach, there is little change considering only the full-reference metrics for the generated videos, being the distribution-based results more outstanding and in favor of the progressive method. But, by analyzing the produced latent spaces, we observe that the non-progressive mapping generates more concentrated regions outside the expected distributions, also showing more distant alignment between the modalities in the video space compared to the progressive mapping. Now, considering the SS approaches of VicReg and BYOL, both generate similar structures in video and semantic shared latent spaces but not identical, with VicReg generating a slightly better quantitative result.

Table 3: Quantitative results of the ablation experiments on the feature alignment between text and video modalities on the SyncDraw-MM data set with 3DConv-Base video autoencoder.

Model/Metrics	B-PSNR ↑	B-SSIM ↑	B-LPIPS ↓	B-DISTS ↓	FVD ↓	KVD ↓
Non-progressive	19.7 ± 0.9	0.856 ± 0.014	0.20 ± 0.01	0.13 ± 0.01	8.47	0.011
Non-progressive w/ InfoNCE	19.0 ± 1.4	0.841 ± 0.027	0.20 ± 0.05	0.14 ± 0.02	8.10	0.008
Progressive - VicReg	19.6 ± 1.7	0.852 ± 0.036	0.20 ± 0.05	0.13 ± 0.02	6.67	0.006
Progressive - BYOL	19.6 ± 1.6	0.856 ± 0.032	0.19 ± 0.05	0.13 ± 0.02	7.14	0.007

We can notice that the video space distribution is similar to our model and the CLIP-based alignment (Figures 4(a) and 4(e)), but they generate different structures in their semantic shared spaces compared to the best models on Section 4.2.2, which split the distribution into two regions, also presenting a slightly more concentrated space when we compare the feature range of the spaces.

Furthermore, both branches of evaluation show the difficulty in aligning the text and video modalities with the SyncDraw-MM set and demonstrate that even considering different strategies, those strategies do not take into account the challenging one-to-many mapping.

5 Limitations

We explored a particular class of video architectures for representation learning based on autoencoder models. Other architectures such as the Vision Transformer (ViT) from VideoMAE (Wang et al., 2023) could be adapted with a full video decoder. Moreover, the representation learning method considered for the target modality could be extended to other approaches with different assumptions on the data distribution to understand its impact on the video semantic space, such as relational regularization (Xu et al., 2020) and diffusion-based VampPriors (Kuzina & Tomczak, 2024). Furthermore, we focus on a scenario in which the video target modality is represented by fixed chunks of video. Future work can consider merging these chunks of video and preserving spatial and temporal consistency for a longer video generation based on a cross-modality approach with feature alignment.

6 Discussion and Conclusions

We give light to an implicit problem of cross-modality generation that is currently underexplored. Although modality alignment in a semantic shared space could benefit from knowledge obtained by pre-trained models for each end-data modality, a challenging aspect arises related to how to map both modalities. We show that the one-to-many case have different levels of complexity in different data sets and impacts the overall result of the text-to-video generation from a semantic shared space. Moreover, this task lacks effective quantitative metrics for its evaluation, which require complementary methods for a robust assessment.

In this work, we focus on autoencoder models as their nature implicitly enables representation learning of the modality and can be adapted for cross-modality generation by feature alignment. We show how some components of this task affect the overall result and demonstrate that video representation plays an important role in it. Overall, tackling the one-to-many case is not straightforward, requiring a different view when considering a semantically shared space between modalities, as current methods and regularization techniques are not designed with this case in mind.

References

Jimmy Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization. *CoRR*, abs/1607.06450, 2016.

Jianmin Bao, Dong Chen, Fang Wen, Houqiang Li, and Gang Hua. CVAE-GAN: Fine-grained image generation through asymmetric training. In *IEEE Inter. Conf. Comput. Vis. (ICCV)*, pp. 2764–2773, 2017.

Adrien Bardes, Jean Ponce, and Yann LeCun. VICReg: Variance-invariance-covariance regularization for self-supervised learning. In *Inter. Conf. Learn. Represent. (ICLR)*, 2022.

Olivier Bousquet, Sylvain Gelly, Ilya Tolstikhin, Carl Johann Simon-Gabriel, and Bernhard Schölkopf. From optimal transport to generative modeling: the vegan cookbook, 2017.

D C. Dowson and B V. Landau. The Fréchet distance between multivariate normal distributions. *J. Multivar. Anal.*, 12:450–455, 09 1982.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *Adv. Neural Inf. Process. Sys. (NeurIPS)*, volume 33, pp. 9912–9924, 2020.

João Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *IEEE/CVF Inter. Conf. Comput. Vis. Pattern Recog. (CVPR)*, pp. 4724–4733, 2017.

Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *IEEE/CVF Inter. Conf. Comput. Vis. Pattern Recog. (CVPR)*, pp. 15745–15753, 2021. doi: 10.1109/CVPR46437.2021.01549.

Andrew M. Dai and Quoc V. Le. Semi-supervised sequence learning. In *Adv. Neural Inf. Process. Sys. (NeurIPS)*, pp. 30793087, 2015.

Keyan Ding, Kede Ma, Shiqi Wang, and Eero P. Simoncelli. Image quality assessment: Unifying structure and texture similarity. *IEEE Trans. Pattern Anal. Mach. Intell.*, 44(5):2567–2581, 2022.

Zhiyu Fang, Xiaobin Zhu, Chun Yang, Zheng Han, Jingyan Qin, and Xu-Cheng Yin. Learning aligned cross-modal representation for generalized zero-shot classification. In *AAAI Conf. Artif. Intell. (AAAI)*, pp. 6605–6613, 2022. doi: 10.1609/AAAI.V36I6.20614.

William Fedus, Mihaela Rosca, Balaji Lakshminarayanan, Andrew M. Dai, Shakir Mohamed, and Ian J. Goodfellow. Many paths to equilibrium: GANs do not need to decrease a divergence at every step. In *Inter. Conf. Learn. Represent. (ICLR)*, 2018.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. SimCSE: Simple contrastive learning of sentence embeddings. In *Conf. Empir. Methods Nat. Lang. Process. (EMNLP)*, pp. 6894–6910, 2021. doi: 10.18653/v1/2021.emnlp-main.552.

Itai Gat, Guy Lorberbom, Idan Schwartz, and Tamir Hazan. Latent space explanation by intervention. In *AAAI Conf. Artif. Intell. (AAAI)*, pp. 679–687, 2022. doi: 10.1609/AAAI.V36I1.19948.

Songwei Ge, Thomas Hayes, Harry Yang, Xi Yin, Guan Pang, David Jacobs, Jia-Bin Huang, and Devi Parikh. Long video generation with time-agnostic vqgan and time-sensitive transformer. In *European Conf. Comput. Vis. (ECCV)*, pp. 102–118, 2022.

Songwei Ge, Aniruddha Mahapatra, Gaurav Parmar, Jun-Yan Zhu, and Jia-Bin Huang. On the content bias in frechet video distance. In *IEEE/CVF Inter. Conf. Comput. Vis. Pattern Recog. (CVPR)*, pp. 7277–7288, 2024.

Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Imagebind: One embedding space to bind them all. In *IEEE/CVF Inter. Conf. Comput. Vis. Pattern Recog. (CVPR)*, pp. 15180–15190, 2023.

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial nets. In *Adv. Neural Inf. Process. Sys. (NeurIPS)*, pp. 2672–2680, 2014.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, koray kavukcuoglu, Remi Munos, and Michal Valko. Bootstrap your own latent - a new approach to self-supervised learning. In *Adv. Neural Inf. Process. Sys. (NeurIPS)*, volume 33, pp. 21271–21284, 2020.

Ishaan Gulrajani, Faruk Ahmed, Martín Arjovsky, Vincent Dumoulin, and Aaron C. Courville. Improved training of wasserstein GANs. In *Adv. Neural Inf. Process. Sys. (NeurIPS)*, pp. 5769–5779, 2017.

Yingqing He, Tianyu Yang, Yong Zhang, Ying Shan, and Qifeng Chen. Latent video diffusion models for high-fidelity long video generation. *CoRR*, abs/2211.13221, 2022.

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 *Adv. Neural Inf. Process. Sys. (NeurIPS)*, pp. 66296640, 2017.

Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Adv. Neural Inf. Process. Sys. (NeurIPS)*, volume 35, pp. 8633–8646. Curran Associates, Inc., 2022.

Tianyang Hu, Fei Chen, Haonan Wang, Jiawei Li, Wenjia Wang, Jiacheng Sun, and Zhenguo Li. Complexity matters: Rethinking the latent space for generative modeling. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Adv. Neural Inf. Process. Sys. (NeurIPS)*, volume 36, pp. 29558–29579. Curran Associates, Inc., 2023.

Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European Conf. Comput. Vis. (ECCV)*, pp. 694–711, 2016.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger (eds.), *Adv. Neural Inf. Process. Sys. (NeurIPS)*, volume 25. Curran Associates, Inc., 2012.

Anna Kuzina and Jakub M. Tomczak. Hierarchical VAE with a diffusion-based vampprior. *Trans. Mach. Learn. Res.*, 2024. ISSN 2835-8856.

Yann Lecun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proc. IEEE*, 86(11):2278–2324, 1998.

Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N. Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick van Kleef, S. Auer, and Christian Bizer. DBpedia—a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*, 6:167–195, 2015.

Yangguang Li, Feng Liang, Lichen Zhao, Yufeng Cui, Wanli Ouyang, Jing Shao, Fengwei Yu, and Junjie Yan. Supervision exists everywhere: A data efficient contrastive language-image pre-training paradigm. In *Inter. Conf. Learn. Represent. (ICLR)*, 2022.

Paul Pu Liang, Amir Zadeh, and Louis-Philippe Morency. Foundations & trends in multimodal machine learning: Principles, challenges, and open questions. *ACM Computing Surveys*, 56(10):1–42, 2024.

Valentino Maiorca, Luca Moschella, Antonio Norelli, Marco Fumero, Francesco Locatello, and Emanuele Rodolà. Latent space translation via semantic alignment. In *Adv. Neural Inf. Process. Sys. (NeurIPS)*, 2023.

Tanya Marwah, Gaurav Mittal, and Vineeth N. Balasubramanian. Attentive semantic video generation using captions. In *IEEE Inter. Conf. Comput. Vis. (ICCV)*, 2017.

L. McInnes, J. Healy, and J. Melville. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *CoRR*, abs/1802.03426, 2018.

Gaurav Mittal, Tanya Marwah, and Vineeth N. Balasubramanian. Sync-DRAW: Automatic GIF generation using deep recurrent attentive architectures. In *ACM Inter. Conf. Multimedia (MM)*, pp. 1096–1104, 2017.

Moritz Piening and Matthias Chung. Paired wasserstein autoencoders for conditional sampling. *CoRR*, abs/2412.07586, 2024.

Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In Yoshua Bengio and Yann LeCun (eds.), *Inter. Conf. Learn. Represent. (ICLR)*, 2016.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Inter. Conf. Mach. Learn. (ICML)*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 2021.

Prajit Ramachandran, Barret Zoph, and Quoc V Le. Searching for activation functions. *CoRR*, abs/1710.05941, 2017.

Anna Rohrbach, Marcus Rohrbach, Wei Qiu, Annemarie Friedrich, Manfred Pinkal, and Bernt Schiele. Coherent multi-sentence video description with variable level of detail. In *German Conference on Pattern Recognition, GCPR*, 2014.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *IEEE/CVF Inter. Conf. Comput. Vis. Pattern Recog. (CVPR)*, pp. 10684–10695, 2022.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, Xi Chen, and Xi Chen. Improved techniques for training GANs. In *Adv. Neural Inf. Process. Sys. (NeurIPS)*, pp. 2234–2242, 2016.

Christian Schuldt, Ivan Laptev, and Barbara Caputo. Recognizing human actions: a local svm approach. In *IEEE Inter. Conf. Pattern Recog. (ICPR)*, pp. 32–36, 2004.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *Inter. Conf. Learn. Represent. (ICLR)*, 2015.

Zineng Tang, Ziyi Yang, Chenguang Zhu, Michael Zeng, and Mohit Bansal. Any-to-any generation via composable diffusion. In *Adv. Neural Inf. Process. Sys. (NeurIPS)*, 2023.

Thomas Theodoridis, Theocharis Chatzis, Vassilios Solachidis, Kosmas Dimitropoulos, and Petros Daras. Cross-modal variational alignment of latent spaces. In *IEEE Inter. Conf. Comput. Vis., Pattern Recog. Wksps. (CVPRW)*, pp. 4127–4136, 2020. doi: 10.1109/CVPRW50498.2020.00488.

Ilya Tolstikhin, Olivier Bousquet, Sylvain Gelly, and Bernhard Schölkopf. Wasserstein auto-encoders. In *Inter. Conf. Learn. Represent. (ICLR)*, 2018.

Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. In *Adv. Neural Inf. Process. Sys. (NeurIPS)*, volume 35, pp. 10078–10093, 2022.

Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphaël Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges. *CoRR*, abs/1812.01717, 2018.

Aäron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *CoRR*, abs/1807.03748, 2018.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *J. Mach. Learn. Res.*, 9(86): 2579–2605, 2008.

Limin Wang, Bingkun Huang, Zhiyu Zhao, Zhan Tong, Yinan He, Yi Wang, Yali Wang, and Yu Qiao. Videomae v2: Scaling video masked autoencoders with dual masking. In *IEEE/CVF Inter. Conf. Comput. Vis. Pattern Recog. (CVPR)*, pp. 14549–14560, 2023.

Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *IEEE/CVF Inter. Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2018.

Yanhui Wang, Jianmin Bao, Wenming Weng, Ruoyu Feng, Dacheng Yin, Tao Yang, Jingxu Zhang, Qi Dai, Zhiyuan Zhao, Chunyu Wang, Kai Qiu, Yuhui Yuan, Xiaoyan Sun, Chong Luo, and Baining Guo. Microcinema: A divide-and-conquer approach for text-to-video generation. In *IEEE/CVF Inter. Conf. Comput. Vis. Pattern Recog. (CVPR)*, pp. 8414–8424, June 2024.

Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Trans. Image Process.*, 13(4):600–612, 2004.

Hongteng Xu, Dixin Luo, Ricardo Henao, Svati Shah, and Lawrence Carin. Learning autoencoders with relational regularization. In *Inter. Conf. Mach. Learn. (ICML)*, pp. 10576–10586, 2020.

Wenju Xu, Shawn Keshmiri, and Guanghui Wang. Stacked wasserstein autoencoder. *Neurocomputing*, 363: 195–204, 2019. ISSN 0925-2312.

Hongwei Xue, Tiankai Hang, Yanhong Zeng, Yuchong Sun, Bei Liu, Huan Yang, Jianlong Fu, and Baining Guo. Advancing high-resolution video-language representation with large-scale video transcriptions. In *IEEE/CVF Inter. Conf. Comput. Vis. Pattern Recog. (CVPR)*, pp. 5036–5045, June 2022.

Hongwei Xue, Yuchong Sun, Bei Liu, Jianlong Fu, Ruihua Song, Houqiang Li, and Jiebo Luo. Clip-vip: Adapting pre-trained image-text model to video-language representation alignment. In *Inter. Conf. Learn. Represent. (ICLR)*, 2023.

Chun-Hsiao Yeh, Cheng-Yao Hong, Yen-Chi Hsu, Tyng-Luh Liu, Yubei Chen, and Yann LeCun. Decoupled contrastive learning. In *European Conf. Comput. Vis. (ECCV)*, pp. 668684, 2022.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE/CVF Inter. Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2018.

A Appendix

A.1 Implementation Details

In this section, we present additional details about the video semantic space method proposed for the cross-modality task, the data sets, metrics, architectures, and the latent space understanding protocol.

A.2 Video Semantic Space

To generate videos, we need to learn a distribution $p(v|z_v)$ that is conditioned on our semantic space and that is similar to the original video data $p^*(v)$. Toward this goal, we minimize the Wasserstein distance between both distributions by using its dual form (Bousquet et al., 2017; Tolstikhin et al., 2018) of optimizing through random encoders $q(z_v \mid v)$ instead of the original distribution couplings. Hence, we minimize

$$D_{\mathrm{W}}\left(p^*(v), p(v \mid z_v)\right) = \inf_{q(z_v \mid v) \in \mathcal{Q}} \left\{ \mathop{\mathbb{E}}_{x \sim p^*(v)} \mathop{\mathbb{E}}_{z_v \sim q(z_v \mid v)} \left[c\left(x, y\right)\right] + \lambda_z \mathcal{D}(q(z_v), p(z_v)) \right\}, \tag{A.1}$$

where $\mathcal{Q}$ is a non-parametric set of probabilistic encoders, $p(v \mid z_v)$ is our generative distribution, $x \sim p^*(v)$ is a ground truth video, $y \sim p(v \mid z_v)$ is a generated video (through decoder G) that depends on the semantic vector $z_v \sim q(z_v \mid v)$, and $\lambda_z > 0$ is a hyperparameter that weights the divergence measure $\mathcal{D}$ between the marginal distribution $q(z_v) = \mathbb{E}_{v \sim p^*(v)}\left[q(z_v \mid v)\right]$ and the prior $p(z_v)$ for our semantic space, and c is a similarity cost.

Video Similarity. The cost function c represents a measure between two videos, which we define as

$$c(x, y) = \lambda_{pixel} \left\| x - y \right\|_1 + \lambda_f \left\| f_{D_v}(x) - f_{D_v}(y) \right\|_1 + \lambda_p \left\| f_{\mathrm{VGG}}(x) - f_{\mathrm{VGG}}(y) \right\|_2^2, \tag{A.2}$$

where $f_{D_v}(x)$ denotes the features of an intermediate layer of the video discriminator D_v, when considering video x; similarly, $f_{\mathrm{VGG}}(x)$ denotes the features of a VGG19 network (Johnson et al., 2016); and $\lambda_{pixel} > 0$, $\lambda_f > 0$, and $\lambda_p > 0$ are hyperparameters that define the weight of each term in the final cost.

This cost function penalizes the discrepancy between the videos on the pixel (left term) and feature space (middle to right term). The penalization on the feature space acts as a perceptual similarity measure between the original and generated samples, since pixel-wise metrics have difficulties capturing perceptual properties of the reconstructed samples. Our perceptual measure is defined as a feature-matching loss (Bao et al., 2017; Salimans et al., 2016) over feature space f_{D_v} of discriminator D_v and feature space f_{VGG}. We introduce the details of D_v later in this section.

Video Latent Space Divergency. The divergence $\mathcal{D}$ represents a cost on the difference between two given spaces. In the original WAE (Tolstikhin et al., 2018), this divergence is obtained using a GAN or Maximum Mean Discrepancy approach. In contrast, we consider a metric based on feature matching (Salimans et al., 2016), which we found to be more stable to train. We convert the WAE-GAN divergence (Tolstikhin et al., 2018), defined as a non-saturating loss (Fedus et al., 2018; Goodfellow et al., 2014), into a distance minimization problem between the semantic feature spaces, f_{D_z}, of both $q(z_v)$ and $p(z_v)$. We empirically found that removing the min-max between the autoencoder (i.e., E_v and G) and the discriminator D_z led to a more stable training compared to the original WAE-GAN loss. Adding a gradient penalty (Fedus et al., 2018; Gulrajani et al., 2017) also leads to stable training, but we found that the feature matching term was enough to stabilize video training. Hence, we define the divergence as the aggregate

$$\mathcal{D}(q(z_v), p(z_v)) = \mathcal{L}_f + \mathcal{L}_{D_z} + \mathcal{L}_{D_v}, \tag{A.3}$$

where the losses $\mathcal{L}_{(\cdot)}$ depend on the same arguments as $\mathcal{D}$. The feature-matching loss $\mathcal{L}_f$ penalizes the semantic feature space induced by discriminator D_z, when it learned to distinguish between the true and a variational approximation of the semantic distributions. The video adversarial loss, $\mathcal{L}_{D_v}$, measures the similarity in the perceptual space as similar videos will have similar underlying semantic distributions, and the semantic discriminator loss, $\mathcal{L}_{D_z}$, induces similarity between prior and approximated semantic distributions.

Table A.1: Data set splits used for training and testing containing the number of text and video pairs along with its corresponding number of buckets.

Model	Train	Validation	Test	Buckets
SyncDraw-MM	10000	2000	2000	20
KTH	21030	5502	6650	150
TACoS	31392	7848	9811	11659

We consider the feature-matching loss as

$$\mathcal{L}_f(q(z_v), p(z_v)) = \mathop{\mathbb{E}}_{\tilde{z_v} \sim p(z_v)} \mathop{\mathbb{E}}_{z_v \sim q(z_v)} \left\| f_{D_z}(\tilde{z_v}) - f_{D_z}(z_v) \right\|_2^2, \tag{A.4}$$

such as $f_{D_z}(z_v)$ denotes the features of an intermediate layer of D_z when considering the latent vector z_v, and the joint semantic space $p(z_v)$ is modeled as a multivariate normal distribution.

Then, we define the semantic discriminator loss to penalize the difference between the true distribution, $p(z_v)$, and our approximation, $q(z_v)$, as

$$\mathcal{L}_{D_z} = -\mathop{\mathbb{E}}_{\tilde{z_v} \sim p(z_v)} [\log D_z(\tilde{z_v})] - \mathop{\mathbb{E}}_{z_v \sim q(z_v)} [\log(1 - D_z(z_v))], \tag{A.5}$$

where D_z is the semantic space discriminator. Finally, the video discriminator D_v, from which we compute f_{D_v} in Equation A.2, tries to differentiate between real, $p^*(v)$, and generated videos, $p(v \mid z_v)$, with a loss similar to Equation A.5 but now considering the videos samples instead of the semantic vectors.

A.3 Data sets

Moving MNIST is an extension of the MNIST (Lecun et al., 1998) data set where one or two digits move up and down, left to right, and vice versa. Each video has a sentence describing the digits and their moving direction. For the KTH dataset, which contains videos of several actions of 25 persons recorded in four different backgrounds with variations in light and clothing, we selected a subset of these actions (i.e., walking, jogging, and running) as in Mittal et al. (2017) and Marwah et al. (2017) experiments. We also provide a new set of text descriptions for this data.[2] Each text description indicates the person in the video, its corresponding action, and direction of movement, such as “person 2 is walking left to right” and “person 5 is jogging right to left.” Lastly, the TACoS dataset contains videos of people cooking with multilevel descriptions, such as one sentence, short, and detailed descriptions for each video. In our experiments, we selected the set of short descriptions, which more enclosed the *one-to-many* case, each depicting an event for a time interval in the video.

We present the data sets splits we used in the experiments in Table A.1 and examples of buckets in Figure A.1.

A.3.1 Metrics

We used the official implementation of LPIPS (Zhang et al., 2018)[3] and the AlexNet (Krizhevsky et al., 2012) backbone to calculate the metric values. Other parameters were defined with the default values used in the official code. We used the official implementation of DISTS (Ding et al., 2022)[4] and the PyTorch version of the metric. The default backbone used was the one based on VGG16 (Simonyan & Zisserman, 2015) with the default repository parameters.

For distribution-based metrics, we considered the following: FVD is calculated with the I3D video features (Carreira & Zisserman, 2017) extracted from the model (RGB stream) available on Kinects-I3D[5] with an extension of the FID metric from Heusel et al. (2017)[6]; and KVD with the polynomial MMD (Unterthiner et al., 2018).

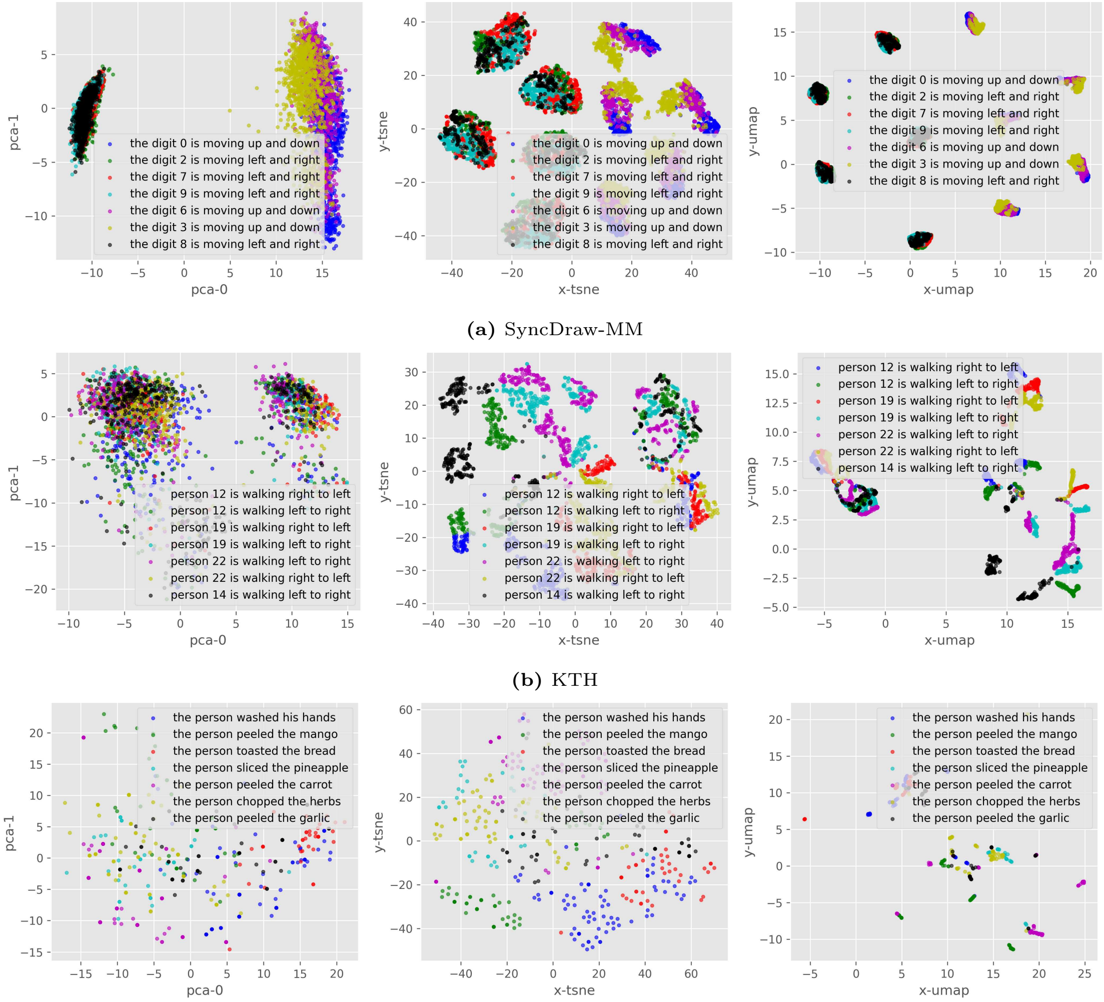
[2] Mittal et al. (2017) and Marwah et al. (2017) also generated a set of text descriptions for the KTH dataset, but they are not publicly available.

[3] https://github.com/richzhang/PerceptualSimilarity

[4] https://github.com/dingkeyan93/DISTS

[5] https://github.com/google-deepmind/kinetics-i3d

[6] https://github.com/bioinf-jku/TTUR/



(a) SyncDraw-MM

(b) KTH

(c) TACoS

Figure A.1: Samples of seven random buckets visualized in the video latent spaces generated with the VideoMAE-v2 (Wang et al., 2023) encoder, for each data set. The columns results correspond to: PCA, t-SNE, and UMAP, respectively.

A.3.2 Latent Space Understanding

To analyze the latent space, we adopt the dimensionality reduction methods: Principal Component Analysis (PCA), t-SNE (van der Maaten & Hinton, 2008), and UMAP (McInnes et al., 2018). The t-SNE visualization is produced by first reducing the input dimensionality to 32 components with PCA, and then applying t-SNE over the resulting components with a perplexity of 40 and a number of iterations equal to 600 for all visualizations.

For the VideoMAE (Tong et al., 2022; Wang et al., 2023) representation, we selected the VideoMAE-v2 (Wang et al., 2023) model, more specifically the Hybrid-PT-SSv2-FT version used in the work of Ge et al. (2024)[7]. The ViT-g encoder features were extracted following their guidelines, generating embeddings with dimension 1408 from the penultimate layer of the encoder that were averaged across all patches.

[7] https://github.com/songweige/content-debiased-fvd

Table A.2: Size of the networks and components used in this work.

Model	Number of parameters
3DConv-Base	9.9M
UnetLDM	268.8M
VDM (Ho et al., 2022)	35.7M
Mapping Function M ($d_{z_t} = 512$ and $d_{z_s} = 64$)	824k
Video semantic shared AE (E_s and G_s) ($d_{z_v} = 64$ and $d_{z_s} = 64$)	1.2M
Components	
Discriminator D_s or D_z with $d_z = 64$	133K
Discriminator D_v (PatchHD-Video)	2.6M
VGG19 (Johnson et al., 2016) network	20M

A.3.3 Architectures

In the next sections, we present more details of each model training setup. We present in Table A.2 an overview of the number of parameters of the models used for the video autoencoder, mapping function and video semantic shared autoencoder.

Progressive Decoupling

In the decoupling process, we also consider a second text description input $\hat{t}_i$ from t_i, where a noise word is added with probability $p = 0.15$ to include variation in text representation in the same bucket b_i, but having the bucket loss considering the original text embeddings t_i. The word dictionary from which the noise sample is obtained did not include any words from the corresponding data set corpus. We also evaluated dropout noise (Gao et al., 2021), but empirically found that the addition of random words worked better. Word removal, on the other hand, was not suitable as it directly interferes with the original bucket semantics since removing some words could join samples from originally different buckets.

For the progressive decoupling architecture, we considered a multilayer perceptron (MLP) with four layers. Except for the last layer, each was defined with a hidden layer size of 512 and is followed by a Layer Normalization (Ba et al., 2016) and Swish activation function (Ramachandran et al., 2017). A dropout layer is used after the second and third layers with a rate of 0.1. This was the base network used for the mapping function and video shared AE, changing only the input and output dimensions to match the corresponding representation sizes. The regularization coefficients were defined as $\lambda_{z_s} = 5.0$, $\lambda_s = 100$, $\lambda_{feat} = 10$, $\lambda^s_{pixel} = 30$. In addition, for the bucket loss, we define $\lambda_{neg} = 1.0$, $\lambda_{pos} = 1.0$, and $\alpha = 2.0$ to weight the direct text-video pairs of the bucket. The training setup also considered Adam optimizer with a learning rate of 10^{-4} with a global clip norm (maximum gradient norm of 4.0). We trained the models for about 70 epochs with varying batch size, which depends on the video AE used and the cross-modality alignment approach, of $32 - 100$.

Video Models

For the video pixel-based discriminator D_v, we adapted the Patch discriminator from Pix2PixHD (Wang et al., 2018) that evaluates video quality on multiple scales. For the video representation, we considered the dimensions: $d_z = 64$ for 3DConv-Base and $d_z = 128$ for UnetLDM. Other regularization coefficients were defined as $\lambda_{pixel} = 10$, $\lambda_f = 10$, $\lambda_z = 5$. In particular, we defined $\lambda_p = 0.0025$ since this term dominated other terms in the final loss and the setting with this value presented satisfactory results in perceptual quality. In this case, the perceptual weight is defined over the VGG19 layers: `block4_conv3` and `block5_conv4`. The training setup for the video AE considered Adam optimizer with a learning rate of 10^{-4} with a global clip norm (maximum gradient norm of 5.0). We trained the video models for about 100 epochs with varying batch size of $32 - 100$, for UnetLDM and 3DConvBase, respectively.

For the video cost in Equation A.2, we found empirically that an L1-based distance converged better for the pixel and feature discriminator terms, while an L2-square distance worked better for perceptual loss.

Text Models

We evaluated the CLIP (Radford et al., 2021) text encoder, which is considered with its pre-trained model from the ViT-B/32 version. The CLIP method used was based on the `transformers` package[8] using the pre-trained model with key `openai/clip-vit-base-patch32` generating a 512-dimensional embedding.

The word dictionary considered as the noise set for sampling a noise word for the text in the cross-modality alignment was built based on DBPedia (Lehmann et al., 2015)[9] and is processed similarly to Dai & Le (2015). First, we treat punctuation as separate tokens. Then, we ignore any non-English characters and words. Since the removal of non-English words can affect the semantics of the text, we also remove entries that have too many `UNKNOWN` tokens after this preprocessing. We have defined a maximum value of 45% of unknown tokens to be considered a valid entry for the set. We also remove words that appear only once in the set, and we do not perform any term weighting or stemming in the preprocessing. This word dictionary with the exception of words in each data set is then the final dictionary set used.

A.4 Additional Results

In this section, we present additional results on the video autoencoder models and the progressive decoupling ablation experiments.

A.4.1 Video Representation Learning

In Figure A.2, we present the latent spaces of the 3DConv-Base and UNetLDM video autoencoder models separated from each other as in the previous Section 4.2.1 we present their joint latent space.

In Figure A.3, we present the qualitative results for the video autoencoder models: 3DConv-Base, UNetLDM, and VDM (Ho et al., 2022). From the SyncDraw-MM set, we observed better quality with UNetLDM. The 3DConv-Base model generates correct results, but has more misleading cases and lacks sharpness in some cases. In the SyncDraw-MM set, for example, there are cases where digit 5 is misplaced with 3, or 9 with 4, and 1 with 7. This occurs at a lower level in the other models. The VDM model, on the other hand, is not consistent with its results, with its major drawback being the lack of filling in the digit (e.g., holes in some digits) and the thin look in most samples. This model also does not correctly generate the digits in a large part of the samples, generating instead frames with black background and random white points in the border without any digit enclosed. For the KTH set, UNetLDM also produces sharper videos compared to 3DConv-Base, which in some cases generates an artifact resembling an aura over the person. In this set, UNetLDM appears to produce a brighter background as well. VDM model produces sharper videos and also a large diversity in the samples, but following the results with the previous set, there is a large amount of poor samples generated where there is no movement or person in the video. Lastly, for the TACoS set, 3DConv-Base generated videos with less fine-grained details. For some cases, this seems to impact the understanding of the movement depicted in the video. The effect of "aura" also occurs in some samples of this set around people. The UNetLDM model generated more blur effects for TACoS and some artifacts resembling "checkboard" artifacts, mostly in brighter parts. The VDM model produced sharper videos for this set, and it was observed that the majority of the samples were generated with people in darker clothes.

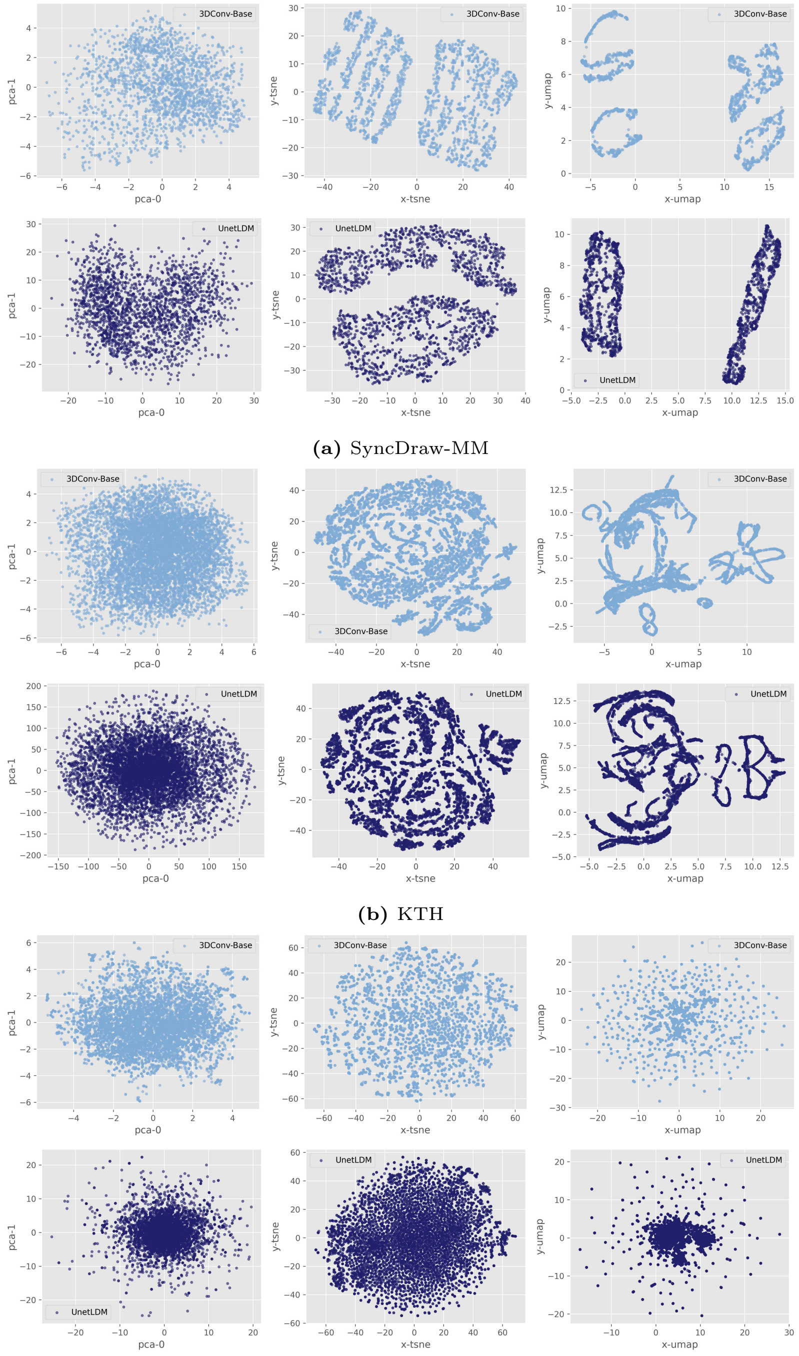
Ablation Experiments

Furthermore, we performed ablation experiments on video representation learning, where the quantitative results are presented in Table A.3, the qualitative result on Figure A.4 and the latent spaces on Figure A.5. We evaluated three main components using the 3DConv-Base architecture: the impact of the dimension size of video latent space; the autoencoder approach by comparing to a Variational Auto-Encoder (VAE); and the impact of the distribution discriminator D_z in the WAE-GAN-based approach.

From the quantitative results, we observe that regarding the dimension size, the variation of d_z does not seem to largely affect the quantitative results. However, switching the approach from a WAE-GAN-based to

[8] https://huggingface.co/docs/transformers/en/model_doc/clip#transformers.TFCLIPTextModel

[9] Downloaded from https://github.com/srhrshr/torchDatasets/. The data set splits ('train' and 'test') provided were the ones used in our experiments as well.



(a) SyncDraw-MM

(b) KTH

(c) TACoS

Figure A.2: Visualization of the video semantic spaces obtained with the 3DConv-Base (first rows) and UnetLDM (second rows) models for the Syncdraw-MM, KTH, and TACoS data sets.

a VAE harms the reconstruction. Removing the distribution discriminator D_z also harms the results, except for the VAE approach, which seems to generate better results without it.

Taking into account the qualitative results, we can observe an inferior reconstruction with the plain VAE approach where more than one digit appears to be reconstructed. Beyond this mirror effect, most videos seem to be concentrated on "up and down" movements rather than "left to right" or vice-versa videos. But this effect also seems to be partially associated with the distribution discriminator D_z, since by removing

Table A.3: Quantitative results of the video ablation experiments performed with SyncDraw-MM and the 3DConv-Base video architecture evaluating the impact of: dimension size of latent space d_z and the general AE adopted approach.

Metrics	PSNR↑	SSIM↑	LPIPS↓	DISTS↓	FVD↓	KVD↓
Dimension						
$d_z = 48$	19.2 ± 1.9	0.89 ± 0.03	0.083 ± 0.023	0.09 ± 0.02	2.81	0.0031
$d_z = 128$	19.0 ± 1.8	0.887 ± 0.030	0.086 ± 0.024	0.09 ± 0.02	2.77	0.003
$d_z = 256$	19.1 ± 1.9	0.889 ± 0.030	0.083 ± 0.023	0.09 ± 0.02	2.75	0.003
General AE approach						
Base w/o D_z	18.8 ± 1.8	0.885 ± 0.030	0.085 ± 0.024	0.09 ± 0.02	2.60	0.0028
VAE w/o D_z	17.3 ± 1.2	0.860 ± 0.020	0.112 ± 0.036	0.11 ± 0.02	3.34	0.0041
VAE	15.3 ± 1.2	0.757 ± 0.040	0.289 ± 0.066	0.16 ± 0.02	9.10	0.0165

it, the model presents better qualitative results for the same instances. Now, considering the removal of the distribution discriminator D_z from our main approach, we observe a small decrease in some full-reference metrics (e.g., PSNR) but with slightly better distribution-based ones (e.g., FVD). This is not particularly noticed in the qualitative results, as the videos are similar, showing for both approaches some cases of a lack of fine-grained details over the digits. Regarding the dimension size of the latent video distribution, we can observe small differences in the fine-grained details of the same instances for the models.

Moreover, considering the corresponding latent spaces of the video distribution illustrated in Figure A.5, we notice a slightly different structure obtained between the models. For the dimension variation, the latent spaces starting from $d_z = 128$ become sparser compared to $d_z < 128$ (also including $d_z = 64$ in Figure A.2. In contrast, VAE-based approaches obtain more concentrated latent spaces when we evaluate their distribution in the PCA visualization. Additionally, removing the distribution discriminator D_z changes the latent space structure when we compare with the reference on Figure A.2.

A.4.2 Progressive Decoupling Learning

In Figures A.6, A.7, and A.8, we present qualitative results for the text-to-video generation produced with the corresponding alignment models of Section 4.2.2 for: SyncDraw, KTH, and TACoS data sets.

From the SyncDraw results, we can observe a more difficult task for the alignment. All models seem to present poor video generation with a lack of fine-grained details for the digits. The worst results being the ones with the ImageBind-based and our method with UNetLDM alignments. They present higher indicators of representation collapse, where the former shows vertical movements even when the input text requires horizontal movements, and the latter generates almost the same exact video for different input texts. Overall, better results were observed for vertical movements compared to horizontal ones, although both are equivalently represented in the data set.

For the KTH set, the models present better results, which is possibly related to the decrease of the one-to-many difficult level of this set. We still observe a level of representational collapse for some input texts, but in a lower level than with the SyncDraw set. For CLIP-based and our method with 3DConv-Base, we observe an aura effect in some persons, which was also identified in the video autoencoder results, indicating a propagation of this effect. On the other hand, CoDi and our method with UNetLDM present less of this artifact. For the ImageBind-based alignment, frames with more blur than the other and an artifact resembling the generation of movement shadow in the legs part were noticed.

For the TACoS set, the results improve when compared with the KTH set, strongly indicating a correlation with the difficulty level of the alignment. The only exception being our method with UNetLDM video baseline, as this model, previously found to have the representational collapse problem in latent space, presents here the video generation indicator of this as well. In this set of results, an aura effect is also observed for models based on 3DConv-Base video AE. The best models being CLIP-based alignment and our method with 3DConv-Base. The aura effect seems more prominent in the ImageBind-based alignment,

Table A.4: Quantitative results of additional ablation experiments on the feature alignment between text and video modalities on the SyncDraw-MM data set with 3DConv-Base video autoencoder.

Model/Metrics	B-PSNR ↑	B-SSIM ↑	B-LPIPS ↓	B-DISTS ↓	FVD ↓	KVD ↓
Dimension $d_z = 256$	19.7 ± 1.9	0.864 ± 0.027	0.27 ± 0.10	0.14 ± 0.02	6.98	0.006

although other methods also use this video autoencoder and this model also present more blur than the others in the overall videos.

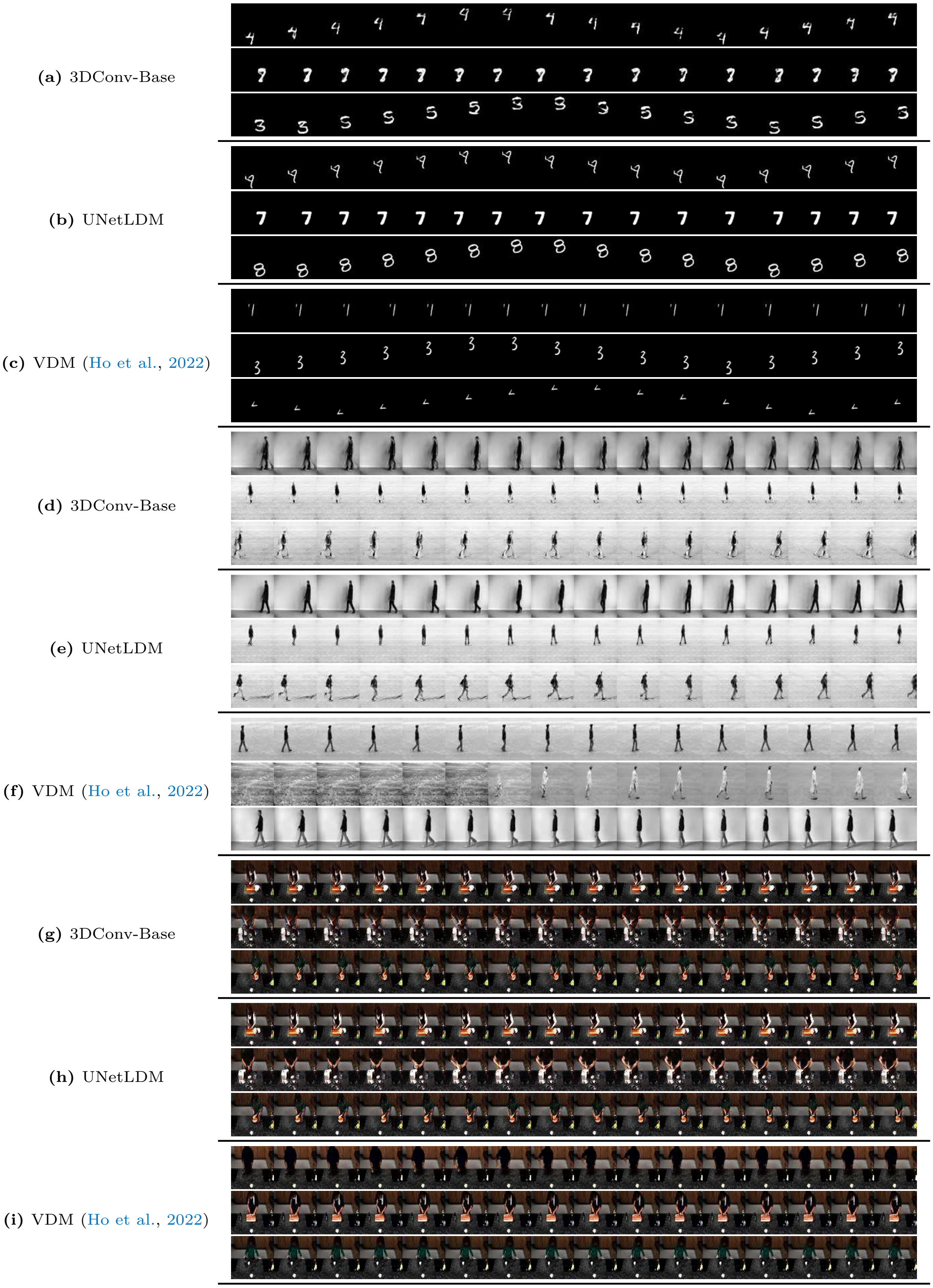
Ablation Experiments

In Figures A.9 and A.10, we present the latent spaces of the video and semantic shared representations obtained in the ablation experiments for the feature alignment approach. In Table A.4, we present the quantitative results of additional ablation experiments on the cross-modality alignment with the best architectures found in the ablation of video representation. Lastly, in Figure A.11, we present qualitative results for the ablation experiments on the alignment process.

The additional ablation shows the impact of representation learning in mapping between the modalities. We note that the structures of latent spaces change when we change the way the target modality is represented. Although it is primarily outstanding in semantic shared spaces, the structure is affected by producing sparser spaces (e.g. $d_z = 256$).

Regarding the qualitative results, we observed poor generation, indicating a poor alignment level for the SyncDraw data set, which is the most challenging one-to-many scenario. The non-progressive method trained with our adapted loss shows a higher indicator of representational collapse, since their generated videos seem to follow, with small differences, the same outlined video. The non-progressive version with the InfoNCE loss seems to suffer less with the representation collapse, although the video quality still lacks fine-grained details of the digits.

The VicReg seems to work better with particular digits, such as the digit 4. This is similar to BYOL results in this regard. But this solution in some videos seems to be generation details of two digits in the same scene, although in this scenario ground-truth data does not have it. Regarding the experiment with the video AE 3DConv-Base with $d_z = 256$, we observe better fine-grained details for some digits but far away to be considered correctly aligned. In general, one case noticed in the results was that the models seem to correctly follow the target motion, being better at the “moving up and down” category than the “left to right” or vice versa. Considering that the SyncDraw set is balanced in this regard, i.e., the number of videos with the “moving up and down” is close to the number of videos of “moving left to right” (or vice versa), this can show a more difficult alignment in the later large bucket of horizontal movement.



(a) 3DConv-Base

(b) UNetLDM

(c) VDM (Ho et al., 2022)

(d) 3DConv-Base

(e) UNetLDM

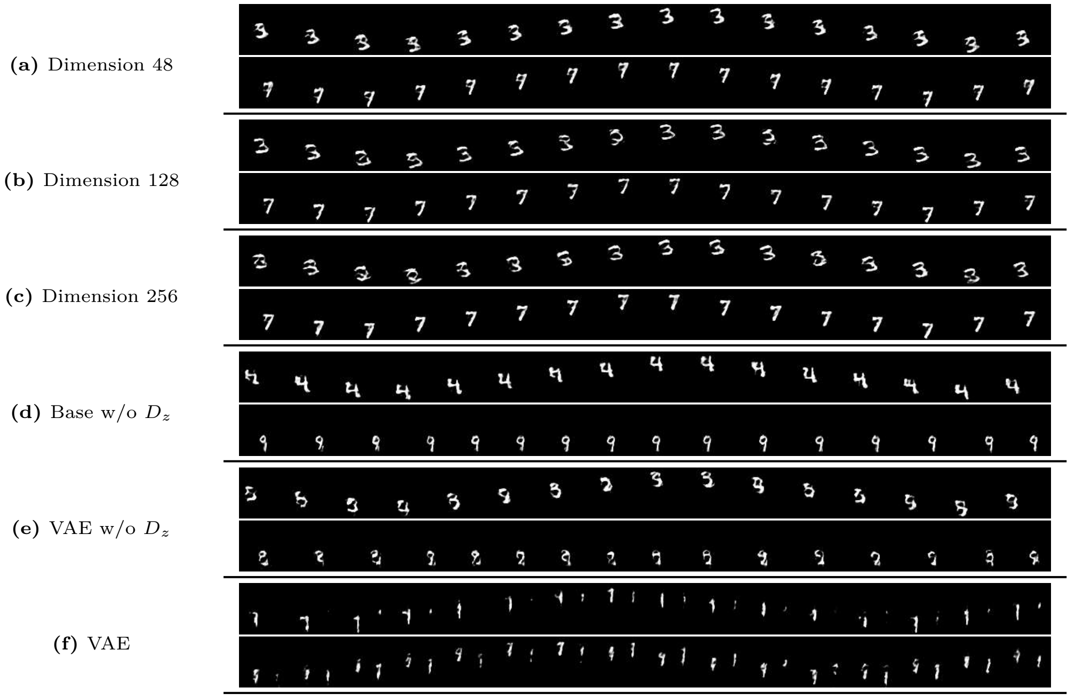
(f) VDM (Ho et al., 2022)

(g) 3DConv-Base

(h) UNetLDM

(i) VDM (Ho et al., 2022)

Figure A.3: Comparison of the generated videos by the video AE models on the SyncDraw-MM (a-c), KTH (d-f), and TACoS (g-i) data sets with 3DConv-Base, UNetLDM, and VDM (Ho et al., 2022).



(a) Dimension 48

(b) Dimension 128

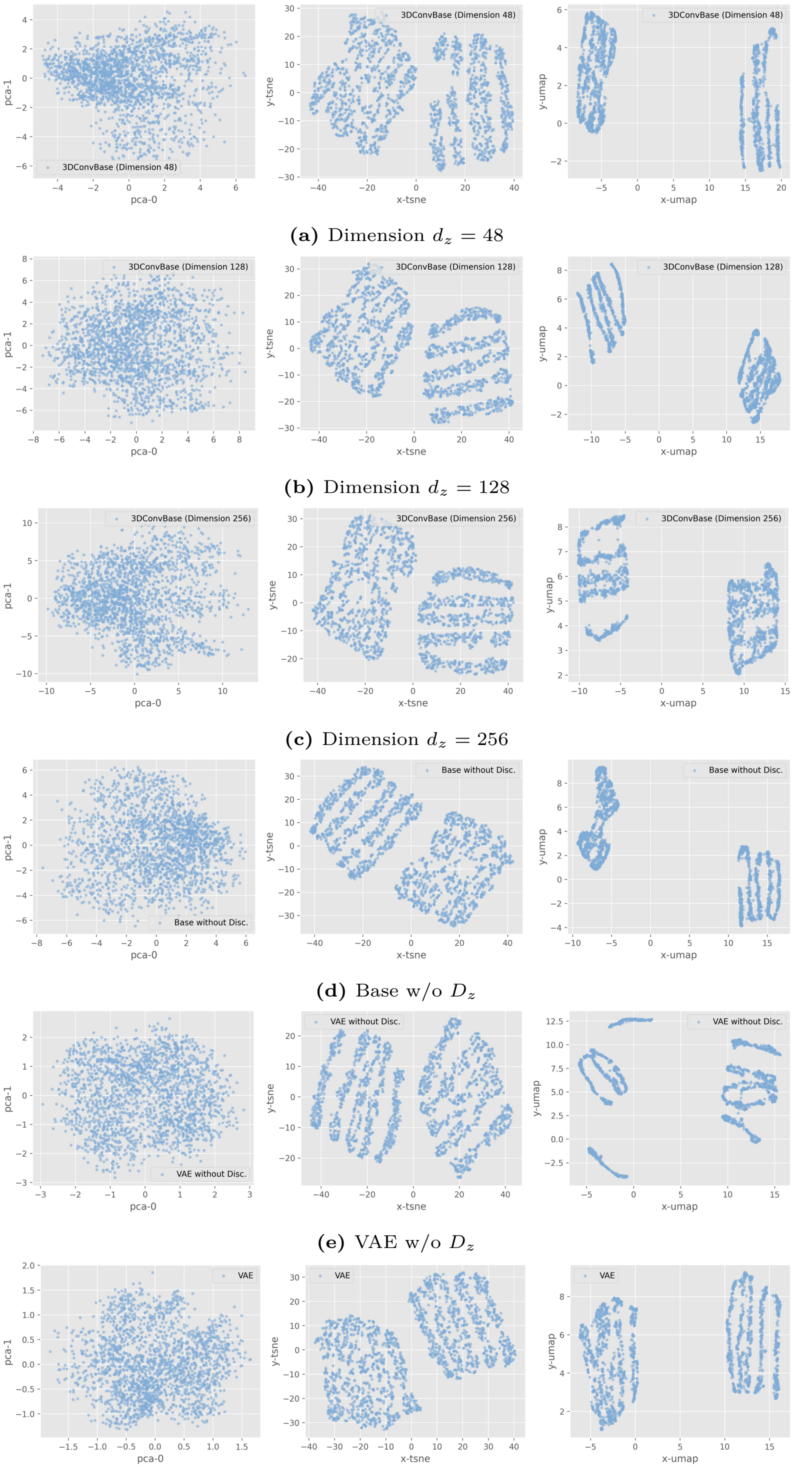
(c) Dimension 256

(d) Base w/o D_z

(e) VAE w/o D_z

(f) VAE

Figure A.4: Comparison of the generated videos by the video AE models from the ablation experiments on the SyncDraw-MM data set with 3DConv-Base architecture and variations on latent space dimension and AE approach.



(a) Dimension $d_z = 48$

(b) Dimension $d_z = 128$

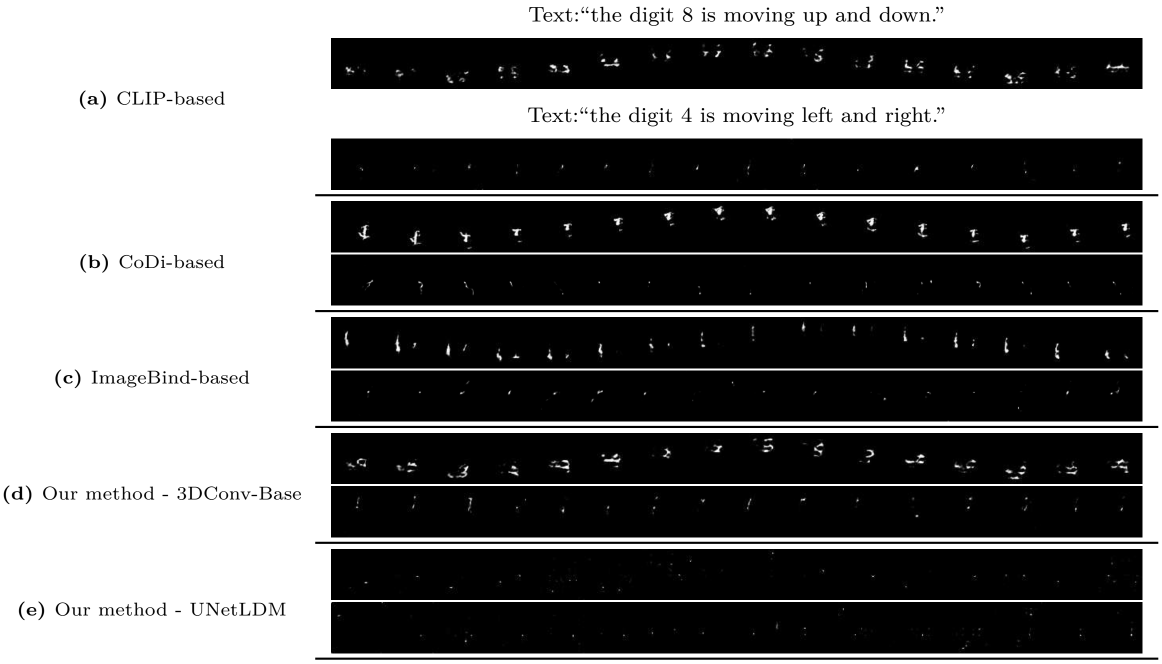
(c) Dimension $d_z = 256$

(d) Base w/o D_z

(e) VAE w/o D_z

(f) VAE

Figure A.5: Visualization of the video semantic spaces obtained with the 3DConv-Base for the video ablation experiments on: impact of dimension size of d_z (a-c), VAE approach, and impact of the distribution discriminator D_z on the produced video distribution.



Text:"the digit 8 is moving up and down."

Text:"the digit 4 is moving left and right."

(a) CLIP-based

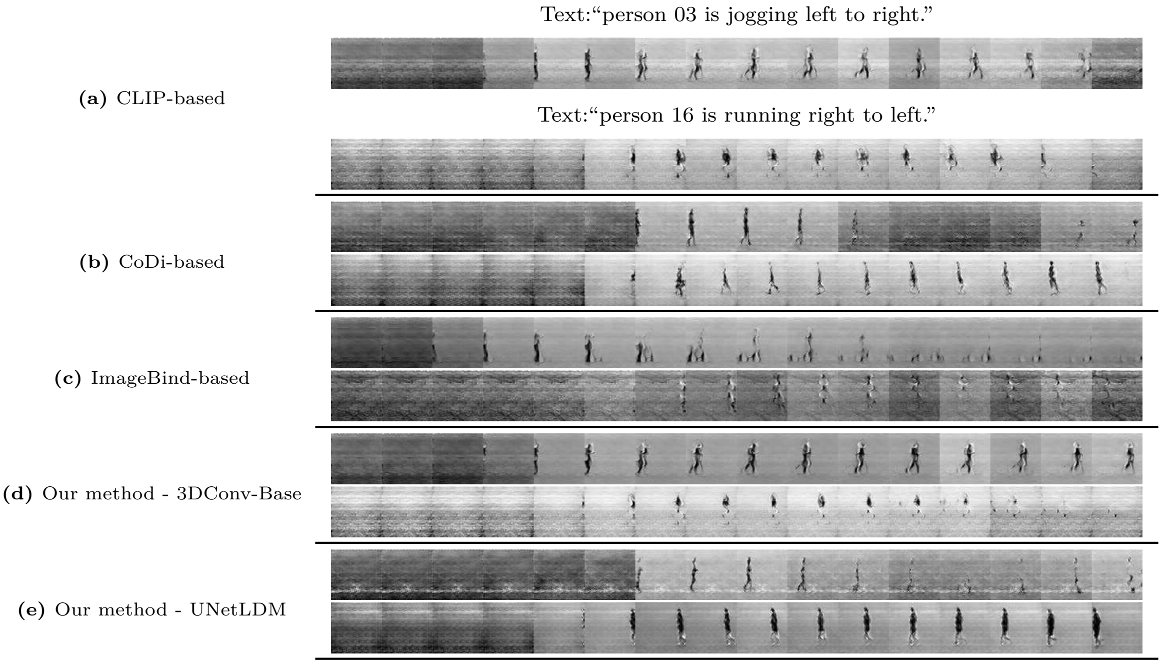
(b) CoDi-based

(c) ImageBind-based

(d) Our method - 3DConv-Base

(e) Our method - UNetLDM

Figure A.6: Comparison of the generated videos by the alignment models: CLIP-based, CoDi-based, ImageBind-based, Our method with both 3DConv-Base and UNetLDM video architectures, on the SyncDraw data set.



Text:"person 03 is jogging left to right."

Text:"person 16 is running right to left."

(a) CLIP-based

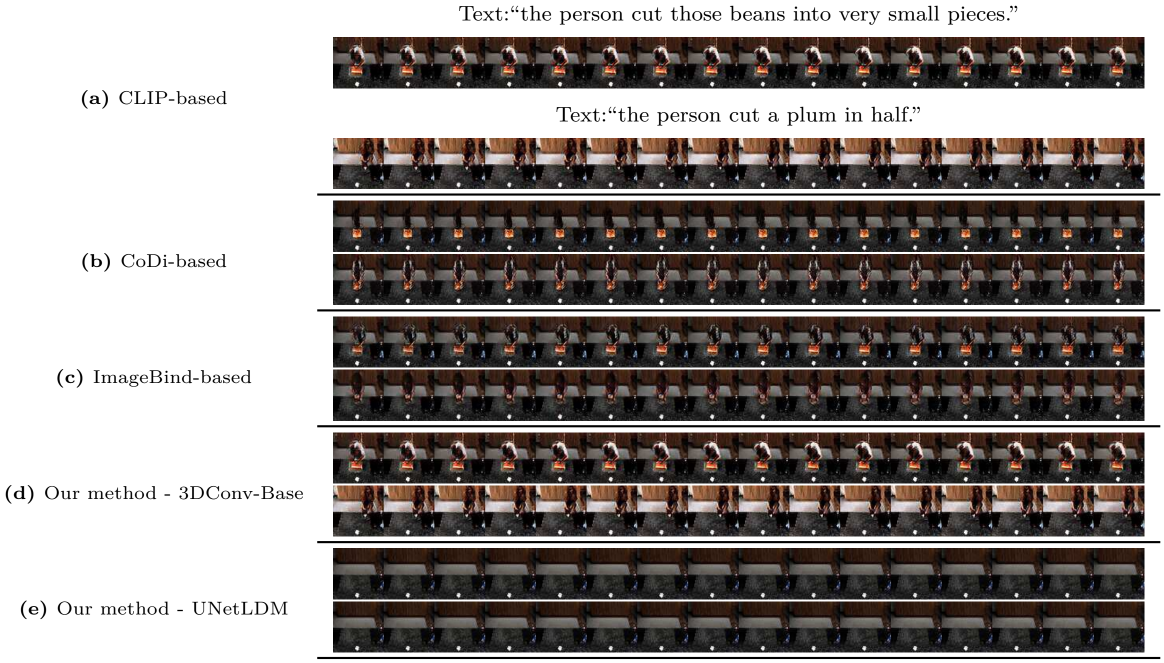
(b) CoDi-based

(c) ImageBind-based

(d) Our method - 3DConv-Base

(e) Our method - UNetLDM

Figure A.7: Comparison of the generated videos by the alignment models: CLIP-based, CoDi-based, ImageBind-based, Our method with both 3DConv-Base and UNetLDM video architectures, on the KTH data set.



Text:“the person cut those beans into very small pieces.”

Text:“the person cut a plum in half.”

(a) CLIP-based

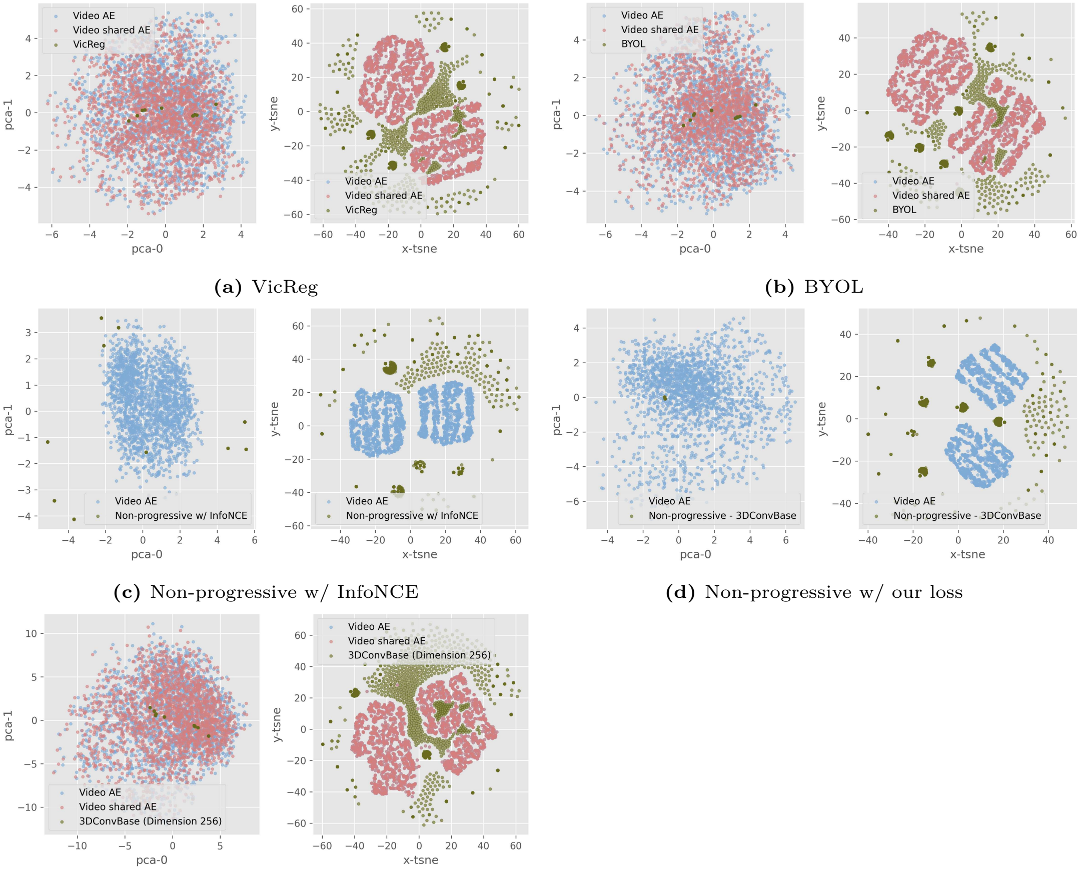
(b) CoDi-based

(c) ImageBind-based

(d) Our method - 3DConv-Base

(e) Our method - UNetLDM

Figure A.8: Comparison of the generated videos by the alignment models: CLIP-based, CoDi-based, ImageBind-based, Our method with both 3DConv-Base and UNetLDM video architectures, on the TACoS data set.



(a) VicReg

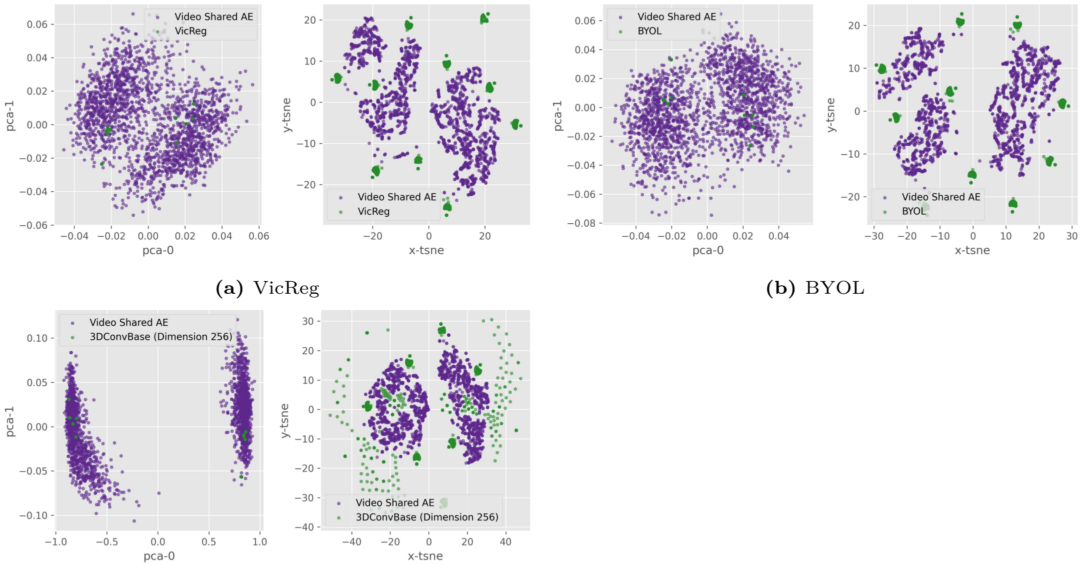
(b) BYOL

(c) Non-progressive w/ InfoNCE

(d) Non-progressive w/ our loss

(e) Dimension $d_z = 256$

Figure A.9: Visualization of the video latent space produced by the feature alignment methods on ablation section, which consists of the embeddings predicted from the corresponding mapping function and the ones obtained from the video AE used in each experiment using PCA (odd columns) and t-SNE (even columns) visualization.

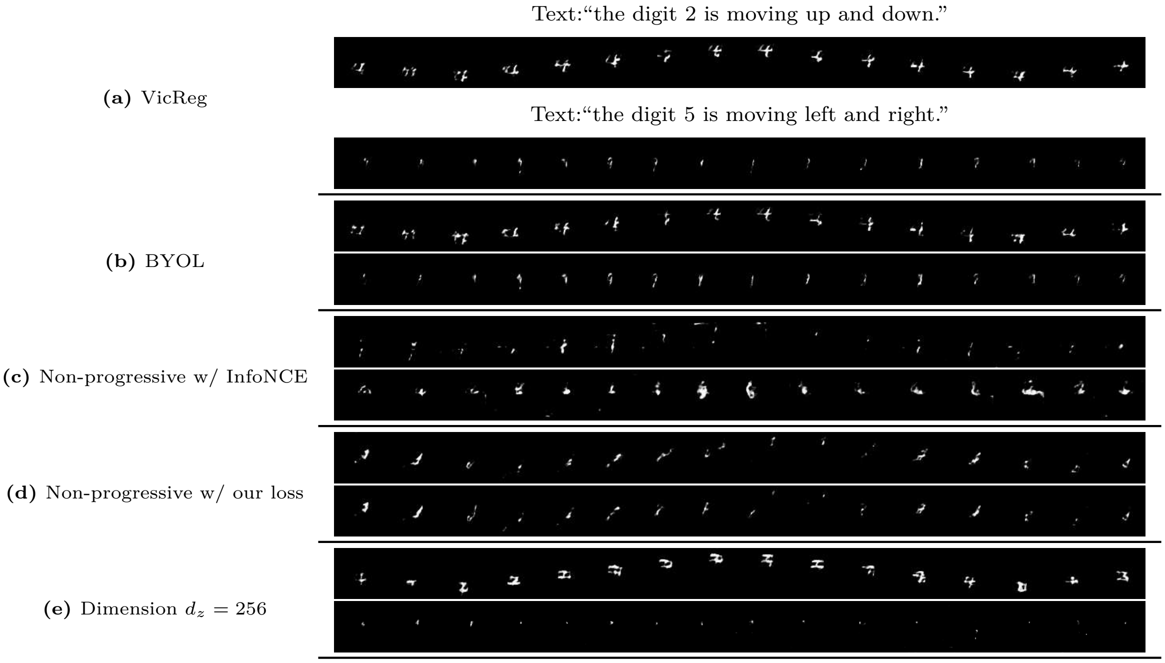


(a) VicReg

(b) BYOL

(c) Dimension $d_z = 256$

Figure A.10: Visualization of the semantic shared latent space produced by the feature alignment methods on ablation section, which consists of the embeddings predicted from the corresponding mapping function and the ones obtained from the video shared AE used in each experiment using PCA (odd columns) and t-SNE (even columns) visualization.



Text:"the digit 2 is moving up and down."

Text:"the digit 5 is moving left and right."

(a) VicReg

(b) BYOL

(c) Non-progressive w/ InfoNCE

(d) Non-progressive w/ our loss

(e) Dimension $d_z = 256$

Figure A.11: Comparison of the generated videos by the ablation alignment models: VicReg, BYOL, Non-progressive w/ InfoNCE, Non-progressive w/ our loss, and video AE baseline with dimension $d_z = 256$, considering the video 3DConv-Base architecture on the SyncDraw data set.