Adversarial learning methods for the generation of human interaction data

Louis Airale

▶ **To cite this version:**

Louis Airale. Adversarial learning methods for the generation of human interaction data. Robotics [cs.RO]. Université Grenoble Alpes [2020-..], 2023. English. NNT : 2023GRALM072 . tel-04612415

HAL Id: tel-04612415
https://theses.hal.science/tel-04612415

Submitted on 14 Jun 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ GRENOBLE ALPES

École doctorale : MSTII - Mathématiques, Sciences et technologies de l'information, Informatique
Spécialité : Mathématiques et Informatique
Unité de recherche : Centre de recherche Inria de l'Université Grenoble Alpes

Modèles adverses pour la génération de données d'interaction humaine

Adversarial learning methods for the generation of human interaction data

Présentée par :

Louis AIRALE

Direction de thèse :

Dominique VAUFREYDAZ MAITRE DE CONFERENCES HDR, UNIVERSITE GRENOBLE ALPES	Directeur de thèse
Xavier ALAMEDA-PINEDA CHARGE DE RECHERCHE HDR, INRIA CENTRE GRENOBLE-RHONE-ALPES	Co-directeur de thèse

Rapporteurs :

HAYLEY HUNG
ASSOCIATE PROFESSOR, DELFT UNIVERSITY OF TECHNOLOGY
RENAUD SEGUIER
PROFESSEUR, CENTRALESUPELEC RENNES

Thèse soutenue publiquement le **4 décembre 2023**, devant le jury composé de :

DOMINIQUE VAUFREYDAZ MAITRE DE CONFERENCES HDR, UNIVERSITE GRENOBLE ALPES	Directeur de thèse
HAYLEY HUNG ASSOCIATE PROFESSOR, DELFT UNIVERSITY OF TECHNOLOGY	Rapporteure
RENAUD SEGUIER PROFESSEUR, CENTRALESUPELEC RENNES	Rapporteur
DAMIEN TENEY SENIOR SCIENTIST, INSTITUT DE RECHERCHE IDIAP	Examinateur
LAURE SOULIER MAITRESSE DE CONFERENCES HDR, SORBONNE UNIVERSITE	Examinatrice
MASSIH-REZA AMINI PROFESSEUR DES UNIVERSITES, UNIVERSITE GRENOBLE ALPES	Président
GEORGES QUENOT DIRECTEUR DE RECHERCHE, CNRS DELEGATION ALPES	Examinateur

Invités :

XAVIER ALAMEDA PINEDA
CHARGE DE RECHERCHE HDR, INRIA CENTRE GRENOBLE-RHONE-ALPES

Abstract

The objective of this thesis work is to explore new deep generative model architectures for diverse human interaction data generation tasks. The applications for such systems are various: social robotics, animation or entertainment, but always pertain to building more natural interactive systems between humans and machines. Owing to their astonishing performances on a wide range of applications, deep generative models offer an ideal framework to address this task. In return, one can learn how to improve the training of such models by adjusting them to tackle the challenges and constraints posed by human interaction data generation. In this thesis, we consider three generation tasks, corresponding to as many target modalities or conditioning signals. Interactions are first modeled as sequences of discrete, high-level actions simultaneously achieved by a free number of participants. Then, we consider the continuous facial dynamics of a conversing individual and attempt to produce realistic animations from a single reference frame in the facial landmark domain. Finally, we address the task of co-speech talking face generation, where the aim is to correlate the output head and lips motion with an input speech signal. Interestingly, similar deep generative models based on autoregressive adversarial networks provide state-of-the-art results on these otherwise slightly related tasks. Training such models can however be long or unstable, in particular when the conditioning signal is weak (e.g. when only an initial state is provided). In light of this, we first devise an autoregressive generative adversarial network (GAN) for the generation of discrete interaction sequences, where we introduce a window-based discriminator network that accelerates the training and improves the output quality. We then scale this approach to the generation of continuous facial landmark coordinates, and exploit the inductive bias of autoregressive models for cumulative sums via residual predictions. In this unconditional setting, jointly generating and discriminating pairs of samples proved essential to allow long-term consistency and reduce mode collapse. In the third and last chapter, we introduce a multi-scale loss function and a multi-scale generator network to allow our autoregressive GAN to produce, for the first time, speech-correlated head and lips motion over multiple timescales. Experiments conducted on benchmark datasets featuring multiple interaction data modalities illustrate the efficiency of the proposed methods.

Résumé

L'objectif de cette thèse est d'explorer de nouvelles architectures de modèles génératifs profonds pour la génération de données d'interaction humaine. Les applications de tels modèles sont multiples, robots sociaux, animation ou encore divertissement, mais ont pour point commun de tendre à rendre plus naturelles les interactions entre l'humain et la machine. De par le réalisme de leurs résultats et leurs nombreuses applications, les modèles génératifs profonds offrent un cadre de travail idéal pour cette tâche. En retour, l'adaptation de ces modèles aux spécificités et aux contraintes liées aux données d'interaction humaine permet d'apprendre comment améliorer leur entraînement. Dans cette thèse sont considérées trois tâches de génération, pour autant de modalités de données ou de méthodes de conditionnement. Premièrement, les interactions sont traitées comme des séquences discrètes d'actions réalisées simultanément par un nombre indéterminé de participants. Puis on modélisera les dynamiques de la tête et des expressions du visage d'une personne en train de dialoguer à partir d'une seule pose initiale. On s'intéressera enfin à la génération des ces mêmes dynamiques à partir d'un signal audio de conditionnement, en veillant à synchroniser les mouvements de la tête et des lèvres avec le signal de parole. De manière remarquable, des modèles génératifs adverses autorégressifs assez proches obtiennent des performances de tout premier ordre sur ces tâches par ailleurs relativement hétérogènes. L'entraînement de ces modèles peut cependant se révéler instable, en particulier lorsque le signal de conditionnement est faible ou absent. La première contribution de cette thèse consiste donc en l'élaboration d'un modèle adverse génératif (GAN) autorégressif pour la génération d'interactions discrètes, assorti d'un discriminateur centré sur de courtes fenêtres temporelles permettant d'accélérer l'entraînement et d'améliorer la qualité des résultats. Cette approche est ensuite étendue à la génération continue de la dynamique du visage, pour laquelle est exploitée la capacité des modèles autorégressifs à représenter des sommes grâce à des connexions résiduelles. Pour cette tâche de génération sans conditionnement, générer et discriminer conjointement des paires d'échantillons s'avère essentiel pour fiabiliser les résultats sur de longues séquences et réduire le "mode collapse" lié aux GAN. Dans un troisième et dernier chapitre est proposée une approche multi-échelle à la fois dans l'objectif et l'architecture d'un modèle adverse autorégressif pour générer, pour la première fois, des mouvements de la tête et des lèvres corrélés avec le signal de parole à de multiples échelles temporelles. Des expériences conduites sur des jeux de données standards et pour différentes modalités d'interaction illustrent l'efficacité des méthodes proposées.

CONTENTS

CHAPTER 1

INTRODUCTION

1.1 GENERATIVE MODELS FOR HUMAN INTERACTION DATA GENERATION

This thesis deals with the challenging problem of generating human interaction data, for which we propose several innovative methods based on deep generative learning. It lies at the crossroads of two lines of research. One, the generation of human interaction data, is driven by applications: generating interaction data means empowering a system with the ability to understand social signals and respond accordingly, which is a prerequisite in many computer or robotic systems that need to fuse in human environments. The second line is a methodological one and relates to the recent and astonishing development of deep generative models on a variety of tasks, that however have not found a definitive answer to all problems arising in the context of human interactions.

Interactions between human beings may take a myriad of forms [119], and the focus will be put here on conversations in a broad definition that includes free-form [15] and more constrained scenarios (namely interviews and debates) [35, 24]. Even so, when it comes to computer science, data that represent such interactions can appear under multiple modalities. This includes head or body position and orientation represented as the coordinates of a center of mass and normal vectors, speaking turns or other discrete action cues constituents of conversations, but also raw speech or video signal. Likewise, the foreseen applications for the generation of interaction data directly stem from the modality: in social robotics, an important application that drove our initial research efforts and where a robot should navigate a human environment and interact with humans, inferring human actions to adjust the response accordingly can be as critical as anticipating people's trajectory. When it comes to generating continuous signals such as in co-speech motion generation, which is the focus of Chapter 5, applications will be found in animation, video editing, or other entertainment-oriented goals. Whatever the considered modality, interaction data generation still faces numerous challenges and this is where this thesis aims to contribute. It is noteworthy that in this thesis work, two situations will be encountered where the focus is put on either one or all of the interaction participants. Finally, beyond

its applications, interaction data generation can be a remedy to the data scarcity problem by augmenting existing data collections with synthetic data. This is especially important in a field where privacy and confidentiality concerns are real and the costs associated with the acquisition and labelling of new datasets may be significant.

Deep generative models offer an attractive framework to address the task of interaction data generation. Their strong performances and flexibility have long surpassed those of classical methods on as diverse use cases as image and video synthesis [94, 73], text generation [13] or sound generation [26], with compelling recent results quality. They have also been successfully employed on several tasks pertaining to interaction data generation in the definition given above [25, 108, 91]. These models are optimized to allow sampling from the original data distribution, which gives a practical way of producing diverse outputs from the same initial condition, e.g. an observed interaction sequence (Chapter 3) or an initial pose (Chapter 4). It is also straightforward to generate new samples thus counteracting data scarcity and fostering downstream applications (see for instance Chen *et al*. [23] for a review on the use of generative models for data augmentation in medical research).

Once the general objective of generating interaction data using deep generative models has been posed, several challenges arise regarding the actual implementation of these models, owing to the specificity of this task. Interactions when understood as conversations imply a coherent sequence of verbal and non-verbal signals from a possibly arbitrary number of participants, and the generative model must account for this complexity. At the time we started our research work, how to achieve this remained an open question. When we shift focus from multi-person action sequences (Chapter 3) to single-person talking head motion (Chapters 4 and 5), the temporal consistency requirement remains, but new challenges arise such as how to produce natural head motion or correlate output motion with the speech input. Therefore our main, transverse research problematic can be stated as follows: how to build generative models able to produce temporally coherent sequences of diverse interaction modalities? And, as a second objective, can these sequences be of arbitrarily long duration, possibly exceeding training sequence length, with minimal error

accumulation?

As we answer these questions, we learn how to improve the training of generative models, and in particular of Generative Adversarial Networks (GANs [36], see also 2.1), on a diversity of sequence generation tasks. To address both issues of temporal consistency and arbitrary sequence length, we devise our generative models as autoregressive functions and build a novel window-based multi-scale discriminator architecture able to enforce multi-scale realism and accelerate training (Chapter 3). Although initially only intended to deal with people participating in interactions, we find that generating and discriminating even unrelated sample pairs altogether improves the overall realism of output sequences and reduces mode collapse (Chapter 4). Finally, we can scale the previous findings to obtain sharp results of a duration of several times that of training sequences by a careful use of the conditioning signal in a co-speech facial landmark generation task (Chapter 5).

1.2 Thesis overview

Here we describe the structure of this manuscript, along with the different problems tackled in the following chapters.

In Chapter 3, we focus on how to embed the ability to process surrounding persons' actions in a deep generative model and introduce a GAN-based architecture and training process to do so. Interactions are represented as concomitant sequences of discrete actions sampled at high frequency (25 hz in our setting), and data come from the frame-by-frame annotated actions of the MatchNMingle dataset [15] (a raw image sample of which is provided in Figure 1.1 (a)). The main challenge is to enable the observation of each surrounding participant's social cues at all times during the generative process. This requires adequate pooling of these cues in a unified representation. A second challenge lies in the choice of the functional form of the sequential generative model, and of the loss function. We turn to an autoregressive formulation, where outputs are produced one at a time, and thus the generated sequence length is only limited by the point where error

(a)

(b)

(c)

Figure 1.1: Raw data samples from three datasets used in this thesis: (a) MatchNMingle [15]; (b) CONFER [35] and (c) VoxCeleb2 [24]. In practice we use the frame by frame discrete action annotations from MatchNMingle, and for CONFER and VoxCeleb2 we extract and work with facial landmark coordinates.

starts accumulating. As for the loss function, we use an adversarial loss to enhance the output diversity compared to maximum likelihood estimation methods. This implies outputting sequences autoregressively *during training*, which, in addition to being possibly inefficient, poses the issue of finding a differentiable surrogate to the sampling of discrete actions from a categorical distribution. We address the training inefficiency issue by introducing a window-based discriminator network and cast the problem as a continuous prediction problem to circumvent the differentiability issue (see Chapter 3 for details).

Chapter 4 explores scaling the approach devised in the previous chapter to higher dimensional spaces. In particular, the discrete interaction generation task becomes a single-person continuous head dynamics generation problem in the facial landmark domain, where the other interaction participants are no longer modeled but simply implied. Facial landmarks (see Figure 1.2) are a set of salient points, typically extracted using a pre-trained model from faces in RGB images. This is a useful representation to work with dynamics alone, as it discards visual information such as texture, color or lighting altogether. As for the choice of modeling single persons, the main reason behind this is the scarcity of adequate multi-person audio-visual datasets. The few that exist are of relatively small scale and do not come with separate speech signals for the different speakers, which makes the problem of multi-person talking head generation particularly challenging (see Figure 1.1 (b) for illustrative samples from one such dataset). A possible alternative is to produce head motion for several interacting people *without conditioning signal*, yet one still faces the difficulty of evaluating the degree of interaction of the output sequences. Owing to the existing applications and open research questions associated with this task, we focus in that chapter on single-person unconditional head motion generation. We show that a continuous-domain extrapolation from the previous approach is well-suited to produce long sequences of natural head dynamics, which is an often overlooked issue in the related literature. Interestingly, several components originally designed to handle interactions provide benefits in a single-person setting, as the joint generation and discrimination of sample pairs which at the same time improves result quality and mitigates the well-known mode collapse problem of GANs.

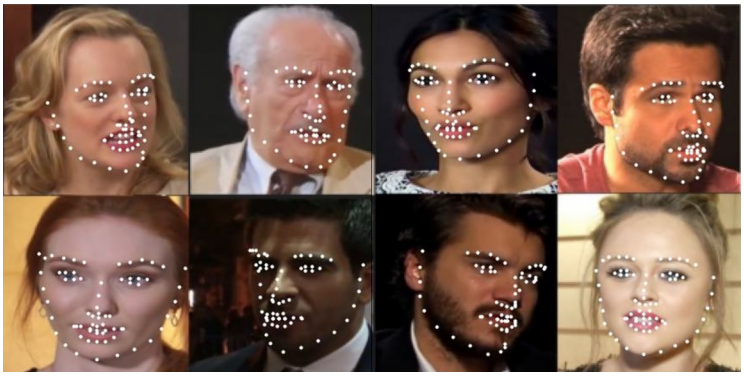


Figure 1.2: Facial landmarks, extracted using FaceAlignment [14], are at the core of several of this thesis contributions.

In Chapter 5, we investigate speech-conditioned head motion generation, taking the autoregressive GAN introduced in Chapter 4 as a baseline model. This task, also known as talking head generation, is well studied but current treatments fail to account for natural head motion, focusing instead on lip synchronization and photorealistic rendering at the cost of static or unrealistic head dynamics. We show however in this chapter that our autoregressive model produces smooth facial landmark dynamics, and devise an original way to correlate these dynamics with the input speech signal over multiple time scales. We thus introduce a novel multi-scale audio-visual contrastive loss function evaluated on multimodal input pyramids, along with a multi-scale generative neural network able to produce complex audio-synchronized dynamics.

CHAPTER 2

TRANSVERSE LITERATURE REVIEW

This chapter reviews transverse topics regarding our research work, such as generative models or talking head generation. More detailed and contextual literature reviews can be found in the relevant sections of the following chapters.

2.1 Deep generative models

Given a training set of observations $\mathcal{X}$, generative models aim to approximate the underlying distribution $p(x)$ to produce new samples, contrary to discriminative models where one needs to estimate $p(y|x)$ given a new observation x, where the random variable y can be for instance the object class of x. A common approach consists in maximizing the log-likelihood $\log p_\theta(\mathcal{X})$, where the parameters θ of the approximate distribution p_θ are computed by a neural network. This objective typically boils down to minimizing a mean squared error loss when $\mathcal{X}$ spans a continuous domain and a Gaussian form is assumed for p, or to a cross-entropy loss when the x is a discrete variable and p is a categorical distribution. Such generative models comprise autoregressive models that can be used for image or speech generation [113, 84, 21], natural language processing [29, 13], and play an important role in many human-related sequence generation tasks, such as talking head generation [141, 38, 32] or human motion prediction [77, 75]. Latent variable models are another class of generative models that rely on latent variables to model complex data distributions, and that are trained using a lower bound of the expected log-likelihood. Variational autoencoders (VAE) [58, 45] and more recently VQ-VAEs [114], where the latent variables are sampled from a finite codebook, have been used as an alternative to GANs thanks to their ability to exploit the full data distribution [71, 92, 31, 101]. Diffusion models [102, 46, 94] have become the de-facto standard for image generation and beyond, with promising applications in talking head generation [30, 104]. Their much longer inference time however still limits their applications. In this thesis, we focus on GANs that are especially useful for autoregressive continuous data generation and provide a good trade-off between output quality and inference time.

2.2 GENERATIVE ADVERSARIAL NETWORKS

GANs have been one of the dominant recent deep generative models. Introduced by Goodfellow *et al*. [36], these models comprise a generator network $\mathcal{G}$ and a discriminator network $\mathcal{D}$ that are jointly trained with the following minimax objective:

$$\min_{\mathcal{G}} \max_{\mathcal{D}} \quad \mathbb{E}_{x \sim p_{data}} \log(\mathcal{D}(x)) + \mathbb{E}_{z \sim p_z} \log(1 - \mathcal{D}(\mathcal{G}(z)) \tag{2.1}$$

where p_{data} is the training dataset and p_z is a noise distribution. $\mathcal{D}$ is a two-class classifier network providing the probability that a sample comes from the original data, while $\mathcal{G}$ is trained to produce data that will be classified as actual samples from the discriminator. It was shown that optimizing this objective amounts to minimizing the Jensen-Shannon divergence between the ground truth and generated data distributions [36]. Several variations of the original objective were further proposed, mainly attempting to overcome the training instabilities often encountered with the original GAN formulation [5, 39, 76, 66]. GANs have shown formidable performances on a wide range of unconditional [11, 56, 12] and conditional generation tasks [83, 51, 125] and for representation learning [89, 133]. Closer to the topic of this thesis, GANs have played an important role in a variety of continuous sequence generation problems [69, 8, 62, 82] for their ability to produce more diverse outputs than maximum likelihood estimation methods. Although a less common use case, they have also been used to generate discrete sequential data (see Section 3.2.2 for an in-depth review). In this thesis, we use GANs on various social interaction generation tasks, both in discrete and continuous domains.

2.3 SOCIAL CUES PREDICTION IN HUMAN INTERACTION

Leveraging social cues from an interaction situation to estimate or predict human behavior is the focus of several works, both anterior and concurrent to this thesis. Joo *et al*. [53] and Tan *et al*. [108] propose discriminative models which exploit the state of surrounding persons to estimate respectively the speaking status, body motion and orien-

tation, and the head orientation of an individual in an interaction. Greenwood *et al*. [37] and Raman *et al*. [91] explore the prediction of the same social cues within a sequence generation framework, while Sangvhi *et al*. [97] use an imitation learning approach to predict conversational action sequences from position, gaze direction and action history of multiple persons. Finally, there is an extensive bibliography on pedestrian trajectory prediction which relies on similar mechanisms to process contextual cues coming from surrounding moving agents and fixed obstacles to adjust a target agent's behavior [2, 42] (see also Section 3.2.1 for a thorough review on the subject).

2.4 TALKING HEAD GENERATION

The task of animating a human face with a neural network can be either guided when the head motion comes from a driving sequence or unguided, in which case the head and lip motion must be inferred by the generative model from a speech input signal. Compelling results have been achieved over the years to improve the photorealistic rendering of guided methods [132, 99, 41, 93]. Among these, several works rely on low dimensional representations, e.g. facial landmarks [137, 79, 131], learned keypoints [99, 126], or morphable models [136] to handle the dynamics, which are later used to warp or normalize the style of the source identity image.

On the other hand, the primary focus of audio-driven talking head synthesis has been on syncing output lip movements and input speech signal, either leaving visual reenactment as a separate task or limiting it to static pose scenarios [109, 55, 105, 38, 103, 142, 139, 121, 32]. For this reason, there have been comparatively few endeavors to generate realistic head motion [18, 141]. As a noticeable improvement over previous research, recent works showed very promising results producing rich head motion in a low-dimensional keypoint space in combination with proficient visual reenactment systems [123, 124]. However, there remains a margin for improvement in particular in the diversity of output head motion and in the time alignment between speech, lips, and head motion over different time scales, which has never been addressed before.

2.5 Language models for interaction generation

Large language models (LLM) [29, 90, 13, 111] occupy an ever-increasing place in the artificial intelligence literature, owing to their success in exploiting huge textual corpora and scale beyond natural language processing. This has been made possible via fine-tuning [48, 49] and prompt-tuning methods [64], that allow to leverage the power of pre-trained language models on various sequence generation tasks [54, 135]. Because of the novelty of the subject, the use of LLMs for the tasks addressed in this thesis, especially that of discrete action generation, is yet to be explored, but will definitely impact greatly the literature.

CHAPTER 3

MULTI-PERSON INTERACTION SEQUENCE GENERATION

Prediction of human actions in social interactions has important applications in the design of social robots or artificial avatars. In this chapter, we focus on a unimodal representation of interactions and propose a novel data-driven approach for interaction generation. In particular, we model human interaction generation as a discrete multi-sequence generation problem and present an adversarial architecture for conditional interaction generation. This model builds on a recurrent encoder-decoder generator network and a dual-stream discriminator, that jointly evaluates the realism of interactions and individual action sequences and operates at different time scales. Crucially, contextual information on interacting participants is shared among agents and reinjected in both the generation and the discriminator evaluation processes. Experiments show that albeit dealing with low dimensional data, our approach succeeds in producing high realism action sequences of interacting people, comparing favorably to a diversity of recurrent and convolutional discriminator baselines, and we hypothesize that it will constitute a first stone towards higher dimensional and multimodal interaction generation. Evaluations are conducted by adapting classical GAN metrics to discrete sequential data. The proposed model is shown to properly learn the dynamics of interaction sequences, while exploiting the full range of available actions.

3.1 INTRODUCTION

Interactions between humans are the basis of social relationships, incorporating a large number of implicit and explicit multimodal signals expressed by the interaction partners [119]. As the number of interacting people increases, so does the complexity of the underlying interpersonal synchrony patterns. For humans, the interaction capability is at the same time innate and acquired through thousands of interaction experiences. For interactive systems, predicting and generating human actions within social interactions is still far from the human-level performance. However, this task is central when it comes to devising, for instance, artificial avatars displaying realistic behavior or a social robot able to anticipate human intentions and adjust its response accordingly. One possible explanation lies in the difficulty to collect and annotate human social behavioral data, illustrated

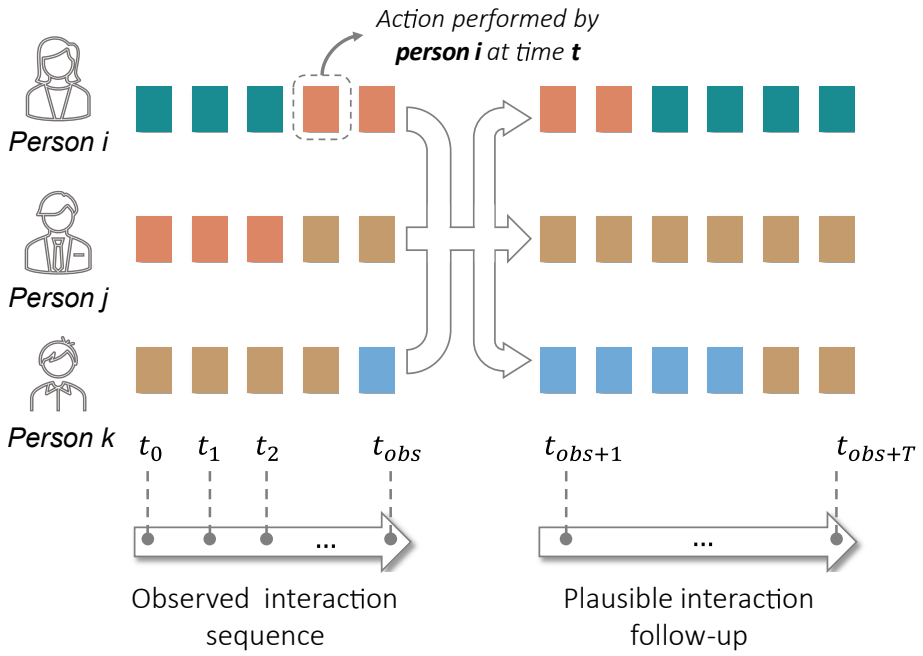


Figure 3.1: Illustration of the task of interaction generation, where an agent performs one action at each time step. Typically, the sampling frequency is chosen such that an action occurs over several time steps. Given an observed interaction sequence, a model should be able to generate realistic follow-ups for every participant.

by the scarcity of corpora available to the community. A second source of difficulty is the intrinsically multimodal nature of human interactions. Ideally, when generating interaction data, a system should deal with social signals as diverse as gaze, pose, facial expression or gestures. In this chapter, we choose to relieve much of this complexity and treat interactions as synchronized sequences of high level discrete actions, and propose to solve the task of generating a realistic continuation to an observed interaction sequence (Figure 3.1). The chosen representation has the advantage of easing the generation task while carrying enough meaningful information to study diverse interpersonal synchrony patterns. This approach has several other benefits. First, it is reasonable to think that parts of the devised architecture will generalize to other modalities such as, for instance, the mechanisms of integration of surrounding social cues that are not particularly linked to the considered data modality. Second, the generated interaction sequences will add up to the original dataset, representing as many fresh, synthetic samples. Provided a dataset that associates these simple actions with richer representations is available, one may use them

as conditioning data for the generation of more complex interaction sequences. Finally, representing the instantaneous state of a person in an interaction as a discrete action is consistent with the annotations provided in current human interaction datasets [15, 3].

The generation of discrete social action sequences of several people considering the interpersonal synchrony shares many ties with text generation in NLP and trajectory prediction in crowded environments. As in text generation, we seek to produce sequences of discrete data, or tokens, although the size of the token dictionary will typically be orders of magnitude smaller than that of any language. Discrete sequence generation in an adversarial setting comes with well identified limitations: non-differentiability that prevents backpropagating gradients from the discriminator, and lack of information on the underlying structure of the sequence [40], due to the too restrictive nature of the binary signal of the discriminator. To overcome the first issue, the generator can be thought of as a stochastic policy trained with policy gradient [130]. However, as noted in Guo *et al*. [40], sequences produced via vanilla policy gradient methods see their quality decrease as their length grows, and therefore do not usually exceed 20 tokens. Another drawback of these methods is that they rely on several rounds of Monte Carlo search to estimate a policy reward, noticeably slowing down training.

As recent works in trajectory prediction exploit path information from surrounding persons to predict pedestrian trajectories that avoid collisions [2, 42, 63], one can think of interaction generation as the prediction of trajectories in a discrete action space where people constantly adapt their behaviour to other persons' reactions. Because the action space is discrete, action generation is yet better modelled by sampling from a categorical distribution, which requires substantial changes from the previous works. Moreover, and very differently from trajectories that are sequences of smoothly-changing continuous positions, discrete sequences of social actions cannot be considered as smooth. Accounting for other persons' behavior in social interactions therefore requires constant re-evaluations, because one must consider the recent actions performed by all the individuals within the same social interaction. On the other hand, it should be possible to exploit the relative steadiness of action sequences: for a sampling rate of 25 fps, an action will typi-

cally extend for several tens of time steps. This fine grained resolution also allows to deal with actions of different duration.

Last, this research faces a new challenge regarding the proper evaluation of the generated sequences of multi-person interactions. The Inception Score (IS) and the Fréchet Inception Distance (FID), two metrics commonly associated with adversarial generation, are primarily intended to assess image quality [96, 44]. The IS consists of two entropy calculations based on how generated images span over object classes, and it is relatively straightforward to adapt it to action sequences. The FID, on the other hand, requires the use of a third party model trained on an independent image classification task. The absence of such an off-the-shelf inception model for discrete sequential data thus requires to devise a new independent task and to train the associated model that will serve to compute the FID.

In this chapter, we present a conditional adversarial network for the generation of discrete action sequences of human interactions, able to produce high quality sequences of extended length. We follow Alahi *et al*. [2] and Gupta *et al*. [42] and perform integration of contextual cues by means of a pooling module. We use a discriminator with two distinct streams, one guiding the network into producing realistic action sequences, the other operating at the interaction level, assessing the realism of the participants' actions relative to one another. Noticeably, we build our model on a classical GAN framework as it demonstrated promising performances without the need of policy gradient. We propose however an essential window-based multi-scale (also referred as "local") projection discriminator inspired from Miyato & Koyama [81] and Isola *et al*. [51] to provide the generator with localized assessments of sequence realism, therefore allowing the signal from the discriminator to be more informative. The result is an adversarially trained network able to predict plausible future action sequences for a group of interacting people, conditioned on the observed beginning of the interaction. Finally, we introduce two new metrics based on the principles of the IS and FID so as to assess the quality and the diversity of generated sequences. In particular, we propose a general procedure to train an independent inception model on discrete sequences, whose intermediate activations can

be used to compute a sequential FID (or SFID).

The contributions of the work presented in this chapter are:

- A general framework for the generation of multi-person discrete interaction sequences;
- A novel dual-stream local recurrent discriminator architecture, that allows to efficiently assess the realism of the generated interaction sequences;
- Two variants of the popular Inception Score and Fréchet Inception Distance metrics, suited to assess the quality of discrete sequences.

The rest of the chapter is structured as follows. First, we review related works in trajectory prediction and text generation in section 3.2. We then describe our architecture in section 3.3. In section 3.4, we introduce novel variants of the Inception Score [96] and Fréchet Inception Distance [44], widely used to assess the visual quality of GAN outputs, suited to our discrete sequence generation task. Experiments conducted on the MatchNMingle dataset [15] show the superiority of our dual-stream local discriminator architecture over a variety of baseline models in section 3.5. Finally, we conclude in section 3.6.

3.2 Related work

Following the above discussion we review related papers on trajectory prediction and generation of discrete sequential data that both share ties with discrete interaction data generation.

3.2.1 Trajectory prediction

Most recent works in trajectory prediction make use of contextual information of surrounding moving agents or static scene elements to infer a trajectory [2, 33, 9, 42, 63, 138, 95, 115]. In Lee *et al.* [63], a conditional VAE is used to generate a set of possible

trajectories, which are then iteratively refined by the adjunction of cues from the scene and surrounding pedestrians. Sadeghian *et al*. [95] propose a GAN and two attention mechanisms to select the most relevant information among spatial and human contextual cues for the considered agent path. As in previous works, Zhao *et al*. [138] use an encoder-decoder architecture, but propose to jointly exploit pedestrian and static scene context vectors using a convolutional network. The resulting vectors are then spatially added to the output of the encoder before the decoding process. In the preceding works, pooling strategies are usually employed to enforce invariance to the number of participants in the scene. Such strategies include local sum pooling [2], average pooling [63], or more sophisticated procedures involving non-linear transformations [42]. In our setting, we posit an equal prior influence of all interacting individuals on the decision to perform an action and leave the exploration of different conditions to future work. This alleviates the need for a pooling strategy aware of the spatial position. However, it is important that the contextual information provided to the decoder should follow the course of the interaction, and thus be recomputed at every time step. This is similar to Fernando *et al*. [33] and Sadeghian *et al*. [95] who rely on a time-varying attention context vector.

In most previous works the models are either trained to minimize the mean square distance from a ground truth trajectory or to maximize the realism of single-person trajectories thanks to an adversarial loss, but few consider trajectories interplay in the training objective. This is the case of SocialGAN [42], where the discriminator outputs a single score for the entire scene, while the adequacy of individual trajectories is ensured by an L_2 reconstruction loss and by the prediction of residual displacements over time steps. On the contrary, we would like our adversarial loss to cover both the realism of individual sequences and of the interaction as a whole, and we achieve this by using a discriminator with two streams. This way we are able to investigate the effects of lessening the weight of the L_2 loss or to simply train the model without it. Note that we could follow a similar strategy to Zhao *et al*. [138] and sum individual hidden states with contextual information into a single vector serving as input to the discriminator classifier. We want however to avoid any possible leakage of information between interaction and individual sequence

evaluations and therefore stick to two separate networks.

3.2.2 Generation of discrete sequential data

GANs have recently been proposed to complement the traditional Maximum Likelihood Estimation training for text generation to mitigate the so-called exposure bias issue, with pretty good results on short word sequences [65, 130, 67]. The use of an adversarial loss for a discrete sequence generation network yet comes with two well-identified caveats. First, workarounds must be found to allow the architecture to be fully differentiable. Second, the binary signal from the discriminator is scarce and may lack information as to what makes a sentence realistic or not [40]. Yu *et al*. [130] and Li *et al*. [65] interpret text generation as a Markov Decision Process where an agent selects a word at each state, constituted of all previously generated words. The generator is trained via stochastic policy gradient descent, while the reward is provided by the discriminator. These ideas are extended in Lin *et al*. [67], where the discriminator is modeled as a ranker rather than a binary classifier to avoid vanishing gradient issues. In Guo *et al*. [40], the generator is implemented as a hierarchical agent, following the architecture proposed by Vezhnevets *et al*. [117], so as to produce text sequences of increased length. Action sequences however differ from text by the limited size of action space compared to a language dictionary. We also hypothesize a shorter "memory" of action sequences: it is likely that one can judge the overall quality of an action sequence by looking only at small chunks of it, provided their length is chosen adequately. Therefore we propose a window-based discriminator operating on short-range receptive fields to allow its signal to be more informative, which can be seen as a recurrent equivalent of PatchGAN [51]. Plus, we found that using a classical adversarial loss was sufficient to achieve high-quality results, which relieved us of the burden of keeping an estimate of the expected reward.

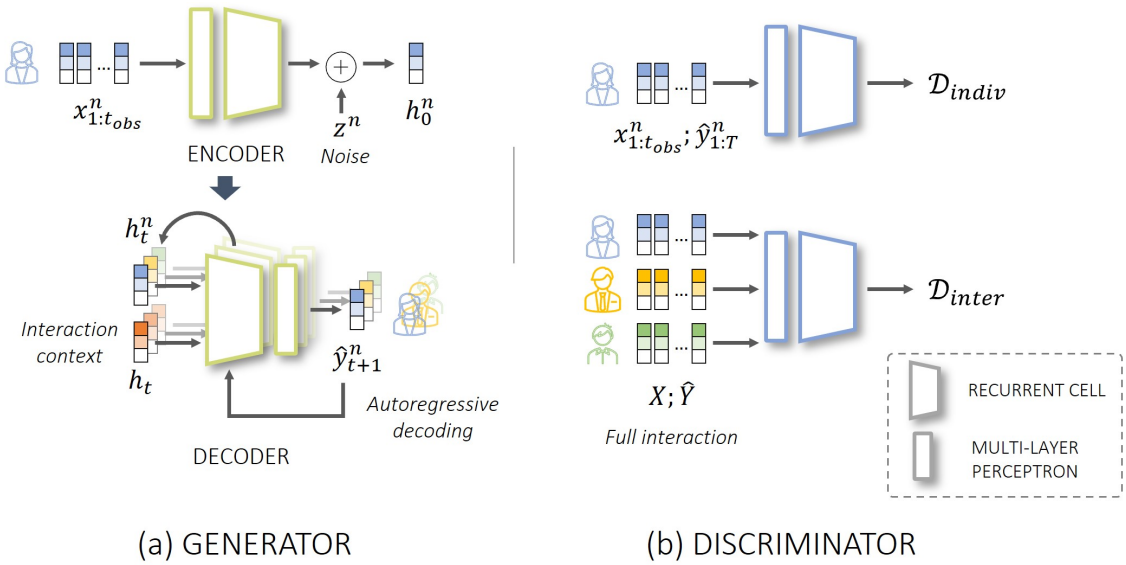


Figure 3.2: Architecture of SocialInteractionGAN. Our model is composed of (a) an encoder-decoder generator and (b) a recurrent dual-stream discriminator that assesses sequences individually and interaction as a whole. Hidden states from all participants in the interaction are pooled and re-injected at each step in the decoder.

3.3 MULTI-PERSON INTERACTION SEQUENCE GENERATION WITH SOCIALINTERACTIONGAN

This section presents the SocialInteractionGAN model, a conditional GAN for multi-person interaction sequence generation. The model takes as input an interaction $\mathbf{X}$ of duration t_{obs}, which is constituted of N synchronized action sequences, one for each participant. We denote the observed action sequence of person n as $\mathbf{x}^n_{1:t_{obs}} = \{x^n_t, 1 \leq t \leq t_{obs}\}$ (in practice we will drop time index to simplify notations and only use $\mathbf{x}^n$ for person n input as it is always taken between $t = 1$ and $t = t_{obs}$). Our goal is to predict socially plausible future actions for all interacting agents. We write the newly generated interaction as $\hat{\mathbf{Y}} = (\hat{\mathbf{y}}^1_{1:T}, \ldots, \hat{\mathbf{y}}^N_{1:T})$, where similarly $\hat{\mathbf{y}}^n_{1:T} = \{\hat{y}^n_t, t_{obs}+1 \leq t \leq t_{obs}+T\}$ is the generated action sequence for person n. Again $\hat{\mathbf{y}}^n$ will be employed when referring to the whole generated sequence. Using similar notations, the ground truth sequence is denoted $\mathbf{Y}$.

3.3.1 SOCIALINTERACTIONGAN ARCHITECTURE OVERVIEW

Our model is illustrated in Figure 3.2. It is composed of a recurrent encoder-decoder generator network (see section 3.3.2) and a dual-stream local recurrent discriminator (see section 3.3.3). The generator encodes the observed part of the interaction and sequentially generates future actions for all participants. We use a pooling module to merge all participants' hidden states into a single vector that is fed to the decoder at each time step. This way, the decoded actions depend on the previous actions of all surrounding persons. The pooling operation is invariant to the number of interacting people, and so is our model that can therefore virtually handle interacting groups of any size. The generated interaction $\hat{\mathbf{Y}}$ is concatenated to the conditioning tensor $\mathbf{X}$ and input to the discriminator, alternatively with the real interaction $[\mathbf{X}; \mathbf{Y}]$. The dual-stream discriminator then assesses the realism of both individual action sequences and interaction as a whole. The following sections detail the various modules that compose the overall architecture.

3.3.2 GENERATOR

Encoder Network The encoder is based on a recurrent network that operates on a dense embedding space of the discrete actions. The input sequences are processed by a LSTM recurrent encoder f_e [47], independently for each person n:

$$c^n = f_e(\mathbf{x}^n). \tag{3.1}$$

A random noise vector z^n of the same dimension as c^n is then added to the encoding of the observed sequence. This is to allow the GAN model to generate diverse output sequences, as the problem of future action generation is by essence non-deterministic:

$$h_0^n = c^n + z^n. \tag{3.2}$$

The resulting vector h_0^n is then used to initialize the decoder's hidden state.

Decoder Network The decoder is also a recurrent network that accounts for the actions of all persons involved in the interaction thanks to a pooling module. This pooling module, which is essentially a max operator, is responsible for injecting contextual cues in the decoding process, so that the generator selects the next action based on the interaction history. Such a pooling is essential to let the number of interacting people vary freely. Concretely, at each time step t the decoder takes as input the preceding hidden state h^n_{t-1} and decoded action $\hat{y}^n_{t-1}$, and a vector h_{t-1} output by the pooling module (which is person-independent). A deep output transformation g, implemented as a multi-layer perceptron, is then applied on the resulting hidden state h^n_t. One ends up with action probabilities $p(y^n_t)$, in a very classical sequence transduction procedure, see e.g. Bahdanau *et al*. [7]. Formally, we have:

$$h^n_t = f_d(h^n_{t-1}; \hat{y}^n_{t-1}, h_{t-1}) \tag{3.3}$$

$$p(y^n_t) = g(h^n_t, \hat{y}^n_{t-1}, h_{t-1}), \tag{3.4}$$

where f_d is the decoder recurrent network, also implemented as a LSTM.

The j-th coordinate of the output of the pooling writes:

$$h_{t,j} = \max_n h^n_{t,j} \tag{3.5}$$

where $h^n_{t,j}$ designates the j-th coordinate of person n hidden state at time t.

The resulting action $\hat{y}^n_t$ then needs to be sampled from $p(y^n_t)$. As we want our discriminator to operate in the discrete action space, we use a softmax function with temperature P as a differentiable proxy when sampling for the discriminator, i.e. the t-th entry of discriminator input sequence for person n writes:

$$\hat{y}^n_t = \text{softmax}(p(y^n_t)/P) \tag{3.6}$$

where P is typically equal to 0.1, i.e. small enough so that softmax output is close to a one-hot vector.

3.3.3 DISCRIMINATOR

The discriminator can be implemented in many ways, but it usually relies on recurrent (RNN) or convolutional (CNN) networks, depending on the application. Trajectory prediction applications usually rely on recurrent networks [42, 95, 138], whereas convolutional networks are preferred for text generation [130, 67, 40], as CNN performances were shown to surpass that of recurrent networks on many NLP tasks [134, 34]. Convolutional networks also have the advantage to be inherently suitable to parallel computing, drastically reducing their computation time. Borrowing from both paradigms, we turn to a recurrent architecture that lends itself to batch computing, while preserving the sequential inductive bias of RNNs.

Dual-stream discriminator Two streams coexist within our discriminator such that the realism of both action sequences and participants interactions can be explicitly enforced. The individual stream (see Figure 3.2), labelled as D_{indiv}, is composed of a recurrent network followed by a two-class classifier, respectively implemented as a LSTM and a shallow feed-forward neural network, whose architecture is detailed in the following section. It operates on single-person action sequences, assessing their intrinsic realism disregarding any contextual information. The interaction stream, D_{inter}, follows the same architectural lines, but a pooling module similar to the one used in the generator is added right after the recurrent network such that the classifier takes a single input for the whole interaction. A factor λ_{inter} controls the relative importance of the interaction stream, such that the full discriminator writes:

$$D_{tot} = D_{indiv} + \lambda_{inter} D_{inter}. \tag{3.7}$$

Local projection discriminator Implementing (any of) the two discriminators as a recurrent network and letting them assess the quality of the entire generated sequences poses several issues. First the contributions from different sequence time steps to weight updates is likely to be unbalanced due to possible vanishing or exploding gradients. Second, some

of those gradients may be uninformative especially at the onset of training when errors are propagated forward in the generation process, degrading the quality of the whole sequence, or when the overall realism depends on localized patterns in the data. On the contrary we seek for a discriminator architecture that is better able to guide the generation with gradients corresponding to local evaluations, so as to entail an even contribution from each location in the generated sequence. To that end, previous works mainly used CNN discriminators [130, 67, 69, 61]. We explore recurrent architectures that conform with that objective. Along the same lines as PatchGAN [51] where photorealism is computed locally at different resolutions, we propose a multi-scale local (or window-based) discriminator, applying on overlapping sequence chunks of increasing width. Another intuition drove us to this choice: it seems reasonable to assume that the realism of an action sequence can be assessed locally, while it would not be the case for text sequences where verb tense or topic consistency would rather be assessed over the whole sentence. To that aim, the generated action sequence $\hat{\mathbf{y}}^n$ is split into K overlapping sub-sequences, or windows, temporally indexed by $t_1, \dots, t_K$, with K uniquely defined by the chunk length τ and the interval $\Delta t = t_{k+1} - t_k$ between successive chunks (see section 3.5 for a detailed discussion on how to select τ and Δt). Each sub-sequence is then processed independently through the recurrent module f of the discriminator:

$$h^n_{t_k} = f(\hat{\mathbf{y}}^n_{t_k:t_k+\tau}), \tag{3.8}$$

$\hat{\mathbf{y}}^n_{t_k:t_k+\tau}$ being the k-th action sub-sequence of person n (hence comprised between time steps t_k and $t_k + \tau$, plus the offset t_{obs}). Next, to account for the conditioning sequence $\mathbf{x}^n$ and its resulting code $h^n = f(\mathbf{x}^n)$, we implement a projection discriminator [81] and dampen the conditioning effect as we move away from the initial sequence thanks to a trainable attenuation coefficient. Discriminator output can finally be written as follows:

$$D(\mathbf{x}^n, \hat{\mathbf{y}}^n) = \frac{1}{K} \sum_k D_{proj}(h^n, h^n_{t_k}), \tag{3.9}$$

with

$$D_{proj}(h^n, h^n_{t_k}) = A(t_k^{-1/\beta}(h^n)^\top V\phi(h^n_{t_k})\mathbb{1}_{d_\psi} + \psi(\phi(h^n_{t_k}))), \tag{3.10}$$

where $\mathbb{1}_{d_\psi}$ is the vector of ones of size d_ψ, ϕ and ψ are fully-connected layers, A and V real-value matrices whose weights are learned along other discriminator parameters, and β a trainable parameter that controls the conditioning attenuation. In our case, given h^n, $h^n_{t_k} \in \mathbb{R}^{d_h}$, $\phi(\cdot) \in \mathbb{R}^{d_\phi}$, $\psi(\cdot) \in \mathbb{R}^{d_\psi}$, then $V \in \mathbb{R}^{d_h \times d_\phi}$ and $A \in \mathbb{R}^{1 \times d_\psi}$.

We repeat the same procedure over different window sizes τ to account for larger or smaller scale patterns, and average the scores to give the final output from the individual stream D_{indiv}. Slight differences arise for the computation of D_{inter}. Summation terms in (3.9) are no longer evaluated independently for each participant of the interaction. Instead, the N vectors $h^1_{t_k}, \ldots, h^N_{t_k}$ output by the encoder for all N participants at time t_k are first processed through the pooling module, yielding the pooled vector h_{t_k}. We do the same for individual conditioning vectors h^n, that are pooled to give a single conditioning vector h for the whole interaction. D_{proj} is then evaluated on h and h_{t_k} in (3.9), and its output indicates how much the interaction chunk starting at t_k is realistic given the observed interaction $\mathbf{X}$.

3.3.4 SocialInteractionGAN training losses

We use an adversarial hinge loss [66] to train our model, i.e. for the discriminator the loss writes:

$$\mathcal{L}_D = \mathbb{E}_{\mathbf{X}}\mathbb{E}_{\hat{\mathbf{Y}}}[\max(0, 1 + D_{tot}(\mathbf{X}, \hat{\mathbf{Y}})) + \max(0, 1 - D_{tot}(\mathbf{X}, \mathbf{Y}))], \tag{3.11}$$

as for the generator, the adversarial loss writes:

$$\mathcal{L}^{adv}_G = -\mathbb{E}_{\mathbf{X}}\mathbb{E}_{\hat{\mathbf{Y}}}[D_{tot}(\mathbf{X}, \hat{\mathbf{Y}})] \tag{3.12}$$

where the expectation is taken over dataset interactions $\mathbf{X}$ and the random matrix of model outputs $\hat{\mathbf{Y}}$, with $\mathbf{Y}$ the ground truth sequence.

We also add the following reconstruction loss to the generator:

$$\mathcal{L}_G^{reco} = \mathbb{E}_{\mathbf{X}}\mathbb{E}_{\hat{\mathbf{Y}}}\left[\|\hat{\mathbf{Y}} - \mathbf{Y}\|_2^2\right] \tag{3.13}$$

(where $\hat{\mathbf{Y}}$ and $\mathbf{Y}$ dependency on $\mathbf{X}$ is implicit) and weight it with a factor λ_{reco}. This last factor must be carefully chosen such that training mainly relies on the adversarial loss. This is important because, as showed in the experiments, the mean squared error fails to explore the diversity of action sequences. However, a small values of λ_{reco} can have an interesting stabilizing effect on training (see section 3.5). The overall generator loss writes:

$$\mathcal{L}_G = \mathcal{L}_G^{adv} + \lambda_{reco}\mathcal{L}_G^{reco}. \tag{3.14}$$

3.4 Evaluation of generated sequences quality

The Inception Score (IS) and Fréchet Inception Distance (FID) are two metrics designed to assess the quality and diversity of GAN-generated images and compare the distributions of synthetic and real data [96, 44]. In the absence of sequence class labels for computing the IS and of an auxiliary inception model for the FID, neither of these two metrics can however directly apply to discrete sequences. We begin by re-writing the calculation of the IS. The same original formula is used:

$$IS = \exp(H_M - H_C) \tag{3.15}$$

with H_M and H_C respectively the marginal and conditional entropies, but we rely on action labels to estimate the sequential equivalents of those two variables. The marginal entropy is computed over the entire set of predicted sequences and measures how well the GAN learned to generate diverse actions. It is therefore expected to be high. As for the conditional entropy, it needs to be low in a model that learned to grasp key features of a class, and thus we define it as the average entropy of individual sequences. Formally, we

write:

$$H_M = -\sum_{a \in \mathcal{A}} f(a) \log(f(a)) \tag{3.16}$$

$$H_C = -\frac{1}{|\mathcal{S}|} \sum_{a \in \mathcal{A}, s \in \mathcal{S}} f_s(a) \log(f_s(a)) \tag{3.17}$$

where $\mathcal{A}$ is the set of actions, $f(a)$ the frequency of action $a \in \mathcal{A}$ over all generated sequences, and $f_s(a)$ the frequency of action a in sequence s of the dataset $\mathcal{S}$ of size $|\mathcal{S}|$.

This new definition of H_C suffers from a limitation. Indeed, the conditional entropy decreases as the model learns to generate steadier and consistent sequences. However, very low values of H_C means that the same action is repeated over the whole sequence. Rather that aiming for the lowest possible values of IS, we therefore compute H_M and H_C from the data and use them as oracle values. In addition, we complement these scores with a sequential equivalent of the FID so as to compare the distributions of real and generated sequences.

The FID measures the distance between two image distributions by comparing the expectations and covariance matrices of intermediate activations of an Inception v3 network [106] trained on image classification [44]. However Iv3 is irrelevant for our discrete sequential data. Therefore we build a recurrent "inception" network that we train on an auxiliary task. The network is implemented as a bidirectional LSTM encoder, followed by five feed-forward layers, and is trained to regress the proportion of each action in input action sequences. The main principle that led to the choice of this task is that its output could be used to characterize the input sequences: it seems plausible for instance to assume that the realism of an action sequence can be partly assessed simply by knowing the proportion of each action. This way we expect the model trained on this task to learn meaningful sequence representations, making it suitable for FID calculations. In particular, the last activations before the regression head are used to compute the FID, which is referred to as SFID for Sequential Fréchet Inception Distance.

3.5 EXPERIMENTS

We conduct our experiments on the MatchNMingle dataset [15], that contains action-annotated multi-person interaction data that are particularly suited to our task. Although other datasets featuring human interaction data exist (e.g. [78, 3, 60, 59]), they do not fit in the framework described here, either because the discrete annotations are not exhaustive, or because they rely on transcripts that are not readily usable or on continuous quantities like body pose or raw video that are out of the scope of the present study. On the contrary, the MatchNMingle dataset contains annotated actions that fully define the state of all participants at each time step.

We carry out pre-processing on the original data, that we detail in section 3.5.2. In the absence of concurrent work, we challenge our architectural choices versus alternative recurrent and convolutional discriminator baselines (sections 3.5.4 and 3.5.4) and conduct an ablation study (section 3.5.4), highlighting the relevance of our dual-stream local discriminator.

3.5.1 MATCHNMINGLE DATASET

The MatchNMingle dataset contains annotated video recordings of social interactions of two kinds: face-to-face speed dating ("Match") and cocktail party ("Mingle"), out of which we exclusively focused on the latter. Mingle data contains frame-by-frame action annotated data for a duration of 10 minutes on three consecutive days, each of them gathering approximately 30 different people. Each frame is decomposed into small groups of chatting people that constitute our independent data samples. Mingle data comprises for instance 33 interactions of three people that amount to a total of 51 minutes of recording, annotated at a rate of 20 frames per second. We focused our experiments on three-people interactions as it offers a good trade-off between a sufficient complexity of action synchrony patterns and generalization capability that is dependent of the quantity of available data. Interactions are composed of eight different labelled actions: *walking*, *stepping*, *drinking*, *hand & head gesture*, *hair touching*, *speaking* and *laughing*, plus an indicator

of occlusion that can be total or partial.

3.5.2 Data pre-processing

The training dataset is built by processed Mingle data as follows. First, we consider each group of interacting people independently and split the interactions into non-overlapping segments of three seconds (60 frames). These 3-second segments constitute the conditioning data $\mathbf{X}$. Then we consider each segment's following actions as the target sequences $\mathbf{Y}$. Out of this dataset, we remove all training samples in which total occlusion accounts for more than ten percent of the sample actions, so as to limit the impact of the occlusions in the dataset. Our dataset finally comprises 600 three-people interaction samples. Finally, in order to ease data manipulation, the action space is restricted to the most common actions or combinations of actions (as some actions can occur together), and we replace the 8-dimensional binary action vectors of the data by one-hot vectors with the dimension of the resulting action space. We let the cumulative occurrences of possible actions amount to 90% of all actions to prevent the model from struggling with rare action combinations, and gathered all remaining actions under an additional extra label. Experiments with actions accumulating up to 99% of total occurrences are reported in section 3.5.4. The resulting action space contains the 14 following actions: *no action*, *speaking* + *hand gesture*, *speaking*, *stepping*, *head gesture*, *hand gesture*, *drinking*, *speaking* + *hand gesture* + *head gesture*, *hand gesture* + *head gesture*, *speaking* + *head gesture*, *stepping* + *speaking* + *hand gesture* + *head gesture*, *stepping* + *hand gesture* + *head gesture*, *stepping* + *hand gesture*, *laughing*.

3.5.3 Experimental details

In all our experiments we use layer normalization [6] in all recurrent networks, including that of SFID-inception network, for its stabilization effect on the gradients, along with spectral normalization [80] after each linear layer and batch normalization [50] in decoder deep output. All recurrent cells are implemented as LSTMs [47] with a hidden state dimension $d_h = 64$, and we choose the same dimension for the embedding space.

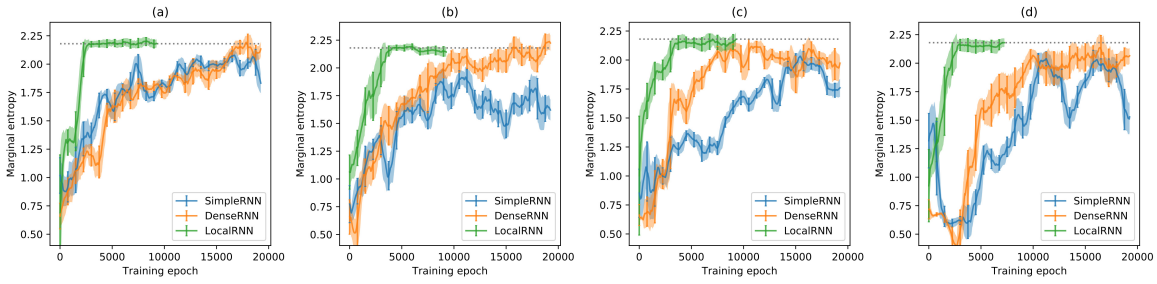


Figure 3.3: Evolution of marginal entropy for different recurrent discriminator architectures and training configurations: (a) sequences of 40 time steps with $\lambda_{reco} = 10^{-3}$; (b) sequences of 40 time steps with $\lambda_{reco} = 0$; (c) sequences of 80 time steps with $\lambda_{reco} = 10^{-3}$; (d) sequences of 80 time steps with $\lambda_{reco} = 0$. Gray dotted line represents the marginal entropy of the data (equal to 2.18). LocalRNN converges in all cases to ground truth marginal entropy values, even for long sequences and unsupervised training (i.e. $\lambda_{reco} = 0$), which is not the case for the two other baselines.

The dimensions of projection discriminator dense layers are $d_\phi = d_\psi = 128$. For the default configuration of our local discriminator, we use four different window sizes τ. Three of them depend on the output sequence length T: T, $T/2$, $T/4$ and we fix the smallest window size to 5. We set $\Delta t = \tau/2$, thus ensuring 50% overlap (rounded down) between consecutive chunks for all four resolutions. Different configurations are explored in section 3.5.4. For comparison purpose we build the CNN baselines in a similar fashion, with parallel streams operating at different resolutions and kernels of the same sizes as the chunk widths defined above. In all experiments, we use Adam optimizer [57] and set learning to 2.10^{-5} for the generator and 1.10^{-5} for the discriminator.

3.5.4 Experimental results

We articulate our experiments as follows: a comparison of recurrent model baselines, a comparison of recurrent and convolutional discriminators and an ablation study, that support the different architectural choices of our model.

Alternative recurrent discriminator baselines We start by comparing several recurrent architecture baselines for the discriminator, under different supervision conditions (i.e. varying the strength of the reconstruction loss) and for 40 and 80 frame-long se-

Table 3.1: Comparison of recurrent discriminator baselines for sequences of 40 time steps, with weak supervision ($\lambda_{reco} = 10^{-3}$) and no supervision ($\lambda_{reco} = 0$). Marginal entropy (H_M), conditional entropy (H_C) and SFID correspond to the training epoch that yields the best results. Best models are those that achieve the closest H_M and H_C values to the real data and the lowest SFID.

Model	λ_{reco}	$H_M(\rightarrow real)$	$H_C(\rightarrow real)$	SFID($\downarrow$)
Real Data		*2.18*	*0.30*	–
SimpleRNN	10^{-3}	2.07 ± 0.06	0.04 ± 0.03	0.72 ± 0.22
DenseRNN	10^{-3}	2.15 ± 0.05	0.04 ± 0.02	0.84 ± 0.21
LocalRNN	10^{-3}	$\mathbf{2.18 \pm 0.04}$	$\mathbf{0.26 \pm 0.06}$	$\mathbf{0.41 \pm 0.09}$
SimpleRNN	0	1.94 ± 0.05	0.38 ± 0.07	1.10 ± 0.25
DenseRNN	0	2.20 ± 0.07	0.22 ± 0.05	0.44 ± 0.09
LocalRNN	0	$\mathbf{2.18 \pm 0.03}$	$\mathbf{0.26 \pm 0.03}$	$\mathbf{0.24 \pm 0.04}$

quences (respectively 2 and 4 seconds). In section 3.3.3, we motivated our architectural choices on the hypothesis that a generator would benefit preferentially from multiple local evaluations rather that fewer ones carried on longer time scales. To support this assumption, we evaluate our local discriminator (hereafter LocalRNN) against two baselines: the first one, labelled as SimpleRNN, only processes the whole sequence at once and outputs a single realism score. The second one, referred to as DenseRNN, also processes the sequence at once, but in this case all intermediate hidden vectors are conserved and used as input for the classifier. This way, the discriminator output contains also localized information about actions, although the contributions to the score and the gradients of different time steps remain unbalanced. Results are gathered in Tables 3.1 and 3.2 for sequences of 40 and 80 time steps respectively, and correspond to the epoch that yields the best results for each model in terms of marginal entropy and SFID (longer training sometimes results in degraded performance, hence the early stopping). Corresponding marginal entropy evolutions are displayed in Figure 3.3. LocalRNN consistently produces the most realistic sequences in terms of marginal and conditional entropies, regardless of sequence length or supervision strength, and achieves the lowest SFID scores. Interestingly, we notice that if SimpleRNN seems to benefit from a weak reconstruction loss as is sug-

Table 3.2: Comparison of recurrent discriminator baselines for sequences of 80 time steps, with weak supervision ($\lambda_{reco} = 10^{-3}$) and no supervision ($\lambda_{reco} = 0$). Best models are those that achieve the closest H_M and H_C values to the real data and the lowest SFID.

Model	λ_{reco}	$H_M(\rightarrow real)$	$H_C(\rightarrow real)$	SFID($\downarrow$)
Real Data		*2.18*	*0.51*	–
SimpleRNN	10^{-3}	1.98 ± 0.06	0.26 ± 0.13	1.41 ± 0.36
DenseRNN	10^{-3}	2.12 ± 0.03	0.05 ± 0.04	0.82 ± 0.06
LocalRNN	10^{-3}	$\mathbf{2.16 \pm 0.05}$	$\mathbf{0.34 \pm 0.09}$	$\mathbf{0.74 \pm 0.27}$
SimpleRNN	0	1.96 ± 0.03	0.26 ± 0.08	1.87 ± 0.36
DenseRNN	0	2.13 ± 0.13	0.29 ± 0.15	1.40 ± 0.53
LocalRNN	0	$\mathbf{2.15 \pm 0.06}$	$\mathbf{0.45 \pm 0.10}$	$\mathbf{0.73 \pm 0.26}$

gested by a lower SFID, it is not the case for the two other models. The reconstruction loss has a squishing effect on the conditional entropy that is particularly detrimental for DenseRNN, yielding a model that does not properly generate action transitions. Finally, as one can see from the training dynamics (Figure 3.3), all LocalRNN models converge within three thousands training epochs, a much shorter time than any of the other recurrent discriminator baselines. Last but not least, the long-range recurrent evaluations of Simple and DenseRNN discriminators (on sequences of length $t_{obs} + T$) are replaced in LocalRNN by many short-range evaluations that can be efficiently batched, resulting in a much shorter inference time. This fast and efficient training advocates for our multi-scale window-based discriminator over recurrent architectures with larger focal length.

Local recurrent vs. convolutional discriminators Convolutional architectures are usually preferred for discriminator networks in adversarial text generation as it was shown to surpass performances of RNN on a diversity of NLP tasks while being naturally suited to parallel computing [134, 34]. We therefore compare SocialInteractionGAN (interchangeably referred to as LocalRNN) with several convolutional baselines, and illustrate the results in Figure 3.4. CNN discriminator baselines are built in a similar fashion to LocalRNN, with outputs averaged from several convolutional pipelines operating at different resolutions. Those are taken similar to section 3.5.3, with kernel width playing the role of

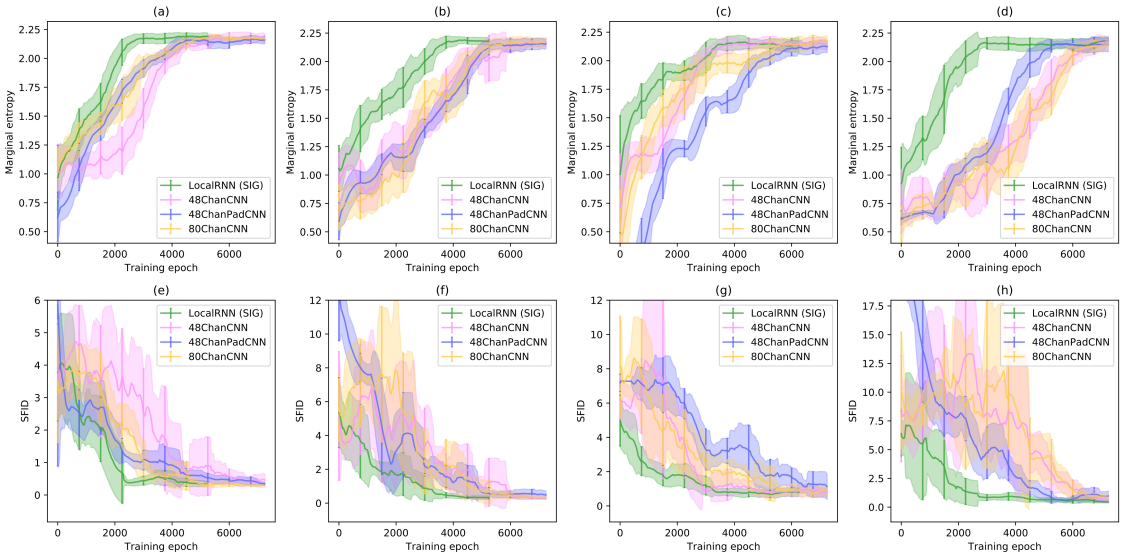


Figure 3.4: Evolution of marginal entropy and SFID of SocialInteractionGAN (LocalRNN) and various CNN discriminator baselines, for the following training configurations: (a) sequences of 40 time steps with $\lambda_{reco} = 10^{-3}$; (b) sequences of 40 time steps with $\lambda_{reco} = 0$; (c) sequences of 80 time steps with $\lambda_{reco} = 10^{-3}$; (d) sequences of 80 time steps with $\lambda_{reco} = 0$.

chunk length τ. Several variations around this standard configuration were investigated, such as increasing the number of channels or mirror padding in the dimension of actions to improve the expressive capability of the network. Models that gave the best results within the limits of the given GPU memory resources are plotted in Figure 3.4. Noticeably, all CNN architectures exhibit final marginal entropies close to the dataset values. In fact action sequences produced by most converged models, including LocalRNN, are hard to distinguish from real data and it is probable that the proposed task does not allow to notice differences in final performances. Nevertheless, SocialInteractionGAN consistently learns faster than the CNN baselines and exhibits a much smoother behaviour, as is particularly clear in charts (e)-(h) of Figure 3.4. Differences in training speed are even stronger when λ_{reco} is set to zero ((b), (d), (f) and (h)). Finally, we investigate the effects of processing action sequences at additional resolutions. We add two other pipelines to our standard configuration (i.e. two additional values of τ), and increase the depth of convolutional blocks to the limits of our GPU capability. The resulting model is labelled as

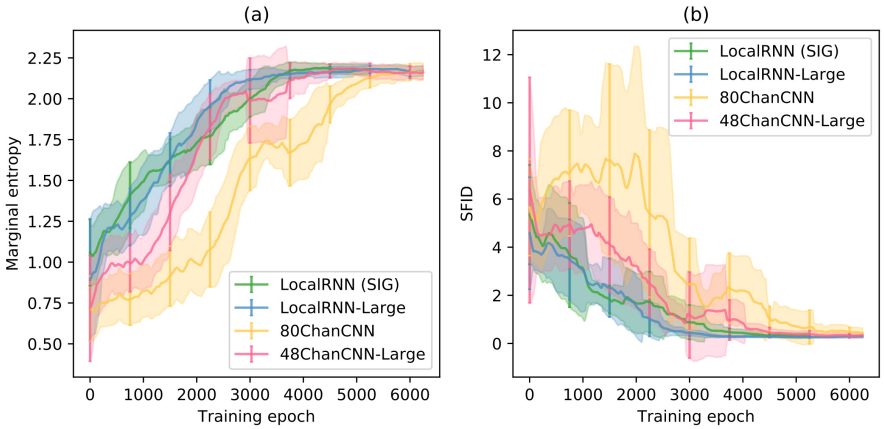


Figure 3.5: Effects of increasing model complexity on marginal entropy (a) and SFID (b), for sequences of 40 time steps and $\lambda_{reco} = 0$. The large CNN architecture remains slightly behind RNN-based discriminators.

Table 3.3: Effects of removing interaction stream (Indiv-stream), individual stream (Inter-stream), or adversarial losses altogether (No-GAN) versus the original model (Dual-stream) for sequences of 40 actions. Models are trained for 6000 epochs. Best models are those that achieve the closest H_M and H_C values to the real data and the lowest SFID.

Model	$H_M(\rightarrow real)$	$H_C(\rightarrow real)$	SFID($\downarrow$)
Real Data	*2.18*	*0.30*	–
Dual-stream (Full)	$\mathbf{2.18 \pm 0.03}$	0.26 ± 0.03	$\mathbf{0.24 \pm 0.04}$
Indiv-stream	2.15 ± 0.05	$\mathbf{0.27 \pm 0.09}$	0.42 ± 0.10
Inter-stream	2.23 ± 0.03	0.35 ± 0.04	0.77 ± 0.14
No-GAN	2.06 ± 0.01	0.14 ± 0.01	0.29 ± 0.01

`48ChanCNN-Large`. We build `LocalRNN-Large` in a similar fashion, augmenting the standard configuration with two additional time scales. Results are shown in Figure 3.5. Although the gap between the two architectures has been partly filled, large LocalRNN still converges faster than its large convolutional counterpart. These experiments show that for the generation of discrete sequences chosen from a finite set of human actions, it is possible to devise very efficient recurrent discriminator architectures that display more advantageous training dynamics compared to CNN discriminators, with noticeably lighter memory footprints.

Dual-stream discriminator ablation study We conduct an ablation study to explore the roles of the two streams of the discriminator. To that end, we run additional experiments on sequences of 40 time steps with $\lambda_{reco} = 0$, cutting off at turns the interaction and individual streams of the discriminator (respectively naming the resulting models `Indiv-stream` and `Inter-stream`). Additionally, we also compare with an architecture that has none of the two streams and that is trained only with the L_2 reconstruction loss (i.e. without adversarial losses), that we call `No-GAN`. The results are reported in Table 3.3, together with the full model (`Dual-stream`). Training without adversarial loss leads to much poorer scores in terms of marginal and conditional entropies, advocating in favor of the use of the adversarial loss. We hypothesize that the adversarial loss allows for a larger exploration of the action space, thus the higher marginal entropy score, and a better learning of the action sequence dynamics, as suggests the higher conditional entropy. Besides, disabling any of the two discriminator streams leads to degraded performances, which is especially clear in terms of SFID. In particular, we see from the high SFID that only relying on the interaction stream (Inter-stream) to produce realistic individual sequences would perform poorly. This supports our dual-stream architecture: an interaction stream alone does not guarantee sufficient individual sequence quality, but is still necessary to guide the generator into how to leverage information coming from every conversing participant. In the following chapter, we give another view of the dual-stream approach (coupled with the joint generation of multiple sequences) as a mean to reduce mode collapse, which provides additional insight on its effectiveness.

Pushing the limits of SocialInteractionGAN This section illustrates the effects of enriching and diversifying the original training dataset. Namely, we explore two variants from the initial setting: the addition of four-person interactions, and the use of a larger set of actions. The first experiment aims at assessing the capacity of the network to generalize its interaction sequence predictions to larger groups of people and learn more complex action patterns; we call it `SIG(3&4P)`. In a second experiment, we add more rarely seen actions to the pool of achievable actions, such that its cumulative occurrence in the original dataset raises from 90% to 99%. This results in a set of 35 actions, more than

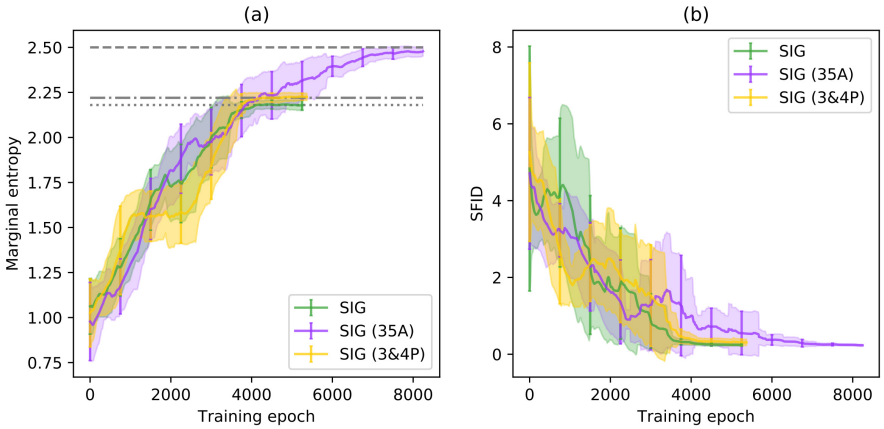


Figure 3.6: Effects of increasing the dimension of action space (`SIG(35A)`) and adding four-people interactions to the original three-people interaction dataset (`SIG(3&4P)`) on marginal entropy (a) and SFID (b). Nominal SocialInteractionGAN (`SIG`) is shown for comparison. All models are trained on sequences of 40 actions with $\lambda_{reco} = 0$. Gray dashed, dash-dotted and dotted lines represent respectively real data marginal entropy for large action space, three- and four-people interactions and original SocialInteractionGAN experiments.

doubling the original size. We label the resulting model `SIG(35A)`. The evolution of marginal entropy and SFID of these models over training steps are reported in Figure 3.6, together with that of the reference model (`SIG`). Models were trained on sequences of 40 time steps with $\lambda_{reco} = 0$. Although `SIG(3&4P)` uses an additional 50% training samples (from 600 sequences to 953), this has very limited effects on the training dynamics. Noticeably, the model learns to match the slight increment in data marginal entropy produced by the richer interaction samples. As it could be expected from the resulting increase in network complexity, expanding the action set of `SIG(35A)` leads to delayed convergence. However the model smoothly converges to the marginal entropy of the real data and also scores low SFID, meaning that SocialInteractionGAN is able to handle a large number of actions given a sufficient amount of training data.

3.6 CONCLUSION

Understanding action patterns in human interactions is key to design systems that can be embedded in e.g. social robotic systems. For generative models of human interactions, the main challenge resides in the necessity to preserve action consistency across individuals and time. In this chapter we presented a novel conditional GAN model for the generation of discrete action sequences of people involved in an interaction. Although interactions are modelled in a low dimensional manifold, we believe that the principles devised here, namely the necessity to share the state between participants at every time step, and to focus the loss function on local evaluations, will apply on richer representations. In the following of this manuscript, we show that these principles indeed scale to higher-dimensional data.

CHAPTER 4

AUTOREGRESSIVE GAN FOR SEMANTIC UNCONDITIONAL HEAD MOTION GENERATION

From this chapter on, we investigate scaling the previous architecture to higher dimensional data, namely videos of talking people. This is of particular interest because video understanding and generation are active topics in computer vision that have many practical applications. When it comes to human interaction, these applications comprise for instance face reenactment, talking head generation or co-speech gesture generation. A major difference from the previous chapter is that mainly due to data scarcity issues, we no longer attempt to generate the behavior of several interaction participants but rather that of a single person *involved in an interaction*. This new setting intersects the well-studied task of talking head generation, which nevertheless still faces a number of open research questions. These concern the improvement of the generated head motion quality and the yet unexplored multi-scale audio-visual syncing that form the core of the two following chapters. Importantly, we will also show that the approach devised in Chapter 3 for discrete interaction data generation remains largely relevant in the following of this thesis. One major challenge however resides in the complexity of video data generation, due to the high dimension of the data and the necessity to maintain the temporal coherence of visual and dynamics cues across multiple frames. In the following chapters, we alleviate this issue by working in a low-dimensional representation of the face known as *landmarks* (see e.g. Figure 4.2). This allows to focus on the dynamics of the head, while leaving the visual reenactment process as a separate task, not treated in this manuscript (see however Sections 4.1 and 5.1 for a review of existing reenactment methods from 2D landmarks). The following two chapters therefore deal with yet unsolved issues associated with head motion generation, that we address in the landmark domain by building on several ideas previously introduced.

In the present chapter, we address the task of unconditional head motion generation which aims to animate still human faces from a single reference pose. Different from traditional audio-conditioned talking head generation that seldom puts emphasis on realistic head motions, the GAN-based architecture devised here learns to synthesize rich head motion sequences over long duration while mitigating error accumulation. In particular, the autoregressive generation of residual outputs ensures smooth trajectories, while

a multi-scale discriminator on input pairs drives the generation toward better handling of high- and low-frequency signals and less mode collapse. We experimentally demonstrate the relevance of the proposed method and show its superiority compared to models that attained state-of-the-art performances on similar tasks.

4.1 Introduction

Talking head generation refers to the task of animating a human face generally using a single reference image, an audio clip, and possible additional conditioning signals such as emotional state or exemplar pose dynamics [109, 105, 55, 128, 140]. Different from face reenactment where a driving video clip is provided, in talking head generation the head pose, facial animation, and lip synchronization need to be inferred from other modalities. To tackle the difficulty of handling both facial dynamics and photorealism directly in the image space, a predominant line of research generates dynamics in a lower dimensional space [118]. Those representations comprise supervised facial landmarks [20, 141], 3D mesh [32] or unsupervised keypoints [123, 124], and following the designation of *high level semantics* used in Villegas *et al*. [118], we refer to this space as the *semantic space*.

Although several works achieved compelling results in lip-syncing and realistic rendering, generating natural head motions has, until recently, consistently received less attention. In the lack of a driving audio signal, it is yet crucial for the synthesis model to produce natural and diverse head motions. This is relevant in applications where no audio signal is available, e.g. when animating background characters in a scene or a video game. In this unconditional generation setting, the focus shifts from audio-visual synchrony toward long-term consistency throughout the sequence in the absence of conditioning signal, which is known to be particularly challenging [120]. Tackling this problem will also be beneficial for audio-conditioned talking head synthesis (see Chapter 5), as it lays the architectural foundations for a fine handling of head dynamics. The present chapter addresses the task of unconditional head motion sequence generation, i.e. synthesizing head pose and facial expression given a single reference pose and no audio driving signal.

Importantly we work in the 2D facial landmarks semantic space, which facilitates the manipulation of head dynamics. Several models were notably proposed to map landmarks to real world images, making this representation relevant in practice [132, 131, 137, 79].

A major difference from the previous discrete action setting is that landmark positions $\{x_t\}_t$ indexed by time t now span a continuous manifold. It is therefore convenient to represent position x_t as the cumulative sum of incremental displacements, or instantaneous velocities, starting from the observed initial position x_0:

$$x_t = x_{t-1} + v_t = x_0 + \sum_{\tau \leq t} v_\tau. \tag{4.1}$$

This approach has been followed successfully, for instance, by Lin & Amer [69] or Kundu *et al*. [62] for human pose generation and by Gupta *et al*. [42] for trajectory prediction. As shown by Martinez *et al*. [77], this formulation allows to use shallower neural network architectures. Another feature of such cumulative sum is that they can be properly described by autoregressive models (see Morrison *et al*. [82] for an experimental validation of this assertion). In a most general definition, an autoregressive function G produces coordinates x_t one by one given the input position x_0 and all previously generated positions:

$$x_t = G(x_0, x_{1:t-1}) \tag{4.2}$$

In practice, conditional independence property assumptions can be made to reduce the necessity to model all previous time steps and allow for the use of a large diversity of network architectures on a fixed history length. Although they can produce sequences of arbitrary length, autoregressive models may however accumulate error, or alternatively end up generating average values over time when trained with a mean squared error loss [77]. This advocates for the use of other loss functions. We hereby introduce an adversarial framework to tackle head motion generation as an autoregressive velocity prediction problem, which to the best of our knowledge has never been done before for head motion prediction. To that end, we leverage ideas outlined in Chapter 3 for the design of the discriminator network, that prove equally relevant to handle head motion sequences. Head

motion dynamics are composed of temporal patterns that evolve over varied timescales. Previous works have addressed the generation of such data with discriminator networks operating on receptive fields of different sizes [125, 61] or on local windows, enabling a better representation of high-frequency components [51]. The discriminator network employed here, directly inspired from the one introduced in the previous chapter, implements a multi-scale window-based architecture in a single network, which allows it to operate at any temporal resolution. Last, in the light of Lin *et al*. [70], we revisit the *interaction stream* of SocialInteractionGAN where pairs of samples are processed by the generator and the discriminator networks as a mode collapse mitigation technique. As we show, this approach does not change the optimization objective but brings a significant performance boost for a limited additional overhead. The proposed GAN architecture, labeled Semantic Unconditional Head Motion or SUHMo, allows for long-term head motion synthesis, and experiments confirm its proficiency against a diversity of models and baselines.[1]

The contributions of this research work are:

- An autoregressive GAN framework for unconditional head motion generation in the 2D-landmarks domain, able to mitigate error accumulation over long sequences that even extend the duration of training sequences,

- A training methodology that can be generalized over diverse architectures, for which we detail two implementations based on LSTM and Transformers,

- Extensive experiments showing that the proposed SUHMo method surpasses competitive methods from closely related tasks on two benchmark datasets.

[1]Source code and animated examples can be found at: https://github.com/LouisBearing/UnconditionalHeadMotion.

4.2 Related work

4.2.1 Deep continuous autoregressive models

Autoregressive models are ubiquitous in sequence modeling, as they enable strong temporal consistency thanks to the explicit relation between consecutive outputs. In the context of deep continuous sequence prediction, autoregressive models proved powerful in as diverse domains as waveform synthesis [61], image generation [110], human trajectory prediction in a crowd [42] or human motion prediction [77, 69, 62, 4]. Surprisingly, the talking face generation literature is much sparser on this subject, Fan *et al*. [32] presenting one of the few autoregressive talking head generation architectures, but they do not attempt to generate head motions. Different from previous works, we leverage the potential of autoregressive models to produce smooth and realistic head motions.

4.2.2 Multi-scale data processing

Learning on representations of the input data over multiple scales has become the standard in computer vision tasks such as object detection or semantic segmentation where objects of the same class can have different sizes [68, 107]. Uncovering multiple patterns with GANs was first addressed in Isola *et al*. [51] where the authors introduced a discriminator network taking image patches as input to enhance high spatial frequency components. In Wang *et al*. [125], an output image pyramid is processed by several discriminators that operate on decreased resolutions and larger receptive fields, driving the generator network to produce realistic patterns at different scales. The multi-scale discriminator has then been extended to sequence generation tasks [69, 61]. An interesting aspect of the latter discriminator architectures is that they combine multi-scale with window-based evaluations in 1D equivalents of PatchGAN [51], and benefit from the advantages of processing short windows, such as a lighter discriminator architecture. One limitation however is that different networks are trained for each resolution, restricting in practice the number of scales considered. We propose to use functional forms that are invariant to the input sequence

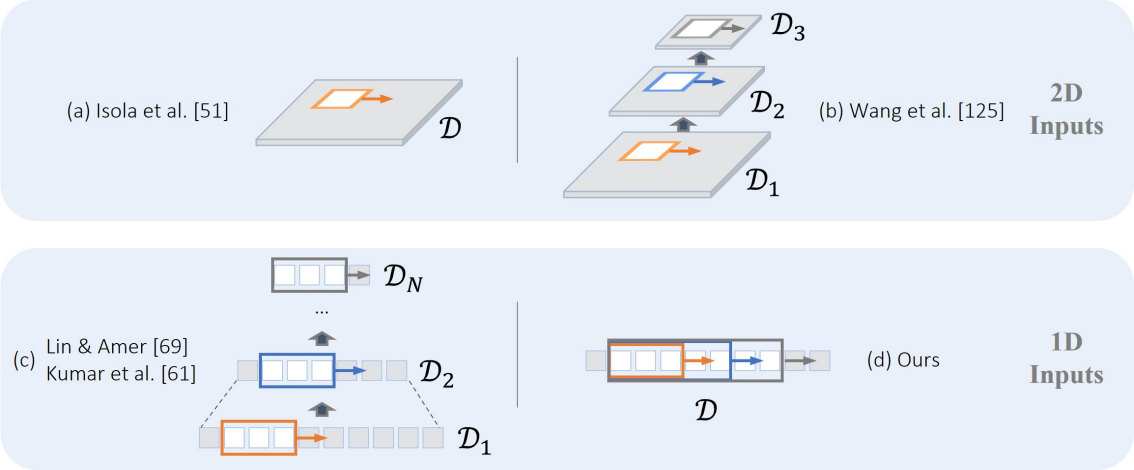


Figure 4.1: One and two-dimension multi-scale & window-based discriminator architectures. (a) The purely window-based PatchGAN discriminator [51]. (b) Extension to a 3-scale architecture in Wang *et al*. [125]. (c) The 1D multi-scale PatchGAN structure used in DVGANs [69] and MelGAN [61] discriminators. (d) The proposed multi-scale window-based discriminator has a unique set of parameters and takes sequences of any size as input, giving a free hand to select the scales.

length, such as recurrent networks. This way, it is possible to define discriminator networks that operate at arbitrary scales with a unique set of parameters. See Figure 4.1 for a visual comparison of the different discriminator architectures. Finally, implementing the generative model itself as a multi-scale network has also proved useful in improving the fidelity of generated images at multiple spatial scales [28, 56].

4.2.3 Mode collapse mitigation

Mode collapse reduction methods in GANs comprise efforts towards better optimization procedures [5], generation space regularization [17], or forcing the network to account for the noise vector [22], among a rich body of literature. Lin *et al*. [70] proposed an intuitive way of driving the generator to produce diverse outputs by feeding the discriminator with several input samples. We extend this framework by generating two inputs *together*, which produced better results while leaving the optimization objective unchanged.

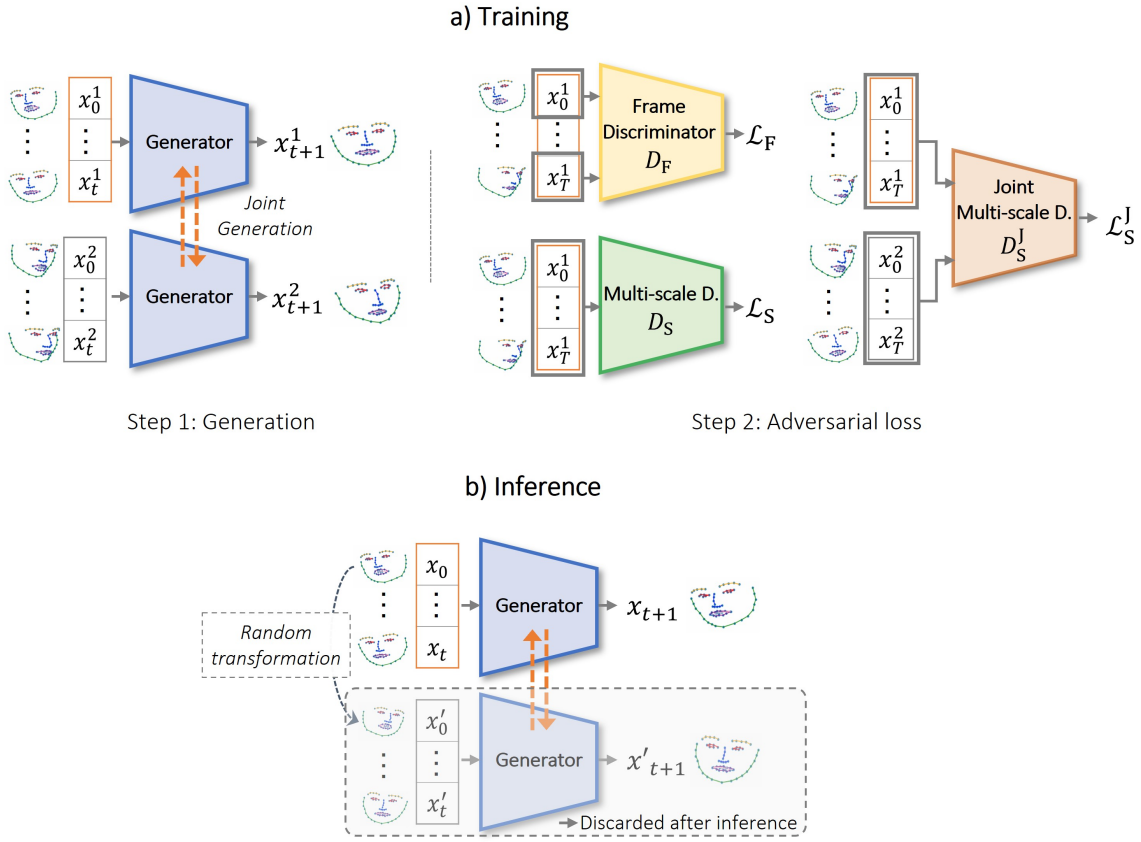


Figure 4.2: (a) Overview of SUHMo training process. The autoregressive generator produces pairs of outputs, that are evaluated by three discriminator networks. (b) At test time, the second sample is replaced by a transformed version of the reference pose.

4.3 AUTOREGRESSIVE UNCONDITIONAL HEAD MOTION GENERATION

In this section, we formally define the unconditional head motion generation task and the key components of our learning framework. Given a set of facial landmarks x_0 representing a face in an initial pose, we seek to generate a sequence $x_{1:T} = (x_1, \ldots, x_T)$ of arbitrary length T such that the probability distributions of the generated and the ground truth data, p_{G} and p_{data}, match:

$$p_{\text{G}}(x_{0:T}) = p_{\text{data}}(x_{0:T}), \quad \forall x_{0:T} \tag{4.3}$$

We hereafter describe our adversarial architecture to address this problem, an overview

of which can be found in Figure 4.2. Its main components include the autoregressive generator, described in Section 4.3.1, and the multi-scale sequence discriminator, presented in Section 4.3.2. As an attempt to mitigate the potential negative impact of mode collapse, we design our architecture to learn to generate and discriminate joint probability distributions, as explained in Section 4.3.3. The overall loss function is presented in Section 4.3.4. Finally, in Section 4.3.5 we propose two implementations of our method to stress its generalizibility.

4.3.1 AUTOREGRESSIVE VELOCITY GENERATION

We implement our generator network G as an autoregressive function of past landmark positions, that at each time steps provides the instantaneous velocity:

$$x_t = x_{t-1} + G(x_{0:t-1}) \tag{4.4}$$

Working with velocities ensures smooth transitions between subsequent time steps but also enables simpler model architectures [77] and provides a convenient way to take advantage of the inherent potential of autoregressive models to represent cumulative sums [82]. On the other hand, autoregressive models tend to accumulate errors over time and special care must be taken in the training process to mitigate it, thus allowing for practical applications. The following sections detail the architecture of our discriminator and the learning strategy that enable long sequence generation. Note several differences from the setting of Chapter 3: first of all, the quantity of interest is now continuous, second, only one observed initial pose is provided in the present use case. This pleads for the use of a single network rather than the previous encoder-decoder architecture, which has the advantage to prevent any discontinuity between the last observed frame and the first decoded frame frequently encountered with recurrent networks. Last, we show in the following that several components previously designed for the generation of interactions have interesting mode collapse reduction properties in a single person setting.

4.3.2 WINDOW-BASED MULTI-SCALE DISCRIMINATOR

We use a multi-scale, window-based discriminator network architecture to train the model to generate temporal patterns unfolding over different timescales. To relieve the burden of training one network per input scale, we follow the ideas of Chapter 3, which allows to considerably simplify the discriminator architecture. Here we extend the previous window-based, multi-scale discriminator beyond RNNs only.

First, let $D_{\text{M}} : (x_{t:t+\tau}, \theta) \in \mathbb{R}^{\tau \times d} \times \mathbb{R}^{d_\theta} \mapsto D_{\text{M}}(x_{t:t+\tau}; \theta) \in \mathbb{R}$ be a discriminator function parameterized by θ that operates on sequences of d-dimension vectors of arbitrary length τ. This definition includes RNNs, Transformers [116], and more generally any function enabling pooling in the time axis or which processes time steps separately. We then define the window-based multi-scale discriminator D_{S} on sequences $x_{0:T}$ as an expectation over evaluations of D_{M} on temporal slices of $x_{0:T}$:

$$D_{\text{S}}(x_{0:T}; \theta) = \mathbb{E}_{\tau,t}[D_{\text{M}}(x_{t:t+\tau}; \theta)], \quad t \geq 0, t + \tau \leq T \tag{4.5}$$

where τ and t are respectively the duration, i.e. the scale, and starting index of the window. In practice both t and τ are sampled from discrete uniform distributions. The advantage of this framework is that it gives a flexible way to adjust the scales by choosing various distributions on τ.

4.3.3 LEARNING TO GENERATE AND DISCRIMINATE JOINT PROBABILITY DISTRIBUTIONS

In the light of the results presented in Lin *et al.* [70], we give here another interpretation of the interaction stream introduced in the preceding chapter. Although originally designed to process outputs from multiple agents, we found that it can have a beneficial effect on mode collapse reduction even when generating interactions is not the current objective. We thus consider the generation and discrimination of joint sample distributions. Let the objective, with generic data points x^1 and x^2, write (superscript J for joint ground truth /

generated distributions):

$$\mathbb{E}_{x^1,x^2\sim p^{\text{J}}_{\text{data}}}[\log D(x^1,x^2)] + \mathbb{E}_{x^1,x^2\sim p^{\text{J}}_{\text{G}}}[\log(1-D(x^1,x^2))] \tag{4.6}$$

This has to be minimized (resp. maximized) w.r.t. the parameters of the generator G (resp. the discriminator D). In the case of independent and identically distributed data and enough network capacity, the joint generated distribution converges to the product of the marginal data distributions [36]:

$$p^{\text{J}}_{\text{G}}(x^1,x^2) = p^{\text{J}}_{\text{data}}(x^1,x^2) = p_{\text{data}}(x^1)p_{\text{data}}(x^2) \tag{4.7}$$

If G produces samples independently, then p^{J}_{G} readily factorizes. This is the setting of Lin *et al*. [70], which proved useful to reduce mode collapse. However, if x and y are produced together, then G simply *learns* to factorize. Both cases lead to the equality of marginal distributions $p_{\text{G}} = p_{\text{data}}$, hence the optimization objective of Goodfellow *et al*. [36] is unaffected. In the real case scenario of limited network capacity, p^{J}_{G} does not factorize, and hence we argue that if the generation is prone to mode collapse then the overall optimization can benefit from this joint generation process. In such cases, it is an easy task for D to identify generated pairs by comparing the two samples, hence driving G to leverage its two inputs to increase the generation diversity.

At test time, a single initial pose is typically provided. Since the model expects a pair of samples, one strategy consists in providing a transformed version of the reference pose as a second input. To that end we use random flip, rescaling and translation. This approach gives a practical way of injecting stochasticity in the generation process (see Section 4.4.3).

4.3.4 Training SUMHo

Following the discussion in 4.3.2 and 4.3.3, we propose to use two window-based multi-scale discriminators on the generated sequences. The joint discriminator D^{J}_{S} operates on sample pairs, while a second network, D_{S}, takes single sequences as input and explicitly

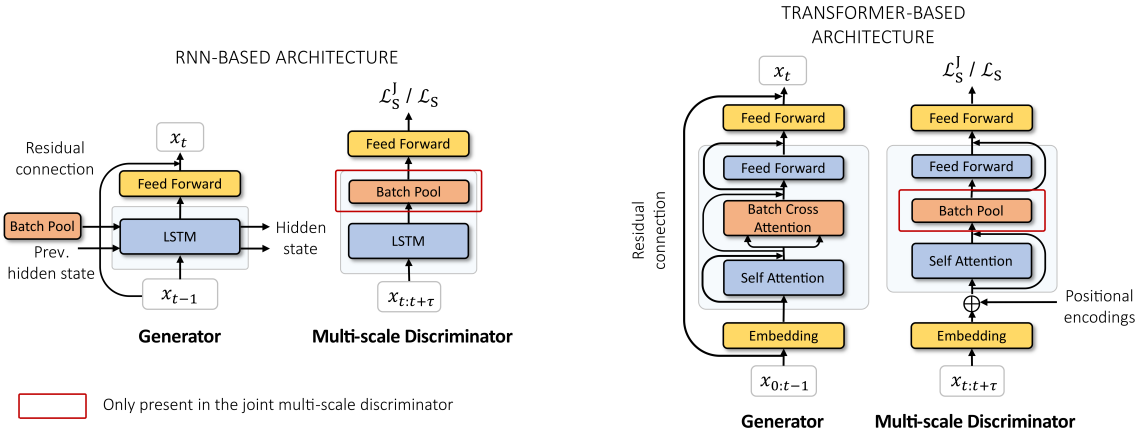


Figure 4.3: The two architecture variants of the proposed SUHMo method.

enforces the convergence of the marginal distributions p_{G} and p_{data}. Finally, to complement the sequential losses, we employ a frame discriminator D_{F} to measure the realism of each time step of the produced sequences (see Figure 4.2). The generator adversarial losses writes:

$$\mathcal{L}_{\mathrm{S}}^{\mathrm{J}} = -\mathbb{E}_{x^1_{0:T}\sim p_{\mathrm{G}}, x^2_{0:T}\sim p_{\mathrm{G}}}\left[D_{\mathrm{S}}^{\mathrm{J}}(x^1_{0:T}, x^2_{0:T})\right], \tag{4.8}$$

$$\mathcal{L}_{\mathrm{S}} = -\mathbb{E}_{x_{0:T}\sim p_{\mathrm{G}}}\left[D_{\mathrm{S}}(x_{0:T})\right], \tag{4.9}$$

$$\mathcal{L}_{\mathrm{F}} = -\mathbb{E}_{x_{0:T}\sim p_{\mathrm{G}}}\left[\frac{1}{T}\sum_{t\geq 1} D_{\mathrm{F}}(x_t)\right]. \tag{4.10}$$

The overall loss function is the sum of these three losses plus a mean squared error term $\mathcal{L}_2^{reco}$ that we scale to remain negligible after the first training epochs:

$$\mathcal{L} = \underbrace{(\mathcal{L}_{\mathrm{S}}^{\mathrm{J}} + \mathcal{L}_{\mathrm{S}} + \mathcal{L}_{\mathrm{F}})}_{\text{Adversarial loss}} + \lambda \mathcal{L}_2^{reco} \tag{4.11}$$

Following Chapter 3, the geometric GAN formulation of Lim & Ye [66] is used for the discriminator loss functions.

4.3.5 IMPLEMENTATION

So far the discussion has not assumed any precise functional form for either the generator or the discriminator network. Here we propose two implementations of the SUHMo method, based on LSTM and Transformers. The motivation is to highlight that the provided methodological tools can be relevant beyond a single architecture, as we further discuss in Section 4.4. An overview of both proposed variants can be found in Figure 4.3. To account for pairs of inputs, we define a batch-pool (BP) operator that acts as a max pooling layer of kernel size 2 along the batch dimension; with the difference that the result is then repeated to preserve the input batch size:

$$x^o = \text{BP}(x^i), \tag{4.12}$$

$$x^o_{2n-1,c,d} = x^o_{2n,c,d} = \max(x^i_{2n-1,c,d}, x^i_{2n,c,d}), \tag{4.13}$$

where the subscripts represent the batch, channel and dimension indices. In the LSTM-based generator, the hidden state h_t goes through a BP layer, yielding a pooled vector p_t that is concatenated with the next input to the LSTM. A multi-layer perceptron is used on h_t to output the landmark positions. The joint discriminator D_{S}^{J} is composed of a LSTM, a BP layer and a feed forward network; the marginal discriminator D_{S} is similar but without the BP layer (see Figure 4.3, left).

In the Transformer generator (Figure 4.3, right), pair mixing is done in a multi-head attention (MHA) layer by inverting the batch indices of paired samples in the key and value vectors. This way, each sample in a pair can attend to the history of the other sample. This layer is labelled batch-cross attention (BXA):

$$\text{BXA}(q, k, v) = \text{MHA}(q, k^r, v^r), \tag{4.14}$$

$$k^r_{2n} = k_{2n-1}, \quad k^r_{2n-1} = k_{2n} \quad \forall \quad 1 \leq n \leq N/2, \tag{4.15}$$

$$v^r_{2n} = v_{2n-1}, \quad v^r_{2n-1} = v_{2n} \quad \forall \quad 1 \leq n \leq N/2, \tag{4.16}$$

with N the batch size, and $q = k = v$ in all experiments, i.e. query, key and value

Figure 4.4: Sample sequences from CONFER and VoxCeleb2 datasets and the associated *motion maps*. Samples featuring little movement produce a very sharp motion map (example c). The other samples give a good illustration of the differences in dataset preprocessing: head translation is suppressed in VoxCeleb2 sequences that only contain rotations, hence the quasi-static position of noise-tip landmarks in d, e and f. On the contrary, both translation and rotation movements are visible in the motion maps of samples a and b.

tokens originate from the same sequences. We do not use positional encoding as we observed no change in performance, while omitting it allows the generation of longer test sequences. As for the discriminator networks, a batch-pool layer replaces the batch-cross attention in D_s^J as it only needs to provide a single score per pair. A learnable class token, prepended to the input sequence, is used to give the final score, as it has been customary for Transformers [29].

4.4 Experiments

4.4.1 Experimental details

All networks in the RNN variant of our method are implemented as 1-layer LSTM with hidden size 1024, while Transformer networks are implemented as a single self-attention block with one head. In the latter architecture, embedding layers produce 1024 dimen-

sional vectors for the generator and 128 dimensional vectors for the discriminators, i.e. the balance between G and D is mainly controlled by the embedding dimension. Models were trained on sequences of 40 time steps, and up to 5 observed frames were given as input seeds to the LSTM to stabilize training. At inference time a single reference frame is provided, and we explore predicting sequences of two different durations, namely 40 and 80 time steps, or respectively $1.6s$ and $3.2s$.

We set λ in equation 4.11 to 10^{-2}. Networks were trained with Adam optimizers with β_1 and β_2 parameters set to 0.5 and 0.999, and with generator and discriminator learning rates set to 2×10^{-5} and 1×10^{-5} respectively. Importantly, a step learning rate decay of a factor 10 was applied once performance started to stall, corresponding to roughly 60k iterations for a batch size of 120 (~ 3000 epochs for CONFER and 1000 epochs for our VoxCeleb2 subset). Training took on average two days on a single Titan RTX GPU.

We investigated concatenating velocities or instantaneous accelerations to landmark positions as input to the generator or the discriminators, expecting that it might help penalizing static sequences produced by G. In practice, we use positions and velocities as inputs to the generator and all three quantities in the discriminator networks.

Experiments were conducted on two talking-head datasets. **CONFER** [35] contains 72 video clips of TV debates between two persons, each about 1 minute long. We preprocessed the data preserving head translations and selected 5 clips as test data featuring persons unseen at training. Second, we trained on a randomly selected subset from **VoxCeleb2** [24], leaving 674 video clips corresponding to 10 unseen identities as test set. In both datasets the video frame rate is 25 fps.

In order to draw robust conclusions despite the inherent variability associated with GAN training, each GAN model was trained three times, such that the results reported in all tables contain both mean values and standard deviations.

Table 4.1: Model comparison on the head motion generation task from a single reference frame on CONFER and VoxCeleb2. The FVD and *t*-FID are sequential metrics computed on fixed sequence lengths reported as subscript. Here all metrics are computed over the 40 last predicted time steps.

Sequence length (frames)	40			80		
Method	FVD_{40} ↓	FID ↓	$t\text{-FID}_{40}$ ↓	FVD_{40} ↓	FID ↓	$t\text{-FID}_{40}$ ↓
CONFER [35]						
HiT-DVAE [10]	368±19	6±0.4	130±7	764±35	50±2	157±12
ACTOR [87]	480±12	8±0.3	147±3	667±20	9±0.8	163±5
Δ-based	318±115	21±3	67±10	357±104	24±3	77±18
MLE	480±42	10±3	133±2	777±54	21±3	159±6
SUHMo - RNN	**162±31**	**3±0.2**	**61±8**	**147±45**	8±2	**48±11**
SUHMo - Trans.	175±46	4±0.7	67±12	169±33	**7±1**	52±4
VoxCeleb2 [24]						
HiT-DVAE [10]	686±37	**1±0.1**	167±4	644±27	**2±0.1**	164±6
ACTOR [87]	357±55	4±0.5	78±9	431±26	5±2	145±21
Δ-based	386±32	48±6	89±4	518±48	60±30	112±31
MLE	530±20	2±0.2	158±6	684±23	8±0.8	172±9
SUHMo - RNN	**76±8**	3±0.7	**21±3**	**135±33**	9±5	**31±7**
SUHMo - Trans.	134±33	3±0.8	42±10	141±31	9±3	55±16

4.4.2 METRICS

The Fréchet Inception Distance (FID) [44] and Fréchet Video Distance (FVD) [112] are used to measure the distance of the generated samples to the ground truth data distribution. While the FID gives a score of static face realism, the FVD measures the smoothness of the dynamics. A preliminary rasterization step is applied on landmarks to cast them in the image domain for the inceptionV3 [106] and I3D [16] networks. We also complement the FVD with a second dynamical metric based on a FID measure on *motion maps*, that we use to represent sequences on a single image. To do so, we compute an exponential moving average centered on the last time frame, thus enforcing a visual correlation between pixel intensity and time step index. The resulting metric, that is relevant in particular to discriminate sequences with little movement, is coined *t-FID* (t standing

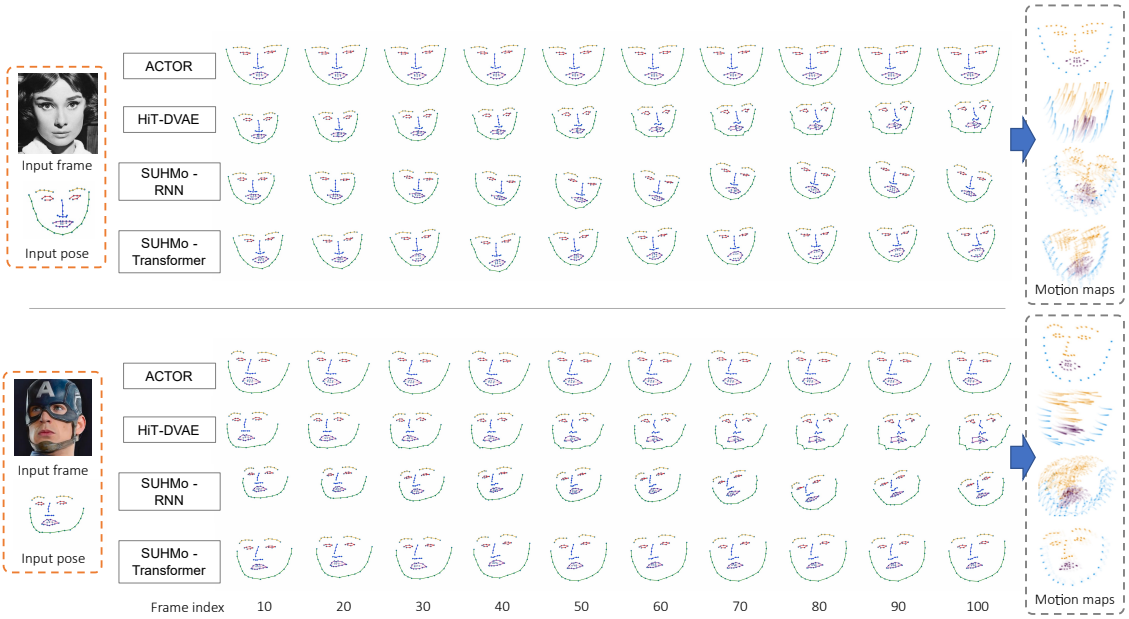


Figure 4.5: Qualitative evaluation of results from different models on in-the-wild images, and for sequence generation of one hundred frames. Models are trained on the CONFER dataset.

for time). Examples of data samples and their corresponding motion maps are illustrated in Figure 4.4.

4.4.3 Models comparison

Quantitative comparison The performances of SUHMo were compared with two state-of-the-art architectures for human pose prediction, **HiT-DVAE** [10] and **ACTOR** [87]. This task consists in predicting future positions of body joints given a short observed sequence or an action label and is therefore comparable to unconditional head motion generation. In our attempt to build the most capable unconditional head motion synthesis model, we thus train these two systems on our talking head datasets and compare their results. One notable difference though arises from the fact that human pose prediction datasets are usually composed of several modes corresponding to a predefined set of actions, and synthesis models typically account for this by conditioning the generation on an action label. A minimal amount of changes is therefore necessary to adapt the previous models to our setting: we replace in particular the action conditioning in ACTOR by

the observed initial frame. We also seek to compare with audio-conditioned talking head generation models. Although it is not possible to evaluate them directly in the absence of audio signal, we take inspiration from common practices in talking head generation to build two additional baselines. The **Δ-based** model reproduces the SUHMo-RNN method, but similarly to Zhou *et al*. [141] and Das *et al*. [27] it produces displacements from a fixed set of reference points, in this case the initial landmark positions. **MLE**, for maximum likelihood estimation, follows a common trend in head motion prediction and relies on a single mean squared error loss. We evaluate the above models and our two architecture variants on both CONFER and VoxCeleb2, on sequences of duration 40 and 80 frames. Note that this corresponds to one time and twice the training sequence duration. Results are reported in table 4.1. SUHMo consistently outperforms all other architectures in terms of dynamics quality. HiT-DVAE and ACTOR attain lower FID values on VoxCeleb2, suggesting slightly sharper faces, but this is at the cost of producing quasi-static sequences, hence the poor FVD and *t*-FID scores (see also next paragraph and Figure 4.5). The same is true for models trained with a L_2 reconstruction loss (the likelihood-based method), advocating for the use of an adversarial loss to ensure realistic dynamics. The Δ-based variant produces dynamics of uneven quality, as per the high standard deviations, and the realism of produced faces falls significantly behind, as suggests the higher FID values. Interestingly, SUHMo exhibits very little drift as time stretches and dynamics metrics remain very low, contrary for instance to HiT-DVAE. We note however that this is an extreme setting for the use of HiT-DVAE in terms of generation over observed length ratio which is typically of the order of 3 to 5 in Bie *et al*. [10], whereas here it exceeds 40.

Qualitative evaluation An illustration of the results of different models on two in-the-wild images is represented in Figure 4.5, along with the associated motion maps. It is clear from the observation of motion maps that ACTOR produces very little movement. HiT-DVAE sequences are likewise almost static, and start drifting after 40 time steps. SUHMo sequences remain sharp after 100 time steps, suggesting a very limited error accumulation. These results suggest that despite many similarities in the addressed problems, current human pose prediction models cannot be readily trained on head motion data without

Table 4.2: FVD scores over different sub-sequence lengths, with and without multi-scale window-based discriminator (CONFER). Subscripts indicate the length associated with the metric.

Method	FVD_{10}	FVD_{20}	FVD_{40}
SUHMo-RNN	**28±8**	**35±4**	162±31
w/o multi-scale discriminator	35±8	42±8	**157±21**
SUHMo-Transformer	**34±9**	**40±12**	**175±46**
w/o multi-scale discriminator	57±6	60±12	236±58

suffering a degradation in performance.

An interesting feature of SUHMo is that the joint generation allows to produce diverse outputs given the same reference pose. We illustrate this in Figure 4.6. This is important for many applications that require the ability to generate different outcomes. These results also show that our training strategy is effective to prevent mode collapse.

4.4.4 Ablation study

Multi-scale discriminator To assess the ability of SUHMo to produce realistic patterns over diverse time scales we measure the FVD on motion chunks of 10, 20, and 40 frames, and compare it with a model trained without the window-based multi-scale discriminator (Table 4.2). Both models were trained to generate sequences of 40 frames and therefore perform on par on this duration. The benefit of the window-based multi-scale approach however clearly appears on shorter timescales, indicating a finer modeling of high frequency patterns.

Joint generation and discrimination We tried removing the pair mixing in the generator and the discriminator at turns (Table 4.3). Models trained with a standard marginal discriminator ("One-sample D") fall behind in terms of FVD and FID, respectively for the RNN and the Transformer model. Surprisingly, suppressing the joint generation ("One-sample G") has an even more detrimental effect, visible on the FVD and FID for both models. In addition to its previously known benefits in mode collapse reduction, we ob-

Table 4.3: Two-samples strategy ablation results on CONFER.

SUHMo variant	RNN			Transformer		
Ablation	FVD_{40}	FID	t-FID_{40}	FVD_{40}	FID	t-FID_{40}
Full	**162±31**	**3±0.2**	**61±8**	175±46	**4±0.7**	67±12
One-sample D	226±76	3±1	71±16	**162±36**	7±1	**65±11**
One-sample G	222±23	5±0.7	58±7	237±58	8±2	74±10

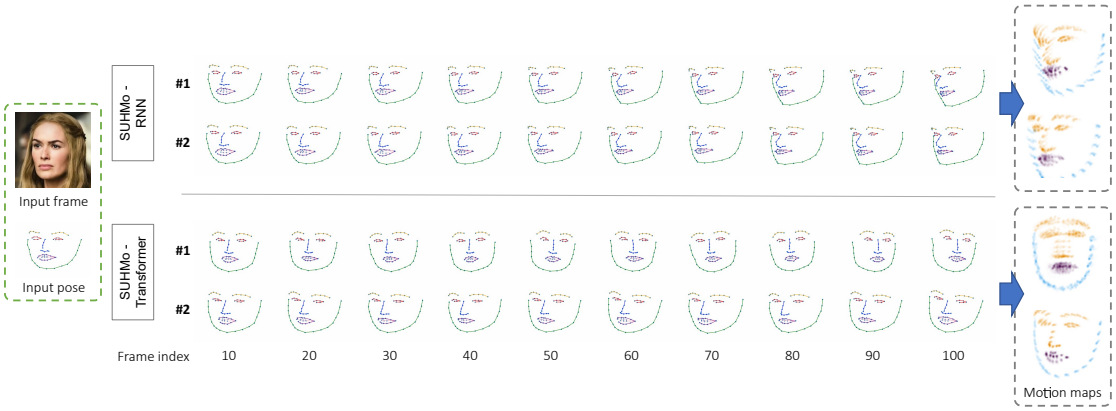


Figure 4.6: Illustrative examples of diverse results given the same reference pose, for both variants of SUHMo. Models are trained on the VoxCeleb2 dataset.

serve that working with pairs of samples also helps improving the overall quality of the generated motion sequences in the unconditional generation setting.

4.5 Conclusion

In this chapter we presented an unconditional head motion generation method able to animate a human face over long sequences from a single initial frame in a semantic space. This method is based on the autoregressive generation of residual displacements, or instantaneous velocities, of pairs of samples, and it is trained using a window-based multi-scale discriminator. We showed that our methodological contributions can accommodate several implementations, consistently outperforming state-of-the-art human pose generation methods and head motion prediction baselines in terms of dynamics quality and pose realism. In the following chapter we extend this method to audio-conditioned talking

head generation, showing that it can improve the realism of head motion together with the audio-visual synchrony over multiple time scales.

CHAPTER 5

A COMPREHENSIVE MULTI-SCALE APPROACH FOR SPEECH AND DYNAMICS SYNCHRONY IN TALKING HEAD GENERATION

Animating still face images given a speech input signal has many practical applications and is therefore an active research topic. However, much of the effort has been put into lip syncing and rendering quality while the generation of natural head motion, let alone the audio-visual correlation between head motion and speech, has often been neglected. In this chapter, we present a multi-scale audio-visual synchrony loss and a multi-scale autoregressive GAN to better handle short- and long-term correlation between speech and the dynamics of the head and lips. In particular, we train a stack of syncer models on multimodal input pyramids and use these models as guidance in a multi-scale generator network to produce audio-aligned motion unfolding over diverse time scales. Following the same approach as before in this thesis, our generator operates in the facial landmark domain, which is a standard low-dimensional head representation. The experiments show significant improvements over the state of the art in head motion dynamics quality and in multi-scale audio-visual synchrony both in the landmark domain and in the image domain. Our code, models and demo are available on the project's GitHub page.[1]

5.1 Introduction

The task of talking face generation, which aims to animate still images from a conditioning audio signal, has seen considerable recent progress. The advent of potent reenactment systems, as that of Siarohin *et al*. [99] or Wang *et al*. [126], and powerful loss functions allowing for a finer correlation between the generated lip motion and the audio input [25] have paved the way for a new state of the art. In both tasks of talking head generation and face reenactment, where lip and head motion are given as a driving video sequence, it is customary to represent face dynamics in a low dimensional space [38, 20, 141, 136, 131, 43, 126, 137, 79]. For this reason recent breakthrough in face reenactment has also benefited the talking head synthesis task. The above approach assumes that image texture and face dynamics can be processed independently, and that all necessary cues to handle the dynamics fit on a low dimensional manifold. It is then a reliable strategy to treat audio-conditioned talking face synthesis as a two-step proce-

[1] https://github.com/LouisBearing/HMo-audio.

dure, where the audio-correlated dynamics are first generated in the intermediate space of an off-the-shelf face reenactment model, which is later used to reconstruct photorealistic video samples [123, 124, 52]. This allows to focus on improving the audio-visual (AV) correlation between the input speech signal and the produced face and lips movements in a much sparser space than that of real-world images.

Nevertheless, synthesising natural-looking head and lip motion sequences adequately correlated with an input audio signal remains a challenging task. In particular, although it has long been known that speech and head motion are tightly associated [127], only recently has this relation attracted the attention of the computer vision community. A likely reason for the difficulty of producing realistic head motion is the lack of an adequate loss function. So far, the most successful strategy to produce synchronized lip movements has relied on the maximization of the cross-modal correlation between short audio and output motion clips, measured by a pre-trained model [25, 142, 88, 86, 129]. This fails, however, to account for lower frequency motion as that of the head which remains quasi-static over the short duration considered, typically of the order of a few hundreds of milliseconds. Surprisingly, there was no attempt to generalize this approach beyond lip synchronization. Neither has possible multi-scale audio-visual correlation been explored in the talking face generation literature. Head motion is often produced through the use of a separate sub-network trained to match the dynamics of a ground truth sequence, which in practice decouples the animation of head and lips.

We argue that to account for motion that unfolds over longer duration such as the head rhythm, a dedicated loss enforcing the synchrony of AV segments of various lengths is needed. We propose to implement this loss using a *pyramid of syncers*, replacing the lip-sync expert of Prajwal *et al*. [88] with a stack of *syncer models* evaluating the correlation between the audio input and the dynamics of the whole face over different time scales. One advantage of this syncer-pyramid loss function is that it allows to produce head and lip movements together; here one may train a single network end-to-end on the dynamics of both head and lips, resulting in overall lighter architecture and training procedure. A natural way to exploit the gradients from the multi-scale AV correlation loss is then

to construct a similar hierarchical structure in the generative model itself. The proposed method, hereafter labelled MS-Sync, is implemented in the landmark domain [14]: for the reasons previously mentioned it is sufficient to parameterize the speech-correlated facial dynamics, which is the focus of the present work. Our generative model, loss functions and most of the metrics used to measure the quality and synchrony of the produced motion therefore apply in this domain. Although, as before, we do not seek to reconstruct face images, several landmark-based reconstruction methods exist [132, 43, 137, 79]. Last, in contrast with the current trend in talking face synthesis, we rely one more time on an autoregressive generative network for its inherent ability to model sequential dependencies, and its flexibility to handle sequences of arbitrary length. To do so, we build on the autoregressive GAN baseline of Chapter 4, and show that the conditioning speech signal has a stabilizing effect that hinders error accumulation on a much longer term than in the unconditional setting. In particular, we demonstrate experimentally that the error drift can be mitigated on test sequences more than five times the length of the training sequences. More importantly, we show that the proposed model, coupled with the multi-scale discriminator of the previous chapter, largely outperforms the state of the art in terms of *multi-scale audio-visual correlation* and head dynamics quality.

The contributions of this chapter's work are:

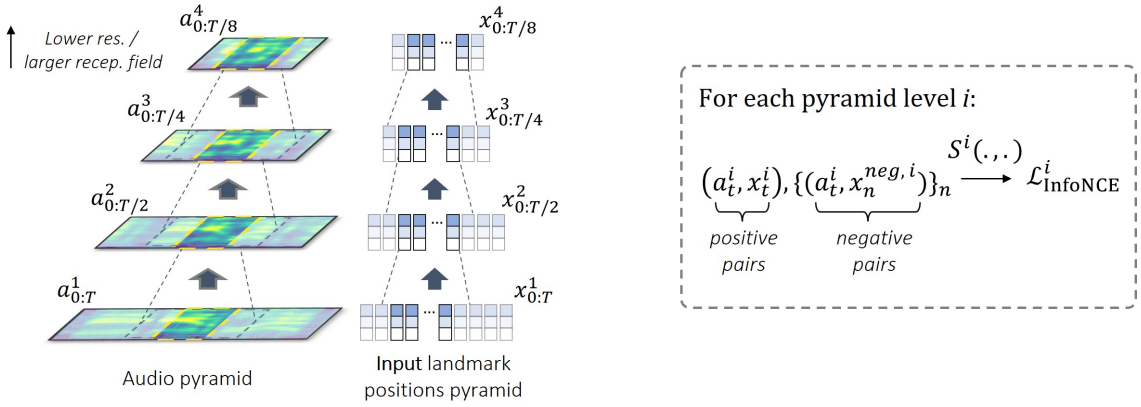
- A multi-scale audio-visual correlation loss based on a pyramid of syncer networks,
- A multi-scale autoregressive GAN for the generation of co-speech head and lip motion in the 2D-landmarks domain with minimal error accumulation,
- Extensive experiments on three datasets that show that our architecture outperforms previous works in terms of both quality of head dynamics and multi-scale AV correlation.

5.2 Related Work

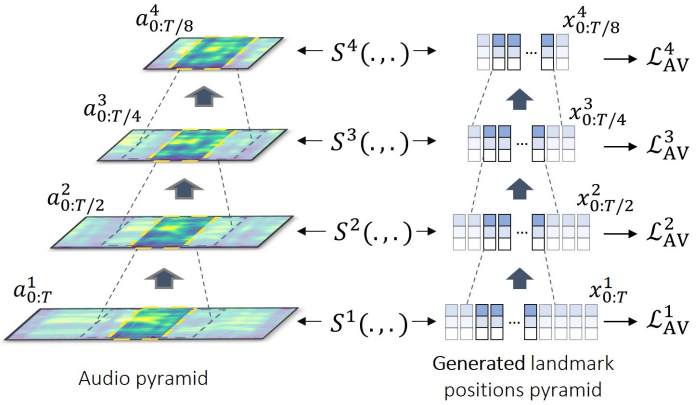
Talking head generation is deeply tied to the work presented in this chapter, and we refer the reader to the Transverse literature review chapter for a discussion of this task. The other closely-related topic of multi-scale data processing in computer vision has been covered in the literature review of Chapter 4. Here we review the background literature on co-speech facial animation.

5.2.1 Learning to align speech and head dynamics

Two trends coexist regarding the synchronization of audio and face dynamics. Originally, learning audio-correlated lip movements was only done with a mean squared error loss to the ground truth sequence [20, 27, 141, 128, 52, 100]. In parallel, following SyncNET [25], the use of contrastive loss variants turned out to be a strong alternative for its effectiveness on cross-modal training tasks [140, 122]. In particular, in Prajwal *et al*. [88] the authors proposed to train a *lip-sync expert* network to regress the cross-modal alignment between short audio and video segments. The expert would later be frozen during the generative model training phase and used as a loss function to enhance output audio-visual alignment. This strategy was later employed successfully in several works [129, 124, 86]. We argue however that the commonly used segment length of 200 ms is insufficient to properly align lower-frequency movements like that of the head, for which several such syncer networks operating on various segment lengths are required. These different syncers should be used on multi-scale feature hierarchies that can be readily computed to align speech and dynamics of various motion frequencies: this approach, which is at the heart of the contributions of this chapter, was never explored in talking head generation so far.



(a) Syncer pyramid training



(b) Multi-scale AV loss

Figure 5.1: (a) A stack of syncer networks S^i are trained on multi-scale positive and negative multimodal pairs using contrastive losses. (b) The syncer models are frozen and used to compute the multi-scale audio visual synchrony loss of the generative model.

5.3 METHOD

Given a set of initial landmark coordinates $x_0 \in \mathbb{R}^{2L}$ (the 2D coordinates of the $L = 68$ landmarks) and a conditioning audio signal $a_{0:T} = (a_0, \dots, a_T) \in \mathbb{R}^{d \times T}$ (here $d = 26$) over T time steps, we aim to produce a sequence of landmark positions $x_{1:T}$ such that the joint distributions over generated and data samples match:

$$p_g(x_{0:T}, a_{0:T}) = p_{\text{data}}(x_{0:T}, a_{0:T}), \quad \forall x_{0:T}, a_{0:T}. \tag{5.1}$$

In this section we describe our procedure to tackle this problem as follows. We start by introducing in 5.3.1 the multi-scale AV synchrony loss which is the major contribution of the present chapter. Then in 5.3.2 we propose a multi-scale generator architecture able to exploit appropriately the devised multi-scale AV loss. Finally section 5.3.3 details our overall training procedure.

5.3.1 MULTI-SCALE AUDIO-VISUAL SYNCHRONY LOSS

The most prominent procedure to align dynamics with speech input relies on the optimization of a correlation score computed on short audio-visual segments of the generated sequence using a pre-trained AV syncer network [88]. Several contrastive loss formulations are possible to train the syncer network, that suppose the maximization of the agreement between in-sync AV segments or positive pairs (a_t, x_t) versus that of out-of-sync or negative pairs. One particularly interesting formulation is the Info Noise Contrastive Estimation loss, that maximizes the mutual information between its two input modalities [85]. Given a set $X = (a_t, x_t, x_1^{neg}, \dots, x_N^{neg})$ containing a positive pair and N negative landmark position segments, this loss writes:

$$\mathcal{L}_{\text{InfoNCE}} = -\mathbb{E}_X \frac{e^{S(a_t, x_t)}}{e^{S(a_t, x_t)} + \sum_{n=1}^{N} e^{S(a_t, x_n^{neg})}}, \tag{5.2}$$

with S the syncer model score function, which is hereafter implemented as the cosine similarity of the outputs from the audio and position embeddings e_a and e_x of the syncer

network:

$$S(a_t, x_t) = \frac{e_a(a_t)^\top e_x(x_t)}{||e_a(a_t)||.||e_x(x_t)||}. \tag{5.3}$$

Following the usual practice, we take a_t and x_t respectively as the MFCC spectrogram and position segment of a 200 ms window centered on time step t. Negative pairs can be indifferently misaligned audio and position segments from the same sequence sample, or segments from different samples, and N is hereafter fixed to 48.

Once trained, the weights of e_a and e_x are frozen and the following term is added to the loss function of the generative model:

$$\mathcal{L}_{\text{AV}} = -\mathbb{E}_{a_t,x_t} S(a_t, x_t), \tag{5.4}$$

where a_t is now part of the conditioning signal and x_t is output by the model.

The above procedure is insufficient when one needs to discover AV correlations over different time scales. One solution consists in building multi-scale representations of the audio-visual inputs and training one syncer network S^i for each level i in the resulting pyramid. The training process of the pyramid of syncers is represented in Figure 5.1 (a). Specifically, the audio and landmark position hierarchies $\{a^i_{0:T/2^{i-1}}\}_i$ and $\{x^i_{0:T/2^{i-1}}\}_i$ are constructed by successive passes through an average pooling operator that blurs and downscales its input by a factor 2, e.g. for positions:

$$x^i_t = \frac{1}{2k+1} \sum_{\tau=-k}^{k} x^{i-1}_{2t+\tau} \tag{5.5}$$

where we choose $k = 3$. The objective is to progressively blur out the highest frequency motion when moving upward in the pyramid, forcing the top level syncers to exploit better the rhythm of the head motion. A total of four syncer networks are trained on the input pyramid following (5.2), input segment duration ranging from the standard 200 ms on the bottom level to 1600 ms at the coarsest scale.

After the training of the pyramid of syncers, all networks S^1 to S^4 are frozen and used to compute the multi-scale audio-visual synchrony loss. The principle of this loss is pre-

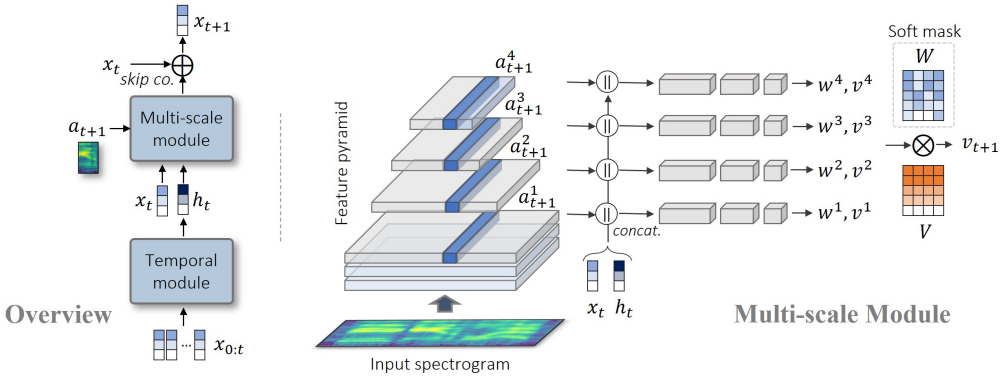


Figure 5.2: Left. Our network is composed of a temporal module, typically a single layer LSTM, and a multi-scale module. Right. Details of the multi-scale module.

sented in Figure 5.1 (b). Similar to the input pyramids used to train the syncer networks, we construct a multi-scale representation of the input speech $a_{0:T}$ and the generated landmark positions $x_{0:T}$. Then for each hierarchy level i one loss term $\mathcal{L}^i_{\text{AV}}$ is computed according to (5.4) using pre-trained syncer S^i. Those terms are then averaged to give the overall multi-scale AV synchrony loss $\mathcal{L}^{\text{MS}}_{\text{AV}}$. To better exploit the effects of this loss, we propose a multi-scale autoregressive generator network that we describe in the following section.

5.3.2 Multi-scale autoregressive generator

Through the multi-scale synchrony loss, the generator receives gradients that push it to produce audio-synced landmark positions over multiple time scales. In this section, we describe the architecture of our generator network, which is itself implemented with a multi-scale structure to better exploit the loss gradients. The overall architecture is described in Figure 5.2.

The proposed generative model is inspired from SUHMo (Chapter 4) which implements an autoregressive model to generate facial landmark velocities. This however requires substantial adaptations to deal with the present multimodal data. Very generally, given landmark positions $x_{0:t}$ until time step t and next frame audio input a_{t+1}, the gener-

ator G produces instantaneous velocities v_{t+1}:

$$v_{t+1} = G(x_{0:t}, a_{t+1}), \tag{5.6}$$

$$x_{t+1} = x_t + v_{t+1}. \tag{5.7}$$

As depicted in Figure 5.2, G is constituted of a temporal module operating on a sequence of landmark positions, and of a multi-scale module that takes the output of the temporal module h_t, the positions x_t and audio a_{t+1} as input to produce v_{t+1}. We implement the multi-scale module as the bottom-up path of a Feature Pyramid Network [68]. Namely, the input spectrogram is processed by several downsampling convolutional layers, producing feature maps $a^1_{0:T}$ to $a^4_{0:T/2^3}$ of the same resolution as those used to compute the AV loss pyramid. Feature maps 2 to 4 are later interpolated back to the length T of the finest map, such that one vector a^i_{t+1} can be extracted from each pyramid level i to produce the next step velocity. Concretely, each vector a^i_{t+1} is concatenated with x_t and h_t and is processed by an independent fully connected branch, the rationale being that processing each input resolution separately would allow the model to produce different motion frequencies.

The outputs of the four branches of the multi-scale generator are merged using a learnable soft spatial mask. Each branch i outputs a velocity vector $v^i \in \mathbb{R}^{2L}$ (note that time index is omitted for the sake of clarity) and a mask vector $w^i \in \mathbb{R}^{2L}$ such that $w^i_{2k-1} = w^i_{2k}, \forall k \leq L$, responsible for enhancing or weakening the contribution of each landmark on the given branch. This is because we expect facial regions to play different roles depending on the scale: the finest resolution branch might emphasize lip landmarks, while at the coarsest scale, more weight may be put on head contour. The output of the multi-scale module finally writes:

$$v_{t+1} = \sum_{i=1}^{4} \Big(\frac{e^{w^i}}{\sum_j e^{w^j}} \Big) v^i \tag{5.8}$$

5.3.3 Overall architecture and training

In addition to the AV synchrony loss $\mathcal{L}_{\text{AV}}^{\text{MS}}$, we make use of the discriminator networks introduced in Chapter 4, that proved effective to train an autoregressive generator on landmark sequences. These consist in one frame discriminator D_f which computes the realism of static landmarks, and two window-based multi-scale networks D_s and D_s^j on sequences. The difference between those is that D_s^j processes pairs of samples to help reducing mode collapse [70]. Although in our audio-conditioned setting mode collapse is at most a minor issue, we found that using this additional loss slightly improves the dynamics quality. Adversarial losses are again implemented with the geometric GAN formulation of Lim & Ye [66]. They are identical to those of the previous chapter, since we found preferable not to condition the discriminators on audio. Nevertheless, we re-write them here for completeness. Given the generated and ground truth landmark position distributions p_g and p_{data}, the generator losses write:

$$\mathcal{L}_{G_f} = -\mathbb{E}_{x_{0:T}\sim p_g}\left[\frac{1}{T}\sum_{t\geq 1} D_f(x_t)\right], \tag{5.9}$$

$$\mathcal{L}_{G_s} = -\mathbb{E}_{x_{0:T}\sim p_g}\left[D_s(x_{0:T})\right], \tag{5.10}$$

$$\mathcal{L}_{G_s^j} = -\mathbb{E}_{x_{0:T}\sim p_g, x'_{0:T}\sim p_g}\left[D_s^j(x_{0:T}, x'_{0:T})\right], \tag{5.11}$$

as for the generic discriminator loss:

$$\mathcal{L}_{D_*} = \mathbb{E}_{x\sim p_g}\left[\max(0, 1 + D_*(x))\right] + \mathbb{E}_{x\sim p_{\text{data}}}\left[\max(0, 1 - D_*(x))\right], \tag{5.12}$$

where D_* is replaced respectively by D_f, D_s and D_s^j and sequences are sampled according to equations 5.9 to 5.11. Additionally, we trained with the weak L_2 reconstruction loss of Chapter 4 but found no significant improvement. Overall, training consists in minimizing alternatively the two following terms:

$$\mathcal{L}_D = \mathcal{L}_{D_f} + \mathcal{L}_{D_s} + \mathcal{L}_{D_s^j}, \tag{5.13}$$

$$\mathcal{L} = \lambda\mathcal{L}_{\text{AV}}^{\text{MS}} + \mathcal{L}_{G_f} + \mathcal{L}_{G_s} + \mathcal{L}_{G_s^j}, \tag{5.14}$$

Table 5.1: Training dataset and head motion preprocessing and generation for the considered methods. The preservation of head translation typically implies re-creating the datasets from the original sources following the strategy in Siarohin *et al.* [99].

Method	Train Dataset	Head transl. in prepro.	Predicted head motion
Wav2Lip [88]	LRS2		
PC-AVS [140]	VoxCeleb2 (I)		
MakeItTalk [141]	VoxCeleb2 (I)		✓
Audio2Head [123]	VoxCeleb2 (II)	✓	✓
OSTF [124]	Obama Weekly Address	✓	✓
MS-Sync (ours)	VoxCeleb2 (II)	✓	✓

Table 5.2: Dynamics quality of rasterized landmarks on VoxCeleb2 (II) test set. All metrics need to be minimized. Bold indicates best score, underline second best. We additionally report the static face Wav2Lip results as reference scores for the reader.

Duration (frames)	40			80					200		
Method	FVD_{40}	FID	$t\text{-}FID_{40}$	FVD_{40}	FID	$t\text{-}FID_{40}$	FVD_{80}	$t\text{-}FID_{80}$	FVD_{40}	FID	$t\text{-}FID_{40}$
Methods that predict head motion											
MakeItTalk [141]	236	4.0	107	234	3.2	101	476	133	224	4.2	114
Audio2Head [123]	406	66.4	109	593	82.5	133	682	149	649	92.5	141
OSTF [124]	113	12.4	**36**	164	25.4	50	249	33	225	30.5	65
MS-Sync-*short*	**105**	3.3	52	126	5.5	47	**239**	41	279	25.2	68
MS-Sync-*long*	134	6.6	42	**104**	6.8	**39**	257	**28**	**144**	8.3	**47**
Methods with fixed head pose											
Wav2Lip [88]	263	**0.9**	167	261	**1.0**	167	557	188	265	**3.2**	182

with $\lambda = 8$ in all experiments.

5.4 Experiments

We conducted three benchmark evaluations to measure the proficiency of our model, assessing respectively head dynamics quality, multi-scale AV synchrony in the landmark domain, and AV synchrony in the image domain.

5.4.1 EXPERIMENTAL PROTOCOL

Datasets. Experiments are conducted on two versions of the VoxCeleb2 dataset [24] with different preprocessing. The first version, labelled VoxCeleb2 (I), follows the standard preprocessing that centers the face in every frame. Second, we use the preprocessing strategy in Siarohin *et al*. [99] to re-generate subsets of respectively ∼18k and 500 short video clips from the original VoxCeleb2 train and test sets. The interest of this preprocessing method is that it keeps the reference frames fixed, thus preserving head motion. We refer to this second version as VoxCeleb2 (II). HDTF dataset [136] contains ∼400 long duration frontal-view videos from political addresses, which despite limited dynamics diversity makes it suitable for AV correlation measurements. Last, we use LRS2 [1], which is preprocessed similarly to VoxCeleb2 (I), to measure the AV synchrony in the image space.

Benchmark models. We compare our method, MS-Sync, with the following prominent speech-driven talking head generation models. Wav2Lip [88] uses a pre-trained lip syncer to learn the AV synchrony, and achieved state-of-the-art performances on the visual dubbing task. However, it only reenacts the lip region and therefore does not produce any head motion. Similarly, PC-AVS [140] produces co-speech talking head videos using a driving head motion sequence, mapping the results directly in image space without any explicit intermediate representation. MakeItTalk [141] was one of the first successful attempts to produce speech-correlated head motion. Its dynamics are learned in the landmark domain on VoxCeleb2 (I), i.e. no head translation was seen at training. Audio2Head [123] and its follow-up model OSTF [124] propose methods to generate vivid dynamics, learning head motion and AV synchrony in a sparse keypoint space using a two-step training procedure. Audio2Head dynamics module is trained on VoxCeleb2 (II), while OSTF is trained on a single identity, namely using Obama Weekly Address dataset. As a noticeable improvement over Audio2Head, in OSTF AV synchrony is controlled with the contrastive loss of Prajwal *et al*. [88]. Information on the different models and training corpora is summarized in Table 5.1.

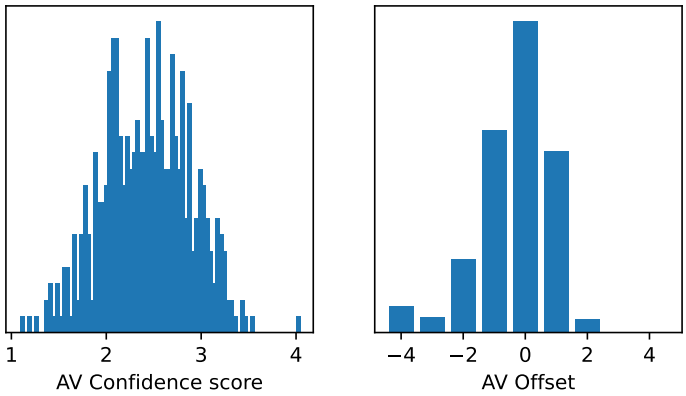


Figure 5.3: Distributions of confidence (AV-Conf$_1$) and offset (AV-Off$_1$, without absolute value) scores measured on VoxCeleb2 (II) test dataset by our metric syncer (equivalent to landmark-domain SyncNET [25]) at the finest time scale.

Training details. The temporal module introduced in Section 5.3 is implemented as a 1-layer LSTM with hidden size 256. All convolutions and fully connected layers are implemented as 1D ConvNeXt blocks (with kernel size 1 for dense layers) [72]. We trained two versions of our model, varying only the training sequence length from 40 to 80 frames, resulting in MS-Sync-*short* and MS-Sync-*long*: the aim is to see how this affects the quality of the produced sequences on various output lengths. Models were trained on VoxCeleb2 (II) for 70k iterations (about 500 epochs) using Adam optimizers with $\beta_1 = 0$ and $\beta_2 = 0.999$ and learning rates 2×10^{-5} and 1×10^{-5} respectively for the generator and the discriminator, after which a decay factor of 0.1 was applied on the learning rates for 5k additional iterations. All audio inputs are sampled at 16 kHz, and to generate the 26-dimensional MFCC spectrogram we used a window size of 400 points and hop size of 160 points.

5.4.2 Dynamics quality

Protocol. The quality of the produced dynamics is evaluated on the 500 videos of VoxCeleb2 (II) test set, which preserve head motion. The Fréchet Inception Distance (FID) is used to measure static face realism, while the Fréchet Video Distance (FVD) and temporal FID (see Chapter 4) metrics measure the distance between the distributions of data and generated motion. The two latter metrics require a fixed sequence length that we set

Table 5.3: Landmark domain multi-scale AV synchrony on VoxCeleb2 (II) test set. A separate syncer model inspired from SyncNet is trained in the landmark domain for each corresponding time scale and used to compute the correlation scores.

Time scale	200 ms (1)		400 ms (2)	
Method	$\lvert\text{AV-Off}_1\rvert \downarrow$	$\text{AV-Conf}_1 \uparrow$	$\lvert\text{AV-Off}_2\rvert \downarrow$	$\text{AV-Conf}_2 \uparrow$
Static	9.45±4.49	0.63±0.23	10.22±4.17	1.07±0.34
Wav2Lip [88]	0.47±0.50	2.42±0.43	0.03±0.51	2.75±0.62
MakeItTalk [141]	1.55±2.75	1.36±0.59	2.48±4.74	1.71±0.70
Audio2Head [123]	2.58±0.51	1.67±0.48	2.41±3.34	1.88±0.64
OSTF [124]	2.13±2.26	1.38±0.50	3.15±4.53	1.62±0.64
MS-Sync-*short*	**0.0±0.0**	**3.37±0.41**	**0.01±0.08**	**3.62±0.52**
MS-Sync-*long*	0.01 ± 0.08	3.26 ± 0.41	**0.01±0.08**	3.56 ± 0.51
Ground truth	0.87±0.93	2.41±0.46	0.52±1.40	2.50±0.62

Time scale	800 ms (3)		1600 ms (4)	
Method	$\lvert\text{AV-Off}_3\rvert \downarrow$	$\text{AV-Conf}_3 \uparrow$	$\lvert\text{AV-Off}_4\rvert \downarrow$	$\text{AV-Conf}_4 \uparrow$
Static	9.01±4.21	1.06±0.35	5.72±4.26	0.99±1.11
Wav2Lip [88]	1.47±3.82	1.76±0.71	3.67±4.22	1.48±1.61
MakeItTalk [141]	4.63±5.81	1.47±0.68	4.02±4.26	1.27±1.42
Audio2Head [123]	3.74±4.89	1.45±0.63	3.43±4.10	1.39±1.43
OSTF [124]	5.47±5.42	1.24±0.50	3.62±4.07	1.40±1.45
MS-Sync-*short*	**0.07±0.67**	2.48 ± 0.74	2.60 ± 3.92	2.18 ± 1.85
MS-Sync-*long*	0.10 ± 0.82	**2.49±0.71**	**1.84 ± 3.38**	**2.20±1.74**
Ground truth	2.39±4.31	1.57±0.67	2.97±3.83	1.69±1.64

to either 40 or 80 frames (equivalent to 1.6 s and 3.2 s at 25 fps), and we refer to the resulting metrics as FVD_{40} (t-FID_{40}) and FVD_{80} (t-FID_{80}), respectively. When generating longer sequences, we measure the FVD_{40}, t-FID_{40} and FID on the last 40 frames. Image-rasterized landmarks are used to compute the metrics (see Figure 5.4).

Results. The results of the dynamics quality evaluations are reported in Table 5.2. MS-Sync-short shows similar FVD and t-FID scores to OSTF but significantly better FID, especially on 40 and 80 frame sequences. Since the faces produced by OSTF are also very sharp, we interpret this result as a hint that this model lacks diversity. Audio2Head suffers from the same limitation to an even greater extent: although visually compelling, the movements it produces are stereotypical and therefore penalized by their too small

Table 5.4: Landmark domain multi-scale AV synchrony on HDTF test set. As above, separate syncer models inspired from SyncNet are trained on each time scale on HDTF dataset and used to compute the correlation scores.

Time scale	200 ms (1)		400 ms (2)	
Method	$\lvert\text{AV-Off}_1\rvert \downarrow$	$\text{AV-Conf}_1 \uparrow$	$\lvert\text{AV-Off}_2\rvert \downarrow$	$\text{AV-Conf}_2 \uparrow$
Static	9.77±4.41	0.63±0.24	9.92±4.22	1.10±0.36
Wav2Lip [88]	**0.29±0.57**	1.84±0.50	0.98±2.62	2.00±0.62
MakeItTalk [141]	1.42±2.51	1.15±0.43	1.53±3.20	1.86±0.59
Audio2Head [123]	1.34±0.80	1.90±0.53	**0.45±2.09**	2.24±0.68
OSTF [124]	0.78±1.89	1.54±0.58	1.71±3.58	1.99±0.73
MS-Sync-*short*	0.76±0.62	**2.68±0.48**	0.93±0.27	**2.80±0.58**
MS-Sync-*long*	0.72 ± 0.62	2.58 ± 0.48	0.87 ± 0.34	2.63 ± 0.61
Ground truth	1.03±0.98	1.95±0.52	1.14±2.21	2.25±0.70
Time scale	800 ms (3)		1600 ms (4)	
Method	$\lvert\text{AV-Off}_3\rvert \downarrow$	$\text{AV-Conf}_3 \uparrow$	$\lvert\text{AV-Off}_4\rvert \downarrow$	$\text{AV-Conf}_4 \uparrow$
Static	9.29±4.14	1.00±0.32	3.75±2.92	0.58±0.48
Wav2Lip [88]	2.32±4.59	1.37±0.52	1.26±2.45	1.07±0.67
MakeItTalk [141]	2.22±4.03	1.57±0.59	1.46±2.28	1.21±0.80
Audio2Head [123]	1.82±3.61	1.59±0.60	1.19±2.05	1.22±0.74
OSTF [124]	3.77±5.15	1.45±0.61	1.78±2.44	1.14±0.82
MS-Sync-*short*	**0.48±2.12**	**2.03±0.67**	**0.86±1.93**	**1.42±0.85**
MS-Sync-*long*	**0.48±2.07**	2.01 ± 0.67	0.96 ± 2.07	1.38 ± 0.85
Ground truth	1.90±3.84	1.58±0.64	1.22±2.06	1.17±0.76

variance in the Fréchet distance calculations. On the other hand, MiT performs well in FID but its dynamics are of noticeable lower quality. Finally, the MS-Sync-long results show that a mere change in training strategy allows to greatly reduce error accumulation over 200 time steps, although it is here at the cost of a slightly lower quality on shorter sequences.

5.4.3 Landmark-domain multi-scale AV synchrony

Protocol. Ideal multi-scale AV synchrony scores should convey how much a model succeeds in exploiting the audio signal to produce motion over diverse time scales. To that end, we resort to audio-visual datasets which preserve motion dynamics, namely VoxCeleb2 (II) and HDTF [136], and carry our evaluations in the landmark domain. We split

Table 5.5: Image domain AV synchrony. † We rescaled the extracted landmarks from PC-AVS that are cropped differently by a factor 0.75 to make it comparable with the original data scale.

Dataset	VoxCeleb2 (I)			
Method	|AV-Off| ↓	AV-Conf ↑	LMD ↓	$\text{LMD}_{\text{front}}$ ↓
Ground truth	1.89±1.92	6.29±1.66	0.0	0.0
GT landmarks + MiT	3.52±4.50	3.55±1.52	1.60±0.30	1.53±0.29
Methods with fixed head pose				
Wav2Lip [88]	2.86±0.34	**8.07±1.33**	**2.90±1.09**	**2.66±0.95**
PC-AVS† [140]	5.18±3.31	3.85±1.55	3.19±1.60	3.00±1.44
Methods that predict head motion				
MakeItTalk [141]	5.23±4.29	3.50±1.49	3.33±1.41	3.07±1.33
Audio2Head [123]	6.83±6.66	2.66±1.38	3.90±1.33	3.61±1.23
OSTF [124]	2.59±4.29	4.12±1.72	3.44±1.45	3.18±1.27
MS-Sync-*short* + MiT	**2.00±2.56**	4.53 ± 1.51	3.15±1.20	2.85±1.11
MS-Sync-*long* + MiT	2.20 ± 2.61	4.35±1.49	3.06 ± 1.17	2.80 ± 1.10
Dataset	LRS2			
Method	|AV-Off| ↓	AV-Conf ↑	LMD ↓	$\text{LMD}_{\text{front}}$ ↓
Ground truth	0.08±0.4	8.36±1.62	0.0	0.0
GT landmarks + MiT	1.72±3.91	4.61±1.70	1.60±0.31	1.54±0.30
Methods with fixed head pose				
Wav2Lip [88]	2.76±0.55	**8.53±1.37**	**2.99±1.05**	2.80 ± 0.89
PC-AVS† [140]	5.48±3.65	4.42±1.65	3.16±1.28	2.96±1.03
Methods that predict head motion				
MakeItTalk [141]	8.43±6.16	2.56±0.96	3.31±1.45	3.08±1.28
Audio2Head [123]	6.78±6.72	3.18±1.43	3.80±1.30	3.60±1.18
OSTF [124]	2.59±4.23	4.56±1.67	3.45±1.46	3.25±1.37
MS-Sync-*short* + MiT	**1.60±2.39**	5.09 ± 1.47	3.07±1.11	2.83±0.83
MS-Sync-*long* + MiT	2.05 ± 3.09	4.83±1.51	3.00 ± 1.18	**2.76±1.02**

HDTF into 291 and 51 train and test identities, and further split the test videos into 1058 80-frame clips. Likewise, measurements on VoxCeleb2 (II) are made on sequences of 80 frames. Pyramids of AV syncers dedicated to metrics calculation, equivalent to landmark-domain SyncNETs [25] on full faces, are trained beforehand on both datasets. Note that contrary to Section 5.3, we train the syncers with the triplet loss used in Chung & Zisserman [25]. The AV synchrony is evaluated using the absolute value of the audio-visual offset (|AV-Off|) and the confidence score (AV-Conf) introduced in the same paper, and detailed hereafter. Given an input audio sequence $\mathbf{a} = a_{0:T}$ and an output landmark sequence $\mathbf{x} = x_{0:T}$, cross-modal distances $d_\tau(\mathbf{x}, \mathbf{a})$ are computed for certain values of the

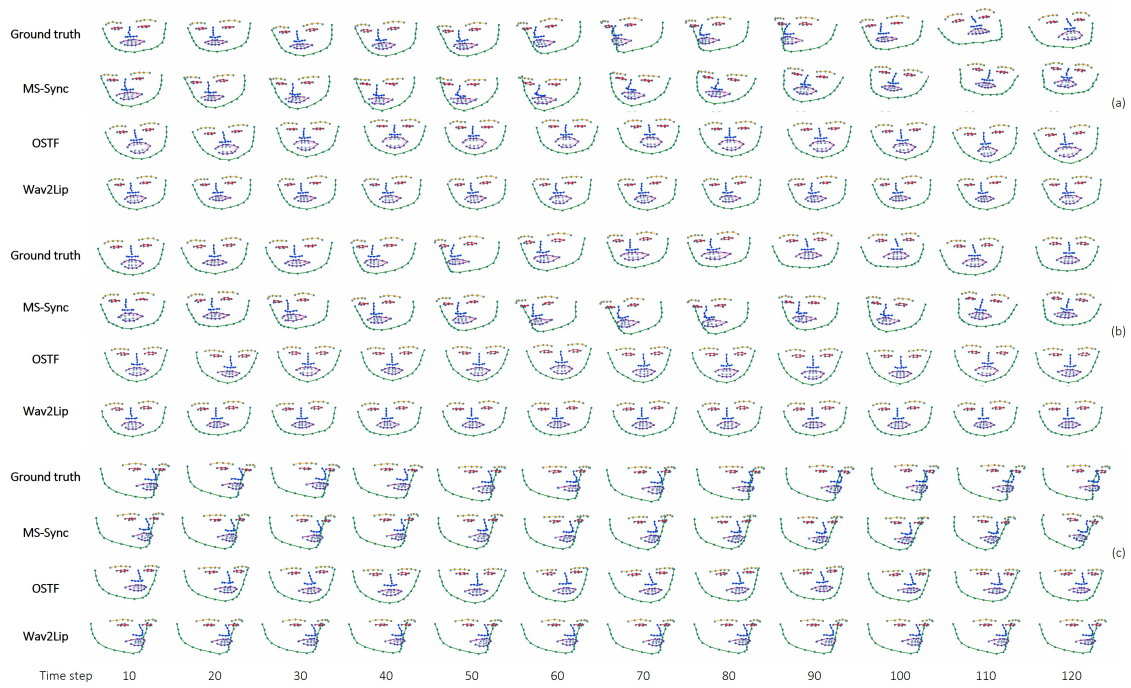


Figure 5.4: Qualitative comparison of the results produced by different methods on three VoxCeleb2 (II) test sequences.

offset τ, thanks to the audio and landmark encoders e_a and e_x of the syncer:

$$d_\tau(\mathbf{x}, \mathbf{a}) = \frac{1}{T+1} \sum_t ||e_x(x_t) - e_a(a_{t+\tau})||_2, \tag{5.15}$$

$$\text{AV-Off}(\mathbf{x}, \mathbf{a}) = \underset{\tau \in [-V_{shift}, V_{shift}]}{\text{argmin}} \; d_\tau(\mathbf{x}, \mathbf{a}), \tag{5.16}$$

$$\text{AV-Conf}(\mathbf{x}, \mathbf{a}) = m(\{d_\tau(\mathbf{x}, \mathbf{a})\}_\tau) - \min_{\tau \in [-V_{shift}, V_{shift}]} d_\tau(\mathbf{x}, \mathbf{a}), \tag{5.17}$$

where m is the median of d_τ values, and V_{shift} is typically equal to 15. Contrary to the standard fashion we then take the absolute value of AV-Off$(\mathbf{x}, \mathbf{a})$ before averaging over the whole dataset as we deem it more informative than averaging over quantities with possible opposite sign, which may falsely provide low offsets. Finally, AV-Conf is also computed as the average over all test sequences. Absolute offset and confidence are measured at four different scales with the different syncers on successively downsampled audio-visual chunks of duration 200 ms, 400 ms, 800 ms, and 1600 ms. Hence an offset of

1 at the finest resolution, sampled at 25 fps, amounts to a misalignment of 40 ms between modalities, whereas at the coarsest scale, this rises to 320 ms. We report in Figure 5.3 the performances of the first level syncer on VoxCeleb2 (II) test set. The distributions of ground truth synchrony scores are closer to Gaussian than perfect Dirac. This is partly because the same audio signal may correspond to more than one facial configuration, and the syncers may not fully grasp this diversity. The synchrony scores reported here should therefore be viewed in light of this assumption: rather than an actual finer AV alignment, results that appear "better" than the ground truth instead correspond to a closer match to the modes of the AV distribution discovered by the syncers.

Results. The AV synchrony scores are reported in Table 5.3 and 5.4 for VoxCeleb2 (II) and HDTF, respectively. We did not include PC-AVS in this section because of distinct cropping strategies producing inconsistent results. Although the loss functions are different, the syncer pyramid used to train the model and the one which serves to compute the metrics were both trained on VoxCeleb2 (II): our model almost perfectly learned to optimize this loss, hence the very strong AV correlation scores. The HDTF results expose the generalization abilities of the different methods. Although Wav2Lip presents the best $|\text{AV-Off}_1|$ at the first scale and Audio2Head the best $|\text{AV-Off}_2|$ at the second scale, MS-Sync possesses the second best scores and largely outperforms all models in terms of AV confidence. What is more, the gap in favor of our model increases at the two coarsest scales, highlighting the effectiveness of the proposed approach to correlate input speech and generated motion on multiple time scales. Last it is noteworthy that although it produces no head motion, Wav2Lip results remain way above the static, uncorrelated boundary even at the coarsest resolution. This means that the mouth region is still partly informative at the top pyramid level, possibly pleading for a stronger blurring strategy.

5.4.4 IMAGE-DOMAIN AV SYNCHRONY

Protocol. In a third batch of experiments, the synchrony is calculated in the image domain, similar to the classical evaluation protocol. To do so, we first map the landmarks

output by our model in the image space using the reenactment system from MakeItTalk (hereafter MiT [141]). Although this leads to blurry results when the pose changes from the initial orientation, we found it sufficient for the sake of AV synchrony measurements. We use two datasets for these experiments: a subset of 2141 videos from the original test set of VoxCeleb2 (I), and LRS2. To cope with the imbalanced duration of VoxCeleb2 videos and keep computation time manageable, we work with the first 40 frames in each clip, while we use the whole LRS2 test set, which contains shorter videos. In addition to the absolute AV offset and confidence score given by SyncNET, we compute the Landmark Distance (LMD [19]), together with a frontalized version $\text{LMD}_{\text{front}}$ that better accounts for face rotation. For a fair comparison, we do not directly use the landmarks produced by MS-Sync to measure the LMD but extract it back from the reenacted video clips; the same procedure is applied to the ground truth landmarks to help assess the effects of each of the previous steps on the metrics.

Results. Although not the primary scope of our study, the landmarks produced by MS-Sync and reenacted with MiT behave surprisingly well in the image domain (Table 5.5). MS-Sync outperforms all other models in terms of AV offset on both VoxCeleb2 (I) and LRS2, simply falling short of Wav2Lip in terms of AV confidence on the two datasets, and of LMD on VoxCeleb2. In particular, MS-Sync performs better than all other models with head motion (and notably MakeItTalk) on all considered metrics. Notice also how the fact that Wav2Lip leaves the whole input face beyond the lips intact seems to bias the calculation of AV confidence in its favor, especially when considering the MiT-reenacted ground truth landmarks. This suggests that SyncNET is sensitive to the image sharpness: the shortcomings of the image reenactment systems probably set limits on the achievable offset and confidence values.

5.4.5 Qualitative results

In Figure 5.4 we present several sequences output by different models and the corresponding ground truth sequences over 120 frames. An examination of these examples shows

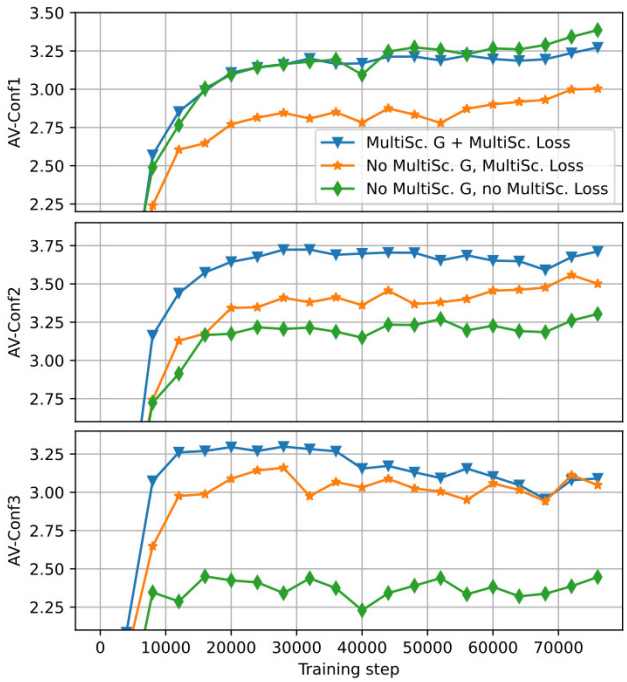


Figure 5.5: Evolution of multi-scale audio-visual confidence over training, measured on VoxCeleb2 (II) validation set. Top is the finest scale, bottom the coarsest.

that mouth closing and opening produced by MS-Sync look correctly aligned with the original, but interestingly this also seems to be the case for head motion although the loss only enforces convergence of distributions. Although the motion produced by OSTF is qualitatively good, it is slightly less diverse and tends to frontalize the face disregarding the original orientation. Wav2Lip, on the other hand, only synchronizes the lips.

5.4.6 Ablation study

In this section, we explore the roles of the multi-scale AV synchrony loss and of the multi-scale generator on the output results, in particular in AV confidence at different resolutions. As can be seen in Figure 5.5 for the evolution of the validation AV confidence along training, almost no difference is visible at the finest resolution between the full MS-Sync model and its single-scale loss, single-scale generator equivalent. However, as expected the confidence of the latter model falls significantly below as one moves upward in the feature pyramid as the loss does not explicitly enforce multi-scale synchrony. It is

possible to circumvent this effect by enabling the multi-scale AV synchrony loss, however, it is clear that if the generator is not itself a multi-scale network, it lacks the capacity to fully exploit the loss, resulting in average performances on every scale.

5.5 Conclusion

The approach presented in this chapter is the first attempt to learn and model audio-visual correlations at multiple scales for talking head generation. This is enabled thanks to a pyramid of syncer models that are trained on hierarchical representations of input audio and landmark position sequences, and then used to compute the loss for the training of the generative model. Importantly, we showed that this model should also be built on a multi-scale backbone, implemented here as a feature pyramid network together with individual branches for each pyramid level that are merged using a soft learnable mask. The very encouraging results of MS-Sync let us foresee numerous applications of similar approaches on other audio-visual generation tasks. One research direction could thus consist in replacing the facial landmarks with other quantities, be it low dimensional keypoints or body joints, or real-world images. Another orthogonal direction may lead to extending the focus to additional cross-modal relationships, such as audio-visual emotions.

CHAPTER 6

CONCLUSION

6.1 CONTRIBUTIONS OF THE THESIS

In this thesis, we devised original ways of training autoregressive generative adversarial networks on three social interaction generation tasks. Autoregressive GANs are an efficient tool to complement or replace the usual maximum likelihood estimation in sequence generation tasks, especially in cases where the latter fails as when data is limited (Chapter 3) or when the action space is continuous (Chapters 4 and 5). The autoregressive formulation is particularly interesting since it allows to produce sequences of arbitrary length at inference, irrespective of the maximum training sequence length. It is also specially adapted to predict residual quantities in continuous sequence generation settings. This approach supposes however to use the own model's predictions *at training*, which albeit suppressing the exposure bias issue, may considerably extend the training duration.

In Chapter 3 we explore a novel approach to discrete interaction generation with an autoregressive GAN and address the previous issue with an original window-based multi-scale discriminator. We show that this implementation of the discriminator alone strongly accelerates training and achieves better end results. We also alleviate the issue of how to differentially sample from a categorical distribution by using a softmax activation with a low-temperature coefficient, effectively turning the discrete action space into a continuous manifold.

In Chapter 4, the focus switches from the action generation in a 14-dimensional discrete action space to the prediction of facial landmark coordinates in a continuous space. This is an especially difficult task as no conditioning is provided other than the initial pose. However, we show that a similar autoregressive GAN model to the one previously proposed, modified to generate residual quantities, performs remarkably well on this problem. We also revisit the roles of the joint generation and discrimination from a theoretical point of view as a mean of mode collapse reduction. These findings result in a general framework that can be implemented indifferently with a Transformer or RNN backbone.

Finally, in Chapter 5 we propose a solution to the problem of syncing both head pose and lip motion with the speech input in talking head generation with a comprehensive

multi-scale autoregressive GAN approach inspired by the work of the previous chapters. This approach combines a multi-scale audio-visual synchrony loss made of a pyramid of audio-visual syncer networks, with a multi-scale generative model built around an audio-input feature pyramid network. The results, in the 2D landmarks domain, outperform all previous talking head generation models in terms of multi-scale audio-visual correlation and head motion quality.

6.2 Prospective research directions

6.2.1 Extensions of presented works

In this section we highlight possible short-term research directions consisting of follow-up extensions of the works presented in this thesis, by means of novel architectures or more global scopes.

Architectures. In light of recent advances in the field, Large Language Models appear as a natural candidate for the task of discrete interaction generation of Chapter 3. The possibility to use prompt-tuning techniques to leverage the representation power of LLMs pre-trained on huge datasets could probably help overcome the limitations posed by current interaction dataset sizes and set a new state-of-the-art. For that, one needs to solve the open problem of how to account for an arbitrary number of interaction participants in the Transformer decoder of LLMs. It is however not clear if LLMs, which are trained to predict discrete tokens with an MLE objective, would benefit the continuous sequence generation tasks considered in Chapters 4 and 5 to the same extent. Other possibilities include diffusion models, some of which have shown promising results in talking head generation. Several strategies emerge where the diffusion process is either used to generate a latent code sequence [98] or to reconstruct the face images via an audio-conditioned U-Net [30, 104]. These existing methods still suffer from several limitations, being either limited to a single identity [30], requiring strong conditioning impairing real-world applications [98] or lacking any treatment of head pose [104], and further research is needed

to overcome these limitations.

Bridging the gap between landmarks and images. A natural follow-up of the results presented in Chapters 4 and 5 consists in enabling the reconstruction of real video images from the output landmark sequences. Although several landmark-to-image methods have been proposed (see for instance Section 4.1), as of the writing of this manuscript those still require additional fine-tuning for unseen identities or do not come with released code and models. This is one limitation of our work, hence the interest of providing additional solutions for image reconstruction. The easiest one would be to extend the generation of landmarks to different latent representations and take advantage of other reenactment models [99]. It is then possible to complement the dynamics loss function with additional reconstruction terms in the image domain to further improve the final rendering [123, 124].

6.2.2 Longer-term challenges

Several methodological challenges remain that need to be addressed to improve the performances of talking head generation models, with possible beneficial implications for other continuous sequence generation tasks.

Improving the quality of unguided talking head generation methods. Despite impressive recent advances, a margin of improvement remains regarding the synthesized video quality and the audio-visual alignment for subject-free (i.e. with inference on persons not seen during training) audio-driven talking head generation methods. To leverage the promising capacity of diffusion models to produce high-quality video sequences without sacrificing alignment between head and lip movements and input speech signal, one needs to abide by a number of previous findings that will probably be part of any final framework. In particular, the generation of the head and lips dynamics must be independent of the visual reconstruction. This makes sense from a computation point of view since dynamics and visual information such as subject identity, colors, texture, or

background are largely independent. Most importantly, the generation of dynamics is a stochastic process following the one-to-many mapping between speech and facial expression, while the visual reconstruction of a target subject given a driving expression is deterministic and has a unique solution. This therefore necessitates a two-stage training which is yet to be proposed within a diffusion framework. It is likely that this approach will also benefit the unconditional head motion generation task described in Chapter 4.

Long sequence generation. A second prospective research direction with perhaps larger implications concerns the generation of long continuous sequences. In this thesis, we presented several methods for the autoregressive generation of motion dynamics over a long duration, however at some point error accumulates which limits the output sequence length. This could be avoided by producing all time steps at once, trading the advantages of autoregressive generation for a guaranteed quality over the entire sequence. However the maximum length will remain limited, this time by the duration of the longest training sample, while the amount of compute will scale with the output length. Therefore it might be preferable to follow an autoregressive VQ-VAE strategy, casting the problem into a discrete sequence generation task by quantizing the facial configuration space into a finite set of adequate dimensions. This approach was previously followed for human motion prediction [101, 135] but also for talking head generation [74]. It is however still unclear what would be the ideal trade-off codebook dimension that would allow to simplify the problem while retaining most of the high complexity of head poses, facial expressions, and lips configuration.

Multi-scale approaches for multimodal social signal generation Finally, we expect multi-scale approaches to be useful in interaction contexts akin to the initial setting of Chapter 3, and particularly for the generation of multimodal social signals. Indeed there is no reason to assume that speaking status and body orientation, to mention only the cues found in the recent literature [53, 91], evolve over the same timescale. And this is certainly not the case for other signals such as the emotional or affective state that one might also want to model. Is is therefore likely that these diverse patterns would be better handled

by either a hierarchical generative model, a multi-scale loss function or a combination of both.

BIBLIOGRAPHY

[1] T. Afouras, J. S. Chung, A. Senior, O. Vinyals, and A. Zisserman. Deep audio-visual speech recognition. In *arXiv:1809.02108*, 2018.

[2] Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. Social LSTM: Human trajectory prediction in crowded spaces. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 961–971, 2016.

[3] Xavier Alameda-Pineda, Jacopo Staiano, Ramanathan Subramanian, Ligia Batrinca, Elisa Ricci, Bruno Lepri, Oswald Lanz, and Nicu Sebe. Salsa: A novel dataset for multimodal group behavior analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(8):1707–1720, 2015.

[4] Sadegh Aliakbarian, Fatemeh Saleh, Lars Petersson, Stephen Gould, and Mathieu Salzmann. Contextually plausible and diverse 3d human motion prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11333–11342, 2021.

[5] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 214–223. PMLR, 06–11 Aug 2017.

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

[7] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *International Conference on Learning Representations (ICLR)*, 2015.

[8] Emad Barsoum, John Kender, and Zicheng Liu. Hp-gan: Probabilistic 3d human motion prediction via gan. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 1418–1427, 2018.

[9] Federico Bartoli, Giuseppe Lisanti, Lamberto Ballan, and Alberto Del Bimbo. Context-aware trajectory prediction. In *International Conference on Pattern Recognition (ICPR)*, pages 1941–1946. IEEE, 2018.

[10] Xiaoyu Bie, Wen Guo, Simon Leglaive, Lauren Girin, Francesc Moreno-Noguer, and Xavier Alameda-Pineda. Hit-dvae: Human motion generation via hierarchical transformer dynamical vae. *arXiv preprint arXiv:2204.01565*, 2022.

[11] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis, 2019.

[12] Tim Brooks, Janne Hellsten, Miika Aittala, Ting-Chun Wang, Timo Aila, Jaakko Lehtinen, Ming-Yu Liu, Alexei Efros, and Tero Karras. Generating long videos of dynamic scenes. *Advances in Neural Information Processing Systems*, 35:31769–31781, 2022.

[13] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[14] Adrian Bulat and Georgios Tzimiropoulos. How far are we from solving the 2d & 3d face alignment problem? (and a dataset of 230,000 3d facial landmarks). In *International Conference on Computer Vision*, 2017.

[15] Laura Cabrera-Quiros, Andrew Demetriou, Ekin Gedik, Leander van der Meij, and Hayley Hung. The matchnmingle dataset: a novel multi-sensor resource for the

analysis of social interactions and group dynamics in-the-wild during free-standing conversations and speed dates. *IEEE Transactions on Affective Computing*, 2018.

[16] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017.

[17] Tong Che, Yanran Li, Athul Paul Jacob, Yoshua Bengio, and Wenjie Li. Mode regularized generative adversarial networks. *arXiv preprint arXiv:1612.02136*, 2016.

[18] Lele Chen, Guofeng Cui, Celong Liu, Zhong Li, Ziyi Kou, Yi Xu, and Chenliang Xu. Talking-head generation with rhythmic head motion. In *European Conference on Computer Vision*, pages 35–51. Springer, 2020.

[19] Lele Chen, Zhiheng Li, Ross K Maddox, Zhiyao Duan, and Chenliang Xu. Lip movements generation at a glance. In *Proceedings of the European conference on computer vision (ECCV)*, pages 520–535, 2018.

[20] Lele Chen, Ross K Maddox, Zhiyao Duan, and Chenliang Xu. Hierarchical cross-modal talking face generation with dynamic pixel-wise loss. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7832–7841, 2019.

[21] Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *International conference on machine learning*, pages 1691–1703. PMLR, 2020.

[22] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. *Advances in neural information processing systems*, 29, 2016.

[23] Yizhou Chen, Xu-Hua Yang, Zihan Wei, Ali Asghar Heidari, Nenggan Zheng, Zhicheng Li, Huiling Chen, Haigen Hu, Qianwei Zhou, and Qiu Guan. Genera-

tive adversarial networks in medical image augmentation: A review. *Computers in Biology and Medicine*, 144:105382, 2022.

[24] J. S. Chung, A. Nagrani, and A. Zisserman. Voxceleb2: Deep speaker recognition. In *INTERSPEECH*, 2018.

[25] Joon Son Chung and Andrew Zisserman. Out of time: automated lip sync in the wild. In *Computer Vision–ACCV 2016 Workshops: ACCV 2016 International Workshops, Taipei, Taiwan, November 20-24, 2016, Revised Selected Papers, Part II 13*, pages 251–263. Springer, 2017.

[26] Jade Copet, Felix Kreuk, Itai Gat, Tal Remez, David Kant, Gabriel Synnaeve, Yossi Adi, and Alexandre Défossez. Simple and controllable music generation. *arXiv preprint arXiv:2306.05284*, 2023.

[27] Dipanjan Das, Sandika Biswas, Sanjana Sinha, and Brojeshwar Bhowmick. Speech-driven facial animation using cascaded gans for learning of motion and texture. In *European conference on computer vision*, pages 408–424. Springer, 2020.

[28] Emily L Denton, Soumith Chintala, Rob Fergus, et al. Deep generative image models using a laplacian pyramid of adversarial networks. *Advances in neural information processing systems*, 28, 2015.

[29] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, 2019.

[30] Chenpng Du, Qi Chen, Tianyu He, Xu Tan, Xie Chen, Kai Yu, Sheng Zhao, and Jiang Bian. Dae-talker: High fidelity speech-driven talking face generation with diffusion autoencoder. *arXiv preprint arXiv:2303.17550*, 2023.

[31] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12873–12883, 2021.

[32] Yingruo Fan, Zhaojiang Lin, Jun Saito, Wenping Wang, and Taku Komura. Faceformer: Speech-driven 3d facial animation with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18770–18780, 2022.

[33] Tharindu Fernando, Simon Denman, Sridha Sridharan, and Clinton Fookes. Soft+ hardwired attention: An LSTM framework for human trajectory prediction and abnormal event detection. *Neural networks*, 108:466–478, 2018.

[34] Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. Convolutional sequence to sequence learning. *Proceedings of the International Conference on Machine Learning (ICML)*, pages 1243–1252, 2017.

[35] Christos Georgakis, Yannis Panagakis, Stefanos Zafeiriou, and Maja Pantic. The conflict escalation resolution (confer) database. *Image and Vision Computing*, 65:37–48, 2017.

[36] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 2672–2680, 2014.

[37] David Greenwood, Stephen Laycock, and Iain Matthews. Predicting head pose in dyadic conversation. In *Intelligent Virtual Agents: 17th International Conference, IVA 2017, Stockholm, Sweden, August 27-30, 2017, Proceedings 17*, pages 160–169. Springer, 2017.

[38] David Greenwood, Iain Matthews, and Stephen Laycock. Joint learning of facial expression and head pose from speech. Interspeech, 2018.

[39] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. *Advances in neural information processing systems*, 30, 2017.

[40] Jiaxian Guo, Sidi Lu, Han Cai, Weinan Zhang, Yong Yu, and Jun Wang. Long text generation via adversarial training with leaked information. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[41] Yudong Guo, Keyu Chen, Sen Liang, Yong-Jin Liu, Hujun Bao, and Juyong Zhang. Ad-nerf: Audio driven neural radiance fields for talking head synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5784–5794, 2021.

[42] Agrim Gupta, Justin Johnson, Li Fei-Fei, Silvio Savarese, and Alexandre Alahi. Social gan: Socially acceptable trajectories with generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2255–2264, 2018.

[43] Sungjoo Ha, Martin Kersner, Beomsu Kim, Seokjun Seo, and Dongyoung Kim. Marionette: Few-shot face reenactment preserving identity of unseen targets. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 10893–10900, 2020.

[44] 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. *Advances in Neural Information Processing Systems (NeurIPS)*, 30:6626–6637, 2017.

[45] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *International conference on learning representations*, 2016.

[46] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[47] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[48] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *International Conference on Machine Learning*, pages 2790–2799. PMLR, 2019.

[49] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

[50] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning (ICML)*, pages 448–456, 2015.

[51] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1125–1134, 2017.

[52] Xinya Ji, Hang Zhou, Kaisiyuan Wang, Qianyi Wu, Wayne Wu, Feng Xu, and Xun Cao. Eamm: One-shot emotional talking face via audio-based emotion-aware motion model. *arXiv preprint arXiv:2205.15278*, 2022.

[53] Hanbyul Joo, Tomas Simon, Mina Cikara, and Yaser Sheikh. Towards social artificial intelligence: Nonverbal social signal prediction in a triadic interaction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10873–10883, 2019.

[54] Sai Shashank Kalakonda, Shubh Maheshwari, and Ravi Kiran Sarvadevabhatla. Action-gpt: Leveraging large-scale language models for improved and generalized zero shot action generation. *arXiv preprint arXiv:2211.15603*, 2022.

[55] Tero Karras, Timo Aila, Samuli Laine, Antti Herva, and Jaakko Lehtinen. Audio-driven facial animation by joint end-to-end learning of pose and emotion. *ACM Transactions on Graphics (TOG)*, 36(4):1–12, 2017.

[56] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019.

[57] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2015.

[58] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *2nd International Conference on Learning Representations (ICLR)*, 2014.

[59] Jean Kossaifi, Robert Walecki, Yannis Panagakis, Jie Shen, Maximilian Schmitt, Fabien Ringeval, Jing Han, Vedhas Pandit, Antoine Toisoul, Björn Schuller, et al. Sewa db: A rich database for audio-visual emotion and sentiment research in the wild. *IEEE transactions on pattern analysis and machine intelligence*, 43(3):1022–1040, 2019.

[60] Maria Koutsombogera and Carl Vogel. Modeling collaborative multimodal behavior in group dialogues: The multisimo corpus. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Paris, France, may 2018. European Language Resources Association (ELRA).

[61] Kundan Kumar, Rithesh Kumar, Thibault de Boissiere, Lucas Gestin, Wei Zhen Teoh, Jose Sotelo, Alexandre de Brébisson, Yoshua Bengio, and Aaron C Courville. Melgan: Generative adversarial networks for conditional waveform synthesis. *Advances in neural information processing systems*, 32, 2019.

[62] Jogendra Nath Kundu, Maharshi Gor, and R Venkatesh Babu. Bihmp-gan: Bidirectional 3d human motion prediction gan. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 8553–8560, 2019.

[63] Namhoon Lee, Wongun Choi, Paul Vernaza, Christopher B Choy, Philip HS Torr, and Manmohan Chandraker. Desire: Distant future prediction in dynamic scenes with interacting agents. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 336–345, 2017.

[64] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*, 2021.

[65] Jiwei Li, Will Monroe, Tianlin Shi, Sébastien Jean, Alan Ritter, and Dan Jurafsky. Adversarial learning for neural dialogue generation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (CEMNLP)*, pages 2157–2169, September 2017.

[66] Jae Hyun Lim and Jong Chul Ye. Geometric gan. *arXiv preprint arXiv:1705.02894*, 2017.

[67] Kevin Lin, Dianqi Li, Xiaodong He, Zhengyou Zhang, and Ming-Ting Sun. Adversarial ranking for language generation. *Advances in Neural Information Processing Systems (NeurIPS)*, 30:3155–3165, 2017.

[68] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

[69] Xiao Lin and Mohamed R Amer. Human motion modeling using dvgans. *arXiv preprint arXiv:1804.10652*, 2018.

[70] Zinan Lin, Ashish Khetan, Giulia Fanti, and Sewoong Oh. Pacgan: The power of two samples in generative adversarial networks. *Advances in neural information processing systems*, 31, 2018.

[71] Ming-Yu Liu, Thomas Breuel, and Jan Kautz. Unsupervised image-to-image translation networks. *Advances in neural information processing systems*, 30, 2017.

[72] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11976–11986, 2022.

[73] Zhengxiong Luo, Dayou Chen, Yingya Zhang, Yan Huang, Liang Wang, Yujun Shen, Deli Zhao, Jingren Zhou, and Tieniu Tan. Videofusion: Decomposed diffu-

sion models for high-quality video generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10209–10218, 2023.

[74] Rayhane Mama, Marc S Tyndel, Hashiam Kadhim, Cole Clifford, and Ragavan Thurairatnam. Nwt: towards natural audio-to-video generation with representation learning. *arXiv preprint arXiv:2106.04283*, 2021.

[75] Wei Mao, Miaomiao Liu, Mathieu Salzmann, and Hongdong Li. Learning trajectory dependencies for human motion prediction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9489–9497, 2019.

[76] Xudong Mao, Qing Li, Haoran Xie, Raymond Y. K. Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks, 2017.

[77] Julieta Martinez, Michael J Black, and Javier Romero. On human motion prediction using recurrent neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2891–2900, 2017.

[78] Iain McCowan, Jean Carletta, Wessel Kraaij, Simone Ashby, S Bourban, M Flynn, M Guillemot, Thomas Hain, J Kadlec, Vasilis Karaiskos, et al. The ami meeting corpus. In *Proceedings of the 5th international conference on methods and techniques in behavioral research*, volume 88, page 100. Citeseer, 2005.

[79] Moustafa Meshry, Saksham Suri, Larry S Davis, and Abhinav Shrivastava. Learned spatial representations for few-shot talking-head synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13829–13838, 2021.

[80] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. *International Conference on Learning Representations (ICLR)*, 2018.

[81] Takeru Miyato and Masanori Koyama. cgans with projection discriminator. *International Conference on Learning Representations (ICLR)*, 2018.

[82] Max Morrison, Rithesh Kumar, Kundan Kumar, Prem Seetharaman, Aaron Courville, and Yoshua Bengio. Chunked autoregressive GAN for conditional waveform synthesis. In *International Conference on Learning Representations*, 2022.

[83] Augustus Odena, Christopher Olah, and Jonathon Shlens. Conditional image synthesis with auxiliary classifier gans. In *International conference on machine learning*, pages 2642–2651. PMLR, 2017.

[84] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, 2016.

[85] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[86] Se Jin Park, Minsu Kim, Joanna Hong, Jeongsoo Choi, and Yong Man Ro. Synctalkface: Talking face generation with precise lip-syncing via audio-lip memory. In *36th AAAI Conference on Artificial Intelligence (AAAI 22)*. Association for the Advancement of Artificial Intelligence, 2022.

[87] Mathis Petrovich, Michael J Black, and Gül Varol. Action-conditioned 3d human motion synthesis with transformer vae. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10985–10995, 2021.

[88] KR Prajwal, Rudrabha Mukhopadhyay, Vinay P Namboodiri, and CV Jawahar. A lip sync expert is all you need for speech to lip generation in the wild. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 484–492, 2020.

[89] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

[90] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.

[91] Chirag Raman, Hayley Hung, and Marco Loog. Social processes: Self-supervised meta-learning over conversational groups for forecasting nonverbal social cues. In *European Conference on Computer Vision*, pages 639–659. Springer, 2022.

[92] Ali Razavi, Aaron Van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. *Advances in neural information processing systems*, 32, 2019.

[93] Yurui Ren, Ge Li, Yuanqi Chen, Thomas H Li, and Shan Liu. Pirenderer: Controllable portrait image generation via semantic neural rendering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13759–13768, 2021.

[94] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[95] Amir Sadeghian, Vineet Kosaraju, Ali Sadeghian, Noriaki Hirose, Hamid Rezatofighi, and Silvio Savarese. Sophie: An attentive gan for predicting paths compliant to social and physical constraints. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1349–1358, 2019.

[96] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *Advances in Neural Information Processing Systems (NeurIPS)*, 29:2234–2242, 2016.

[97] Navyata Sanghvi, Ryo Yonetani, and Kris Kitani. Mgpi: A computational model of multiagent group perception and interaction. In *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*, AAMAS '20,

page 1196–1205, Richland, SC, 2020. International Foundation for Autonomous Agents and Multiagent Systems.

[98] Shuai Shen, Wenliang Zhao, Zibin Meng, Wanhua Li, Zheng Zhu, Jie Zhou, and Jiwen Lu. Difftalk: Crafting diffusion models for generalized audio-driven portraits animation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1982–1991, 2023.

[99] Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe. First order motion model for image animation. *Advances in Neural Information Processing Systems*, 32, 2019.

[100] Sanjana Sinha, Sandika Biswas, Ravindra Yadav, and Brojeshwar Bhowmick. Emotion-controllable generalized talking face generation. *arXiv preprint arXiv:2205.01155*, 2022.

[101] Li Siyao, Weijiang Yu, Tianpei Gu, Chunze Lin, Quan Wang, Chen Qian, Chen Change Loy, and Ziwei Liu. Bailando: 3d dance generation by actor-critic gpt with choreographic memory. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11050–11059, 2022.

[102] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. PMLR, 2015.

[103] Yang Song, Jingwen Zhu, Dawei Li, Xiaolong Wang, and Hairong Qi. Talking face generation by conditional recurrent adversarial network. *arXiv preprint arXiv:1804.04786*, 2018.

[104] Michał Stypułkowski, Konstantinos Vougioukas, Sen He, Maciej Zieba, Stavros Petridis, and Maja Pantic. Diffused heads: Diffusion models beat gans on talking-face generation. *arXiv preprint arXiv:2301.03396*, 2023.

[105] Supasorn Suwajanakorn, Steven M Seitz, and Ira Kemelmacher-Shlizerman. Syn-

thesizing obama: learning lip sync from audio. *ACM Transactions on Graphics (ToG)*, 36(4):1–13, 2017.

[106] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2818–2826, 2016.

[107] Mingxing Tan, Ruoming Pang, and Quoc V Le. Efficientdet: Scalable and efficient object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10781–10790, 2020.

[108] Stephanie Tan, David MJ Tax, and Hayley Hung. Multimodal joint head orientation estimation in interacting groups via proxemics and interaction dynamics. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 5(1):1–22, 2021.

[109] Sarah Taylor, Taehwan Kim, Yisong Yue, Moshe Mahler, James Krahe, Anastasio Garcia Rodriguez, Jessica Hodgins, and Iain Matthews. A deep learning approach for generalized speech animation. *ACM Transactions on Graphics (TOG)*, 36(4):1–11, 2017.

[110] Lucas Theis and Matthias Bethge. Generative image modeling using spatial lstms. *Advances in neural information processing systems*, 28, 2015.

[111] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[112] Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges. *arXiv preprint arXiv:1812.01717*, 2018.

[113] Aäron Van Den Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel recurrent neural networks. In *International conference on machine learning*, pages 1747–1756. PMLR, 2016.

[114] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

[115] Pavan Vasishta, Dominique Vaufreydaz, and Anne Spalanzani. Building Prior Knowledge: A Markov Based Pedestrian Prediction Model Using Urban Environmental Data. In *Proceedings of the International Conference on Control, Automation, Robotics and Vision (ICARCV)*, pages 1–12, 2018.

[116] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[117] Alexander Sasha Vezhnevets, Simon Osindero, Tom Schaul, Nicolas Heess, Max Jaderberg, David Silver, and Koray Kavukcuoglu. Feudal networks for hierarchical reinforcement learning. *Proceedings of the International Conference on Machine Learning (ICML)*, 2017.

[118] Ruben Villegas, Jimei Yang, Yuliang Zou, Sungryull Sohn, Xunyu Lin, and Honglak Lee. Learning to generate long-term future via hierarchical prediction. In *international conference on machine learning*, pages 3560–3569. PMLR, 2017.

[119] Alessandro Vinciarelli, Maja Pantic, and Hervé Bourlard. Social signal processing: Survey of an emerging domain. *Image and vision computing*, 27(12):1743–1759, 2009.

[120] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Generating videos with scene dynamics. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 613–621, 2016.

[121] Konstantinos Vougioukas, Stavros Petridis, and Maja Pantic. Realistic speech-

driven facial animation with gans. *International Journal of Computer Vision*, 128(5):1398–1413, 2020.

[122] Duomin Wang, Yu Deng, Zixin Yin, Heung-Yeung Shum, and Baoyuan Wang. Progressive disentangled representation learning for fine-grained controllable talking head synthesis. *arXiv preprint arXiv:2211.14506*, 2022.

[123] Suzhe Wang, Lincheng Li, Yu Ding, Changjie Fan, and Xin Yu. Audio2head: Audio-driven one-shot talking-head generation with natural head motion. In *IJCAI*, 2021.

[124] Suzhen Wang, Lincheng Li, Yu Ding, and Xin Yu. One-shot talking face generation from single-speaker audio-visual correlation learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 2531–2539, 2022.

[125] 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 *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8798–8807, 2018.

[126] Ting-Chun Wang, Arun Mallya, and Ming-Yu Liu. One-shot free-view neural talking-head synthesis for video conferencing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10039–10049, 2021.

[127] Hani C Yehia, Takaaki Kuratate, and Eric Vatikiotis-Bateson. Linking facial animation, head motion and speech acoustics. *Journal of phonetics*, 30(3):555–568, 2002.

[128] Ran Yi, Zipeng Ye, Juyong Zhang, Hujun Bao, and Yong-Jin Liu. Audio-driven talking face video generation with learning-based personalized head pose. *arXiv preprint arXiv:2002.10137*, 2020.

[129] Fei Yin, Yong Zhang, Xiaodong Cun, Mingdeng Cao, Yanbo Fan, Xuan Wang, Qingyan Bai, Baoyuan Wu, Jue Wang, and Yujiu Yang. Styleheat: One-shot high-

resolution editable talking face generation via pretrained stylegan. *arXiv preprint arXiv:2203.04036*, 2022.

[130] Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. Seqgan: Sequence generative adversarial nets with policy gradient. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2017.

[131] Egor Zakharov, Aleksei Ivakhnenko, Aliaksandra Shysheya, and Victor Lempitsky. Fast bi-layer neural synthesis of one-shot realistic head avatars. In *European Conference on Computer Vision*, pages 524–540. Springer, 2020.

[132] Egor Zakharov, Aliaksandra Shysheya, Egor Burkov, and Victor Lempitsky. Few-shot adversarial learning of realistic neural talking head models. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 9459–9468, 2019.

[133] Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attention generative adversarial networks. In *International conference on machine learning*, pages 7354–7363. PMLR, 2019.

[134] Xiang Zhang and Yann LeCun. Text understanding from scratch. *arXiv preprint arXiv:1502.01710*, 2015.

[135] Yaqi Zhang, Di Huang, Bin Liu, Shixiang Tang, Yan Lu, Lu Chen, Lei Bai, Qi Chu, Nenghai Yu, and Wanli Ouyang. Motiongpt: Finetuned llms are general-purpose motion generators. *arXiv preprint arXiv:2306.10900*, 2023.

[136] Zhimeng Zhang, Lincheng Li, Yu Ding, and Changjie Fan. Flow-guided one-shot talking face generation with a high-resolution audio-visual dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3661–3670, 2021.

[137] Ruiqi Zhao, Tianyi Wu, and Guodong Guo. Sparse to dense motion transfer for face image animation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1991–2000, 2021.

[138] Tianyang Zhao, Yifei Xu, Mathew Monfort, Wongun Choi, Chris Baker, Yibiao Zhao, Yizhou Wang, and Ying Nian Wu. Multi-agent tensor fusion for contextual trajectory prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12126–12134, 2019.

[139] Hang Zhou, Yu Liu, Ziwei Liu, Ping Luo, and Xiaogang Wang. Talking face generation by adversarially disentangled audio-visual representation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 9299–9306, 2019.

[140] Hang Zhou, Yasheng Sun, Wayne Wu, Chen Change Loy, Xiaogang Wang, and Ziwei Liu. Pose-controllable talking face generation by implicitly modularized audio-visual representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4176–4186, 2021.

[141] Yang Zhou, Xintong Han, Eli Shechtman, Jose Echevarria, Evangelos Kalogerakis, and Dingzeyu Li. Makelttalk: speaker-aware talking-head animation. *ACM Transactions on Graphics (TOG)*, 39(6):1–15, 2020.

[142] Hao Zhu, Huaibo Huang, Yi Li, Aihua Zheng, and Ran He. Arbitrary talking face generation via attentional audio-visual coherence learning. *arXiv preprint arXiv:1812.06589*, 2018.