Speech-conditioned Face Generation with Deep Adversarial Networks

Francisco Roldan Snchez

Abstract

Image synthesis have been a trending task for the AI community in recent years. Many works have shown the potential of Generative Adversarial Networks (GANs) to deal with tasks such as text or audio to image synthesis. In particular, recent advances in deep learning using audio have inspired many works involving both visual and auditory information. In this work we propose a face synthesis method using audio and/or language representations as inputs. Furthermore, a dataset which relates speech utterances with a face and an identity has been built, fitting for other tasks apart from face synthesis such as speaker recognition or voice conversion.

Index Terms

deep learning, adversarial learning, image synthesis, face synthesis, face synthesis, computer vision

I. Introduction

WE humans are capable to identify other people by their different traits. The simplest of those traits is probably the name, since we synthesize a person with just a single word. However, we can recognize other people even if we lack this information by using inputs received from the environment, that is, visual, auditory or even olfactory information. Furthermore, when we lack some of this traits we tend to imagine how they are. For example, when listening to music and hearing a high-pitched voice we tend to relate it to a female, and sometimes we even imagine the ethnicity just from the voice.

This work's motivation is based on two facts. First, image generation has caught the attention of AI community during the latest years. Thanks to Variational Auto-Encoders (VAE) [1] and Generative Adversarial Networks (GAN) [2] we are now able to generate realistic synthetic images. Second, there is a large amount of data on the cloud of people talking in front of a camera, therefore we can relate a speech frame to a facial expression. Combining these two facts, in this work we aim to generate realistic images of faces using just auditory information. Following this motivation, the work itself is divided into two different blocks: a first one related to the data collection and a second one, with the generative image modeling.

Regarding to the data collection, we take advantage of the audiovisual content offered by the so-called *youtubers*, who usually use high quality recording hardware in suitable recording environments, which leads to a clean recording signal. After some pre-processing steps which are detailed in Section III, we use this data to train a generative model which takes as input raw speech frames and have as output images of faces. Moreover, we also explore different combinations of data domains as input for the network.

Until now, the typical solution to this kind of task was extracting an audio embedding and a representation of the target identity, normally features extracted from a CNN using as input a random image of the target, and decode the combination of both forming an image, as shown in Figure 1.

A. Project Goals and Requirements

Different goals have been defined in order to set a roadmap for the project:

- Collect data from *youtubers*' videos to build a dataset of cropped faces with its associated speech.
- Explore different techniques to generate realistic images given its description.
- Explore different techniques to generate realistic images of faces given the name of the person.
- Explore different techniques to generates realistic images of faces given a speech frame.

Furthermore, some requirements have been specified:

- Use PyTorch as deep learning framework.
- Build the project using programming best practices in order to open-source it.

Advisor 1: Xavier Gir-i-Nieto, Grup de Processat de Imatge (GPI), UPC
Advisor 2: Santiago Pascual de la Puente, Language and Speech Technologies and Applications (TALP), UPC
Advisor 3: Amaia Salvador Aguilera, Grup de Procesat de Imatge (GPI), UPC
Advisor 4: Kevin McGuinness, Insight Centre for Data Analytics, DCU
Thesis dissertation submitted: July 2018

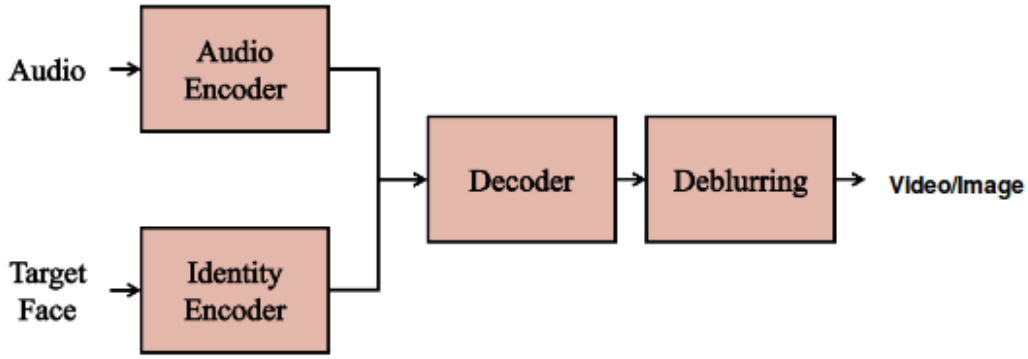

Fig. 1. High level representation of an audio to image synthesizer. Image from [9]

B. Work Methodology

This work is the result of a collaboration of different research groups from Universitat Politcnica de Catalunya (UPC) and the Insight Centre for Data Analytics from the Dublin City University (DCU), having a regular weekly meeting to discuss decisions to be made.This meting has been complemented with a weekly seminar with other students developing their bachelor, master or Ph.D thesis at GPI to present our research and discuss about topics related to them.

The project has been divided into different Work Packages (WP):

- WP1: State-of-the-art review
- WP2: Data collection
- WP3: Image generation from descriptions
- WP4: Faces generation given the name
- WP5: Faces generation given speech

A Gantt Diagram that describes the different tasks completed during the project is included in Figure 2.

C. Incidents and Modifications

During the planning period different risks that could obstruct the development of the project were considered. One of them was related to the large amount of data needed when dealing with deep learning methods and the fact that the dataset used is self-built. This work package has been the one that most attention has required, as after successfully building the dataset

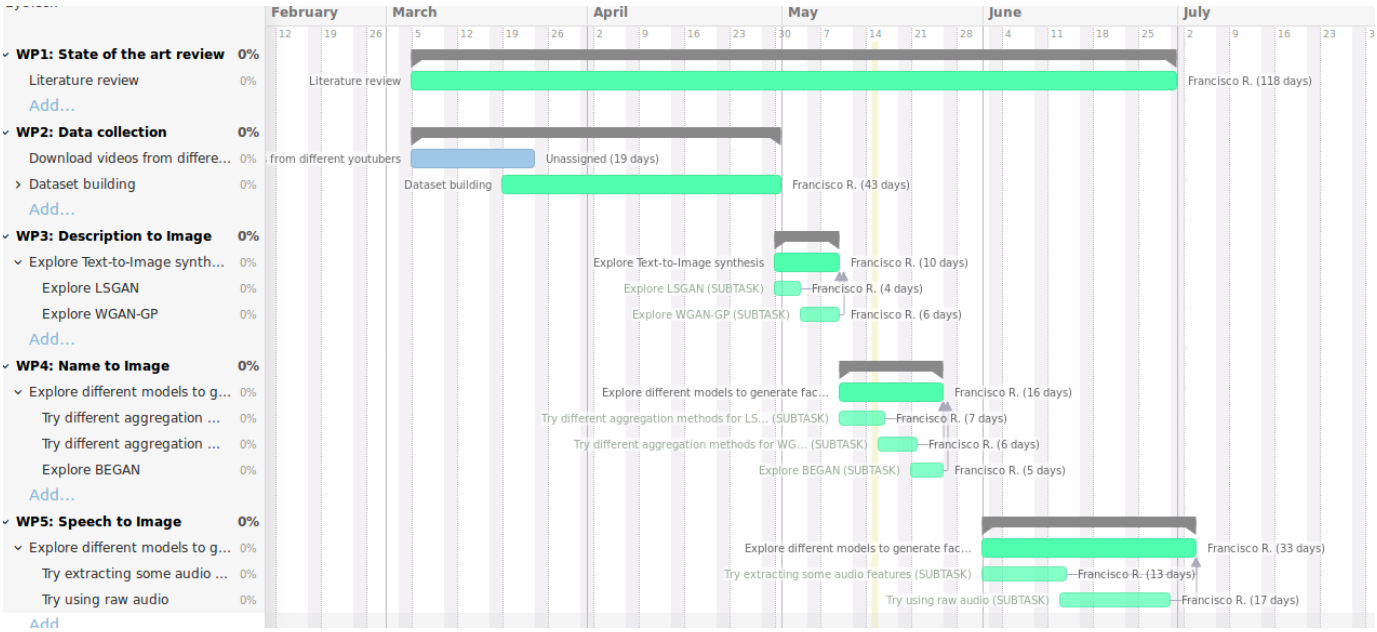

Fig. 2. Gantt diagram of the project

many issues appeared, such as corrupted images and/or speech frames, forcing us to spend more time than expected refining the dataset.

The other major risk considered was related to the inherent difficulty of the task, which forces us to follow a very disciplined schedule due to the limited time available. In that sense, this risk have been successfully avoided. However, we predicted that generating images from raw audio was going to be a more challenging task that doing it from the name, which has turned out to be the opposite. This has lead us to dedicate more time to WP4 than WP5.

II. Related Work

In the recent years the community has put lots of efforts into generative models. This task mainly consists on the learning of a probability distribution from your data to be able to create new samples. Currently, the most common way to solve this problem is to define a random variable Z and feed it into some kind of neural network to directly generate new samples which should follow similar distributions to the real data. Thanks to this approach, we can represent distributions confined to a low-dimensional manifold. Examples of these networks are VAEs [1] and GANs [2]. Here, different cross-modal generative models are presented.

A. Background

As mentioned earlier, Generative Adversarial Networks are a generative method that learns to map samples z from a prior distribution Z to samples x from another distribution X.

This method consists on two networks: a generator G, that learns the mapping between distributions, and a discriminator D, that learns to detect whether a sample is real or fake. This means that the discriminator must classify the samples coming from X as real, while the samples coming from G as fake, while the generator tries to fool the discriminator. This adversarial learning is formulated as a minmax game which tries to optimize the following objective:

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{data}(\mathbf{x})}\left[\log \mathrm{D}(\mathbf{x})\right] + \mathbb{E}_{z\sim p_z(\mathbf{z})}\left[\log(1-\mathrm{D}(\mathrm{G}(\mathbf{z})))\right] \tag{1}$$

From this main idea some other variations have been proposed which improve the quality of the inferred samples. An example of those are Least Square GAN's (LS-GAN), which replace the cross-entropy loss function by the least-square function, avoiding the vanishing gradient problem. [3] Other examples are Wasserstein GAN (W-GAN), which offers more stability during training and defines a meaningful loss function for the discriminator [4] or the Boundary-Equilibrium GAN (BEGAN), that proposes a new equilibrium enforcing method by making use of a loss derived from the Wasserstein distance. [5]

B. Language and Vision

When it comes to the combination of natural language and vision together with generative models, the obvious task that comes into our minds is to generate realistic images from a caption of them. The applications that could benefit from such technology cover a wide range of fields, from civil security to animation generation. Thanks to the latest advances in natural language feature representation and image generation this is now a reality.

In 2016, Reed *et. al.* proposed a text-to-image synthesis method based on GAN's and joint embeddings which was able to generate realistic images of birds and flowers given its descriptions [6]. A diagram of the model can be seen in Figure 3.

Their approach is based on feeding a Deep Convolutional Generative Adversarial Network (DC-GAN) with an embedded representation of the text description. To achieve this they use deep convolutional and recurrent encoders that learn a correspondence between image and text in the generator.

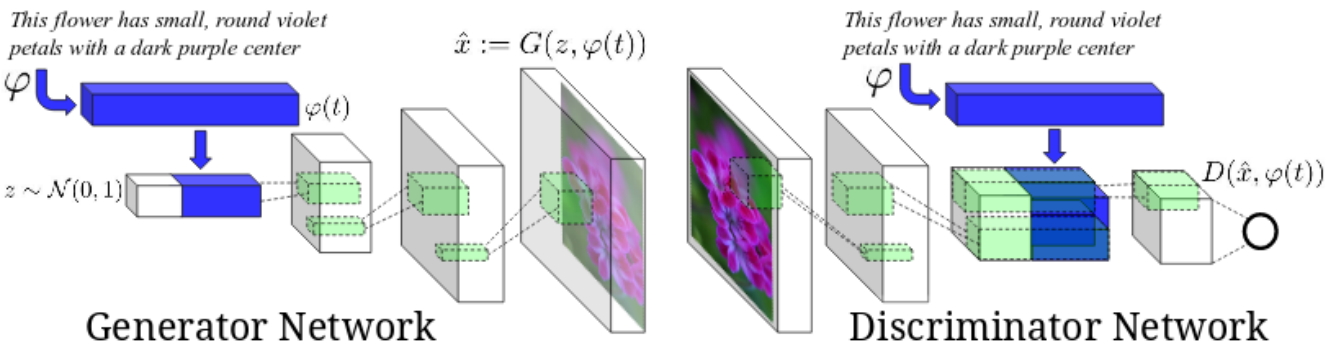

Fig. 3. Diagram of Reed *et.al.*'s approach for text to image generation, using a combination of joint embeddings with DC-GAN.

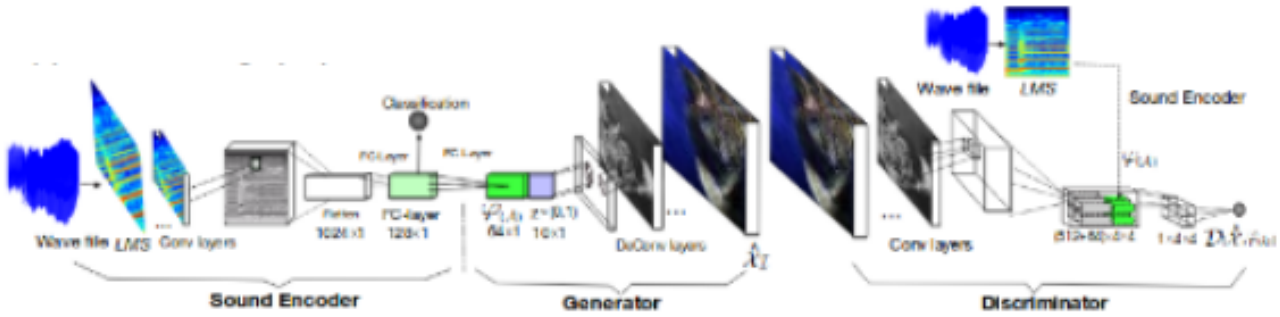

Fig. 4. Sound to image (S2I) network diagram, proposed by Chen *et. al.*.

Once obtained the text embedding, they project it into a lower dimensional space and concatenate it with the noise prior z, which follows a normal distribution such that $z \in \mathbb{R}^Z \sim \mathcal{N}(0, 1)$. Then, the result of the concatenation is passed through an standard deconvolutional neural network that produces a fake image, using batch normalization in all the layers.

On the discriminator side, they input the generated/real image into several convolutional layers until the feature map spatial dimension obtained is 4x4, to then replicate the description embedding spatially and perform a depth concatenation. Then, a 4×4 convolution is done to obtain the discriminator score. As well as in the generator, they use batch normalization for each layer.

Furthermore, in order to improve the learning dynamics of the network, they use a training algorithm called *matching aware discriminator* based on the fact that the it does not only have to distinguish between real and fake samples, but also needs to identify as fake those realistic images that do not match with the description. In order to deal with that, they train the discriminator with pairs of (real image, real description), (fake image, real description) and (real image, wrong description).

This last pair can be produced just by associating a random image of the dataset with a description associated to another image, or by a manifold interpolation. This second approach is useful to generate large amount of additional text embeddings with no additional labeling cost due to the lack of need of having a correspondence with an actual human written text.

Reed's method roughly reflects the semantic meaning of the input text description, but it fails in reflecting the object details. However, in 2017 Zhang *et. al.* propose Stacked Generative Adversarial Networks (StackGAN's) to synthesize high-quality realistic images from their caption [7].

Their approach decomposes the image generation process into two different stages, in analogy to how human painters draw:

- Stage I: where it generates the basic shapes and colors and draws a background layout, resulting on a low-resolution image.
- Stage II: where it corrects defects of the result of Stage I and adds more details on the generated image by reading again the text embedded representation, resulting on a high-resolution image.

The Stage II block of this method is designed using an encoder-decoder architecture, which is fed using the 64×64 generated images from the Stage I and the text embedding representation trained also in previous module. The text embedding is concatenated along the channel dimension with the latent vector and, then, they upsample the feature vector to obtain a 256×256 generated image.

C. Speech and Vision

With the success of generative models with other modalities, combining vision and speech in the context of generative methods has quickly attracted attention from the community, either generating speech from a video or vice versa.

In 2017, Ephrat *et. al.* proposed a method for speech reconstruction from silent videos [8], achieving to infer speech from human facial movements.

Their method also follows an encoder-decoder architecture. It consists on two CNN branches called *towers*, a first one fed with grayscale images of cropped faces and the second one with the optical flow between frames of the given sequence. An embedding is created by concatenating the outputs of each tower. The decoder is formed by fully connected layers which output mel-scale spectrogram, and a post-processing network which outputs linear-scale spectrogram. Ephrat's method is an example of speech inference given a sequence of image frames, but latest research has shown that is also possible to generate realistic images given a frame of speech. One of the first approaches to solve this task was proposed by Chung *et. al.*, who presented a method for generating a video of a talking face.[9] Their method take as input the Mel-frequency cepstral coefficients (MFCC) of a frame of speech of the target speaker (audio) and an image of him/her (identity), and as output an image synchronized with the speech. They used deep convolutional networks for each of the modules of Figure 1.

Similarly to them, Suwajanakorn *et. al.* synthesized high quality video of President Barack Obama. [10] However, they did not generate faces, but images of lips instead, achieving synchronization between speech and the inferred mouth articulations. In order to do it, they decompose the problem in two different steps: firstly a mapping from audio features (MFCC) to sparse shape coefficients, using an LSTM, and a second one to map the mouth shape to the texture.

Chen *et. al.* also explored several methods to synthesize images from sound. [11] Their main approach in summarized in Figure 4. To build the sound encoder they tested different sound representations, achieving best results using Log-Amplitude of Mel Spectrum (LMS), and fed a CNN with them. They projected the resulting sound embedding into a lower dimensional space and concatenated it with the random noise. The rest of the network followed the same architecture as in [6].

D. Adversarial Learning using Raw Audio

In the previous modules there have been shown different examples of generative methods using audio and images. Nevertheless, all of them were in need of an audio feature extraction before feeding the data to the network. This clearly could limit the network's learning, as we are manually extracting information from the audio.

In contrast, Pascual *et. al.* proposed a method for speech enhancement in which they do not work on the spectral domain, but at the waveform level instead. [12] That means, no hand-crafted features were used to train the network.

In this method, the generator consists on an encoder-decoder architecture, both of them fully convolutional, and is trained end-to-end. Furthermore, it learns from different speakers and noise types using a single parametrization, reason why it is suitable for generalization in those dimensions.

III. Dataset

As mentioned in Section I, the data used in this work was taken from YouTube. However, before addressing the data collection task, some other public datasets were considered but discarded because they did not exactly fit our needs. Some examples are the Lip Reading sentences [13], which contains thousands of spoken sentences from BBC, and Lip Reading in the wild [14], which contains 1000 utterances of 500 words from many different speakers. Any of these datasets suit to our needs as none of them contain clean audio and we can relate it to an identity.

Nevertheless, it exists a large-scale speaker identification dataset called VoxCeleb [15] that offers 100k utterances of more than 1000 celebrities, all of them extracted from videos uploaded to YouTube. This dataset seemed to work for our task, although the speech captured was sometimes too noisy for our purpose.

On the other hand, we are now seeing how the *youtuber* phenomenon is growing more and more, reaching in many cases even more spectators than very popular TV shows. This new lifestyle requires from having good quality recording hardware and a well-equipped studio, therefore the audio captured is normally very clean. Moreover, *youtubers* need to attract viewers attention, and they usually do it by overacting a lot. This means that they offer a wide range of expressions, as in the same video they can be laughing, crying or screaming, and, therefore, with such data we can potentially model better how facial expressions are according to a frame of speech.

A. Data Collection

This section describes the procedure of the data collection module, from the video dowloading to all of the preprocessing steps applied to the audio signals and the video frames. A high level representation of this block is shown in Figure 5.

- YouTubers Collection: A list of 62 different Spanish speaker *youtubers* was built, consisting on 29 males and 33 females from different ethnicities and accents. Then, the last 15 videos uploaded to the channel of each of them was directly downloaded from YouTube, together with its synchronized audio.

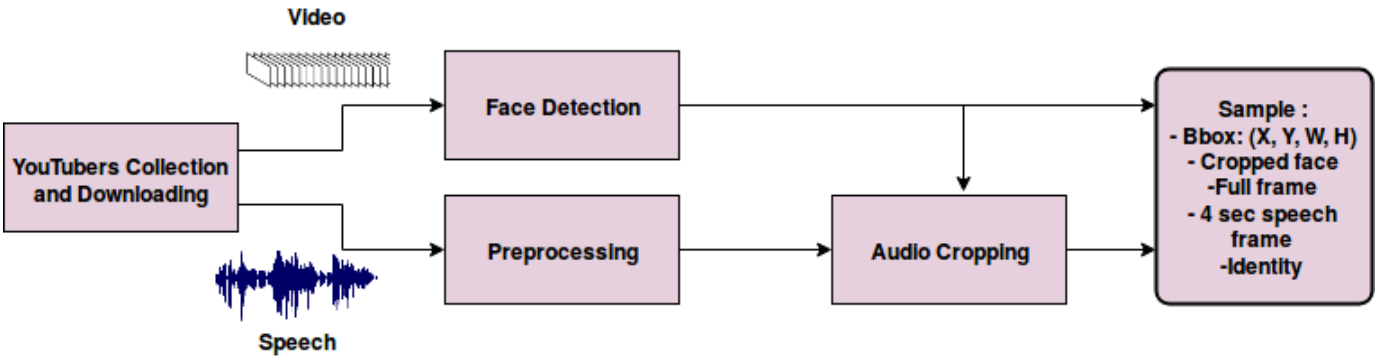

Fig. 5. High level representation of the data collection and preparation module.

TABLE I
SUMMARY OF THE DATASET MAKING A BREAKDOWN BASED ON THE SEX OF THE SPEAKERS. RESULTING DATASET CONTAINS APPROXIMATELY 47 HOURS OF SPEECH.

Sex	Speakers	Faces	Speech (sec)
Male	29	26299	105196
Female	33	15900	63600
TOTAL	62	42199	168796

- Audio preprocessing: The downloaded audio was in Advance Audio Coding (AAC) format at 44100 Hz and stereo. It was converted to WAV, as well as sampled to 16 KHz with 16 bits per sample and converted to a mono signal.
- Face Detection: In order to perform the detection task, a Haar Feature-based Cascade Classifier [16] was used, using pre-trained frontal face features. As we want to relate an audio frame with a single face, we prevent the method from having false positives by taking only the most confident detection for each frame. From each detection are saved the bounding box coordinates, an image of the cropped face in BGR format, the full frame and a 4 seconds length speech frame, which encompasses 2 seconds ahead and behind the given frame. Moreover, we keep an identity (name) for each sample, being able to distinguish between speakers.
- Audio overlapping: Whenever it has been possible, that is, whenever there have been detected faces in consecutive frames, it has been applied an overlapping of 2 seconds between speech frames.
- Image preprocessing: All images, before starting working with them, have been normalized and resized to 64x64.
- Speech frames preprocessing: Each speech frame has been normalized between -1 and 1 as well. Moreover, there has been applied a pre-emphasis step to increase the amplitude of the higher frequency bands while decreasing the amplitude of the lower ones, as higher frequencies are more important for signal disambiguation.

The resulting dataset contains 42199 different samples from 62 different identities, which corresponds to approximately 47 hours of speech.

IV. METHOD

In this section all the different models tested will be detailed, beginning with the changes applied to our baseline mode, and ending with our final speech to image synthesis model.

A. Generative Adversarial Networks

For the development of this project many different GANs have been used in order to compare the quality of the generated results. In particular, there have been explored three different networks for image generation.

One of those GAN's tested is the Least-Squares Generative Adversarial Networks (LS-GAN), which adopts the least-squares loss function for the discriminator instead of the sigmoid cross-entropy that may lead to the vanishing gradients problem during the learning process. In other words, an LS-GAN differs from a typical GAN in the loss function to optimize:

$$\begin{aligned} \max_D V(D) = \frac{1}{2}\mathbb{E}_{x\sim p_{data}(\mathbf{x})}\left[(\mathrm{D}(\mathbf{x})-b)^2\right] + \frac{1}{2}\mathbb{E}_{z\sim p_z(\mathbf{z})}\left[(\mathrm{D}(\mathrm{G}(\mathbf{z}))-a)^2\right] \\ \min_G V(G) = \frac{1}{2}\mathbb{E}_{z\sim p_z(\mathbf{z})}\left[(\mathrm{D}(\mathrm{G}(\mathbf{z}))-c)^2\right] \end{aligned} \tag{2}$$

Where a and b are the labels for fake data and real data respectively and c denotes the value that G wants D to believe for fake data. The authors show that minimizing Equation 2 yields minimizing the Pearson χ^2 divergence between $p_{data}+p_g$ and $2p_g$ if $b-c=1$ and $b-a=2$, where p_{data} and p_g are the probability distribution of the real and the fake data respectively. Another method is to force G to generate samples as close to the real data by setting $c=b$. Experimental results show that both approaches have similar performances and that LS-GANs generate higher quality images than regular GANs.

Another generative model explored was the Wasserstein GAN (W-GAN), which minimizes the Wasserstein Distance or Earth-Mover Distance (EM distance), based on the idea to move a probability distribution towards another distribution:

$$W(p_d, p_g) = \inf_{\gamma\sim\prod(p_d,p_g)} \mathbb{E}_{(x,y)\sim\gamma}\left[\|x-y\|\right] \tag{3}$$

Where p_d stands for p_{data} for simplicity. In equation 3, $\gamma \in \prod(p_d, p_g)$ is the set of all join distributions of p_d and p_g, the "total amount of probability moved" is $\mathbb{E}_{(x,y)\sim\gamma}$ and the distance is $\|x-y\|$. The final EM distance corresponds to the set of joint distributions with the lowest cost.

However, it is unfeasible to consider all the possible joint distributions in order to calculate the cost. Instead, the authors propose a transformation of the formula based on Kantorovich-Rubinstein duality [17]:

$$W(p_d, p_g) = \frac{1}{K} \sup_{\|f\|_L \leq K} \mathbb{E}_{x \sim p_d}[f(x)] - \mathbb{E}_{x \sim p_g}[f(x)] \tag{4}$$

Where f is a mapping function which should be K-Lipschitz continuous. As the loss function decreases during training, the Wasserstein distance gets smaller and the generator models output is closer to the real data distribution. A big issue is to keep the continuity of f along the training process. In order to achieve that, after every gradient update weights are clipped to a small fixed range resulting in a compact parameter space.

A third image generative model explored was the Auxiliary Classifier GAN (AC-GAN) [18], which employs a label conditioning, so that every generated sample has a corresponding class label that must satisfy.

The generator uses bot the class label and the noise prior to produce new samples while the discriminator evaluates the probability distribution over the generated samples and their class labels. Therefore, an additional objective function must be added to our cost: the log-likelihood of the correct class. Then, the discriminator is trained to maximize $L_C + L_S$ and the generator to maximize $L_C - L_S$, where L_C corresponds to the log-likelihood of the correct class and L_S to the log-likelihood of the correct source (fake/real).

B. Text-to-Image Synthesis

In Section II-B has been explained a method proposed by Reed *et. al.* that synthesize images of flowers and birds given a textual description of them by using a LS-GAN [6]. This method was not tested with faces as there is not any available dataset that relates textual descriptions with face images.

In their approach, they use as loss function for the discriminator a binary cross entropy (BCE) as the last layer they use is a sigmoid, as they try to classify if a sample is real or fake. Instead, as mentioned in SectionIV-A, we can directly evaluate the outputs of the last layer using a Mean Square Error (MSE) by removing the sigmoid function at the end of the network, achieving a more stable training and improved results. In the case of the generator, as its loss value directly depends on the discriminator's loss, it is also altered.

In this case, we used the default configuration proposed by the authors (Adam optimizer with a learning rate of 0.0002) and kept batch normalization in all the layers of the generator and the discriminator.

C. Name-to-Image Synthesis

Taking the previous model as baseline, we replaced the input textual embedding by a one-hot vector representing the name (identity) of each of the *youtubers*. For this particular problem, W-GAN, LS-GAN and AC-GAN were tested and several methods were explored within the framework of how to work with a one-hot vector as an identity descriptor in a generative model.

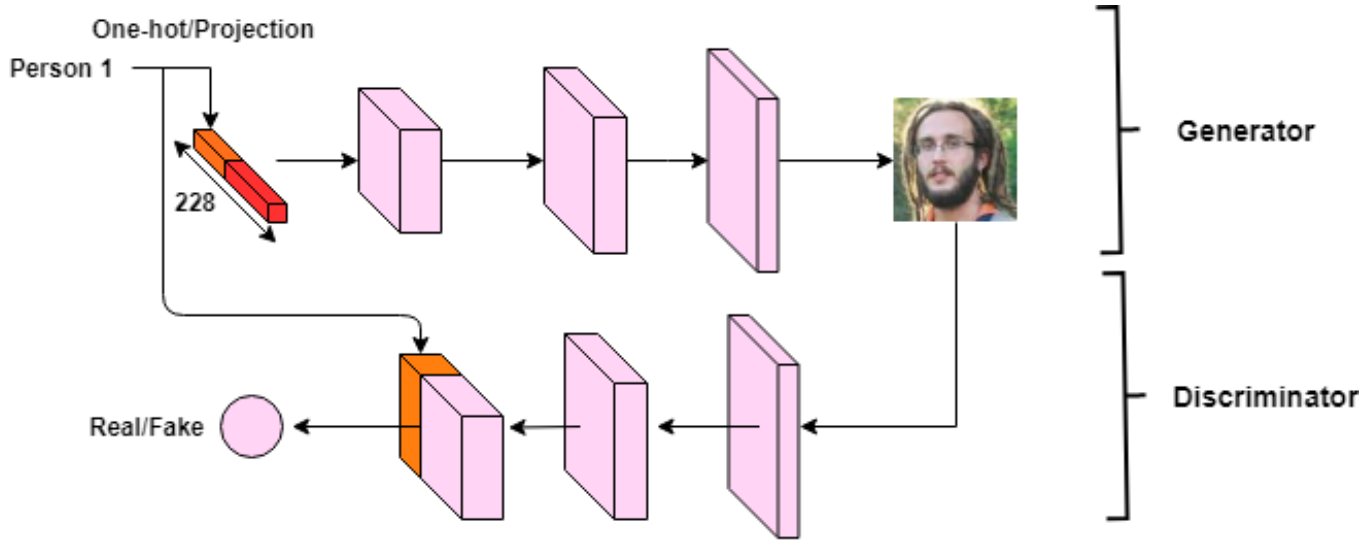

Fig. 6. Diagram of the name-to-image synthesis method. Orange blocks stand for the one-hot vector representation. In the case of the projection it produces a 128 representation, while when concatenating the one-hot vector is repeated and padded with zeros to keep dimensionality untouched. Red blocks stand for the noise prior, of size 100, while pink blocks represent convolutional/deconvolutional blocks.

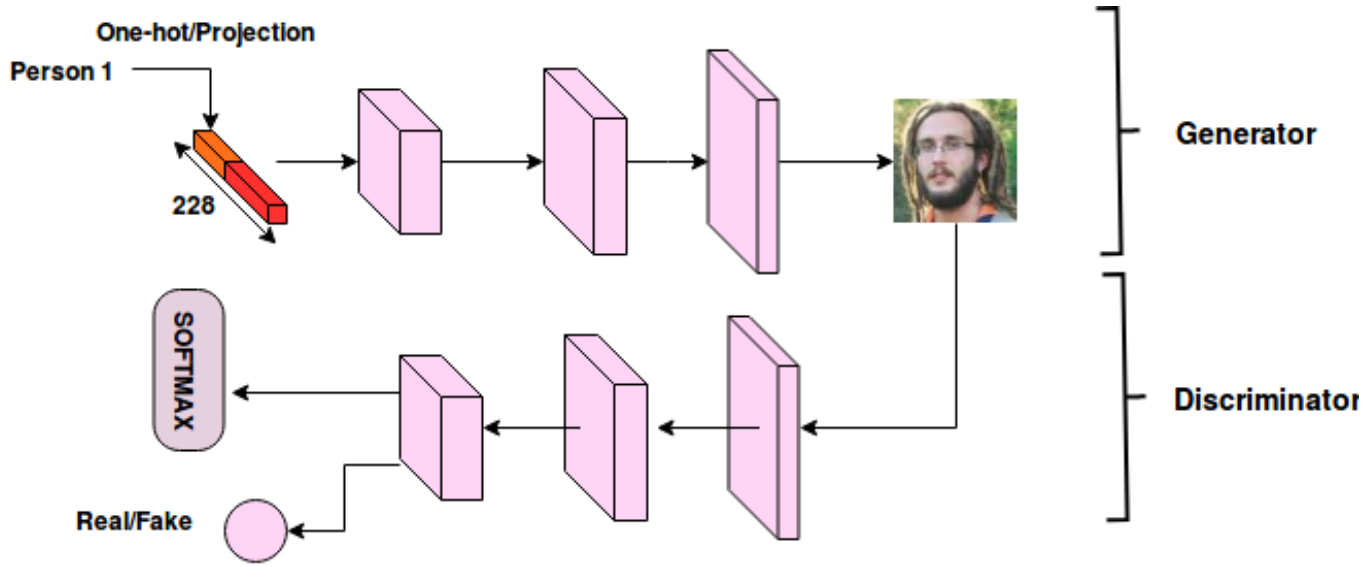

Fig. 7. Diagram of the name-to-image synthesis method using AC-GAN. Instead of introducing the one-hot vector into the discriminator the identity is learned through an auxiliary classifier. The softmax contains as units as *youtubers* the dataset has plus an additional which stands for the samples which are fake.

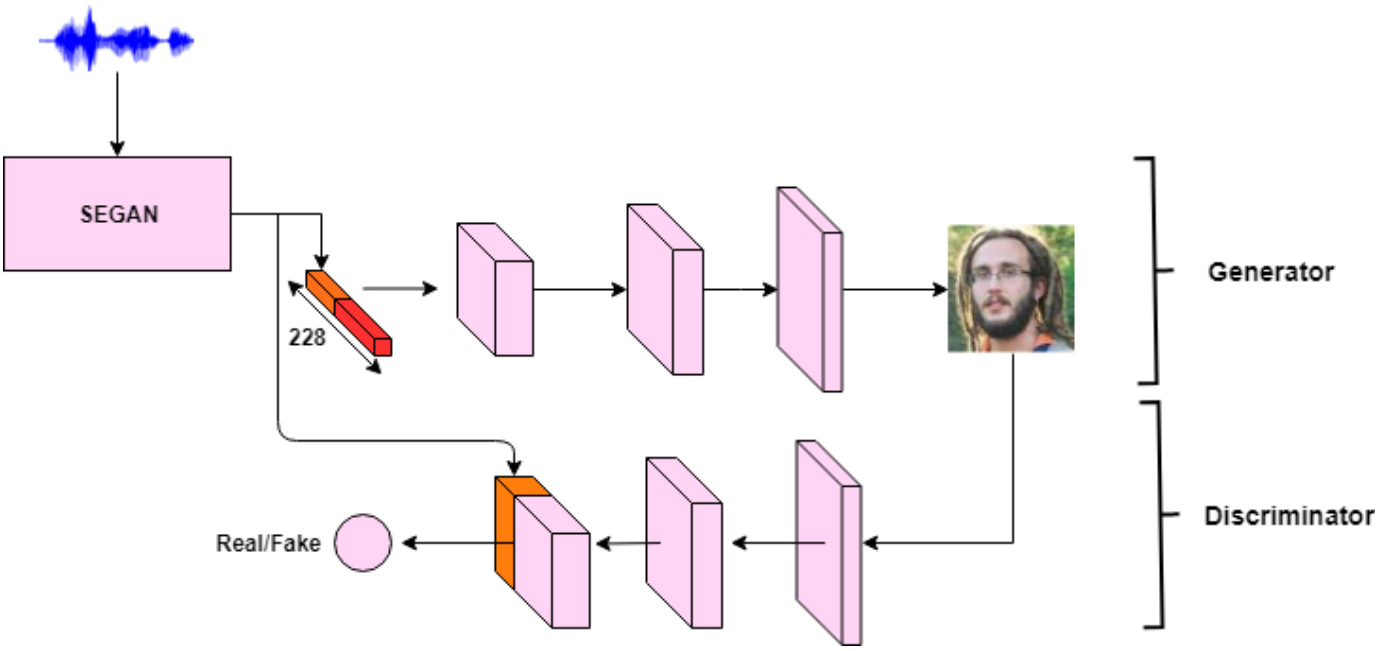

Fig. 8. Diagram of the speech-to-image synthesis method. Orange blocks stand for the audio embedding vector of size 128, red blocks stand for the noise prior, of size 100, while pink blocks represent convolutional/deconvolutional blocks.

One first approach is to concatenate the one-hot vector to the noise along the channel dimension to feed them to the generator, while in the discriminator's side the vector must be repeated along the spatial dimension to later concatenate it with the along the channel dimension.

Another approach is to project the one-hot vector to the same dimensionality of the text embeddings that we were using in the text-to-image synthesis task, and then repeat the procedure of the simple concatenation. Note that in this case we can train the projection both in the generator and the discriminator or, instead, just train it in the generator and, through a skip connection, use this projection in the discriminator.

One of the main issues when training a GAN is the instability of the discriminator. In [19] Miyato *et. al.* propose a new method called spectral normalization, which performs normalization at each layer by using the largest singular value of the weight matrix W. However, they do not apply singular value decomposition at each round of the algorithm, as it could become computationally heavy. Instead, they use the power iteration method which introduces a very small additional computational time [21].

For this method all batch normalization layers were removed from the discriminator while replaced in the generator by spectral normalization layers, and the learning rate of the Adam optimizer was set to 0.0004 in the discriminator and to 0.0001 in the generator, as suggested in [20] by Zhang *et. al.*.

D. Speech-to-Image Synthesis

The main goal of this work was to synthesize images of faces given a frame of speech. In order to do that, the model built is also based in Reed's *et. al.* method, but generating the speech embedding with the discriminator module of SEGAN.

In this case, we built SEGAN using 6 discriminator blocks of sizes 64, 128, 256, 512, 1024 and 1024 respectively. The kernel size for all of these blocks was set to 15 and batch normalization was kept. We applied a pooling of 4 to avoid having lots of layers. Note also that SEGAN is only trained in the generator and the resulting embedding is also used in the discriminator. As well as in the previous case, there are no batch normalization layers in the discriminator and in the generator is used spectral normalization instead.

Two different alternatives were explored when dealing with this task: a first one using as input just the speech utterance and a second one feeding the network also with the one-hot vector representing the name, in which they are combined using the same procedure as in Section IV-C.

V. Experiments and Results

Along section IV some specific hyperparameter and model configurations have been specified. In this section we aim to discuss the effect of those in the context of face generation.

As explained in Section IV-C, one of these experiments was comparing different ways to input de one-hot vector into the generator: directly concatenating it to the noise prior or previous to the concatenation apply a projection to expand the one-hot vector dimensionality.

Results obtained from both methods show how the identity of the generated image is correct, although the generator experience mode collapse as for each identity only one image is being formed. Regarding to the image quality obtained, we see how projecting the one-hot vector there are more details on the images although in many cases faces are not realistic. On the other hand, concatenating directly the identity representation we obtained more realistic results but considerably blurred (Figure 9).

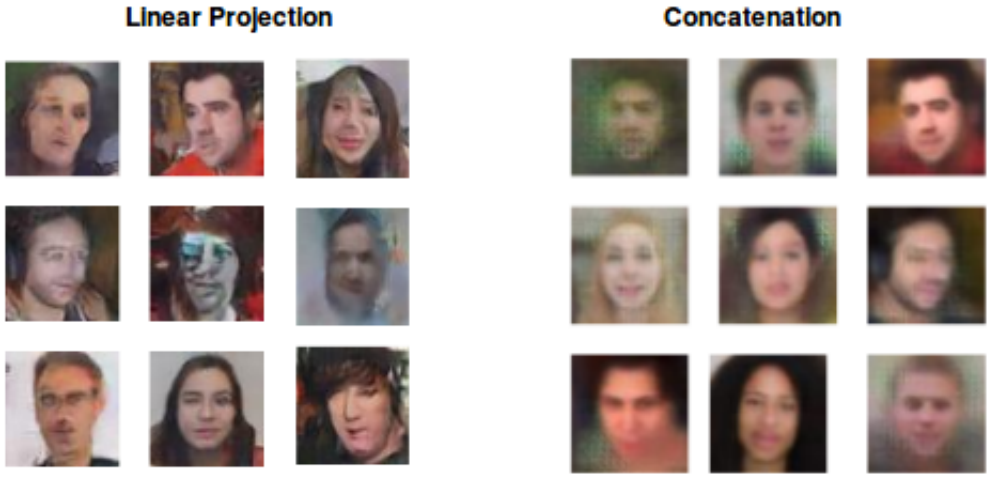

Fig. 9. Comparison of random results generated using the linear projection and concatenation methods with a LS-GAN in the name-to-face synthesis task.

Considering that the LS-GAN was experiencing mode collapse, other generative models were explored like the W-GAN and the AC-GAN. In the case of the first one, generated results were very unrealistic and completely random, not being able to learn the identity. On the other hand, with the AC-GAN, identities were correctly predicted but images were very noisy and, moreover, the network was collapsing to the mode as well (Figures 10 and 11).

Therefore, other methods needed to be explored in order to avoid mode collapse. One of the firsts experiments done was removing batch normalization layers in the discriminator and replace them in the discriminator by spectral normalization. This modification did not have any effect on the mode collapse issue. However, results obtained improved as some noise is erased from the generated images (Figure 12).

Another approach to deal with it was to aggregate gaussian noise to the one-hot vector. Again, this procedure was applied to both methods: linear projection and concatenation. Results show that this method can only be applied using an embedding representation, as when aggregating the noise directly to the one-hot vector the network only predicted one image for all of the

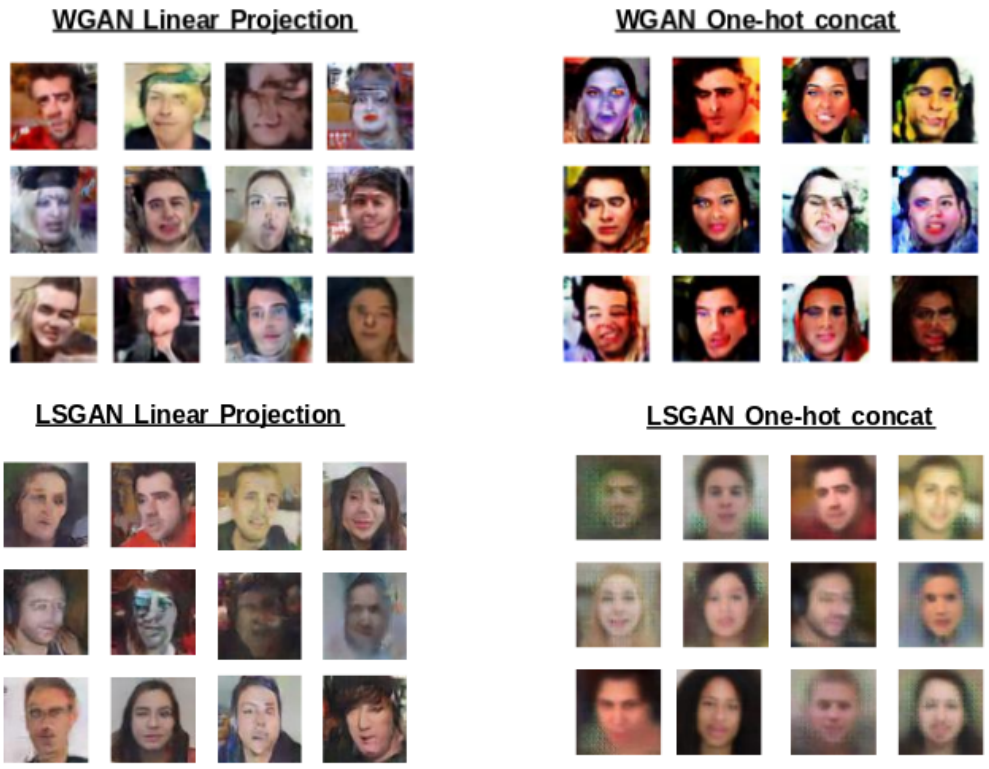

Fig. 10. Comparison of random results generated using the linear projection and concatenation methods with the WGAN and the LSGAN in the name-to-image synthesis task.

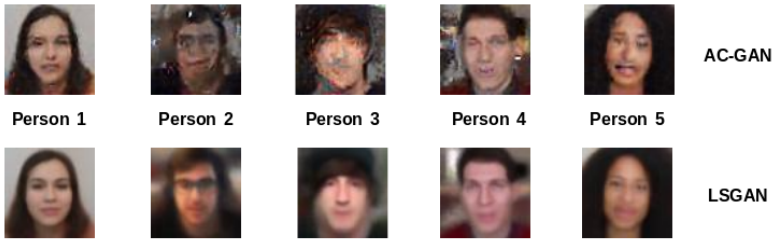

Fig. 11. Comparison of random results generated using LS-GAN and AC-GAN in the name-to-image synthesis task.

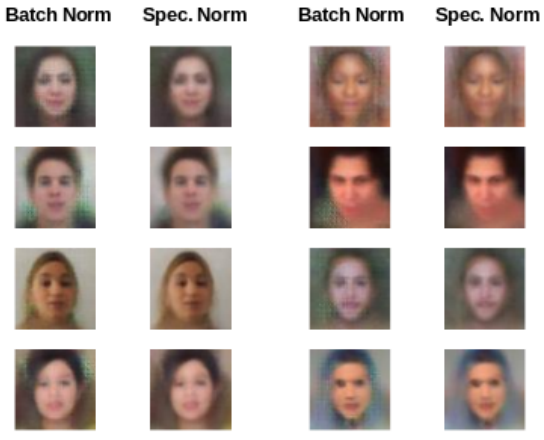

Fig. 12. Comparison of random results generated using batch normalization and spectral normalization in the generator of a LS-GAN in the context of name-to-image synthesis.

people regardless of the identity. Nevertheless, this method did not work to solve the mode collapse issue and the generated images were of worse quality with respect to the ones obtained without the noise addition.

The learning rate value of both the discriminator and generator were also changed to 0.0004 and 0.0001 respectively. Again, this modification did not help with the mode collapse but results obtained improved substantially, as the blurriness disappear (Figure 13).

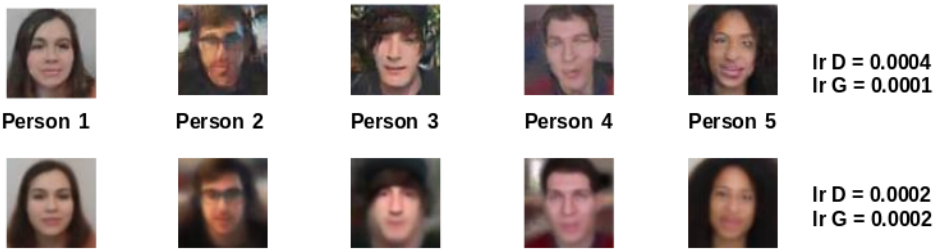

Fig. 13. Comparison of random results generated using different learning rate values with a LS-GAN in the context of name-to-image synthesis.

Finally, the weights of the first convolutional layer were checked and showed how the network was omitting the noise prior information, as the values were very close to zero and more than 3 orders of magnitude lower than the ones associated to the one-hot vector. That means that the noise prior is being ignored by the network and, instead, as suggested in [22], stochasticity in the generated results can be achieved by introducing controlled dropouts after each convolutional layer of the generator except the last one, with the cost of having worse image quality. However, differences between samples are very subtle. (Figure 14).

Fig. 14. Results obtained in the name-to-image synthesis task using controlled dropouts to introduce stochasticity.

Once solved the name-to-face synthesis task, we moved forward towards working with speech utterances. In this case, a couple of methods were tested. Firstly, a combination of the one-hot vector and the speech embedding obtained from SEGAN, and another one using only auditory information. Both methods obtained very similar results, as predicted faces are very realistic during training time and by adding controlled dropouts stochasticity is introduced, although it affects in subtle details (Figure 15). Nevertheless, the model is not able to generalize to speech utterances not present in the dataset nor to external speakers, as the method produces random results working under these scenarios.

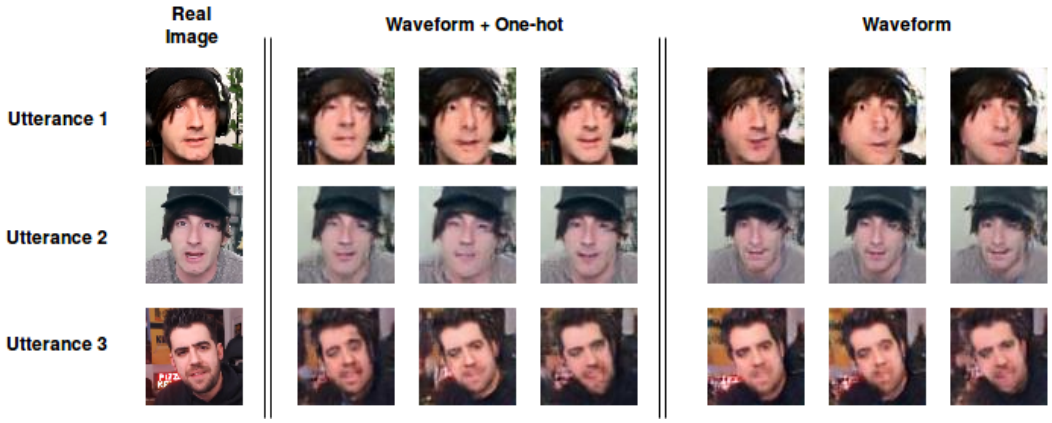

Fig. 15. Comparison of different results obtained in the speech-to-image synthesis task from utterances of the training set.

In order to compare the results with the most typical solutions, which consist on using handcrafted features instead of working at the waveform level, we extracted the Mel Spectrogram for each of the utterances and extracted an embedded representation using SEGAN in order to reduce its dimensionality. It was used a FFT window size of 512 with a hop size of 128. However, using Mel Spectograms the model was not able to learn the identity of the speakers.

TABLE II
INCEPTION SCORE OBTAINED FOR DIFFERENT EXPERIMENTS OF THE NAME-TO-FACE SYNTHESIS TASK.

Experiment	Mean Score	Std
LS-GAN projecting input	2.912896	0.114554
LS-GAN concat. input	2.051858	0.028261
LS-GAN concat. input, LR from [20]	**3.002615**	0.074115
LS-GAN concat. input, LR from [20] + dropouts	2.634878	0.094980
AC-GAN projecting one-hot vector	2.120474	0.043865

TABLE III
INCEPTION SCORE OBTAINED IN THE SPEECH-TO-FACE SYNTHESIS TASK DEPENDING ON THE INPUT.

Experiment	Mean Score	Std
Waveform + one-hot vector	**3.598630**	0.157528
Waveform	3.513372	0.162294
Waveform + controlled dropouts	3.339188	0.168021

A. Quantitative Results

Evaluation of image generative models has been a significant challenge for the community, as defining an appropriate performance measure is complex.

One approach which achieves that is the Inception Score [23] (IS), which correlates the image quality with the human perception. To calculate this score, generated images must be fed into an Inception model trained on ImageNet [24] to obtain the conditional distribution of an image given a label. This distribution should have low entropy energy, as the output should mostly be focused on one label. On the other hand, the entropy of the marginal distribution over all the different images should be high (high variance over the images), as we expect to have stochasticity in the outputs of the network. Combining these two requirements, Salimans *et. al.* define a compact metric (Equation 5, where x represents an image and y is the label).

$$\exp \mathbb{E}_x KL(p(y|x)|p(y)) \tag{5}$$

Tables II and III show the results obtained for different experiments. These results allow us to compare between models and ratify our extracted conclusions after observing qualitatively the outputs. First thing to notice is that, despite qualitative results show that many samples are pretty unrealistic when using a linear projection instead of directly concatenating the one-hot representation, the score obtained for this model is higher. However, note that the standard deviation of the scores with this method id a lot higher than the other experiments. As stated in [25], this could be happening because IS only consider the distributions of the generated samples, ignoring real data, reason why it may favor models that simply learn sharp images. That means that blurriness is very penalized in this score, although qualitatively all results are acceptable.

Another aspect to remark is the increasing of image quality after replacing batch normalization by spectral normalization and changing the learning rate values for the ones applied in [20]. Again, this matches with the explanation above, as doing this changes we achieved to remove the blurriness. Our observation that the introduction of controlled dropouts decreases image quality is ratified by this measure, as the score decreases by 0.4 in average.

Regarding to the speech-to-face results we observe how results using just the waveform or adding the one-hot vector are very similar. Again, stochasticity can be barely achieved but with the cost of losing image quality.

Note that during this section always we have referred the IS as an image quality evaluation method. This means that models could achieve higher score than others while producing unrealistic samples which do not follow real data distribution. Moreover, this metric favors models that memorizes training samples (unable to penalize overfitting) and it is agnostic to mode collapse.

Therefore, other evaluation methods could have been considered, such as the Fréchet Inception Distance (FID), in which real data is also considered [26]. In this technique some intermediate features from an Inception network are extracted and modeled as a multivariate Gaussian distribution. However, due to time limitations and resources constraints this measure has not been computed at writing time.

VI. CONCLUSIONS

In this work several cross-modal generative models have been explored. We have shown how using a non-strong identity descriptor like could be the name (one-hot vector) models tend to collapse, only achieving stochasticity in the generated samples by introducing controlled dropouts in the generator. Nevertheless, variations between samples are minor, normally involving little changes in the mouth, the eyes or details of the background.

On the other hand, when using speech information, samples generated are more realistic. Stochasticity for a certain utterance must also be introduced by controlled dropouts. However, the model only recognizes the identity with known utterances, as no generalization has been achieved.

This lack of generalization could happen due to many facts. Probably adding more data to the dataset, as well as leaving the training last more epochs would help to accomplish better generalization. Another option could be training the audio encoder network with a speaker recognition task to extract a meaningful audio embedding to use as input for our generative model. Applying the matching aware discriminator algorithm explained in Section II-B could also be an option. Nevertheless, despite this issues, we show the potential of image generative models from auditory information.

Some quantitative results have been computed. However, Inception Score method do not offer a good performance evaluation in term of realism of the results, but it does in terms of image quality instead. Moreover, as it has been used a self built dataset, comparison with any state of the art method is difficult with this measure. The model could have been tested on other datasets such as VoxCeleb, as mentioned in Section III, but this dataset have some inherent challenges regarding to speech analysis not applicable to this task at current status.

Besides of all the generative experiments, a dataset built from content uploaded by *youtubers* have been proposed. Normally *youtubers*' videos have clean speech signals and a wide range of emotions are expressed by them because they need to overact to attract audience. The resulting dataset can be applied to many tasks, such as face or speaker recognition, voice conversion or synthesis problems. As a future work it is planned to remove false positives from the dataset and expand the dataset in terms of number of identities and amount of data for each of them.

ACKNOWLEDGMENT

Firstly, I want to thank my thesis supervisor, Xavier Giro i Nieto, for all his help and efforts not only throughout this work but also during the last year and a half, thanks to whom I have had the opportunity to be in touch with the world of research. Likewise, I also want to thank Santiago Pascual de la Puente and Amaia Salvador Aguilera whose advices have always been very useful and from whom I learned a lot. Moreover, the idea of creating a dataset from *youtubers* content was theirs. I would also like to thank Kevin McGuinness for allowing me to go to Insight Centre at Dublin and for all his help during this project. I also need to thank Loc Cardone, from the INP-ENSEEIHT (Toulouse), who has helped me with the evaluation of the results for this work.

REFERENCES

[1] Kingma, D. P., and Welling, M. (2013). *Auto-encoding variational bayes*. arXiv preprint arXiv:1312.6114.
[2] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., and Bengio, Y. (2014). *Generative adversarial nets.* In Advances in neural information processing systems (pp. 2672-2680).
[3] Mao, X., Li, Q., Xie, H., Lau, R. Y., Wang, Z., and Smolley, S. P. (2017, October). *Least squares generative adversarial networks.* In 2017 IEEE International Conference on Computer Vision (ICCV) (pp. 2813-2821). IEEE.
[4] Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., and Courville, A. C. (2017). *Improved training of wasserstein gans.* In Advances in Neural Information Processing Systems (pp. 5769-5779).
[5] Berthelot, D., Schumm, T., and Metz, L. (2017). *Began: Boundary equilibrium generative adversarial networks.* arXiv preprint arXiv:1703.10717.
[6] Reed, S., Akata, Z., Yan, X., Logeswaran, L., Schiele, B., and Lee, H. *Generative adversarial text to image synthesis.* Proceedings of The 33rd International Conference on Machine Learning (ICML), 2016
[7] Zhang, H., Xu, T., Li, H., Zhang, S., Huang, X., Wang, X., and Metaxas, D. (2017, October). *Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks.* In IEEE Int. Conf. Comput. Vision (ICCV) (pp. 5907-5915).
[8] Ephrat, A., Halperin, T., and Peleg, S. (2017, August). *Improved speech reconstruction from silent video.* In ICCV 2017 Workshop on Computer Vision for Audio-Visual Media.
[9] Chung, J. S., Jamaludin, A., and Zisserman, A. (2017). *You said that?*. arXiv preprint arXiv:1705.02966.
[10] Suwajanakorn, S., Seitz, S. M., and Kemelmacher-Shlizerman, I. (2017). *Synthesizing obama: learning lip sync from audio.* ACM Transactions on Graphics (TOG), 36(4), 95.
[11] Chen, L., Srivastava, S., Duan, Z., and Xu, C. (2017, October). *Deep Cross-Modal Audio-Visual Generation.* In Proceedings of the on Thematic Workshops of ACM Multimedia 2017 (pp. 349-357). ACM.
[12] Pascual, S., Bonafonte, A., and Serra, J. (2017). *SEGAN: Speech enhancement generative adversarial network.* arXiv preprint arXiv:1703.09452.
[13] J. S. Chung, A. Zisserman (2017) *Lip Reading in Profile* In British Machine Vision Conference
[14] J. S. Chung, A. Senior, O. Vinyals, A. Zisserman (2017) *Lip Reading Sentences in the Wild* In IEEE Conference on Computer Vision and Pattern Recognition
[15] A. Nagrani, J. S. Chung, A. Zisserman (2017) *VoxCeleb: a large-scale speaker identification dataset* In INTERSPEECH
[16] Viola, P., and Jones, M. (2001). *Rapid object detection using a boosted cascade of simple features*. In Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on (Vol. 1, pp. I-I). IEEE.
[17] Cedric Villani. *Optimal Transport: Old and New*. Grundlehren der mathematischen Wissenschaften. Springer, Berlin, 2009
[18] Odena, A., Olah, C., and Shlens, J. (2016). *Conditional image synthesis with auxiliary classifier gans*. arXiv preprint arXiv:1610.09585.
[19] Miyato, T., Kataoka, T., Koyama, M., and Yoshida, Y. (2018). *Spectral normalization for generative adversarial networks.* arXiv preprint arXiv:1802.05957.
[20] Zhang, H., Goodfellow, I., Metaxas, D., and Odena, A. (2018). *Self-Attention Generative Adversarial Networks*. arXiv preprint arXiv:1805.08318.
[21] Golub, G. H., and Van der Vorst, H. A. (2001). *Eigenvalue computation in the 20th century*. In Numerical analysis: historical developments in the 20th century (pp. 209-239).
[22] Isola, P., Zhu, J. Y., Zhou, T., and Efros, A. A. (2017). *Image-to-image translation with conditional adversarial networks*. arXiv preprint arXiv:1611.07004
[23] T. Salimans, I. J. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen. *Improved techniques for training GANs*. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, Advances in Neural Information Processing Systems 29, pages 2

[24] Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., and Fei-Fei, L. (2009, June). *Imagenet: A large-scale hierarchical image database*. In Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on (pp. 248-255). Ieee.
[25] Borji, A. (2018). *Pros and Cons of GAN Evaluation Measures*. arXiv preprint arXiv:1802.03446.
[26] Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. (2017). *Gans trained by a two time-scale update rule converge to a local nash equilibrium*. In Advances in Neural Information Processing Systems (pp. 6626-6637).