Human Laughter Generation using Hybrid Generative Models

Nadia Mansouri* and Zied Lachiri
National Engineering School of Tunisia, El Manar University
Le Belvèdére, 1002, Tunis, Tunisia
[e-mail: {nadia.mansouri, zied.lachiri}@enit.utm.tn]
*Corresponding author: Nadia Mansouri

Received May 22, 2020; revised February 1, 2021; accepted March 8, 2021; published May 31, 2021

Abstract

Laughter is one of the most important nonverbal sound that human generates. It is a means for expressing his emotions. The acoustic and contextual features of this specific sound are different from those of speech and many difficulties arise during their modeling process. During this work, we propose an audio laughter generation system based on unsupervised generative models: the autoencoder (AE) and its variants. This procedure is the association of three main sub-process, (1) the analysis which consist of extracting the log magnitude spectrogram from the laughter database, (2) the generative models training, (3) the synthesis stage which incorporate the involvement of an intermediate mechanism: the vocoder. To improve the synthesis quality, we suggest two hybrid models (LSTM-VAE, GRU-VAE and CNN-VAE) that combine the representation learning capacity of variational autoencoder (VAE) with the temporal modelling ability of a long short-term memory RNN (LSTM) and the CNN ability to learn invariant features. To figure out the performance of our proposed audio laughter generation process, objective evaluation (RMSE) and a perceptual audio quality test (listening test) were conducted. According to these evaluation metrics, we can show that the GRU-VAE outperforms the other VAE models.

Keywords: Laughter Synthesis, Variational Autoencoder, Autoencoder, Objective and Subjective Evaluation

A preliminary version of this paper appeared in International Conference on Advanced Technologies for Signal and Image Processing (ATSIP) 2020, September 2-5, Sousse, Tunisia. This version includes a concrete analysis and supporting implementation results on the audio laughter synthesis procedure.

http://doi.org/10.3837/tiis.2021.05.001 **ISSN : 1976-7277**

1. Introduction

The human speech mechanism produces two types of sounds: verbal and nonverbal speech. Verbal speech conveys a message which contains linguistic information and has a clear articulatory description, whilst nonverbal speech, such as paralinguistic sounds carries nonlinguistic messages and occurs in unexpected, variable and natural manner. Nowadays, the interaction and communication between human (phone, robot, virtual agent etc) and machines became parts of the daily activities. To ensure a better conversation, machines should be capable of analyzing, detecting, recognizing and synthesizing the different sounds produced by human. Since birth, laughter is the first sound that child generates, it is his way to communicate with others before pronoucing his first words [1]. Laughter is an essential feature of human communication, it conveyes messages about emotions [2] and may also have positive effects on health [3]. Despite the importance of this sound which occurs frequently in our daily life, its synthesis is an under-explored domain. The main reasons behind the few attempts to synthesize laughter is its diversity. Unlike speech, laughter occurs in many forms (snore-like, grunt-like and cackle-like laugh) and in different situations (mockery, happy and sometimes in a sad situation). Therefore, it is difficult to gather a sufficient amount of laughter data [4]. Also, it has been demonstrated that the acoustic characteristics of laughter signals changes according to the contextual information like laughter surrounding with speech or laughter occurring simultaneously with speech (speech-laugh) or pure laugh [5]. Another problem that encounters the laughter synthesis procedure is the modelization of the acoustic features (vocal fold vibration and strong aspiration driven by a sudden burst of the airflow).

Until nowadays, the technics used for laughter synthesis are those for speech synthesis. So, inspired by the works done in speech synthesis based on the Hidden Markov Model (HMM), which is considered as a basic model of statistical parametric speech synthesis (SPSS) [6], Jerome Urbain [7] and Thomohiro [8] explore this generative model to synthesize laugh. This procedure is known as a sequence-to-sequence regression problem determine the relation between linguistic and acoustic features. The principal idea of this process is to extract in the first place a context-dependent model (linguistic features) which describes the contextual structure of laughter like the position of the syllable within the word, the identity of the current phone, number of words in the sentence, the number of phones in the syllable, the position of the word within the sentence etc... This context-dependent is then used for the prediction of the acoustic features. This model gives information not just about the linguistic features but also about the factors and events that can influence and lead to the acoustic features production of a phone. Since it is impossible to cover all the context combinations decision trees were used [9]. The results obtained in this study are acceptable but far from human-like laugh. This is due to the fact that the used corpus includes pure laughter than that in conversational scenes and decision trees have some limitations in expressing complicated functions of input features such as XOR. By using the same protocol, the authors in [8] investigates the influence of contextual information to synthesize a more realistic laughter sound where a conversational speech corpus is used. The naturalness of the synthesized laughter was improved. In [10, 11] audio laughter and visual laughter were synthesized independently and joined together to provide a 3D avatar with an audio-visual laugh. In [12, 13] speech-laughs were synthesized either by changing the vowels of neutral speech with those of laughter or by concatenating speech-smile with laughter bursts. For instance, other works have focused on synthesizing laughter by making changes to the acoustic features of the signal (fundamental frequency and strength of excitation) [14] or by modeling the laughter structure with a mass-spring system and synthesized it by the Linear Prediction technic [15]. Results of the later method shows

that the synthesized laughter is perceived as unnatural.

Recently deep learning (DNN) is explored into laughter synthesis. In [16] the authors used the Merlin toolkit as a benchmark for the DNN-based laughter synthesis. This methodology have improved the quality of the synthesized laughter compared to the HMM model since deep neural networks are used as a replacement of decision trees. However, the quality of the laughter is still poor. The main reasons behind this failure into synthesizing a human-like laugh with DNN is the speaker dependent nature of the used model which needs a large amount of data for one speaker that is not available. In addition, the authors in [17] proposed a synthesis approach of conversational laughter with wavenet. The main idea of their suggested procedure is to conditionne the wavenet by power contour predicted from the HMM model. So, the results of this method depend on the generated power contour, a poor estimated contour leads to the degradation of the synthesized laughter.

As we said previously, the mojority of laughter synthesis work was based on the HMM and deep naural network. However, for being able to generate a more realistic laughter with these models we need an accurate process to extract the linguistic features without the need of an expert. But as we know laughter is a non verbal sounds and there is no specific rules guiding to its production. It is produced by a series of sudden bursts of air, released from the lungs, keeping the vocal tract almost steady [8]. So, based on this characteristic it is so hard to extract the context-dependent model of a signal that did not obey to either any production or any contextual rules like speech. In addition, to avoid going through these problems we purposed to use another types of generative models that produce audio laughter without thinking about the contextual and linguistic features and able to model its acoustique features.

Lately, deep neural network based on unsupervised learning process such as the Autoencoder (AE) and the Variational autoencoder (VAE) shows their effectiveness in data resolution. So far, autoencoders were used in many audio applications as an analysis-synthesis scheme where the input signals dimension is reduced to a latent vector (encoding), and the signal is regenerated from it (decoding). In [18] authors used Denoising AE to reduce noise and enhance the quality of synthesized speech. In addition, deep autoencoder is used to extract significant features from the spectral envelop which improve the text to speech synthesis procedure [19]. In [20] different architectures of the AE were investigated and used to ameliorate the music synthesis process.

On the other side, some researchers used the VAE [21], which is known as a probabilistic version of the AE, to synthesize data. In the first place, VAE are designed for image processing [22]. Newly, variational autoencoder is employed for music and speech production. In [23], the authors used the wavenet speech synthesis model as a reference to suggest a Wavenet autoencoder based on the conditional autoregressive decoder which learns the temporal codes from the raw audio waveform and demonstrated the performance of their proposed benchmark by using a large dataset of musical notes the Nsynth dataset. The same dataset is used by the authors in [24], where they proposed the use of variational autoencoder as a generative model to reproduce audio musical sounds and to evaluate the effectiveness of their models they compared it with a linear technique the principal component analysis (PCA). Because of their success, VAE is extended for speech processing . For example, in [25] VAE is used for modeling the magnitude spectrogram (STFT) for speech enhancement. For instance, the authors in [26] propose a new sequence to sequence model, an RNN semantic variational autoencoder (RNN–SVAE). This model solve the problem of preserving global latent information from a long sequence of words. All the results achieved in data synthesis when using these two models motivates the reasearch in this paper. This paper is an extension of our work presented in [27]. The principle contributions of this studies are:

1) We are able of using laughter data from different speakers and even merged between two database: the AVLaughterCycle and the Amus database which helped us into collecting sufficient data for the training process. This cannot be used with the previous laughter synthesis methodology because they are speaker-dependant models.

2) Most of the laughter synthesis methodology are those used for speech synthesis. These methods run into some difficulties during the synthesis of laugh. Among these problems, the modelization of contextual and linguistic information related to the production of laughter signals and the modelization of its acoustic characteristics (vocal fold vibration, power contour, glottal closure instance) since there is no extraction tools dedicated to paralinguistic sounds. In this work we are capable of generating and creating laughter from the spectrogram and avoid going through the exhausted engineering tools in order to derive an accurate representation of this signal.

3) To more improve the laughter synthesis procedure, we performe an alliance between the VAE and the different variety of neural networks (recurrent neural network and convolutional neural network). The obtained results prove that the combination between Gated Recurrent Unit and VAE (GRU-VAE) achieve better performance in RMSE (8.6 dB) and the listening test (4).

The proposed paper is organized as follows: the audio laughter generation process is detailed in section 2. Section 3 describes the different generative models (VAE, LSTM-VAE, VAE-CNN and AE) used for this purpose. Section 4 shows the experimental set up starting from the database until the implementation of the different architecture. Resuts evaluation and discussion are given in section 5. Finally, we conclude this paper in section 6.

2. Audio Laughter Synthesis Methodology

The global methodology of laughter synthesis based on the VAE and the AE is inspired from some works [20, 24, 27]. This analysis-transformation-synthesis procedure is elaborated in **Fig. 1** and detailed in the next subsection. To our knwoledge, this is the first time that these models were used to process laughter signals. According to this figure the audio laughter generation process is a combination of three main stages: the analysis, the training and synthesis procedure.

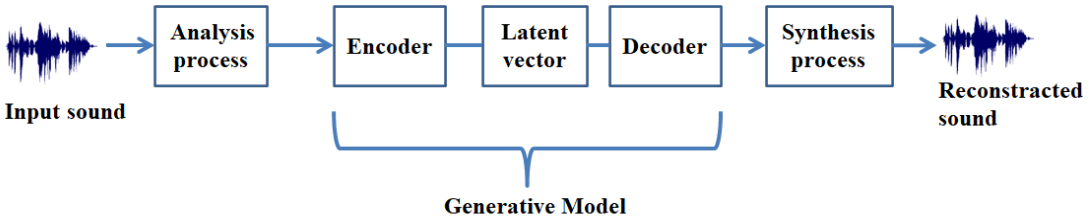


Fig. 1. Audio laughter synthesis procedure

The analysis step: This step is about the process of extracting the acoustic features of laughter signals more precisely the log magnitude spectrogram where the Short Time Fourier Transform (STFT) is used.

The training step: This step involves the learning process of the generative models. The main objective of the AE and the VAE is to reconstruct the input vector x (the log magnitude spectrogram) by learning its representation. This learning process is performed through an encoder, which is capable of transforming the input vector into a high-level representation

(latent vector) and a decoder responsible of predicting $\hat{x}$ from this latent vector. More details about the structure of the AE and the VAE and their training scheme is presented next.

The synthesis step: This step concentrates on the laughter waveform generation. In order to regenerate it an intervening vocoder is essential. Numerous vocoders have been used for this purpose, such as STRAIGHT [28], WORLD [29], DSM [30], etc... However, each of these vocoders requires a specific type of parameters. Taking as an example the STRAIGHT and the WORLD vocoder, in order to use them, we need the frequency (F_0), the aperiodicities (BAP) and the spectral envelop coefficients. In our case, we have only used the log magnitude spectrogram and discard the phase information. So, just providing the predicted log magnitude spectrogram prevent us from generating the laughter waveform. One of the most famous vocoders that can be used during our study is the Griffin-Lim [31] because it is designed to estimate the phase information of the signals based on a de-normalization of the estimated log magnitude spectrogram and finally reconstruct the time-domain signal. The de-normalization is performed by changing the log scale to the linear scale.

The same topologies are applied to the different architecture of the VAE and the AE.

3. Description of the Different based-laughter Synthesis Models

During this section, we details the different generative models (VAE and AE) used in the synthesis procedure. Besides, we propose the use of Recurrent Neural Network (precisley the Long Short Term Memory cell and the Gated Recurrent Unit) and Convolutional neural network along side to the VAE.

3.1 Auto-encoders (AE)

The Autoencoder is an unsupervised kind of artificial neural network [19, 20], used to reduce the dimension of the input data. This model is an association of an encoder and a decoder (**Fig. 2**).

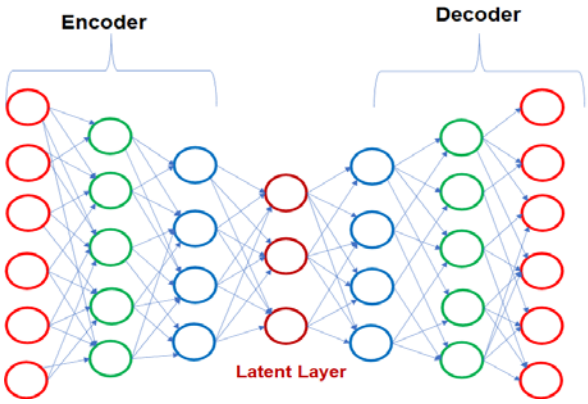


Fig. 2. The Deep Autoencoder architecture

The encoder is a feed forward neural network designed to compress a high-dimensional low-level input vector to a low-dimensional high-level latent vector z. The main objective of this encoding process is to extract a more relevant information and properties of x. The decoder is a symmetric network of the encoder, its aim is to reconstruct an estimate $\hat{x}$ of the input vector data x from the latent vector z. The autoencoder model is formulated by the following equations:

$$z = f_{enc}(W_{enc}x + b_{enc}) \tag{1}$$

$$\hat{x} = f_{dec}(W_{dec}z + b_{dec})$$

Where W_{enc} , W_{dec} are the weight matrix and b_{enc} , b_{dec} are the bias vector of the encoder and decoder respectively; f_{enc} is an activation function and $\hat{x}$ is the predicted vector.
During the training process, the weight matrix and the bias vector are learned and updated by minimizing a cost function between x and $\hat{x}$. The cost function used in our study is the mean squared error (MSE):

$$MSE = \frac{1}{N}\sum_{n=1}^{N}\left\|x_n - \hat{x}\right\|^2 \tag{2}$$

The shallow autoencoder can be refined by stacking more hidden layers to build a deep autoencoder (DAE), as provided in **Fig. 2**. This makes the AE more powerful and qualified to extract more relevant information.

3.2 The variational autoencoder (VAE)

Generative models are unsupervised learning technique known by their effectiveness in learning the true data distribution of a training set in order to reproduce a new data with some variations [32]. Variational autoencoder is one of the most famous generative models and a modified version of the classical deterministic autoencoder. The idea behind Variational autoencoder resides in generating an observed data x from a hidden latent variable z , in a mathematical point of view this model is illustrated by a probability distribution function (in (3)) [21].

$$p_\theta(x, z) = p_\theta(x \mid z)\, p_\theta(z) \tag{3}$$

where $p_\theta(z)$ is the prior distribution of the latent variable z, $p_\theta(x|z)$ is the likelihood of the observation x and θ denotes the set of distribution parameters. To make this model solvable a posterior distribution $p_\theta(x|z)$ should be computed in order to infer the characteristics of the latent variable z. This can be presented by [32]:

$$p_\theta(z \mid x) = \frac{p_\theta(x \mid z)\, p_\theta(z)}{p_\theta(x)} \tag{4}$$

According to this equation the denominator $p_\theta(x)$ is called the evidence and is calculated by marginalizing out the latent variables from the joint distribution $p_\theta(x, z)$.

$$p_\theta(x) = \int p_\theta(z)\, p_\theta(x \mid z) dz \tag{5}$$

To make (5) computed, variational inference is proposed and its aim is to approximate the posterior distribution $p_\theta(z|x)$ to a tractable distribution $q_\varphi(z|x)$ [32, 33]. In other words, to make a best approximation between the real posterior distribution and their correspondence

one, a minimization of the Kullback-Leibler Divergence is performed. Note that the Kullback-Leibler divergence (KLD) is a means to measure the similarities between two probability distributions. And, in general, $q_\varphi(z|x)$ can be a gaussian or a Bernoulli distribution. The minimization of the KLD leads to the following equation [33]:

$$\log p_\theta(x) = L(\varphi,\theta,x) + KLD(q_\varphi(z\,|\,x)\,||\,p_\theta(z\,|\,x)) \tag{6}$$

The term $logp_\theta(x)$ known as log evidence and is constant, the KLD divergence between $q_\varphi(z|x)$ and $p_\theta(z|x)$ is non- negative and equivalent to zero only and only if $q_\varphi(z|x)$ is equal to the true posterior distribution, the last term $L(\varphi,\theta,x)$ is called the variational lower bound and is a lower bound on the log-likelihood of the data which means that $L(\varphi,\theta,x) < logp_\theta(x)$. The $L(\varphi,\theta,x)$ is given by:

$$L(\varphi,\theta,x) = -\beta KLD(q_\varphi(z\,|\,x)\,||\,p_\theta(z)) + E_{q_\varphi(z|x)}[\log p_\theta(x\,|\,z)] \tag{7}$$

Hence, in practice minimizing *KLD* divergence is equivalent to maximizing the variational lower bound $L(\varphi,\theta,x)$. This term is defined as the association of a regularization and a reconstruction accuracy terms [21, 33]. Where the regularization term define the approximation between the posterior and prior distributions. Regarding the reconstruction term, the cross-entropy error or the mean squared error can be used. As we are dealing with the problem of audio generation where the input vector *x* is nothing more then a set of real valued spectral magnitude and the loss error is the difference between the predicted output and the real input it is appropriate to use the mean squared error instead of the cross-entropy. To make the balance between these two terms, a β value is defined and it should be chosen carefully [34]. According to what is described previously VAE uses neural networks to illustrate the variational inference model (encoder) and the generative model (decoder).

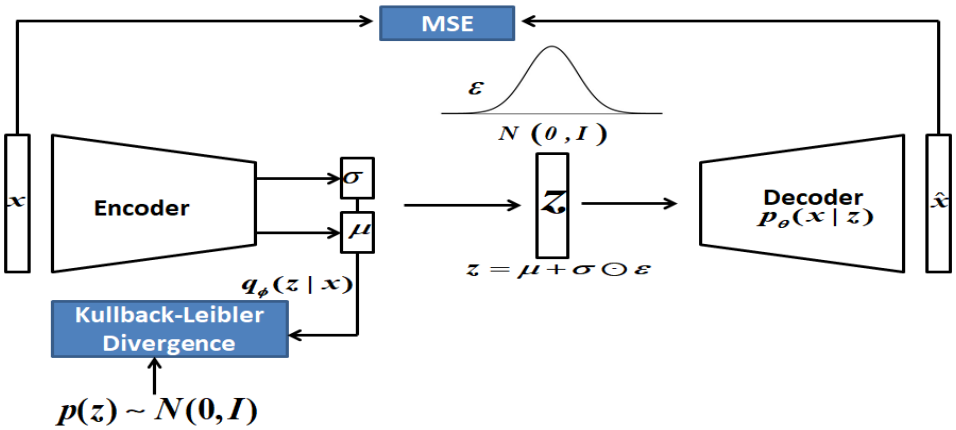


Fig. 3. The Variational autoencoder architecture

The **Fig. 3** gives an outline of the whole procedure.The input data vector *x* is entering to the encoder neural network which is defined as an approximation of the posterior distribution $q_\varphi(z|x)$ and outputs a mean vector $\mu(x)$ and the covariance $\sigma^2(x)$ which is a diagonal matrix

used to sample the latent vector z. So $q_\varphi(z|x)$ is usually a gaussian distribution:

$$q_\varphi(z \mid x) = N(z \mid \mu(x,\varphi), \sigma^2(x,\varphi)) \tag{8}$$

To regularize the structure of the latent space a prior distribution is used. This regularization is performed by the Kullback-Liebler divergence between $q_\varphi(z|x)$ and $p_\theta(z)$. In general the standard $p_\theta(z)$ is a normal distribution $N(0, 1)$, according to that the KLD divergence is given by (9):

$$KLD(q_\varphi(z \mid x) \| N(0,1)) = \frac{1}{2}(1 + \log(\sigma^2) - \mu^2 - \sigma^2) \tag{9}$$

Thus sampling z by just following σ and μ prevent the update of the network parameter sets by backpropagation, to solve this problem a reparameterization trick is suggested [33, 35] and consist of sampling z from $\varepsilon \sim N(0,1)$ as presented below:

$$z = \mu + \sigma \otimes \varepsilon \tag{10}$$

where $\otimes$ denote an element-wise multiplication.

The decoder $p_\theta(x|z)$ uses z as an input to estimate the mean and the variance and generates the output x. $p_\theta(x|z)$ is a gaussian distribution:

$$p_\theta(x \mid z) = N(z \mid \mu(z,\theta), \sigma^2(z,\theta)) \tag{11}$$

φ and θ are respectively the parameter set of the encoder and decoder network (the weights and bias).

3.3 LSTM-VAE

During training, deep variational autoencoder architecture (with Sigmoid, Hyperbolic tangent, Linear and Relu activation function) ignore the sequential nature of laughter. So, the better choice to include this special feature for audio signals is the use of the Recurrent Neural Network (RNN) [36]. RNN are known by their capacities in memorizing information learnt from prior inputs when generating outputs. According to this characteristic, the RNN output depends on the output of the previous layer and the internal hidden states (hidden neurons). This makes them convenient for modeling the time series data with their temporal dependencies. During our study, we used the Long Short Term Memory cell (LSTM) [9] which is one of the various RNN architecture, created to solve the vanishing gradient problem and to take into consideration the long term dependencies of the data. The LSTM-VAE model is presented by **Fig. 4**. This figure shows that the LSTM-VAE model is also a combination of an encoder and a decoder [37-39], where the encoder network is a stacked of two LSTM layers dedicated to approximate the posterior $p_\theta(z_t|x_t)$ and extract the latent vector z by feeding their output into a linear model in order to estimate the mean $(\mu(x_t))$and the variance $(\sigma(x_t))$ parameters. The decoder network includes the same number of LSTM layers and units as the encoder network. Its aim is the reconstruction of the input log spectrogram from the sampled latent vector z. Besides to the LSTM layers, a merging between the VAE and the Gated Recurrent Units (GRU) is performed. The GRU-VAE structure shows prominence results compared to the LSTM-VAE. Thus, the major benefits behind using the RNN-VAE [26] is that

the compressed information of the input log magnitude spectrogram is learned as a region of the latent space rather than a single point.

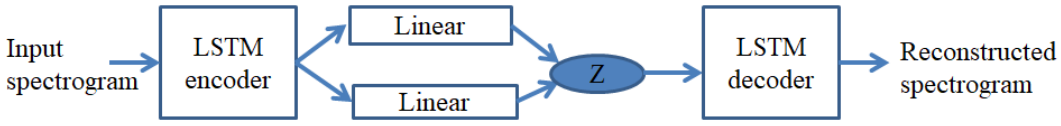


Fig. 4. The LSTM-VAE architecture

3.4 CNN-VAE

Convolutional neural network (CNN) is known as a powerful tool for image generation and recognition [40]. In this section, we will join the CNN and VAE together for laughter audio generation procedure. For this purpose a 128*128 mel log spectrogram is extracted from 5 s audio length. The CNN- VAE architecture is given by **Fig. 5**.

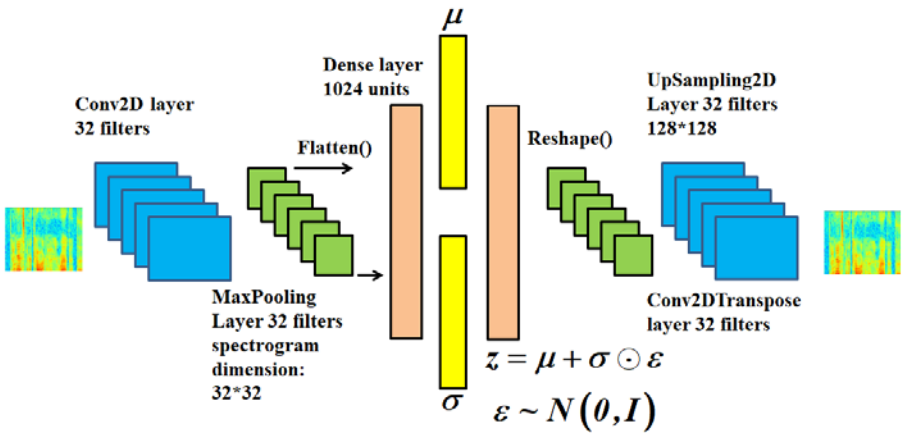


Fig. 5. The CNN-VAE architecture

The encoder is composed of a convolutional and max- pooling layers, however the decoder is an association of deconvolutional and upsampling layers. Below we give an overview of the encoder and decoder networks.

- The encoder: {input:[128*128*1] → Conv2D: [128*128*32] → MaxPooling2D: [64*64*32] → Conv2D: [64*64*1] → MaxPooling2D: [32*32*1] → FC-1024}
- The decoder: {input z:1024 → Reshape: [32*32*1]→ Conv2DTranspose: [32*32*32] → UpSampling2D: [64*64*32] → Conv2DTranspose: [64*64*1]→ UpSampling2D: [128*128*1]}

4. Experiments

4.1 Database

During this study we used two differents laughter databases: the AVLaughterCycle and the Amus database.

The AVLaughterCycle database is an audio-visual laughter database that incorporates both the audio signals and facial motion tracking. Twenty four subjects, 6 females and 15 males, paticipate in the recording process. To elicite spontaneous laugh, participants were structured to watch hilarious video, the experiment is held in an isolated room in order to collect a clear laughter sound. In total, The corpus comprise about 1024 laughter samples. This database contains various laughter sounds some of them are voiced (song-like laughter) others are unvoiced (snort-like and grunt-like laughter). In addition, the phonetic labels of these laughter sounds are clustered under 7 groups : vowels, consonants, fricatives, nareal fricatives, plosive, cackles and hum-like. The Amus database is recorded to study the effect of amusement in speech. This database is around 3 hours of data collected from one female and two males. The corpora includes diverse kind of data such as: speech-smile, speech-laugh, neutral speech and 148 laugh samples. In order to gather this collection, the subjects were asked to read sentences with different styles in english and french. During this work only the laughter audio signals were kept. All the audio sounds were sampled at 16 kHz. More details about the recording and the transcription procedure of these two databases can be found in [5, 41].

To train our model a total of 1100 laughter sounds are used, we split it into a training set of 80% and a test set of 20%. The remaining 100 laughter sounds are used for the evaluation task.

4.2 Data pre-processing and Experimental setup

For the data processing task, we first remove the silence frame from the laughter audio signals. For the magnitude spectrogram extraction, the Short Time Fourier Transform (STFT) with 1024-point is carried out to the input signal using the hamming window with 25 ms length and 10 ms of overlap. The extracted 513 points positive frequency magnitude spectrogram were transformed to the log scale and normalized between -1 and 1 which makes them useful by the neural network of the VAE and the AE. At the synthesis stage, before the reconstruction of the time-domain waveform from the decoded log magnitude spectrogram with the Griffin-Lim vocoder a denormalization from log to linear scale is applied.

As we said previously, either for the standard VAE or the LSTM-VAE, GRU-VAE and CNN-VAE, these models are composed of an encoder aiming at approximate the posterior $p_\theta(z|x)$ to a known distribution $q_\varphi(z \mid x) \sim N(\mu(x), \sigma(x), \varphi)$ and sampled a latent vector z from this distribution. A reparameterization trick is performed on z (in (11)). For this trick we perform two differents choices:

- ε is sampled from the gaussian distribution $N(0,1)$.
- ε is sampled from an isotropic gaussian distribution $N(0, \sigma_\varepsilon * I)$, where we treat the value of σ_ε as an hyperparameter.

Besides to ε, we noticed during our experiments that the choice of the β parameter in (8) can affect the results. In the next section we will discuss how the variation of these two parameters (ε and σ_ε) have an impact on laughter synthesis procedure.

For all the models, the encoder and decoder are trained on the extracted log magnitude spectrogram to minimize a cost function (MSE for the AE and variational lower bound for the VAE) between the real log magnitude spectrogram and the decoded one. Several encoding dimensions from 4 to 100 and various pairs of activation functions were used for the hidden and output layers (Tanh, Sigmoid), (Tanh, Linear) and (Relu, Sigmoid). The architecture used for VAE is [513,256,128,*encod*,128,256,513]. Where the *encod* parameter is the latent vector dimension and 513, 256, 128 are the neurones number in each layer**.** In addition, We investigated diverse architecture of the autoencoder, starting with a shallow AE to a deeper one and they are respectively: [513,*encod*,513], [513,256,*encod*,256,513] and [513,256,128,*encod*,128,256,513]. The same activation functions were used. Concerning the LSTM-VAE and GRU-VAE models, we choose the following structure [513,256,128,*encod*, 128,256,513] where the encoder and the decoder are composed of three LSTM (GRU) layers with linear activation function.

The keras deep learning toolkit is used to implement the VAE and AE, the training is performed by the adam optimizer [42] with a learning rate of 10^{-3} over 300 epochs with early stopping criterion, a batch size of 32 and the MSE is used as the reconstruction accuracy in (8). The AE is trained in an end-to-end method. The scikit-learn toolkit was used to implement the PCA.

5. Evaluation and Discussion

To asses the effectiveness of our audio laughter synthesis process based on the VAE and the AE, objective and subjective evaluation metrics were proposed.

5.1 Objective evaluation

For the objective evaluation metrics we used the RMSE (in decibel dB) to compute the prediction error between the real log magnitude spectrogram and the decoded one performed on the evaluation set.

Variational autoencoder case: During this section, we discuss the objective results for laughter audio generation process based on the different architecture of the VAE model (VAE, LSTM-VAE and CNN-VAE).

For simplicity, we only Consider the case of the (Tanh, Linear) configuration with *encod* = 100. The **Fig. 6** indicates the evolution of the RMSE when varying the values of β and σ_ε. As you can see, a high value of β and σ_ε leads to bad results. However, by reducing the $\sigma_\varepsilon$the performance of the model can be improved even when β has a higher values. Therefore, if we take a high value of β we need to reduce the value of σ_ε and vice versa. As a conclusion, we realized that the reconstruction of the log magnitude spectrogram is sensitive to the β and σ_ε parameters, where choosing a high or a low value of β can affect the audio generation procedure in a negative way. So, it is important to make the best choice between these two parameters in order to be able to get a better results.

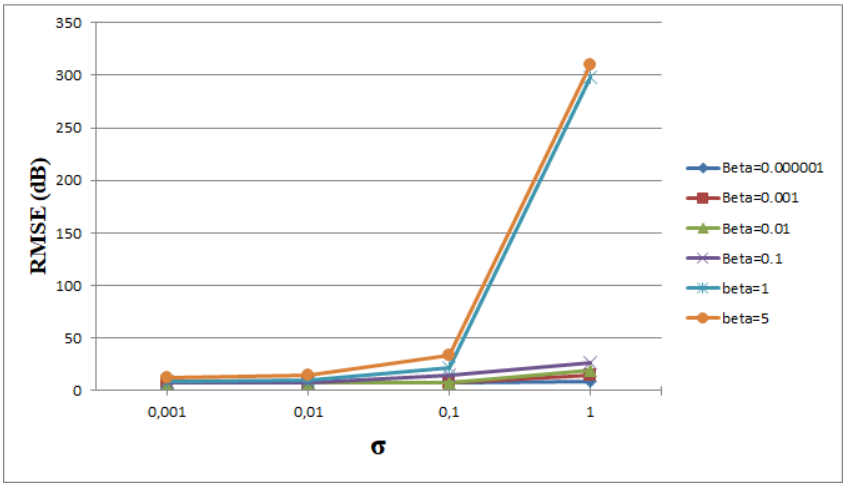


Fig. 6. The RMSE for different values of β and σ_ε

For the rest of our experiments, we choose to set $\beta = 1 * 10^{-6}$ and $\varepsilon \sim N(0, I)$. The **Fig. 7** and **Fig. 8** shows the evolution of the RMSE of a male and a female laughter audio generation as a function of the latent vector dimension (*encod*). As expected, the RMSE decreases according to the latent vector dimension for all architectures. For the standard VAE model, the (Tanh, Linear) configuration gives better results compared to the (Relu, Sigmoid) and (Tanh, Sigmoid) configurations where the RMSE decreases from 20 dB for *encod* = 4 to 7dB for *encod* = 100. The GRU-VAE outperforms the LSTM-VAE model and the standard VAE and even the CNN-VAE, which proves the importance in taking advantage of the sequential nature of the data for the audio laughter generation procedure.

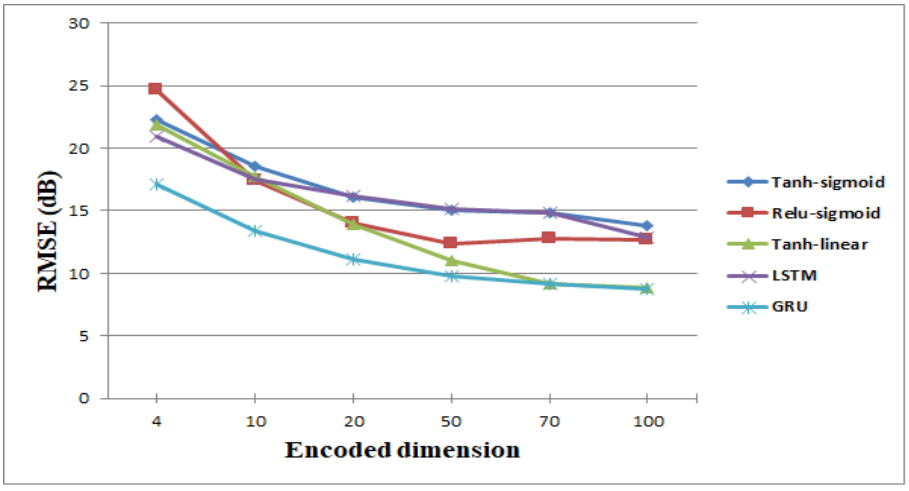


Fig. 7. The RMSE for the VAE as a function of the latent vector dimension for a female laughter audio generation

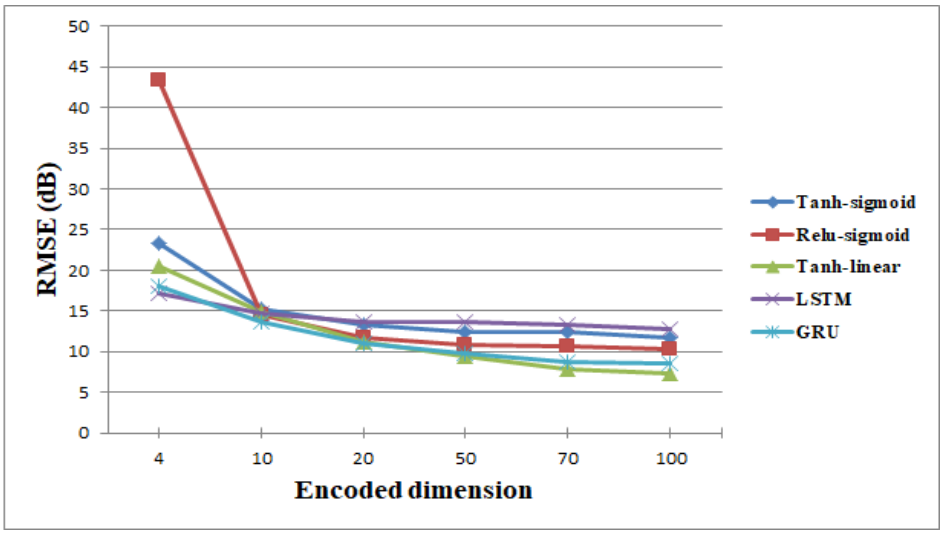


Fig. 8. The RMSE for the VAE as a function of the latent vector dimension for a male laughter audio generation

Autoencoder and PCA case: The next figures (**Fig. 9**, **Fig. 10** and **Fig. 11**) indicate the RMSE (dB) values obtained for each configuration of the auto-encoder model.

- The **Fig. 9** represents the RMSE of the deep autoencoder model with the configuration [513,256,128, *encod*,128,256,513] for the various pairs of activation functions and those of the PCA. According to this figure the DAE outperforms the PCA when employing the (Tanh, Linear) as activation function. In addition, The PCA gives better RMSE values, about 7.01 for 100-dimensional latent vector, compared to the DAE (15.63) when using the pair (Tanh, Sigmoid).

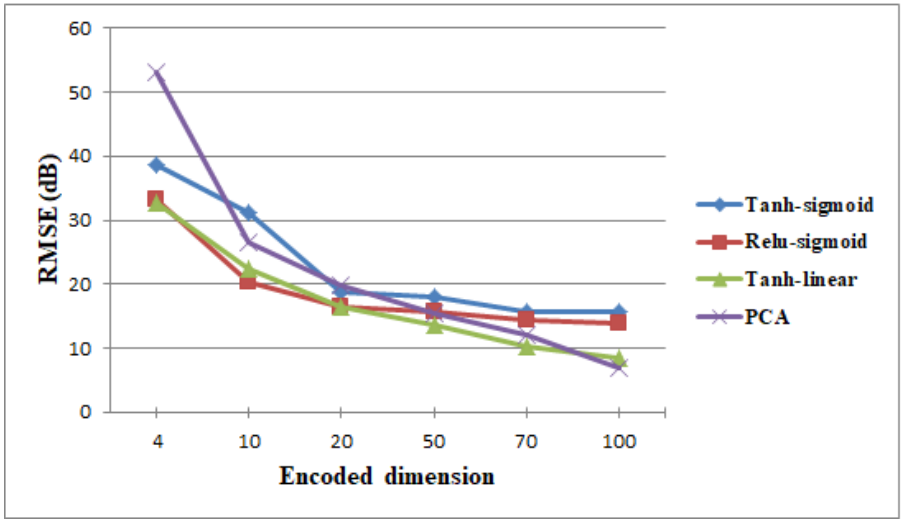


Fig. 9. The RMSE evolution of the DAE architecture ([513, 256,128, *encod*,128, 256, 513]) and the PCA.

- The **Fig. 10** shows the reconstruction error (RMSE) of the autoencoder with the configuration [513,256, *encod*,256,513]. As we can see, for this configuration the AE model provides better results than the PCA especially when employing as activation

function either (Tanh, Linear) or (Relu, Sigmoid).

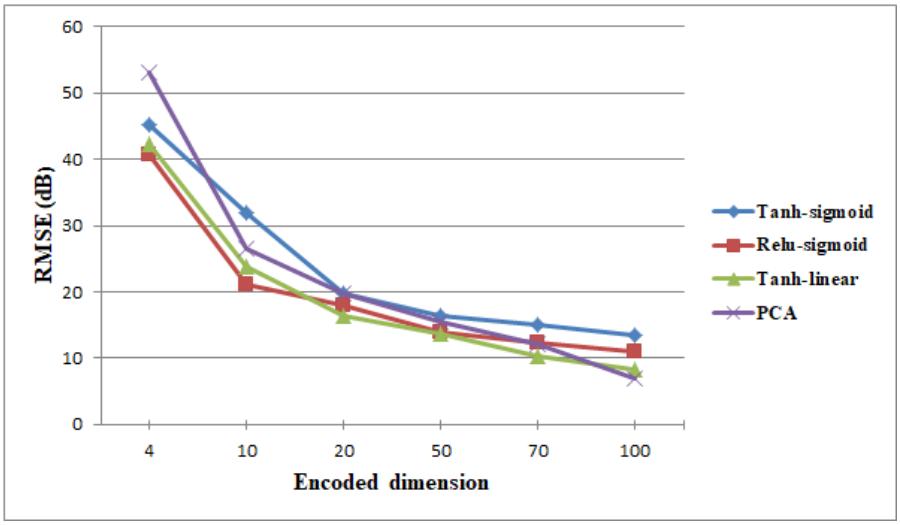


Fig. 10. RMSE measures of the autoencoder architecture ($[513, 256, encod, 256, 513]$) and the PCA

- The **Fig. 11** shows that PCA outperforms the AE model only in case of employing as activation functions the pairs (Tanh, Sigmoid) and (Relu, Sigmoid) where the RMSE decreases from 53.01 to 7.01.

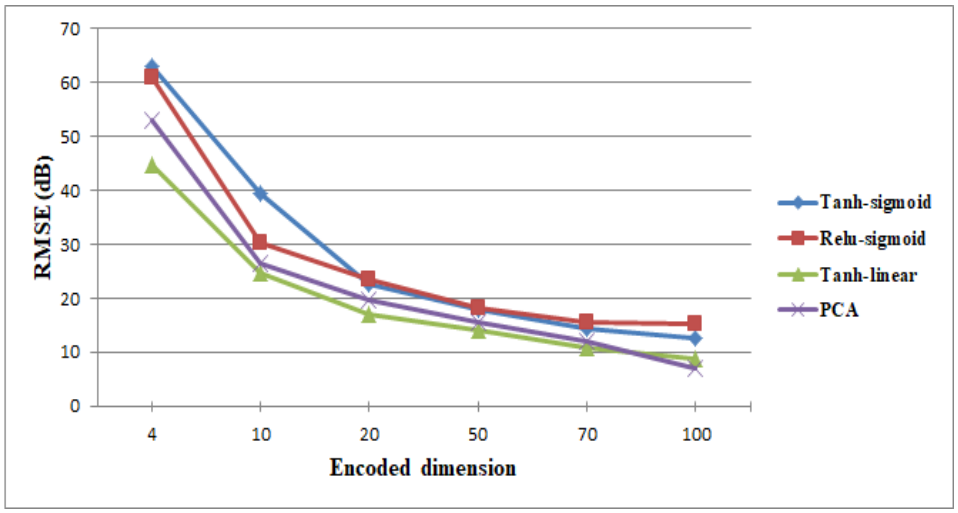


Fig. 11. RMSE evolution for the SAE architecture ($[513, encod, 513]$) and the PCA

In addition, it can be noticed that the PCA outperforms the three architectures of the AE model when the encoding dimension is high ($encod = 100$). Besides, the DAE model gives better results (lower RMSE) compared to the shallow AE (SAE) which proves the benefits of using deep architecture to extract high level features and that the RMSE decreases depending on the size of the latent vector for all models.

DAE vs VAE vs PCA: The **Fig. 12** shows the RMSE evolution for the PCA, DAE and the VAE models. For clarity, we only consider the case of using as activation functions the pairs

(Tanh, Linear). As we can see, The VAE outperforms the PCA and the DAE. For a low-dimensional latent vector (*encod* = 4, *encod* = 10, *encod* = 20) the DAE and the VAE can extract more relevant information from data compared to the PCA.

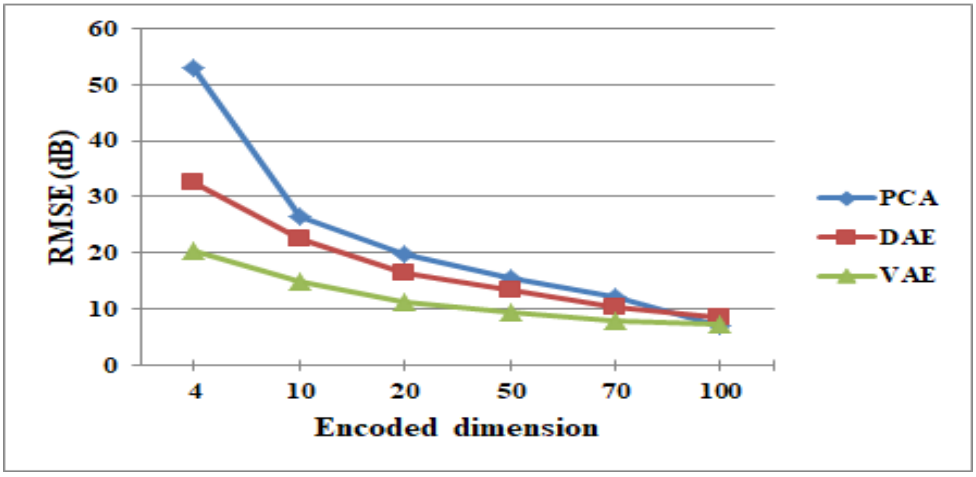
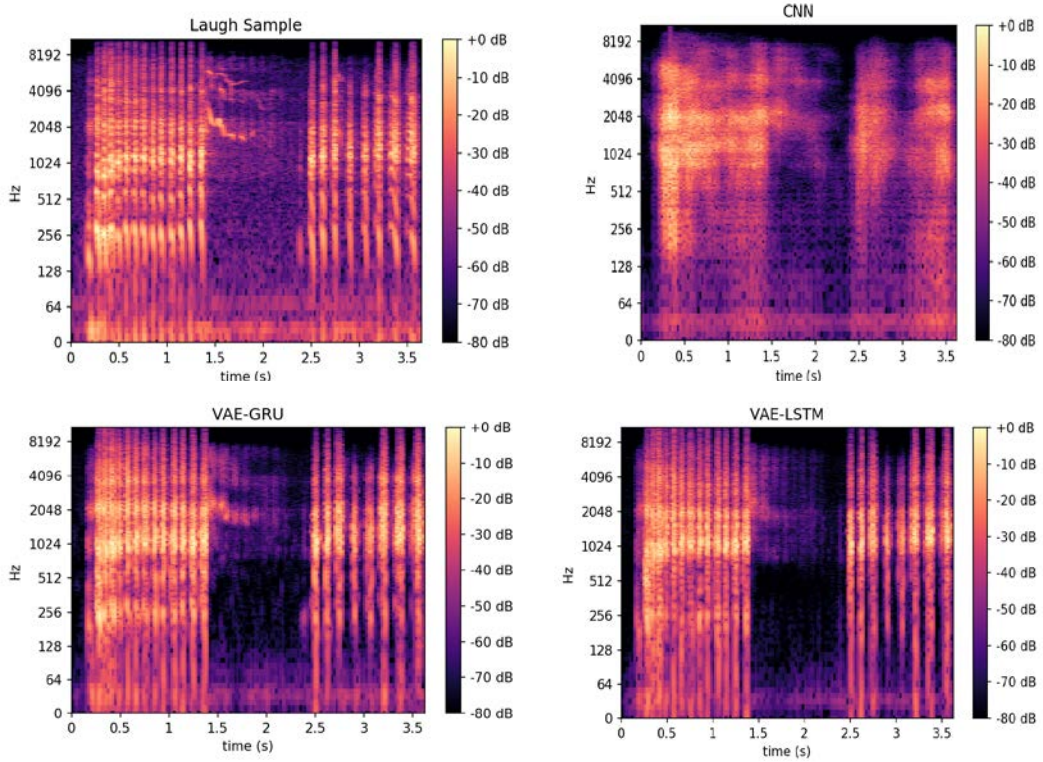


Fig. 12. RMSE evolution of the PCA, VAE and DAE architectures

The **Fig. 13** represents the log magnitude spectrogram of an original male laughter sound and those reconstructed from the differents architectures that we have discussed earlier. We can notice that the CNN-VAE model fails in reconstructing the log magnitude spectrogram, we can't differentiate between the harmonics which leads to the degradation of the laughter synthesized quality and the sound becomes unclear. However, the GRU-VAE model can be considered as the best model that succeeded in reconstructing very well the harmonics even the information that yield between them, compared to the other models.



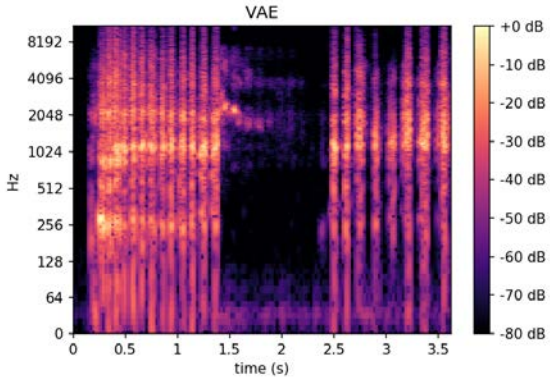


Fig. 13. Examples of original and reconstructed log magnitude spectrograms of a laughter male sound obtained from the CNN-VAE, GRU-VAE, LSTM-VAE and the standard VAE models

The RMSE (dB) overviews information about the acoustic features distortion. To figure out the audio perception quality, a listening test is suggested.

5.2 Subjective evaluation

The subjective evaluation is known by a listening test, consist of evaluating the quality of the synthesized laughter by given a score from 1 to 5 where 1 means very poor, 2 poor, 3 average, 4 good and 5 excellent. This test is carried out with the help of 10 subjects, five of them are a PhD students related to the field of signal processing. The participants aged between 24 and 32 years. Before starting the test, participants were structured to wear a headphones. For each architecture and for each gender, 10 samples of synthesized laughs were used for this evaluation task. In the end of this test, we collect a 100 scores for each configuration and we consider the mean score for the evaluation. This test is effectuated for all the models described in this paper. Just like in the previous section we'll analyse the results of the listening test for each model in a separate way.

VAE: **Table 1** gives an overview of the listening test results for the different architecture of the VAE model. For this test we examine the laughter synthesized with the two-dimensional latent vector *encod* = 100 and *encod* = 4. We also synthesized a male and a female laughter sound.

Table 1. Subjective results for different architectures of the VAE

	***encod* = 100**		***encod* = 4**	
	Male	Female	Male	Female
Tanh-Sigmoid	1.83	2.16	1	1.16
Relu-Sigmoid	3.16	3.33	1.33	1.66
Tanh-Linear	3.16	3.33	1.5	1.33
LSTM-VAE	2.16	2	1.83	1.33
GRU-VAE	3.33	4	1.83	1.83

As stated by **Table 1** (*encod* = 100), the GRU-VAE model outperforms the other architecture, whereas, the VAE possesses the same score for the pairs of activation function (Tanh, Linear) and (Relu, Sigmoid). These results are in accordance with the objective results. Concerning the CNN-VAE model, it gives inadequate results compared to the other models, where the score is equal to 1 for the female laughter sound and is equal to 1.2 for the male laughter sound.

DAE and PCA: **Table 2** represents the mean opinion score ratings of the autoencoder and the PCA model for the 2 latent vector dimension *encod* =100 and *encod* = 4. As stated by this table the DAE gives a better score, (4.16) in case of using (Tanh, Sigmoid) as activation function, compared to the PCA (3). The perception quality of the synthesized laughter decreases with the dimension of the latent vector.

Table 2. Subjective results for different architectures of the DAE and the PCA

	Rate	
	encod=100	*encod*=4
Tanh-Sigmoid	4.16	2.16
Relu-Sigmoid	4	2.33
Tanh-Linear	4	2.16
PCA	3	2.16

6. Conclusion

During this study, we have investigated the audio laughter generation process based on unsupervised generative models: Variational autoencoder and the autoencoder. These two models are considered as a dimensionality reduction technique, responsible of transforming the input data vector x from a low-level to a high-level representation (latent vector). The latent vector is employed by a decoder network to reconstruct the input vector $\hat{x}$. For this purpose, different configuration were examined and various combination between the VAE and other neural networks were performed such as: LSTM-VAE, GRU- VAE and CNN-VAE. According to our experiments, we can conclude that: 1) The DAE outperform the shallow AE and the PCA, 2) The GRU-VAE have successfully reconstructed the log magnitude spectrogram compared to the VAE, LSTM-VAE and CNN-VA, which leads to a better laughter audio quality perception, 3) The VAE gives better results than the DAE and the PCA especially for a low-dimensional latent vector (*encod* = 4).

References

[1] W. Chafe, "The importance of not being earnest: The feeling behind laughter and humor," *Phonetica*, pp. 192-197, Nov. 2011. Article (CrossRef Link)

[2] M. Soury and L. Devillers, "Smile and laughter in human-machine interaction: a study of engagement," in *Proc. of the 9th International Conference on Language Recourses and Evaluation (LREC)*, pp. 3633-3637, May 2014. Article (CrossRef Link)

[3] R. A. Martin, "Humor, laughter, and physical health: Methodological issues and research findings," *Psychological Bulletin*, vol. 127, no. 4, pp. 504-519, 2001. Article (CrossRef Link)

[4] N. Tits, K. El Haddad, and T. Dutoit, "Laughter Synthesis: Combining Seq2seq modeling with Transfer Learning," *arXiv:2008.09483*, 2020. Article (CrossRef Link)

[5] K. Haddad, I. Torre, E. Gilmartin, H. Cakmak, S. Dupont, T. Dutoit, and N. Campbell, "Introducing AmuS: The Amused Speech Database," in *Proc. of the 5th International Conference on Statistical Language and Speech Processing(SLSP)*, pp. 229-240, 2017. Article (CrossRef Link)

[6] G. N. Nguyen and T. N. Phung, "Reducing over-smoothness in HMM-based speech synthesis using exemplar-based voice conversion," *ERASIP Journal on Audio Speech and Music Processing*, vol. 14, June 2017. Article (CrossRef Link)

[7] J. Urbain, H. Çakmak, and T. Dutoit, "Evaluation of HMM-Based laughter synthesis," in *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing(ICASSP)*, pp. 7835-7839, May 2013. Article (CrossRef Link)

[8] T. Nagata and H. Mori, "Defining Laughter Context for Laughter Synthesis with Spontaneous Speech Corpus," *IEEE Transactions on Affective Computing*, vol. 11, no. 3, pp. 553-559, Sep. 2018. Article (CrossRef Link)

[9] H. Zen, A. Senior, and M. Schuster, "Statistical Parametric Speech Synthesis Using Deep Neural Networks," in *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing(ICASSP)*, pp. 7962-7966, 2013. Article (CrossRef Link)

[10] H. Cakmak, J. Urbain, J. Tilmanne, and T. Dutoit, "Evaluation of HMM-based visual laughter synthesis," in *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing(ICASSP)*, pp. 4578-4582, 2014. Article (CrossRef Link)

[11] H. Çakmak, K. Haddad, and T. Dutoit, "Audio-visual laughter synthesis system," in *Proc. of the 4th Interdisciplinary Workshop on Laughter and Other Non-Verbal Vocalisations in Speech*, pp. 11-14, Apr. 2015. Article (CrossRef Link)

[12] K. Haddad, H. Çakmak, S. Dupont, A. Moinet, and T. Dutoit, "An HMM Approach for Synthesizing Amused Speech with a Controllable Intensity of Smile," in *Proc. of International Symposium on Signal Processing and Information Technology(ISSPIT 2015)*, pp. 7-11, 2015. Article (CrossRef Link)

[13] K. Haddad, H. Çakmak, S. Dupont, and T. Dutoit, "Laughter and Smile Processing for Human-Computer Interactions," in *Proc. of Workshop Just Talking Casual Talk Among Humans and Machines*, pp. 21-25, May 2016. Article (CrossRef Link)

[14] S. A. Thati, K. Sudheer Kumar, and B. Yegnanarayana, "Synthesis of laughter by modifying excitation characteristics," *The Journal of the Acoustical Society of America*, vol. 133, 2013. Article (CrossRef Link)

[15] S. Sundaram and S. Narayanan, "Automatic acoustic synthesis of human- like laughter," *Journal of the Acoustical Society of America*, vol. 121, pp. 527-535, Jan. 2007. Article (CrossRef Link)

[16] N. Mansouri and Z. Lachiri, "DNN-based laughter synthesis," in *Proc. of International Conference on Control, Automation and Diagnosis (ICCAD)*, pp. 1-6, 2019. Article (CrossRef Link)

[17] H. Mori, T. Nagata, and Y. Arimoto, "Conversational and Social Laughter Synthesis with WaveNet," *Proceedings of Interspeech*, pp. 520-523, 2019. Article (CrossRef Link)

[18] C. Yu, R. E. Zezario, S. Wang, J. Sherman, Y. Hsieh, X. Lu, H. Wang, and Y. Tsao, "Speech Enhancement Based on Denoising Autoencoder With Multi-Branched Encoders," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2756-2769, 2020. Article (CrossRef Link)

[19] S. Takaki and J. Yamagishi, "A deep auto-encoder based low-dimensional feature extraction from FFT spectral envelopes for statistical parametric speech synthesis," in *Proc. of the 41st IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 5535-5539, 2016. Article (CrossRef Link)

[20] A. Sarroff and M. Casey, "Musical audio synthesis using auto-encoding neural nets" in *Proc. of the International Society for Music Information Retrieval Conference(ISMIR2014)*, 2014. Article (CrossRef Link)

[21] C. Doersch, "Tutorial on variational autoencoders," *arXiv:1606.05908*, 2016. Article (CrossRef Link)

[22] H. Huang, Z. Li, R. He, Z. Sun, and T. Tan, "IntroVAE: Introspective Variational Autoencoders for Photographic Image Synthesis," *arXiv:1807.06358*, pp. 52-63, 2018. Article (CrossRef Link)

[23] J. Engel, C. Resnick, A. Roberts, S. Dieleman, D. Eck, K. Simonyan, and M. Norouzi, "Neural audio synthesis of musical notes with WaveNet Autoencoders," *arXiv:1704.01279*, 2017. Article (CrossRef Link)

[24] F. Roche, T. Hueber, S. Limier, and L. Girin, "Autoencoders for music sound modeling: A comparison of linear, shallow, deep, recurrent and variational models," May 2019. Article (CrossRef Link)

[25] S. Leglaive, X. Alameda-Pineda, L. Girin, and R. Horaud, "A Recurrent Variational Autoencoder for Speech Enhancement," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing(ICASSP)*, pp. 371-375, May 2020. Article (CrossRef Link)

[26] M. Jang, S. Seo, and P. Kang, "Recurrent neural network-based semantic variational autoencoder for Sequence-to-sequence learning," *Information Sciences*, vol. 490, pp. 59-73, 2019. Article (CrossRef Link)

[27] N. Mansouri and Z. Lachiri, "Laughter synthesis: A comparison between Variational autoencoder and Autoencoder," in *Proc. of the* 5th *International Conference on Advanced Technologies for Signal and Image Processing(ATSIP)*, pp. 1-6, 2020. Article (CrossRef Link)

[28] M. Airaksinen, L. Juvela, B. Bollepalli, J. Yamagishi, and P. Alku, "A Comparison Between STRAIGHT, Glottal, and Sinusoidal Vocoding in Statistical Parametric Speech Synthesis," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 9, pp. 1658-1670, Sep. 2018. Article (CrossRef Link)

[29] M. Morise, F. Yokomori, and K. Ozawa, "WORLD: a vocoder-based high-quality speech synthesis system for real-time applications," *IEICE Transactions on Information and Systems*, vol. E99, no. 7, pp. 1877-1884, 2016. Article (CrossRef Link)

[30] T. Drugman and T. Dutoit, "The Deterministic Plus Stochastic Model of the Residual Signal and Its Applications," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 3, pp. 968-981, Mar. 2012. Article (CrossRef Link)

[31] N. Perraudin, P. Balazs, and P. L. Søndergaard, "A fast Griffin-Lim algorithm," in *Proc. of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 1-4, 2013. Article (CrossRef Link)

[32] D. M. Blei, A. Kucukelbir, and J. D. McAuliffe, "Variational inference: A review for statisticians," *Journal of the American Statistical Association*, vol. 112, pp. 859-877, 2017. Article (CrossRef Link)

[33] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," in *Proc. of the International Conference on Learning Representations(ICLR-14), arXiv:1312.6114*, 2014. Article (CrossRef Link)

[34] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner, "β-VAE: learning basic visual concepts with a constrained variational framework," in *Proc. of International Conference on Learning Representations*, 2017. Article (CrossRef Link)

[35] D. P. Kingma, T. Salimans, and M. Welling, "Variational dropout and the local reparameterization trick," *arXiv:1506.02557*, 2015. Article (CrossRef Link)

[36] S. Achanta, T. Godambe, and S. V. Gangashetty, "An Investigation of Recurrent Neural Network Architectures for Statistical Parametric Speech Synthesis," in *Proc. of the Conference of INTERSPEECH*, 2015. Article (CrossRef Link)

[37] J. Chien and C. Wang, "Variational and Hierarchical Recurrent Autoencoder," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing(ICASSP)*, pp. 3202-3206, 2019. Article (CrossRef Link)

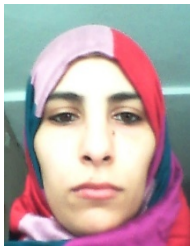
[38] J. Chung, K. Kastner, L. Dinh, K. Goel, A. C. Courville, and Y. Bengio, "A recurrent latent variable model for sequential data," *Advances in Neural Information Processing Systems*, pp. 2980-2988, 2015. Article (CrossRef Link)

[39] S. Lin, R. Clark, R. Birke, S. Schönborn, N. Trigoni, and S. Roberts, "Anomaly Detection for Time Series Using VAE-LSTM Hybrid Model," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing(ICASSP)*, pp. 4322-4326, 2020. Article (CrossRef Link)

[40] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv:1409.1556*, 2014. Article (CrossRef Link)

[41] J. Urbain, R. Niewiadomski, E. Bevacqua, T. Dutoit, A. Moinet, R. Niewiadomski, C. Pelachaud, B. Picart, J. Tilmanne, and J. Wagner, "The AVLaughterCycle database," in *Proc. of the International Conference on Language Resources and Evaluation(LREC'10)*, pp. 47-58, May 2010. Article (CrossRef Link)

[42] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv:1412.6980*, 2014. Article (CrossRef Link)



Nadia Mansouri is a PhD student at the National Engineering School of Tunisia. She recieived her B.Eng and M.S degree in Electrical and Automation Engineering from the National Engineering School of Gabes in 2012. Her research interest includes signal processing, speech synthesis, deep learning.

Zied Lachiri was born in Tunis, Tunisia. He received the M.S. degree in automatic and signal processing and the PHD. Degree in Electrical Engineering from the National Engineering School of Tunis (ENIT-Tunisia), in 1997 and 2002, respectively. He joined National Engineering School of Tunis in the fall of 2015, where he is presently serving as Professor of Electrical Engineering, as well as Director of Signal, Image and Information Technology Research Laboratory (LR-SITI, ENIT). From 2002 until 2015, he was with the National Institute of Applied Sciences and Technology (INSAT) where he served as Department Chairman, Assistant Professor, Associate Professor and Professor in the Department of Physic and Instrumentation. His research interests span the areas of digital speech processing, machine learning and pattern recognition, signal processing and image processing applied in biomedical, multimedia and man machine communication. He is the author/coauthor of several publications including journals, book chapters and conference papers.