AI-Synthesized Voices Clusterization using Bispectral Analysis and Dimensionality Reduction

Aline Gabriel de Almeida* and Alexandre da Silva Simões†

*, †*São Paulo State University (UNESP)*
Institute of Science and Technology of Sorocaba (ICTS)
Campus of Sorocaba, Brazil

Email: **aline.gabriel.almeida@gmail.com, †alexandre.simoes@unesp.br*

***Abstract*—With the recent advances in voice synthesis, AI-synthesized speeches are indistinguishable to human ears and are widely applied to produce realistic and natural DeepFakes, exhibiting useful applications, but also creating real threats to our society. This paper proposes the use of bispectral analysis and statistical tools to create features capable to catch the difference between human and synthesized voices. We show that is possible to visually distinguish the clusters of human and synthesized voices by inspecting a 2-D plot constructed using a dimensional reduction technique.**

***Keywords*-Dimensionality reduction, Deepfake, clusterization, bispectral analysis, audio synthesis.**

I. Introduction

Recent advances in AI-synthesized content-generation are leading to the creation of highly realistic media like voices [1], images [2] and videos [3], known as DeepFakes.

An DeepFake audio is when a “cloned” voice that is potentially indistinguishable from the real person’s is used to produce synthetic audio.

Most DeepFake audios are intended for helpful purposes as they can enable applications, such as transferring a voice across languages for more natural speech-to-speech translation [4], or generating realistic speech from text in low resource settings [5].

However it could potentially also be used for harmful intents, like identity theft and misinformation exacerbation [6].

Many approaches and efforts have risen trying to identify these manipulated media, like the use of emotion recognition [7], or cepstral and bispectral statistics analysis [8].

This paper proposes one approach for a visual clusterization and inspection of synthesized and real speeches using the bispectral statistics analysis combined with dimensionality reduction.

In the next sections we present the main aspects of the techniques we used to extract the features and perform the dimensionality reduction of the data. Then we explain our approach to cluster and plot the data, and show our results and conclusions.

II. Literature Review

A. Dimensionality reduction

Dimensionality reduction is widely utilized to facilitate data exploration and visual analysis.

The key to the success of the dimensionality reduction and visualization is the ability to preserve local geometry of highly nonlinear manifolds in high-dimensional spaces and properly unfold them into lower dimensional hyperplanes.

One of the most popular high-dimensional data visualization methods to date is the *Uniform Manifold Approximation and Projection for Dimension Reduction (UMAP)* [9].

The UMAP algorithm (see *Algorithm 1*) constructs a high-dimensional graph representation of the data and then optimizes a low-dimensional graph to be as structurally similar as possible.

Algorithm 1: UMAP algorithm

```
Function UMAP ($X, n, d, min\text{-}dist, n\text{-}epochs$):
    # Constructs a high-dimensional graph
    for $x \in X$ do
        $fs\text{-}set[x] \leftarrow$
          $LocalFuzzySimplicialSet(X, x, n)$
    end
    $top\text{-}rep \leftarrow \cup_{x \in X} fs\text{-}set[x]$
    # Optimization of a low-dimensional graph
    $Y \leftarrow SpectralEmbedding(top\text{-}rep, d)$
    $Y \leftarrow OptimizeEmbedding(top\text{-}rep, Y, min\text{-}dist,$
    $n\text{-}epochs)$
    return $Y$
```

The inputs to the algorithm are:

- X: the data set to have dimension reduced

- n: the neighborhood size to use for local metric approximation
- d: the dimension of the target reduced space
- $min\text{-}dist$: an algorithmic parameter controlling the layout
- $n\text{-}epochs$: the parameter that controls the amount of optimization work to perform.

1) High-dimensional graph construction: In order to construct the initial high-dimensional graph, UMAP algorithm builds something called a fuzzy simplicial set using the variable X as the data set and the n as the neighborhood size.

A simplicial set is made up of simplices, where 0-simplices are vertices, 1-simplices are edges, 2-simplices are triangles, 3-simplices tetrahedras, etc., as shown in Figure 1.

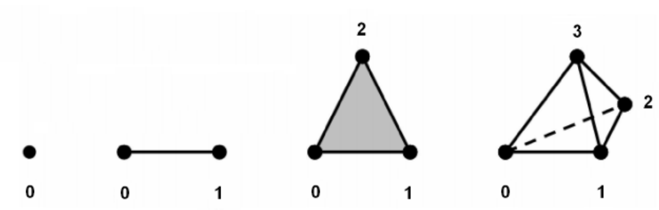

Figure 1. Examples of 0-, 1-, 2-, and 3-simplices [10]

A fuzzy simplicial set is a generalization of a simplicial set and can be represented as a weighted graph with edge weights representing the likelihood that two points are connected (see Figure 2).

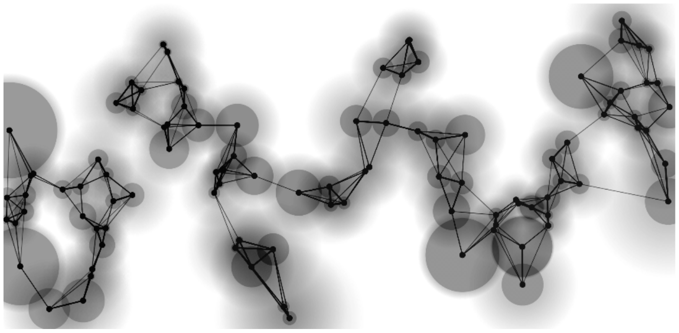

Figure 2. Examples of a fuzzy simplicial set, where the edges thickness represent the likelihood of two points are connected [11]

It also stipulate that each point must be connected to at least its closest neighbor, thus UMAP ensures that local structure is preserved in balance with global structure.

2) Low-dimensional graph optimization: Once the high-dimensional graph is constructed, UMAP algorithm optimizes the layout of a low-dimensional analogue (with d dimensions) to be as similar as possible.

The $min\text{-}dist$ parameter shown in the *Algorithm 1* is used to control the balance between local and global structure in the final projection.

In practice UMAP uses a force directed graph layout algorithm in low dimensional space that utilizes of a set of attractive forces applied along edges and a set of repulsive forces applied among vertices.

In UMAP the attractive force between two vertices i and j at coordinates y_i and y_j respectively, is determined by:

$$\frac{-2ab||y_i - y_j||_2^{2(b-1)}}{1 + ||y_i - y_j||_2^2} w((x_i, x_j))(y_i - y_j), \tag{1}$$

where a and b are hyper-parameters and w the weights.

The repulsive forces are given by:

$$\frac{2b}{(\epsilon + ||y_i - y_j||_2^2)(1 + a||y_i - y_j||_2^{2b})}(1 - w((x_i, x_j)))(y_i - y_j), \tag{2}$$

where ϵ is a small number to prevent division by zero.

The algorithm proceeds by iteratively applying attractive and repulsive forces, (it is referred by the $OptimizeEmbedding$ function) at each edge or vertex, until reach the number of $n\text{-}epochs$.

B. Bispectral Analysis

The bispectral analysis is a method of signal processing that quantifies the degree of phase coupling between the components of a signal.

The bispectrum of a signal represents higher-order correlations in the Fourier domain. It is defined as:

$$B(\omega_1, \omega_2) = Y(\omega_1)Y(\omega_2)Y^*(\omega_1 + \omega_2), \tag{3}$$

where B is the complex-valued bispectrum quantity, Y denotes the Fourier transform of the signal and ω_1 and ω_2 are the harmonic frequencies.

Given that the bispectrum is simply a complex number, it can be thought of as consisting of a magnitude:

$$|B(\omega_1, \omega_2)| = |Y(\omega_1)|.|Y(\omega_2)|.|Y(\omega_1 + \omega_2)|, \tag{4}$$

and a phase:

$$\angle B(\omega_1, \omega_2) = \angle Y(\omega_1) + \angle Y(\omega_2) - \angle Y(\omega_1 + \omega_2). \tag{5}$$

From an interpretive stance it is helpful to work with the normalized bispectrum, called the bicoherence of a signal:

$$B_c(\omega_1, \omega_2) = \frac{Y(\omega_1)Y(\omega_2)Y^*(\omega_1 + \omega_2)}{\sqrt{|Y(\omega_1)Y(\omega_2)|^2 |Y*(\omega_1 + \omega_2)|^2}}, \tag{6}$$

This normalized bispectrum yields magnitudes in the range [0, 1].

III. Proposed approach

The usage of deep-neural network based systems for generating AI-synthesized voices introduce specific and unusual spectral correlations in the audio signal not typically found in human speech. These correlations can be measured using tools from bispectral analysis, to distinguish human from synthesized speech. The proposed approach is to apply bispectral statistical analysis to human and synthesized speeches and try to visually separate them using UMAP dimensionality reduction.

IV. Materials and methods

We begin by describing the creation of the data set that have human and synthesized speeches. We then describe the bispectral statistical analysis performed in order to create the data features. We conclude this section with a description of the clusterization and visualization methods applied that could allow the visual distinction of the human from AI-synthesized speeches, in a 2D plot.

A. Data set

We downloaded the YouTube videos [12]–[17] and create a set of audio files composed of 15 female speech files and 15 male speech files, lasting 10 seconds each, to form the human voices data set.

For the synthesized audio generation we used the *Transfer Learning from Speaker Verification to Multispeaker Text-To-Speech Synthesis (SV2TTS)* [18] architecture implementation [19], that can generate AI-synthesized natural speeches for taking just a few seconds of the voice from a target speaker to synthesize a new speech in that speaker's voice.

Using SV2TTS system we generate 30 synthesized speeches (15 male and 15 female), using each audio from the human data set as the input to the audio generation system.

Our final data set is structured as shown in table I.

Table I
Audio data set

	Number of 10 seconds speech files	
	Male	Female
Human	15 files	15 files
AI-synthesized	15 files	15 files

B. Bispectral features

Starting with the raw audio files from the data set, we use the *Stingray library* [20] to calculate the normalized bispectrum of the signal, for each human and synthesized speech, from which the magnitude and phase are computed.

We then employ the following statistical tools to get the features for each bispectral value:

- Average of the magnitude component;
- Average of the phase component;
- Standard deviation of the magnitude component;
- Standard deviation of the phase component;
- Maximum value of the magnitude component.

Therefore, we ended up with 5 bispectral-based features for describing each audio file.

C. Visualization and dimensionality reduction

For the visualization end we use the *UMAP library* [21] to reduce the dimensionality of our feature-based data set, from 5 dimensions to 2 dimensions, so it can be easily visually inspected with the goal of differentiate between human and AI synthesized speech.

We finally employ a grid search hyper-parameter optimization scheme to find the best visualization plots.

V. Results and discussion

Shown in the Figures 3 and 4 are the normalized bispectrum magnitudes and phases of a human speech (Figure3) and a synthesized voice (Figure4). The magnitude plots are displayed on an intensity scale of [0, 1] and the phase plots are displayed on a scale of [-π, π].

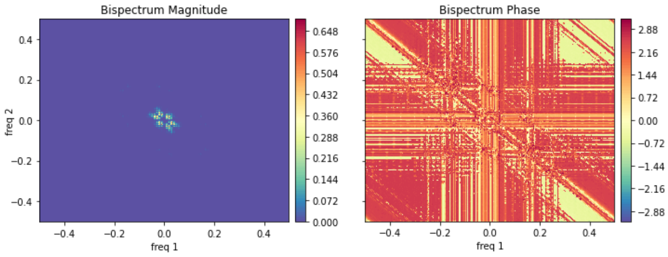

Figure 3. Bispectrum magnitude and phase for a human speaker.

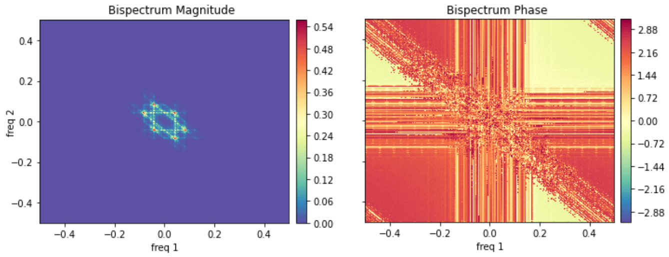

Figure 4. Bispectrum magnitude and phase for a synthesized voice.

The Figure 5 shows a scatter plot of the 2-dimensions reduced plot of the human speeches and the synthesized speeches, where we vary the UMAP hyper-parameter to find the different clusterization results.

The green circles correspond to human speeches and the remaining pink circles correspond to synthesized voices. Even in this reduced dimensional space, we can see the human speech cluster is distinct from the synthesized speech.

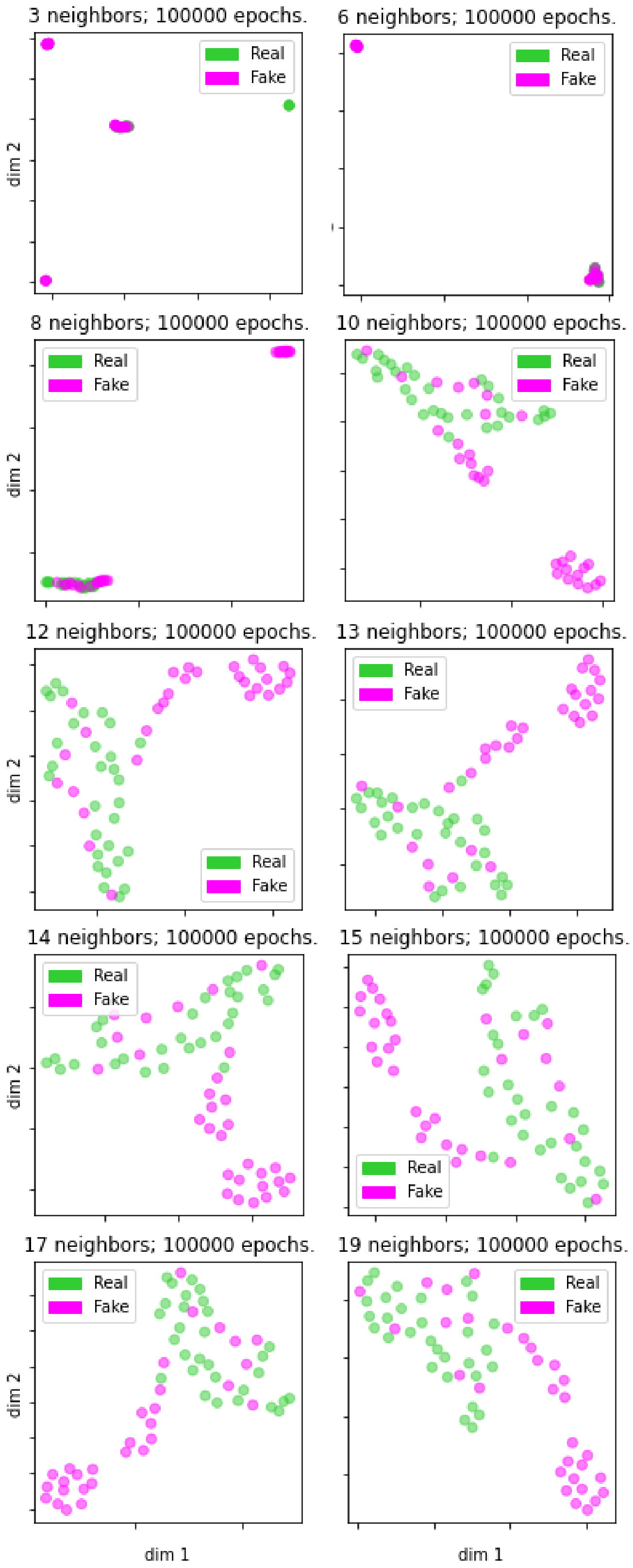

Figure 5. A 2-D plot of the dimension reduced bispectral features of the real and synthesized audios.

VI. Conclusions

In this paper, we proposed to use dimensionality reduction and bispectral statistics analysis to clusterize and discern AI-synthesized voices from human speeches.

Experiments on the data set demonstrate its effectiveness, as shown in the 2D-plots, that it is possible to visually separate the real and the synthesised clusters, even though there are some overlap.

Also, interestingly, there are synthesized media that were mistakenly classified as real, but there are not human speeches that were classified as fake, because the real media did not present the same high-order correlation patterns that were found in the synthesized media.

References

[1] Dessa, "Realtalk: We recreated joe rogan's voice using artificial intelligence," 2019. [Online]. Available: https://www.theverge.com/2019/5/17/18629024/joe-rogan-ai-fake-voice-clone-deepfake-dessa

[2] P. Wang, "This person does not exist," 2019. [Online]. Available: https://thispersondoesnotexist.com

[3] A. Amini, "Barack obama: Intro to deep learning," 2020. [Online]. Available: https://youtu.be/l82PxsKHxYc

[4] Y. Jia, R. J. Weiss, F. Biadsy, W. Macherey, M. Johnson, Z. Chen, and Y. Wu, "Direct speech-to-speech translation with a sequence-to-sequence model," 2019.

[5] Y. Yang, B. Shillingford, Y. Assael, M. Wang, W. Liu, Y. Chen, Y. Zhang, E. Sezener, L. C. Cobo, M. Denil, Y. Aytar, and N. de Freitas, "Large-scale multilingual audio visual dubbing," 2020.

[6] W. A. Galston, *Is Seeing Still Believing? The Deepfake Challenge to Truth in Politics.* Brookings, www.brookings.edu/research/is-seeingstill-believing-the-deepfake-challenge-to-truth-in-politics, 2020.

[7] T. Mittal, U. Bhattacharya, R. Chandra, A. Bera, and D. Manocha, "Emotions don't lie: An audio-visual deepfake detection method using affective cues," 2020.

[8] A. K. Singh and P. Singh, "Detection of ai-synthesized speech using cepstral and bispectral statistics," 2020.

[9] L. McInnes, J. Healy, and J. Melville, "Umap: Uniform manifold approximation and projection for dimension reduction," 2020.

[10] G. Friedman, "An elementary illustrated introduction to simplicial sets," 2016.

[11] A. P. Andy Coenen, "Understanding umap," 2019. [Online]. Available: https://pair-code.github.io/understanding-umap

[12] TED, "The mathematics of love - hannah fry," 2020. [Online]. Available: https://youtu.be/yFVXsjVdvmY

[13] DeepMind, "Deepmind: The podcast - ep. 1: Ai and neuroscience - the virtuous circle," 2020. [Online]. Available: https://youtu.be/ExrXs7PCQpU

[14] A. for Project Management, “Apm conference 2018 - hannah fry,” 2020. [Online]. Available: https://youtu.be/Y4_fnAosulo

[15] NPR, “Npr’s interview with president obama about ’obama’s years’ - morning edition - npr,” 2020. [Online]. Available: https://youtu.be/jkfiF_tugU0

[16] N. News, “President barack obama’s greatest speeches - nbc news,” 2020. [Online]. Available: https://youtu.be/hWLf6JFbZoo

[17] C-Span, “C-span: President-elect barack obama victory speech (full video),” 2020. [Online]. Available: https://youtu.be/jJfGx4G8tjo?list=FLm_pfad72C6vhz6NDLGQfCQ

[18] Y. Jia, Y. Zhang, R. J. Weiss, Q. Wang, J. Shen, F. Ren, Z. Chen, P. Nguyen, R. Pang, I. L. Moreno, and Y. Wu, “Transfer learning from speaker verification to multispeaker text-to-speech synthesis,” 2019.

[19] G. Louppe, “Master thesis : Automatic multispeaker voice cloning,” 2019.

[20] D. Huppenkothen, M. Bachetti, A. L. Stevens, S. Migliari, P. Balm, O. Hammad, U. M. Khan, H. Mishra, H. Rashid, S. Sharma, and et al., “Stingray: A modern python library for spectral timing,” *The Astrophysical Journal*, vol. 881, no. 1, p. 39, Aug 2019. [Online]. Available: http://dx.doi.org/10.3847/1538-4357/ab258d

[21] L. McInnes, J. Healy, N. Saul, and L. Grossberger, “Umap: Uniform manifold approximation and projection,” *The Journal of Open Source Software*, vol. 3, no. 29, p. 861, 2018.