

Sistem automat de reducere a reverberației și eliminare a zgomotului din semnale audio

Răzvan-Paul Ciubotaru

Conducători:

Prof. Univ. Dr. Ing. Corneliu Burileanu

As. Univ. Drd. Ing. Ana-Antonia Neacșu

Facultatea de Electronică, Telecomunicații și Tehnologia Informației

iulie 2021

Cuprins

- 1 Introducere
- 2 Noțiuni teoretice
- 3 Eliminarea efectului de reverberație
- 4 Eliminarea zgomotului folosind filtre digitale
- 5 Eliminarea zgomotului folosind rețele neurale
- 6 Rezultate experimentale
- 7 Concluzii

Introducere

Motivație și aplicabilitate

Motivație

- Găsirea unor metode de îmbunătățire a calității semnalelor audio precum vocea umană sau semnale muzicale;
- Evoluția domeniului de inteligență artificială.

Motivație și aplicabilitate

Motivație

- Găsirea unor metode de îmbunătățire a calității semnalelor audio precum vocea umană sau semnale muzicale;
- Evoluția domeniului de inteligență artificială.

Aplicabilitate

- Evenimente precum concerte live sau spectacole muzicale;
- Îmbunătățirea calității semnalelor vocale.

Objective

- Înlăturarea efectului reverberației folosind un algoritm iterativ;
- Crearea bazei de date de lucru pornind de la melodiile de reverberate;
- Eliminarea zgomotului folosind sisteme bazate pe metode clasice și pe metode de învățare automată;
- Testarea performanței acestor sisteme și compararea rezultatelor.

Noțiuni teoretice

Semnale audio

- Spectrul de frecvențe cuprins în gama [20Hz – 20KHz].
- Reprezentare duală timp - frecvență

Semnale audio

- Spectrul de frecvențe cuprins în gama [20Hz – 20KHz].
- Reprezentare duală timp - frecvență

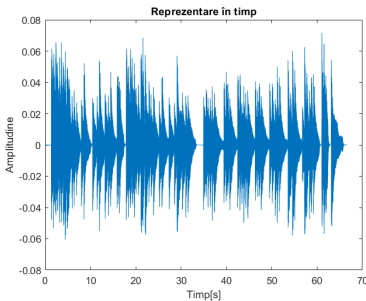


Figura 1: Reprezentare în timp a semnalului audio

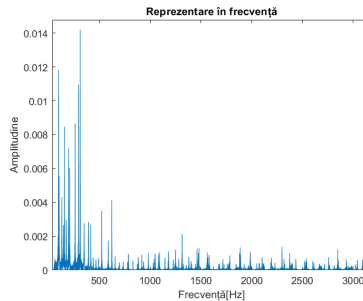


Figura 2: Reprezentarea în frecvență a semnalului audio

Zgomot alb Gaussian

Variabilă aleatoare modelată după o densitate de probabilitate a unei distribuții gaussiene:

$$f_X(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{(x - \mu)^2}{2\sigma^2}\right] \tag{1}$$

În cazul zgomotului Gaussian standard ($\mu = 0, \sigma = 1$):

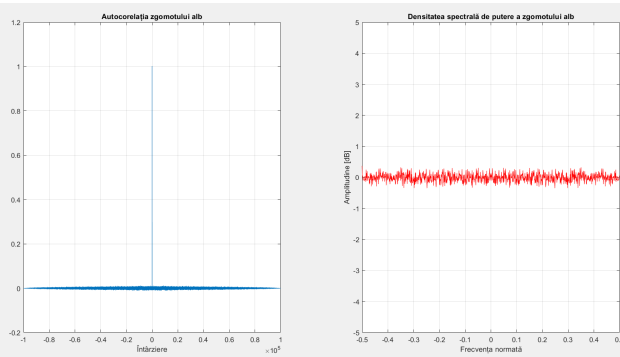


Figura 3: Autocorelația și densitatea de probabilitate a zgomotului gaussian standard.

Aspecte teoretice ale rețelelor neurale

Sisteme inteligente ce se adaptează prin învățarea din experiență.

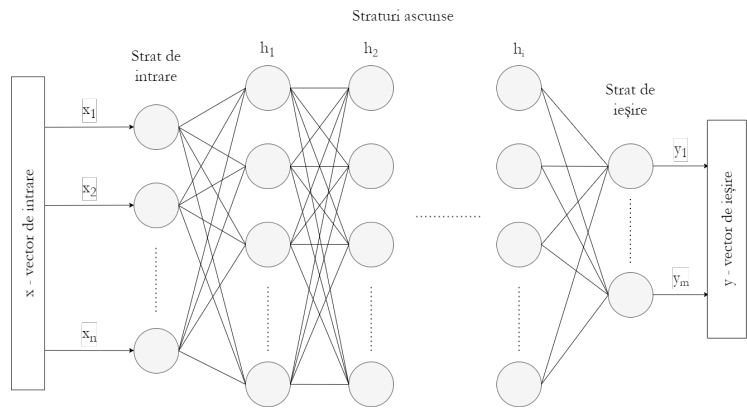


Figura 4: Simplu exemplu de rețea neurală.

Eliminarea efectului de reverberație

Modelarea semnalelor zgomotoase

Plecând de la un semnal curat s și presupunând un sistem liniar și invariant în timp, expresia matematică a unui semnal reverberat și zgomotos $x(k)$ este:

$$x(k) = (s * h)(k) + n(k)$$

$h \rightarrow$ răspunsul la impuls al încăperii¹

$$n \in \mathcal{N}(\mu = 0; \sigma^2 = ct)$$

σ ales așa încât $SNR \in [5, 30]dB$

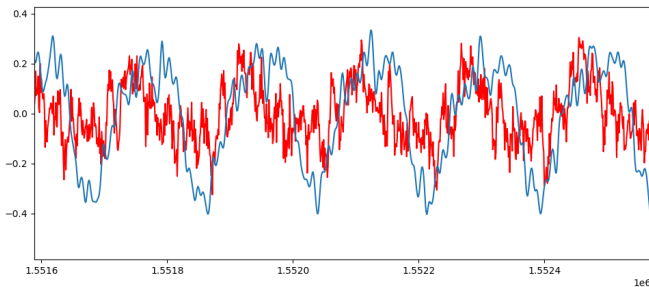


Figura 5: Semnal curat vs. semnal reverberat și zgomotos

¹<https://www.iks.rwth-aachen.de/en/research/tools-downloads/databases/>

Algoritm de deverberare

Problema deverberării semnalelor audio se reduce la minimizarea funcției obiectiv:

$$\underset{\hat{s}}{\text{minimize}} \quad \frac{1}{2} |H\hat{S} - X|^2 + \varphi(\hat{s})$$

$$\varphi(\hat{s}) = \lambda \sum_k |\hat{s}(k)| \rightarrow \text{funcție de cost}$$

$$X = \mathcal{F}(x)$$

$$\lambda \rightarrow \text{parametru de regularizare}$$

$$\hat{S} = \mathcal{F}(\hat{s})$$

Algoritm de deverberare

Problema deverberării semnalelor audio se reduce la minimizarea funcției obiectiv:

$$\underset{\hat{s}}{\text{minimize}} \quad \frac{1}{2} |H\hat{S} - X|^2 + \varphi(\hat{s})$$

$$\varphi(\hat{s}) = \lambda \sum_k |\hat{s}(k)| \rightarrow \text{funcție de cost}$$

$$X = \mathcal{F}(x)$$

$$\lambda \rightarrow \text{parametru de regularizare}$$

$$\hat{S} = \mathcal{F}(\hat{s})$$

Se propune algoritmul:

$$s_{n+1} = \text{prox}_{\gamma\lambda}(s_n - \gamma \mathcal{F}^{-1}\{H^*(HS_n - X)\})$$

$H \rightarrow$ funcția de transfer a camerei

$$S_n = \mathcal{F}\{s_n\} \quad \gamma = 1.9 / \max |H|^2$$

$\gamma \rightsquigarrow$ pasul algoritmului

$\gamma\lambda \rightsquigarrow$ valoarea de prag

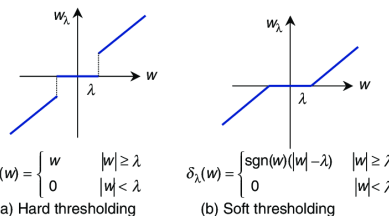


Figura 6: Soft-Thresholding vs. Hard-Thresholding

Rețele neuronale nonexpansive

O rețea neurală formată din m straturi poate fi modelată după legea matematică:

$$T = T_m \circ \dots \circ T_1$$

$(\forall i \in \{1, \dots, m\})$

$$T_i(x_i) = R_i(W_i x_i + b_i)$$

$x_i \in \mathbb{R}^{N_{i-1}}$ – vector de intrare al stratului i

$b_i \in \mathbb{R}^{N_i}$ – termenul liber

$W_i \in \mathbb{R}^{N_i \times N_{i-1}}$ – matrice pondere

$R_i: \mathbb{R}^{N_i} \rightarrow \mathbb{R}^{N_i}$ – operatorul de activare

Rețele neuronale nonexpansive

O rețea neurală formată din m straturi poate fi modelată după legea matematică:

$$T = T_m \circ \dots \circ T_1$$

$$(\forall i \in \{1, \dots, m\}) \quad T_i(x_i) = R_i(W_i x_i + b_i)$$

$x_i \in \mathbb{R}^{N_{i-1}}$ – vector de intrare al stratului i

$b_i \in \mathbb{R}^{N_i}$ – termenul liber

$W_i \in \mathbb{R}^{N_i \times N_{i-1}}$ – matrice pondere

$R_i: \mathbb{R}^{N_i} \rightarrow \mathbb{R}^{N_i}$ – operatorul de activare

Presupunem x - date de intrare și z - perturbație aleatoare. Atunci:

$$\|T(x+z) - T(x)\| \leq \theta_m \|z\| \quad \theta_m - \text{constanta Lipschitz}$$

Rețele neurale nonexpansive

O rețea neurală formată din m straturi poate fi modelată după legea matematică:

$$T = T_m \circ \dots \circ T_1$$

$$(\forall i \in \{1, \dots, m\}) \quad T_i(x_i) = R_i(W_i x_i + b_i)$$

$x_i \in \mathbb{R}^{N_{i-1}}$ – vector de intrare al stratului i
 $b_i \in \mathbb{R}^{N_i}$ – termenul liber

$W_i \in \mathbb{R}^{N_i \times N_{i-1}}$ – matrice pondere
 $R_i: \mathbb{R}^{N_i} \rightarrow \mathbb{R}^{N_i}$ – operatorul de activare

Presupunem x - date de intrare și z - perturbație aleatoare. Atunci:

$$\|T(x+z) - T(x)\| \leq \theta_m \|z\| \quad \theta_m - \text{constanta Lipschitz}$$

Observație

$$\text{Rețea nonexpansivă} \iff \theta_m \leq 1$$

Aproximarea constantei Lipschitz

O aproximare grosieră a constantei Lipschitz este dată de formula:

$$\theta_m = \prod_{i=1}^m \|W_i\|_s$$

m - numărul de straturi

$\|W_i\|_s$ - norma spectrală a fiecărei ponderi

Aproximarea constantei Lipschitz

O aproximare grosieră a constantei Lipschitz este dată de formula:

$$\theta_m = \prod_{i=1}^m \|W_i\|_s$$

m - numărul de straturi

$\|W_i\|_s$ - norma spectrală a fiecărei ponderi

Dacă ponderile sunt **pozitive** și operatorii de activare sunt **nonexpansivi**, atunci:

$$\theta_m = \|W_m \dots W_1\|_s$$

Constrângeri propuse

■ Ponderi pozitive:

$$(\forall i \in \{1, \dots, m\}) \quad W_i \in \mathcal{D}_i$$

$$\text{unde } \mathcal{D}_i = [0, +\infty]^{N_i \times N_{i-1}}$$

Constrângeri propuse

■ Ponderi pozitive:

$$(\forall i \in \{1, \dots, m\}) \quad W_i \in \mathcal{D}_i$$
$$\text{unde } \mathcal{D}_i = [0, +\infty]^{N_i \times N_{i-1}}$$

■ Controlul constantei Lipschitz:

$$\mathcal{C}_i^1 = \{W_i \mid \|W_i\|_s \leq \theta_m, \forall i \in \{1, \dots, m\}\}$$

$$\mathcal{C}_i^2 = \{W_i \mid \prod_{i=1}^m \|W_i\|_s \leq \theta_m\}$$

$$\mathcal{C}_i^3 = \{W_i \mid \|W_m \dots W_1\|_s \leq \theta_m\}$$

Constrângeri propuse

- Ponderi pozitive:

$$(\forall i \in \{1, \dots, m\}) \quad W_i \in \mathcal{D}_i$$

$$\text{unde } \mathcal{D}_i = [0, +\infty]^{N_i \times N_{i-1}}$$

- Controlul constantei Lipschitz:

$$\mathcal{C}_i^1 = \{W_i \mid \|W_i\|_s \leq \theta_m, \forall i \in \{1, \dots, m\}\}$$

$$\mathcal{C}_i^2 = \{W_i \mid \prod_{i=1}^m \|W_i\|_s \leq \theta_m\}$$

$$\mathcal{C}_i^3 = \{W_i \mid \|W_m \dots W_1\|_s \leq \theta_m\}$$

- Setul propus pentru implementarea constrângerilor menționate:

$$\mathcal{C}_i = \{W_i \in \mathbb{R}^{N_i \times N_{i-1}} \mid \|A_i W_i B_i\|_s \leq \theta_m\}$$

$$A_i = W_m \dots W_{i+1},$$

$$B_i = W_{i-1} \dots W_1$$

Constrângeri propuse

- **Ponderi pozitive:**

$$(\forall i \in \{1, \dots, m\}) \quad W_i \in \mathcal{D}_i$$

$$\text{unde } \mathcal{D}_i = [0, +\infty]^{N_i \times N_{i-1}}$$

- **Controlul constantei Lipschitz:**

$$\mathcal{C}_i^1 = \{W_i \mid \|W_i\|_s \leq \theta_m, \forall i \in \{1, \dots, m\}\}$$

$$\mathcal{C}_i^2 = \{W_i \mid \prod_{i=1}^m \|W_i\|_s \leq \theta_m\}$$

$$\mathcal{C}_i^3 = \{W_i \mid \|W_m \dots W_1\|_s \leq \theta_m\}$$

- **Setul propus pentru implementarea constrângerilor menționate:**

$$\mathcal{C}_i = \{W_i \in \mathbb{R}^{N_i \times N_{i-1}} \mid \|A_i W_i B_i\|_s \leq \theta_m\}$$

$$A_i = W_m \dots W_{i+1},$$

$$B_i = W_{i-1} \dots W_1$$

- **Proiecția ponderilor pe spațiul $\mathcal{C}_i \cap \mathcal{D}_i$: $\tilde{P}_{\mathcal{C}_i \cap \mathcal{D}_i}(W_i) \simeq P_{\mathcal{D}_i}(W_i) \circ P_{\mathcal{C}_i}(W_i)$**

Eliminarea zgomotului folosind filtre digitale

Filtrarea semnalelor zgomotoase

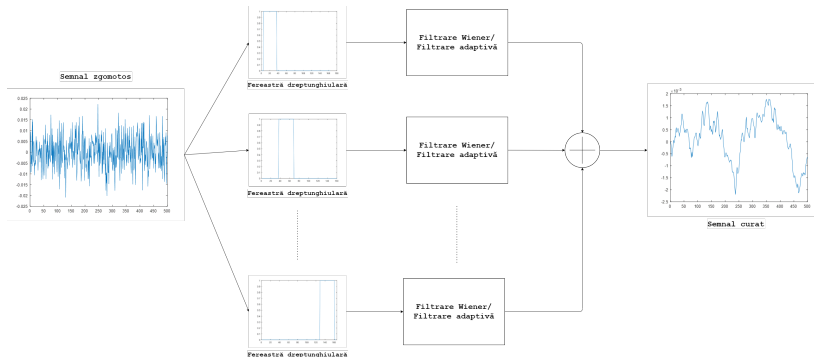


Figura 7: Schema de prelucrare a semnalelor pentru filtrare

Filtrarea semnalelor zgomotoase

Filtrul Wiener

- Minimizarea erorii $e(n) = \hat{d}(n) - d(n)$;
- Ecuațiile Wiener - Hopf sub formă matriceală:

$$\begin{bmatrix} R_{xx}(0) & R_{xx}(1) & R_{xx}(2) & \dots & R_{xx}(L-1) \\ R_{xx}(1) & R_{xx}(0) & R_{xx}(1) & \dots & R_{xx}(L-2) \\ R_{xx}(2) & R_{xx}(1) & R_{xx}(0) & \dots & R_{xx}(L-3) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ R_{xx}(L-1) & R_{xx}(L-2) & R_{xx}(L-3) & \dots & R_{xx}(0) \end{bmatrix} \begin{bmatrix} h_{opt}(0) \\ h_{opt}(1) \\ h_{opt}(2) \\ \vdots \\ h_{opt}(L-1) \end{bmatrix} = \begin{bmatrix} R_{dx}(0) \\ R_{dx}(1) \\ R_{dx}(2) \\ \vdots \\ R_{dx}(L-1) \end{bmatrix}$$

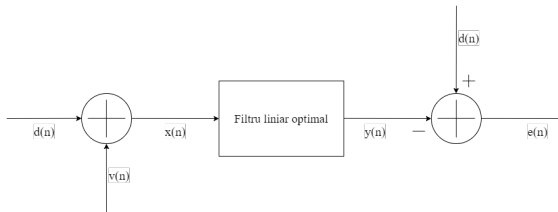


Figura 8: Schema de funcționare a filtrării Wiener

Filtrarea semnalelor zgomotoase

Filtrarea adaptivă

- Își adaptează coeficienții folosind un algoritm iterativ;
- Se urmărește minimizarea unei funcții de cost:

$$J = \frac{1}{N} \sum_{n=0}^{N-1} (d(n) - y(n))^2 \quad (2)$$

- Actualizarea coeficienților:

$$h_k(n+1) = h_k(n) - \gamma \frac{\partial J}{\partial h_k(n)} \quad (3)$$

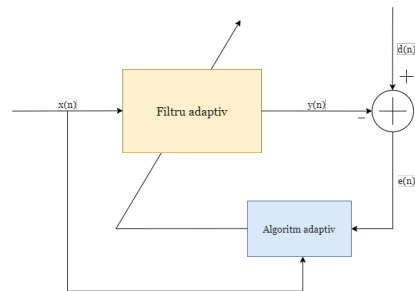


Figura 9: Schema de funcționare a filtrării adaptive

Eliminarea zgomotului folosind rețele neurale

Descrierea seturilor de date

Setul de date folosit → melodii deverberate create pornind de la o bază de date publică²;

100 de înregistrări la orgă electrică, eșantionate cu $f_s = 44100\text{Hz}$, împărțite astfel: 90 pentru antrenare, 9 pentru validare și 1 melodie pentru testare;

Semnalele au fost segmentate folosind o fereastră Hanning cu lungime de 23ms, cu suprapunere de 50%;

Pentru fiecare segment a fost calculată transformata Fourier în 1024 puncte, din care au fost selectate primele 513 puncte.

²<https://speed.pub.ro/downloads/music-datasets/>

Rețea neurală complet conectată (FCN)

- Toți neuronii interconectați între ei;
- Date de antrenare → modulul coeficienților Fourier ai semnalelor de reverberate;
- Operatorul de activare: $ReLU = \max(0, x)$;
- Minimizarea funcției de cost MSE (Mean Squared Error):

$$MSE = \frac{1}{N} \sum_{i=1}^N (Y_i - \hat{Y}_i)^2 \quad (4)$$

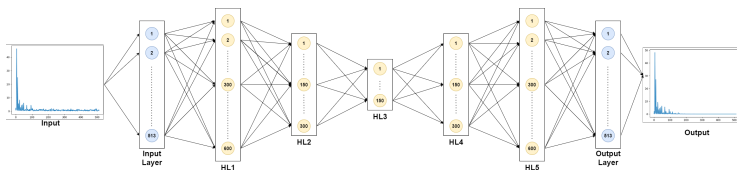


Figura 10: Rețea FCN pentru predicția modului Fourier.

Rețea *capsulă* (CapsNet)

- Fiecare strat este împărțit în grupuri de neuroni (capsule), fiecare prelucrând un anumit tip de date.
- Pentru antrenare → partea reală și imaginară a coeficienților Fourier extrași din semnalele de reverberație.
- Funcții de activare folosite:

$$f(z) = \frac{8}{3\sqrt{3}} \frac{|z|}{1+|z|^2} z, \quad f(z) = \frac{z}{1+|z|}, \quad f(z) = \text{CReLU}(z), \quad z \in \mathbb{C}$$

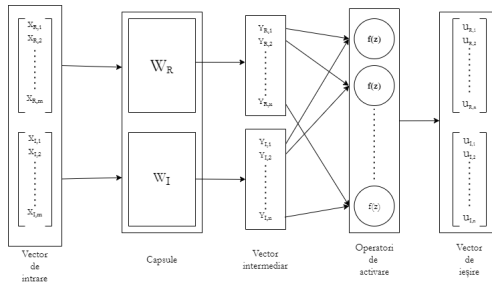


Figura 11: Structura unui strat neural cu capsule.

Rezultate experimentale

Performanța metodelor propuse

- Comparație între rețelele propuse și metodele clasice de eliminare a zgomotului;
- Validarea rezultatelor folosind un semnal de test;
- Evaluarea performanței folosind 3 metrici: SNR [dB], PSNR [dB] și CC:

Semnal	Model	SNR [dB]	PSNR [dB]	CC
Semnal zgomotos	$s * h + n$	5.00	21.30	0.87
Semnal deverberat	-	3.13	19.39	0.72
Semnal recuperat	Filtrul Wiener	9.23	25.49	0.94
	Filtrul adaptiv	8.12	24.40	0.92
	FCN	16.82	33.07	0.99

Tabela 1: Metode clasice vs. FCN

Performanța metodelor propuse

Semnal	Model	Activare	SNR [dB]	PSNR [dB]	CC
Semnal zgomotos	$s * h + n$	-	5.00	21.3	0.87
Semnal deverberat	-	-	3.13	19.39	0.72
Semnal recuperat	Filtrul Wiener	-	9.23	25.49	0.94
	Filtrul adaptiv	-	8.12	24.40	0.92
	CapsNet 1	$f(z) = \frac{8}{3\sqrt{3}} \frac{ z }{1+ z ^2} z$	12.24	28.49	0.97
	CapsNet 2	$f(z) = \frac{z}{1+ z }$	12.01	28.25	0.97
	CapsNet 3	$f(z) = \mathbb{C}ReLU(z)$	11.98	28.22	0.97

Tabela 2: Metode clasice vs. CapsNet

Performanța metodelor propuse

Semnal	Model	Constrângere	θ_m	SNR[dB]	PSNR[dB]	CC
Semnal zgomotos	$s * h + n$	-	-	5.00	21.3	0.87
Semnal deverberat	-	-	-	3.13	19.39	0.72
Semnal reconstruit	Filtrul Wiener	-	-	9.23	25.49	0.94
	Filtrul adaptiv	-	-	8.12	24.40	0.92
	FCN	\mathcal{D}_i	2.74×10^{16}	12.94	29.19	0.97
		$\mathcal{D}_i \cap \mathcal{C}_i^1$	0.85	9.72	25.97	0.96
		$\mathcal{D}_i \cap \mathcal{C}_i^3$	1.01	6.31	22.56	0.89

Tabela 3: Metode clasice vs. rețele constrânse

Performanța metodelor propuse

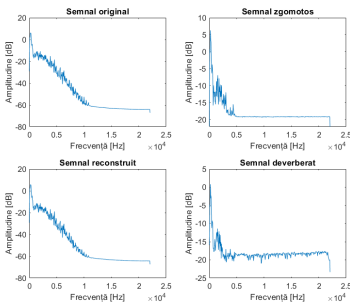


Figura 12: Densitățile spectrale de putere ale semnalului de test în diferite stadii

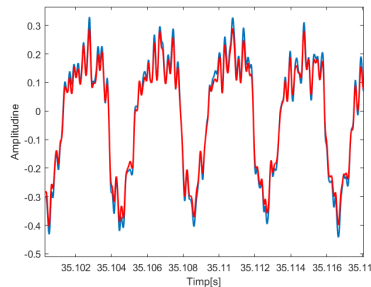


Figura 13: Semnalul de test reconstruit suprapus peste cel inițial

Concluzii

Contribuții personale

- Am implementat un algoritm de deverberare a semnalelor audio;
- Am implementat filtre digitale pentru eliminarea zgomotului suprapus peste semnale audio;
- Am implementat două tipuri de rețele neurale pentru a elimina zgomotul suprapus peste semnale audio;
- Am implementat constrângeri pentru a antrena rețele neurale nonexpansive controlând constanta Lipschitz;
- Am testat performanța sistemelor implementate și am comparat rezultatele.

Contribuții personale

- Am implementat un algoritm de deverberare a semnalelor audio;
- Am implementat filtre digitale pentru eliminarea zgomotului suprapus peste semnale audio;
- Am implementat două tipuri de rețele neurale pentru a elimina zgomotul suprapus peste semnale audio;
- Am implementat constrângeri pentru a antrena rețele neurale nonexpansive controlând constanta Lipschitz;
- Am testat performanța sistemelor implementate și am comparat rezultatele.
- **Obiective de viitor** → folosind modelul rețelelor neurale constrânse, vom implementa un algoritm iterativ de tip Plug and Play (PnP) care înlătură simultan efectul de reverberație și cel de zgomot din diferite tipuri de semnale audio.

Referințe



Dragoș Burileanu

Prelucrarea digitală a semnalelor

Note de curs. UPB ETTI. 2019



Dragoș Burileanu

Tehnici avansate de prelucrare digitală a semnalelor

Note de curs. UPB ETTI. 2019



Mathieu Terris and Audrey Repetti and Jean-Christophe Pesquet and Yves Wiaux

Building firmly nonexpansive convolutional neural networks

IEEE International Conference on Acoustics, Speech and Signal Processing ,2020



Patrick L. Combettes and Jean-Christophe Pesquet

Lipschitz Certificates for Layered Network Structures Driven by Averaged Activation Operators

2020

Vă mulțumesc pentru atenție!
