

CLASIFICAREA INSTRUMENTELOR MUZICALE

*Proiect pentru Procesarea Semnalelor,
Raducan Laurentiu-Alex & Serban Comisin*

DESCRIERE

Proiectul isi propune sa clasifice instrumente muzicale folosind metoda k-ului mai apropiati vecini (**k-NN**) [1]. Pentru a usura iteratii consecutive de testare, rezultatele perioadei de “training” vor fi salvate in directorul proiectului. Distinctia dorita este intre sunete ce apartin unui pian, unei chitare acustice, unui saxofon si a unei tamburine. Setul de date va fi 20 pentru fiecare instrument.

In vederea coordonatelor folosite pentru acest algoritm de clasare se iau in considerare :

- 4 parametri temporali:
 - Rata de Trecere Zero (**ZCR**)
 - Radacina Medie Patrata (**RMS**)
 - Energia pe Termen Scurt (**STE**) [2]
 - Fluxul (**FLU**) [3]
- 3 parametri spectrali:
 - Centroidul Spectral (**SC**) [4]
 - Latimea de Banda (**BW**) [3]
 - Intensitatea Semnalului Aproximat prin Suma Subarmonicilor (**PI**) [5]
- Toti Coeficientii Cepstrului Frecventei Mel (MFCC) [6]

Datele de intrare sunt scurte inregistrari audio de tip WAVE (.wav) colectate din surse publice[7] si fiind sub licenta “Non-Comercial Creative Commons” [8]. Toate au frecventa de esantionare (fs) 44100 Hz si 2 canale, din care este folosit doar primul si primele 5000 de esantioane.

Primii 7 parametrii sunt scalati individual in functie de cel mai mare element in modul din setul de test, iar coeficientii Mel, avand in vedere ca frecventa de esantionare este constanta, sunt normalizati.

In urmatoarele notatii definim:

- $x(t)$ → valoarea semnalului la momentul t
- $X(k)$ → valoarea spectrului semnalului pentru frecventa $k : X = \text{dft}(x)$
- $P(k)$ → puterea spectrului semnalului pentru frecventa $k : P(k) = |X(k)|^2$
- fs → frecventa de esantionare
- N → numarul total de esantioane in domeniul timp
- F → numarul de subdiviziuni pana la jumatatea frecventei de esantionare

ZCR: se calculeaza procentul de schimbare al semnului relativ la $N - 1$.

$$\frac{1}{N-1} \sum_{t=2}^N (\text{sign}(x(t)) - \text{sign}(x(t-1)))$$

RMS: $\sqrt{\frac{1}{N} \sum_{t=1}^N x^2(t)}$

STE: energia totala este definita drept $\sum_{t=-\infty}^{\infty} x^2(t)$ din aceasta formula putem extrage

energia pe termen scurt ca fiind $\sum_{t=1}^N x^2(t)$ careia ii putem aplica o scalare prin impartirea la N

FLU: prin flux intelegem in acest context diferenta de energie dintre momentul final si cel initial.

SC: centroidul spectral este baricentrul spectrului care in cazul discret are formula

$$\sum_{k=0}^F \frac{\frac{k}{F} fs/2 P(k)}{\sum_{k=0}^F P(k)}$$

BW: in contextul acestui proiect definim latimea de banda ca fiind

$$\sqrt{\sum_{k=0}^F \frac{(\frac{k}{F} fs/2 - SC) P(k)}{\sum_{k=0}^F P(k)}}$$

PI: intensitatea sunetului poate fi aproximata folosind sume de subarmonici. Pentru a genera subarmonicile calculam amplitudinile spectrului, gasim varfurile pozitive si egalam cu 0 orice element al amplitudini spectrului ce nu este varf sau este la mai mult de doua pozitii discrete departare de un varf. Aplicam asupra acestui rezultat o fereastră hanning si o atangenta ridicata de forma $\tan(\frac{x-1}{5})/\frac{\pi}{2} + 1$; pentru a genera subarmonicile dorite (sa zicem S) vom shifta rezultatul prin a il inmulti cu 0.84^n si vom aduna aceste rezultat. Frecventa pentru care rezultatul acestui proces este maxim va fi aproximarea intensitatii sunetului de intrare

MFCC: algoritmul de calculare al coeficientilor urmeaza umratorii pasi

- Extragerea de cadre din semnal, cadre ce se suprapun partial
- Pentru fiecare cadru se calculeaza o aproximatie a periodigramei
- Aplica filtrele specifice si aduna rezultatul pentru fiecare filtru
- Aplicam logaritm in baza 10 apoi DCT asupra rezultatului

In cazul proiectului cadrele sunt de 1200 de esantioane cu defazare de 600, se folosesc 2048 de puncte pentru calculul spectrului cu DFT si 10 filtre.

Limbajul folosit este Octave iar mediul de rulare este LXC pe Windows 10, 64bits. Pentru a testa fisierele "test_xx.wav" se deschide consola octave in directorul de baza al proiectului si se apeleaza "runTest" daca se doreste (re)generarea datelor instrumentelor cunoscute sau "runTestFromSave" in cazul in care se doreste testarea rapida.

REFERINTE

<https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/> [1]

<https://pdfs.semanticscholar.org/c20b/e6aaeaed889444e225e2bd18a6eb44e9cb36.pdf> [2] - pag. 2

<https://core.ac.uk/download/pdf/143870996.pdf> [3] pag. 3

http://recherche.ircam.fr/equipes/analyse-synthese/peeters/ARTICLES/Peeters_2003_cuidadoaudiofeatures.pdf [4] pag. 13

https://www.researchgate.net/publication/19813760_Measurement_of_pitch_by_subharmonic_summation [5]

<http://practicalcryptography.com/miscellaneous/machine-learning/guide-mel-frequency-cepstral-coefficients-mfccs/> [6]

<https://freesound.org/> [7]

<https://creativecommons.org/licenses/by-nc/4.0/legalcode> [8]