Music Genre Classification using Deep Neural Networks

Albert Jiménez Sanfiz , Ferran José Torra

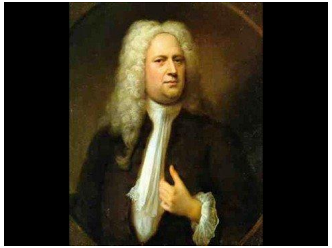
Code available at GitHub

Outline

1. Introduction
2. Related Work
3. Our Proposal
4. Experiments
5. Conclusions

1. Introduction

Music Genre Classification



Classical = 89 %
Jazz = 7 %
.
.
.

Country = 85 %
Blues = 11 %
.
.
.

Disco = 85 %
Classical = 10 %
.
.
.

Rock = 85 %
Metal = 9 %
.
.
.

Music Genre Classification Applications

Automatically:

- Create reproduction lists
- Recommend new music
- Organize music libraries

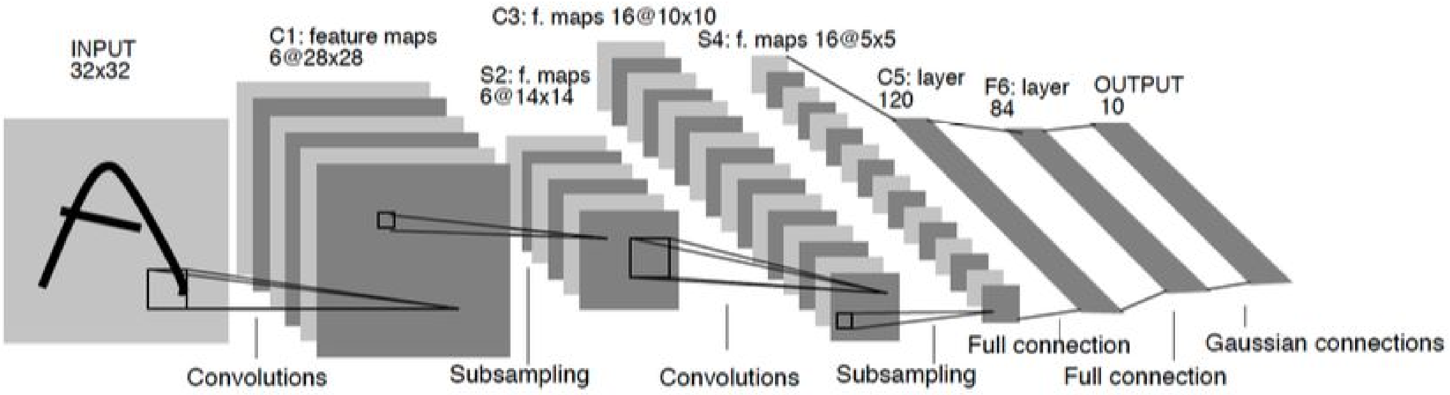
Music Genre Classification Difficulties

Difficulties:

- Overlapping of genres during songs
- Different songs' length

2. Related Work

CNN brief Explanation (Le Net)

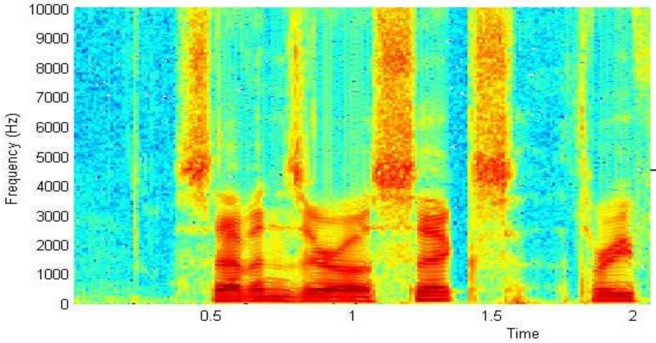


Convolutions layers → Learn Filters (Feature extraction)
Pooling layers→ Downsampling
Fully connected layers → Act as classifier

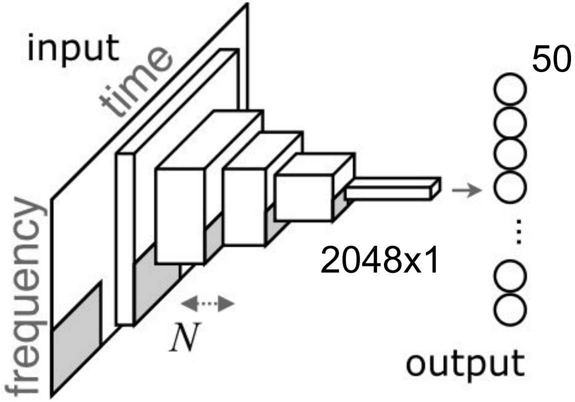
CNNs for music Classification

Song Audio File

29.12s = 96×1366



Mel-Spectrogram

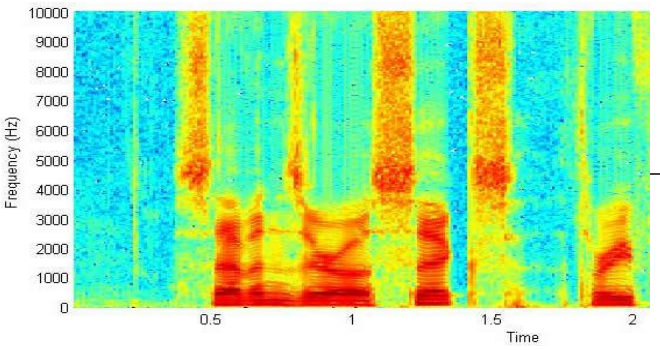


Feature extraction & Classification

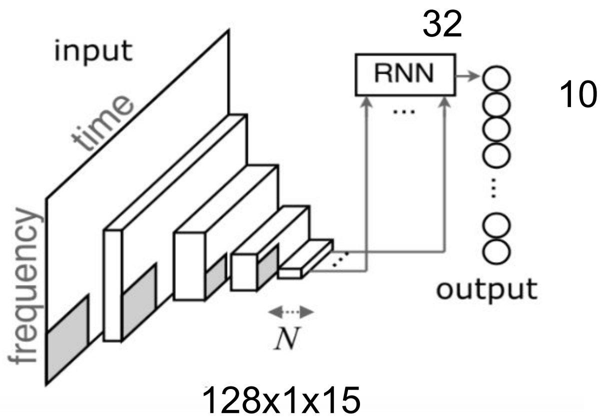
[Automatic Tagging Using Deep Neural Convolutional Networks](#)
[Keunwoo Choi, George Fazekas, Mark Sandler *(2016)*](#)

CRNNs for music Classification

Song Audio File



Mel-Spectrogram

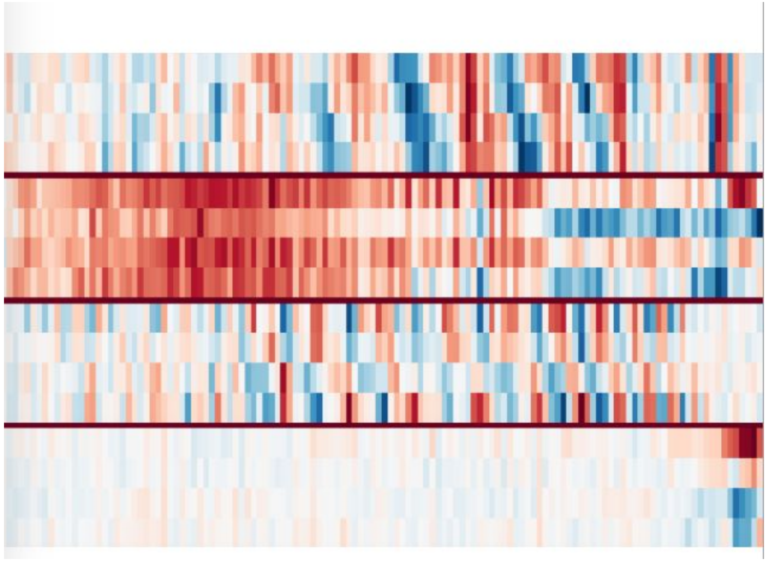


Feature extraction & Classification

Convolutional Recurrent Neural Networks For Music Classification

Keunwoo Choi, George Fazekas, Mark Sandler *(2016)*

What are the filters learning?



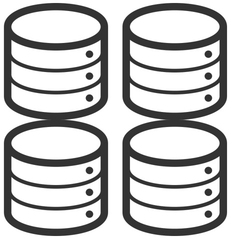
Filter 14		Vibrato singing
Filter 242		Ringing ambience
Filter 250		Vocal thirds
Filter 253		Bass drum sounds

http://benanne.github.io/2014/08/05/spotify-cnns.html

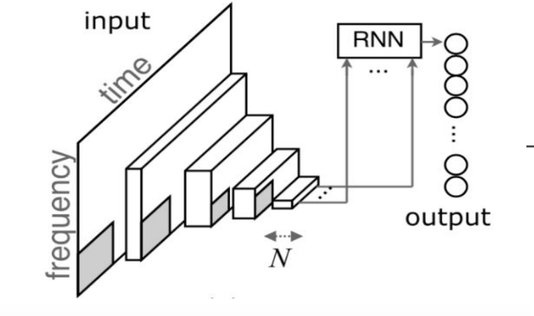
3. Our Proposal

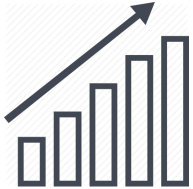
Training Paradigm

- Training a model from scratch is very expensive:  +  = 



Large Datasets





Good Performance

Huge Computational Power

Our Situation

Small dataset

Low computational budget (CPUs)

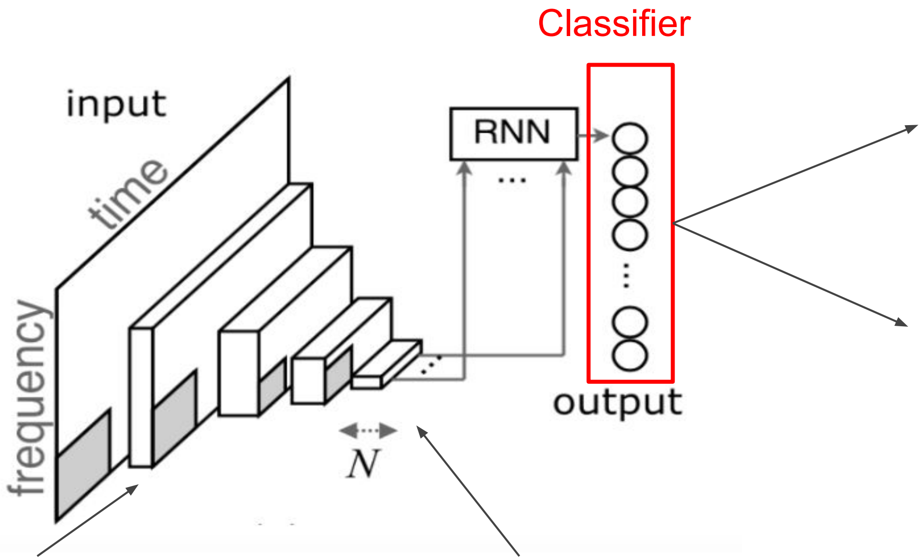
No money

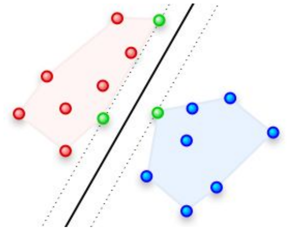
Proposal for training

Transfer Learning

- Take a pre-trained model on a large dataset and fine-tune its weights to adapt it to perform a new task
    - Less data needed
    - Less training time

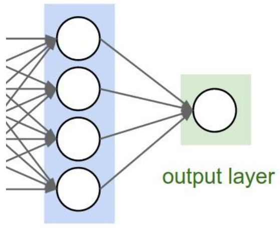
Training with transfer learning





SVMs

Loss
Function



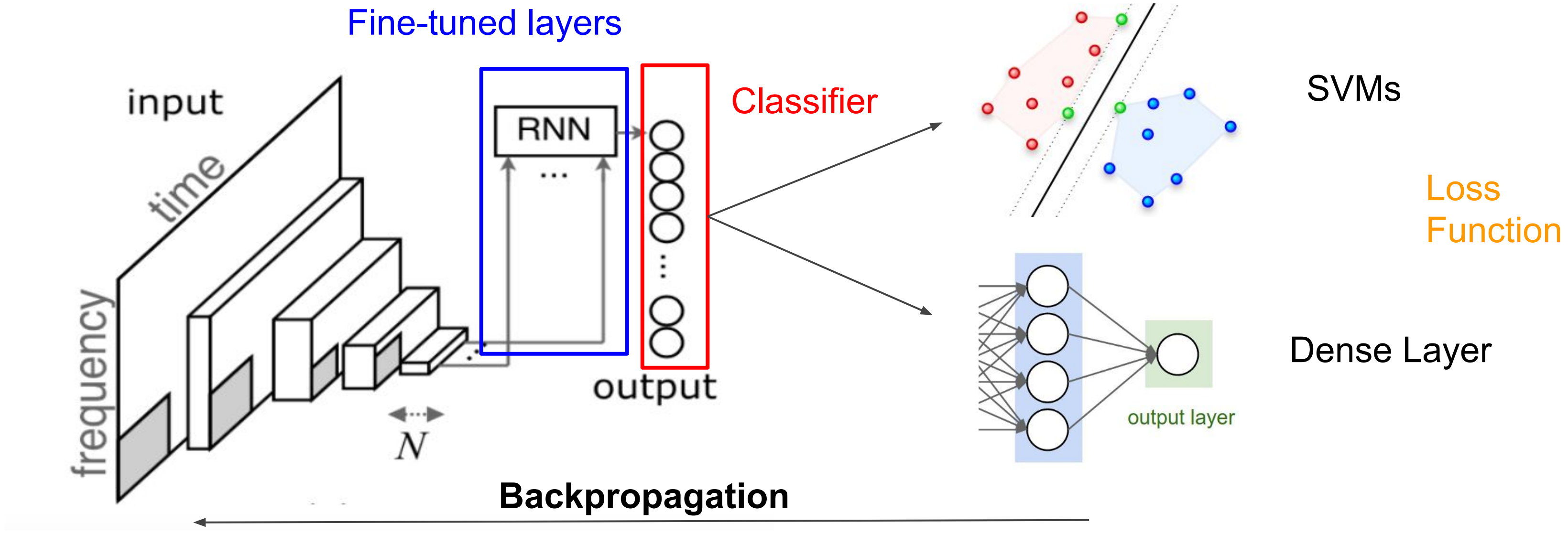
Dense Layer

Low level features

High level features

Use pre-trained model as feature extractor and only change the classifier

Training with transfer learning



Change the classifier and also fine-tune the weights of the network

Multiframe

Motivation

- Most of the works only use the highlight of the song to extract features and evaluate
- Lost of relevant information

Multiframe

Advantages

- At training time
  - More data available

- At test time
  - Averaging over the frame scores
  - KNN algorithm

4. Experiments

Framework

- Audio libraries: libROSA

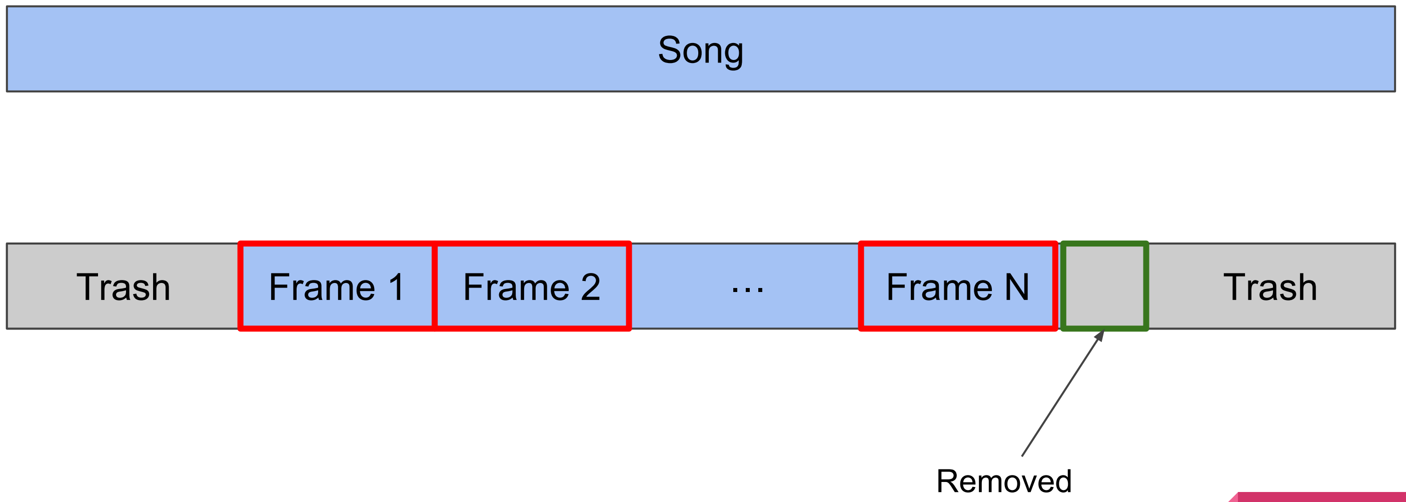
- Deep learning framework: Keras and Theano

- Music datasets:
    - A million song dataset
    - GTZAN dataset
    - Handmade dataset

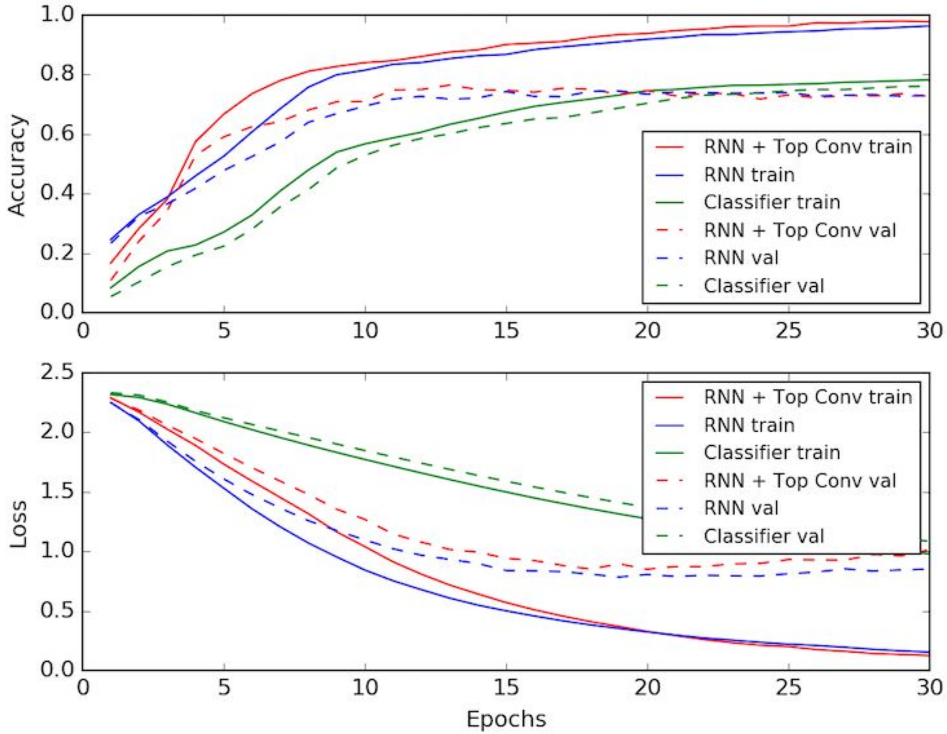
Handmade Dataset

- 30 Songs per genre
  - Split in 20 train/10 test
  - 1468 / 747 frames
- Tags
  - Blues
  - Classical
  - Country
  - Disco
  - Hiphop
  - Jazz
  - Metal
  - Pop
  - Reggae
  - Rock

Frames acquisition



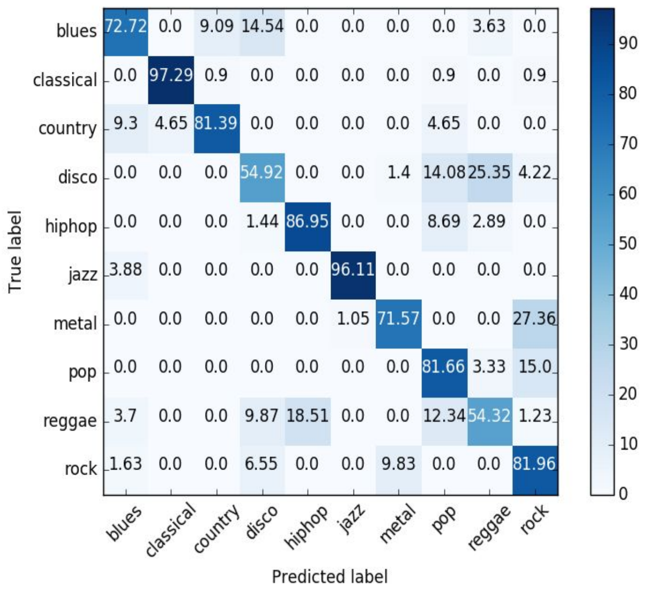
Training and GTZAN evaluation



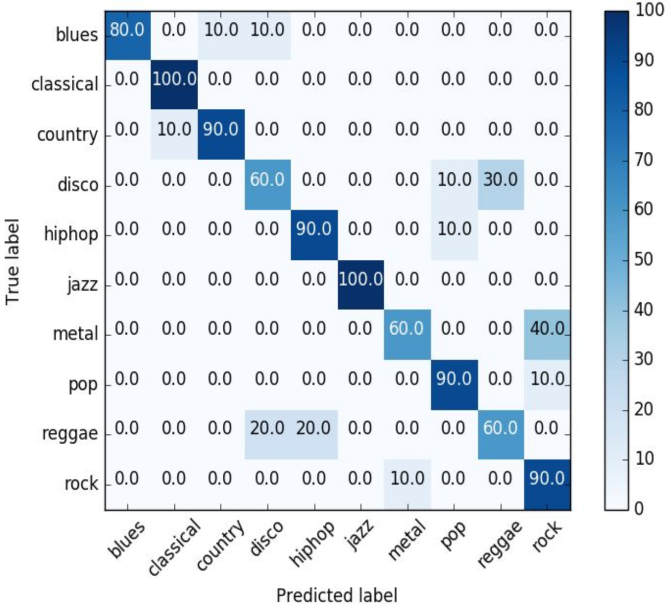
- Overfitting problems when fine-tuning more than the classifier layer
- The classifier layer is not complex enough to not get stalled
- Final performance on GTZAN around 78% of accuracy
- The lack of **DATA** is the main obstacle to improve

Results

Confusion matrices:



Average stage



5. Conclusions

Conclusions

- This kind of networks need large quantities of data to be trained from scratch
- Transfer learning is a good alternative for a low-budget scenario
- Our multiframe approach with an average stage improves the single-frame song model

Future work

- Training of the model using larger datasets

- Prove other techniques to obtain a single genre tag per song from multiple frames
  - Knn
  - Geometric mean


Thank you


Q&A