Update Thesis 11-04-2022

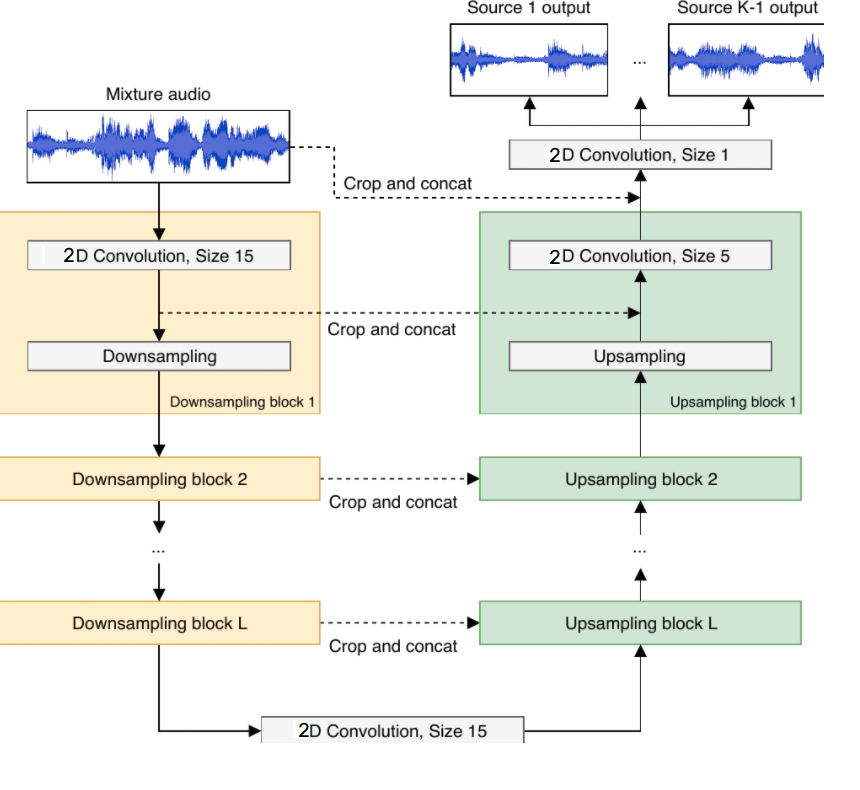
Zie code**: https://github.com/DiegoLigtenberg/Workspace-MasterThesis-MSS**

tijdje geleden:

* 4 april jaar geleden dat mijn broertje overleden was; voorbereiding / mind was even niet bij scriptie
* 2 dagen erna ziek geworden (griep), t/m gisteren dagen in bed gelegen
* Goed om weer verder te gaan!

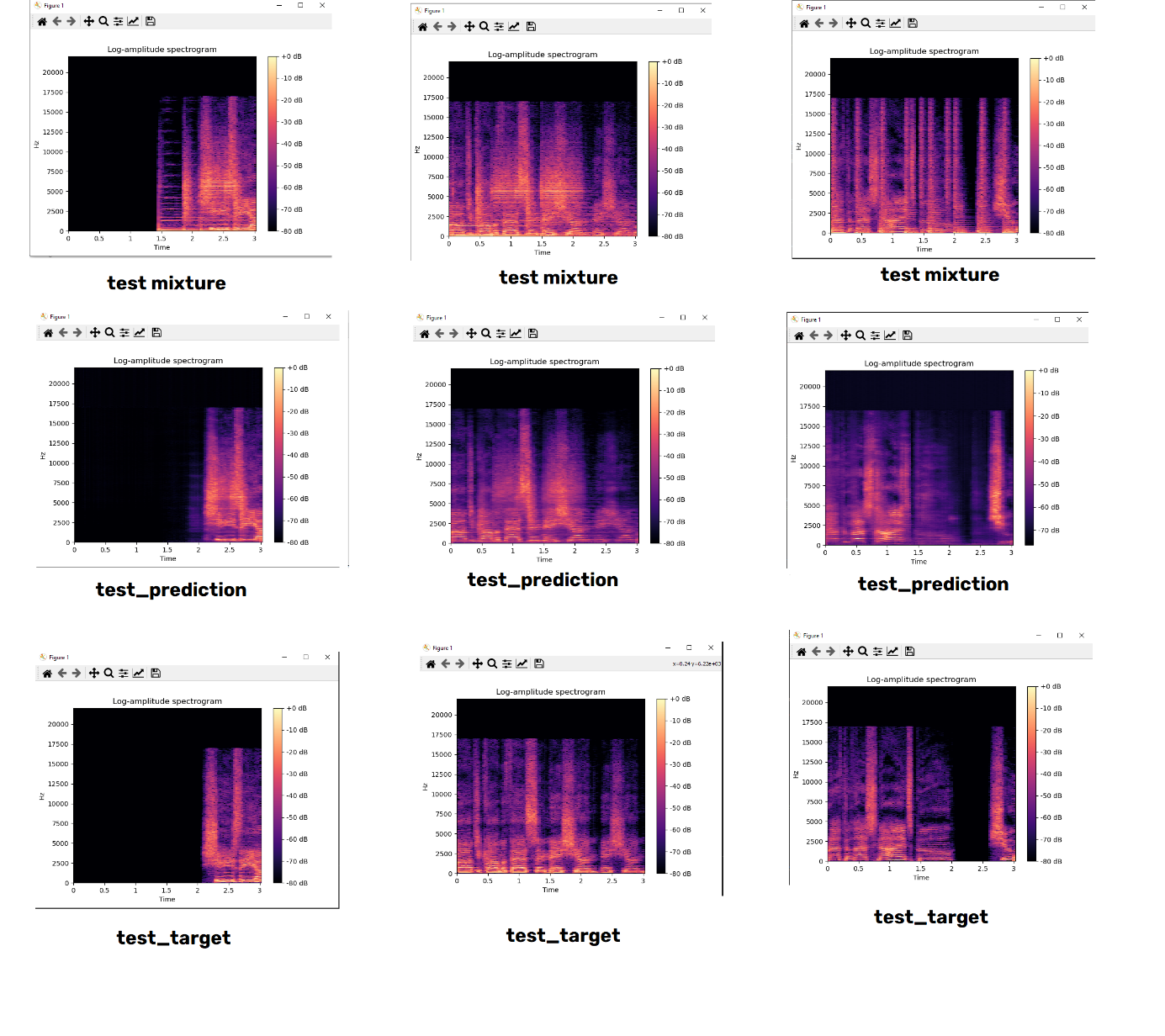
**PREPROCESSING**

* Preprocessing pipeline gemaakt voor Musdb dataset: 150 songs met gevarieerde lengte
* De 150 songs zijn verdeeld in 85 train, 15 validation, 50 test.
* De Preprocessing pipeline ziet er als volgt uit:
  1. Laad de volledige song
  2. Extract 3 second segments
  3. Augment de segments (zit in preprocessing pipeline, maar nog niet opgeslagen)
  4. Pad het signaal (laatste stuk van een song soms 2.1213 seconds -> pad naar 3 seconds door 0 toe te voegen
  5. Extract log spectrogram using stft
  6. Normalize spectrogram based on global statistics (/160)
  7. Save normalized spectrogram as .npy



**AUTO ENCODER MODEL**

* Auto encoder gemaakt met skip connections.
* Train dataset bestaat uit 6403 x 2048 x 128 npy files -> te groot om in ram te passen, dus **train on batch** implementation.
* Eerst getrained op 1 sample **mixture -> vocal**, om te kijken of het overfit met close to 0 loss.
  + Dit werkte! mse loss van 2e-5 geeft ‘perfect’ results -> architecture kan werken
* Daarna getest op grotere dataset (veel bugfixes)
  + Niet shuffelen
  + Shuffelen van X en Y waardoor labels mismatchte
  + Geen validation set
  + Verschillende learn rates getest
  + Skip connections werkte niet
* Uiteindelijk stabalized en 1 setting aan main parameters opgeslagen en mee gaan trainen
  + Results:
    - Werkt, maar overfit (voor grootste gedeelte had ik nog geen validation stats om te checken voor overfit) -> pas 2 dagen geleden implemented



* **GOALS VOOR MIXTURE -> “OTHER” (alleen instruments) om beter te maken (tot 30 april)**
  + **Add regularisation (l2 regularisation + dropout)**
  + **Loss curve voor train/test goed monitoren**
  + **Evt andere visualisations?**

Loss curves

* Validation loss decreased minder snel
* Regularisation techniques implemented -> niet super veel success

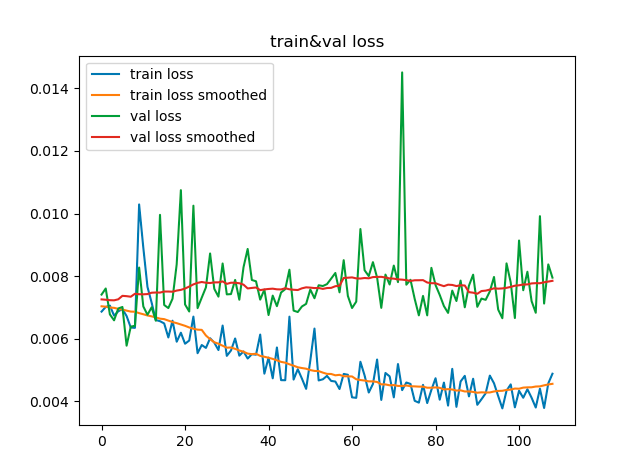
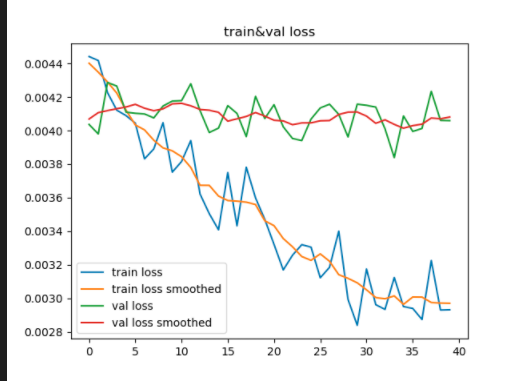
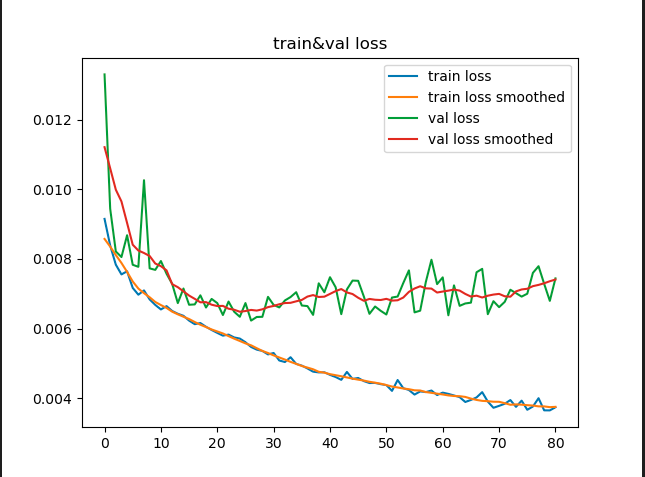


Figure No Batch Normalisation

Figure : Plain model

Figure Dropout + regularisation

Batch normalisation:

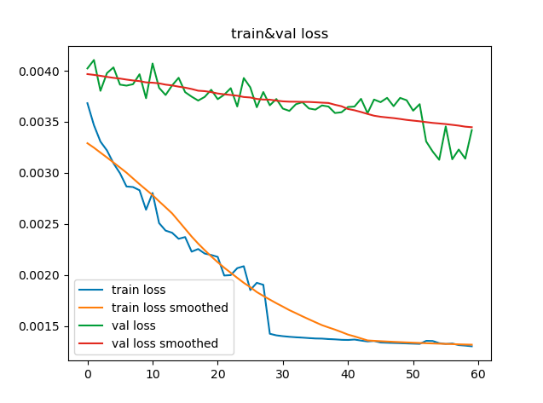
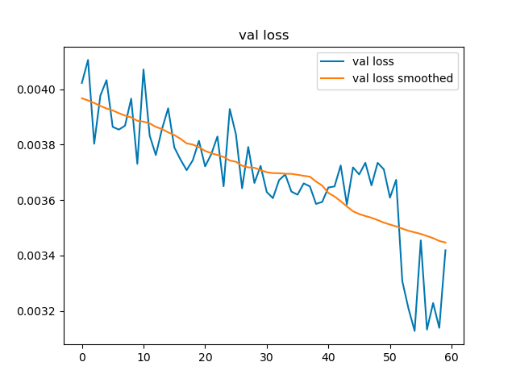
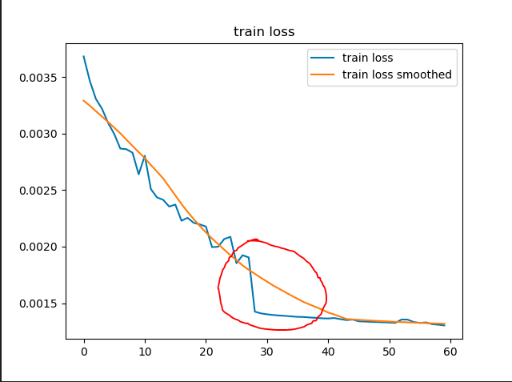
* Gekeken naar weights -> erg groot (hoort niet)

Data augmentation

* Polarity inversion
* Reverse signal
* Time stretching
* Pitch shifting

**April 26th**

**Results beginning week:**

* Trained Mixture - > Other (instrumentation without vocal) model
* Experimented with loss functions (but during training)
* Training was mostly done with grid search in mind
  + Trying different loss functions
  + Different learning rates
  + Augmented vs non augmented dataset
  + Model architecture
* Combination of everything gave final model. works ‘okay’ on test set, but does not generalize to other instrumentation. Especially saxphone + vocalinstrumentation got completely blurred.

Train/val train validation

**Mid week:**

* Started looking more into IRMAS dataset
* Cello, clarinet, flute, guitar (elec and acoustic), organ, piano, sax, trumpet, violin, **voice**
* **Voice** is also a recognition instrument -> thus creating any artefacts by removing voice does not make sense!
* **Trained new model that has other + vocal** -> thus removing the drums + bass
* **Because major change** -> looked at proposal + other paper’s doing the same thing and found 3 papers supporting my hypothesis

**3 papers supporting my hypothesis:**

**Singing Voice Separation**

<https://www.researchgate.net/profile/Bidisha-Sharma/publication/335829325_On_the_Importance_of_Audio-Source_Separation_for_Singer_Identification_in_Polyphonic_Music/links/60dd762ca6fdccb745f90fc2/On-the-Importance-of-Audio-Source-Separation-for-Singer-Identification-in-Polyphonic-Music.pdf>

* When converting songs (train + test) to only vocals using WaveUNET -> Singer classification was way easier

**Instrument Classification OLD FASST**

<https://ismir2012.ismir.net/event/papers/559_ISMIR_2012.pdf>

* When using **separated sources on both train + test ->** Detecting instruments is better

They used separate classification algorithm for each separated source -> combined the labels

* Already found slight increase when only test data was source separated with normal train data

**Instrument Classification New** (STILL FASST AND HARMONIC/Percusive from 2014 very old)

https://archives.ismir.net/ismir2018/paper/000145.pdf

* They tested whether using the separation algorithms can be used to increase performance of instrument recognition
* when the **training dataset + test dataset BOTH got separated**, they found best performance

But they used transfer learning -> because source separation is so bad that they had to use weights from previously trained model

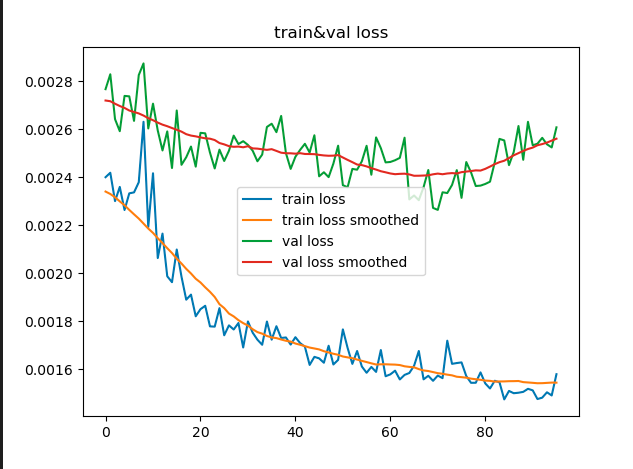
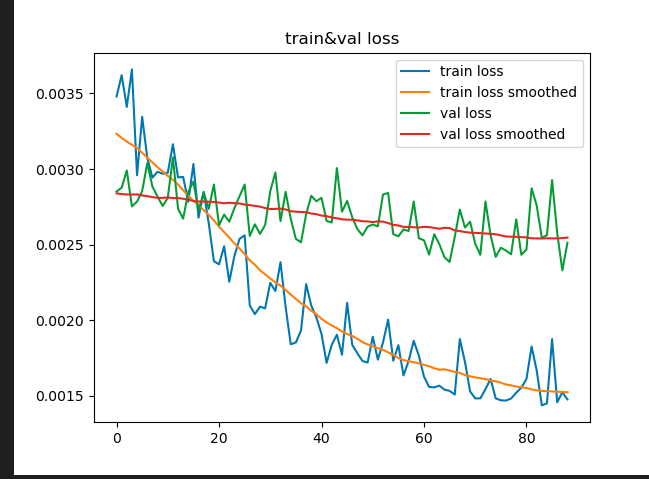
Afbeelding met tekst, hond, zoogdier, wolf

Automatisch gegenereerde beschrijvingDetection of singing foice, or instrument better if you isolate the main source -> in image you would preprocess and potentially remove the background to make network focus on the source of interest!

**End week**

**Created new Dataset -> Vocal + Other (removed drums + bass)**

* tried different architectures and attempt to systematically evaluate different hyperparameters
* Loss function mse 3e-4
* Batch normalisation -> Relu -> (Concatenate) (best order)



Other + Vocal ReLU -> BN Other + vocal BN -> ReLU

Shortly tested on IRMAS dataset -> at 50 epoch

* Model is less aggressive because only removing bass and drums
* Less artefacts but also sounds bit ‘dof’
* This source separation at least feels useable! Still not perfect, but it does what it has to do
* Compared to previous papers using harmonic/percussion separation -> sounds good!