Preprint of the paper to appear in Proc. ICPR, 2020.

Ballroom Dance Recognition from Audio Recordings

Tomáš Pavlín, Jan Čech, Jiří Matas
Center for Machine Perception, Department of Cybernetics,
Faculty of Electrical Engineering, Czech Technical University in Prague

***Abstract*—We propose a CNN-based approach to classify ten genres of ballroom dances given audio recordings, five latin and five standard, namely Cha Cha Cha, Jive, Paso Doble, Rumba, Samba, Quickstep, Slow Foxtrot, Slow Waltz, Tango and Viennese Waltz. We utilize a spectrogram of an audio signal and we treat it as an image that is an input of the CNN. The classification is performed independently by 5-seconds spectrogram segments in sliding window fashion and the results are then aggregated. The method was tested on following datasets: Publicly available Extended Ballroom dataset collected by Marchand and Peeters, 2016 and two YouTube datasets collected by us, one in studio quality and the other, more challenging, recorded on mobile phones. The method achieved accuracy 93.9%, 96.7% and 89.8% respectively. The method runs in real-time. We implemented a web application to demonstrate the proposed method.**

I. Introduction

This study presents a method for balroom dance classification by convolutional neural networks (CNN). Dance classes (Waltz, Tango, Jive, etc.) are recognized from a given audio recording. The problem is interesting from both theoretical and practical perspectives. The core of the problem is to encode low-level information from a raw audio signal and to produce a semantic high-level information, the dance class. We benefit from employing the dance music, because a large amount of audio data is publicly available. Various audio recordings of dance music can be extracted either from YouTube or from video recordings of dance competitions, that experience increasing popularity. The categorization of such music data is usually within video or track titles, thus the data can be easily labeled. Moreover, dance recognition is beneficial for beginners or amateur dancers that have difficulties to recognize the dance from the music. To the best of our knowledge, there is no commercial application that would help dancers to recognize a dance from music. Rather than a dance genre classification, many researches focused on a related problem, music genre recognition (e.g. Jazz, Pop, Rock, etc.), for the past decades. The problem of predicting a dance class from audio recordings seems unexplored.

Therefore, we propose a simple dance recognition method. See Fig. 1 for an overview. A CNN is trained to classify short spectrogram segments (of length 5 seconds). When testing, the CNN is executed in a scanning window fashion and finally, the softmax results of the overlapping segments are averaged over the input recordings.

The contributions of the paper are: (1) We proposed a novel CNN based method for dance recognition, (2) we collected a test dataset that we made publicly available for research community, and (3) we implemented the web application as a demonstration.

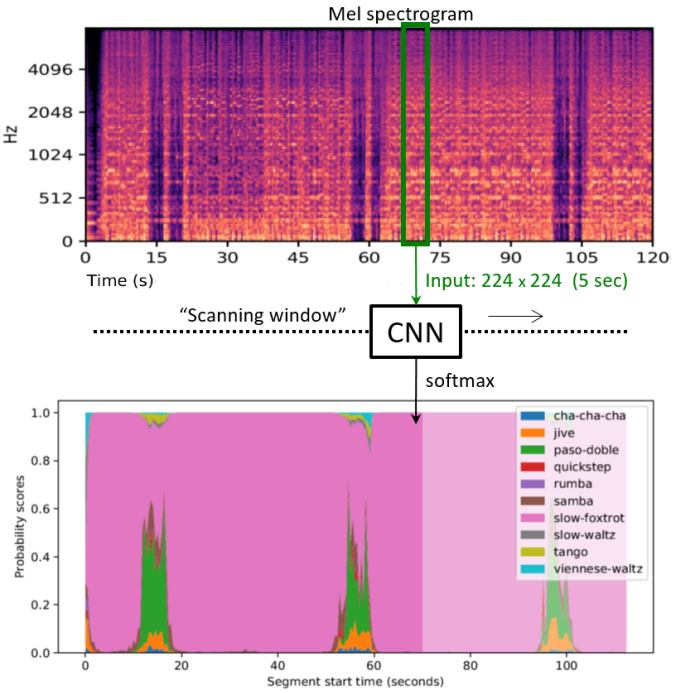


Fig. 1. MEL spectrogram is computed from an input audio signal. The Convolution Neural Network (CNN) is run on spectrogram segments (of length 5 sec) in a scanning window fashion. The CNN outputs a softmax distribution over the dance classes (shown in colors). The class of the whole recording is eventually found by averaging.

The remainder of the paper is organized as follows. Related work is presented in Sec. II. The proposed method is explained in Sec. III. Experiments are given in Sec. IV. Finally, Sec. V concludes the paper.

II. Related Work

Dance recognition of audio samples relates to interdisciplinary area called Music Information Retrieval (MIR) [1]. Significant research has been made recently in the field with different objectives. Music genre classification task is common in this area, although there is no significant focus on dance music recognition. Among the small number of studies regarding dance music recognition, we highlight three. Work [2] classify dance music by timing information as tempo

and meter (periodicity). The timing information is used to describe each dance class by functional language rules. The approach was further improved in [3] extending the periodicity patterns by accent patterns. Paper [4] presents scale and shift-invariant time-frequency representation of audio content. A classifier of ballroom dance music with promising results is proposed, although the classification method is not detailed and the testing protocol is unclear.

Instead of the dance recognition, a related problem, music genre classification (MGC) has been studied much more in literature. The goal is to predict music genre, e.g. classical, electronic, jazz, rock, metal, etc. The problem shares similar audio features as instruments, tempo, chords and rhythmic patterns [5]. However, the definition of the classes may be more ambiguous. It is often a challenging task even for humans [6].

Many papers attempted to solve the problem by extracting handcrafted features from audio, such as Mel Frequency Cepstral Coefficients [7]. These features, routinely used in speech recognition, are given as input to machine learning classifier such as Support Vector Machine [8], [9]. Novel architectures use spectrograms, i.e. image-like representation of the audio, along with convolution neural networks. The lastly mentioned is the state-of-the-art approach [10]. Other modern approaches use neural networks, as well as relying on spectrograms and convolutional neural networks (CNN) [11]–[13]. A comaparison of spectrogram-based methods with CNN taking raw audio signals is made in [14], however the the spectrogram approaches were not outperformed.

Paper [10] employs neural networks pre-trained for image categorization. The architecture takes advantage of transfer learning in order to train the classifier. In [15], hierarchical Long Short Term Memory (LSTM), a recurrent neural network, is used to recognise music genres. A multimodal approch is studied in [15], where images of cover photos and text of reviews are used besides the audio.

III. Proposed Method

Given an audio recording of a ballroom dance, the goal is to classify it into one of ten ballroom dance classes, five standard and five latin: *Cha Cha Cha, Jive, Paso Doble, Quickstep, Rumba, Samba, Slow Foxtrot, Slow Waltz, Tango, Viennese Waltz.*, International Style dances popular on dance competitions.

An overview of the proposed methods is shown in Fig. 1. A raw audio signal is converted into image representation called spectrogram. The spectrogram is cut into short overlapping segments that are then classified independently by CNN. Finally to classify the whole recording, the classification results from all segments are aggregated.

A. Converting Audio to Image Representation

Before we employ neural networks to classify the audio, we perform pre-processing by converting the raw audio signal to MEL-spectrogram, frequency-temporal 2D representation. We have chosen this approach since it is currently the state-of-the-art method in music genre classification [10]. The 2D (image) representation of the input allows us to use advanced CNN architectures that have been used with a great success in computer vision for image categorization [16].

Next, we cut the spectrogram to segments of size 224×224. It means the segment has the spectrogram height, number of frequencies ($n\_mels = 224$), and the horizontal size correspond to time span 5.2 seconds.

Experiments show that the used segment length is long enough to predict correct dance style accurately. Nevertheless, section III-C describes mechanism to classify recordings longer than segment length. The spectrogram segment of size 224×224 is used as input to our CNN-model.

B. Convolutional Neural Network

While there is significant number of convolutional neural networks architectures, we use Dense Convolutional Network (DenseNet) [17]. DenseNet is a recent convolutional network that outperforms state-of-the-art approaches such as ResNet [18] in various aspects. It requires less computational power to achieve high accuracy.

For training, we create a batch as follows. First, we select an audio recording by uniform sampling of the dance classes to compensate class-imbalance in the training set. Next, given the audio recording, we cut a random 224×224 spectrogram segment of the MEL spectrogram. The process is repeated 8 times to populate the training batch. The batch is then fed into the neural network.

The softmax output of the CNN can be interpreted as probability scores predicting that a given audio recording x_j belongs to particular dance class. The ground-truth labels y_j are used for loss computation. Negative Log Likelihood Loss (NLL) [19] is used as the loss function.

The model is trained to improve the prediction by repeating the training steps while new random batch is generated for each of the step. We rely on *Adam* optimizer [20] with learning rate $lr = 0.0005$.

C. Aggregation of segment results

While we trained the network to predict short segments, it is beneficial to predict samples that are longer than the segment duration of 5.2 seconds. More information is naturally encoded in longer recordings.

The CNN is executed for all segments $S_1, S_2, \ldots, S_n$ as a scanning window over the input recording. For a given segment S_i, resulting softmax output $\boldsymbol{o}_i = model(S_i) = \{o_i^{(1)}, \ldots, o_i^{(10)}\}$ is a vector that represents probability scores that the segment belongs to a particular dance class. Next, vectors $\boldsymbol{o}_1, \ldots, \boldsymbol{o}_n$ are averaged by arithmetic mean. The resulting vector represents probability scores of the recording, and the maximum is the predicted label.

Note that the segments overlap. Shorter stride leads to higher accuracy while longer stride leads to faster classification. We achieved good results with $stride = 200 \approx 4.6$ seconds

Discussion: We chose the approach of classifying the segments independently, for a stationary nature of a dance music. A dance music is a specific type of music containing repetitive patterns (beats, melody, etc.) and stationary features (musical instruments, tempo, key, etc.) that can be found and detected in each of the independent segments.

Popular means for sequence processing are recurrent neural networks (RNNs). However, contrary to RNNs that are suitable for time dependent problems as speech recognition, machine translation or action recognition, we suppose that each 5-seconds segment contains sufficient information for classification. This gives us a possibility to use CNN that is much faster than RNN and it is easier to train.

We can further benefit from averaging nature of aggregation, to classify dance music in real-time. The real-time classification is chieved by aggregating the segments in incremental-average fashion. Precisely, we classify newly-recorded segments independently, and update the aggregated average with each recent segment prediction.

Implementation details: We generated MEL spectrogram from audio signal by *Librosa*[1], a Python library. Parameters were as follows: sampling rate $sr = 22050$, number of frequencies in the vertical axis $n\_mels = 224$, time advance between frames $hop\_length = 512$, window size for STFT $n\_fft = 2048$, maximum frequency $fmax = 11025$.

Concerning the CNN, DenseNet 161 is used with the following parameters: dense blocks sizes: 6, 12, 36, 24, initial number of features: 96 and growth rate: 48. Pre-trained DenseNet is used with parameters learned on ImageNet [16]. The last classification layer is replaced with layer of 10 neuron output corresponding to the number of dance classes. The model is trained by epochs of 20 batches each consisting of 8 spectrogram segments.

The model corresponding to the epoch with the highest accuracy on the validation data (the best epoch) is finally selected. The accuracy on the validation data is computed without the segment aggregation employed for finer scale of the results because aggregation leads to accuracy 100% for some epochs. The model corresponding to the best epoch is then evaluated using the test dataset.

IV. Experiments

A. Datasets

We use datasets with ballroom dance music labeled by dance genres. The model was trained on our private dataset only, validated on validation split of YouTube dataset, and cross-tested on five other datasets. Overview of the dataset is depicted in Tab. I.

a) Private training set: The model was trained with a private dataset that consists of 4655 audio recordings of ballroom dance music that belong to 10 dance classes. We collected this dataset from dance music albums of various interpreters. The audio is recorded in studio quality and each recording is about 4 minutes long.

[1]https://librosa.github.io

Dance Genre	T	EB	YT	DC	SD	LQ
Cha Cha Cha	711	455	12	37	4	14
Jive	490	350	12	36	10	14
Paso Doble	112	53	12	35	3	13
Quickstep	458	497	12	37	6	11
Rumba	658	470	12	35	3	14
Samba	721	468	12	37	8	16
Slow Foxtrot	421	507	12	37	0	11
Slow Waltz	411	594	12	35	7	12
Tango	395	464	12	36	6	14
Viennese Waltz	281	252	12	38	3	9
Total	**4655**	**4110**	**120**	**363**	**50**	**128**

TABLE I

Number of recordings for each of the classes in the training set (T), the modified Extended Ballroom dataset (EB), Youtube dataset (YT), Dataset from Dance Competitions (DC), StarDance (SD) and Low Quality Recordings (LQ).

b) Extended Ballroom Dataset: The largest, publicly available dataset is probably the Extended Ballroom Dataset [21]. It was collected from *www.ballroomdancers.com* that sells audio CDs of ballroom dances and offers 30 seconds preview of each track to listen for free. The dataset contains 4180 recordings. There are two extra classes, Salsa and West-coast swing, which we do not consider. Classes Waltz and Slow Waltz were merged to one class as in our setting. Besides the class labels, the dataset also provides annotations as tempo, artist, song title and album name.

c) Youtube dataset: For testing and validation, we collected the dataset from YouTube to minimize overlap with the training dataset. We created the dataset by extracting an audio track from videos in YouTube channel with ballroom music[2]. Libraries *youtube_dl*[3] and *ffmpeg*[4] were used to downloading the dataset and converting to audio format. Moreover, we make YouTube dataset publicly available[5] to be used by other researchers to compare their results with our method. To balance classes among the downloaded dance music, we selected 12 recordings for each of the dance class and split it uniformly to testing and validation datasets. Thus, both testing and validation datasets consist of 10 classes of 6 recordings each, which provides $10 \cdot 6 \cdot 2 = 120$ recordings in both testing and validation datasets together. The recordings are about 3 minutes long and are in studio quality. Some of the recordings have a fixed few-seconds intro, that was added to the dance music by authors of the channel. Since the intro does not relate to any dance genre and it is short compared to the rest of the recording, we suppose it has minor impact for classification and we keep it in the recording.

d) Dance Competitions dataset: We leverage the popularity of dance competitions to collected another dataset. It was created by extracting music from 363 YouTube dance videos[6]. Videos from various dance competitions of the World

[2]https://www.youtube.com/channel/UC0bYSnzAFMwPiEjmVsrvmRg
[3]https://github.com/ytdl-org/youtube-dl
[4]https://github.com/FFmpeg/FFmpeg
[5]http://dance.ironbrain.net/testset.zip
[6]https://www.youtube.com/user/DanceSportTotal/

DanceSport Federation[7] were used. Both latin and standard dances are included.

e) StarDance: With increasing popularity of TV show called StarDance, we evaluate the model using audio recordings extracted from the show. StarDance is a Czech dance competition, where couples (a celebrity + professional dancer) dance ballroom dances, similarly to the show Dancing with the Stars popular in UK. While both latin and standard dances are competed, the background music is not typical dance music, but rather a popular music, that was not composed for a specific dance. This makes the dance recognition from such audio tracks particularly challenging even for humans. The dataset was collected by extracting an audio from 50 videos of dances from 10th season[8] broadcasted in 2019. The StarDance dataset does not contain any recordings of Slow Foxtrot.

f) Low Quality Recordings: In order to test the model robustness to noise and low quality input in general, we extracted audio from 128 YouTube videos of dance competitions, that were recorded using a mobile phone camera. The audio quality of such recordings is very low, including echo, people applauding, dancers steps sounds, and other noise and audio artifacts.

Note that we made a special attention to verify that none of the test set does overlap with our private training set. This was achieved by correlating spectrogram segments between training and test sets. Highly correlated cases, were manually verified. Conflicting recordings, including same songs played by different orchestra, were removed from the training set.

B. Tested Methods

a) Baseline algorithm: As a baseline method, we implemented a simple SVM classifer that relies on hand-crafted audio features [10]. Features were extracted from both time domain and frequency domain [10], namely Zero Crossing Rate (ZCR) [22], Chromagram [23], Spectral Centroid [24], Spectral Band-width [24], Spectral Roll-off [24], Mel-Frequency Cepstral Coefficients (MFCC) [25]. The dimension of the resulting feature vector is 72. An SVM classifier [26] with RBF kernel was trained.

b) MASSS [4]: The acronym stands for Modulation Scale Spectrum with Auditory Statistics. The paper focus on a representation of audio content. Classification results on [21] are reported. Nevertheless, the paper does neither reveal the classifier details nor the test protocol.

c) Dance-CNN: Our proposed method, that was presented in Sec. III.

C. Method comparison

The methods were tested on the YouTube and the Extended Ballroom datasets. Results are shown in Tab. II and in Tab. III respectively.

For the proposed method (Dance-CNN), both results of independent segment classification (without aggregation) and with the aggregation of segment results over the entire recordings are reported. The aggregation improves accuracy. Note that Top-2 accuracy is saturated for small (60 recordings) YT dataset. On the other hand, accuracy of the baseline algorithm is much lower.

Method	**Top-1 accuracy**	**Top-2 accuracy**
Dance-CNN with aggregation	96.7%	100.0%
Dance-CNN without aggregation	92.2%	-
Baseline algorithm	40.0%	-

TABLE II
RESULTS ON YOUTUBE TEST DATASET.

Method	**Top-1 accuracy**	**Top-2 accuracy**
MASSS [4]	94.9%	-
Dance-CNN with aggregation	93.9%	97.5%
Dance-CNN without aggregation	86.6%	-

TABLE III
RESULTS ON THE EXTENDED BALLROOM DATASET. RESULTS OF THE MASSS METHOD [4] ARE TAKEN FROM THE PAPER AND ARE NOT FULLY COMPARABLE.

High accuracy of the proposed method is confirmed on much larger Extended ballroom dataset (4000+ recordings), despite the recordings are only 30 seconds long. The accuracy is similar to MASSS [4], however, the results are most likely not fully comparable, since the paper does not reveal the test protocol and how the method was trained. It is unclear if the training and test sets were independent. Note that our method (Dance-CNN) and the Baseline algorithm were tested in cross-dataset setting (trained on the Private training set).

For detailed insight, we provide confusion matrices of Dance-CNN in Fig. 2. On Youtube dataset, the highest confusion occurs between Slow-waltz and Viennese-Waltz. These dances are related having highly similar patterns.

D. Other cross-dataset tests

Besides testing the proposed Dance-CNN on the studio quality Extended Ballroom and the YouTube datasets, we evaluated the method on other datasets presented earlier. Results are shown in Tab. IV

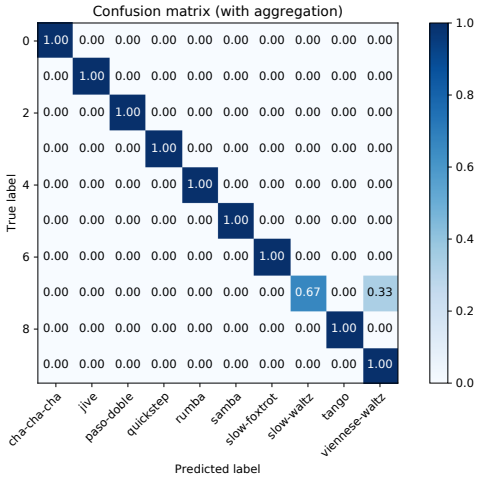
Accuracy on the Dance competitions dataset is slightly lower compared to previously tested studio quality datasets. The sound is recorded using a microphone placed in the dancing hall and, apart from the dance music, audio contains noise as steps of the dancers in the background. Nevertheless,

Dataset	**Top-1 accuracy**	**Top-2 accuracy**	**Top-1 without aggregation**
Extended ballroom	93.9%	97.5%	86.6%
YouTube test set	96.7%	100.0%	92.2%
Dance competitions	87.9%	98.6%	70.6%
StarDance	68.0%	78.0%	45.2%
Low Quality Recordings	72.7%	86.7%	58.0%

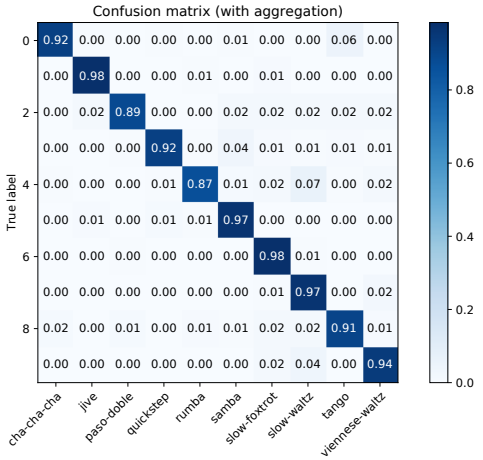
TABLE IV
RESULTS OF OUR METHOD ON VARIOUS DATASETS.

[7]https://www.worlddancesport.org/

[8]https://www.ceskatelevize.cz/porady/12607522764-stardance-x/



(a) YouTube test dataset



(b) Extended ballroom dataset

Fig. 2. Confusion matrices of our model (with aggregation).

the top-2 accuracy of the dance competitions dataset is similar to the top-2 accuracy of the YouTube test set.

Our model performed unsurprisingly worse on the StarDance dataset. The accuracy is low because the dataset contains popular songs rather than a typical ballroom music and the classification is ambiguous even for humans.

Inferior results are seen in Low quality recordings. It is not surprising because the model was trained using high-quality audio with no noise or artifacts. This problem will be addressed in the following subsection.

E. Sensitivity to Low Quality Data

To further illustrate the model sensitivity on data with background noise, we mixed a recording of crowd noise, extracted from YouTube video[9], into each audio recording in YouTube test dataset with given intensity α.

The perturbed signal is a convex combination in the temporal domain:

$$\widetilde{x}(t) = (1-\alpha)\ x_{signal}(t) + \alpha\ x_{noise}(t),$$

[9]https://www.youtube.com/watch?v=IKB3Qiglyro

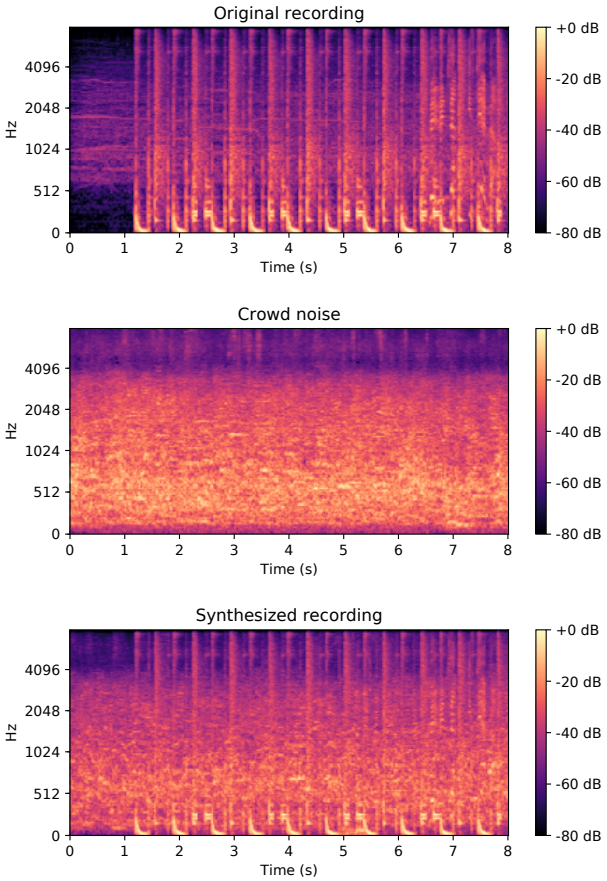


Fig. 3. Perturbed data. MEL spectrograms of Jive (top) mixed, with the recording of crowd noise (middle), resulting in the synthesized recording (bottom), $\alpha = 0.5$.

where x_{signal} is the original signal and x_{noise} the signal of the noise recording. We call the resulting set of audio recordings created by mixing the noise into the recordings in the YouTube test set, as the Synthetic test set.

Both signals, the original from the test set and the crowd noise, are normalized such that $\max x(t) = 1$.

Root Mean Square power, $RMS = \sqrt{1/n \cdot \sum_t x(t)^2}$, of the normalized signal was $RMS_{signal} \doteq 0.23$ in average, and $RMS_{noise} \doteq 0.16$ of the noise. Thus, for $\alpha = 0.5$, Signal to noise ratio was $SNR = RMS^2_{signal}/RMS^2_{noise} \doteq 1.95$. An example is shown in Fig. 3 to illustrate a strength of the perturbation.

a) Synthetic test: First, we test the method on the YouTube dataset with increasing level of perturbation, spanning from $\alpha = 0$ (original data) to $\alpha = 1$ (background noise only). The results are shown in Fig. 4 (top). As seen, results are deteriorating with increasing noise level.

b) Extending the training set: To improve the model accuracy on low quality data, we added low quality audio recordings to the training dataset. These recordings are similar to those low quality recordings used for testing, as they are extracted from the same YouTube channel. However, the

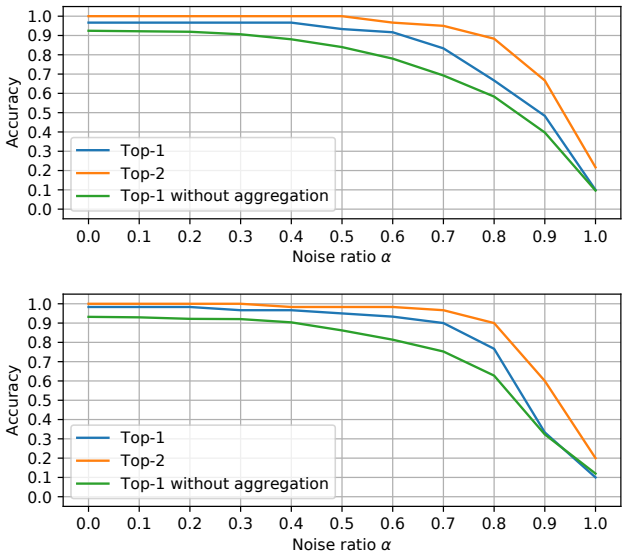


Fig. 4. Perturbation experiment. Accuracy on YouTube test set with increasing intensity, for the original model (top) and for the model trained the extended training set (bottom).

Dataset	Top-1 accuracy	Top-2 accuracy	Top-1 without aggregation
Extended ballroom	92.4%	96.9%	84.7%
YouTube test dataset	**98.3%**	100.0%	**92.7%**
Dance competitions	**93.7%**	**98.9%**	**75.7%**
StarDance	46.0%	74.0%	38.6%
Low Quality Recordings	**89.8%**	**95.3%**	**71.7%**

TABLE V
RESULTS OF A MODEL, WHERE LOW QUALITY RECORDINGS WERE ADDED TO THE TRAINING SET. ACCURACIES THAT INCREASED COMPARED TO THE ORIGINAL MODEL IN TABLE IV ARE BOLDED.

datasets do not overlap. We added 239 low quality recordings which corresponds to 4.88% of training set. Then, new model is trained on the extended dataset.

To evaluate the effect of extending the training set, we first repeat the test with synthetically perturbed data. See Fig. 4 (bottom) for results. New model has higher accuracy, Top-1 accuracy is almost constant until $\alpha = 0.6$.

Moreover, we re-tested the new model on all datasets. Results are depicted in Tab. V. Compare with results of the original model trained on studio quality recordings only in Tab. IV. Substantially better performance is achieved on the Low quality dataset. The resulting accuracy increased by $89.8\% - 72.7\% = 17.1\,\text{p.p.}$. Accuracy on the Dance competitions dataset also increased. The reason is probably that the dance competitions have certain amount of background noise that the extended model can handle. Decreased accuracy is seen in StarDance dataset and it decreased by $68\% - 46\% = 12\,\text{p.p.}$ The accuracy decrease of the StarDance dataset is mostly caused by ambiguity of the StarDance dataset which is indicated by the top-2 accuracy with slight decrease of $78\% - 74\% = 4\,\text{p.p.}$

Architecture	Top-1 accuracy	Top-2 accuracy	Top-1 without aggregation
VGG 16	25.0%	41.7%	24.8%
ResNet-18	**96.7%**	**100.0%**	89.9%
ResNeXt-50 32x4d	95.0%	**100.0%**	89.6%
DenseNet 161	**96.7%**	**100.0%**	**92.2%**

TABLE VI
ACCURACIES ON THE TEST DATASET FOR GIVEN CNN ARCHITECTURES. THE HIGHEST VALUES ARE **BOLDED**.

F. Comparison of CNN Architectures

We experiment with several recent CNN architectures. We compare DenseNet 161 [17] with VGG 16 [27], ResNet-18 [18] and ResNeXt-50 32x4d [28]. The training procedure was exactly the same for all architectures.

Results of tests on YouTube dataset in Tab. VI show the DenseNet outperforms other CNN architectures. While there is a significant difference between the accuracy of VGG and DenseNet, the accuracies of novel architectures DenseNet, ResNet and ResNeXt are similar.

We further experimented with training the DenseNet from scratch (random initial weights), and starting from the model pre-trained on ImageNet [16] categorization. The results were not significantly different, Top-1 accuracy without aggregation was 92.2% and 91.1% respectively.

G. Qualitative results

In following figures, we show classification results of 5.2 sec spectrogram segments of a scanning window over recordings. Each dance class have a color assigned. The output soft-max scores of classes are visualized by proportion of the colors. Note that, the plots are not cumulative scores, but the segment classification output.

An example of Slow Foxtrot music classification (https://youtu.be/ZyYn7Bw3EqA *0:00-2:00*) is Shown in Fig. 1. The model clearly predicts the class correctly, except for three problematic events before 20, 60, and 100 seconds. The events are verse changes or bridges, where the music plays with a low volume, as seen in the corresponding spectrogram. Nevertheless, aggregated results are not significantly affected.

Another example is shown in Fig. 5. It is a classification of Samba, where the recording contains a gradual music beginning that belongs to the dance music, however the patterns that are required to correctly predict dance class, as beats and timing, are not present in the beginning of the song. For these segments, our model predicts the classes almost uniformly. This is a good sign and confirms the model does not tend to be over-confident.

Example in Fig. 6 shows a classification of Slow Waltz. We can see, that high score of Viennese Waltz is predicted. These two dances are related and cause confusion more often as seen in Fig. 2.

Last example, a failure case, is shown in Fig. 7. The recording is a Waltz from Stardance, a challenging atypical music. The triple meter timing is not significant, which caused confusions.

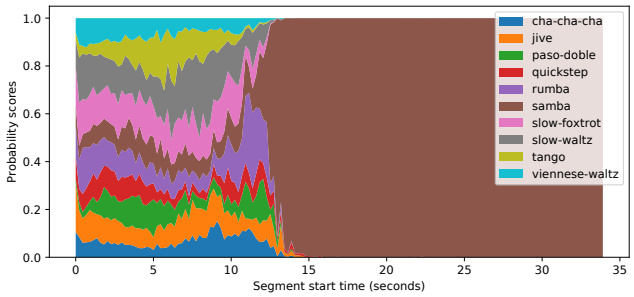


Fig. 5. Classification of Samba (https://youtu.be/h7hj3KSQ2Ec *0:00-0:40*) that contains both digitally added intro and gradual music beginning. Color-coded probability scores for consecutive segments.

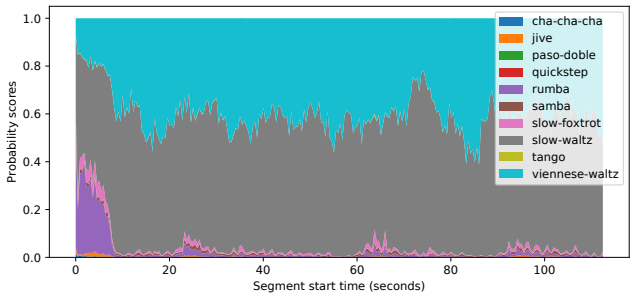


Fig. 6. Classification of Slow Waltz (https://youtu.be/dVdI1mo_va0 *0:00-2:00*) which is similar to Vienesse Waltz. Color-coded probability scores for consecutive segments.

H. Web demonstration

Our method is demonstrated using a responsive web application[10]. The application enables a user to upload audio file or record from a microphone on mobile devices, and shows the classification results. The results are represented by probability scores over the dance classes. The classification is performed using the model, that is trained on the training set extended by low quality audio recordings, as described in Section IV-E.

V. Conclusion

We presented a technique for dance genre classification. The technique relies on spectrograms that represent a segment of raw audio signal as an image. It was shown, that computer vision approaches can be effectively utilized for predicting dance genres from the spectrograms and we have achieved the best results when relying on pretrained Dense Convolutional Network [17], referred as DenseNet, with parameters learned on ImageNet [16].

We achieved remarkable results of accuracy **96.7%** on our independent test dataset of 60 recordings about four minutes long and **93.9%** accuracy on novel publicly available Extended Ballroom dataset [21] of 4000+ recordings, ten seconds long each. Our results are competitive and probably achieve the state-of-the-art in dance genre recognition, since MASSS [4] method does not provide the test protocol. We further evaluated our model on various datasets, as audio recordings

[10]http://dance.ironbrain.net

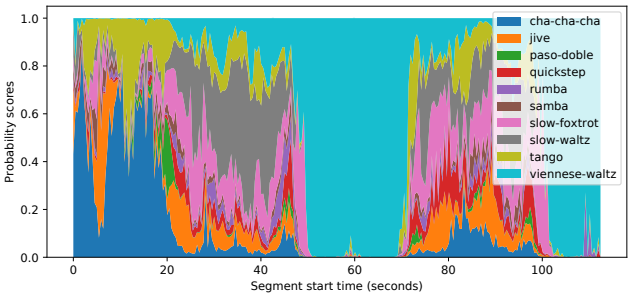


Fig. 7. Classification of ambiguous recordings of Waltz from StarDance TV show (Stardance 10, *Gabriela Koukalová a Martin Prágr - Valčík SD 6 0:00-2:00*). The recording accompanied triple meter Waltz dance, but the triple-meter timing is not significant in the music.

extracted from dance competitions, audio recordings extracted from StarDance TV show, and on Low quality dataset recorded by mobile phones severely contaminated by background noise in the dancing hall.

Based on the results of our method on low quality data, we extended the training set by adding low quality audio recordings. We have shown, that accuracy on independent Low quality dataset increased significantly, despite the ratio of the low quality data in the training was as low as 4.9%. The proposed method achieved **89.8%** accuracy on these challenging data, recorded by mobile phones, containing speech, claps, foot steps, etc. All our experiments were carried out in cross-dataset setup.

As a demonstration of our method, we provide a web application with functionality to predict dance from audio recordings uploaded by user. We made our YouTube testing set public, so any future method can be compared with our technique.

To name some limitations, we are aware the method is missing a reject option that would extend the model to abstain from prediction in ambiguous cases or on classes that the model is not trained for. At the moment, we rely on the output distribution entropy. If it is too high, close to uniform distribution, the prediction is not conclusive, as seen e.g. in the beginning in Fig. 5. Nevertheless, we have not studied out of class data, non-music data, etc. The problem might be handled by training training a proper confidence, as suggested by [29].

Another future work might be to explore options on segment aggregation. Our current approach utilizes simple averaging to aggregate the classification results of spectrogram segments. Future technique could employ more sophisticated aggregation mechanisms as recurrent neural networks.

As our method employs spectrogram as a representation of an audio signal, other methods for audio signal representation could be investigated. Moreover, end-to-end techniques could be employed to classify the dance class from the raw signal without utilizing any intermediate representation [11]. The end-to-end approach could work if enough data is available since the neural network has to learn the representation of the raw signal from scratch.

To achieve more accurate results, ensemble of classifiers could be utilized to predict a dance class [30]. Especially, CNN-based approach for music classification could be ensembled with approaches employing manually extracted features. A hand-crafted feature with a clear interpretation, e.g. music tempo, could further improve the results.

REFERENCES

[1] J. S. Downie, "Music information retrieval," *Annual review of information science and technology*, vol. 37, no. 1, pp. 295–340, 2003.

[2] S. Dixon, E. Pampalk, and G. Widmer, "Classification of dance music by periodicity patterns," in *ISMIR 2003*. Johns Hopkins University, 2003.

[3] E. Chew, A. Volk, and C.-Y. Lee, "Dance music classification using inner metric analysis," in *The next wave in computing, optimization, and decision technologies*. Springer, 2005, pp. 355–370.

[4] U. Marchand and G. Peeters, "Scale and shift invariant time/frequency representation using auditory statistics: Application to rhythm description," in *2016 IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP)*. IEEE, 2016, pp. 1–6.

[5] Y. M. Costa, L. Oliveira, A. L. Koerich, F. Gouyon, and J. Martins, "Music genre classification using lbp textural features," *Signal Processing*, vol. 92, no. 11, pp. 2723–2737, 2012.

[6] S. Lippens, J.-P. Martens, and T. De Mulder, "A comparison of human and automatic musical genre classification," in *2004 IEEE international conference on acoustics, speech, and signal processing*, vol. 4. IEEE, 2004, pp. iv–iv.

[7] B. Logan *et al.*, "Mel frequency cepstral coefficients for music modeling." in *ISMIR*, vol. 270, 2000, pp. 1–11.

[8] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE Transactions on speech and audio processing*, vol. 10, no. 5, pp. 293–302, 2002.

[9] T. Nakai, N. Koide-Majima, and S. Nishimoto, "Encoding and decoding of music-genre representations in the human brain," in *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 2018, pp. 584–589.

[10] H. Bahuleyan, "Music genre classification using machine learning techniques," *arXiv preprint arXiv:1804.01149*, 2018.

[11] S. Dieleman and B. Schrauwen, "End-to-end learning for music audio," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 6964–6968.

[12] J. Pons, T. Lidy, and X. Serra, "Experimenting with musically motivated convolutional neural networks," in *2016 14th International Workshop on Content-Based Multimedia Indexing (CBMI)*. IEEE, 2016, pp. 1–6.

[13] K. Choi, G. Fazekas, and M. Sandler, "Automatic tagging using deep convolutional neural networks," *arXiv preprint arXiv:1606.00298*, 2016.

[14] J. Pons, O. Nieto, M. Prockup, E. Schmidt, A. Ehmann, and X. Serra, "End-to-end learning for music audio tagging at scale," *arXiv preprint arXiv:1711.02520*, 2017.

[15] C. P. Tang, K. L. Chui, Y. K. Yu, Z. Zeng, and K. H. Wong, "Music genre classification using a hierarchical long short term memory (lstm) model," in *Third International Workshop on Pattern Recognition*, vol. 10828. International Society for Optics and Photonics, 2018, p. 108281B.

[16] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[17] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[19] D. Zhu, H. Yao, B. Jiang, and P. Yu, "Negative log likelihood ratio loss for deep neural network classification," *arXiv preprint arXiv:1804.10690*, 2018.

[20] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[21] U. Marchand and G. Peeters, "The extended ballroom dataset," in *2016 IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP)*, 2016.

[22] F. Gouyon, F. Pachet, O. Delerue *et al.*, "On the use of zero-crossing rate for an application of classification of percussive sounds," in *Proceedings of the COST G-6 conference on Digital Audio Effects (DAFX-00), Verona, Italy*, 2000, p. 26.

[23] D. Ellis, "Chroma feature analysis and synthesis," *Resources of Laboratory for the Recognition and Organization of Speech and Audio-LabROSA*, 2007.

[24] A. Klapuri and M. Davy, *Signal processing methods for music transcription*. Springer Science & Business Media, 2007.

[25] S. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE transactions on acoustics, speech, and signal processing*, vol. 28, no. 4, pp. 357–366, 1980.

[26] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[27] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[28] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1492–1500.

[29] V. Franc and D. Prusa, "On discriminative learning of prediction uncertainty," in *International Conference on Machine Learning*, 2019, pp. 1963–1971.

[30] C. N. Silla Jr, C. A. Kaestner, and A. L. Koerich, "Automatic music genre classification using ensemble of classifiers," in *2007 IEEE International Conference on Systems, Man and Cybernetics*. IEEE, 2007, pp. 1687–1692.