MULTI-PITCH ESTIMATION MEETS MICROPHONE MISMATCH: APPLICABILITY OF DOMAIN ADAPTATION

Franca Bittner **Marcel Gonzalez** **Maike Richter**
Hanna Lukashevich **Jakob Abeßer**
Semantic Music Technologies Group, Fraunhofer IDMT, Ilmenau, Germany
jakob.abesser@idmt.fraunhofer.de

ABSTRACT

The performance of machine learning (ML) models is known to be affected by discrepancies between training (source) and real-world (target) data distributions. This problem is referred to as domain shift and is commonly approached using domain adaptation (DA) methods. As one relevant scenario, automatic piano transcription algorithms in music learning applications potentially suffer from domain shift since pianos are recorded in different acoustic conditions using various devices. Yet, most currently available datasets for piano transcription only cover ideal recording situations with high-quality microphones. Consequently, a transcription model trained on these datasets will face a mismatch between source and target data in real-world scenarios. To address this issue, we employ a recently proposed dataset which includes annotated piano recordings covering typical real-life recording settings for a piano learning application on mobile devices. We first quantify the influence of the domain shift on the performance of a deep learning-based piano multi-pitch estimation (MPE) algorithm. Then, we employ and evaluate four unsupervised DA methods to reduce domain shift. Our results show that the studied MPE model is surprisingly robust to domain shift in microphone mismatch scenarios and the DA methods do not notably improve the transcription performance.

1. INTRODUCTION

Recent advances in Automatic Music Transcription (AMT) enable its practical application in musical education applications where students can record themselves while playing a musical instrument and retrieve a performance feedback in near real-time. The underlying algorithms are driven by deep learning and commonly trained on audio data (source domain), which was gathered in specific and ideal recording setups such as music studios with high-quality microphones [1–3]. In real-life scenarios however, the recording setups may vary from user to user due to different recording devices, room acoustics, and music instrument timbres (target domain). Due to the resulting distribution discrepancy between both domains (domain shift), AMT algorithms might exhibit performance degradation. To overcome this issue, one approach would be to fine-tune pre-trained transcription models using labeled data recorded in real-world settings [4]. This comes with two main drawbacks. First, this procedure would have to be repeated for each user. Second, it requires a lot of effort to obtain perfectly aligned score annotations by manually transcribing audio recordings. For this reason, domain adaptation (DA) methods are used to bridge the gap between different data domains and ensure a good model performance even on previously unseen data. These methods can align the target data distribution to the source data distribution (or vice-versa) [5].

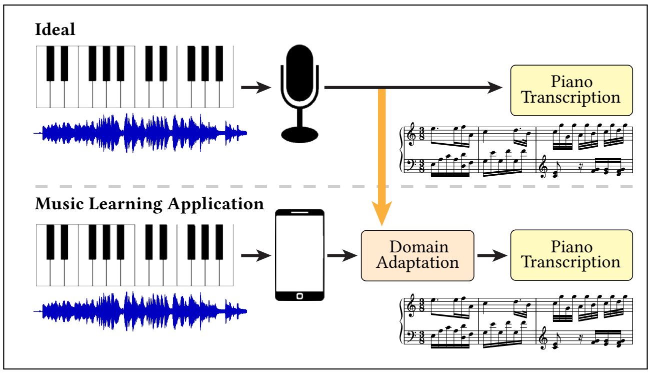

Figure 1. Illustration of the piano transcription process including domain adaptation. Piano music is captured with different recording devices (high quality microphone, mobile devices), adapted to the source domain, and transcribed by an MPE model.

The contributions of the paper are as follows: We study the task of piano multi-pitch estimation (MPE), i. e., the estimation of simultaneously sounding note pitches, from audio recordings captured with different mobile devices. We first analyze and quantify the mismatch of recording devices by comparing their frequency responses. Then, we study the effectiveness of four different DA methods as a pre-processing step to improve MPE algorithms. We do this by investigating to what extent the microphone mismatch impacts the performance of a deep learning-based

MPE model with and without DA. We use the representation shift metric to assess whether the domain shift was reduced by DA.

2. DOMAIN ADAPTATION

2.1 Application in Audio Domain

DA was successfully applied for various tasks in different audio domains. Several acoustic scene classification (ASC) algorithms were proposed during the Detection and Classification of Acoustic Scenes and Events (DCASE) challenge [6,7] to compensate domain shift caused by mismatched recording devices. Furthermore, Yang et al. [8] proposed a two-stage domain adaptation approach to improve the robustness of sound event detection (SED) models by aligning synthetic and real audio data distributions in feature space during training. DA was also applied in Music Information Retrieval (MIR) tasks, such as expressivity analysis in piano recordings [9], representation learning for music processing [10], and instrument activity detection [11], and in automatic speech recognition, DA methods are employed to avoid overfitting of a model which was trained on limited data by transferring knowledge from a source model [12].

2.2 Selected Methods

2.2.1 Zero-mean Unit Variance (ZMUV) Normalization

A common pre-processing step for machine learning (ML) models is the standardization of input features to zero mean and a standard deviation of one. We implemented four variants of the zero-mean unit variance (ZMUV) standardization process from [4] as unsupervised DA methods, which do not require data annotations. The statistics used to standardize the target domain data (mean and standard deviation coefficients) are computed either from the source dataset (*global* variants) or from the target dataset (*adaptive* variants). As a consequence, global variants require access to the source domain data whereas adaptive variants only need fractions of the target domain data.The statistics can be computed either over all available files of the selected domain or individually per file. Furthermore, it is possible to have a finer resolution when computing frequency-wise, i. e., by averaging over all time frames and obtaining coefficients per frequency bin, or patch-wise statistics, i. e., by averaging all frequencies within a patch of 16 time frames. These methods are summarized in Table 1. Finally, in addition to standardizing the target domain features, the source domain data will also be standardized before training the model.

Table 1. Overview of the implemented standardization methods w.r.t. their type (global or adaptive), data scope (whole domain or per file), and resolution (subdivision by frequency bins or patches).

Type	**Data Scope**	**Resolution**
Global	Domain (all)	Frequency
Adaptive	Domain (all)	Frequency
Adaptive	File	Frequency
Adaptive	File	Patch

2.2.2 Band-wise Statistics Matching (BWSM)

Band-Wise Statistics Matching (BWSM) is an unsupervised DA method, which involves a band-wise alignment of the first and second statistical moments of the target domain data to the ones of the source domain data [13]. Similarly to standardization, the method is applied on the data level and avoids a re-training of a machine learning model. First, the sample mean and standard deviation values are computed over source and target domain data per frequency bands. Then, a band-wise standardization is applied to the target domain in a similar way as in the adaptive ZMUV normalization per frequency as discussed in the previous section. At the final stage, the adapted features in the target domain $\mathbf{X}_{\mathrm{T}}$ are aligned to the source domain $\mathbf{X}_{\mathrm{S}}$, sharing the same means and standard deviations. In contrast to other assessed DA methods, source domain data remains unchanged and the originally trained ML models can be re-used.

2.2.3 Correlation Alignment (CORAL)

Sun et al. [14] proposed CORrelation ALignment (CORAL) as an unsupervised DA method to align the statistical moments of source and target data. After whitening the source domain data (i. e., removing correlation among features), the covariance matrix of the target domain data distribution $\mathbf{C}_{\mathrm{T}}$ is transferred to the source distribution (recoloring). Then, a new model needs to be trained on the adapted source domain data.

Given that the distribution of target domain data varies with each mobile device, room, and piano type in our given scenario, it is not feasible to train a new MPE model each time one of these parameters changes. In contrast to the original publication, we implemented CORAL such that the target domain data is adapted using the statistics obtained from the source domain data. We follow [14] and use a regularization parameter $\lambda = 1$ for traditional whitening without a singular value decomposition (SVD). The target domain data $\mathbf{D}_{\mathrm{T}}$ is whitened as in [14] and then re-colored with the source domain covariance matrix $\mathbf{C}_{\mathrm{S}}$ as $\mathbf{D}_{\mathrm{T}} \leftarrow \mathbf{D}_{\mathrm{T}} * \mathbf{C}_{\mathrm{T}}^{-\frac{1}{2}} * \mathbf{C}_{\mathrm{S}}^{\frac{1}{2}}$. This way, the source domain data remains unchanged and the same classifier can be used for all target domains. However, Sun et al. [14] observed a lower performance with this approach and supposed that a model trained on adapted source domain data may benefit from the knowledge inherited by the target data distribution. This is not possible if only the target domain data is modified by DA, as the model is trained on the original source domain data independent of the target domain data.

2.2.4 Feature Projection-Based Unsupervised Domain Adaptation (FPDA)

Mezza et al. [15] proposed Feature Projection-based Unsupervised Domain Adaptation (FPDA) to address the mismatch between the distributions of training and test data acquired under different recording conditions. FPDA is based on the projection of both source and target domain features onto a common subspace using a subset of the eigenvectors of the sample covariance matrix of the source domain data. After standardizing the spectrograms extracted from the source domain in a band-wise fashion, the sample covariance matrix $\mathbf{V}_{\mathrm{s}}$ of the source domain data matrix and the respective eigenvectors and eigenvalues are calculated. The eigenvectors corresponding to the L largest eigenvalues are retained as matrix $\mathbf{V}_{\mathrm{s}}^{(L)}$ and using this matrix source domain and target domain data are projected onto a common subspace as $\mathbf{X}_{\mathrm{S}} = \bar{\mathbf{S}}_{2\mathrm{D}}\mathbf{V}_{\mathrm{s}}^{(L)}$ and $\mathbf{X}_{\mathrm{T}} = \bar{\mathbf{T}}_{2\mathrm{D}}\mathbf{V}_{\mathrm{s}}^{(L)}$.

There are several differences between the application of FPDA in Acoustic Scene Classification (ASC) [15] and our application scenario for real-time piano transcription. Firstly, in [15], source and target domain data contain the same audio signal recorded simultaneously on different recording devices with a constant length of ten seconds, whereas the files contained in our source domain data have entirely different musical content as compared to the target domain data. Moreover, the number of samples varies greatly throughout both the source and target domain dataset in our case. For this reason and for FPDA to be applicable w.r.t. a real-time piano transcription scenario, we chose $L = 16$, so that the DA can be applied in a block-wise fashion, 16 frames at a time. To use the domain-adapted features as input for the piano transcription model, we project the features back to the original feature space as $\hat{\mathbf{S}}_{2\mathrm{D}} = \mathbf{X}_{\mathbf{S}}\mathbf{V}_{\mathrm{s}}^{(L)^T}$ as $\hat{\mathbf{T}}_{2\mathrm{D}} = \mathbf{X}_{\mathbf{T}}\mathbf{V}_{s}^{(L)^T}$. Then, the MPE model is trained on the resulting modified source domain data.

3. MULTI-PITCH ESTIMATION

3.1 Recent Approaches

While traditional MPE methods mostly relied on spectral decomposition techniques such as non-negative matrix factorization (NMF) [16], modern methods are based on different neural network (NN) architectures [17]. Hawthorne et al. [18] propose a general purposed piano transcription model that combines two branches of convolutional recurrent neural networks (CRNNs), which jointly predict the pitches and onset times of played piano notes. Kong et al. [19] proposed a high-resolution AMT system consisting of several CRNNs acoustic models dedicated to different tasks such as velocity, onset and offset regression. More recently, generic encoder-decoder architectures such as transformer networks have been used to remove the need of custom NN design, as done by Hawthorne et al. [20].

3.2 Model Architecture

Following recent advances in the field of computer vision, the U-net architecture is employed for different MIR tasks. In particular, it was used for different AMT tasks such as bass transcription [21], melody transcription [22, 23], as well as polyphonic piano transcription [24]. The structure of a U-net resembles an autoencoder, but is extended by skip connections between encoder and decoder blocks. Input features are first processed by an encoder that sequentially reduces the time-frequency resolution. The intermediate features are then handed over to the decoder part at the so-called bottleneck, where the compression is strongest. In the decoder, the time-frequency resolution is gradually restored by upsampling and interpolation. The main goal is to train the U-net such that it learns a mapping function from a spectrogram-like audio representation to a piano roll representation. In this study, we consider MPE as binary classification task, where each pitch can be either active or inactive at a given time frame. The encoder of our U-net model comprises three layers with alternating blocks of convolutional filter blocks and max pooling. In the decoder, bilinear interpolation operations are used for upsampling. Intermediate activations at similar levels in the encoder and decoder are connected by skip connections. The model output lies in the same feature space as the input features, but with a reduced frequency resolution of 72 bins instead of 216 bins.

The model was trained for 300 epochs with the Adam optimizer at an initial learning rate of $5 \cdot 10^{-4}$, early stopping patience of 20, and the binary crossentropy loss function.

3.3 Audio Representation

The MPE model expects input features as an harmonic Constant-Q transform (HCQT) [25] $H_{h,f,t}$ indexed by harmonic ratio h, frequency f, and time t. For any harmonic $h > 0$, we compute a standard Constant-Q transform (CQT), where the minimum frequency is scaled by the harmonic ratio h. For this MPE task, the HCQTs are computed for the harmonic ratios $h \in \{0.5, 1, 2, 3, 4, 5\}$, that is, one sub-harmonic below the fundamental frequency ($h = 0.5$), the fundamental frequency ($h = 1$) and four harmonics above ($h = [2, 3, 4, 5]$). As the fundamental frequency and the first harmonic often contain similar harmonic patterns [25], one sub-harmonic below the fundamental is included to distinguish between the two.

Features are extracted with the *librosa* Python library [26] with a hopsize of 512 samples, a sampling rate $f_{\mathrm{s}} = 22.05$ kHz, a frequency resolution of 36 bins per octave (bpo), and a minimum frequency for the CQT of $f_{\min} = 32.7$ Hz. This results in input features of dimension $\mathbf{X} \in \mathbb{R}^{M \times K \times C}$, with the number of time frames $M = 16$ for patch-wise processing, number of frequency bins $K = 216$, and amount of channels $C = 6$.

4. DATASET

4.1 Source Dataset

The U-net MPE model was trained and tested on subsets of the MIDI Aligned Piano Sounds (MAPS) dataset [1]. We used the dataset split labelled "Configuration 2" by Sigtia et al. [27], which divides MAPS into 210 synthesized audio files for training and 60 real-world audio recordings for test data. The audio and aligned MIDI data used for training were generated using synthesis software based on high-quality sample libraries, whereas the test set contains piano recordings from a MIDI-controlled Disklavier which was recorded on two omnidirectional Schoeps microphones in a studio setting. We refer to the training data as `MAPS-train` and test data as `MAPS-test`.

4.2 Target Dataset

To assess the selected DA methods, we require a dataset of multiple transcribed piano recordings which cover different mobile devices as recording devices, recording locations with different acoustical properties, and various acoustic piano models (upright and grand piano). Because no openly available dataset fulfilled all our requirements, we created a new dataset called `IDMT-PIANO-MM` [28].

`IDMT-PIANO-MM` contains a total of 432 audio files, with a total duration of four hours and seven minutes, and 72 MIDI files corresponding to 72 unique piano performances recorded simultaneously on five recording devices (3 smartphones, 2 tablets) and a high quality stereo-microphone. The 72 MIDI files were generated and aligned manually to match the actual piano performance. The recording environments range from small rooms to large lecture halls. Three grand pianos, four upright pianos, and one electronic stage piano of various age, brand and price segments were recorded. In each room we recorded a human piano performance of a self-composed swing pattern, chord progression, a chromatic scale, as well as sections (between 24 s and 54 s) out of five classical music pieces and one ragtime piece.

5. ANALYZING DOMAIN SHIFT

5.1 Investigation of Microphone Mismatch

As the first step to quantify microphone mismatch, we compute the spectrum correction coefficients as described in [29] to compare the acoustic characteristics of different recording devices. To start, we choose a reference device r, which corresponds to a high-quality recording microphone, and compute the correction coefficients of all mobile devices d with respect to the reference. By comparing the frequency responses of the device and the reference for each frequency bin, we obtain a vector of multiplicative correction coefficients, which allows to make the frequency response of the device d resemble the one of the reference device r. Figure 2 illustrates the band-wise coefficients obtained for each mobile device. The correction coefficients show that there are considerable differences among the recordings associated to the different devices and their microphones. This confirms the microphone mismatch condition and suggests there is a domain shift between the training data from MAPS and the test data recorded with different mobile devices.

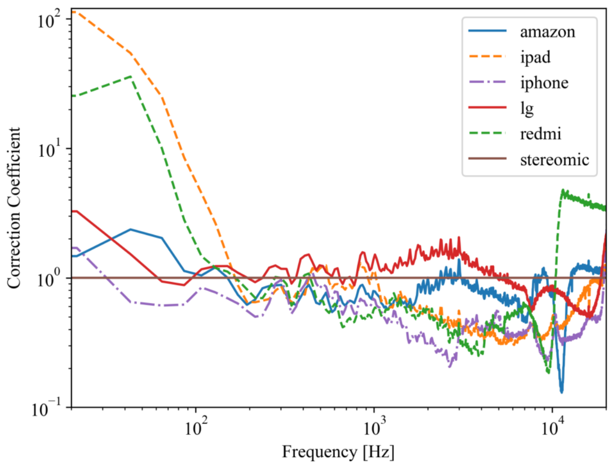

Figure 2. Frequency-dependence of the spectrum correction coefficients for all devices in respect to the `stereomic` microphone.

5.2 Quantifying Domain Shift

Due to the observed differences in the acoustic characteristics of the recording devices, we expect a measurable domain shift, which we quantify using the representation shift metric proposed by Stacke et al. [30]. The main idea is to compute an abstract representation of source and target domain features from a latent representation within a deep neural network and compare the data distributions between both domains in this latent feature space. In our experiments, we computed the activations after the last convolutional layer of the encoder in the U-net MPE model to measure domain shift, as we expect the highest degree of abstraction here.

In [30], the layer activations at layer l and filter f are averaged across the two spatial dimensions of a feature map. Since we deal with audio instead of images, these dimensions correspond to time frames and frequency bins. As the convolutional neural network (CNN) structure used in [30] is not directly comparable with the U-net architecture, we assumed that the U-net only has one filter operation per layer for simplicity. The mean value of a layer activation $\phi_l(x)$ at layer l is denoted as

$$c_l(x) = \frac{1}{N}\frac{1}{K}\sum_{i=1}^{N}\sum_{j=1}^{K}\phi_l(x)_{i,j} \tag{1}$$

with the number of time frames N, number of frequency bands K, and i and j as time and frequency index of the feature map, respectively. The mean values of layer activations are aggregated over all input features of a domain. The probability density function (PDF) of all mean values is approximated by computing a probability density histogram with 500 bins for each HCQT channel. The resulting PDFs are averaged over all HCQT channels. Finally, we compute the representation shift r as the mean distance between the distributions of the source and target domain. We employ the Wasserstein distance, which is a symmetric distance measure that can be seen as the least amount of

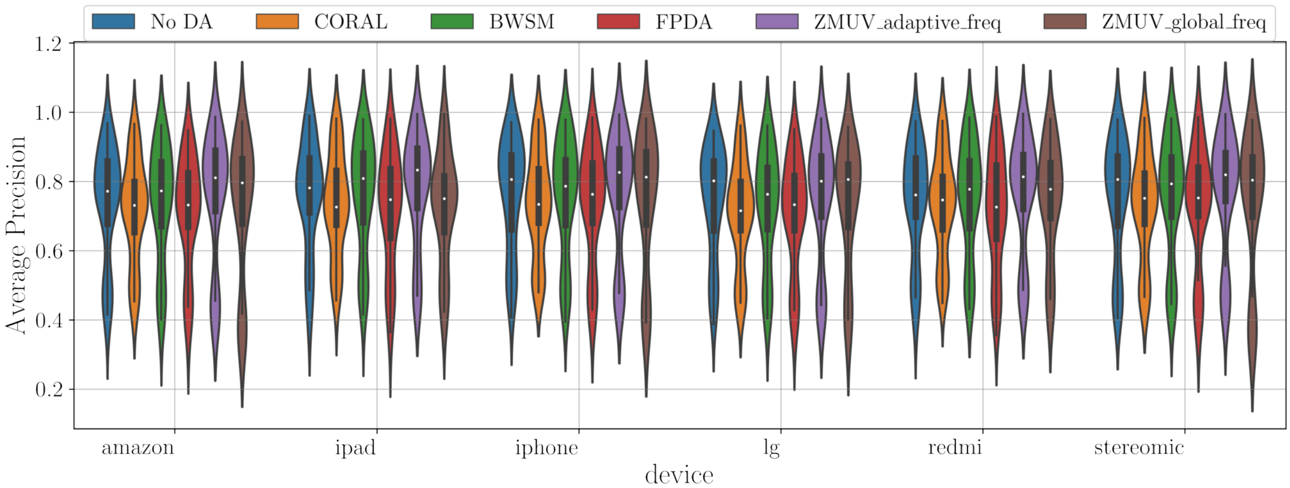

Figure 3. Performance of the MPE model on `IDMT-PIANO-MM`. The average precision score is presented for different DA methods grouped by recording devices.

Domain Pairs	r
`MAPS-train` vs. `IDMT-PIANO-MM`	0.035
`MAPS-train` vs. `IDMT-PIANO-MM-BWSM`	0.029
`MAPS-train` vs. `MAPS-test`	0.029

Table 2. Representation shift r based on the Wasserstein distance between source training data and source test data (`MAPS-train` vs. `MAPS-test`), source training data and target data (`MAPS-train` vs. `IDMT-PIANO-MM`), source training data and with BWSM domain-adapted target data (`MAPS-train` vs. `IDMT-PIANO-MM-BWSM`).

effort required to align two distributions w.r.t. the amount of data and the distance that has to be moved [30]. The representation shift between the used datasets is given in Table 2. The DA method BWSM was chosen as, in contrast to other DA methods, source domain data is not modified, which enables to directly compare the implementation with and without DA with identical MPE models.

6. IMPACT OF DOMAIN SHIFT ON PERFORMANCE

The performance of the MPE model is tested by comparing the estimated pitches with MIDI data as ground truth per frame. We chose mean Average Precision (mAP), i. e., the area under the precision-recall curve [31], as an evaluation metric, which is denoted as Average Precision (AP) in the following. Unlike other popular ML evaluation metrics like F-score or accuracy, AP does not depend on a particular binarization threshold for the pitch activity predictions.

6.1 Effect of Domain Adaptation

Figure 3 shows the AP scores of the U-net MPE model on the benchmark dataset separated by recording device. It shows that the microphone mismatch has no significant effect on the performance of the MPE model. Presumably as a direct consequence, we observe only little variation in transcription performance caused by DA. `ZMUV adaptive` performs consistently best, `ZMUV global` and `BWSM` report a similar performance as using no DA, and with `FPDA` and `CORAL` the AP scores are lowest amongst all DA methods, including not applying any DA.

6.2 Impact of Musical Content and Acoustic Context

We also assessed the impact of other parameters on the MPE performance, such as room acoustics and musical pieces. Our results showed no significant influence of the room and instrument characteristics towards the MPE performance. In contrast, the musical content shows the greatest impact on transcription performance. Figure 4 displays the influence of music content on the AP score when no DA or different DA methods are applied. Due to missing annotations of offsets, we assume that the MPE performance is lower for `beethoven1` and `beethoven2` as these pieces involve the use of the sustain pedal. While some of the DA methods vary in effectiveness for different music content, like FPDA and CORAL, overall the results of DA seem to be quite consistent and do not improve performance significantly regardless of the musical content.

7. DISCUSSION

Although the examined DA methods reduced the domain shift between `MAPS-train` and our test dataset `IDMT-PIANO-MM` (see Table 2), the MPE performance could not be improved significantly by DA. We see several possible explanations for this.

First, it should be noted that the applied method to quantify domain shift assumes that each convolutional layer of the MPE model contains only one filter, which is not given with the U-net MPE model. Further research must investigate how this discrepancy leads to a biased measurement. Yet, we expect that the computed domain shift values can

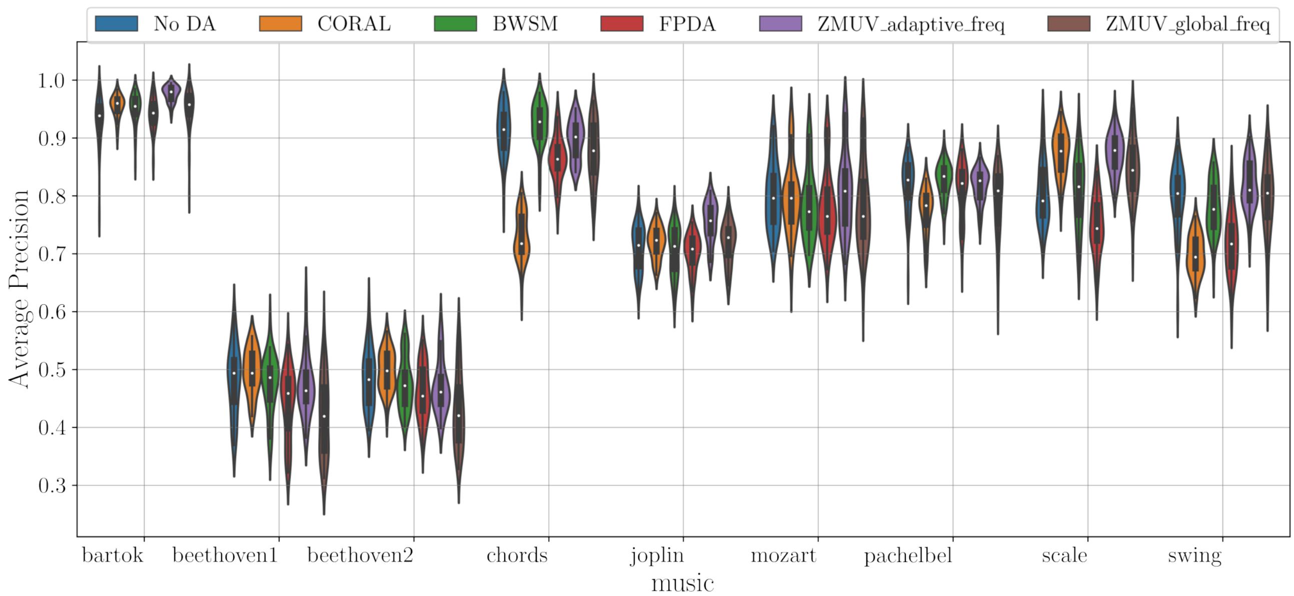

Figure 4. Performance of the MPE model on `IDMT-PIANO-MM`. The average precision score is presented for different DA methods grouped by music pieces.

still provide a general tendency.

Second, the employed MPE model might already generalize well enough to the given task. In fact, the U-net model itself can be considered as a reconstruction-based deep DA implementation as defined in [5]. The encoding procedure ensures that domain-specific features are disregarded, while the decoding part enables to attribute a feature within a domain. Therefore, it would be interesting to examine the presented DA methods for MPE models with other network architectures.

Third, the benchmark dataset `IDMT-PIANO-MM` does not provide precise offset annotations, which can introduce label noise to the frame-wise MPE evaluation.

Finally, there are further restrictions for particular DA methods. Our implementation of FPDA involves cutting the features after 16 time frames (about 0.372 s), whereas the original approach [15] keeps 10 s of data for the transformation matrix. As we reduced the size of the transformation matrix extremely, our implementation could be inaccurate. Moreover, unlike in the original paper [15] it was not possible to use recordings with identical content (denoted as "parallel data" in [13]). This requires audio data to be simultaneously captured using source and target domain devices, which is not applicable in this study. Besides, CORAL is implemented as time-independent DA since frequency patterns are only compared within one time frame. The domain-specific knowledge of time is disregarded due to buffer memory limits. As there is no reference implementation for CORAL in the audio domain to our knowledge, we cannot compare this approach to other findings. The effectiveness of adaptive ZMUV normalization is restricted by the sole dependency on recorded target data. Unlike in global ZMUV normalization, no source domain data is used to modify target data. Hence, if the amount of cached target data is too small, the DA may fail.

8. CONCLUSION

In this paper, we studied the influence of domain shift to piano MPE under microphone mismatch conditions. As our first contribution, we created and published a novel benchmark dataset of piano recordings captured with various mobile devices. In our initial experiments, we verified differences in the used recording devices using the spectrum correction coefficient method. As a second step, we quantified the domain shift between the source domain (`MAPS-train`) and different target domains of a novel benchmark dataset (`IDMT-PIANO-MM`) using the representation shift based on the Wasserstein distance between distributions of intermediate activation map values of the MPE model. We investigated in particular the performance of an MPE model based on the U-net architecture. As expected, the domain shift was greater between source and target data than between two subsets of the MAPS dataset (0.035 vs. 0.029 respectively), possibly due to different types of recording settings.

As a third step, we investigated the influence of domain shift on the MPE performance and found that the U-net model is surprisingly robust to domain shift conditions. We assume the main reason for this is that spectral peaks contain the main information for the MPE. Domain shift, however, mainly causes deviations in the overall spectral envelope of the signal, which are mostly irrelevant for the given task. Finally, we evaluated four different unsupervised DA methods in order to reduce the domain shift. The DA method BWSM was able to reduce the measured domain shift from the initial 0.035 to 0.029.

In future research, we aim to evaluate additional MPE models based on other neural network architectures, such as CRNNs or transformer models, to assess the influence of the network architecture on the robustness against domain shift.

9. ACKNOWLEDGEMENTS

This work has been supported by the German Research Foundation (AB 675/2-2). We would like to thank the reviewers for their valuable suggestions and comments.

10. REFERENCES

[1] V. Emiya, N. Bertin, B. David, and R. Badeau, "MAPS - A piano database for multipitch estimation and automatic transcription of music," Telecom ParisTech, Paris, France, Research Report, 2010.

[2] M. Müller, V. Konz, W. Bogler, and V. Arifi-Müller, "Saarland music data (SMD)," in *Late-Breaking and Demo Session of the 12th International Conference on Music Information Retrieval (ISMIR)*, Miami, United States, 2011.

[3] C. Hawthorne, A. Stasyuk, A. Roberts, I. Simon, C.-Z. A. Huang, S. Dieleman, E. Elsen, J. Engel, and D. Eck, "Enabling Factorized Piano Music Modeling and Generation with the MAESTRO Dataset," in *Proceedings of the International Conference on Learning Representations (ICLR)*, New Orleans, United States, 2019.

[4] D. S. Johnson and S. Grollmisch, "Techniques Improving the Robustness of Deep Learning Models for Industrial Sound Analysis," in *Proceedings of the 28th European Signal Processing Conference*. Amsterdam, The Netherlands: IEEE, 2021, pp. 81–85.

[5] M. Wang and W. Deng, "Deep Visual Domain Adaptation: A Survey," *Neurocomputing*, vol. 312, pp. 135–153, 2018.

[6] A. Mesaros, T. Heittola, and T. Virtanen, "Acoustic Scene Classification in DCASE 2019 Challenge: Closed and Open Set Classification and Data Mismatch Setups," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2019 Workshop (DCASE2019)*. New York City, United States: New York University, 2019, pp. 164–168.

[7] Y. Kawaguchi, K. Imoto, Y. Koizumi, N. Harada, D. Niizumi, K. Dohi, R. Tanabe, H. Purohit, and T. Endo, "Description and Discussion on DCASE 2021 Challenge Task 2: Unsupervised Anomalous Sound Detection for Machine Condition Monitoring under Domain Shifted Conditions," *arXiv e-prints: 2106.04492*, 2021.

[8] L. Yang, J. Hao, Z. Hou, and W. Peng, "Two-stage Domain Adaptation for Sound Event Detection," in *Detection and Classification of Acoustic Scenes and Events (DCASE)*, Tokyo, Japan, 2020, pp. 230–234.

[9] S. Chowdhury and G. Widmer, "Towards Explaining Expressive Qualities in Piano Recordings: Transfer of Explanatory Features Via Acoustic Domain Adaptation," in *Proceedings of the ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 561–565.

[10] Y.-J. Luo and L. Su, "Learning Domain-Adaptive Latent Representations of Music Signals Using Variational Autoencoders," in *19th International Society for Music Information Retrieval Conference (ISMIR)*, Paris, France, 2018, pp. 653–660.

[11] C. Nyströmer, "Musical Instrument Activity Detection using Self-Supervised Learning and Domain Adaptation," Master's thesis, KTH Royal Institute of Technology, Stockholm, Sweden, 2020.

[12] T. Asami, R. Masumura, Y. Yamaguchi, H. Masataki, and Y. Aono, "Domain adaptation of DNN acoustic models using knowledge distillation," in *Proceedings of the 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 5185–5189.

[13] A. I. Mezza, E. A. P. Habets, M. Müller, and A. Sarti, "Unsupervised Domain Adaptation for Acoustic Scene Classification Using Band-Wise Statistics Matching," in *Procedings of the 2020 28th European Signal Processing Conference (EUSIPCO)*, Amsterdam, The Netherlands, 2021, pp. 11–15.

[14] B. Sun, J. Feng, and K. Saenko, "Return of Frustratingly Easy Domain Adaptation," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 30, no. 1, pp. 2058–2065, 2016.

[15] A. I. Mezza, E. A. P. Habets, M. Müller, and A. Sarti, "Feature Projection-Based Unsupervised Domain Adaptation for Acoustic Scene Classification," in *Proceedings of the 2020 IEEE 30th International Workshop on Machine Learning for Signal Processing (MLSP)*, Espoo, Finland, 2020, pp. 1–6.

[16] P. Smaragdis and J. C. Brown, "Non-negative matrix factorization for polyphonic music transcription," in *Proceedings of the 2003 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA) (IEEE Cat. No.03TH8684)*, New Paltz, NY, USA, 2003, pp. 177–180.

[17] E. Benetos, S. Dixon, Z. Duan, and S. Ewert, "Automatic Music Transcription: An Overview," *IEEE Signal Processing Magazine*, vol. 36, no. 1, pp. 20–30, 2019.

[18] C. Hawthorne, E. Elsen, J. Song, A. Roberts, I. Simon, C. Raffel, J. Engel, S. Oore, and D. Eck, "Onsets and Frames: Dual-Objective Piano Transcription," in *19th International Society for Music Information Retrieval Conference (ISMIR)*, Paris, France, 2018, pp. 50–57.

[19] Q. Kong, B. Li, X. Song, Y. Wan, and Y. Wang, "High-Resolution Piano Transcription With Pedals by Regressing Onset and Offset Times," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3707–3717, 2021.

[20] C. Hawthorne, I. Simon, R. Swavely, E. Manilow, and J. Engel, "Sequence-to-Sequence Piano Transcription with Transformers," in *Proceedings of the 22nd International Society for Music Information Retrieval Conference (ISMIR)*, Online, 2021, pp. 246–253.

[21] J. Abeßer and M. Müller, "Jazz Bass Transcription Using a U-Net Architecture," *Electronics*, vol. 10, no. 6, 670, 2021.

[22] T.-H. Hsieh, L. Su, and Y.-H. Yang, "A Streamlined Encoder/decoder Architecture for Melody Extraction," in *Proceedings of the ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Brighton, UK, 2019, pp. 156–160.

[23] G. Doras, P. Esling, and G. Peeters, "On the use of U-Net for dominant melody estimation in polyphonic music," in *Proceedings of the International Workshop on Multilayer Music Representation and Processing (MMRP) 2019*, Milano, Italy, 2019, pp. 66–70.

[24] Y.-T. Wu, B. Chen, and L. Su, "Polyphonic Music Transcription with Semantic Segmentation," in *Proceedings of the ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Brighton, UK, 2019, pp. 166–170.

[25] R. M. Bittner, B. McFee, J. Salamon, P. Li, and J. P. Bello, "Deep Salience Representations for F0 Estimation in Polyphonic Music," in *Proceedings of the 18th International Society for Music Information Retrieval Conference*, Suzhou, China, 2017, pp. 63–70.

[26] B. McFee, A. Metsai, M. McVicar, S. Balke, C. Thomé, C. Raffel, F. Zalkow, A. Malek, Dana, K. Lee, O. Nieto, D. Ellis, J. Mason, E. Battenberg, S. Seyfarth, R. Yamamoto, viktorandreevichmorozov, K. Choi, J. Moore, R. Bittner, S. Hidaka, Z. Wei, nullmightybofo, D. Hereñú, F.-R. Stöter, P. Friesch, A. Weiss, M. Vollrath, T. Kim, and Thassilo, "Librosa/librosa: 0.8.1rc2," DOI: 10.5281/zenodo.4792298, 2021.

[27] S. Sigtia, E. Benetos, and S. Dixon, "An End-to-End Neural Network for Polyphonic Piano Music Transcription," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 5, pp. 927–939, 2016.

[28] J. Abeßer, F. Bittner, M. Richter, M. Gonzalez, and H. Lukashevich, "A Benchmark Dataset to Study Microphone Mismatch Conditions for Piano Multipitch Estimation on Mobile Devices," in *Proceedings of the Digital Music Research Network One-day Workshop*. London, United Kindom: Centre for Digital Music (C4DM), 2021, p. 22.

[29] M. Kośmider, "Spectrum Correction: Acoustic Scene Classification with Mismatched Recording Devices," in *Proceedings of INTERSPEECH 2020*, Shanghai, China, 2020, pp. 4641–4645.

[30] K. Stacke, G. Eilertsen, J. Unger, and C. Lundström, "Measuring Domain Shift for Deep Learning in Histopathology," *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 2, pp. 325–336, 2021.

[31] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes (VOC) Challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010.