Mixing Time-Frequency Distributions for Speech Command Recognition using Convolutional Neural Networks

1st Reemt Hinrichs
Institut für Informationsverarbeitung
Leibniz Universität Hannover
Hannover, Germany
hinrichs@tnt.uni-hannover.de

2nd Jonas Dunkel
Institut für Informationsverarbeitung
Leibniz Universität Hannover
Hannover, Germany

3rd Jörn Ostermann
Institut für Informationsverarbeitung
Leibniz Universität Hannover
Hannover, Germany

***Abstract*—Automatic speech command recognition systems have become a common technology of the day to day life for many people. Smart devices usually offer some ability to understand more or less complex spoken commands. Many such speech recognition systems use some form of signal transformation as one of the first steps of the processing chain to obtain a time-frequency representation.**

A common approach is the transformation of the audio waveforms into spectrograms with subsequent computation of the mel-spectrograms or mel-frequency cepstral coefficients. However, superior time-frequency distributions (TFDs) have been proposed in the past to improve on the spectrogram.

This work investigates the usefulness of various TFDs for use in automatic speech recognition algorithms using convolutional neural networks. On the Google Speech Command Dataset V1, the best single TFD was found to be the spectrogram with a window size of 1024 achieving a mean accuracy of $93.1\,\%$. However, a mean accuracy of $95.56\,\%$ was achieved through TFD mixing. Mixing of the TFDs thereby increased the mean accuracy by up to $2.46\,\%$ with respect to the individual TFDs.

***Index Terms*—time-frequency distribution, automatic speech recognition, s-transform, wigner-ville distribution, convolutional neural networks**

I. Introduction

Automatic speech recognition deals with the problem of extracting text representations and meaning from recorded waveforms of human speech [1], [2]. For a long time, hidden markov models together with gaussian mixture models represented the state-of-the-art [2], [3], however, with the rise of machine learning during the past decade hybrid systems combining HMM with deep neural networks achieved better performance [2]. Recent research focused on end-to-end automatic speech recognition systems, where an input audio waveform is directly mapped to some text representation through the use of deep neural networks [4].

This work is concerned with the time-frequency representation of audio waveforms which forms the first step in many speech command recognition and audio classification systems [5]–[7]. Only information contained in this representation can be used by subsequent signal processing.

The commonly used spectrogram, while easy to use and computationally inexpensive, has the downside of requiring to decide for a trade-off between time- and frequency resolution [8], implemented through the window- and overlap-choice of the underlying short-time fourier transform. Moreover, no matter the window, the spectrogram never yields an exact representation of certain important signal properties like instantanous frequency or group delay [9].

In this sense, time-frequency distributions (TFDs) like the Wigner-Ville distribution (WVD) are superior to the spectrogram. While the WVD does not require the selection of a window, it also for example correctly yields the instantanous frequency and group delay [9]. The WVD is regarded as a TFD with generally optimal time-frequency resolution [10], at least within Cohen's class of TFDs.

However, despite these desirable properties, the WVD has not claimed the position of the spectrogram in the field of signal analysis due to two main reasons: the acausality and the occurence of so called cross-terms which generally arise in multicomponent signals [9], [10].

While the cross-terms might be detrimental to the interpretation by humans, it is not obvious that they have a per se negative impact on machine learning algorithms [11], which might be able to extract information or detect patterns where a human cannot. The WVD and its derivates have been used succesfully in numerous publications [12]–[14].

In this work, the spectrogram, the Wigner-Ville Distribution, the Pseudo Wigner-Distribution, the Filtered Wigner-Distribution as well as the Stockwell Transform are being investigated for audio classification.

While other authors have investigated the benefit of TFDs other than the spectrogram for audio classification [15], these are mostly limited to wavelet or gabor transform or related TFDs. The "classical" TFDs related to the WVD have been mostly neglected for audio classification. Aside from [13], there appears to be no publication dedicated to audio classification based on the WVD using machine learning.

Aside from investigating Cohen's class TFDs for audio classification, the major idea of this work is the mixing of TFDs. The intuitive idea why this could be beneficial is that each TFDs should provide a different view of a processed signal

and thus could deliver novel information to a classification algorithm.

This idea for audio classification, also called feature stacking or feature mixing, was recently also suggested in other work [15]–[18]. However, these authors did not investigate Cohen's class TFDs. Furthermore, they do not provide an indepth analysis of the benefits like our work, where we investigate the usefulness in three different approaches, individually, in a post-mix and in a pre-mix. Finally, we performed a thorough investigation of hyperparameters to guarantee the superiority of distribution mixing.

In Section II the TFDs investigated in this work are briefly explained. In Section III the applied speech command recognition approaches are presented. In Section IV the performance of the individual TFDs and the mixing approaches are presented. Finally, the results are discussed in Section V and the paper concludes in Section VI.

II. Time-Frequency Distributions

In this section the time-frequency distributions used in this work are discussed. All integrals are improper integrals from $-\infty$ to ∞. $s(t)$ denotes a given signal.

The spectrogram $P_h(t,\Omega)$ is defined as [9]

$$P_h(t,\Omega) = |S_h(t,\Omega)|^2 = |\frac{1}{\sqrt{2\pi}} \int e^{-j\Omega\tau} s(\tau) h(t-\tau) d\tau|^2, \tag{1}$$

where $S_h(t,\Omega)$ is the short-time fourier transform (STFT) with window (function) $h(t)$. Ω is the circular-frequency. The dependency on $h(t)$ was made explicit to highlight the spectrogram technically being a class of time-frequency distributions. The window function determines the time and frequency resolution of the spectrogram. The Wigner-Ville distribution $W(t,\Omega)$ is defined [9] as

$$W(t,\Omega) = \frac{1}{2\pi} \int \overline{s}(t-\frac{1}{2}\tau) s(t-\frac{1}{2}\tau) e^{-j\tau\Omega} d\tau, \tag{2}$$

where $\overline{s}(t)$ denotes the complex-conjugate of $s(t)$. While $W(t,\Omega)$ satisfies many desireable properties [10], the existence of cross-terms due to its nonlinearity usually is considered a large drawback. Closely related is the Pseudo Wigner-Ville distribution $W_{PS}(t,\Omega)$, which is defined as

$$W_{PS}(t,\Omega) = \int h(\tau) \overline{s}(t-\frac{1}{2}\tau) s(t-\frac{1}{2}\tau) e^{-j\tau\Omega} d\tau, \tag{3}$$

where the window function $h(t)$ was introduced with the aim of surpressing the detrimental cross-terms of the Wigner-Ville distribution.

Many distributions including all of this work except for the Stockwell transform, are special cases of Cohen's class of TFDs [9], which is given by

$$\frac{1}{4\pi^2} \int e^{-j-j\tau\Omega+j\theta u} \phi(\theta,\tau) \overline{s}(u-\frac{1}{2}\tau) s(u-\frac{1}{2}\tau) du d\tau d\phi, \tag{4}$$

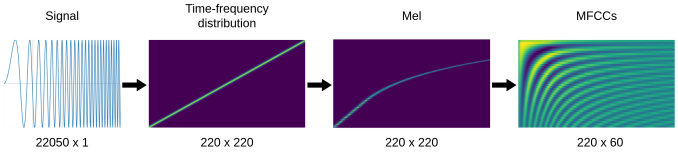


Fig. 1: Depiction of the signal processing applied to obtain the mel-frequency cepstral coefficients (MFCC). The audio waveforms were first transformed using a time-frequency distribution and then by applying first a mel-transform and then a logarithm followed by a discrete cosine transform yielding the mel-frequency cepstral coefficients.

where the kernel $\phi(\theta,\tau)$ determines the precise TFD. Setting e.g. $\phi(\theta,\tau) = 1$ yields the Wigner-Ville distribution. Kernels for several TFDs can be found in [9].

The Stockwell- or S-transform $S_T(t,f)$ is not a representant of Cohen's class and is defined [19] as

$$S_T(t,f) = \int s(\tau) \frac{|f|}{\sqrt{2\pi}} e^{\frac{-(\tau-t)^2 f^2}{2}} e^{-j2\pi f\tau} d\tau. \tag{5}$$

It is related to the wavelet transform and can also be shown to be a special case of the short-time fourier transform.

These TFDs were implemented using several commonly available toolboxes for Python and Matlab, e.g. [20]. The filtered Wigner-Ville distribution, implemented in [20], is obtained by multiplying the Wigner-Ville distribution with the STFT, thereby removing some of the cross-terms.

A. Application of the Time-Frequency Distributions

The general signal flow from the input waveform to the Mel-Frequency Cepstrum Coefficients (MFCC) is depicted in Figure 1. The audio waveforms were first transformed into the time-frequency domain by the respective TFD. Then, by applying the mel–transform and librosa's MFCC algorithm, 60 MFCC per frame were obtained. These MFCCs were then used as the input to the convolutional neural networks (CNNs). Except for the spectrogram, all TFDs were scaled down from a dimension of $22,050 \times 22,050$ to a dimension of 220×220 to reduce the computational load and required ressources to a managable level.

III. Audio Classification Model

We investigated three different classification approaches: i) A baseline approach, where individual classifiers were trained separately using only one of the TFDs. This served as reference for the two mixing approaches.

ii) A so called "post-mix", where the baseline classifiers were combined or mixed by a subsequent neural network to yield a final classification. A sufficiently large neural network should never achieve a lower accuracy than the best baseline classifier.

iii) A so called "pre-mix", where a CNN was provided a mix of the TFDs at once. The MFCCs derived from the respective TFDs each were filtered by separate, independent convolutional layers. The output of these convolutional layers

were then combined in subsequent dense layers such that the CNN should be able to learn optimal convolution kernels.

The main difference of the pre-mix to the post-mix was that, during the training the CNN classifier should be able to learn optimal feature extraction convolutional kernels for one TFD with respect to all other TFDs. Two approaches for the pre-mix were tested: random initialization and baseline initialization. In the baseline initialization the post-training weights of the respective baseline CNNs were used to initialize the convolutional layers. Because the baseline initizalization performed consistently better, improving the accuracy by about 0.5%, only the baseline initialization is presented. All three approaches are depicted in Fig. 2. The structure of the pre-mix CNN is shown in Fig. 3. The number of convolution kernels and convolution kernel sizes were varied to guarantee optimal settings to avoid suboptimal hyperparameters. Up to four convolutional layers were tested and the number of filters per layer was swept between eight to 64 using powers of two. The filter dimension was varied between three and seven.

Because of the large number of TFDs, hyperparameter settings and classification approaches, we conciously selected a simple CNN structure with around 300k parameters to allow for faster training.

A. Training

The CNNs were trained using Monte Carlo cross validation using the adam optimizer with a learning rate of 0.001, 30 epochs, a batchsize of 128 and the categorical crossentropy loss. A 80% / 20% training/validation split of the entire dataset was used. It was confirmed by samply survey of the loss curves that the CNNs had converged after 30 epochs.

For the Monte Carlo cross validation, five repetitions for the baseline and ten for the post- and pre-mix were used.

B. Dataset

The Google Speech Command Dataset V1 [21] was used in all experiments. It consists of 30 single word commands like "yes", "on" or "stop", each with a duration of exactly one second and a total of 65,000 samples. While most other work uses the V2 version [21], due to the large number of TFDs evaluated in this work, a thorough investigation was considered too time consuming on the considerably larger V2 version.

IV. Results

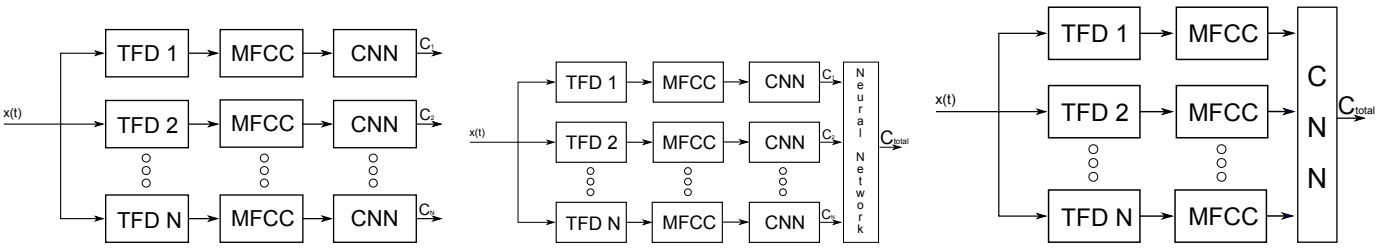
All reported results were achieved on the validation data. The average accuracies across five repetitions of the baseline classifiers are given in Table I. Shown are only the results for three convolutional layers as CNNs with two and four layers consistently achieved lower accuracies. The best configuration is shown in bold font. The spectrograms (SG) were computed with a hann-window, $50\,\%$ overlap and window sizes, denoted as indices, between 256 and 4096, where a window size of 1024 proved to be optimal. The $95\,\%$ confidence interval of the SG_{1024} was $93.1\% \pm 0.29\,\%$. Except for SG_{256}, the spectrogram always outperformed all other TFDs of the baseline approach. The $95\,\%$ confidence intervals of the accuracies across ten repetitions of selected pre-mix and post-mix classifiers are given in Table II. The post-mix consistently achieved higher accuracies by about $0.2\,\% - 1.5\,\%$ than the pre-mix with very few exceptions.

Mixing spectrograms with window sizes of 512, 1024 and 2048 yielded a post-mix mean accuracy of $95.1\,\%$. Adding all other TFDs except for the Pseudo Wigner-Distribution allowed to achieve the highest accuracy of all combinations with a post-mix mean accuracy of $95.56\,\%$. Several combinations achieved very similar mean accuracies, both in the pre-mix and the post-mix. A Wilcoxon sign-rank test found no significant difference ($p > 0.05$) between the top five post-mixes as given in Table II, however, a significant difference ($p < 0.02$) was found between e.g. post-mixing the three spectrograms and the best post-mix listed in Table II. The pre-mix achieved its highest mean accuracy of $94.46\,\%$ (not shown in Tab II) when all TFDs except for the filtered Wigner-Ville distribution (n = 512) and the Pseudo Wigner-Ville distribution were used. However, the difference in accuracy of the pre-mix when using the best TFD mix of the post-mix was found to be not significant ($p > 0.05$). While for the best performing post- and pre-mixes the accuracy increase due to mixing the TFDs was always below $2.5\,\%$, for the Wigner-Ville distribution and the Pseudo Wigner-Ville distribution a considerably increase by about $6\,\% - 8\,\%$ was observed, both in the post- and pre-mix. This seemed to be a general trend, where TFDs with poorly performing baseline classifiers benefited the most from mixing the TFDs.

V. Discussion

This work investigated the benefits of mixing several time-frequency distributions for audio classification on the Google Speech Command Dataset. The aim was to investigate the benefits of improved signal representation for audio classification instead of more sophisticated neural network structures. The aim was not to achieve the absolute best accuracy, but rather to investigate by how much the accuracy achieved of a given classifier can be raised using other TFDs than the spectrogram, or by mixing of TFDs.

As expected, mixing several TFDs proved to be beneficial and superior to any single TFD, with a maximum improvement of $2.85\,\%$ of the post-mix compared to the best baseline when using all TFDs except for the Pseudo Wigner-Ville distribution. However, this mix is not special in the sense that several TFD mixes achieved accuracies without significant difference. The pre-mix achieved its maximum accuracy with a slightly different TFD mix, however the difference to the best post-mix TFDs is not significant. Therefore one can conclude that generally the pre-mix and post-mix achieve their best performance with the same mixes of TFDs. Mixing three spectrograms with different window sizes yielded a considerable improvement, both in the pre-mix and the post-mix. The reason has to be the higher and lower frequency resolution and lower and higher time resolution. Different window functions could potentially improve the spectrograms



(a) Baseline (b) Post-mix (c) Pre-mix

Fig. 2: Sketch of the three tested approaches to implementing the different time-frequency distributions (TFDs). In (a) separate audio classifiers are trained independently using only one of the TFDs yielding independent classifications C_i. This approach was supposed to investigate the individual performance serving as baseline. (b) is a straight-forward stacked classifier approach, where the individual audio classifiers are individually trained, and in a second step another neural network is trained to find an optimal mix of the individual classifications C_i, resulting in the final classification C_{total}. In approach (c), the TFDs are fed to the same audio classifier allowing to learn an optimal feature extraction based on all of the TFDs.

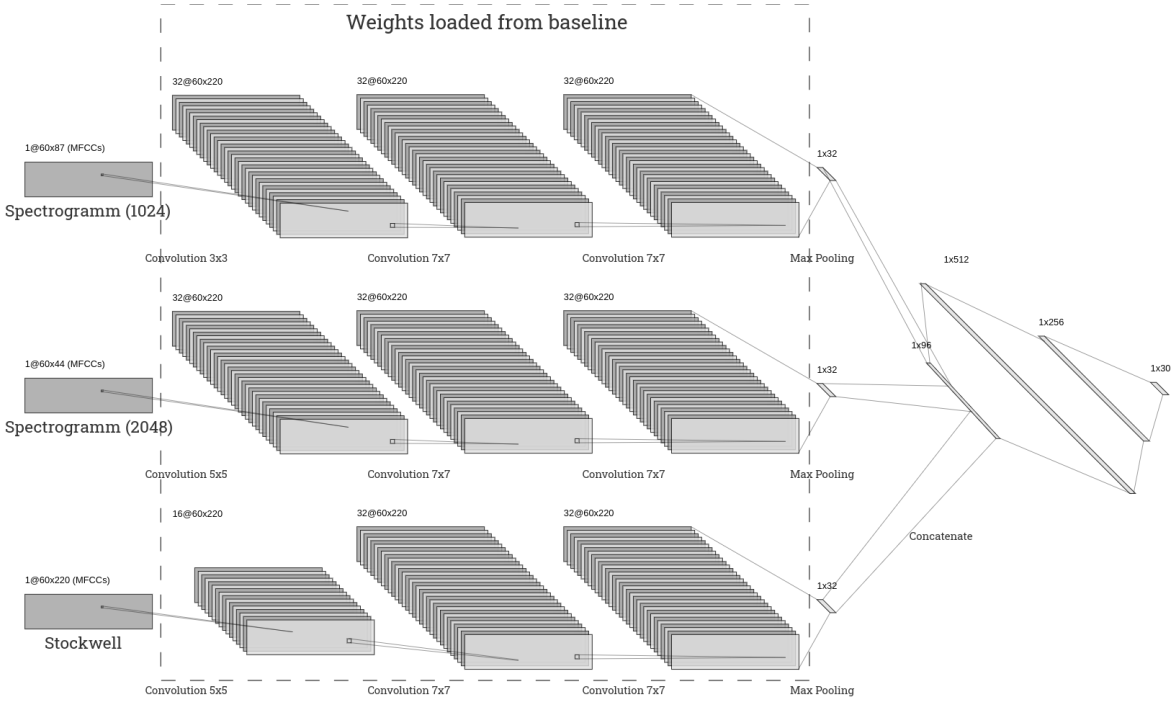


Fig. 3: Example structure of the convolutional neural network (CNN) as used in the pre-mix. The number of filter kernels etc. in the figure are exemplary only. The optimal choice depended on the respective time-frequency distribution. The pre-mix CNNs consisted of separate convolutional layers for the mel-frequency cepstrum coefficients (MFCC) derived from the respective time-frequency distributions. It proved to be beneficial to use the weights obtained from the baseline models for initializing the weights of the convolutional layers. The features extracted by the convolutional layers were then concatenated and fully connected layers yielded the classification of the overall pre-mix model.

TABLE I: Selection of baseline mean accuracies of all investigated time-frequency distributions as achieved using convolutional neural networks (CNN) with the specified convolution kernels. While two and four convolutional layers were also tested, the best performance was achieved using three layers only. Shown are the results for the spectrogram (SG) where the indices denote the window size, the filtered Wigner-Ville Distribution (WVDF), the Wigner-Ville Distribution (WVD), the Pseudo Wigner-Ville Distribution (PWVD) and the Stockwell-Transform.

Convolutional Kernels ([number, dimension])/ Time-Frequency Distribution	SG_{256}	SG_{512}	SG_{1024}	SG_{2048}	SG_{4096}	Stockwell	$WVDF_{256}$	$WVDF_{512}$	$WVDF_{1024}$	$WVDF_{2048}$	$WVDF_{4096}$	WVD	PWVD
[16, 3], [32, 7], [32, 7]	89.14	91.92	92.99	92.79	91.85	85.28	90.28	90.74	89.38	88.07	85.02	75.61	73.22
[16, 5], [32, 7], [32, 7]	89.53	92.12	93.03	92.83	**91.99**	**86.57**	90.95	91.18	89.84	88.85	85.77	**76.42**	74.7
[32, 3], [32, 7], [32, 7]	**88.73**	91.91	**93.1**	92.97	**91.99**	84.63	90.04	90.72	89.88	88.85	85.47	75.38	73.89
[32, 5], [32, 7], [32, 7]	89.48	**92.17**	92.97	**93.05**	91.97	86.43	**90.89**	90.8	**90.13**	88.43	85.36	**76.42**	74.93

TABLE II: A selection of accuracies of the mixed time-frequency distributions. Both, pre- and post-mix accuracies are given. Specified are the 95 % confidence intervals (assuming a gaussian distribution) of the accuracies across ten repetitions.

Time-Frequency Distribution	Mix						
Spectrogram (n = 1024)	x	x	x	x	x		x
Spectrogram (n = 2048)	x	x	x	x	x	x	x
Spectrogram (n = 512)	x		x	x		x	x
Stockwell	x	x	x	x	x	x	x
Wigner-Ville Distribution	x	x				x	x
Wigner-Ville Filtered (n = 1024)	x	x	x		x	x	x
Wigner-Ville Filtered (n = 256)	x	x	x	x	x	x	x
Wigner-Ville Filtered (n = 512)	x	x	x	x	x	x	x
Pseudo Wigner-Ville							x
Accuracy (Pre-Mix) [%]	94.4 ± 0.36	94.21 ± 0.27	94.29 ± 0.19	94.15 ± 0.25	93.96 ± 0.36	94.15 ± 0.26	94.29 ± 0.15
Accuracy (Post-Mix) [%]	95.56 ± 0.04	95.55 ± 0.04	95.55 ± 0.07	95.52 ± 0.05	95.51 ± 0.05	95.25 ± 0.06	95.48 ± 0.04

Time-Frequency Distribution	Mix						
Spectrogram (n = 1024)	x	x		x			
Spectrogram (n = 2048)	x		x	x			
Spectrogram (n = 512)	x	x	x				
Stockwell					x	x	
Wigner-Ville Distribution						x	x
Wigner-Ville Filtered (n = 1024)							
Wigner-Ville Filtered (n = 256)						x	
Wigner-Ville Filtered (n = 512)							
Pseudo Wigner-Ville					x	x	x
Accuracy (Pre-Mix) [%]	94.18 ± 0.21	93.8 ± 0.24	94.13 ± 0.24	94.22 ± 0.13	89.39 ± 0.2	92.33 ± 0.33	82.18 ± 0.23
Accuracy (Post-Mix) [%]	95.1 ± 0.07	94.28 ± 0.04	94.37 ± 0.05	94.57 ± 0.04	89.1 ± 0.05	93.18 ± 0.07	82.19 ± 0.07

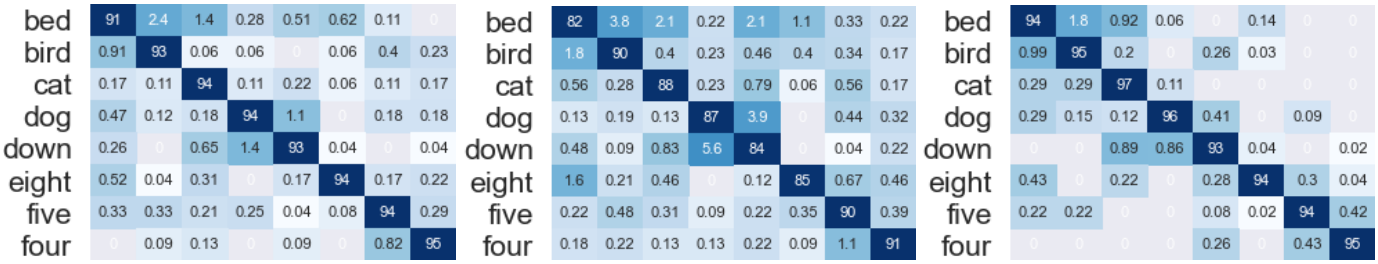
performance further. For practical applications it could be viable to mix the several spectrograms as they are inexpensive to compute. However, the further increase in accuracy by adding further TFDs like the Wigner-Ville distribution does not appear to justify the computational load coming with them. Interestingly, the highest accuracy was not achieved using the pre-mix but rather the post-mix classifiers. Although several different configurations and hyperparameter settings were tested, most likely the cause were suboptimal training or hyperparameter settings, as the pre-mix should be able to at least achieve the same accuracy as the post-mix. More research is required to understand the exact cause of this observation. The input images obtained from the TFDs had to be scaled down as explained in Section II-A. This in general should lead to a loss of information and might partially explain the poor performance of some of the TFDs like the filtered Wigner-Ville distribution, which in preliminary tests appeared to somewhat outperform the spectrogram.

A. Comparison to other Work

TFD-mixing has been mostly used in environmental sound classification and has not been tested on the Google Speech Command Dataset. Chi et al. [18] concatenated spectrograms in a CNN audio classifier similiar to our pre-mix classifier and observed an improvement of about $2.3\,\% - 2.8\,\%$ to the best individual TFD. This is a somewhat larger benefit than observed by us for the pre-mix, however, Chi et al. used a considerably larger CNN with a few million parameters, compared to about 300k as in our work. Sharma et al. [15] combined MFCC with Gammatone Frequency Cepstral Coefficients and other features and observed an increase of about $12\,\%$ in accuracy when combining all TFDs investigated by them. This is a significantly larger improvement than what was observed in our work and was observed on two separate datasets. While they used a considerably larger CNN than the one used in this work, and different TFDs, such a large benefit is still surprising. An explanation could be that their results were obtained on enviromental sound datasets. There, the audio to be classified certainly will be more diverse than that of the Google Speech Command Dataset and adding TFDs could indeed allow a CNN novel insight into the audio. Su et al. [22], whose approach is the closest to our own, observed an improvement in accuracy of about $2\,\%$ through mixing of TFDs, close to the results of this work, in the manner of the presented post-mix with respect to a similar approach as our pre-mix. However, they do not present baseline results so that the benefit of TFD-mixing did not become apparent.

B. Cause of Improvement

Excerpts of the confusion matrices of two baseline classifiers based on the spectrogram (n = 1024), the Stockwell distribution as well as their post-mix are depicted in Fig. 4. The baseline using the spectrogram consistently achieved a greater accuracy than the corresponding Stockwell baseline with very few exceptions. Despite this, the combined accuracy, represented by the post-mix classifier, improved the accuracy by about 1%. The cause most likely is both baselines returning a different, incorrect class as the most likely prediction, however, both also returning the correct class with lower, e.g. second best, probability. In these cases, deciding for the second best class appears to be reasonable and could be implemented by the neural network that mixes the baseline predictions in the post-mix.



(a) Baseline spectrogram (n = 1024) (b) Baseline Stockwell (c) Post-mix

Fig. 4: Excerpts of dimension 8 x 8 of the 30 x 30 confusion matrices of the respective baseline classifiers using the (a) spectrogram (n = 1024) and (b) Stockwell and the (c) respective post-mix classifier of the two distributions. Shown are the first eight classes (according to alphabetic order) and the achieved accuracy in percent. While the baseline classifier based on the Stockwell distribution achieved almost consistently worse accuracy, their post-mix is considerably better.

VI. Conclusion

This work compared time-frequency distributions (TFDs) for audio classification using convolutional neural networks (CNN). Using the Google Speech Command Dataset V1, several TFDs including the Wigner-Ville distribution and its derivates as well as the Stockwell-transform were evaluated. Hereby, mixing of the TFDs was implemented and synergetic effects investigated. The TFDs were applied invidivually as well as in pre- and post-mix implementations.

It was found that a mixture of eight TFDs achieved the best mean accuracy $95.56\,\%$ which represented a considerable improvement of 2.46 % over the best, single TFD, that was found to be the spectrogram with a window size of 2048 and that achieved a mean accuracy of $93.1\,\%$.

Our results show that CNN-based audio classification can be improved using mixing of time-frequency representation, allowing the CNN to learn better feature extraction, thus improving classification accuracy.

References

[1] Nguyen Anh, Yongjian Hu, Qianhua He, Tran Linh, Pham Viet, and Chen Guang. Lis-net: An end-to-end light interior search network for speech command recognition. *Computer Speech Language*, 65:101131, 07 2020.

[2] Dong Wang, Xiaodong Wang, and Shaohe Lv. An overview of end-to-end automatic speech recognition. *Symmetry*, 11(8), 2019.

[3] Dong Yu and Li Deng. *Automatic Speech Recognition: A Deep Learning Approach*. Springer Publishing Company, Incorporated, 2014.

[4] Chanwoo Kim, Dhananjaya Gowda, Dongsoo Lee, Jiyeon Kim, Ankur Kumar, Sungsoo Kim, Abhinav Garg, and Changwoo Han. A review of on-device fully neural end-to-end automatic speech recognition algorithms. *arXiv*, abs/2012.07974, 2020.

[5] Douglas Andrade, Sabato Leo, Martin Viana, and Christoph Bernkopf. A neural attention model for speech command recognition. *arXiv*, abs/1808.08929, 08 2018.

[6] Xuejiao Li and Zixuan Zixuan. Speech command recognition with convolutional neural network. 2017. Accessed: 2021-05-26.

[7] Somshubra Majumdar and Boris Ginsburg. Matchboxnet: 1d time-channel separable convolutional neural network architecture for speech commands recognition. *arXiv*, abs/2004.08531, 2020.

[8] Gert Hoey, Wilfried Philips, and Ignace Lemahieu. Time-frequency analysis of eeg signals. 01 1998.

[9] Leon Cohen. *Time-Frequency Analysis: Theory and Applications*. Prentice-Hall, Inc., USA, 1995.

[10] F. Auger and P. Flandrin. Improving the readability of time-frequency and time-scale representations by the reassignment method. *IEEE Transactions on Signal Processing*, 43(5):1068–1089, 1995.

[11] Boualem Boashash. Time-frequency signal analysis: Methods and applications, special conference on time-frequency analysis, methods and applications. Melbourne, 1992. Longman Cheshire. Special Conference on Time-Frequency Analysis.

[12] M. E. Gorbunov, K. B. Lauritsen, and S. S. Leroy. Application of wigner distribution function for analysis of radio occultations. *Radio Science*, 45(06):1–11, 2010.

[13] J. Brynolfsson and M. Sandsten. Classification of one-dimensional non-stationary signals using the wigner-ville distribution in convolutional neural networks. In *2017 25th European Signal Processing Conference (EUSIPCO)*, pages 326–330, 2017.

[14] Shasha Mo and Jianwei Niu. A novel method based on ompgw method for feature extraction in automatic music mood classification. *IEEE Transactions on Affective Computing*, 10(3):313–324, 2019.

[15] Jivitesh Sharma, Ole-Christoffer Granmo, and M. Goodwin. Environment sound classification using multiple feature channels and attention based deep convolutional neural network. In *INTERSPEECH*, 2020.

[16] Tomás Arias-Vergara, P. Klumpp, Juan Vasquez, Elmar Noeth, Juan Rafael Orozco, and Maria Schuster. Multi-channel spectrograms for speech processing applications using deep learning methods. *Pattern Analysis and Applications*, pages 1–9, 09 2020.

[17] Z. Zhang, Shugong Xu, S. Cao, and Shunqing Zhang. Deep convolutional neural network with mixup for environmental sound classification. *ArXiv*, abs/1808.08405, 2018.

[18] Z. Chi, Y. Li, and C. Chen. Deep convolutional neural network combined with concatenated spectrogram for environmental sound classification. In *2019 IEEE 7th International Conference on Computer Science and Network Technology (ICCSNT)*, pages 251–254, 2019.

[19] Boualem Boashash. *Time-Frequency Signal Analysis and Processing: A Comprehensive Reference*. Academic Press, 2nd edition, 2016.

[20] Frank Zalkow. Python time-frequency toolbox. https://www.frank-zalkow.de/en/the-wigner-ville-distribution-with-python.html. Accessed: 2021-05-18.

[21] Pete Warden. Speech commands: A dataset for limited-vocabulary speech recognition. *arXiv*, abs/1804.03209, 04 2018.

[22] Yu Su, Ke Zhang, Jingyu Wang, and Kurosh Madani. Environment sound classification using a two-stream cnn based on decision-level fusion. *Sensors*, 19:1733, 04 2019.