Physically-Augmented Deep Learning (PADL): Integration of Physical Context for Improved Seismic Event Discrimination

Mitchell L. Solomon[1], Micah Billouin[1], Jad Zeineddine[2], Kevin Chow[2], Alex Almaraz[2], Anand Rangarajan[2], Anthony O. Smith[1], Adrian M. Peter[1]

[1]Center for Advanced Data Analytics and Systems (CADAS), Florida Institute of Technology
[2]University of Florida

Key Points:

- Inclusion of the velocity model as a prior in multiview fusion model improves classification accuracy over models trained on signals alone.
- Multiview deep feature fusion of velocity models and explosion waveforms is robust to weakly discriminatory inputs from one view.
- The transferability of learned features between synthetics and real waveforms is low, likely due to the simplicity of the physical model.

Abstract

Deep learning has established itself as the go-to methodology for automated event discrimination. However, these methodologies still provide suboptimal performance in many classification tasks, including the seismic source categorization problem considered in this article. Here, we develop a novel deep learning framework that allows for the direct integration of environmental context, which we refer to as Physically-Augmented Deep Learning (PADL). Specifically, we augment the learning process by incorporating seismic velocity models generated from a physics-based simulator. Our experiments couple real observational waveform data and synthetic velocity models from the Tularosa Basin region and demonstrate near-perfect classification accuracy when employing PADL. A robust set of ablation studies on joint and independent convolutional neural networks and various combinations of real and simulated input data confirm the efficacy of our PADL framework.

1 Introduction

With greater research emphasis being placed on automation and low-yield events (Maceira et al., 2017), robust solutions to signal classification become increasingly difficult to realize. Recent advances in the fields of machine learning and especially deep learning have prompted their adaptation to identify signals of interest (SOI) by automatic extraction of features from the abundance of seismic data gathered over a long history of recorded events.

To meet the challenges posed by seismic event discrimination, we adopted deep learning frameworks for the development of our classification engines. Deep learning networks are biologically-inspired, computational reasoners that have demonstrated unprecedented successes on a variety of decision tasks, including recognition of objects in images and video, language translation, self-driving automation, and competitive gaming. Their ability to automatically learn rich feature representations, often with low fidelity data, and identify distinguishing attributes make them ideally suited for the tactical environments where similar data quality issues exist.

Furthermore, human decision-making is rarely supported from a single source of information. Decisions made based on one source can lead to high degrees of Type 1 and Type 2 errors, which is a growing concern for power-based signal classifiers as the yield of source events decreases (Maceira et al., 2017). A way to overcome erroneous detections is to corroborate the detections of many classifiers; such is the basis of array-level processing.

But what if we are limited to station- level processing where no corroboration between stations is allowed? Then additional sources of information are restricted to those that are co-located and co-collected. In this situation, which is the target performance environment for next-generation signal detectors and classifiers (Maceira et al., 2017), we are compelled to utilize all streams of data of the source event to maximize confidence in our predictions. In conjunction with the rise of deep learning, this leads us to consider Multiview Deep Learning (MDL) (Sun et al., 2019) techniques to capture features of multimodal data in this effort.

A strength of deep neural networks lies in their modularity. Just like putting together the pieces of a puzzle, we can harness multiple data types of the same source event, called *views*, to extract richer feature descriptors of the event. We adopt techniques from a major category of MDL, focused on joint representation learning, where, in our application, hierarchical seismic x_S time-series features are fused with environmental x_E features to form the joint representation x.

$$x = f(x_S, x_E) \tag{1}$$

Within the structure of a deep neural network, any data view can be represented modularly by their own branch of sequential layers $x_{S,E} = f_{S,E}(x_{S,E})$. The merging of single-view representations to form the joint representation is called representation fusion, an operation in which the flexibility and modularity of neural networks has enjoyed a broad range of applicability, for example, in computer vision (Ouyang et al., 2014; Jiang et al., 2018), high-performance computing (Wang et al., 2015), and sentiment analysis (Nojavanasghari et al., 2016). So far as we are aware, no effort has been made to utilize these techniques on seismic data with environmental context simultaneously. Therefore, we aim to bring state-of-the-art MDL models to time-series classification with environmental context to form our Physically-Augmented Deep Learning (PADL) framework in an attempt to attain highly accurate and robust models for classification of explosive sources.

2 Related Work

There has been a myriad of methodologies investigated for event discrimination based on seismic waveform characteristics Seismic event discrimination has been explored for events such as earthquakes, explosions, and some tactical signatures like vehicles. However, to the best of our knowledge, there is a dearth of research, in the open literature, that attempts to

utilize our novel approach of integrating physical context with deep learning in a multiview fusion architecture.

3 Seismic Velocity Models as Physical Context Priors for Joint Representation Learning

Resultant seismograms are a function of the geological media through which they propagate. Multiview representations of such geological media are, by and large, left out of the training process when performing seismic signal classification. Automatic joint representation learning paradigms offer an opportunity to utilize seismic velocity models a priori where joint features of the time series and known seismic profiles are conjointly processed to produce a signal classification. In this section, we describe the creation of a synthetics dataset and our efforts to realize the ability of PADL models to jointly represent these synthetics with the velocity models used to create them. Additionally, we report on the efficacy of pretraining on this relatively large synthetic dataset in a transfer learning scheme for PADL models.

3.1 Imposed-Variance Synthetic Explosion Dataset

We embark on an effort to generate a dataset of synthetic seismograms that reflect the known conditions of the Humming exercises within the Tularosa Basin. Waveforms for the Imposed-Variance Synthetics Explosion Dataset (IVSED) are generated using an open source software for seismic simulation called Computer Programs in Seismology (CPS) (Herrmann, 2013) from St. Louis University. The specific CPS routine used takes in a seismic velocity model and station layout and then outputs a set of synthetic waveforms as depicted in Figure 1. Depiction of the modeling environment utilized with Computer Programs in Seismology (CPS) software to generate synthetic seismograms, Figure 1. The seismic model we use is one-dimensional, isotropic, flat-layered, and constant-velocity earth, meaning that sequential flat layers (denoted l_i) of homogenous physical parameters with one-dimensional boundaries conditions define the model. Explosion origins are defined at the earths surface. We choose a constant station layout for synthetic generation runs consisting of 19 stations logarithmically spaced between 0.01km and 600km from the explosion origin. Each station records 2.56 seconds around the initial phase arrival with seven channels sampled at 100Hz, so there are 133 waveforms per set, each 256 points in length.

Each layer of the seismic velocity models consists of ten parameters describing physical properties of the earths interaction with pressure waves. We chose six of those ten to alter in the creation of IVSED:

- **H** - layer thickness (km)
- $\mathbf{V_P}$ - compressional wave velocity (km/s)
- $\mathbf{V_S}$ - shear wave velocity (km/s)
- ρ - rock density (gm/cm^3)
- $\mathbf{Q_P}$ - compressional wave quality factor
- $\mathbf{Q_S}$ - shear wave quality factor

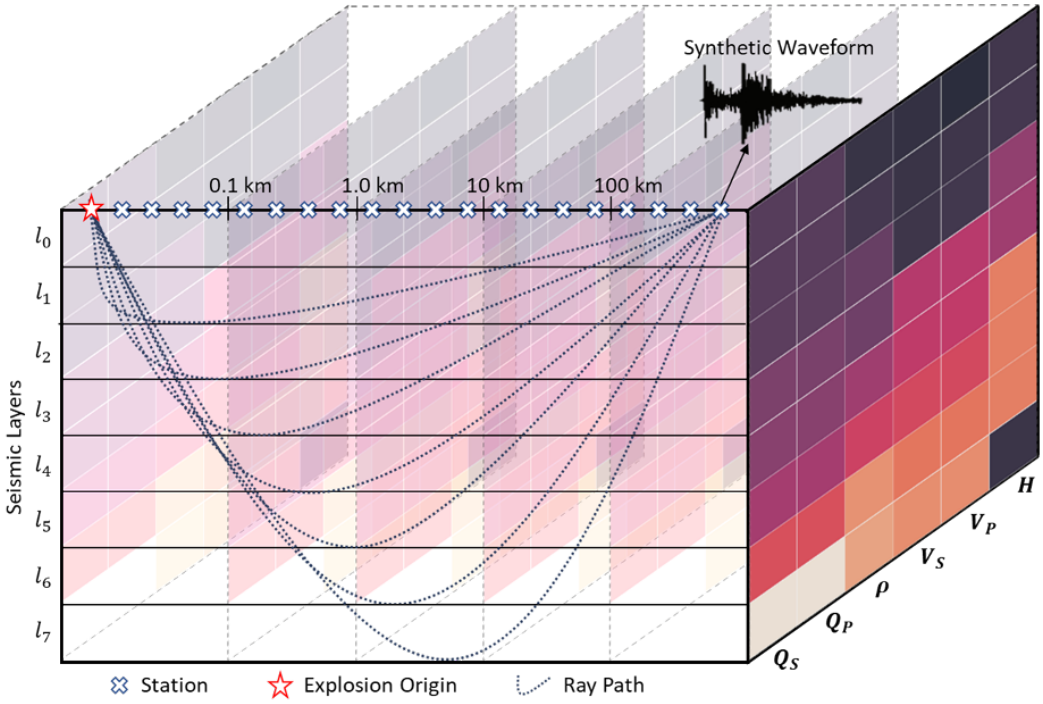


Figure 1. Depiction of the modeling environment utilized with Computer Programs in Seismology (CPS) software to generate synthetic seismograms. Seismograms are collected at each of the stations.

Based on reported details of the surface rock material in the Humming exercises as well as seismological differences of each explosion origin within the Tularosa Basin, we choose four velocity models as the basis environments for all synthetics. As a result, these four baseline velocity models represent three classes of explosions (Figure S5) with the

ANFO explosion class comprising two of the four baseline models. The most prominent difference between these four baseline velocity models is the top-layer rock material, one of rhyolite, granite, quartz-latite, or alluvium. These four rock materials were specified in their respective Humming exercise data reports. The parameters of subsequent layers are derived from seismological studies of the Tularosa Basin (Reinke & Herrin, 1977; Adams & Keller, 1994; Sinno et al., 1986; Bonner et al., 2013) where each of the parameters are identical from the fourth layer onwards. We exploit the uniformity of the last four seismic layers in an experiment in Section 4 to highlight the contribution of the difference that makes a difference for inputs to a joint classifier.

Because our primary goal is to understand the utility of jointly representing explosion waveforms and velocity model features, sufficient variation for both inputs per class is necessary within the classification pipeline. Therefore, we impose variance on the parameters of each of the four baseline velocity models by drawing from a Gaussian centered on the value of the parameter. The variance of each Gaussian is chosen as a fraction of the parameter value upon which the Gaussian is centered. We generate synthetics from velocity models drawn in this way at eight levels of imposed variance: 0.001, 0.01, 0.05, 0.1, 0.2, 0.3, 0.4, and 0.5. Thus, the six parameters from the velocity models depicted in Figure S5 at each layer are varied by as little as 0.1% up to 50% of their baseline value. Velocity model parameters for each level of imposed variance were drawn twice, thus constituting 16 different perturbations of the baseline velocity model per shot. Four shots for each class were chosen to further increase the population of velocity models drawn. Thus, the IVSED theoretically consists of 25,536 waveforms. However, some combinations of parameters yield NaN values for time series or flat waveforms with no signal. We remove these instances from the resultant dataset, leading to a reduced size of 24,192 waveforms. Each synthetic is detrended by a 10th order polynomial function, filtered between one and 30Hz, and then tapered to zero from 1% of each end.

3.2 PADL with Seismic Time Series and Velocity Models

In this section, we apply MDL techniques in four experiments designed to characterize the IVSED and Humming datasets as well as the PADL architecture in terms of classification performance. We prepare a convolution-based PADL fusion architecture (Figure 2) to jointly learn features of the time series and physical context in the form of the seismic velocity models. We define the PADL model as three separate branches: 1) the time-series branch,

dedicated to capturing convolutional features of the input waveforms, 2) the velocity model branch, dedicated to learning independent features from the input velocity models, and 3) the fusion branch, dedicated to learning combinations of features from the independent branches via independent feature fusion.

The time-series branch is comprised of seven convolutional blocks, which are sequential convolution, activation, batch normalization, and max pooling operations, followed by a flatten operation and fully-connected layer before being concatenated to the output of the velocity model branch. To capture multi-scale features in the convolutional layers, we use a decreasing kernel size from seven to three as shown in Figure 2 as well as L-2 regularization of the weights with coefficient 10^{-5}. The first, second, and third convolutional layers have 128, 64, and 32 filters respectively. All convolutions use a stride size of one. The output of each convolutional layer is zero-padded to be the same shape as its input and is followed by a ReLU activation function. Each max pooling layer uses a pool size factor of two.

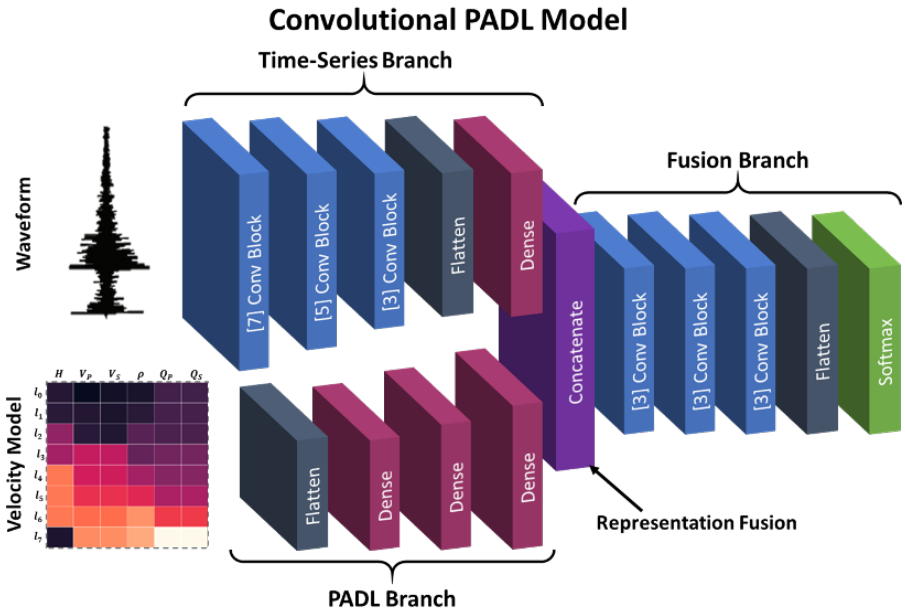


Figure 2. PADL fusion architecture used to test performance on IVSED dataset as well as transfer learning generalizability to the real Humming dataset. We utilize convolutional blocks in the Time-Series branch and simple fully-connected layers in the PADL branch. Concatenation fusion joins the two branches, then fully-connected layers are used in the Fusion Branch to learn joint features.

The PADL branch is comprised of three fully-connected layers with the number of nodes per layer scaled up to match the output dimensionality of the time-series branch as shown in Figure 2 to enable feature-wise concatenation of the independent branch outputs. ReLU activation function is used for these fully-connected layers. The fusion branch is designed to learn combinations of features from the input branches. We design the fusion branch to be three convolutional blocks with 32 filters per layer with ReLU activation functions. We flatten the penultimate layer before normalizing into classification probabilities with a softmax layer. All models in this section are trained using the Adam optimizer (Kingma & Ba, 2014) with the custom learning rate schedule from Section S2, which we have found to be an essential tool for enabling consistent results.

We make use of IVSED in a transfer learning scheme where the larger synthetics dataset is used to pretrain PADL model weights to as a starting point for training on real Humming signals. We further test the effect of velocity models by implementing class-matched and random assignment rules to corresponding signals. One model for each combination of transfer learning stage (pretraining or finetuning) and velocity model assignment method (random or matched) is trained and evaluated on a test set of real signals from the Humming dataset and velocity models from IVSED (Figure S9). In the experiment, weights of the pretrained model are either frozen, which halts the weight updates to the pretrained model, or trainable in the finetuning phase. Results of this experiment are presented in Figure 5.

4 Results

Classification results are presented on the real Humming dataset as well as on the synthetic IVSED dataset in various training schemes to elucidate performance behavior on single view and multiview deep learning models. To set the stage for comparison to multiview PADL models, we first examine single view baselines by implementing and studying the effect of a learning rate scheduler described in Section S2 as well as further characterizing the Humming dataset by visualizing geographically distributed test performance (Section S3). Finally, we present performance benchmarks on IVSED and comparisons that quantify the discriminatory benefit of adding seismic velocity models in a PADL architecture in Section 4.1, as well as a study of synthetics feature transferability to real explosions in Section 4.2.

4.1 Effect of Velocity Model Parameter Variance

Because IVSED features perturbations of the velocity models with increasing variance, we find it prudent to first evaluate the effect of this imposed variance on the performance of an independent convolutional deep learning model for time-series classification. In this experiment, the time-series branch is used alone to learn on the waveforms only in IVSED. We partition IVSED first into groups of imposed variance and then into a training set and a testing set in a ratio of 4:1. From there, the training set is used in a 10-fold cross validation scheme where class-balanced splits are used to compute training and validation accuracies. The examples from each validation split in the cross-validation scheme are not used to train on. After training on each of the k-folds, accuracy on the test set is computed. Each model is trained for 200 epochs, and the accuracies reported in Figure 3 are computed from each model at the end of the 200 epochs. No early stopping criteria were used during training.

The results of the experiment on imposed variance are presented in Figure 3. As imposed variance increases, we observe a slight drop in training accuracy and a large drop in both validation and test accuracy. Because all other factors are held constant during training, we attribute this loss of generalizability to increasing dissimilarity of waveforms drawn from velocity models of increasing imposed variance. In other words, as velocity model distributions overlap, so to do the waveforms that they produce.

To further quantify the effect of adding velocity models to a multiview model, we design another experiment in which we test two different assignment rules to pair velocity model with corresponding real waveforms in the Humming dataset. We train 30 models by randomly assigning velocity models to a waveform of the same class and 30 models by randomly assigning velocity model to a waveform regardless of class. This procedure is repeated on the seismic phase arrivals of the full Humming dataset. Validation accuracies of this experiment are reported in Figure 4.

The difference between matched (green) and random (orange) performance distributions clear and expected. Corroborating time-series features with highly discriminatory velocity model features by matching within a class leads to more accurate and precise generalization. When velocity models are completely randomly matched, their discriminatory power is compromised, and the mean validation accuracy falls to within expectation of our classification benchmarks in earlier experimentation.

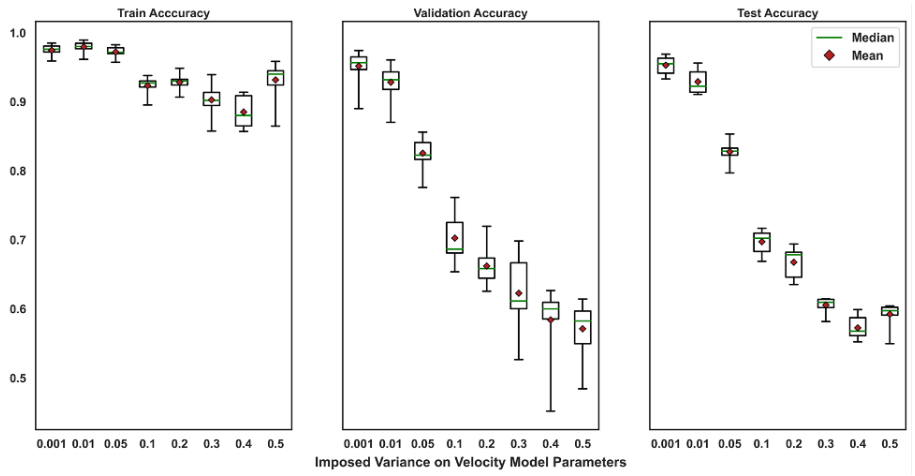


Figure 3. Boxplots of training, validation, and test accuracies of a deep convolutional neural network trained on the IVSED seismograms. Each box represents a 10-fold cross-validation training paradigm where model hyperparameters and architecture are held constant. White circles represent possible outliers. As imposed variance increases, the generalization becomes harder likely due to overlap in class-conditional distributions.

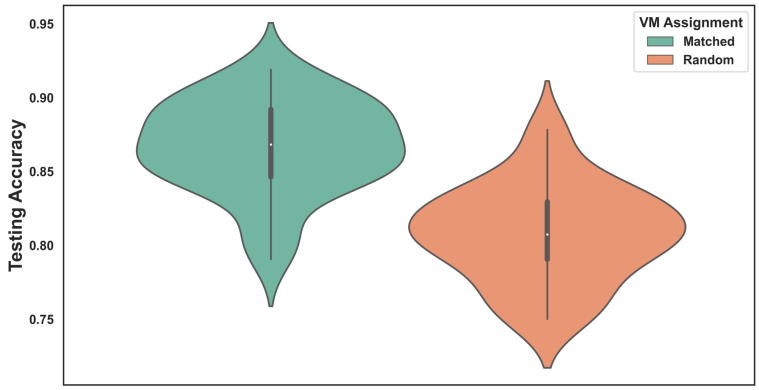


Figure 4. Test accuracy results of the velocity model assignment experiment where we assign velocity models constructed for each class of explosion in IVSED to waveforms from the Humming dataset. Velocity models are either assigned randomly (orange) or matched (green) to the class label of the time series. 30 models were trained for each assignment method to create the test accuracy KDE curves.

4.2 Transferability of Deep Synthetic Features

Two PADL models were pretrained on IVSED one for each velocity model assignment rule (class-matched or random). The expectation here is that assigning velocity models to signals randomly would yield results similar to single view models where the view is the signal alone. We see a large rise in training accuracy (89.42% to 100%) and a slight rise in test accuracy (74.39% to 78.35%), which are above expectation but within reasonable limits according to the results in Figure 4. Both pretrained models achieve perfect training accuracy, but as expected, we see a sharp drop in test accuracy (100% to 78.35%) when velocity model assignment is randomized.

Results indicate that velocity model assignment in the pretraining phase is the main driver of generalizability when pretrained weights are frozen for the finetuning phase. It is noteworthy that the only combination of velocity model matching and weight freedom that does not classify with near perfect accuracy on the training set is Matched-Random and Frozen. Intuition about this result within the context of the other evidence suggests that, when provided signals and matched velocity models, very specific and non-generalizable features are learned, such that when randomly assigned velocity models are provided during finetuning, the new features learned are interacting destructively with previously learned features. We call this behavior negative transfer. In general, negative transfer occurs when features from unrelated sources are used in conjunction to perform a target task. When pretrained weights are frozen, low testing accuracies are observed as the level of randomness increases which further corroborates the presence of significant negative transfer. An alternate conclusion from these observations is that IVSED signals are too unlike the real Humming signals to from which to learn conjointly beneficial features. Therefore, if we wish to utilize synthetics for scarce data situations, more sophisticated physics models are likely necessary in their synthesis.

When pretrained weights are trainable during finetuning (Figure 5, bottom), we observe much better generalization behavior and a trend of decreasing generalizability when more class mismatch is introduced. Humming dataset baselines reveal that even random matching of velocity models can produce good results, and we see a mirroring of that behavior here when synthetic features are transferred. If our hypothesis that significant negative transfer is present, we can conclude that significant unlearning of synthetics features occurs. This would seem to be the case as trainable weight produce better generalization performance.

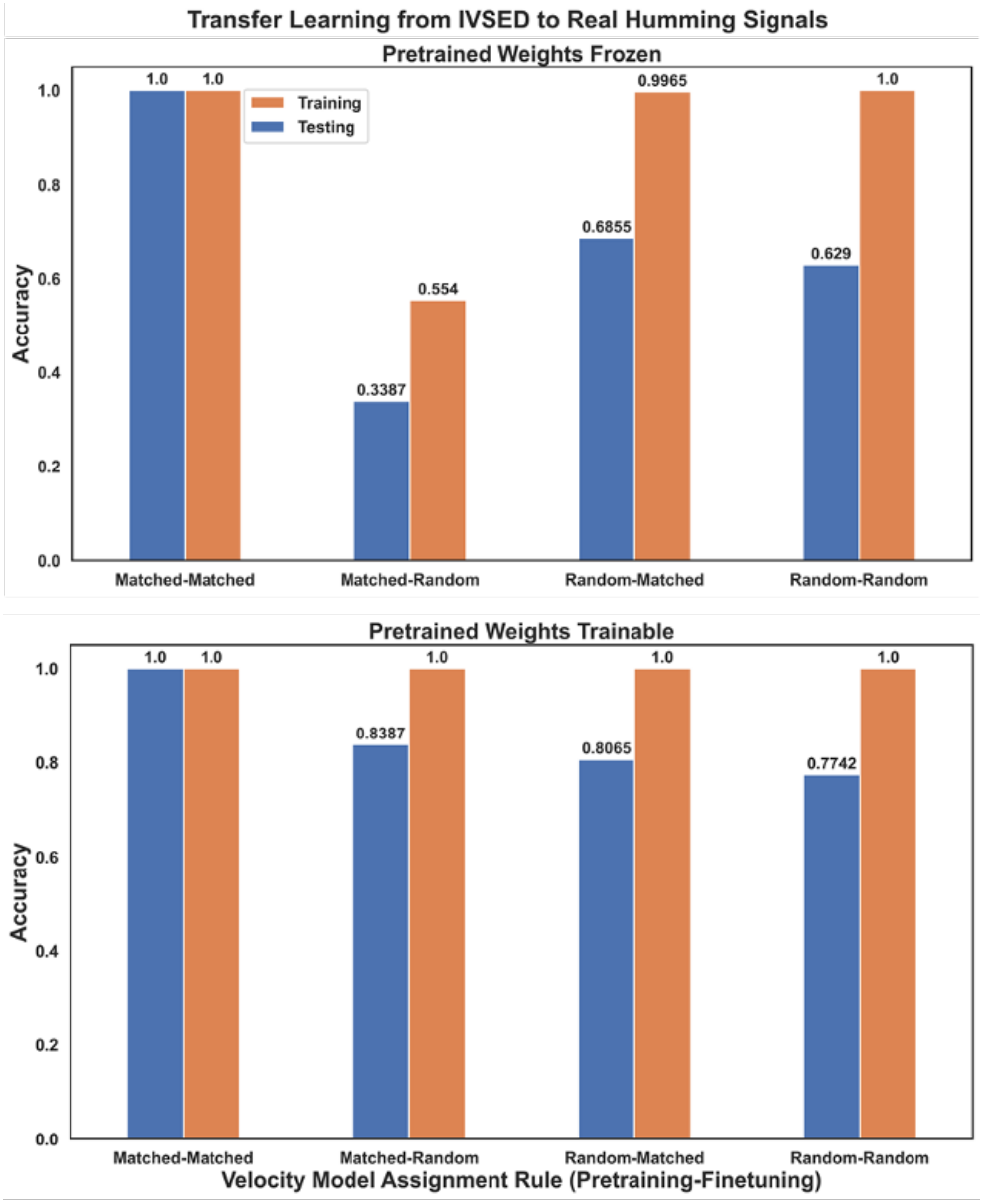


Figure 5. Results of the velocity model assignment experiment in a transfer learning scheme where we attempt to boost classification results on the real Humming dataset by pretraining on IVSED synthetics. The weights of the pretrained PADL model are either frozen (top) or trainable (bottom) during finetuning to test the value of extracting features from synthetics as a seed to learn on real signals. Velocity model pairing rules are shown to greatly affect generalizability while allowing the pretrained weight to be retrained alleviates performance degredation due to the likely presence of negative transfer.

5 Conclusion

Regional explosion classification on both physics-based synthetics and real data was attempted with multiview deep learning models. Because the geographical makeup of the earth directly influences resultant pressure waves captured by seismological monitoring stations, we can view their joint representation as correlating source physics with waveform, an idea that would be highly beneficial to seismic signal classification. Thus we explore physically-relevant auxiliary information in the form of the seismic velocity models used to generated IVSED synthetic waveforms and supply them *a prior* to the multiview deep learning model, which defines the PADL framework.

Indeed, our PADL model pairing seismic signals and velocity models performs well across a variety of experimental conditions. By controlling the variance imposed to the parameters in each layer of the velocity models, we can even say that, under certain conditions, the joint representation of the multiple views induces improved performance over their singular views independently. Because of this result we conclude that the PADL model does not simply overfit to a singular view but utilizes both views even when the discriminatory evidence of one view is particularly weak.

Cross-validated classification results indicates the inclusion of velocity models improves the test accuracy over single view models trained on the waveforms alone. In fact, PADL models trained on velocity models alone score perfectly, leading us to conclude that the discriminatory information within seismic velocity models is dominant, specifically in the first two layers where the greatest difference between origin-based seismic parameters were greatest. However, it is likely that the learned velocity model features would not generalize well to new areas, which is why the inclusion of the waveform as a joint input is important.

We test a transfer learning scheme between IVSED and the real Humming dataset, which returns similar results as synthetics alone since the discriminatory information within the velocity models is so strong. However, there is low transferability between synthetic signal and real signal features likely due to the simplistic physical models used to generate these synthetics. We believe the usefulness of synthetic waveforms here to be underrepresented and would recommend a more rigorous study of transferability between features of real signals and more sophisticated physical models

Data Availability Statement

All waveform and auxiliary data for seismograms and velocity models are available for download through Zenodo `https://doi.org/10.5281/zenodo.5762098`

References

Adams, D. C., & Keller, G. R. (1994). Crustal structure and basin geometry in south-central New Mexico. *Special Paper of the Geological Society of America*, *291*. doi: 10.1130/SPE291-p241

Bonner, J. L., Russell, D. R., & Reinke, R. E. (2013). Modeling surface waves from aboveground and underground explosions in alluvium and limestone. *Bulletin of the Seismological Society of America*, *103*(6). doi: 10.1785/0120130069

Herrmann, R. B. (2013). Computer programs in seismology: An evolving tool for instruction and research. *Seismological Research Letters*, *84*(6). doi: 10.1785/0220110096

Jiang, Y. G., Wu, Z., Wang, J., Xue, X., & Chang, S. F. (2018). Exploiting Feature and Class Relationships in Video Categorization with Regularized Deep Neural Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *40*(2). doi: 10.1109/TPAMI.2017.2670560

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Maceira, M., Blom, P. S., MacCarthy, J. K., Marcillo, O. E., Euler, G. G., Begnaud, M. L., ... Slinkard, M. E. (2017). Trends in Nuclear Explosion Monitoring Research & Development - A Physics Perspective. *DOE Report*(June 2017).

Nojavanasghari, B., Gopinath, D., Koushik, J., Baltrušaitis, T., & Morency, L. P. (2016). Deep multimodal fusion for persuasiveness prediction. In *Icmi 2016 - proceedings of the 18th acm international conference on multimodal interaction.* doi: 10.1145/2993148.2993176

Ouyang, W., Chu, X., & Wang, X. (2014). Multi-source deep learning for human pose estimation. In *Proceedings of the ieee computer society conference on computer vision and pattern recognition.* doi: 10.1109/CVPR.2014.299

Reinke, R. E., & Herrin, E. (1977). *Geophysical model studies of the tularosa basin, new mexico.* (Tech. Rep.). SOUTHERN METHODIST UNIV DALLAS TEX DALLAS GEOPHYSICAL LAB.

Sinno, Y. A., Daggett, P. H., Keller, G. R., Morgan, P., & Harder, S. H. (1986). Crustal

structure of the southern Rio Grande Rift determined from seismic refraction profiling. *Journal of Geophysical Research*, *91*(B6). doi: 10.1029/jb091ib06p06143

Sun, S., Mao, L., Dong, Z., & Wu, L. (2019). *Multiview machine learning*. doi: 10.1007/978-981-13-3029-2

Wang, D., Cui, P., Ou, M., & Zhu, W. (2015). Deep multimodal hashing with orthogonal regularization. In *Ijcai international joint conference on artificial intelligence* (Vol. 2015-Janua).

Figure 1.

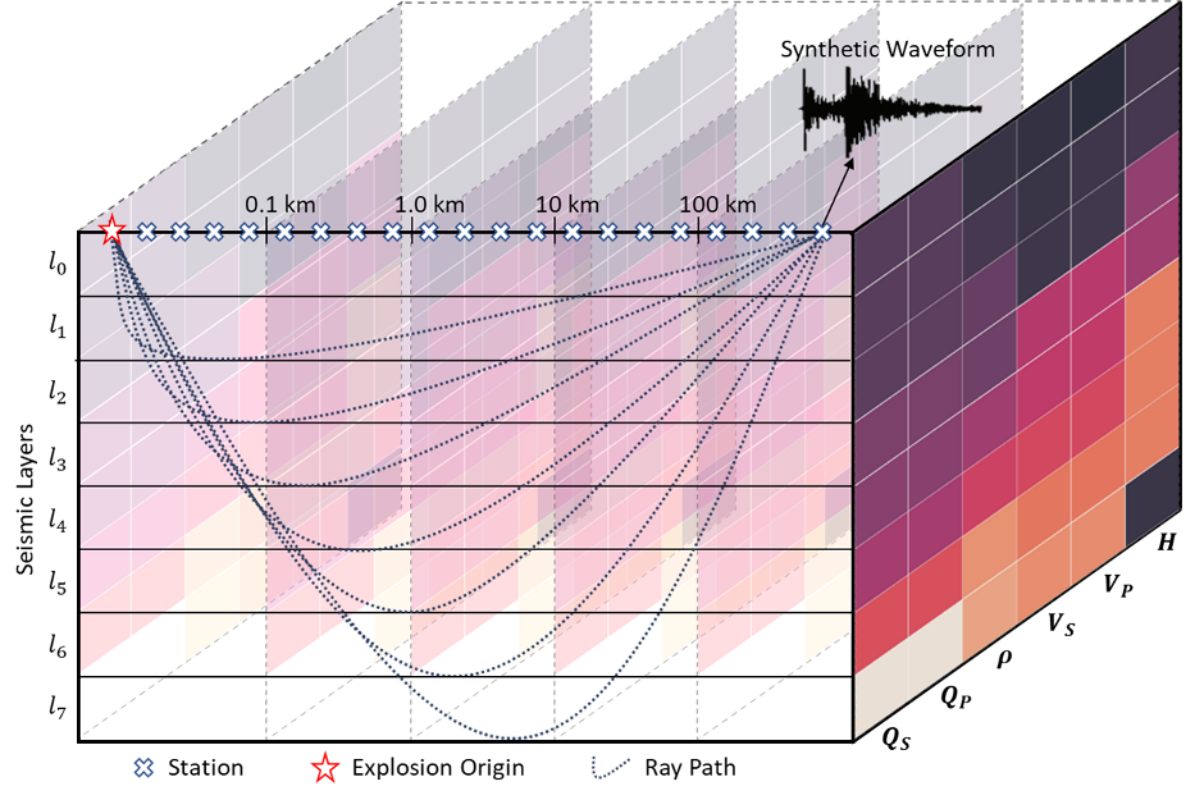

Synthetic Waveform
0.1 km
1.0 km
10 km
100 km
Seismic Layers
l_0
l_1
l_2
l_3
l_4
l_5
l_6
l_7
H
V_P
V_S
ρ
Q_P
Q_S
Station
Explosion Origin
Ray Path

Figure 2.

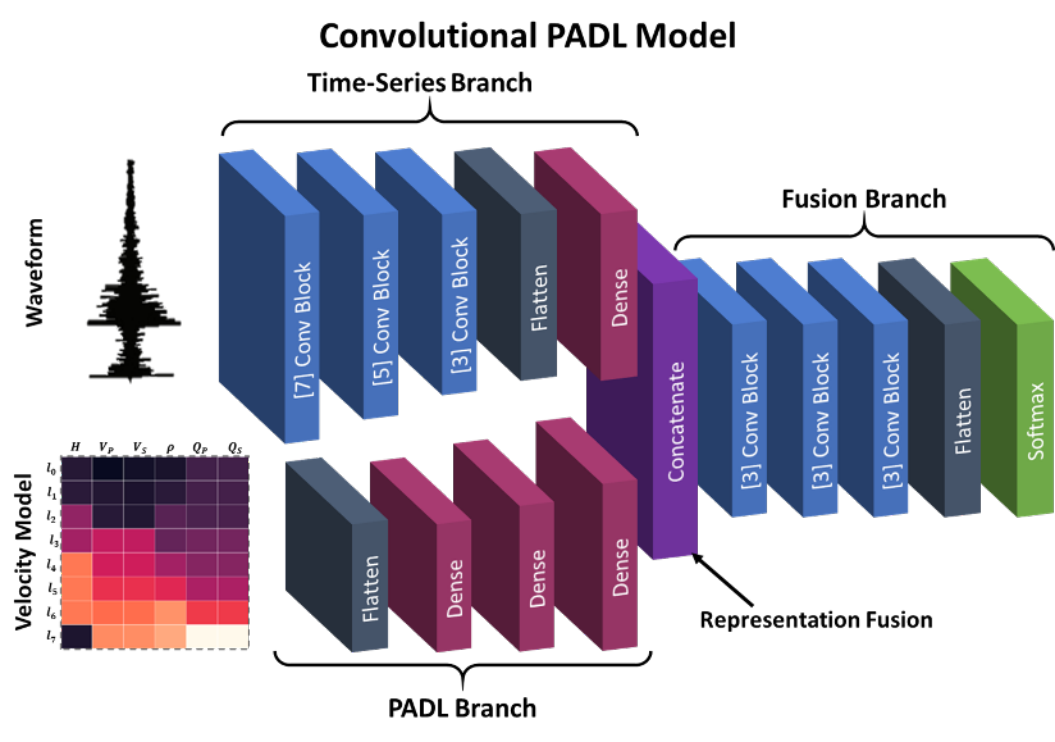

Convolutional PADL Model
Time-Series Branch
Waveform
[7] Conv Block
[5] Conv Block
[3] Conv Block
Flatten
Dense
Fusion Branch
Concatenate
[3] Conv Block
[3] Conv Block
[3] Conv Block
Flatten
Softmax
Velocity Model
H
V_P
V_S
ρ
Q_P
Q_S
l_0
l_1
l_2
l_3
l_4
l_5
l_6
l_7
Flatten
Dense
Dense
Dense
Representation Fusion
PADL Branch

Figure 3.

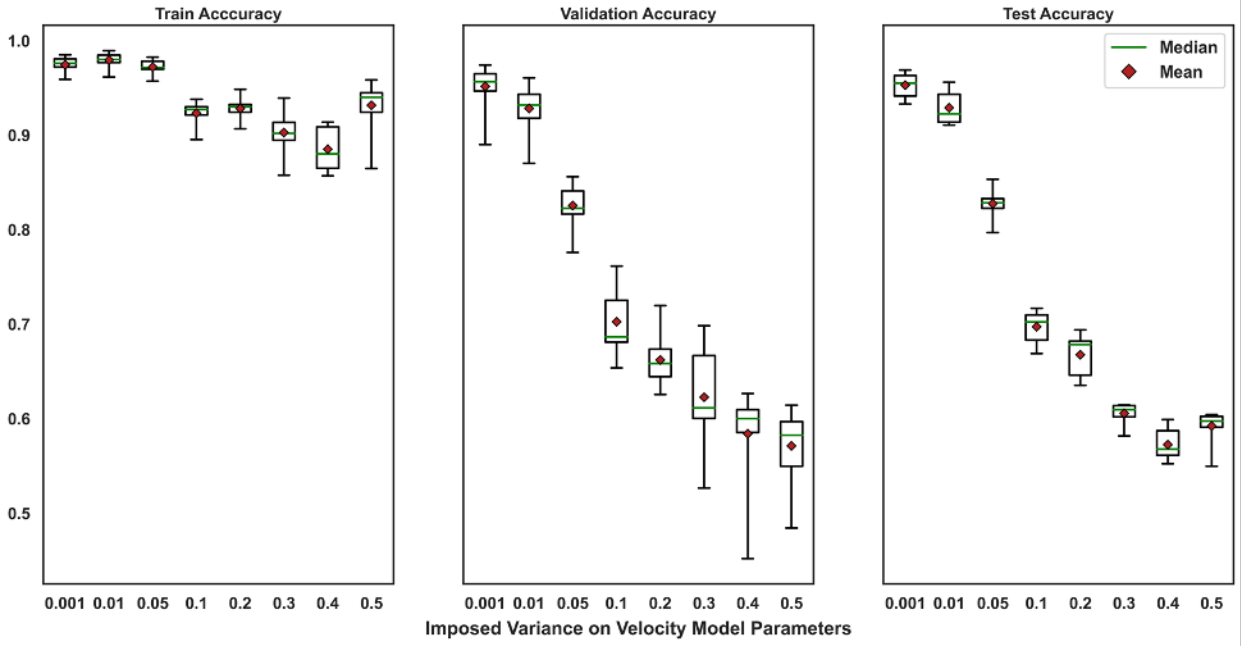

Train Acccuracy
Validation Accuracy
Test Accuracy
Median
Mean
1.0
0.9
0.8
0.7
0.6
0.5
0.001 0.01 0.05 0.1 0.2 0.3 0.4 0.5
0.001 0.01 0.05 0.1 0.2 0.3 0.4 0.5
0.001 0.01 0.05 0.1 0.2 0.3 0.4 0.5
Imposed Variance on Velocity Model Parameters

Figure 4.

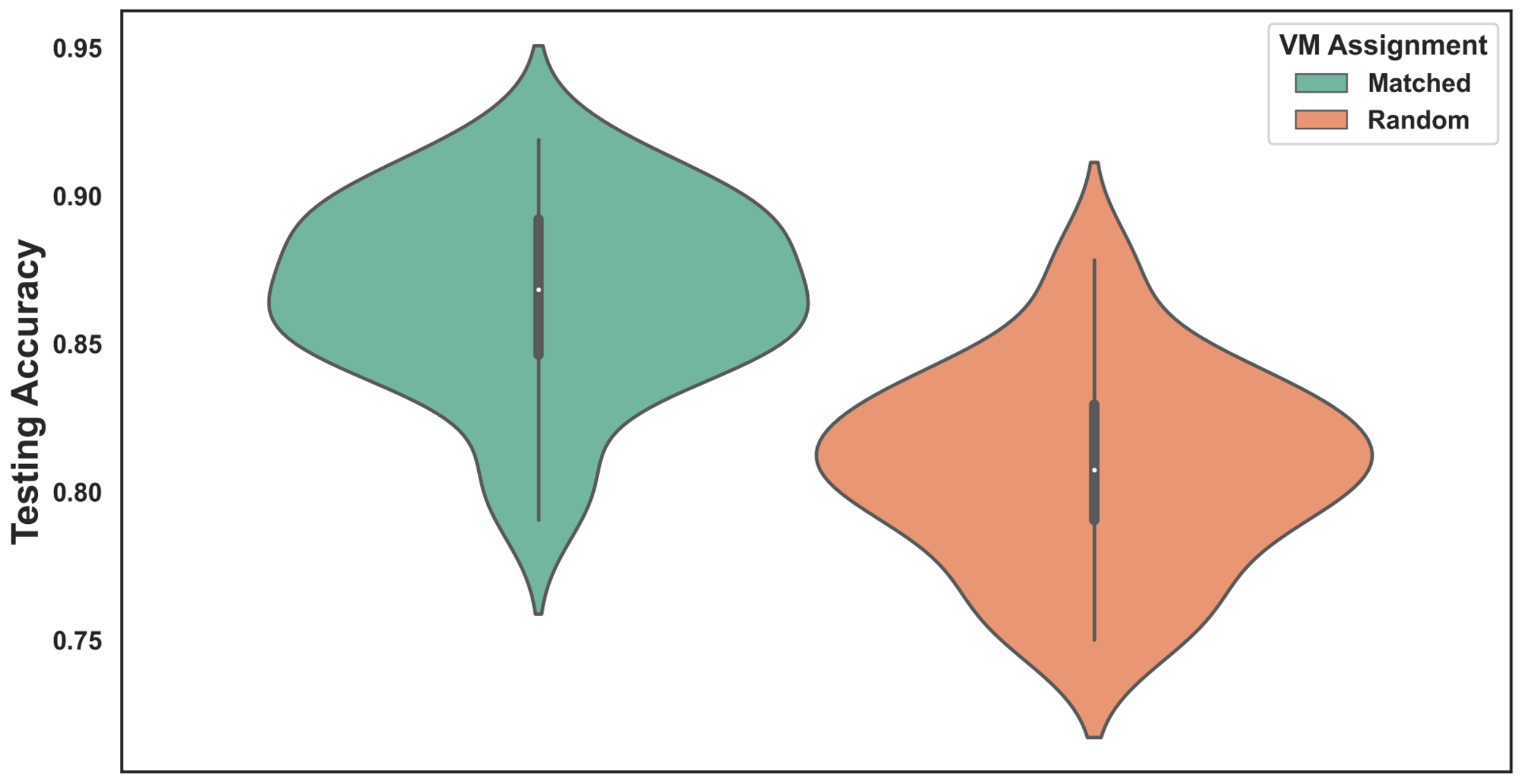
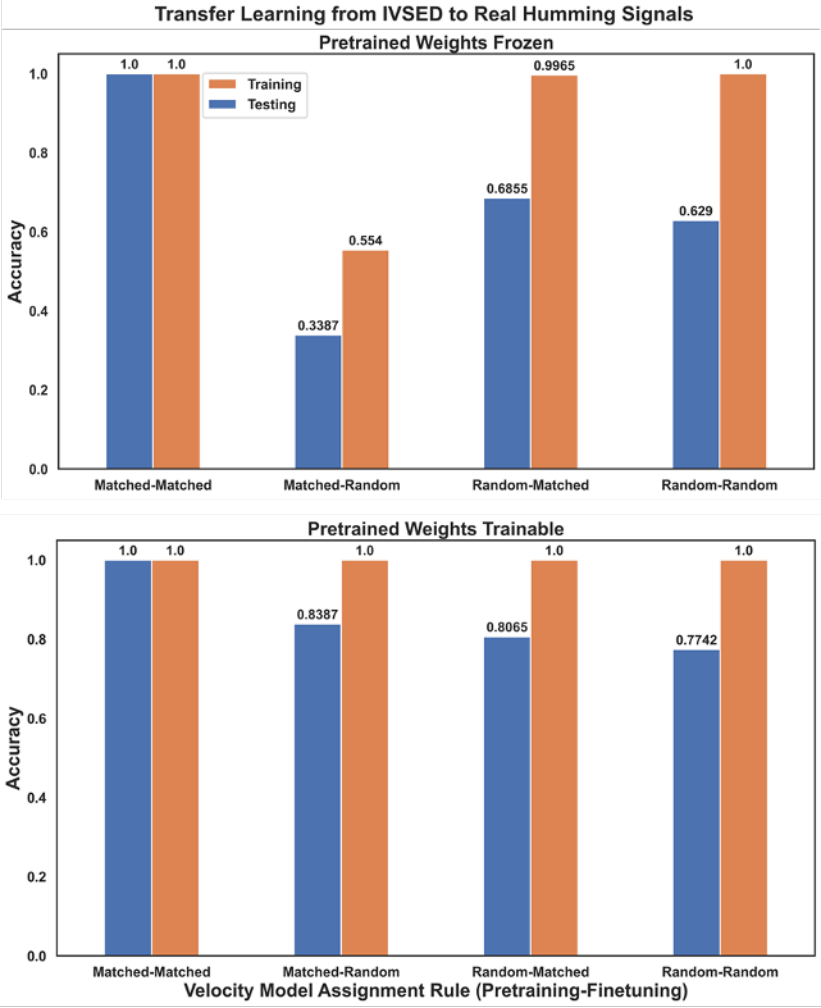

VM Assignment
Matched
Random
Testing Accuracy
0.95
0.90
0.85
0.80
0.75

Figure 5.


Transfer Learning from IVSED to Real Humming Signals
Pretrained Weights Frozen
Training
Testing
Accuracy
1.0
1.0
0.3387
0.554
0.6855
0.9965
0.629
1.0
Matched-Matched
Matched-Random
Random-Matched
Random-Random
Pretrained Weights Trainable
1.0
1.0
0.8387
1.0
0.8065
1.0
0.7742
1.0
Velocity Model Assignment Rule (Pretraining-Finetuning)