Interspeech 2022
18-22 September 2022, Incheon, Korea

PISA: PoIncaré Saliency-Aware Interpolative Augmentation

Ramit Sawhney[1*], *Megh Thakkar*[2*], *Vishwa Shah*[3*], *Puneet Mathur*[4], *Vasu Sharma*[5], *Dinesh Manocha*[4]

[1] University of Marburg, [2] BITS Pilani, [3] BITS Pilani, K.K. Birla Goa Campus,
[4] University of Maryland, College Park, [5] Carnegie Mellon University

rsawhney@mathematik.uni-marburg.de, dmanocha@umd.edu

Abstract

Saliency-aware portion-wise mixup has proven to be an effective data augmentation technique for different modalities and tasks. However, it involves calculating the saliency over gradient vectors in the Euclidean space, representations that often possess complicated geometries and inherent hierarchical structure. We propose PISA, saliency-aware interpolative regularization operating in the hyperbolic space, to better capture the complex geometries of representations. To this end, we also formulate a saliency-aware mixup for speech signals. PISA outperforms existing state-of-the-art interpolative augmentation methods on 7 benchmark and low-resource datasets from the domains of speech signal processing and computer vision. PISA results in more stable training than existing data augmentation methods while being robust to adversarial attacks. It can be generalized across modalities, models and downstream tasks.

Index Terms: speech recognition, human-computer interaction, computational paralinguistics

1. Introduction

Neural network architectures, though effective across numerous modalities and tasks, are vulnerable to over-fitting in the absence of necessary training data [1]. Data augmentation aims at utilizing limited training data to expand the training distribution, enabling the neural network to generalize better. Sample-level data augmentation methods involve altering the shapes or properties of sounds or adding a background noise [2] or cropping and rotation [3]. These methods are however dataset dependent and do not generalize across modalities.

Interpolation-based approaches such as Mixup [4] that generate synthetic samples from convex combinations of inputs and their labels have shown improved performance and generalizability across different modalities. However, because Mixup randomly interpolates inputs to generate synthetic data, the generated samples are locally ambiguous and unnatural, confusing the model [5]. CutMix performs input-level span-wise Mixup, where a portion of a sample is replaced with a portion of another sample thereby generating more interpretable synthetic samples. Since random span-wise Mixup can provide misleading supervisory signals to the network, saliency-based methods [6, 7] utilize the saliency information and the underlying statistics of the natural examples to preserve the locality of the source inputs. This saliency calculation is performed in the Euclidean space, which is not capable in capturing the complex geometries possessed by network gradient vectors used to calculate saliency.

The hyperbolic space has a more generalized geometric representation and has proven effective in modeling representations with complex hierarchical natures [8]. The interference of sound and light waves is hyperbolic [9], indicating that the latent vectors representing them also inherently possess characteristics more expressive in the hyperbolic space [10]. Building on prior research in interpolative data augmentation, and the hyperbolic characteristics of speech and vision representations, we propose PISA, a saliency-aware mixup method capable of leveraging the hyperbolic space. In order to formulate PISA, we introduce saliency-based input interpolation for speech signals and extend the mean saliency calculation to the hyperbolic space. We probe the effectiveness of PISA on 7 benchmark and low-resource datasets used for speech emotion recognition and image classification; these datasets have varying class distributions and language roots.

PISA outperforms existing interpolative data augmentation techniques while enabling more stable training and is robust to adversarial attacks. Our key contributions are:

- PISA, a novel saliency-aware interpolative data augmentation method leveraging the hyperbolic space. To this end, we also formulate saliency-aware mixup for speech signals.
- PISA outperforms 5 speech and 2 vision benchmarks, showing great improvement on low-resource datasets with varying levels of class imbalances and labelled data sources.
- PISA leads to more stable training because it achieves threshold error rate scores in 35% fewer epochs than the existing methods and is more robust to adversarial attacks.

2. Methodology

2.1. Preliminaries

Mixup [4] is a data augmentation technique that creates virtual examples using linear interpolation of existing instances. It generates convex combinations from a pair of instances (x_i, x_j) and labels (y_i, y_j) with mixing ratio λ for training neural network architectures on synthetic samples,

$$\begin{aligned} x' &= \text{mix}(x_i, x_j) = \lambda \cdot x_i + (1-\lambda) \cdot x_j, \\ y' &= \text{mix}(y_i, y_j) = \lambda \cdot y_i + (1-\lambda) \cdot y_j. \end{aligned} \tag{1}$$

Hyperbolic Space [8] is a non-Euclidean geometry which is a Riemannian manifold with a constant negative sectional curvature. Similar as in [8], We use the Poincaré ball model of the hyperbolic space to perform mathematical operations.
Möbius Addition produces $\oplus$ for a pair of points x, y,

$$x \oplus y := \frac{(1 + 2\langle x, y\rangle + ||y||^2)x + (1 - ||x||^2)y}{1 + 2\langle x, y\rangle + ||x||^2||y||^2}. \tag{2}$$

where $\langle ., . \rangle$ denotes the Euclidean inner product and $||\cdot||$ denotes the Euclidean norm.
To project vectors between Euclidean and hyperbolic space, we use exponential and logarithmic maps. Exponential Mapping

*Equal contribution.

10.21437/Interspeech.2022-11001

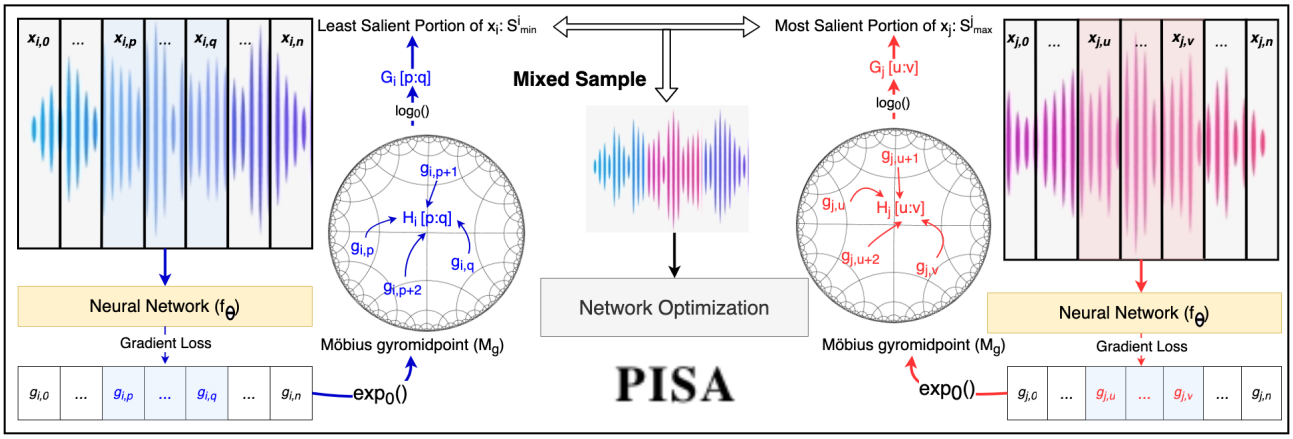

Figure 1: *Overview of* PISA *with Möbius gyromidpoint to replace the least salient set of x_i with the most salient set of x_j.*

maps the tangent vector u to the point $\exp_x(u)$ on the Poincaré ball, where $\lambda_x = \frac{2}{1-||x||^2}$,

$$\exp_x(u) = x \oplus \left(\tanh\left(\frac{\lambda_x \|u\|}{2}\right)\frac{u}{\|u\|}\right). \tag{3}$$

Logarithmic Mapping maps a point y to a point $log_x(y)$ on the tangent space at x,

$$\log_x(y) = \frac{2}{\lambda_x}\tanh^{-1}(\|-x\oplus y\|)\frac{-x\oplus y}{\|-x\oplus y\|}. \tag{4}$$

For exponential and logarithmic mapping, we choose the tangent space center $x=0$ and use $\exp_0(\cdot)$ and $\log_0(\cdot)$. Möbius Scalar Multiplication $\odot$ multiplies matrix $\boldsymbol{x} \in \mathcal{B}$ with scalar $r \in \mathcal{B}$,

$$r \odot \boldsymbol{x} = \tanh\left(r\tanh^{-1}(\|\boldsymbol{x}\|)\right)\frac{\boldsymbol{x}}{\|\boldsymbol{x}\|}. \tag{5}$$

Weighted Möbius gyromidpoint M_g of a set of points $x_1, .., x_n$ according to weights $\alpha_1, .., \alpha_n$ calculates the weighted pooling in the hyperbolic space,

$$\begin{aligned} &M_g(x_1,\ldots,x_n,\alpha_1,\ldots,\alpha_n) \\ &= \frac{1}{2}\odot\left(\sum_{i=1}^{n}\frac{\alpha_i\lambda_{x_i}}{\sum_{j=1}^{n}\alpha_j(\lambda_{x_j}-1)}x_i\right). \end{aligned} \tag{6}$$

Hyperbolic Linear Layer ($HL(\cdot,\cdot)$) performs Möbius matrix vector multiplication of input x with weight matrix $W : \mathbb{R}^n \rightarrow \mathbb{R}^m$, This serves for mapping or representing an input from n dimension to m dimension.

$$HL(x,W) = \tanh\left(\frac{\|Wx\|}{\|x\|}\tanh^{-1}(\|x\|)\right)\frac{Wx}{\|Wx\|}. \tag{7}$$

2.2. PISA: PoIncaré Saliency-Aware Interpolation

Compared to randomly interpolating raw inputs, combining inputs while preserving their locality has proven to be more effective for various modalities [6]. We perform input-level mixup where a portion of a sample is replaced with a portion of another sample. This portion is selected based on its mean saliency to make the mixed sample more closely related to the output prediction while preserving the locality of the source inputs [7]. Gradient-based methods are being widely used for saliency computation [11, 12]. For an input $x = [x_1, x_2, \ldots, x_n]$ made up of n units (we define a unit as a sound pressure level for speech and a pixel for images) with the gradient of classification loss $\mathcal{L}$, the saliency g_k for unit s_k is given as,

$$g_k = \delta\mathcal{L}/\delta s_k \tag{8}$$

Because the inputs to the neural networks and their latent representations inherently possess complex geometries, their classification gradients cannot be captured effectively in the Euclidean space [8]. We leverage the hyperbolic space to calculate the mean saliency $G_{p:q}$ for a set of contiguous units $x[p:q] \subset x$, where p and q are each used to index the respective unit. We first map the saliency g_k for each unit k in the set $x[p:q]$ to the hyperbolic space using $\exp_0(\cdot)$ and apply Weighted Möbius gyromidpoint (M_g) with equal weight 1 to all the input units to obtain hyperbolic mean saliency $H[p:q]$. We map the hyperbolic saliency to the Euclidean space using $\log_0(\cdot)$ and obtain the mean saliency $G[p:q]$. Formally,

$$H[p:q] = M_g(\exp_0(g_p),\ldots,\exp_0(g_q),1,\ldots,1) \tag{9}$$

$$G[p:q] = \log_0(H[p:q]) \tag{10}$$

In our approach PISA, for mixup between inputs x_i and x_j, we replace the least salient portion $x_i[p:q]$, S^i_{min} in x_i with the most salient portion in $x_j[u:v]$, S^j_{max}. We formally define PISA to generate sample $\bar{x}$ with transport η from $[p:q] \rightarrow [u:v]$,

$$\bar{x} = \text{PISA}(x_i, x_j), \quad \bar{x}_k = \begin{cases} x_{i,k} & k \notin [p:q] \\ x_{j,k+\eta} & k \in [p:q] \end{cases} \tag{11}$$

Network Optimization We optimize the network $f_\theta(\cdot)$ using Cross Entropy loss CE between the mixed example $\bar{x}$ and its weighted label $\bar{y} = \text{mix}(y_i, y_j)$. We define mixup loss $\mathcal{L}_M$,

$$\begin{aligned} \mathcal{L}_M(x_i,x_j) = \lambda * \text{CE}(y_i||f_\theta(\text{PISA}(x_i,x_j))) + \\ (1-\lambda) * \text{CE}(y_j||f_\theta(\text{PISA}(x_i,x_j))) \end{aligned} \tag{12}$$

For samples x_i and x_j, we optimize our network as a mean of four losses to generate $\mathcal{L}$,

$$\begin{aligned} \mathcal{L} = \frac{1}{4} * \Big(\text{CE}(y_i||f_\theta(x_i)) + \text{CE}(y_j||f_\theta(x_j)) + \\ \mathcal{L}_M(x_i,x_j) + \mathcal{L}_M(x_j,x_i)\Big) \end{aligned} \tag{13}$$

PISA for Speech Following the above described method, for an n length speech input x, where each unit represents the sound pressure level, we choose contiguous spans $x[p:q]$ of length $m = \lambda_0 * n$ from x as portions of speech to perform PISA. We compute the mixing ratio λ as $\lambda = (1 - \lambda_0 * \frac{n}{n_0})$, where n_0 is the unpadded length of the input. We pool over the input to find mean saliency for all contiguous spans using a stride. The network is optimized using the loss from mixed samples with equations (12) and (13).

PISA for Vision As presented in [6], the authors compute gradient with respect to the image input representation and use the mean L2 norm across the channels as the saliency representation for each pixel. For PISA, we transform this step to the Hyperbolic space where the squared gradient vector p_g for each pixel of dimension equal to the number of channels n_c is passed through the Hyperbolic Linear Layer $HL(.,.)$ whose weight W inputs are set to a constant value of $\frac{1}{n_c}$. The operation, $HL(p_g, W)$ maps to a single value for each pixel which serves as pooling along the channels. Post this, we follow [6] for network optimization during training to learn optimal mixing mask and transport based on the saliency information computed.

3. Experiments

Table 1: *Datasets, languages, # classes and # samples.*

	Dataset	Language	# Classes	# Samples
Speech	Urdu SER [13]	Urdu	4	400
	EMOVO [14]	Italian	7	588
	EmoDB [15]	German	7	500
	SAVEE [16]	English	7	480
	ShEMO [17]	Persian	5	3000
	Dataset	**Class**	**# Classes**	**# Samples**
Vision	CIFAR-10 [18]	Object	10	60,000
	STL-10 [19]	Object	10	13,000

We consider benchmark and low-resource datasets across speech and vision spanning a varying number of classes, lower language resources, different structures, and language roots for a comprehensive evaluation of PISA. For speech, we evaluate on speech emotion classification across datasets in five languages: Urdu SER(Urdu) [13], EMOVO(Italian) [14], EmoDB(German)[15], SAVEE (English)[16] and ShEMO(Persian)[17]. For vision we evaluate across image classification datasets CIFAR-10 [18] and STL-10 [19]. [1]

3.1. Baselines

Speech Following previous works, we use EnvNet-v2 with strong augmentation [2] as our base architecture. We compare PISA with BC Learning [2] and the state-of-the-art interpolative method Speechmix [20].

Vision Following [21], we use PreActResNet18 as our base model and compare PISA with Mixup [4], Manifold mixup [21], and state-of-the-art PuzzleMix [6].

3.2. Data Preprocessing

Speech We completely remove the silent sections in which the value was 0 at the beginning and the end of the samples, and then implement center padding to ensure all samples are identical in length. Next, all sound files are converted to 16-bit WAV files. Following [2], we perform random cropping during training and validation. We divide the input data by $32,768$ to regularize each input data into a range of [-1,1].

Vision Following [6], we normalize all the images at pixel level. We center crop STL-10 images to 64x64 dimensions.

3.3. Training Setup

Speech For PISA, λ_0 is set to 0.2 and a stride of 0.10*n is used while pooling over an n-length input. We use Nesterov's accelerated gradient [22] with a momentum of 0.9, weight decay of 5e-4, learning rate of 0.01 and mini-batch size of 64 for 1200 epochs. We use a 80:20 split data split for training and testing respectively and use the absolute value of the gradient vector as a measure of saliency.

Vision We use stochastic gradient descent (SGD) with an initial learning rate of 0.2 decayed by factor of 0.1 at epochs 350 and 500 and train for a total of 600 epochs. We set the momentum as 0.9 and add a weight decay of 1e-4. PuzzleMix[6] has hyperparameters of β for the label smoothness term, γ for the data smoothness term, η for prior term, and ξ for the transport cost. We use $\beta = 1.2, \gamma = 0.5, \eta = 0.2$ and $\xi = 0.8$ for both PuzzleMix and PISA. We sample the mixing ratio λ randomly from $Beta(1,1)$ for all experiments except Manifold Mixup which uses $Beta(2,2)$ in the original paper. We use a batch size of 100 for CIFAR-10. For STL-10 we use a training batch of size 96 and test batch of size 256.

3.4. Training Details

For the computing infrastructure we use Tesla P100 GPU for all our experiments.

Speech The time taken by PISA to complete 1200 epochs for Urdu SER: 15 hours, SAVEE: 25 hours, EmoDB: 20 hours, EMOVO: 23 hours and ShEMO: 66 hours. EnvNet-v2 [2] has about $101M$ parameters. PISA is applied over EnvNet-v2 with the same number of trainable parameters.

Vision For the highest # Samples $n = 500$ for CIFAR-10 and STL-10, PISA trains for 600 epochs in approximately 5 hours and 11 hours respectively. PuzzleMix [6] has about $11.2M$ parameters. PISA is applied over Puzzle Mix with the same number trainable parameters.

4. Results and Analysis

4.1. Performance Comparison: Speech

We compare the performance of PISA on five low-resource and benchmark speech datasets in Table 2. On applying BC Learning [2] and SpeechMix [20] over EnvNet-v2 with strong augmentation, we observe that mixup-based approaches improve performance over standard learning models. This corroborates the significance of interpolative acoustic mixup based on the auditive perception of the input samples in improving the model performance. EISA is our Euclidean counterpart of PISA, replacing hyperbolic mean saliency calculation (Equation 9) with standard weighted average. EISA significantly ($p < 0.01$) outperforms existing interpolative augmentation methods, indicating that saliency-aware mixup can preserve the locality of input samples and incorporates discriminative salient audio spans which are more closely related to the output prediction class. We observe the greatest improvement in the case of PISA, which leverages the hyperbolic space to calculate the portion mean

[1] "https://anonymous.4open.science/r/PISA-4BBF": Our code is available at this link

Table 2: *Performance comparison of* PISA *with baseline methods in terms of % Error rate (mean of 10 runs).* ***Bold*** *shows the best result.* * *shows significant* $(p < 0.01)$ *improvement over SpeechMix under Wilcoxon's signed rank test.*

Model	Urdu	EMOVO	EmoDB	SAVEE	ShEMO
EnvNet-v2 [2]	8.62	26.13	23.36	41.11	24.79
BCLearning [2]	7.39	25.57	15.51	35.66	24.45
SpeechMix [20]	6.20	22.19	16.34	34.38	22.09
EISA(Ours)	5.77*	23.01	15.20*	33.94*	21.13*
PISA(Ours)	**5.00***	**22.03***	**14.02***	**33.33***	**20.31***

Table 3: *Performance comparison of* PISA *on vision data with baseline methods in terms of % Error rate (mean of 10 runs).* n *is the number of labeled training samples per class.* ***Bold*** *shows the best result.* * *shows significant* $(p < 0.01)$ *improvement over PuzzleMix under Wilcoxon's signed rank test.*

Model	**CIFAR-10**			**STL-10**		
# Samples n	10	100	500	50	100	500
PreActResNet [23]	65.70	34.20	14.90	50.53	49.33	18.78
Input Mixup [4]	65.10	32.49	14.18	47.99	37.19	17.84
Manifold Mixup [21]	64.60	32.30	14.10	46.04	36.51	16.71
EISA [6]	64.73	31.82	12.60	47.14	36.24	**15.29**
PISA (Ours)	**64.50***	**31.53***	12.60	**46.69***	**36.00***	15.64

saliency. This validates that the hyperbolic geometry better captures the mean portion saliency mapping of speech signals and acoustic wave interference and leads to more suitable saliency measurements compared to the Euclidean space.

4.2. Performance Comparison: Vision

To further probe the effectiveness of PISA, we apply it to benchmark vision datasets in various resource settings by varying the number of samples present in the training set in Table 3. PuzzleMix [6] performs better than standard mixup techniques in most settings because it utilizes saliency information to perform mixup, preventing misleading signals generated from random mixup being input to the model. PISA which uses Möbius operations for mean saliency calculation further improves performance over EISA [6](PuzzleMix), with the gains being higher in extremely low resource conditions. This demonstrates the capability of hyperbolic space to better model the network gradient vectors compared to the Euclidean space on account of their complex geometries.

4.3. Diversity: Probing Stability of PISA

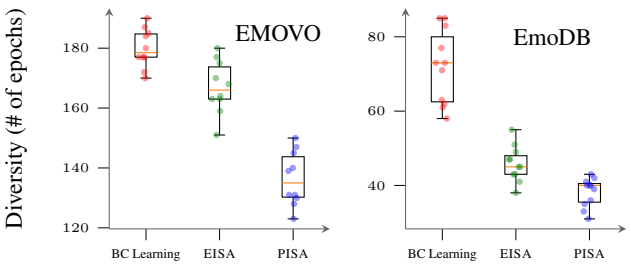

Figure 2: *Comparison of* PISA *with* EISA *and BC Learning in terms of number of training epochs required to achieve benchmark error rates.*

Table 4: *% Error rates on adversarial examples generated using white box FGSM and I-FGSM attacks.*

Model	**Urdu SER**		**EMOVO**	
	FGSM	**I-FGSM**	**FGSM**	**I-FGSM**
SpeechMix [20]	68.74	82.50	85.59	96.61
EISA	65.97	77.42	83.05	94.06
PISA	65.37	76.81	82.71	93.88

We probe the quality of generated synthetic samples using PISA by examining the training stability of the base models compared with different interpolative data augmentation methods. For our evaluation, we consider the proxy definition of diversity given by [24], defined as the number of training epochs required to obtain a benchmark error rate. We observe that across all datasets, PISA achieves a benchmark error rate score in fewer training epochs than BC Learning [2] (Figure 2). Since PISA performs interpolation using saliency of the input represented in the hyperbolic space, it is able to generate more suitable interpolations leading to a faster generalization of the model, achieving threshold error rate scores in 35% fewer epochs than the existing interpolative data augmentation methods.

4.4. Robustness to Adversarial Attacks

Adversarial input examples are specifically crafted to confuse models, resulting in misclassification of inputs. We compare the robustness of PISA and EISA with Speechmix [20] by performing white-box adversarial attacks using Fast Gradient Sign Method (FGSM) [25] and Iterative Fast Gradient Sign Method (I-FGSM) [26] and present the Error rates in Table 4. We observe that EISA is more robust than SpeechMix, indicating that saliency-aware methods generate more suitable samples as they retain more salient parts of inputs. PISA is even more robust than EISA, suggesting that the hyperbolic space is able to capture the complex geometries of loss gradients and better model the inputs' locality statistics for saliency calculation.

5. Conclusion

We propose PISA, a novel saliency-aware interpolative regularization method operating in hyperbolic space. We leverage the fact that speech and vision gradient vectors possess localized statistics and a hyperbolic nature owing to their complex geometry. While being model-, data- and modality- agnostic, PISA outperforms existing state-of-the-art interpolative augmentation methods on 7 benchmark, low-resource datasets across speech (in 5 languages) and vision. PISA observes more stable training than existing data augmentation methods and is more robust to white-box adversarial attacks. For future work, we plan to evaluate PISA on a wider range of tasks and multimodal settings.

6. References

[1] D. Zou and Q. Gu, "An improved analysis of training over-parameterized deep neural networks," in *Advances in Neural Information Processing Systems*, 2019.

[2] Y. Tokozume, Y. Ushiku, and T. Harada, "Learning from between-class examples for deep sound recognition," 2018.

[3] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, 1998.

[4] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," 2018.

[5] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, and Y. Yoo, "Cutmix: Regularization strategy to train strong classifiers with localizable features," in *International Conference on Computer Vision (ICCV)*, 2019.

[6] J.-H. Kim, W. Choo, and H. O. Song, "Puzzle mix: Exploiting saliency and local statistics for optimal mixup," in *International Conference on Machine Learning (ICML)*, 2020.

[7] S. Yoon, G. Kim, and K. Park, "SSMix: Saliency-based span mixup for text classification," in *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, Online, Aug. 2021. [Online]. Available: https://aclanthology.org/2021.findings-acl.285

[8] O. Ganea, G. Becigneul, and T. Hofmann, "Hyperbolic neural networks," in *Advances in Neural Information Processing Systems*, 2018.

[9] M. N. Khan and S. Panigrahi, *Interference*. Cambridge University Press, 2016, p. 98–185.

[10] R. Sawhney, M. Thakkar, S. Agarwal, D. Jin, D. Yang, and L. Flek, "Hypmix: Hyperbolic interpolative data augmentation," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP 2021)*, 2021.

[11] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," 2014.

[12] J. Li, X. Chen, E. Hovy, and D. Jurafsky, "Visualizing and understanding neural models in NLP," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, San Diego, California, Jun. 2016. [Online]. Available: https://aclanthology.org/N16-1082

[13] S. Latif, A. Qayyum, M. Usman, and J. Qadir, "Cross lingual speech emotion recognition: Urdu vs. western languages," 2020.

[14] G. Costantini, I. Iaderola, A. Paoloni, and M. Todisco, "EMOVO corpus: an Italian emotional speech database," in *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, May 2014. [Online]. Available: http://www.lrec-conf.org/proceedings/lrec2014/pdf/591_Paper.pdf

[15] F. Burkhardt, A. Paeschke, M. Rolfes, W. Sendlmeier, and B. Weiss, "A database of german emotional speech," 2005.

[16] S. Haq and P. J. B. Jackson, "Speaker-dependent audio-visual emotion recognition," in *AVSP*, 2009.

[17] O. M. Nezami, P. J. Lou, and M. Karami, "Shemo - A large-scale validated database for persian speech emotion detection," *CoRR*. [Online]. Available: http://arxiv.org/abs/1906.01155

[18] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," *Master's thesis, Department of Computer Science, University of Toronto*, 2009.

[19] A. Coates, A. Ng, and H. Lee, "An analysis of single-layer networks in unsupervised feature learning," in *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, 2011. [Online]. Available: https://proceedings.mlr.press/v15/coates11a.html

[20] A. Jindal, N. E. Ranganatha, A. Didolkar, A. G. Chowdhury, D. Jin, R. Sawhney, and R. R. Shah, "Speechmix—augmenting deep sound recognition using hidden space interpolations." *Proc. Interspeech 2020*, pp. 861–865, 2020.

[21] V. Verma, A. Lamb, C. Beckham, A. Najafi, I. Mitliagkas, D. Lopez-Paz, and Y. Bengio, "Manifold mixup: Better representations by interpolating hidden states," in *Proceedings of the 36th International Conference on Machine Learning*, vol. 97, 2019, pp. 6438–6447. [Online]. Available: http://proceedings.mlr.press/v97/verma19a.html

[22] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, "On the importance of initialization and momentum in deep learning," in *Proceedings of the 30th International Conference on Machine Learning*. [Online]. Available: https://proceedings.mlr.press/v28/sutskever13.html

[23] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *Computer Vision – ECCV 2016*, 2016.

[24] R. Gontijo-Lopes, S. J. Smullin, E. D. Cubuk, and E. Dyer, "Affinity and diversity: Quantifying mechanisms of data augmentation," *arXiv preprint arXiv:2002.08973*, 2020.

[25] I. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *International Conference on Learning Representations*, 2015. [Online]. Available: http://arxiv.org/abs/1412.6572

[26] A. Kurakin, I. Goodfellow, S. Bengio *et al.*, "Adversarial examples in the physical world," 2016.