

# Long Tail Image Generation Through Feature Space Augmentation and Iterated Learning

Rafael Elberg      Denis Parra      Mircea Petrache  
PUC Chile

{rafael.elberg, denis.parra, mpetrache}@uc.cl

## Abstract

*Image and multimodal machine learning tasks are very challenging to solve in the case of poorly distributed data. In particular, data availability and privacy restrictions exacerbate these hurdles in the medical domain. The state of the art in image generation quality is held by Latent Diffusion models, making them prime candidates for tackling this problem. However, a few key issues still need to be solved, such as the difficulty in generating data from under-represented classes and a slow inference process. To mitigate these issues, we propose a new method for image augmentation in long-tailed data based on leveraging the rich latent space of pre-trained Stable Diffusion Models. We create a modified separable latent space to mix head and tail class examples. We build this space via Iterated Learning of underlying sparsified embeddings, which we apply to task-specific saliency maps via a K-NN approach. Code is available at <https://github.com/SugarFreeManatee/Feature-Space-Augmentation-and-Iterated-Learning>*

## 1. Introduction

With the rise of large multimodal models [10, 22, 33] and Latent Diffusion models [26, 30], both image analysis and generation tasks have become dependent on the availability and quality of balanced training data [3]. Due to the representation capabilities of these large models with several million, or even billions of parameters, volumes of data in similar magnitudes are required to avoid issues such as overfitting or learning undesirable biases.

Obtaining extensive and well-distributed data is not an option for several essential domains. For example, for the analysis and generation of medical images, obtaining data can be complex (as they are subject to patient confidentiality) and expensive (as they are tied to real-world medical procedures and imaging exams) [11]. Moreover, it is often impossible to obtain large, well-distributed samples of im-

ages corresponding to certain diseases and anomalies that, while relevant, do not occur frequently [12].

Two avenues for solving the lack of data in these domains are *resampling* and *data augmentation*. The former consists of artificially oversampling examples from low-frequency data while sometimes reducing the number of samples from high-frequency data. The latter consists of generating synthetic data for under-represented classes to even out the distribution of a dataset.

Resampling techniques have been used with relative success in several long-tailed problems, but can introduce *unwanted biases* into downstream tasks [31] and often lead to *overfitting* [18].

Data augmentation is the natural response to these issues. It represents a booming area of research comprising several different families of algorithms, such as geometric transformations (rotations, scaling, cropping, etc.), creation of synthetic samples, mixing-based methods [2, 7, 35] domain translation-based methods [38] and generative methods [23, 24].

We propose a new data augmentation method that manipulates latent space representations of images from pre-trained diffusion models, thereby generating new images to augment under-represented classes. Specific features of the data are selected via activation maps, which are then combined to produce images similar to the ones from actual data belonging to long tail classes.

The combination of latent space representations is challenging to perform through naive methods due to *interference phenomena* between the post-processing of the features. We tackle this issue as a problem of *compositional generalization* and apply the framework of *iterated learning* (IL) [28] with sparsified embeddings to our target data augmentation framework.

The main inspiration of IL comes from models of cultural evolution [19] in which iterations of teacher-student interactions encourage useful compression and the formation of a "shared language" adapted to a task [4, 20, 28]. In particular, recently [29] have obtained favorable results related to composing distinct features when using sparsified state spaces,

with a sparsification method called *Simplicial Embedding* (SE) [21]. The concept entails that if, in an IL iteration, the "teacher" version of the model is obliged to propagate a sparsified version of latent vectors to a new "student" version before training, it will impose an information bottleneck across iterations, resulting in significant improvements at downstream tasks that require compositional reasoning.

In short, we applied a new version of the IL+SE method to map the already rich latent space of a pre-trained Stable Diffusion Model [30] using task-specific activation masks [8]. We propose mixing existing points in this sparse latent space to achieve better fusion at the feature level.

## 2. Related Work

As previously mentioned, several data augmentation techniques have been used recently; see [34] for a review. Of the methods described, the most relevant to this work are Image Mixing and Deep Generative Model approaches.

### 2.1. Image mixing

Image mixing generally consists of creating new data points by combining two or more existing ones. MixUp[36] and SMOTE[7] utilize convex combinations of existing data to create new samples, with SMOTE selecting the same class neighbors as pairs. CutMix[35] randomly samples from the base dataset and removes patches from an image to replace with a patch from another. These methods can also be performed in the feature space.

The main problems of these methods are that they fail to produce novel, realistic samples and lack consistency in preserving labels before and after augmentation. We aim to solve the first issue by leveraging the latent space of a pre-trained stable diffusion model. For the second issue, we attempt to solve the problem using saliency methods and a separable sparse latent space.

### 2.2. Deep Generative Models

Approaches based on generative models sample new training examples through model inference. GANs [14] are the most popular generative framework for data augmentation due to their fast inference and realistic generation [5]. However, GANs are widely known to be unstable at training time, and are prone to mode collapse [37].

Latent Diffusion Models [30] have long surpassed GANs in image generation quality [9], and off-the-shelf models have been used to significant effect in data augmentation [27, 32]. However, Diffusion Models generally suffer from slow inference speed and quality degeneration with long-tailed data [27]. To address these issues, we work within a modified sparse version of the rich latent space defined by a trained Stable Diffusion Model, to mix existing data points instead of generating new ones from scratch. By doing this,

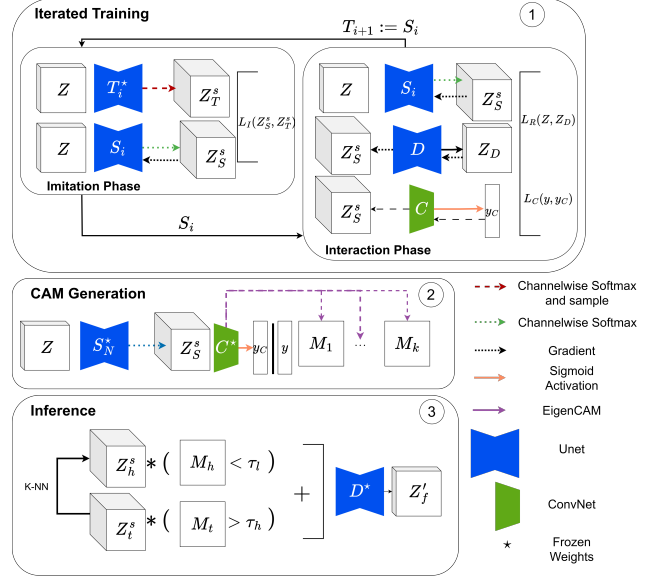


Figure 1. Proposed method: Stage 1 (Iterated training), iteratively train a student network  $S_i$  to imitate a frozen teacher network  $T_i$ , which corresponds to the student network of the previous iteration  $S_{i-1}$  in mapping the original latent vectors  $Z$  to a semantically separable sparse domain  $Z^s$ . Also, jointly train said student with a classifier  $C$  and a decoder  $D$  to classify and map vectors from the sparse domain back to the original domain. In Stage 2 (CAM generation), we use EigenCAM to generate class activation maps ( $M_i$  for classes  $I$  in  $[1, k]$ ) for each vector, using the classifier trained in Stage 1. Finally, in stage 3 (Inference), we find a head class near neighbor  $Z_h^s$  for each tail class vector  $Z_t^s$ , and we combine them using their respective Class Activation Maps (CAM) as masks, taking the top activations from the tail vector and the bottom activations from the head vector. Finally, we combine these activations and pass them through  $D$  to generate a new tail class vector.

we can create high-quality samples with very few (if any) diffusion inference steps.

## 3. Method

Our proposed method, shown in Figure 1, consists of three stages: (i) iterated training, (ii) class activation map generation, and (iii) inference. Note that each stage is applied to the latent vectors produced by a pre-trained Latent Diffusion Model, and not to the images themselves.

### 3.1. Iterated training

In this stage, we learn a translation from the diffusion latent space to a sparse high dimensional representation [21], while training a convolutional classifier for this space.

To do this, we implement a student-teacher [1] training regime, which iteratively runs through two phases: imitation and interaction.

We train a student network  $S_i : \theta \times Z \rightarrow Z^s$  during the

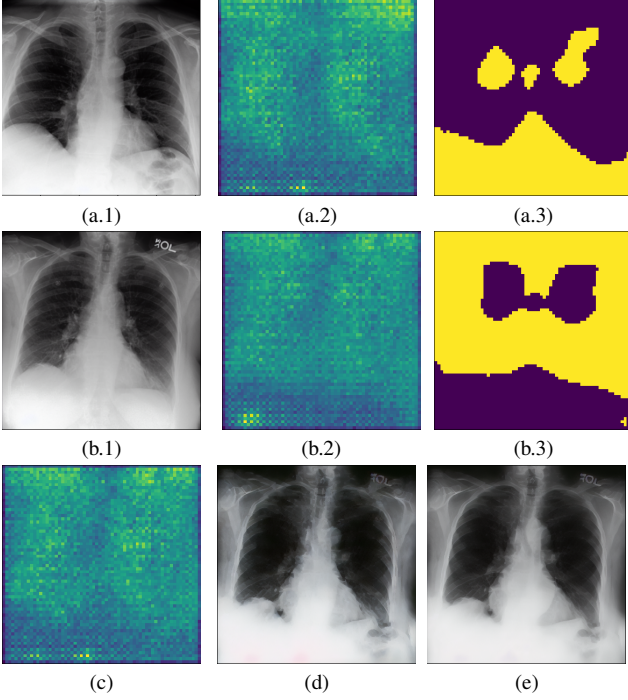


Figure 2. Fusion process applied to an image from the tail class Tortuous Aorta (a.1) and one of its neighbor images from the head class Atelectasis (b.1). (a.2) and (b.2) are channelwise Maximum Intensity Projections of the sparse vectors obtained from (a.1) and (b.1) respectively. In (a.3) and (b.3), we use EigenCAM to find attention maps for each sparse vector and define binary masks (yellow is one and dark purple is zero) using  $\tau_h = \tau_l = 0, 4$  as thresholds. We combine the masked sparse vectors into (c) and decode the vector into a fused image (d). Finally, we apply five inference steps in (e) to obtain a less noisy image.

imitation phase from scratch. This network transforms latent vectors  $Z \subseteq \mathcal{R}^{H \times W \times C}$  into sparse high dimensional vectors  $Z_S^s \subseteq \mathcal{R}^{H \times W \times C'}$ , with  $C < C'$ . We enforce sparsity by applying a channel-wise softmax to these output vectors, as follows:

$$z_{ijk}^S := \frac{\exp(S_i(z_{ijk}))}{\sum_{k=1}^{C'} \exp(S_i(z_{ijk}))}. \quad (1)$$

We train  $S_i$  to imitate a teacher network  $T_i$ , which is the student of the previous iteration,  $T_i = S_{i-1}$ . Instead of imitating  $T_i$  directly, we use the channel-wise probabilities coded in  $T_i(z^S)$  to sample a binary vector  $z_b^S$ , where each spatial coordinate is a one-hot vector of dimension  $C'$ . Minimizing the imitation loss then becomes a multilabel classification problem:

$$L_I = BCE(z_b^S, S_i(z^S)) \quad (2)$$

During the interaction phase, we minimize losses related to two other networks. First, we aim to make the sparse vectors good representations of the original latent vectors. To this

end, we jointly train a network  $D : \theta \times Z' \rightarrow Z$  with  $S_i$ , to minimize the reconstruction loss:

$$L_R = MSE(z, D(S_i(z_s))). \quad (3)$$

Our objective entails ensuring that the sparse vectors are easily separable in their respective image classes; with this in mind, we trained a classifier  $C : \theta \times X' \rightarrow Y$  to minimize the multilabel classification loss:

$$L_C = (y, C(S_i(z))). \quad (4)$$

We couple the two losses via a new hyperparameter  $\lambda$ , and the interaction phase translates into the optimization:

$$\min_{\theta_{S_i}, \theta_D, \theta_C} \mathbb{E}_{z, y \in Z} [\lambda L_R + (1 - \lambda) L_C]. \quad (5)$$

### 3.2. Class Activation Map Generation

In this stage, we use the classifier  $C$  from the previous stage to generate simple and interpretable activation maps for each class  $c \in K$ , to select relevant or non-relevant coordinates for classification as  $c$ .

First, based on the classifier  $C$ , we define class activation maps  $M_c \subseteq \{0, 1\}^{H \times W}$  for every sparse vector  $z^s \in Z^s$  and every class  $c \in K$ , where  $M_c \approx 1$  defines which spatial coordinates of the sparse vectors are important for classification as class  $c$ . The CAMs are generated with EigenCAM [25], which is adapted to the task because it does not require correct classification to generate attention maps.

Next, following [8], we separate class-specific and class-generic features by a threshold method, as follows. With upper threshold value  $\tau_h < 1$  and lower value  $\tau_l > 0$ , set

$$M_c^S := \text{sgn}(M_c - \tau_h) \quad M_c^G := 1 - \text{sgn}(\tau_l - M_c), \quad (6)$$

whose interpretation is that  $M_c^S$  contains coordinates of  $z^s$  relevant to classify the vector as  $c$ , while  $M_c^G$  contains non-relevant coordinates.

### 3.3. Inference

In this stage, we generate new samples from tail classes. Let  $H$  and  $T$  be sets of head and tail classes. We first find the head class  $H_c$  of highest confusion for a given tail class, provided by the highest estimated value index in the  $C$  classifier prediction for the head classes. We then sample an element for each tail class example from their highest confusion head class.

A possible naive approach would have been to create a fused vector by adding the class-specific tail features with the class generic head features, using their respective masks. However, note that the class-specific mask for the first vector and the class-generic mask for the second vector might overlap or leave empty spaces in the final vector. For these uncovered and overlapping parts of the vectors, it is not clear how

to update their values in an interpretable manner. To solve this, whenever both masks  $M_c^S, M_c^G$  are 0 or both are 1, we choose which vector to use at random, introducing a random mask  $M^R$ . The values of the maps also need to be broadcasted across  $C'$  channels as follows: for  $M \in \{0, 1\}^{H \times W}$ , let  $\overline{M} \in \{0, 1\}^{H \times W \times C'}$  be the tensor given by repeating  $C'$  times the values of  $M$ . Then, define

$$M^R \sim [\text{Unif}(\{0, 1\})]^{H \times W}, \quad (7)$$

$$z_R^s := z_t^s \odot \overline{M^R} + z_h^s (\mathbf{1} - \overline{M^R}), \quad (8)$$

$$\tilde{z}_R^s := z_R^s \odot (\mathbf{1} - \overline{M_c^G} - \overline{M_c^S} + 2\overline{M_c^G} \odot \overline{M_c^S}), \quad (9)$$

$$\tilde{z}_t^s := z_t^s \odot (\overline{M_c^S} \odot (\mathbf{1} - \overline{M_c^G})), \quad (10)$$

$$\tilde{z}_h^s := z_h^s \odot (\overline{M_c^G} \odot (\mathbf{1} - \overline{M_c^S})), \quad (11)$$

$$z_F^s := \tilde{z}_h^s + \tilde{z}_t^s + \tilde{z}_R^s. \quad (12)$$

Having defined the sparse fusion vector  $z_F^s$ , we translate it back into the base latent space  $Z$  using the decoder network  $D$  trained in stage 1:

$$z_F := D(z_F^s) \quad (13)$$

Our results suggest that, for fast data augmentation, this approach tends to suffice; however, for high-fidelity image generation, an extra step is added, as this vector does not necessarily lie within the pre-trained VAE domain. Using the frozen pre-trained Unet  $U^*$  from the Stable Diffusion Model, conditioned on the class names associated with the tail class image, the vector can be translated into the VAE domain. We take  $N/d$  denoising steps, where  $N$  is the number of inference steps in the original Stable Diffusion model, and  $d \in [1, N]$ .

## 4. Results

The method was tested by generating tail classes of a sampled version of MIMIC-CXR-LT 2023 [13, 16], a multiclass multiclass long-tailed dataset of chest X-rays. Our reduced sample version has only five head classes and five tail classes.

We compare our method with SMOTE [7] and RoentGEN [6], which is used as our Stable Diffusion Model, on image generation quality using FID [15]. We also evaluate using no extra inference steps, one step, and five. We train a Densenet121 [17] model using each augmented dataset to classify our sample train set and test it on a separate test set from the same challenge. We evaluate the Mean Average Precision for head and tail classes for each described method and compare them with a classifier trained on our unaugmented sample set. As Tab. 1 shows, we obtain lower image quality than the other methods when using 0 and 1 inference steps. However, we get a lower FID with only five inference steps than RoentGEN using 75 inference steps. Unexpectedly, all tested augmentation methods have worse results when used to augment a classifier than the baseline.

Model	Avg Tail FID↓	Head mAP↑	Tail mAP↑
Baseline	-	0.618	0.155
SMOTE	171.864	0.578	0.151
RoentGEN@75	138.963	<b>0.618</b>	<b>0.152</b>
Ours@0	191.873	0.607	<b>0.152</b>
Ours@1	191.646	0.595	0.143
Ours@5	<b>130.110</b>	0.595	0.144

Table 1. Results measuring FID [15] and Mean Average Precision for tail and head classes. We use 75 inference steps for RoentGEN, following [6], and 0, 1, and 5 steps for our method.

One possible explanation for this behavior is how labels are assigned to new data: As a multi-labeled classification problem, our approach assigns labels from the corresponding head and tail images to the fused latent vectors. However, this may introduce biases that lead to misclassification of real samples. Furthermore, despite the good FID performance, our method performs noticeably worse in the classification task (mAP). This might be a consequence of the diffusion process, which does not guarantee maintaining the existing labels in the fused latent vector.

## 5. Conclusion

In this work, we present a novel method for data augmentation and data generation on long-tailed datasets. By leveraging pre-trained Latent Diffusion Models, compositional learning, and saliency methods, we generate new examples of underrepresented classes. We give a detailed mathematical description of our method and run experiments on image generation and data augmentation in the medical domain for multi-label classification using a small subset of MIMIC-CXR-LT [13, 16].

Using the Latent Diffusion Model RoentGEN [6] to run five inference steps on our generated vectors, we obtain competitive results in image generation quality. Counter-intuitively, we found that using more inference steps negatively impact the downstream image classification when using our augmented data. This might be a consequence of our approach not necessarily maintaining data labels throughout the diffusion process.

In future steps, we plan to experiment with larger datasets and different techniques for assigning labels and compare our results with a broader range of generation and augmentation methods across a more comprehensive range of tasks.

## 6. Acknowledgements

This work was supported by ANID Chile, Fondecyt Regular grant 1231724, as well research centers of excellence with code FB210017 (Basal CENIA), ICN2021.004 (Millenium iHealth), and ICN17.002 (Millenium IMFD).

## References

- [1] Sajjad Abbasi, Mohsen Hajabdollahi, Nader Karimi, and Shadrokh Samavi. Modeling teacher-student techniques in deep neural networks for knowledge distillation, 2019. [2](#)
- [2] Wenxuan Bao, Francesco Pittaluga, Vijay Kumar B G, and Vincent Bindschaedler. Dp-mix: Mixup-based data augmentation for differentially private learning, 2023. [1](#)
- [3] Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. On the dangers of stochastic parrots: Can language models be too big? . In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, page 610–623, New York, NY, USA, 2021. Association for Computing Machinery. [1](#)
- [4] Aaron Beppu and Thomas Griffiths. Iterated learning and the cultural ratchet. In *Proceedings of the annual meeting of the cognitive science society*, 2009. [1](#)
- [5] Angona Biswas, MD Abdullah Al Nasim, Al Imran, Anika Tabassum Sejuty, Fabliha Fairouz, Sai Puppala, and Sajedul Talukder. Generative adversarial networks for data augmentation, 2023. [2](#)
- [6] Pierre Chambon, Christian Bluethgen, Jean-Benoit Delbrouck, Rogier Van der Sluijs, Małgorzata Połacin, Juan Manuel Zambrano Chaves, Tanishq Mathew Abraham, Shivanshu Purohit, Curtis P. Langlotz, and Akshay Chaudhari. Roentgen: Vision-language foundation model for chest x-ray generation, 2022. [4](#)
- [7] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16: 321–357, 2002. [1](#), [2](#), [4](#)
- [8] Peng Chu, Xiao Bian, Shaopeng Liu, and Haibin Ling. Feature space augmentation for long-tailed data, 2020. [2](#), [3](#)
- [9] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis, 2021. [2](#)
- [10] Nanyi Fei, Zhiwu Lu, Yizhao Gao, Guoxing Yang, Yuqi Huo, Jingyuan Wen, Haoyu Lu, Ruihua Song, Xin Gao, Tao Xi-ang, Hao Sun, and Ji-Rong Wen. Towards artificial general intelligence via a multimodal foundation model. *Nature Communications*, 13(1):3094, 2022. [1](#)
- [11] Fabio Garcea, Alessio Serra, Fabrizio Lamberti, and Lia Morra. Data augmentation for medical imaging: A systematic literature review. *Computers in Biology and Medicine*, 152: 106391, 2023. [1](#)
- [12] Evgin Goceri. Medical image data augmentation: techniques, comparisons and interpretations. *Artificial Intelligence Review*, 56(11):12561–12605, 2023. [1](#)
- [13] Ary L. Goldberger, Luis A. Amaral, Leon Glass, Jeffrey M. Hausdorff, Plamen Ch. Ivanov, Roger G. Mark, Joseph E. Mietus, George B. Moody, Chung-Kang Peng, and H. Eugene Stanley. Physiobank, physiotoolkit, and physionet. *Circulation*, 101(23), 2000. [4](#)
- [14] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014. [2](#)
- [15] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium, 2018. [4](#)
- [16] Gregory Holste, Song Wang, Ajay Jaiswal, Yuzhe Yang, Mingquan Lin, Yifan Peng, and Atlas Wang. *CXR-LT: Multi-Label Long-Tailed Classification on Chest X-Rays (version 1.1.0)*, 2023. [4](#)
- [17] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks, 2018. [4](#)
- [18] Xiaoyu Jiang and Zhiqiang Ge. Data augmentation classifier for imbalanced fault classification. *IEEE Transactions on Automation Science and Engineering*, 18(3):1206–1217, 2021. [1](#)
- [19] Simon Kirby, Hannah Cornish, and Kenny Smith. Cumulative cultural evolution in the laboratory: An experimental approach to the origins of structure in human language. *Proceedings of the National Academy of Sciences*, 105(31): 10681–10686, 2008. [1](#)
- [20] Simon Kirby, Monica Tamariz, Hannah Cornish, and Kenny Smith. Compression and communication in the cultural evolution of linguistic structure. *Cognition*, 141:87–102, 2015. [1](#)
- [21] Samuel Lavoie, Christos Tsirigotis, Max Schwarzer, Ankit Vani, Michael Noukhovitch, Kenji Kawaguchi, and Aaron Courville. Simplicial embeddings in self-supervised learning and downstream classification, 2022. [2](#)
- [22] Chunyuan Li, Zhe Gan, Zhengyuan Yang, Jianwei Yang, Linjie Li, Lijuan Wang, and Jianfeng Gao. Multimodal foundation models: From specialists to general-purpose assistants. *arXiv preprint arXiv:2309.10020*, 1(2):2, 2023. [1](#)
- [23] Giovanni Mariani, Florian Scheidegger, Roxana Istrate, Costas Bekas, and Cristiano Malossi. Bagan: Data augmentation with balancing gan, 2018. [1](#)
- [24] Francisco J. Moreno-Barea, José M. Jerez, and Leonardo Franco. Improving classification accuracy using data augmentation on small data sets. *Expert Systems with Applications*, 161:113696, 2020. [1](#)
- [25] Mohammed Bany Muhammad and Mohammed Yeasin. Eigen-cam: Class activation map using principal components. In *2020 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2020. [3](#)
- [26] Walter H. L. Pinaya, Petru-Daniel Tudosiu, Jessica Dafflon, Pedro F. Da Costa, Virginia Fernandez, Parashkev Nachev, Sebastien Ourselin, and M. Jorge Cardoso. Brain imaging generation with latent diffusion models. In *Deep Generative Models*, pages 117–126, Cham, 2022. Springer Nature Switzerland. [1](#)
- [27] Yiming Qin, Huangjie Zheng, Jiangchao Yao, Mingyuan Zhou, and Ya Zhang. Class-balancing diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18434–18443, 2023. [2](#)
- [28] Yi Ren, Shangmin Guo, Matthieu Labeau, Shay B Cohen, and Simon Kirby. Compositional languages emerge in a neural iterated learning model. In *International Conference on Learning Representations*, 2019. [1](#)

- [29] Yi Ren, Samuel Lavoie, Mikhail Galkin, Danica J. Sutherland, and Aaron Courville. Improving compositional generalization using iterated learning and simplicial embeddings, 2023. [1](#)
- [30] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022. [1](#), [2](#)
- [31] Vimalraj S Spelman and R Porkodi. A review on handling imbalanced data. In *2018 International Conference on Current Trends towards Converging Technologies (ICCTCT)*, pages 1–11, 2018. [1](#)
- [32] Brandon Trabucco, Kyle Doherty, Max Gurinas, and Ruslan Salakhutdinov. Effective data augmentation with diffusion models, 2023. [2](#)
- [33] Shengqiong Wu, Hao Fei, Leigang Qu, Wei Ji, and Tat-Seng Chua. Next-gpt: Any-to-any multimodal llm, 2023. [1](#)
- [34] Suorong Yang, Weikang Xiao, Mengchen Zhang, Suhan Guo, Jian Zhao, and Furao Shen. Image data augmentation for deep learning: A survey, 2023. [2](#)
- [35] Sangdoon Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features, 2019. [1](#), [2](#)
- [36] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization, 2018. [2](#)
- [37] Zhaoyu Zhang, Mengyan Li, and Jun Yu. On the convergence and mode collapse of gan. In *SIGGRAPH Asia 2018 Technical Briefs*, New York, NY, USA, 2018. Association for Computing Machinery. [2](#)
- [38] Xu Zheng, Tejo Chalasani, Koustav Ghosal, Sebastian Lutz, and Aljosa Smolic. Stada: Style transfer as data augmentation. In *Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*. SCITEPRESS - Science and Technology Publications, 2019. [1](#)