Mixup for Robust Image Classification - Application in Continuously Transitioning Industrial Sprays

Anonymous Author(s)
Affiliation
Address
email

Abstract

Image classification with deep neural networks has seen a surge of technological breakthroughs with promising applications in areas such as face recognition, object detection, etc. However, in engineering problems, e.g. high-speed imaging of engine fuel injector sprays, deep neural networks face a fundamental challenge - the availability of adequate and diverse data. Typically, only hundreds or thousands of samples are available for training. In addition, the transition between different spray classes is a "continuum" and requires a high level of domain expertise to label images accurately. Thus, this work leverages pre-trained Neural Network models to build classifiers and employed Mixup to systematically deal with the data scarcity and ambiguous class boundaries found in industrial spray applications. Comparing to traditional data augmentation methods, Mixup that linear interpolates different classes naturally aligns with the continuous transition between different classes in spray applications. Results also show that Mixup can train a more accurate and robust deep neural network classifier with only hundreds samples.

1 Introduction

Deep Convolutional Neural Networks (CNNs) have led to a series of technological breakthroughs in computer vision applications [9]. Among the most important factors that contributed to this tremendous success are publicly available, large, high-quality datasets such as ImageNet [2], CIFAR [6], CelebA [7]. However, pre-trained deep CNNs face unique challenges when directly applied to problems in scientific domains where datasets are not only very different but also scarce and require expert domain knowledge for accurate labeling and annotation. The data scarcity, combined with class overlap, naturally leads to overfitting and poor model performance.

Although many strategies such as less complex models, data augmentation, dropout [8], and regularization can be used to prevent overfitting, their effectiveness in small datasets that contain only hundreds of training samples can be limited. In addition, these methods often lack physical interpretation, as is critical for model acceptance in scientific applications.

In this work, we aim to apply a pre-trained deep CNNs to build classifiers for such an engineering problem where high speed images of engine fuel sprays need to be classified into different categories. Those images are collected using Mie scattering and are used to help engine experts to understand the mechanisms of spray breakup and mixture formation. It is generally accepted that, depending on the macroscopic features of spray images, spray morphology can be classified into three regimes - no collapse, transitional, and collapse. It is also widely accepted that spray collapse should be avoided at all cost since it can worse engine performance and increase combustion emissions. Unfortunately, the spray collapse phenomenon is not well understood. The detection of spray collapse has so far remained a manual and time consuming process, and is highly relied on domain experts.

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

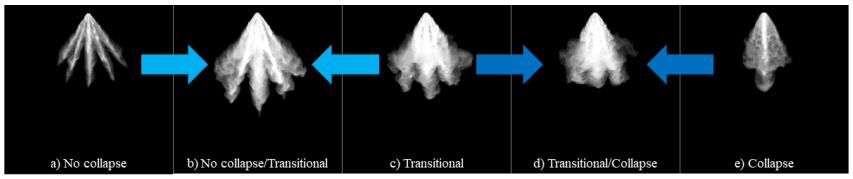

Figure 1: Continuous transition between different spray morphology classes: (a), (c) and (e) show distinct features of the corresponding classes, (b) and (d) show "blending" of two neighbor classes.

To build accurate and robust classifier for such applications, there are two main challenges. First, the transition between different spray classes is not sharp or even perceptibly different, as is shown in Figure 1, where fuel sprays injected into a constant volume chamber were recorded with Mie-Scattering imaging. Three spray morphology classes - no collapse, transitional, and collapse are shown in (a), (c), and (e), respectively. It is clear that the transition between spray morphologyies is a "continuum" and some spray images exhibit features from two different classes, as shown in (b) and (d). These mixed images, represent class overlap and are difficult to label even for domain experts. For example, (b) can be labeled as no collapse, but its close variant from the next camera frame may be labeled as transitional. This might unintentionally lead to some "corrupt labels" in the training set, and results in poor model performance despite the remarkable capability of deep learning.

The second challenge is small data size. Due to limited testing resources and testing/labeling manpower, initially only 900 images are collected and labeled from lab. The combination of a small data set and "corrupt labels" impact data quality and balance, thereby affecting model performance.

To overcome those challenges in a physically meaningful manner, the Mixup approach proposed by Zhang *et al.* [10] is employed for data augmentation. Our study suggests that a convex linear interpolation of Mixup naturally aligns with the continuous class transition observed in the spray dataset. Depending on whether two candidate images belong to a same class or not, Mixup can either function as a data augmentation technique that reduces overfitting or a label smoothing technique that mimics the continuous class transitions. Our study also shows that though traditional data augmentation can mitigate overfitting to some degree, it does not bring additional benefits and may negate the performance boost if stacked with Mixup.

The key contributions of this work are listed below:
1. Proposed an explainable training technique to build robust and accurate deep CNN classifier for engineering applications that have limited data, exist continuum between classes and 'corrupt labels'.
2. Showcased that Mixup worked well with small datasets (900 images collected from continuously transitioning industrial sprays). This benefit has not been discussed in the original work by Zhang *et al.* [10], nor such engineering applications have not yet been discussed in literature to our knowledge.

2 Data, methods and results

2.1 Introduction of training data and baseline model development

In this study, the entire dataset is composed of proprietary spray images collected from internal spray lab. We aim to built classifier that is capable of classifying spray structures over various testing conditions, injection timings, and experimental parameters. This allows using this model for spray images collected from multiple sources, covering a wide range of operating conditions, injector geometries, and measurement events. In addition, to cover the entire injection event, an additional class named "Pre/Post" (short for pre-injection and post-injection) is added to the dataset so the final model can take any input from the Mie Scattering without human attention.

In total, there are 878 images collected- 173 pre/post, 199 no collapse, 241 transitional, and 265 collapse. Here, 75% of data is set for training, 15% for validation, and 10% for testing. Examples of spray images in each class are shown in Appendix B. Since the data were collected from multiple sources, they are not necessarily the same size. In addition, some images may carry auxiliary boundary lines and/or text annotations processed by Mie Scattering tool. Those lines were remained during the manual labeling process and we anticipate the model would either treat them as noise or recognize them as useful features.

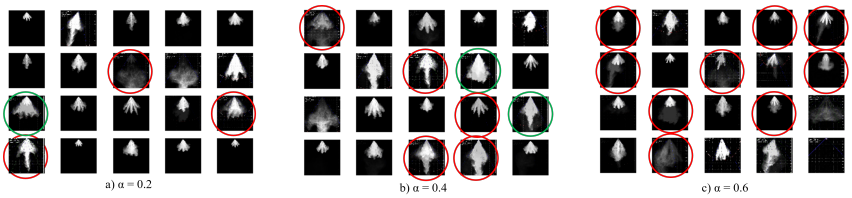

Figure 2: Examples of training images after Mixup data generator. Note that both the frequency and strength of interpolation increase as α increases from 0.2 to 0.6.

Given small dataset, the most feasible starting point is transfer learning by reusing lower layers of pre-trained models. This study employed TensorFlow version 2.4.1 [1]. Among many available pre-trained models on Keras, ResNet [4] family is employed after an internal evaluation. All images were pre-processed as 224×224 pixel as expected by ResNet. Finally, with 5-fold cross validation, the ResNet50 is selected out of 7 ResNet model variants due to its best accuracy and least overfitting. Detailed model architecture and performance comparison are shown in Appendix C.

2.2 Mixup and data augmentation

Mixup is a simple and data-agnostic method proposed by Zhang *et al.* [10]. Given two images, it can either select one input image as the output or blend both input images and generate a linearly interpolated image. The detailed Mixup introduction is shown in Appendix D. It is noted that there are many choices of Mixup implementations as suggested by Zhang *et al.* [10], while this work implemented the vanilla version

Figure 2 shows some examples after Mixup with different α values. Note that we only tested α up to 0.6 as Zhang *et al.* [10] found Mixup only improved performance over traditional data augmentation with $\alpha < 0.4$. For larger α values, Mixup leads to underfitting. At $\alpha = 0.2$, four (circled in Figure 2,a)) out of 20 randomly selected images are visually discernible as being augmented by Mixup. Among those four images, one of them was interpolated between two images belonging to a same class as highlighted by a green circle. The resulting image expands data distribution of that class and hence Mixup acts as a data augmentation tool. The other three (circled in red) were interpolated between two different classes, the resulting labels are no longer one-hot vectors. This is considered critical to our application as those cross-class augmented images mimic the smooth transition observed in the spray dataset. As α increases, the interpolation becomes stronger, and more images can be observed as being interpolated.

Data augmentation is widely used to combat overfitting of deep neural networks. However, among many choices of data augmentation methods such as shifting, rotation, flip, zooming, one must be careful as the effectiveness of data augmentation is dataset dependent and use of domain knowledge is usually needed. Although Mixup was found to improve performance over data augmentation [10], they can easily work together. In this work, Mixup is performed before any alternate data augmentation since it guarantees that there will be only one injector tip located on the top of each augmented spray image. If image augmentation such as rotation and shifting are applied first, then the Mixup augmented image may have two injector tips, which violates physics pollutes training data.

To compare different data augmentation techniques, we first fine tuned the baseline model-ResNet50 with only tranditional data augmentation (without Mixup). Again, only the last block of ResNet50 (i.e., *conv5 block3*) is retrained with our training data. To align with our experiment setup, the following traditional data augmentation is applied: random rotation up to 20 degrees since our injector can only be installed at a slightly angled position; random shifting up to 20% as the spray image is always centered at the camera window. Horizontal flip but not vertical flip since it gives an upside-down image that makes no physical sense. Brightness, saturation, hue, or contrast adjustments are excluded given Mie Scattering imaging produces consistent images without much distortion.

The ensemble-averaged training-validation accuracy gaps over the last 50 epochs and the test accuracy for each final model are reported in Table 1. Comparing to the baseline model without any data augmentation, all single data augmentations lead to improved generalization. Also, combining two or three augmentation methods do not bring additional benefits.

Table 1: Training-validation gaps averaged over the last 50 epochs for ResNet50 with different data augmentation and Mixup methods

Model	Gap	Test Acc.
Baseline	4.9%	94.8%
Rot 10	3.9%	96.5%
Rot 20	4.3%	96.5%
Shift 0.1	1.6%	96.5%
Shift 0.2	**1.3%**	**96.5%**
H. Flip	3.5%	97.7%
Shift 0.2+H.Flip	**1.3%**	**96.5%**
Rot 10+H.Flip	5.3%	96.5%
Shift 0.2+Rot10	2.2%	96.5%
Shift0.1/Rot10/H.Flip	2.8%	96.5%

Model+Mixup	Train Acc	Gap	Test Acc.
Baseline	99.7%	4.9%	94.8%
$\alpha = 0.2$	**96.5%**	**0.9%**	**98%**
$\alpha = 0.4$	95.1%	0.3%	98%
$\alpha = 0.6$	94.3%	0.3%	98%

Next, we evaluated Mixup without data augmentation. As shown on Table 1, all Mixup tests lead to improved performance over data augmentation. The training-validation accuracy gaps are below 1.0% with all three α values. Larger α leads to better generalization, but under-fits the training set as evidenced by the decreasing ensemble-averaged training accuracies. This finding is consistent with the work by Zhang *et al.* [10], though they used much bigger datasets (ImageNet).

In addition, Zhang *et al.* [10] showed Mixup can mitigate the memorization of "corrupted labels" by replacing up to 80% images with "corrupted" labeled images in the CIFAR dataset. In our application, "corrupted labels" are unintentionally introduced because of blurry boundaries between continuous class transitions. We hypothesize that Mixup can help combat the memorization of those "corrupted labels" as well. For example, if a "collapse/transitional" image is labeled as "collapse" and its neighbor from the next camera frame is labeled as "transitional", then a neural network without Mixup would learn to memorize those labels and over-fit the data. On the other hand, all training images are generated from Mixup (although not all of them are interpolated), so those two images may produce a new training image with an interpolated label. This is equivalent to labeling the newly generated image as "collapse/transitional", thereby reduces the memorization of "corrupted labels". From the physics point of view, this interpolation aligns with the actual transition in the experiments.

At last, we performed a test with Mixup ($\alpha = 0.2$) followed by the "best-practice" data augmentation i.e., shifting by 20% plus horizontal flip. The training accuracy drops to 93.7% and the test accuracy is lower than all Mixup-only tests. This indicates additional data augmentation may add excessive regularization.

3 Conclusion

Data scarcity is one of main challenges that large deep neural networks face when applied to scientific problems. In this paper, we showcased a robust and accurate training method for an industrial engine injector spray classification problem where the transition between classes is a "continuum". With Mixup, a deep CNN classifier (ResNet50) can be built with only hundreds images with 98% accuracy on a real-world spray dataset. The study showed that Mixup improved performance over data augmentation methods, and can provide benefit of reducing the memorization of "corrupted labels" that unintentionally introduced during manual labeling. As we understand, the linear interpolation of both data and labels also agrees with the "continuum" nature of class transitions in injector sprays.

As future work, we are interested in incorporating simulation data from high-fidelity computational fluid dynamics (CFD) tools into the experimental dataset and continue to explore the properties of Mixup and variants. We also wish to understand the impact of Mixup on assisting these models to grasp knowledge that embedded differently in simulation and experiment datasets.

References

[1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[2] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[3] Todd D Fansler and Scott E Parrish. Spray measurement technology: a review. *Measurement Science and Technology*, 26(1):012002, dec 2014.

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[5] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014. cite arxiv:1412.6980Comment: Published as a conference paper at the 3rd International Conference for Learning Representations, San Diego, 2015.

[6] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

[7] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

[8] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958, 2014.

[9] Athanasios Voulodimos, Nikolaos Doulamis, Anastasios Doulamis, and Eftychios Protopapadakis. Deep learning for computer vision: A brief review. *Computational Intelligence and Neuroscience*, 2018, 2018.

[10] Hongyi Zhang, Moustapha Cissé, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *CoRR*, abs/1710.09412, 2017.

A Background of fuel injection spray applications

Although the automotive industry appears to be on the verge of transitioning to electric vehicles, the vast majority of passenger cars and commercial vehicles on the road are still powered by Internal Combustion Engines (ICE). ICEs are complex systems that convert the energy stored in the hydrocarbon bonds of chemical compounds in fossil fuels to mechanical energy used to power vehicles. A basic ICE works like follows: the fuel system injects gasoline (or diesel) into the intake port (or combustion chamber) via high-pressure fuel injectors, the fuel then evaporates and mixes with the induced fresh air, the fuel-air mixture ignites by spark plug(or auto-ignites by compression), creating a high temperature, high pressure explosion that provides motive power.

The fuel injection and mixing with ambient air is one of the most important factors impacting engine performance (power) as well as engine emissions (such as CO2, and NO). Therefore, extensive studies have been carried out in the combustion domain to optimize fuel sprays for maximizing the power to emissions index. Among the many measurement techniques used in spray testing, Mie scattering remains one of the prominent imaging methods for spray visualization [3]. Mie scattering uses a light source (e.g., laser) and a camera to record the macroscopic spray development inside a quiescent chamber filled with pressurized air. Light is elastically scattered by fuel droplets similar to or larger than the wavelength of the incident light. The signal collected by a camera is proportional to the integral of cubic of droplet diameters along the line of sight. Examples of Mie scattering imaging are shown in Figure 1, where the dark color is background and the light color are liquid sprays. The grayscale in each image roughly indicates the intensity of liquid volume fractions.

One of many insights gained from Mie scattering is spray morphology which helps engine experts to understand the mechanisms of spray breakup and mixture formation. However, spray development is a complicated process affected by many factors such as fuel volatility, fuel temperature, injector geometry, ambient conditions, and turbulence. Despite the continuous efforts in academia and industry, spray development is still not fully understood. It is generally accepted that, depending on the macroscopic features of spray images, spray morphology can be classified into three regimes, namely no collapse, transitional, and collapse. Characteristics of each regime are listed blow:

- No collapse regime: Characterized by visually discernible narrow plumes or branches. The separations of spray plumes occur very closely to the injector tip (located on the top of the image). This is the desirable spray pattern.
- Transitional regime: Characterized by wide spray plumes that begin to interact with each other. The separations of plumes move downstream and the exact locations becomes hard to visually discern. The spray structure still resembles a cone shape.
- Spray collapse regime: Characterized by one single prolonged central plume. There may be one or two discernible spray plumes but the majority of them are coalesced.

It is widely accepted that spray collapse should be avoided at all cost since it can lead to spray impingement, reduced total surface area for fuel-air mixing, and poor atomization. The consequences of those issues are worsened engine performance and increased combustion emissions. Unfortunately, the spray collapse phenomenon is not well understood. The detection of spray collapse has so far remained a manual process requiring many hours of subjective evaluations by domain experts. This challenge is the main motivation to use deep convolution neural networks for automating robust classification of spray morphology and detection of spray collapse.

B Example of training data

Examples of spray images in each class are shown in Figure 3.

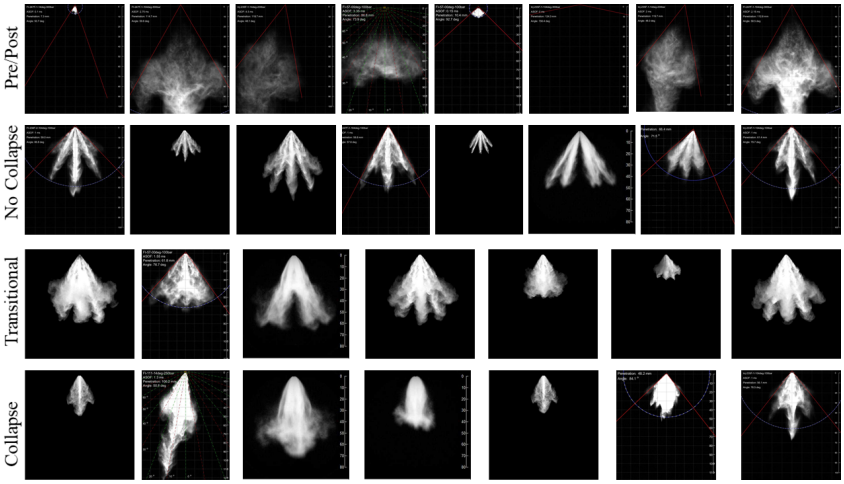

Figure 3: Examples of dataset used for training deep CNN models. An additional class, namely "Pre/Post" was added to the dataset to cover the entire injection event. Note that some examples shown in this figure have text annotations, solid red or dotted white boundary lines from the processing tool of Mie Scattering. During the data collection and labeling process, we did not remove these "noises" for the sake of training a robust model.

C Baseline model network architecture

Given a small dataset, the most feasible starting point is transfer learning by reusing the lower layers of pre-trained models. In this study, TensorFlow version 2.4.1 [1] is used. Among many available pre-trained models on Keras, ResNet [4] family is employed after an internal evaluation. All the images were pre-processed as 224×224 pixel images as expected by ResNet. In addition, the grayscale images were loaded in Red, Green, and Blue channels.

To select a suitable model for fine tuning, 5-fold cross validation is performed using all 878 available spray images. Seven variants of ResNet models were tested. Note that we implemented ResNet34 from scratch since it is not available from Keras. For the remaining six models, we reused all the pre-trained lower and mid layers and only made the last block (namely, *conv5 block3*) trainable. This leaves about 4.4 million trainable parameters for each model except for ResNet34. We also removed the fully connected top layer and replaced it with global average pooling layer, followed by a dense output layer with one neuron per class. For pre-trained ResNet models, we used Adam optimizer [5] and fixed the learning rate, batch size, and total epochs to be 0.001, 32, and 100, respectively. For ResNet34, we used stochastic gradient descent with a learning rate of 0.0001 and momentum of 0.9.

The ensemble-averaged test accuracies and standard deviations are reported in Table 2. ResNet50 outperforms all other models with an impressive test accuracy of 96%, though other models are not far behind. Although ResNet101 and larger models have more representation power than ResNet50, they are prone to overfitting with small training datasets and therefore have slightly poor performance on the test set. On the other hand, ResNet34 seems too shallow to learn all necessary low-level and/or high-level features. Given its highest accuracy, ResNet50 is used as the base model for further experiments.

D Mixup introduction

Mixup is a simple and data-agnostic method proposed by Zhang *et al.* [10]. Suppose we have two input image vectors, x_1 and x_2, and their corresponding one-hot label encoding vectors are y_1 and y_2, then the Mixup augmented training image vectors are given by:

Table 2: Ensemble-averaged test accuracies and standard deviations of seven ResNet models for the 5-fold cross-validation. Note that ResNet34 was implemented from scratch while the other six were pre-trained models from Keras.

Model	Average test accuracy	STD
ResNet34	0.9522	0.0142
ResNet50	**0.9602**	0.0196
ResNet50V2	0.9488	0.019
ResNet101	0.9556	0.0213
ResNet101V2	0.9351	0.0284
ResNet152	0.9476	0.018
ResNet152V2	0.9385	0.0149

$$x = \lambda x_1 + (1.0 - \lambda) x_2 \tag{1}$$

$$y = \lambda y_1 + (1.0 - \lambda) y_2 \tag{2}$$

where λ is the interpolation coefficient randomly drawn from the Beta distribution, $\lambda \sim \mathbf{Beta}(\alpha, \alpha)$. As shown in Figure 4, λ approaches 0 or 1.0 as $\alpha \to 0$, this essentially eliminates interpolation and selects one input image as the output. As α increases, a realization of λ would have a higher chance of being close to 0.5, leading to strong blending of both input images.

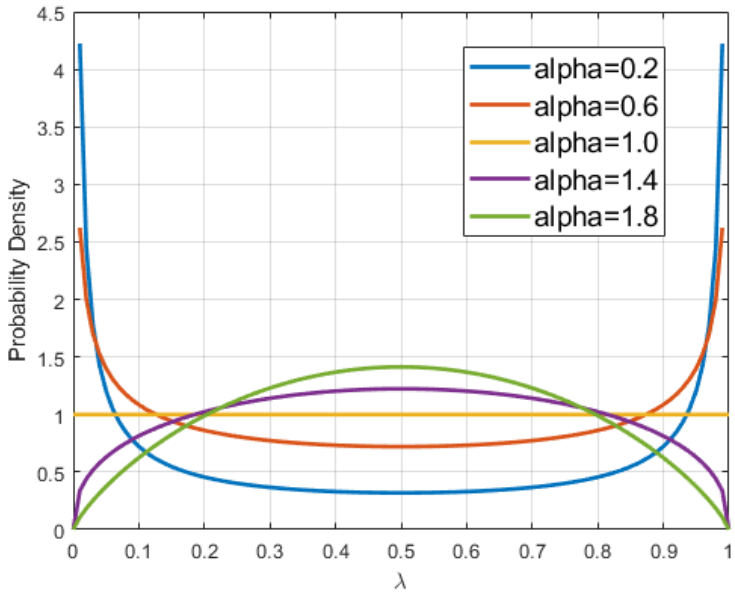

Figure 4: The interpolation coefficient λ is beta-distributed $\lambda \sim \mathbf{Beta}(\alpha, \alpha)$. Note that as α increases, a realization of λ would have a higher chance of being close to 0.5.