Sugestions ML

Thank you for providing detailed information about your data and the model architecture you are using. Based on your input, here are some suggestions for varying configurations in your ensemble training:

1. **Batch Size:** Since you have 2035 images in your training dataset, you can experiment with different batch sizes. Smaller batch sizes can lead to faster convergence and might be suitable for your noisy and distorted data. You can try batch sizes like 16, 32, and 64 to see how it impacts training stability and convergence.
2. **Epochs:** You're currently training for 37 epochs. You can experiment with this value. For example, you could try a lower value like 20 or increase it to 50 to see if the model's performance continues to improve or starts to plateau.
3. **Model Architecture:** Your current model architecture seems like a reasonable starting point. However, since you mentioned your images have some complexity, you might consider experimenting with more complex architectures. You could try deeper CNNs (convolutional neural networks) or even architectures with attention mechanisms, such as the Transformer.
4. **Regularization and Dropout:** You're using L2 regularization and dropout in your model. You can experiment with different regularization strengths and dropout rates. Higher regularization strengths can help mitigate overfitting, especially since your dataset is not very large and the images are noisy.
5. **Learning Rate:** You're using the Adam optimizer, which adapts the learning rate during training. However, you can try experimenting with different initial learning rates, such as 0.001 or 0.0001, to see if it affects convergence.
6. **Data Augmentation:** Since your images are noisy and distorted, data augmentation can be beneficial. Experiment with different augmentation techniques such as rotation, translation, and adding noise to artificially increase your training dataset's diversity.
7. **Ensemble Size:** You're currently using an ensemble of 2 models. You can experiment with different ensemble sizes, like 3, 4, or more models, to see how it affects the ensemble's overall performance.
8. **Metric Tracking:** Track not only accuracy but also other metrics like precision, recall, and F1-score. These metrics can give you a better understanding of the model's performance, especially when dealing with imbalanced classes.
9. **Learning Rate Scheduling:** Implementing learning rate scheduling can help fine-tune the learning process. Experiment with different scheduling strategies like step decay, exponential decay, or cyclic learning rates.