COSE471
Team Project: Design What You Want!

Handwriting-to-Personality Prediction Model Using a CNN

Team 14
DouYoung Kwon 2020130024
Dahyeon Kang 2025951181

Abstract

In this project, we use a convolutional neural network (CNN) to predict one of the five OCEAN personality traits (Agreeableness, Conscientiousness, Extraversion, Neuroticism, or Openness) based on handwriting images. Our study uses 3,432 labeled samples, each augmented with small rotations, translations, and brightness variations to increase diversity. To avoid data leakage, we split writers into an 80% training set and a 20% test set so that no writer's samples appear in both. We implement a custom three-block custom CNN alongside three transfer-learning models (ResNet18, ResNet34, ResNet50), freezing all pretrained layers except the final classification layer. The best result was achieved by ResNet50 with 48.64% accuracy, slightly outperforming our custom CNN (45.76%). These results demonstrate that both custom and pretrained CNNs capture some handwriting cues related to personality, though overall performance remains limited by dataset size and label variability.

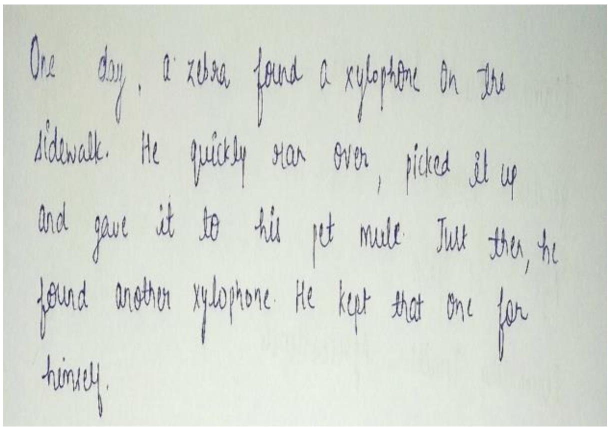

Figure 1. Input example

I. Introduction

In recent years, researchers have been interested in using handwriting to understand personality traits. Instead of looking at things like slant or pressure by hand, deep learning models, especially convolutional neural networks (CNNs), can now find patterns directly from handwriting images. In our project, we build on this idea by comparing a lightweight custom CNN with deeper, pre-trained ResNet variants, we investigate how model capacity interacts with the handwriting data. We aim to test how well the model can predict personality traits and whether including pre-trained ResNet variants helps improve its accuracy.

II. Related Work

Earlier handwriting analysis used simple machine learning models like decision trees or support vector machines. These models needed people to manually choose features like how slanted the writing is or how hard the pen was pressed. They worked moderately well but weren't very accurate. More recent work uses deep learning, especially convolutional neural networks (CNNs), to look at handwriting images directly. These models can learn useful patterns by themselves and usually perform better.

More recent approaches apply CNNs directly to scanned handwriting, sometimes augmenting input channels with edge or gradient information. Also, transfer learning from large image datasets has also shown promise when labeled handwriting samples are scarce. In this project, we explore this transfer learning by fine-tuning the final layer of pre-trained ResNet18, ResNet34, and ResNet50 models, comparing it to the original CNN model.

III. Method

Dataset	Agreeableness	Conscientiousness	Extraversion	Neuroticism	Openness
Augmented Train	720	648	576	864	608
Augmented Test	137	131	112	136	144

Table 1. Augmented Dataset Sample Counts by Personality Trait

In this work, the dataset for this study was sourced from a public Kaggle competition on handwriting-based personality detection, comprising 3,432 images labeled with one of the five OCEAN traits. We randomly split the data into an 80 % training set (2,745 images) and a 20 % validation set (687 images).

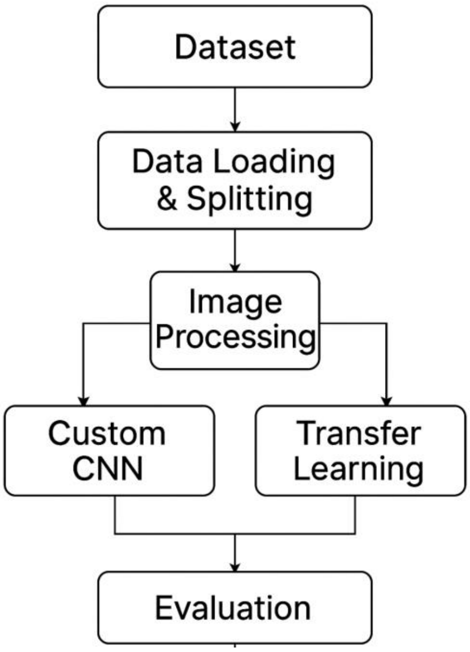

Figure 2. CNN and Transfer Learning Pipeline Flowchart

3.1 First Step

First, each image was resized to 224 × 224 pixels to match the input dimensions required by our CNN. The images were then converted to tensors and normalized using channel-wise means [0.485, 0.456, 0.406] and standard deviations [0.229, 0.224, 0.225], which are standard values from the ImageNet dataset. These preprocessing steps ensured consistent input scaling and facilitated effective training of the model.

3.2 Second Step

Following preprocessing, we trained the model using the cross-entropy loss function and the Adam optimizer with a fixed learning rate of 1e-3. Training was conducted in mini-batches of 32. For transfer learning, we adopted a pre-trained CNN, replacing its final fully connected layer with a new layer for five-way classification while keeping the remaining layers frozen.

3.3 Third Step[1]

We evaluated both a custom CNN and several transfer learning models. Our custom CNN consists of two convolutional blocks (Conv → ReLU → MaxPool) with increasing channel depths of 32 and 64, followed by two fully connected layers: a hidden layer with 128 units and a final five-unit softmax output for classification. This simplified architecture was deliberately chosen to reduce the risk of overfitting given the limited dataset size.

For transfer learning, we employed ResNet18, ResNet34, and ResNet50 architectures pre-trained on ImageNet. In each case, we replaced the final fully connected layer with a new five-unit classifier to match our personality trait classes. All other layers remained frozen to retain the pretrained features.

3.4 Last Step

Lastly, for evaluation, we measured the test loss, overall accuracy, and per-class precision, recall, and F1-score. In addition, we plotted the training and validation loss curves to monitor learning progress and computed a confusion matrix to visualize misclassification patterns across the five personality traits.

IV. Result

Model	Test Accuracy
Custom CNN	45.76%
ResNet18	39.55%
ResNet34	41.97%
ResNet50	48.64%

Table 2. Model Comparison Based on Accuracy

[1] Code and Models are available at https://github.com/douyoung89/COSE471-02-Final-Project

4.1 Custom CNN Performance

Test Accuracy: 0.4576

	precision	recall	f1-score	support
Agreeableness	0.36	0.25	0.29	137
Conscientiousness	0.54	0.63	0.58	131
Extraversion	0.58	0.52	0.55	112
Neuroticism	0.35	0.57	0.44	136
Openness	0.55	0.35	0.43	144
accuracy			0.46	660
macro avg	0.48	0.46	0.46	660
weighted avg	0.47	0.46	0.45	660

Figure 3. Classification Result for the Custom CNN Model

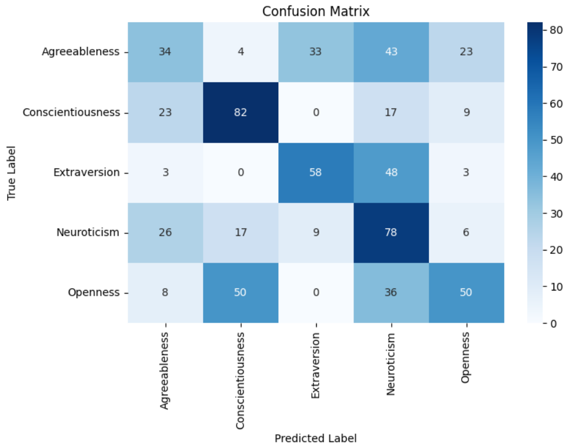

Figure 4. Classification Result for the Custom CNN Model

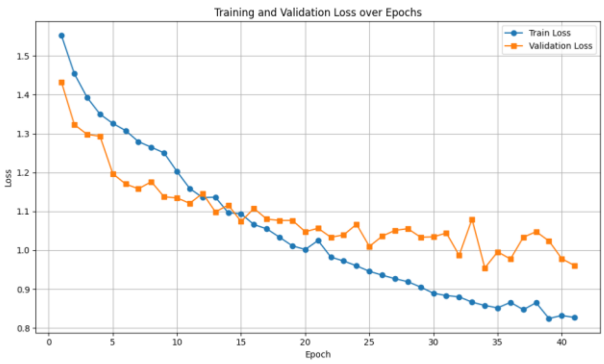

Figure 5. Classification Result for the Custom CNN Model

The custom CNN achieves 45.76% accuracy, and the model achieved the highest F1-scores on Conscientiousness (0.58) and Extraversion (0.55), while performance was lower for Openness (0.43), Neuroticism (0.44), and especially Agreeableness (0.29). This performance indicates that simple convolutional filters can capture some broad stroke and texture patterns related to Conscientiousness and Extraversion, as reflected by higher recall on those classes. However, traits like Agreeableness and Openness remain challenging, suggesting that nuanced handwriting cues for these dimensions are weak or inconsistent.

As shown in Figure 5, the custom CNN was trained for 50 epochs and stopped early after 42 epochs. Both training and validation losses steadily decreased, indicating stable convergence and good generalization.

4.2 ResNet18 Results

Test Accuracy: 0.3955

	precision	recall	f1-score	support
Agreeableness	0.35	0.16	0.22	137
Conscientiousness	0.45	0.54	0.49	131
Extraversion	0.66	0.54	0.59	112
Neuroticism	0.24	0.26	0.25	136
Openness	0.36	0.50	0.42	144
accuracy			0.40	660
macro avg	0.41	0.40	0.39	660
weighted avg	0.40	0.40	0.39	660

Figure 6. Classification Result for ResNet18

ResNet18 records 39.55% accuracy. And the model performed best on Extraversion (0.56), followed by Conscientiousness (0.49), Openness (0.42), and Neuroticism (0.25), while performance on Agreeableness was the weakest with an F1-score of 0.22. Despite having pretrained convolutional filters from ImageNet, its comparatively shallow depth limits its ability to refine representations for handwriting-specific features. The slight drop relative to the custom CNN highlights that generic image filters alone do not guarantee better handwriting trait discrimination.

4.3 ResNet34 Results

Test Accuracy: 0.4197

	precision	recall	f1-score	support
Agreeableness	0.41	0.42	0.42	137
Conscientiousness	0.40	0.64	0.49	131
Extraversion	0.78	0.50	0.61	112
Neuroticism	0.23	0.22	0.23	136
Openness	0.46	0.34	0.39	144
accuracy			0.42	660
macro avg	0.46	0.43	0.43	660
weighted avg	0.44	0.42	0.42	660

Figure 7. Classification Result for ResNet34

With increased depth, ResNet34 attains 41.97% accuracy. And the model showed the best performance on Extraversion (0.61), followed by Conscientiousness (0.49), Agreeableness (0.42), and Openness (0.39), while Neuroticism remained the most difficult to predict with an F1-score of 0.23. The additional layers enable learning more complex feature hierarchies, but improvements are marginal, implying that data volume, rather than model capacity, is the primary bottleneck.

4.4 ResNet50 Results

```
Test Accuracy: 0.4864
                   precision    recall  f1-score   support

    Agreeableness       0.55      0.64      0.59       137
Conscientiousness       0.33      0.78      0.46       131
     Extraversion       0.94      0.78      0.85       112
      Neuroticism       0.18      0.04      0.06       136
         Openness       0.57      0.28      0.37       144

         accuracy                           0.49       660
        macro avg       0.51      0.50      0.47       660
     weighted avg       0.50      0.49      0.45       660
```

Figure 8. Classification Result for ResNet50

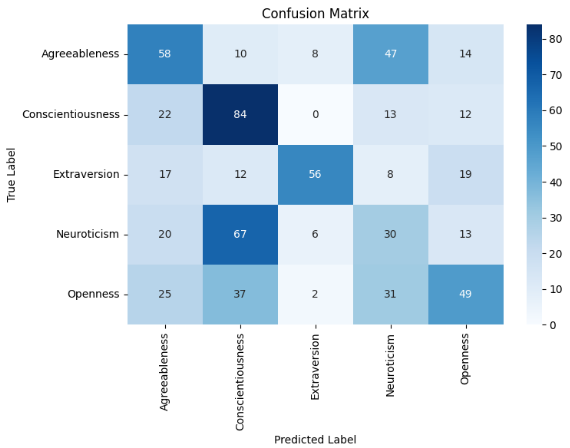

Figure 9. Confusion Matrix for ResNet34

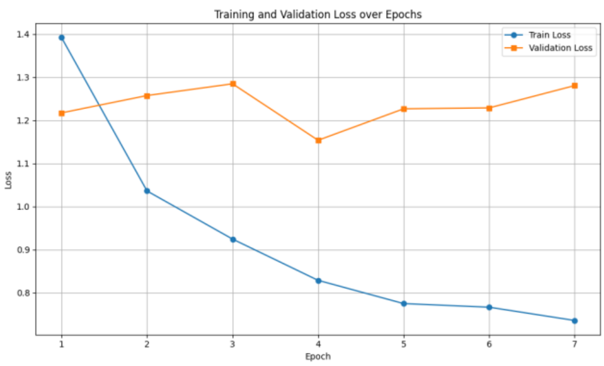

Figure 10. Loss Graph for ResNet34

Among all models, ResNet50 achieves the highest accuracy at 48.64%, slightly outperforming the custom CNN (45.76%). And the model achieved the highest F1-score on Extraversion (0.85), followed by Agreeableness (0.59), Conscientiousness (0.46), and Openness (0.37), while Neuroticism remained the weakest with an F1-score of 0.06. This was the best result among ResNet method. Although this model slightly outperforms shallower variants, the gains are modest. The fact that only the final layer was tuned may limit ResNet50's potential, but fully fine-tuning risks overfitting given the dataset size.

Figure 10 presents the loss curve for the ResNet34 model, which stopped early at epoch 7. While the training loss decreased rapidly, the validation loss rise after over epochs, suggesting early signs of overfitting.

V. Conclusion and Limitations

Our dataset, though augmented, remains limited in both quantity and diversity. Only five trait labels per writer may oversimplify the continuous nature of personality, and augmentation cannot substitute for genuine variability in handwriting styles. Label quality is also a concern: self-reports can be noisy, and matching a writer's dominant trait to discrete categories reduces fidelity.

Model-wise, while our custom CNN captured meaningful stroke and texture patterns, it was ultimately outperformed by ResNet50, the deepest model we tested. This suggests that deeper architectures with pretrained weights can better generalize to handwriting traits, even when only the final classification layer is fine-tuned. However, freezing all but the final layer may still limit the model's ability to adapt fully to handwriting-specific features. On the other hand, fully fine-tuning large models risks overfitting, especially with a limited dataset like ours.

Another major limitation is the fundamental difficulty of the task itself: the scientific relationship between handwriting style and personality traits remains unclear, and it is possible that no consistent visual pattern exists for CNNs to learn. Furthermore, the dataset suffers from limitations in size, diversity, and quality. The number of handwriting samples is relatively small, and the labels are simplified representations of complex personality constructs.

Future improvements will require collecting larger and more diverse datasets of high quality, with richer annotations and possibly additional modalities such as time-based handwriting dynamics from digital devices.

References

[1] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770–778.

[2] Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6), 679–698.

[3] Paszke, A., Gross, S, ... & Chintala, S. (2019). PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, *32*.

[4] Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. (2014). Microsoft COCO: Common objects in context. In *European Conference on Computer Vision* (pp. 740–755). Springer.

[5] Kaggle. (2023). *Handwriting Personality Prediction Dataset*. https://www.kaggle.com/datasets/koolshehzad/personality-detection-using-handwriting.

[6] Kwon, Douyoung. *COSE471-02-Final-Project: Handwriting Personality Classification*. GitHub, 2025, https://github.com/douyoung89/COSE471-02-Final-Project.