Hiragana Recognition with CNN

Ting-Wei Chang

Dataset

- 1405 handwritten Hiragana images

  - 1000 images produced by Matheus Inoue.

  - 405 images produced by Wishyut.

  - 52 Hiragana characters, each character corresponding to 20-29 images.

- Extract labels from filenames

  - Each filename contains the romanji corresponding to each Hiragana, e.g. "kanaA0.jpg" indicates "a" (あ).

  - Regex is used to extract the file name.

- Training data and testing data

  - Data is split into 1124 training data and 281 testing data.
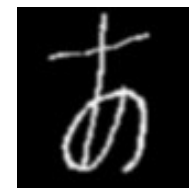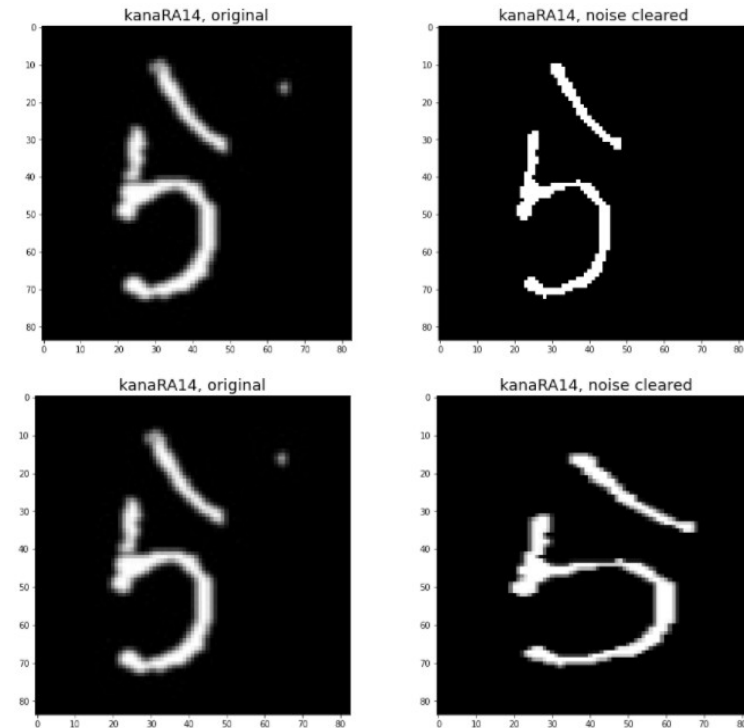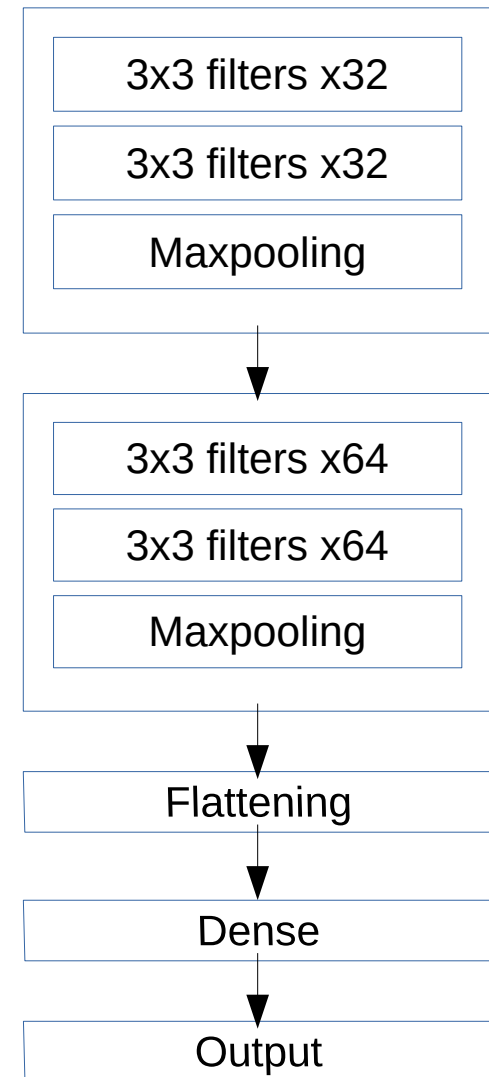
Image Preprocessing

- Noise removal
  - Morphology techniques and thresholds are applied to input images using openCV.

- Reshape images
  - Each image is reshaped into 64x64 pixels and put inside a 84x84 frame.

- Augmentation
  - Small rotations and shears are applied to training images.

CNN Model

- The CNN structure
  - Two convolution blocks
    - Each block contains 2 convolution layers and 1 maxpooling layer
  - One flattening layer
  - One dense layer with 256 neurons
  - Output layer with 52 neurons
    - Corresponds to 52 characters

3x3 filters x32
3x3 filters x32
Maxpooling

3x3 filters x64
3x3 filters x64
Maxpooling

Flattening

Dense

Output

CNN Model

- Training details
  - Use RMSprop as optimizer
  - 16 iterations with early stopping criterion

- Training history
  - Run through 16 iterations
  - Final validation loss: 0.2790

CNN train history
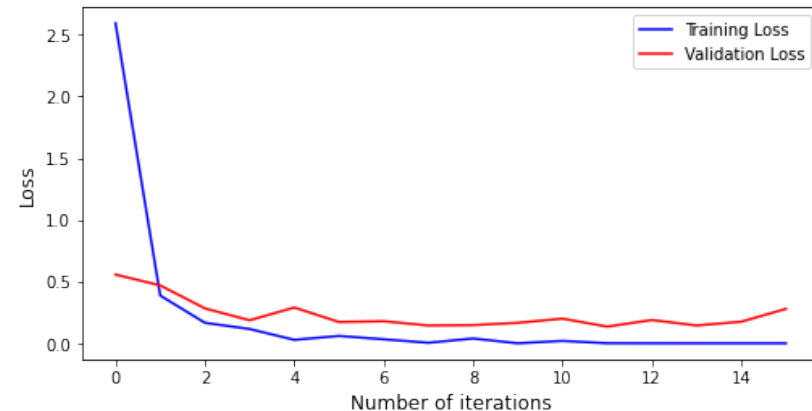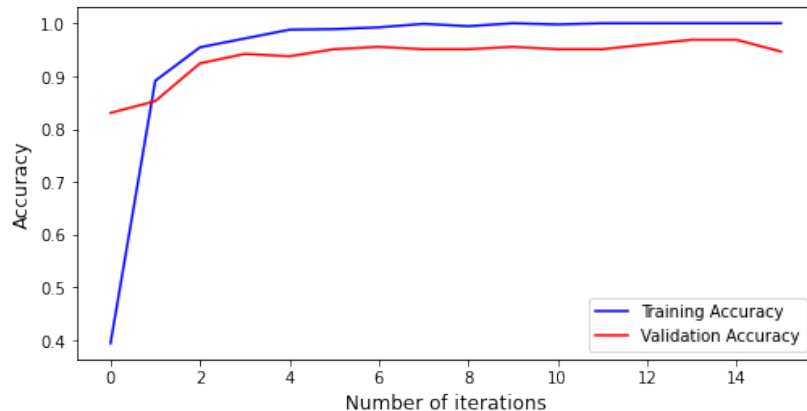
Testing results

- 281 testing data from the original dataset

  - **97%** accuracy

- 92 images of my handwritten Hiragana

  - **83%** accuracy

  - These images are not from the original dataset; in other words, they may have different features.

- Conclusion

  - More diverse handwriting styles, alongside increased data quantity, can lead to a better model with higher accuracy.

Confusion matrix
of the total
281 testing data

Confusion matrix, without normalization

Predicted label

True label

Normalized confusion matrix of the total 281 testing data

Confusion matrix, with normalization