Palm-reading-detection-and-prediction

Presenter: Cheng-Ching Lin
Yu-Hsiang Wang

Cloud Computing and Big Data Analyze Final Project

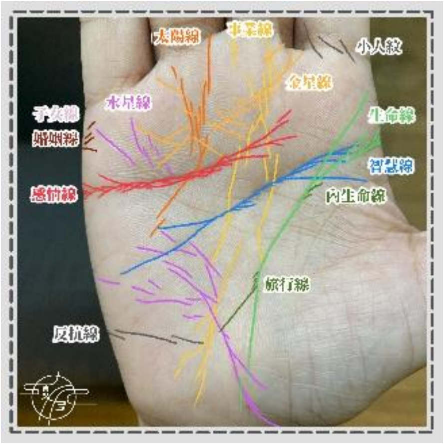
Motivation

- A friend of mine works part-time as a palmistry analyst. He can analyze 2-3 people a day, but at present, more than 1,000 people have made an appointment to ask him to read palmistry. Therefore, we wants to develop an automated process to help him analyze palmistry.

Data Source: h_kshad0w

Palm Data:

Unlabel



Label

Problem Statement

- We think we can to build a deep learning model, which can judge the features of hand lines by itself after inputting a photo of hand, and further give a numerology explanation. The following is the palmistry analysis process:

1. Obtain the original palmistry image.
2. Identify important palm lines.
3. Mark each palm line and capture the features of each line.
4. Give professional analysis and explanation of the meaning of each line pattern according to the palm pattern characteristics.

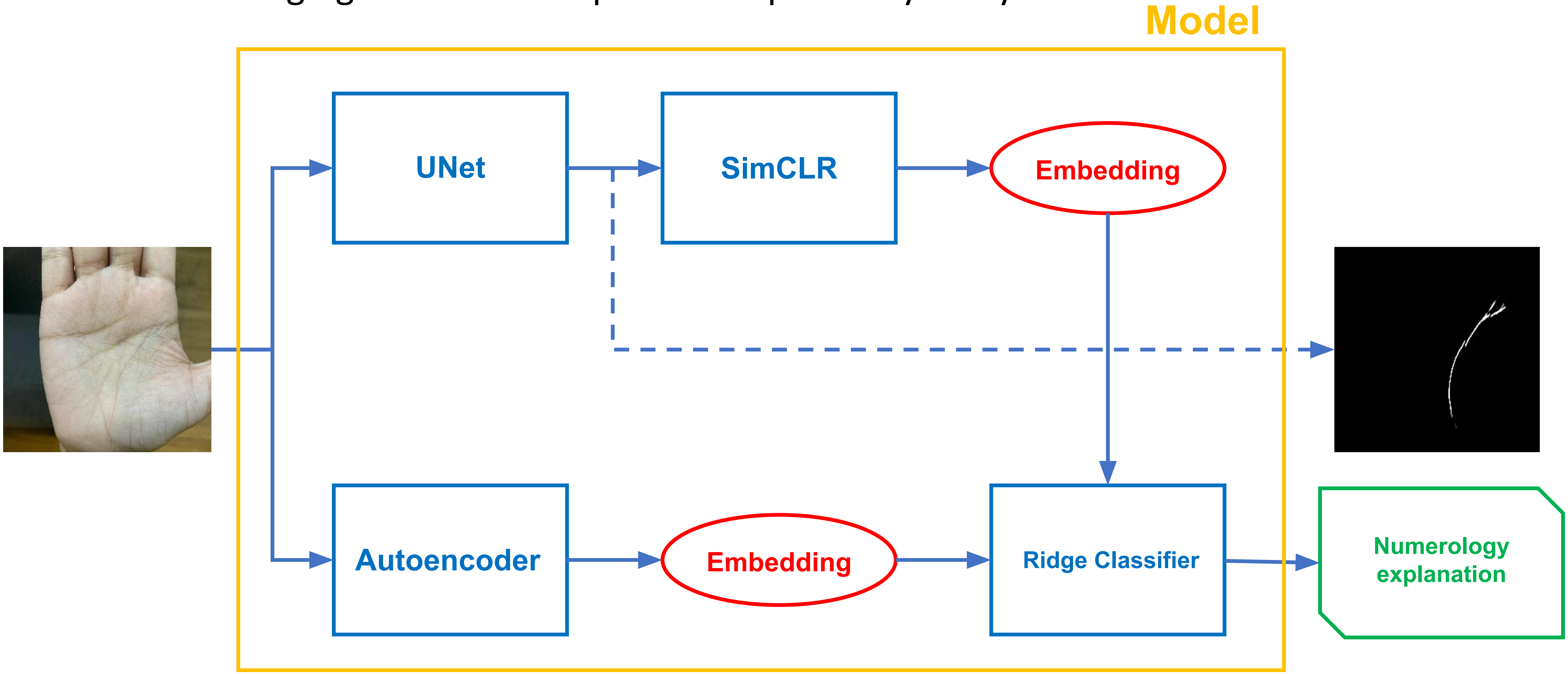
Deep Learning Model

掌紋特徵:
發端偏高
發端呈尖三角形
發端穿過智慧線
間斷在前段
間斷在末段
整體較長
整體較粗
整體筆直

命理解釋範例:
生命線:
發端偏高→個性較有爭鬥性，有自制力且刻苦進取，但容易讓人感到缺少情趣。
發端呈尖三角形→感性敏銳，適合做精巧的工作。個性挑剔、易怒且氣量小。
發端穿過智慧線→事業發展要到三十歲左右才會開始有所成就。
間斷在前段→年幼時環境複雜或常搬遷住所，懂事較早而與父母不親。
間斷在末段→晚年會因為工作而搬家。
整體較長→生命力頑強。
整體較粗→一生無大災大病。
整體筆直→對疾病的抵抗力較差。

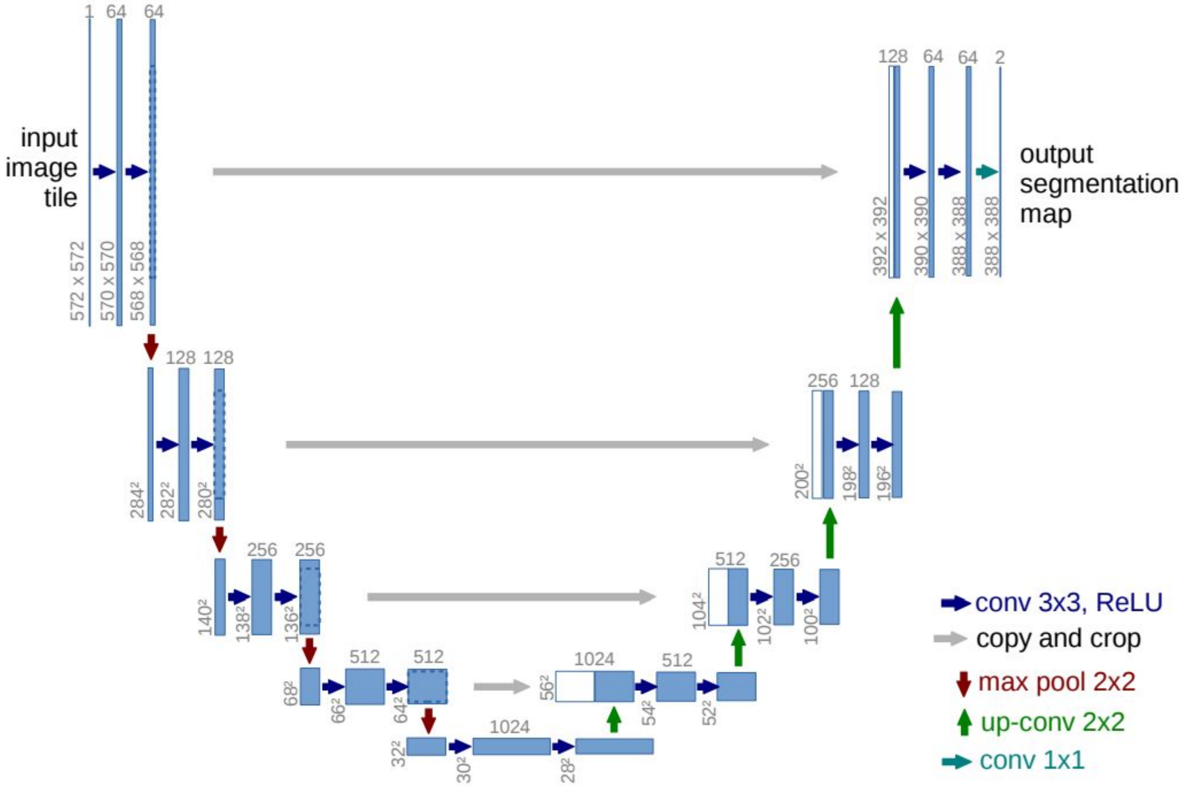
Methodology

- The following figure shows the process of palmistry analysis:



UNet

- The mission of drawing Life Line from palm is a kind of image segmentation mission, we use the popular model structure in medical images called UNet.
- The structure is like U-shaped, it also has encoder and decoder
- However, the difference between from Autoencoder and UNet is that UNet pass results from every layers to decoder, so decoder can get the information of whole image and detailed at the same time.



Dice Loss

- The dice loss solves the problem that our target is tiny in whole image (imbalance problem).
- The definition of dice loss is

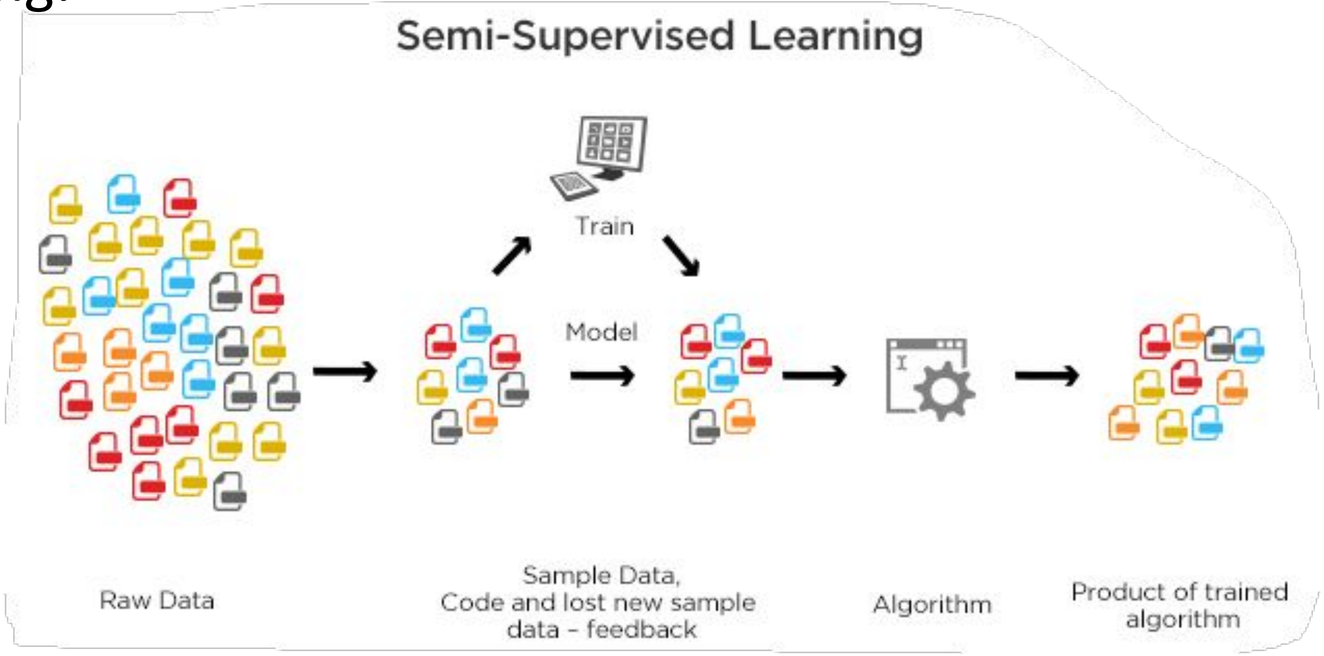
$$L_{dice} = 1 - \frac{2|X \cap Y|}{|X| + |Y|}$$

- |X| means the number of element X, |Y| means the number of element Y, $|X \cap Y|$means the number of intersection of X and Y
- However, since dice loss is not stable, we also use binary cross entropy plus dice loss as our loss.

Semi-supervised Learning

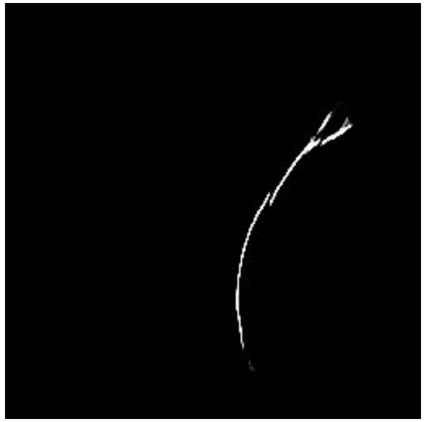
- Since we have few labeled images (240) and lots of unlabeled images (1450), and also we have to detect the tiny object in image, we decide use semi-supervised learning to train the model.
- The process of semi-supervised learning:

1. Use labeled data to train model
2. Use model to predict unlabeled data, and use sigmoid to get probability of image.
3. When probability over 0.7, we set it as pseudo label, and put it into train set to train the model again.



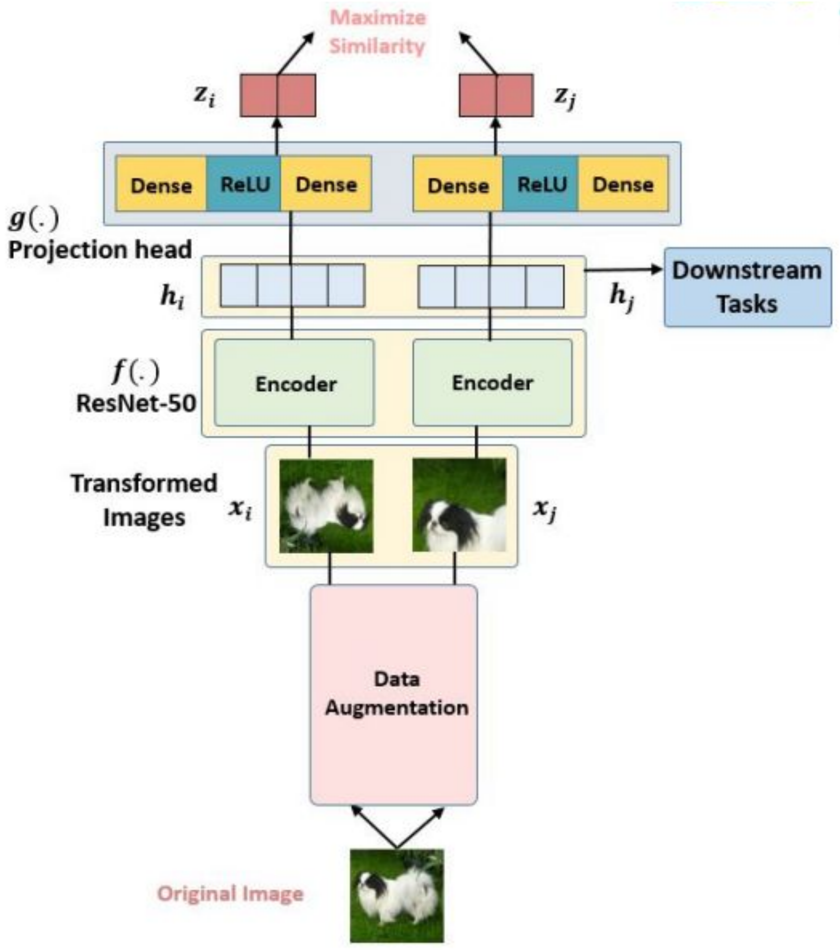
Process of Drawing Life Line

1. We use hand images as inputs, and masks of Life Line as outputs. The size of images and masks are (256,256).
2. Hand images put into the UNet. Here, we train 50 epochs, use Adam optimizer with initial learning rate 0.0001 and step scheduler.
3. After first epoch, we add unlabeled data to do semi-supervised learning. Every epoch, we randomly select 80% unlabeled data with pseudo label into next training dataset to avoid the bias.



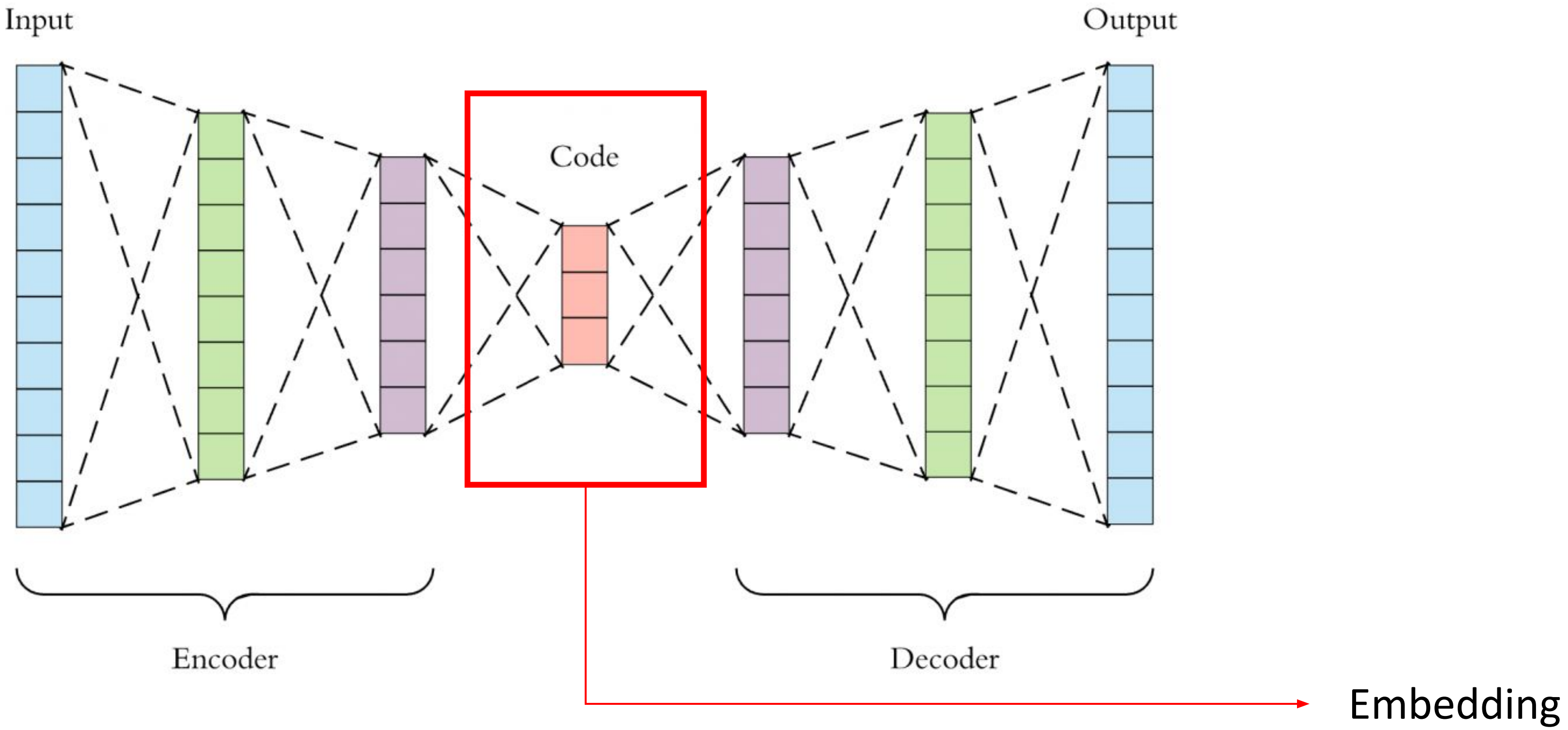
SimCLR

- After getting the mask of life line, we use SimCLR to get the embedding of the life line.
- The reason why we use SimCLR on UNet is because after using contrastive learning, the model need to use the details of line to find the similarity of each lines, and that is what we want.



Autoencoder

- To get the features of the whole hand, we use an autoencoder and take the embedding.
- The reason why we choose to use autoencoder to obtain the features is that if the palm can be restored after being encrypted by the encoder, some important features of the hand must be preserved in the embedding.

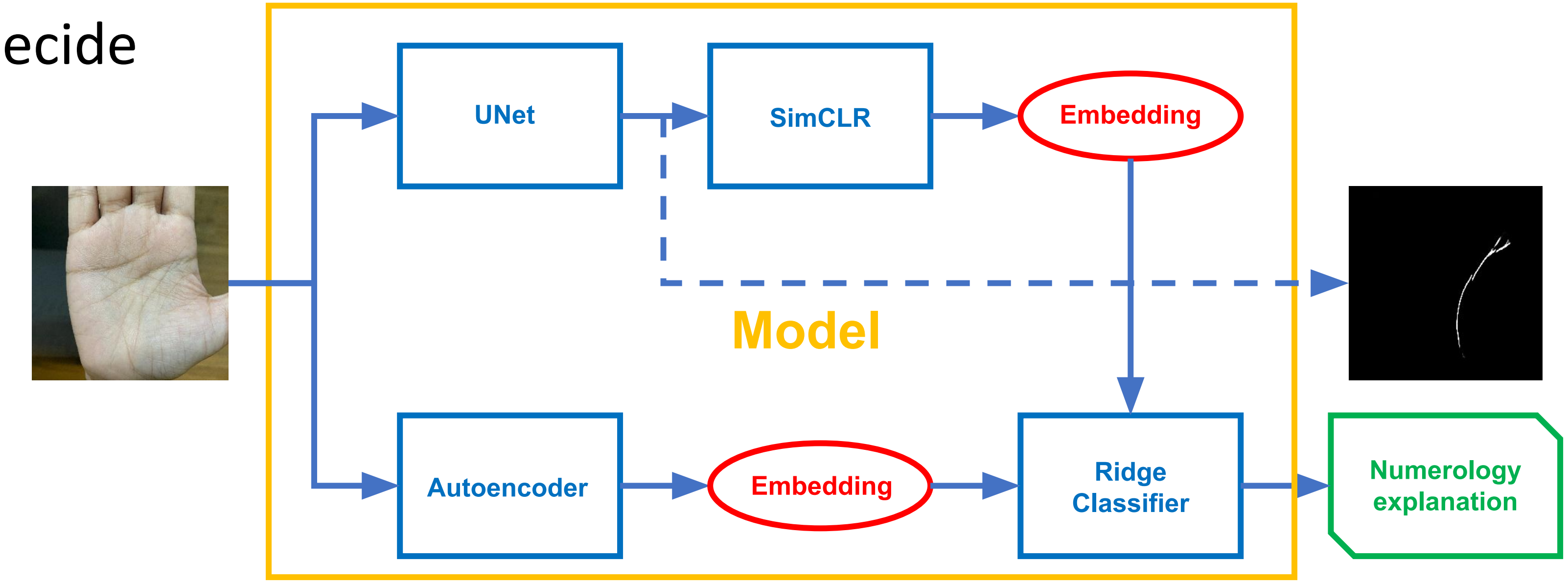


Demo Time!

Summary

- The following figure shows the process of palmistry analysis:

1. Get photo of customer's hand.
2. Put the photo into the UNet, and the output will be hand lines.
3. Put the image output by UNet into SimCLR to get Embedding.
4. Put the photo into the Autoencoder to get Embedding.
5. Combine the two Embeddings into the Ridge Classifier to decide whether it has the feature.
6. We use results from the classifier to generate numerology explanation.



Thank You!

Reference

- Ronneberger, O., Fischer, P., & Brox, T. (2015, October). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention* (pp. 234-241). Springer, Cham.
- Tan, M., & Le, Q. (2019, May). Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning* (pp. 6105-6114). PMLR.
- Chen, D., Ao, Y., & Liu, S. (2020). Semi-supervised learning method of u-net deep learning network for blood vessel segmentation in retinal images. *Symmetry*, *12*(7), 1067.
- Abdollahi, A., Pradhan, B., & Alamri, A. (2020). VNet: An end-to-end fully convolutional neural network for road extraction from high-resolution remote sensing data. *IEEE Access*, *8*, 179424-179436.
- Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. (2020, November). A simple framework for contrastive learning of visual representations. In *International conference on machine learning* (pp. 1597-1607). PMLR.
- https://github.com/usuyama/pytorch-unet
- https://github.com/LovreAB17/Eff-UNet
- https://speech.ee.ntu.edu.tw/~hylee/ml/2021-spring.php
  - HW2 Classification
  - HW8 Anomaly Detection