INT3404E20 - Image Processing: Final Report
Sino-nom Character Retrieval

Lê Minh Đức - 22028267
Nguyễn Hoàng Quân - 22028130

May 29, 2024

Abstract

This report presents the development of an image retrieval pipeline for Sino-nom characters as part of a school project for the course INT3404E - Image Processing. The pipeline aims to accurately retrieve the most similar 3D items from a database given a query image. Various approaches were explored, including data augmentation, data generation using different fonts and localization techniques, and deep metric learning with different loss functions. The final model, utilizing Triplet Loss with Online Hard Triplet Mining, achieved an MRR@5 score of 0.957, demonstrating high retrieval accuracy.

1 Introduction

Sino-nom characters, blending elements of Chinese and Vietnamese scripts, present unique challenges in image processing. Efficiently retrieving similar characters from a database is crucial for various applications such as character recognition and document analysis. In this study, we aim to develop an image retrieval pipeline specifically tailored for Sino-nom characters.

Although many local feature matching algorithms like SIFT, SURF, or ORB exist, these are designed to achieve speed and efficiency in the retrieval process. We decided to tackle this problem with a deep learning approach, which is expected to outperform traditional methods.

We experimented with different approaches to improve the retrieval performance. We started by exploring data augmentation and data generation techniques to increase the diversity and granularity of the dataset. Additionally, we tested various deep learning models such as EfficientNet and CAFormer to determine the most effective feature extraction methods for Sino-nom characters.

Furthermore, we delve into deep metric learning techniques to learn embeddings that capture the similarity between images. Through experimentation, we evaluate the performance of

Figure 1: Convert 3D Objects to 2D images by processing the depth buffer.

different loss functions, including Contrastive Loss and Triplet Loss with Online Hard Triplet Mining, to optimize the retrieval pipeline. Our analysis focuses on achieving high Mean Reciprocal Rank (MRR@5) scores, indicating the pipeline's accuracy in retrieving similar 3D items for query images.

2 Approach

2.1 Handling 3D Objects with Depth Buffer

The first step is to convert 3D Objects to 2D images. We found out that we could get a nice capture by processing the depth image of the mesh, which represents the distance from the camera to each point on the object's surface. We used the Open3D library to generate depth images using off-screen rendering techniques, then used various processing steps to contrast the details of the image from the background. The results look like figure 1.

2.2 Feature Extraction

The initial straightforward approach is to train a classifier to extract a feature vector from an input image, then use some distance function to calculate the similarity of 2 images.

2.2.1 Image classification models

There are many architectures that achieve state-of-the-art image classification performance, but we only experiment with 2 architectures that are both efficient and predictive:

- EfficientNet[6]: Outperforming conventional ConvNets by systematically scaling all dimensions of an existing architecture like ResNet and MobileNet. Researchers then utilized Neural Architecture Search (NAS) to discover a new efficient and high-performing architecture that synergizes well with the scaling algorithm.

- CAFormer[7]: Built upon a systematic study of the MetaFormer architecture, then combine convolution architecture with self-attention to achieve high performance and efficiency.

We also utilize ML-Decoder[4], an attention-based classification head recently proposed with state-of-the-art performance.

2.2.2 Distance Metrics

After training a classifier, we conducted tests using various distance metrics to determine their impact on the retrieval performance. The metrics comprise Euclidean Distance, Cosine Similarity of the features vector, as well as KL Divergence and Cross Entropy of the final classifying distribution. The results are shown in table 1.

Table 1: Benchmark of retrieval performance of different distance metrics

	Euclidean	Cosine Similarity	KL Divergence	Cross Entropy
MRR@5	0.92	0.81	0.82	0.88

Euclidean Distance, though simple, proved to be the most effective. This result can be explained by the fact that cosine similarity is more suited for text, and probabilistic metrics, while promising in theory, may have reduced the feature vector dimension too drastically, potentially failing to capture complex image features effectively.

This provides a good baseline for our retrieval problem. However, there is still room for improvement. In subsequent steps, we plan to explore some other approaches to enhance the retrieval accuracy further.

2.3 Data Generation

Why do we need more data, even though the baseline performance is already good? After some exploration, Quan realized that there are many Nom fonts available online. Additionally, there is a large Nom dataset called NomNaOCR that already exists. Due to their variety, these sources could potentially improve our baseline.

2.3.1 Generate more data by using different fonts

The inclusion of different fonts allows us to introduce variations in stroke thickness, character style, and overall appearance, which may not be adequately represented in our baseline dataset. We found 47 fonts online, drew the fonts with different stroke levels and noisy backgrounds, and collected 350k images of 7500 different characters. Some of the images are shown in figure 2.

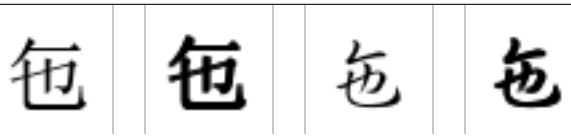

Figure 2: Generate more data with different fonts.

2.3.2 Generate more data by processing NomNaOCR

Since NomNaOCR has transcripts associated with roughly 40k patches of characters, we can perform localization on these patches and extract each image of each character in respect to its transcript. We achieved this by training a simple localization model based on the YOLOv8 architecture, then collecting transcribed characters for each bounding box. The simple model accurately localized approximately 95% of the patches, which resulted in roughly 400k images with about 7500 different characters. Some of the images are shown in figure 3.

Figure 3: Generate more data from processing NomNaOCRs.

2.4 Deep Metric Learning Experiments

2.4.1 What is metric learning?

The goal of Metric Learning is to learn a representation function that maps objects into an embedded space. The distance in the embedded space should preserve the objects' similarity — similar objects get close and dissimilar objects get far away. The pipeline is shown in figure 4.

The key to achieving a good metric learner is to define an effective loss function. Various loss functions have been developed for Metric Learning. Additionally, Hard Negative Mining is also an important technique to achieve optimal performance.

2.4.2 Contrastive loss

Contrastive loss[2] is one of the earliest training objectives used for deep metric learning in a contrastive fashion. Contrastive loss guides the objects from the same class to be mapped to the same point and those from different classes to be mapped to different points whose distances are larger than a margin.

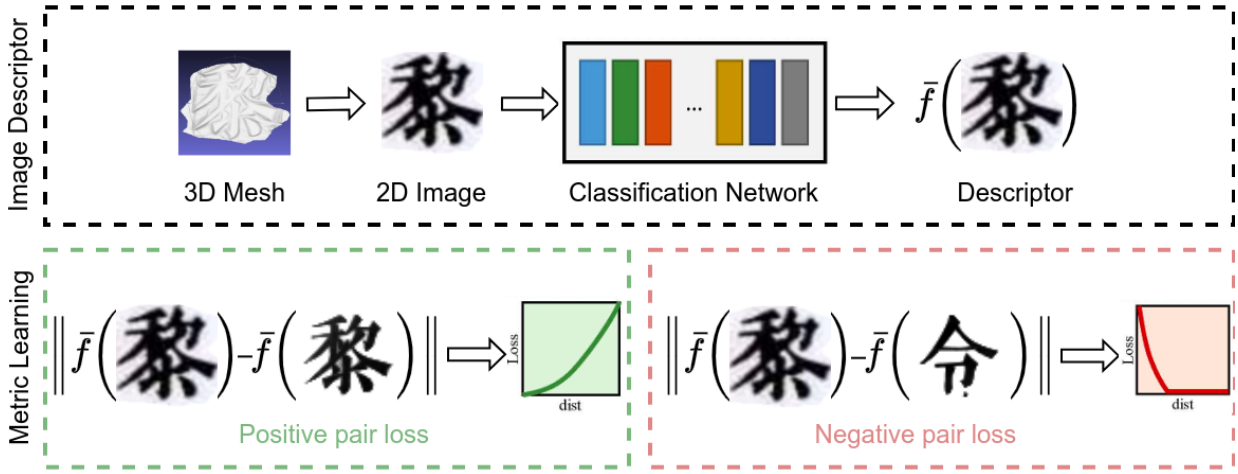

Figure 4: Metric learning pipeline.

2.4.3 Recall@5 Surrogate Loss

The Recall@5 Surrogate Loss[3] addresses the challenge of optimizing non-differentiable evaluation metrics like recall in retrieval tasks. The loss function is a differentiable version of the recall@k metric, which enables direct optimization through gradient descent.

2.4.4 Triplet loss

Triplet loss was originally proposed in the FaceNet[5] paper and was used to learn face recognition of the same person at different poses and angles. Triplet loss requires the distance between the anchor sample and the positive sample to be smaller than the distance between the anchor sample and the negative sample. The illustration of the loss is shown in figure 5.

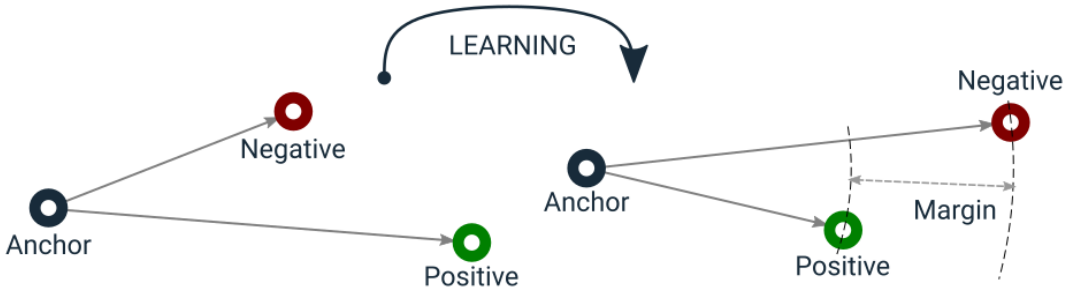

Figure 5: Triplet loss learning.

2.4.5 Online Hard Triplet Mining

Triplet models are notoriously tricky to train. In fact, our training process with triplet loss was slow to converge and didn't show any improvement. As we tried to debug this, we found out that the loss is very spiky, which means there are only a few batches with high losses, while other batches have consistently low, near 0 losses. To address this issue, we implemented Online Hard Triplet Mining, which instead of composing a triplet at random, we choose the triplets with the highest loss available in a batch. This method significantly speeds up the training of triplet loss and improves the retrieval performance.

3 Experiments

3.1 Dataset

We use the recognition dataset, localization dataset and retrieval dataset provided by Teacher Assistant. Additionally, we generated 750k images of 7500 different Nom characters from different fonts online and extracted from NomNaOCR.

3.2 Metric

Mean Reciprocal Rank (MRR) is one of the metrics that help evaluate the quality of recommendation and information retrieval systems. Mean Reciprocal Rank (MRR) at K evaluates how quickly a ranking system can show the first relevant item in the top-K results. Here is the formula that defines MRR:

$$\text{MRR@5} = \frac{1}{U}\sum_{u=1}^{U}\frac{1}{\text{rank}_u}$$

3.3 Results

Table 2 shows the retrieval performance of different approaches. All the methods use the Euclidean Distance as the distance metric.

Table 2: Results of different approaches.

	No Metric Learning	Contrastive Loss	Triplet Loss	Surrogate Loss
MRR@5	0.92	0.948	0.957	0.92

3.4 Additional Analysis

The results in Table 2 demonstrate that deep metric learning approaches significantly improve retrieval performance compared to the baseline method that relies solely on feature extraction and distance metrics. Triplet Loss with Online Hard Triplet Mining achieved the

best MRR@5 score (0.957), indicating it retrieved the most similar 3D objects for query images compared to other approaches. The Recall@5 Surrogate Loss did not improve the retrieval performance, since we were using a low batch size, and increasing the batch size was not feasible due to resource constraint.

4 Future Works

There are several interesting avenues to explore further and potentially improve retrieval accuracy:

- Data Augmentation Techniques: Experiment with more advanced data augmentation techniques like enhance image, perspective shift, and elastic deformations to introduce even more variations in the training data.
- Alternative Loss Functions: Explore other metric learning loss functions like quadruplet loss, batch all mining, which might lead to further improvements in performance.
- Increase batch size: Increasing the batch size can potentially lead to more hard triplets in one batch, making training more stable and improving performance, if additional computational resources are available.
- Backbone Architectures, Transfer Learning: Experiment with different backbone architectures like large vision transformers. Distil knowledge from large pretrained models like Unicom[1].

5 Conclusion

This report presented the development of an image retrieval pipeline for Sino-nom characters. The pipeline achieved high retrieval accuracy (MRR@5 of 0.957) using Triplet Loss with Online Hard Triplet Mining for deep metric learning. The findings demonstrate the effectiveness of deep learning approaches for Sino-nom character retrieval tasks. Future work can focus on exploring more advanced data augmentation techniques, alternative loss functions, and backbone architectures to achieve even better retrieval performance.

References

[1] Xiang An, Jiankang Deng, Kaicheng Yang, Jaiwei Li, Ziyong Feng, Jia Guo, Jing Yang, and Tongliang Liu. "Unicom: Universal and compact representation learning for image retrieval". In: *arXiv preprint arXiv:2304.05884* (2023).

[2] S. Chopra, R. Hadsell, and Y. LeCun. "Learning a similarity metric discriminatively, with application to face verification". In: *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. Vol. 1. 2005, 539–546 vol. 1. DOI: 10.1109/CVPR.2005.202.

[3] Yash Patel, Giorgos Tolias, and Jiri Matas. "Recall@k Surrogate Loss with Large Batches and Similarity Mixup". In: *CoRR* abs/2108.11179 (2021). arXiv: 2108.11179. URL: https://arxiv.org/abs/2108.11179.

[4] Tal Ridnik, Gilad Sharir, Avi Ben-Cohen, Emanuel Ben-Baruch, and Asaf Noy. *ML-Decoder: Scalable and Versatile Classification Head*. 2021. arXiv: 2111.12933 [cs.CV].

[5] Florian Schroff, Dmitry Kalenichenko, and James Philbin. "FaceNet: A Unified Embedding for Face Recognition and Clustering". In: *CoRR* abs/1503.03832 (2015). arXiv: 1503.03832. URL: http://arxiv.org/abs/1503.03832.

[6] Mingxing Tan and Quoc V. Le. "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks". In: *CoRR* abs/1905.11946 (2019). arXiv: 1905.11946. URL: http://arxiv.org/abs/1905.11946.

[7] Weihao Yu, Chenyang Si, Pan Zhou, Mi Luo, Yichen Zhou, Jiashi Feng, Shuicheng Yan, and Xinchao Wang. "Metaformer baselines for vision". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2023).