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

Nguyen Khoa Dang - 21021478

May 29, 2024

Abstract

This paper presents a novel system for 3D object retrieval using Sino-nom characters as image input 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. The final model, utilizing Triplet Loss and multi-stream model achieved an MRR@5 score of 0.087.

1 Introduction

Sino-nom characters, a mix of Chinese characters and Vietnamese script, pose difficulties for computers to understand images. Finding similar characters quickly from a large collection is important for tasks like recognizing characters and analyzing documents. This research proposes a new way to retrieve images specifically designed for Sino-nom characters, using powerful deep learning techniques instead of simpler methods that prioritize speed (SIFT and SURF).

This research proposes a novel approach to address this challenge. We leverage the power of deep learning to develop a robust image retrieval system specifically designed for Sino-nom characters. Deep learning algorithms have demonstrated remarkable capabilities in extracting complex features from images, making them ideally suited for this task, for both 2D images or 3D objects. I tested various deep learning models such as MobileNet and Resnet to determine the most effective feature extraction methods for 2d Sino-nom characters and PointNet for 3D point cloud.

Furthermore, i utilize deep learning techniques to learn embeddings that capture the similarity between 2D images and 3D point cloud. The pipeline results although not very high but show the capabilities of the 3D information in retrieval problem at the moment.

2 Methods

2.1 Point Cloud Sampling with 3D Objects

The first step is to convert 3D mesh into point cloud. I used the Open3D library to generate that using the uniform sampling method, results in 1024 points. The results look like this

2.2 Feature Extraction

The initial approach is to train a classifier to extract a feature vector from an input 2d image and one for the point cloud, then use the sigmoid function to calculate the probability of matching between the two objects and using the cross-entropy loss. But during my implementation of this approach, the results is very poor as it almost turn the problem into a multi-class classification with only one sample of data. Then i switch to using the 2 features extractor to encode the image and point cloud then use some distance function to calculate the similarity of the point cloud and the image.

2.2.1 Feature extraction models

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

- MobileNet[3]: This lightweight convolutional neural network (CNN) is known for its efficiency while maintaining good accuracy. Its focus on reducing computational cost makes it well-suited for deployment on mobile devices or resource-constrained environments.

- Resnet50d[1]: This powerful CNN architecture utilizes residual connections to address the vanishing gradient problem that can hinder training in deep networks. ResNets achieve high accuracy on image classification tasks and can be adapted for various retrieval applications.
- PointNet[2]: This pioneering architecture specifically designed for point clouds directly processes unordered sets of 3D points. It learns features from the point cloud data, enabling tasks like object classification and retrieval.

For the two 2d features extraction models, i use the pre-trained weights on the ImageNet dataset. For the point cloud model, i use weights on the ModelNet40 dataset

2.3 Multi-Stream Architecture Experiments

2.3.1 Multi-Stream Architecture

This work leverages a multi-stream architecture for robust 3D object retrieval using Sino-nom characters. One branch of the network processes the 2D Sino-nom character images, while a separate branch handles the 3D point cloud data representing the objects in the database and 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 1.

The key to achieving a good metric learner is to define an effective loss function. Due to the nature of the retrieval problem, i implemented the triplet loss function.

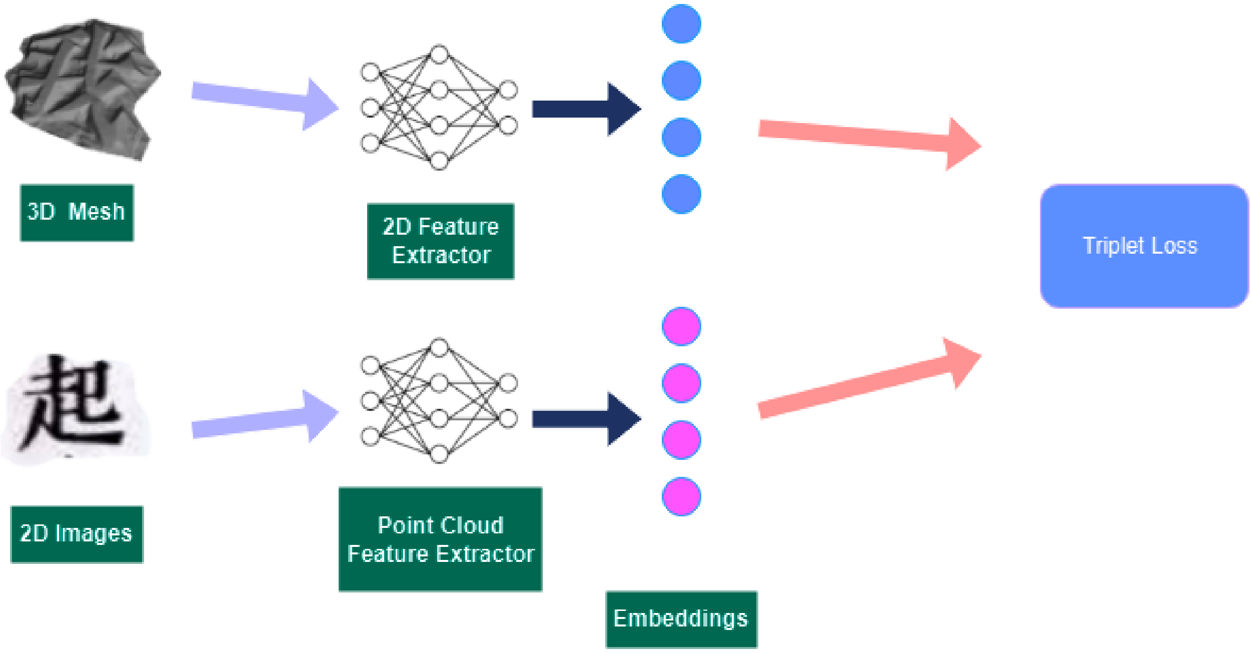

Figure 1: Retrieval pipeline.

2.3.2 Triplet loss

Triplet loss was originally proposed in the FaceNet[4] 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 2.

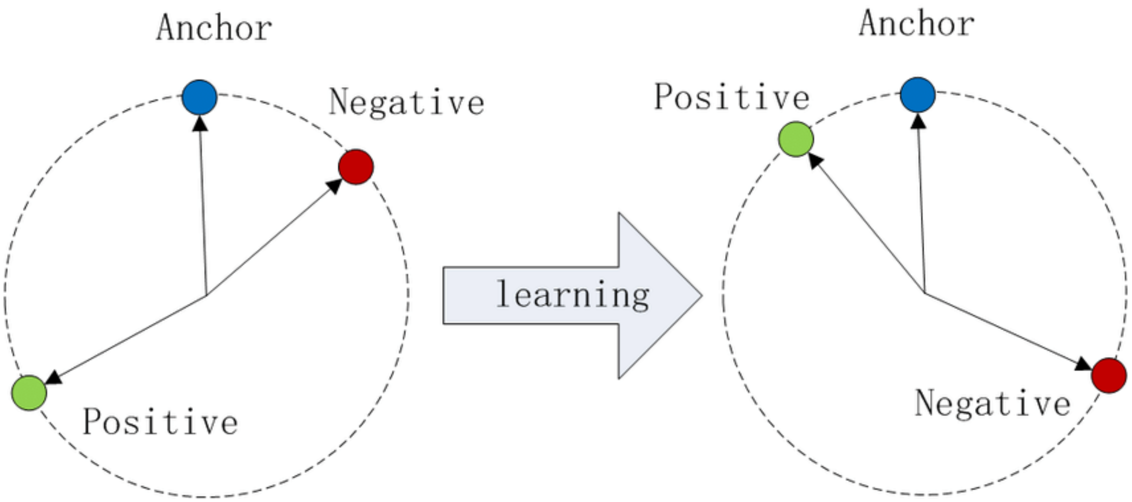

Figure 2: Triplet loss learning.

3 Experiments

3.1 Dataset

I use the retrieval dataset provided by Teacher Assistant, which contain about 300 query image and 300 3d mesh.

3.1.1 2D Images

The query image, which contain a Sino-nom character on a white background

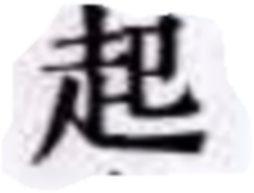

Figure 3: The 2D query image

3.1.2 3D Mesh

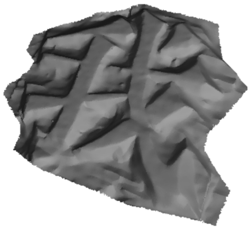

Figure 4: The STL Mesh

The 3D mesh is in the form of an stl file, which standing for stereolithography (a 3D printing technology), is a common file format used in 3D printing and computer-aided design. STLs describe this surface using a series of connected triangles, also known as tessellation. The more triangles used, the more detailed the representation of the object's surface becomes

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 1 shows the retrieval performance of my approach, using the Euclidean distance as the distance metric.

Table 1: Results of the approach.

	MobileNet	Resnet50d
MRR@5	0.089	0.067

3.4 Additional Analysis

Experiments evaluating the 2D image branch of the network indicated that ResNet50d achieved slightly better performance compared to MobileNetV2. While MobileNetV2 offers advantages in terms of computational efficiency, the deeper architecture of ResNet50d appears to be more effective in capturing the intricacies of Sino-nom characters. This suggests that for this specific task, the benefits of improved accuracy with ResNet50d outweigh the potential efficiency gains of MobileNetV2. Further investigation could explore techniques for optimizing ResNet50d's architecture or leveraging transfer learning to achieve a balance between accuracy and efficiency.

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.
- Backbone Architectures, Transfer Learning: Experiment with different backbone architectures like large vision transformers.

5 Conclusion

This report presented the development of an image retrieval pipeline for Sino-nom characters. The pipeline achieved high retrieval not so high results using Triplet Loss for the Multi-stream model. This work on a new way to approach the 3D to 2D matching problem by utilize the point cloud representation although the result is not so promising. Future work can reflect on what when wrong in this approach to better tackle this challenging problem.

References

[1] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. *Deep Residual Learning for Image Recognition.* 2015. arXiv: `1512.03385 [cs.CV]`.

[2] Charles R. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. *PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation.* 2017. arXiv: `1612.00593 [cs.CV]`.

[3] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. *MobileNetV2: Inverted Residuals and Linear Bottlenecks.* 2019. arXiv: `1801.04381 [cs.CV]`.

[4] 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`.