507 Project Proposal

Jiaqi Sun

November 2024

1 Overview

During Apple's fall iPhone event this year, the company introduced a new feature called *Photographic Styles*, which offers great creative flexibility for adjusting image colors while preserving characteristic colors, such as skin tones and hair colors. Humans are particularly sensitive to these colors, as deviations can make photos look unnatural. This feature ensures that editing the background of an image does not overly alter a subject's skin tone or brighten their hair.

However, this feature is exclusive to Apple's latest devices, as it relies on advanced hardware to analyze photos during capture. The algorithm segments parts of the image, such as skin and hair, and store them into distinct layers, making it incompatible with older iPhones or Android devices. My goal is to develop a similar feature that works across older iPhones and other Android phones, enabling color adjustments while preserving characteristic colors.

To achieve this, I plan to train a supervised learning model that is capable of generating a mask image from the input photos. Each pixel in the mask will represent a specific class (e.g., face, hair, arms, background). This task, known as semantic segmentation, involves identifying and classifying individual image components. I plan to fine-tune Apple's MobileViT+DeepLabV3 model on a labeled human parsing dataset, leveraging pre-trained segmentation models for efficient and accurate training. After successfully identifying semantic features in the image, I will apply Lookup Table (LUT) filter to modify other pixels while keeping the characteristic parts (e.g., skin and hair) unaffected. This part of the project will involve the use of OpenCV and NumPy libraries for image processing.

Through this project, I aim to develop skills in Python programming and the application of deep learning models. Additionally, I seek to gain a comprehensive understanding of semantic segmentation advancements, the principles of image filter application, and the operational workflow of Great Lakes ARC.

2 Prior Work

Semantic segmentation is a foundational task in computer vision, aiming to partition an image at the pixel level into semantically coherent regions. It evolves from earlier tasks such as image classification and object recognition and has been extended by instance segmentation and panoptic segmentation. Semantic segmentation has wide applications in fields like medical imaging and autonomous driving. The development of semantic segmentation can be divided into three major stages. Early methods relied on hand-crafted features such as edge detection or color segmentation. While straightforward, these approaches struggled with complex images and image variations.

In 2015, the Fully Convolutional Network (FCN) replaced fully connected layers with convolutional layers, enabling input images of arbitrary sizes. FCN also introduced upsampling techniques, which achieved end-to-end semantic segmentation for the first time. Later, U-Net refined this approach by incorporating an encoder-decoder structure with "short-cuts", combining low-level feature maps (rich in detail) with high-level ones (rich in semantic information). This architecture improved the precision of segmentation masks and excelled in medical imaging tasks. Meanwhile, the DeepLab series introduced dilated convolutions, extending the receptive field without increasing parameters. This was combined with multi-scale feature map fusion, enhancing the adaptability to complex scenes.

More recently, the introduction of Transformers has driven further advancements. Transformers replaced convolutions with self-attention mechanism, enabling the modeling of the global dependencies. Vision Transformer (ViT) was the first to demonstrate the application of Transformers to image data, dividing images into patches (16×16 pixels) and flattening them. This approach redefined image tasks in a manner similar to textual tasks and highlighted the potential of Transformers for multimodal data. SegFormer combined multi-level transformer decoder blocks with multi-layer perceptrons (MLPs), balancing accuracy and efficiency. Swin Transformer introduced a hierarchical structure with a sliding-window attention mechanism, significantly reducing computational complexity while retaining the ability to capture global dependencies. Transformer-based models currently dominate semantic segmentation benchmarks, achieving state-of-the-art performance across various datasets.

3 Preliminary Results

For the project, I chose Apple's MobileViT+DeepLabV3-small model as the initial model. MobileViT was pretrained on PASCAL VOC with a resolution of 512×512, and a DeepLabV3 head is integrated for semantic segmentation tasks. This combination provides a strong starting point for building an efficient and accurate model. This model has a

relatively small parameter size of 6.4 million, which makes it easier and more suitable for deployment on mobile devices with limited computational power.

I used the Human Parsing Data dataset for fine-tune, which consists of 17,706 samples of fashionably dressed individuals. Each sample includes an 600×400 original image and a corresponding mask image. The mask has 18 categories that segment the image pixels into regions such as background, hair, face, arms, and legs. This dataset is highly suitable for my project as it provides detailed labels of human skin and hair, which is crucial for the task of fine-tuning the model. However, the large number of images in the dataset made the training process slow without hardware acceleration. To overcome this, I utilized the Great Lakes ARC at the university, leveraging GPU acceleration to speed up the training process.

In terms of implementation, I used basic Python knowledge and the PyTorch library to fine-tune the model. For visualizing the results, I used the Matplotlib library to plot images. I also explored NumPy and OpenCV for image manipulation and applying filters. The performance of the model is evaluated using Intersection over Union (IoU), a common metric in semantic segmentation. Figure 1 and 2 present a comparison between segmentation performance of the original model and the 3-Epoch fine-tuned model on a same image. It shows how fine-tuning improves the model's ability to segment the image.

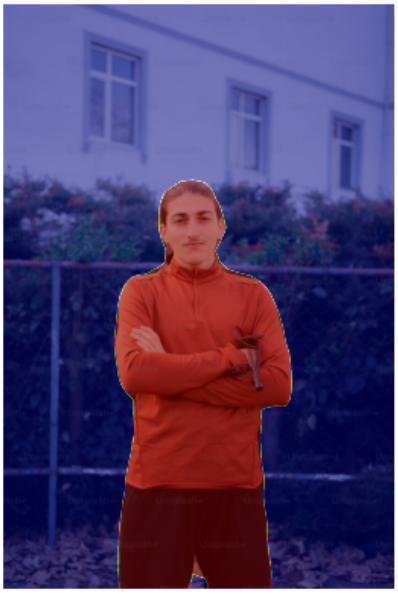

Figure 1: Original Model

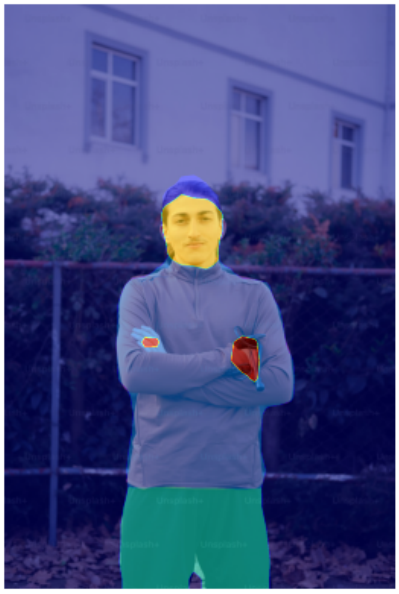

Figure 2: Fine-Tuned Model

4 Project Deliverables

- Successfully fine-tune the chosen model (MobileViT+DeepLabV3-small) to achieve strong segmentation performance, evaluated using Intersection over Union (IoU) as the primary metric.
- Successfully apply the fine-tuned model to preserve characteristic colors during image adjustment tasks.
- Gain proficiency in using Git and Great Lakes ARC.
- Learn the techniques for fine-tuning existing deep learning models for domain-specific tasks.
- Acquire a comprehensive understanding of key models in image semantic segmentation.
- Build fluency in Python programming and practical experience in PyTorch, NumPy.

5 Timeline

- Week 1-2: Learn version control using Git through ChatGPT; explore Hugging Face to find a suitable model and dataset; study the usage of Great Lakes ARC and hardware acceleration.
- Week 3-4: Conduct literature review on segmentation; fine tune the model on Great Lakes; complete the project proposal.
- Week 5-6: Apply LUT on images; refine the code into a notebook; analyze the performance; finalize the project report.