Clustrex Data Private Limited

Case Study
UNET for Eyewear Segmentation (Full Rim)

1. Introduction

Problem Statement:

Accurately identifying and segmenting eyewear in full-rim spectacle images is critical for applications in virtual try-ons. Preparing production-ready images has traditionally involved manual editing, often requiring skilled Photoshop experts to fine-tune the details. This process can be time-consuming and labor-intensive, demanding significant human resources to ensure the precision needed to create an image which can be used in virtual tryon. The complex structure of eyewear, combined with varying lighting conditions in real-world images, further complicates the task. Precise segmentation, especially around lens boundaries, is essential to guarantee the smooth and realistic overlays required for high-quality virtual try-on experiences. Automating this process with a UNet-based model can drastically reduce the need for manual intervention, saving time and resources while enhancing consistency and accuracy.

Objective:

To develop a UNet-based segmentation model capable of accurately detecting and segmenting full-rim eyewear from images. The model should be robust enough to handle diverse image inputs, accounting for variations in lighting conditions, frame styles, and lens transparency. The goal is to ensure consistent and reliable detection of both the frame and lenses, reducing the need for manual editing and preparing production-ready images for virtual try-on applications efficiently. By automating this segmentation process, the model will streamline the workflow, significantly reducing human effort while maintaining high accuracy in lens and frame detection.

2. Data

Dataset:

The dataset used for detecting eyewear in full-rim spectacles consists of a collection of images, each paired with manually annotated masks for two separate regions: the outer frame and the inner glass area. This separation allows for more precise segmentation of both the eyewear structure and the lens itself. Initially, the dataset contained 150 images, but it was later expanded to 200 images to improve model robustness and generalization. These images were carefully curated to represent a wide range of lighting conditions, frame styles, and lens transparencies, ensuring the model's ability to handle various real-world scenarios.

We maintained two datasets using the same set of images:

- **Outer Frame Dataset**: This set contains images with annotated masks that focus on segmenting the outer frame of the eyewear.
- **Glass Area Dataset**: This set contains images with annotated masks that specifically segment the inner lens or glass area.

By having separate datasets for the frame and lens, we ensure that the model learns to distinguish between these two distinct components, enabling more accurate and context-aware segmentation.

Data Preprocessing:

Several preprocessing steps were applied before passing the images to the UNet model:

- Resizing: All images were resized to 900x600 pixels to standardize the input size while maintaining aspect ratio.
- Normalization: Pixel values were normalized to a [0, 1] range to stabilize training and improve convergence.

Train-Test Split:

The dataset was split into three subsets for balanced evaluation:

- Training Set: 80% of the images were used to train the model.
- Test Set: The remaining 20% was reserved for testing the model on unseen data.

MODEL ARCHITECTURE

U-Net's architecture is distinctive, comprising a contracting path and an expansive path. The contracting path serves as an encoder, capturing context by progressively reducing the spatial resolution of the input, while the expansive path acts as a decoder, reconstructing the segmentation map by using information from the contracting path through skip connections.

In the contracting path, encoder layers focus on identifying key features by applying convolutions that downsample the input, creating deeper and more abstract representations. This process is similar to the feedforward layers found in other convolutional networks. Meanwhile, the expansive path decodes these representations and restores spatial resolution. The decoder layers upsample the feature maps, with skip connections from the contracting path providing additional spatial information that allows for more precise localization of features in the segmentation output.

5. Training Process

Training Setup:

The UNet model was trained using a CUDA-enabled GPU to speed up computations. The following settings were applied during the training process:

- **Hardware**: CUDA-enabled GPU for faster model training and parallel processing.
- **Epochs**: The model was trained for a set number of epochs (e.g., 100-300) to allow sufficient learning while avoiding overfitting.

- **Batch Size**: A batch size of 1-4 was used to balance memory usage and model performance.
- **Learning Rate**: The learning rate was set at 0.001 initially, with a learning rate scheduler to reduce the rate when the model's performance plateaued.

Challenges:

One of the main challenges was segmenting the lens boundaries accurately, especially in cases where the lenses were transparent or reflections caused misleading boundaries. Additionally, thin lens frames posed difficulties for the model in distinguishing between the frame and the lens.

6. Evaluation

Metrics:

To assess the performance of the UNet model on lens segmentation, the following metrics were used:

Binary Cross-Entropy (BCE) Loss: Used during training and testing to measure the difference between the predicted and true masks.

Results:

Test Loss (BCE): 0.04

These metrics indicate that the model generally performs well on images with clear lens and frame boundaries. However, for images where the color of the lens, frame, and background are similar, the model struggles to correctly predict the lens region.

Visual Observation:

For images with distinct lens frames, the predicted masks closely match the ground truth. However, when the background or frame color is similar to the lens, the predicted masks tend to blur the boundaries, leading to inaccuracies.

7. Model Deployment

Deployment Setup:

The model is deployed to process uploaded images of spectacles and return the production ready images. The prediction is saved as an image file for further use in virtual eyewear try on. The model runs on a server using a CUDA-enabled GPU for fast processing.

8. Limitations

Model Limitations:

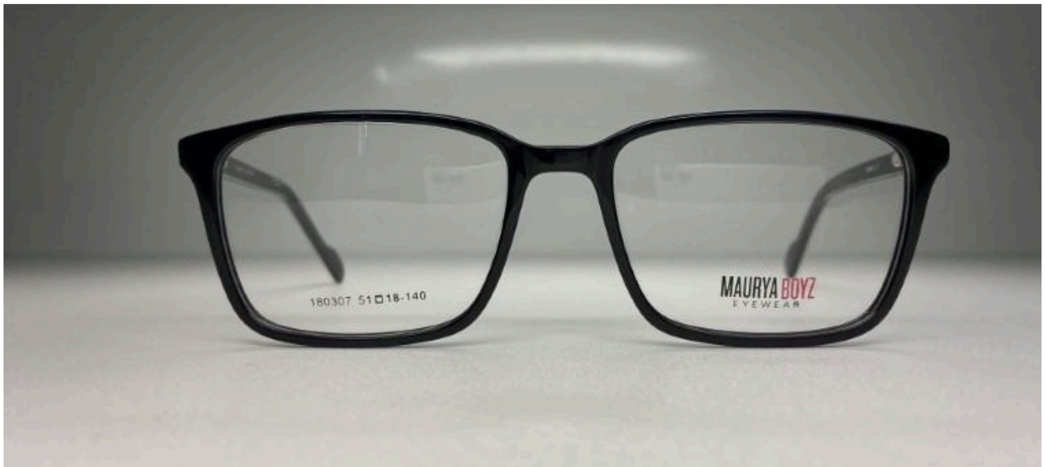
- Color Similarity: The model struggles with images where the background, frame, and lens have similar colors. This leads to poor segmentation results for those cases, as the boundaries between the lens and other elements become indistinct.

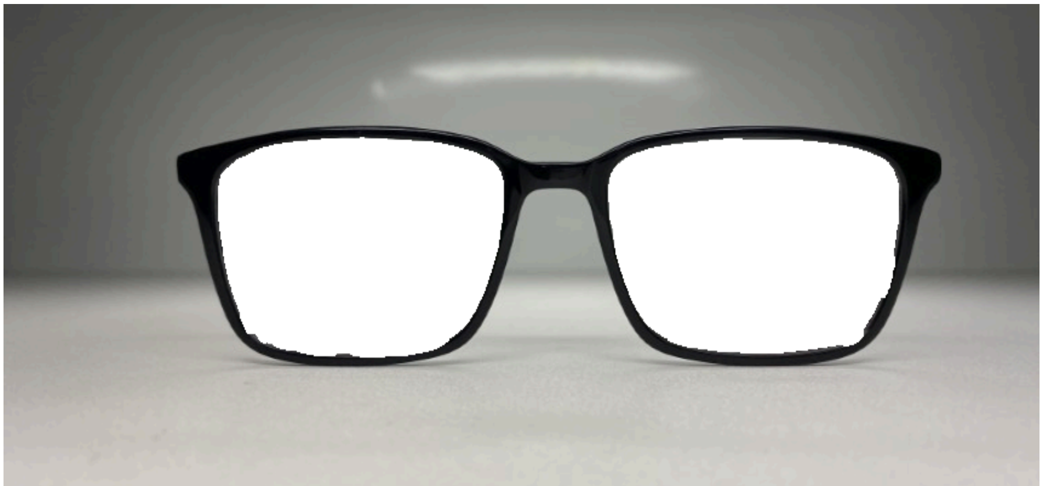
- Transparency and Reflection: The model also has difficulty with transparent lenses and reflections, where the lens boundaries are harder to detect due to light distortions.

9. Conclusion

In conclusion, the UNet-based segmentation model demonstrates strong performance in detecting lenses in full-rim spectacles under normal conditions. While challenges remain in cases with similar-colored backgrounds and lenses, the model provides a solid foundation for further improvements. With additional data diversity and advanced segmentation techniques, the model can be enhanced to handle more complex real-world scenarios.

Sample Images:





180307 51□18-140
MAURYA BOYZ
EYEWEAR