EYE GLASS SEGMENTATION PROJECT ASSIGNMENT

Name: Naveen krishna Makkena

Email: naniwtu@gmail.com

GitHub link: https://github.com/kishorsumathi/mask_rcnn_codebugged_AI

Project Overview:

Objective: Develop a machine learning solution for image segmentation to accurately segment eyeglasses from images.

Challenges: Address variations in lighting, scale, and orientation; ensure smooth edges and a white background for segmented output.

Technical Stack:

Programming Language: Python

Segmentation Models: U-Net (due to its effectiveness in biomedical image segmentation)

Libraries and Tools: TensorFlow for model building, OpenCV for image processing

Steps Involved:

Data Preparation: From the given dataset containing images of eyeglasses on various backgrounds, under different lighting conditions, scales, and orientations.

Organize the dataset into training, validation, and test sets.

Ensure that each input image has a corresponding ground truth mask image indicating the segmentation of eyeglasses.

Data Preprocessing:Resize images to a uniform size suitable for training the U-Net model.

Normalize pixel values to a range between 0 and 1.

Optionally, augment the training data with techniques like rotation, flipping, and shifting to increase the diversity of training samples.

Model Building: Define the architecture of the U-Net model and compile the model with appropriate loss function (e.g., binary cross-entropy) and optimizer (e.g., Adam).

Model Training: Train the U-Net model using the prepared dataset. Split the dataset into batches and feed them to the model for training. Then monitor the training progress by observing metrics such as loss and accuracy on both training and validation sets. Adjust hyperparameters if necessary to improve performance.

Post-processing: Apply post-processing techniques to improve segmentation quality, such as smoothing edges and ensuring a white background. Use OpenCV or similar libraries to refine the segmentation masks and remove noise if present.

Model Evaluation: Evaluate the trained model on the test set to assess its performance in segmenting eyeglasses from unseen data. Compute metrics such as accuracy, precision, recall, and F1-score to quantify the model's performance.

Result Analysis: Analyse the segmentation results visually to ensure that the model accurately segments eyeglasses while maintaining smooth edges and a white background. Identify any areas where the model may be performing poorly and consider fine-tuning or retraining the model if necessary.

Deployment: Once satisfied with the model's performance, deploy it for real-world applications. Integrate the model into a production pipeline or application where it can be used to segment eyeglasses from input images.

Code Implementation:

Segmentation Algorithm: Implemented using U-Net architecture.

Precision and Smooth Edges: Achieved through careful design of the convolutional network and post-processing techniques.

White Background: Post-segmentation processing to isolate the eyeglasses and change the background to white.

Model Selection and Training:

Model Choice: U-Net was selected for its architecture specifically designed for segmentation tasks.

Retraining: The model was retrained on a dataset of eyeglass images using TensorFlow.

Training Details: The model was trained for 50 epochs with a batch size of 32, using Adam optimizer and binary cross-entropy loss.

Conclusion:

The developed segmentation solution meets the project's objectives by providing high-precision segmentation of eyeglasses from images, accounting for variations in conditions, and ensuring smooth edges with a white background in the output.