Project 1: Image Manipulation and Background Replacement

Abhay Chaudhary

Objective

The goal of this project is to manipulate a set of car images by performing background replacement, realistic shadow placement, and perspective matching. We were provided with car images, background masks, and shadow masks. The task was to recreate the provided sample output by:

- **Replacing the Background:** The provided background removal masks contain noise, which needs to be handled appropriately. The car's background should be replaced with given images of a floor and a wall.
- **Placing Shadows:** Using shadow mask cutouts, realistic shadows should be placed beneath the car. The shadows must align with the car's position and lighting conditions to ensure realism.
- **Perspective Matching:** Ensuring that the perspective between the car, floor, and background is properly aligned for a natural and realistic final composition.

Tasks

1. **Background Replacement:**
    - The car image background was replaced using a composite of two images: a wall and a floor.
    - Noise in the provided background removal masks was handled using morphological operations like closing and opening to clean up the mask.
    - We used image processing methods such as thresholding and resizing to ensure the car image was seamlessly integrated with the new background.
2. **Shadow Placement:**
    - Realistic shadows were created using the shadow masks provided.
    - The shadows were scaled and positioned under the car using offsets, with a scaling factor applied to match the perspective.
    - The intensity and appearance of the shadows were controlled to ensure they blended with the lighting conditions of the scene.
3. **Perspective Matching:**
    - The perspective was adjusted by aligning the car image and floor to ensure the composition was realistic.
    - The `resize_and_position_mask` function was designed to handle resizing and repositioning of masks with offsets and scale factors, ensuring correct alignment.

Approach

1. **Image Preprocessing:**
    - We loaded the car image, car mask, and shadow mask using OpenCV.
    - The car mask was processed with morphological operations to clean any noise and produce a clear boundary for the car.
2. **Background Creation:**
    - The background was created by combining two images: a wall and a floor.
    - The floor and wall were blended at a specific split point (60% of the image height), ensuring proper positioning.
3. **Shadow Application:**
    - Shadows were added under the car by applying the resized shadow mask with controlled opacity.
    - The shadow mask was scaled based on the perspective.
4. **Final Composition:**
    - The car image was masked and combined with the background. The cleaned car mask was used to isolate the car, while the shadow mask was applied to create natural-looking shadows.

Challenges Faced

1. **Handling Noisy Background Masks:**
    - The background removal masks contained noise that required careful handling. We applied morphological operations like closing and opening to clean up the mask.
    - The mask processing pipeline had to handle imperfections such as overlapping boundaries or incomplete regions in the mask.
2. **Shadow Positioning and Scaling:**
    - Properly aligning the shadow mask with the car required precise control over offsets and scaling factors. Any misalignment would result in unrealistic shadows.
    - We implemented custom logic to ensure that shadow placement remained consistent and proportional to the car's perspective.
3. **Perspective Matching:**
    - Aligning the car, shadow, and background perspective was a critical aspect of this project. Any mismatch in scale or position would break the illusion of realism.
    - To address this, we carefully adjusted the scaling of the shadow and the background floor to match the perspective of the car.

Results

- The final output was a realistic composition of the car with the new background and shadows.
- The shadows were seamlessly integrated, appearing realistic in both scale and intensity.
- The perspective was carefully matched, creating a cohesive and natural-looking final image.

Code Implementation

The core Python code was written using OpenCV and NumPy for image manipulation. Below is an overview of the key components of the code:

1. **Image Preprocessing:**
   - We read the car image, car mask, and shadow mask using `cv2.imread()`.
   - Noise in the car mask was cleaned using morphological operations: closing and opening.
   - Masks were resized and repositioned using the `resize_and_position_mask()` function to match the perspective.
2. **Background Creation:**
   - We read the floor and wall images, resized them, and blended them to create a composite background using the `create_background()` function.
   - The floor-to-wall ratio was adjusted to ensure proper perspective matching.
3. **Final Composition:**
   - Shadows were placed under the car by applying the resized shadow mask with controlled opacity.
   - The car image was blended with the background using the cleaned mask to isolate the car, ensuring a seamless final composition.
4. **Saving and Displaying Results:**
   - The final composition was saved as a JPEG file using `cv2.imwrite()`.
   - Optionally, the image was displayed using `cv2.imshow()` for real-time verification.