Automated Low-Light Car Image Enhancer and Car Detector for Legal and Forensic Applications

Team Members:
Skanta Samvartan R-3122225001137
Vijaya Prasath L S-3122225001159
Vineeth U-3122225001160
Vishal K-3122225001161

Introduction:-

In this document, we present a simple approach to car detection in low-light environments, designed to enhance image clarity and improve object recognition for applications in legal and investigative scenarios. Detecting vehicles in challenging lighting conditions poses significant technical hurdles due to issues like poor visibility, shadowing, and image noise. Our approach integrates a series of image enhancement techniques - such as bilateral filtering, gamma correction, CLAHE (Contrast Limited Adaptive Histogram Equalization), and sharpening - to increase image brightness and clarity. These preprocessing methods are followed by the application of the YOLO (You Only Look Once) object detection algorithm, particularly YOLOv8n, optimized for identifying cars in low-visibility settings. The combination of enhancement and detection processes offers a robust framework for accurately identifying cars, aiding in legal investigations, surveillance, and other security-related fields.

Car Recognition methodology:-

Image Preprocessing Techniques

Low-light conditions introduce noise, reduce contrast, and obscure details, making preprocessing crucial for effective car detection. Here, several enhancement techniques improve image quality and clarity before detection.

- **Bilateral Filtering**: This edge-preserving filter reduces noise while retaining edges. It smooths regions with minor pixel variations but maintains sharpness around edges, enhancing object boundaries without introducing blur.
- **Gamma Correction**: Low-light images often suffer from low contrast, where details are hidden in darker regions. Gamma correction adjusts image intensity, brightening dark areas without saturating highlights, making it easier for detection algorithms to recognize shapes and edges.
- **Contrast Limited Adaptive Histogram Equalization (CLAHE)**: CLAHE boosts contrast by enhancing local regions within an image. It applies histogram equalization but limits amplification in brighter areas, preventing artifacts while highlighting faint details, especially in underexposed areas.
- **Sharpening**: By increasing the contrast around object edges, sharpening enhances the definition of objects in low-light images, making contours more distinct and detectable.

---

2. Object Detection Using YOLO (You Only Look Once)

YOLO is a real-time object detection model known for its high accuracy and speed. It processes an image as a whole, enabling it to detect multiple objects simultaneously.

Image Grid Division and Bounding Box Prediction

- YOLO divides the input image into a grid (e.g., an 8x8 grid), where each cell in the grid is responsible for predicting several bounding boxes.
- For each bounding box, YOLO predicts:

- The **coordinates of the bounding box**: These are represented by xxx, yyy, width www, and height hhh, where xxx and yyy are the center coordinates relative to the grid cell.
- The **confidence score**: This score reflects the model's confidence that the bounding box contains an object and that it correctly identifies the object type.
- **Class probabilities**: YOLO outputs probabilities for each class (e.g., car, truck, pedestrian), allowing the model to identify the specific object type within each bounding box.

Convolutional Neural Network (CNN) Architecture

- YOLO's CNN is designed to extract features from the entire image, allowing the model to understand spatial relationships. This approach is beneficial for low-light images, where partial shadows or blurred outlines may obscure object boundaries.
- The CNN architecture in YOLO learns spatial information through multiple convolutional layers that progressively refine feature extraction. This feature extraction process is robust to noise and distortions typical in low-light images, allowing YOLO to detect cars even if parts of the image are shadowed or underexposed.

Class Prediction and Confidence Scores

- Each grid cell can predict multiple bounding boxes with associated confidence scores and class probabilities. However, only boxes with confidence scores above a specified threshold are retained to minimize false positives.
- In a low-light setting, adjusting this confidence threshold is essential, as a lower threshold may include boxes around noise or unimportant features, while a higher threshold might miss faint objects. Careful tuning of this threshold improves the model's ability to focus on actual cars while ignoring irrelevant elements.

Non-Max Suppression (NMS) for Filtering Overlapping Boxes

- Non-Max Suppression is applied after initial predictions to remove redundant bounding boxes that overlap significantly.
- During NMS, bounding boxes with high overlap (based on an Intersection over Union or IoU score) and lower confidence are suppressed, leaving only the bounding boxes with the highest confidence scores.
- This step is particularly useful for scenarios where multiple overlapping detections might occur, such as crowded environments or images with reflective surfaces. NMS ensures that only the most c
-

---

3. Post-Detection Color Analysis Using K-Means Clustering

After detecting cars, we apply color analysis to extract dominant colors within each detected bounding box. This process aids in recognizing car color, which is valuable in investigative scenarios where visual identification plays a role.

- **Color Region Extraction**: After bounding boxes are generated by YOLO, each box is cropped to isolate the car region. This allows precise color analysis only within the detected vehicle, reducing interference from background or other objects.
- **K-Means Clustering for Dominant Colors**: The car region is analyzed using the K-Means clustering algorithm, which groups pixels with similar RGB values into clusters. Each cluster represents a dominant color within the car region.
    - **Cluster Centroids**: The RGB values of cluster centroids represent the main colors found in the car, allowing us to identify the predominant hues present.
    - **Cluster Size as Dominance Indicator**: By measuring the size of each cluster (number of pixels it includes), we rank colors based on their dominance in the image. The most prominent clusters indicate the vehicle's main colors.

- **HSV Conversion and Color Naming**: RGB values are converted to HSV (Hue, Saturation, Value) format, which is more intuitive for determining perceptual color names (e.g., red, blue, gray). We assign labels based on predefined HSV ranges corresponding to familiar color names, allowing us to describe the car's color in easily understandable terms.

---

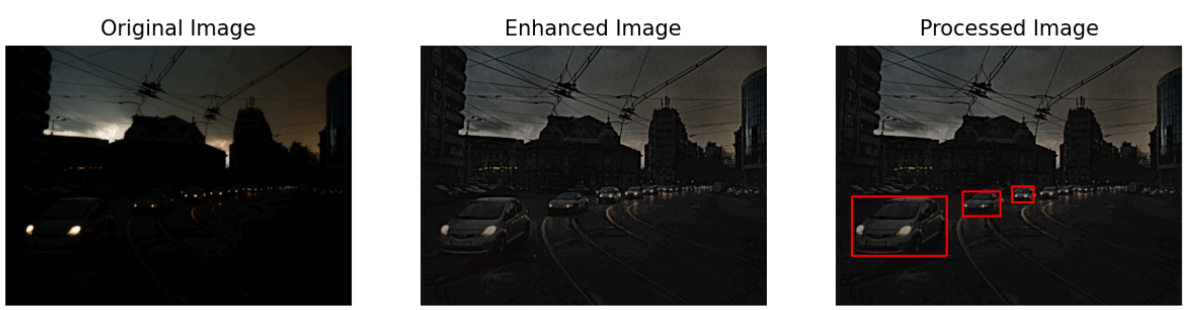
4. Integrating Detection and Analysis for Enhanced Forensic Utility

The combination of detection and color analysis in this pipeline allows for a comprehensive approach to vehicle identification in low-light conditions:

- **Bounding Box Output**: Each detected car is surrounded by a bounding box, with information about its dominant colors, creating a visually rich output that highlights both position and color characteristics.
- **Multi-Object Handling**: YOLO's multi-object detection capability ensures that even if multiple cars are present, each is detected independently, and color analysis is applied separately to each detected instance.
- **Detailed Reporting**: The output includes the coordinates of each bounding box and a list of dominant colors with their proportions. This data supports further analysis or matching with witness descriptions.

The image enhancement techniques and filters we've applied aim to improve the visibility and detail of low-light car images, making them suitable for legal purposes such as identifying key features like number plates and vehicle color. Here's an explanation of the methods used in your code:

Sample Input/Output Images:



Dataset Link: Image Processing Assignment