

Savitribai Phule Pune University



Department Of Technology

PG in Data Science & Artificial Intelligence

SEMESTER – II

Subject Name : Natural Language Processing

Project Name : Automating License Plate Recognition

Student Name: Dadaso Shivaji Patil

Roll No. : PGD23DS49

Project Participants : 1) Dadaso Shivaji Patil

Submission Date : 24/05/2024

Batch : June 2023

Internal Examiner

Signature

Course-Coordinator

Signature

AIM

The aim of this project is to create a comprehensive system that automates the detection and reading of license plates from car images using advanced image processing techniques and Optical Character Recognition (OCR). This project involves several critical steps that ensure the accurate extraction and visualization of readable text from license plates in the given images.

Objective

- To read and process car images from a specified folder.
- To detect the license plates in the images.
- To extract text from the detected license plates using OCR.
- To display the original images, detected license plates, and images with contours highlighting the license plates.

Sub-components

Image Reading: Read images from a specified folder.

Image Resizing: Resize images to a standard size.

Image Preprocessing: Convert images to grayscale and apply edge detection.

License Plate Detection: Identify potential license plate regions using contour detection.

OCR: Extract text from the detected license plate regions.

Visualization: Display the processed images and results using Matplotlib.

Task

Setup Environment: Import necessary libraries and initialize OCR reader.

Process Images: Implement the function to process each image.

Detect License Plates: Use contour detection to identify potential license plates.

Extract Text: Use EasyOCR to read text from the detected license plates.

Display Results: Visualize the original image, detected license plate, and image with contours.

OCR and Visualization

1) OCR

Extract text from the detected license plate regions using EasyOCR.

2) Visualization

Display the original images, detected license plates, and images with contours highlighting the license plates.

Code Section

Overview

This code segment is designed to complete the second phase of the license plate detection and recognition project. Specifically, it focuses on Optical Character Recognition (OCR) and visualizing the results. The tasks are handled as follows:

1. Import Libraries

```
import matplotlib.pyplot as plt
```

```
from easyocr import Reader
```

- `matplotlib.pyplot`: Used for visualizing images and plotting results.

- `easyocr.Reader`: Used to initialize the OCR reader for extracting text from images.

2. Initialize the OCR Reader

```
reader = Reader(['en'], gpu=False, verbose=False) # Change language as needed
```

- Initializes the OCR reader with English language support. GPU is set to `False`, and verbosity is turned off for cleaner output.

3. Define the `extract_text_and_visualize` Function

```
def extract_text_and_visualize(car, plate_cnt, gray):
```

```
    try:
```

...

This function extracts text from the detected license plate region and visualizes the results. It takes three parameters: the original image (`car`), the contour of the detected license plate (`plate_cnt`), and the grayscale version of the image (`gray`).

-Extract Text Using OCR:

if `plate_cnt` is not `None`:

`(x, y, w, h) = cv2.boundingRect(plate_cnt)`

`plate_roi = gray[y:y + h, x:x + w]`

`detection = reader.readtext(plate_roi)`

- Checks if a license plate contour is detected.
- Extracts the Region of Interest (ROI) for the license plate from the grayscale image.
- Uses EasyOCR to read text from the ROI.

- Handle OCR Results:

if `len(detection) == 0`:

`text = "Impossible to read the text from the license plate"`

`cv2.putText(car, text, (20, 40), cv2.FONT_HERSHEY_SIMPLEX, 0.75, (0, 0, 255), 3)`

else:

`cv2.drawContours(car, [plate_cnt], -1, (255, 0, 0), 3)`

`text = f"{detection[0][1]} {detection[0][2] * 100:.2f}%"`

`cv2.putText(car, text, (x, y - 20), cv2.FONT_HERSHEY_SIMPLEX, 0.75, (0, 255, 0), 2)`

`print(text)`

- If no text is detected, displays a message on the image.
- If text is detected, draws the contour on the image and overlays the detected text with its confidence score.

- Visualize Results:

`plt.figure(figsize=(12, 6))`

```
plt.subplot(1, 3, 1)

plt.imshow(cv2.cvtColor(car, cv2.COLOR_BGR2RGB))

plt.title("Image")
```

if plate_cnt is not None:

```
    plt.subplot(1, 3, 2)

    plt.imshow(cv2.cvtColor(plate_roi, cv2.COLOR_BGR2RGB))

    plt.title("License Plate")
```

```
plt.subplot(1, 3, 3)

plt.imshow(cv2.cvtColor(car, cv2.COLOR_BGR2RGB))

plt.title("Image with Contours")
```

```
plt.show()
```

- Sets up a figure with three subplots using Matplotlib.
- Displays the original image, the extracted license plate, and the image with contours highlighting the license plate.

4. Process and Visualize All Images in the Folder:

for filename in os.listdir(image_folder):

if filename.lower().endswith(('.jpg', '.jpeg', '.png')):

image_path = os.path.join(image_folder, filename)

car, plate_cnt, gray = process_image(image_path)

if car is not None and plate_cnt is not None:

extract_text_and_visualize(car, plate_cnt, gray)

- Loops through each file in the specified image folder.
- Checks if the file is an image based on its extension.
- Processes the image to detect the license plate (handled by Student 1).
- Extracts text from the detected license plate and visualizes the results (handled by Student 2).

This code completes the second phase of the project by leveraging OCR to read text from detected license plates and visualizing the results. The process includes initializing the OCR reader, defining the function for text extraction and visualization, and processing all images in the specified folder to display the final outcomes. This segmentation ensures a collaborative approach where Student 1 prepares the images and Student 2 handles the OCR and visualization.

Output



Overall Conclusion

The project achieved its primary goal of automating the detection and reading of license plates from car images. By leveraging image processing techniques and OCR, the system can efficiently process and analyze images to extract readable text from license plates. The collaborative approach ensured a robust and well-rounded solution, demonstrating the effectiveness of dividing tasks for improved efficiency and clarity. This project lays a solid foundation for further enhancements, such as improving OCR accuracy and handling more complex image scenarios.