People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research

University of Abdelhamid Mehri – Constantine 2

Faculty of New Technologies of Information and Communication (NTIC)

Department of Softwae Technologies and Information Systems (TLSI)

PROJECT

Option: Data Science and Information System (SDSI)

System Architecture for Automatic Number Plate Recognition

Realized by:

Adrar Ahlem Lamia

Bousbia Mohamed Akrem

Ziari Nazim

Menaifi Maya Fairouz

Under supervision of:

Dr. Naila Marir

Dr. Ghammaz Wafa

2024

Introduction

The ANPR system aims to recognize number plates from annotated images, process the data using Big Data and Machine Learning techniques, and provide business intelligence insights through visualization tools.

Step 1: Ingestion Layer

In this phase, data is collected from various sources such as satellites, radar cameras, drones, lidar sensors, datasets, and CSV files. These data are then imported into a data warehouse for further processing.

Data Sources

Collect raw data on vehicles and their license plates, continuously and in real-time, from various points. Large volumes of data are generated from different types of devices. These devices are connected to the central data source where they transfer raw data. The data source is responsible for storing structured, semi-structured, and unstructured data in raw format. Data from this warehouse is transferred to both the "Batch" and "Speed" layers for different types of data processing.

Central point for ingestion and initial storage of data, with a distinction between real-time data (sent to Kafka in the "Speed" layer) and other data (stored in Hadoop HDFS in the "Batch" layer).

Hadoop HDFS Configuration

To store annotated image data acquired from the dataset.

Kafka

Message queue for real-time data, providing high availability, low latency, and efficient data stream management.

Step 2: Processing Layer

In this phase, data is processed both in batch and real-time to detect characters in vehicle license plates.

Batch Layer (Spark Batch)

Preprocesses data stored in HDFS, including image preprocessing and training of the character detection model using Spark Batch.

Model Learning

We will apply distributed training techniques using Apache Spark and Spark MLlib for machine learning tasks on our dataset . Spark Task Development: Manages preprocessing operations in parallel across the cluster.

Streaming Layer (Spark Streaming)

Processes real-time data from Kafka for real-time preprocessing and deployment of the character detection model using Spark Streaming.

Step 3: Storage Layer

In this phase, processed data is stored for further analysis.

Storage in HBase

For real-time processed data, offering horizontal scalability and fast data retrieval.

Storage in Redis

Stores data in fast memory for quick retrieval of processed data for real-time analytics.

Data Analysis

Once the data is stored, it can be analyzed using Power BI and a web application to gain insights into character detection in vehicle license plates.

Figure 0.1: pic

Conclusion

This architecture ensures efficient data management at all stages of the process, from initial storage to final analysis, ensuring optimal scalability, availability, and performance.