BDD100K Data Management

Efficient Schema Design and Query Optimization

Alexander Schad
Jonathan Jacobs
DATA 514

Project Overview

This project focuses on designing a comprehensive database schema for the BDD100K dataset, an extensive and varied driving video database created to support computer vision research for autonomous driving. The dataset includes 100,000 video clips, each 40 seconds long, recorded at 720p and 30 fps, capturing diverse driving scenarios across different locations, weather conditions, and times of day in the United States.

Our aim is to create a manageable subset of this dataset to facilitate efficient data access and use. This will involve constructing entities for videos, annotations, objects, and trajectories, ensuring well-defined relationships and optimized performance for future analysis.

Objectives

1. **Database Schema Design**: Develop a robust schema that accurately represents the diverse data within the BDD100K dataset, ensuring well-defined and optimized relationships between entities such as videos, annotations, objects, and trajectories.
2. **Data Governance**: Identify and address key data governance issues, including managing sensitive information like GPS data and video footage, and propose measures to protect this information and ensure compliance with privacy laws.
3. **Query Development**: Write and execute SQL queries to demonstrate practical applications of the database schema, covering use cases such as retrieving metadata, counting object categories, and listing specific annotations.
4. **Query Optimization and Indexing**: Analyze and optimize the logical plans of our queries to improve performance, and identify appropriate indexes to support use cases and enhance query efficiency.
5. **DBMS Selection:** Recommend a suitable DBMS for managing the BDD100K dataset, considering performance, scalability, and feature support.

By achieving these objectives, we aim to develop a schema that enhances future users' ability to access and analyze the BDD100K data efficiently.

Dataset Description

BDD100K is a comprehensive driving video dataset designed to advance research in autonomous driving and multitask learning within computer vision. It contains 100,000 annotated video clips capturing a wide range of driving scenarios across diverse geographic locations, environmental conditions, and weather variations. This extensive dataset supports ten different tasks, including image tagging, lane detection, drivable area segmentation, road object detection, semantic segmentation, instance segmentation, multi-object detection tracking, multi-object segmentation tracking, domain adaptation, and imitation learning. By providing such a rich variety of tasks and scenarios, BDD100K enables researchers to develop and evaluate models that can handle the complexities of real-world driving situations, making it a vital resource across multiple tasks within computer vision.

The dataset is designed to overcome the limitations of existing driving datasets, which often lack sufficient scene variation, annotation richness, and geographic diversity. BDD100K facilitates the study of heterogeneous multitask learning, where models are required to perform a variety of tasks with different complexities. This aspect is crucial for developing autonomous driving systems that can generalize well to new and unexpected conditions. The benchmarks and evaluations provided with BDD100K offer insights into the challenges and strategies for effective multitask learning, highlighting the need for specialized training strategies to improve performance across multiple tasks. BDD100K represents a significant step forward in creating robust and versatile autonomous driving models, paving the way for future research and applications in the field.

The BDD100K dataset encompasses a comprehensive set of features to support a variety of tasks essential for autonomous driving. For object detection, the dataset includes frame identifiers, category IDs (covering 10 classes such as pedestrians, cars, and traffic signs), bounding box coordinates, and attributes like occlusion, truncation, and traffic light color. Instance segmentation, box tracking, and segmentation tracking share similar features but focus on the first eight categories, with additional details like segmentation masks, instance IDs, and specific attributes indicating crowd and ignore status. Semantic segmentation utilizes one-channel PNG images where pixel values represent one of 19 categories, while panoptic segmentation employs RGBA PNG images encoding category ID, instance attributes, and instance ID.

Additional tasks include drivable area segmentation, lane marking, and pose estimation. Drivable area segmentation differentiates between direct, alternative, and background areas using one-channel PNG images. Lane marking involves three sub-tasks: lane categories, directions, and styles, encoded in specific bits of a one-channel PNG image. Pose estimation features 18 keypoints representing various body joints. The dataset also includes frame-level attributes such as weather conditions, scene types, and time of day. To facilitate various segmentation tasks, the dataset supports both JSON and mask formats, and provides tools for format conversion, ensuring compatibility with multiple frameworks and evaluation tools.

Data Governance Issues

Sensitive Information

The BDD100K dataset contains various types of sensitive information, including GPS data, video footage, and personal identifiers. These data points could potentially be used to identify individuals or specific locations, posing significant privacy risks.

Protection Measures

To mitigate these risks, several protection measures should be implemented:

1. **Anonymization**: Personal identifiers and location details should be anonymized to prevent identification.
2. **Secure Storage**: Data should be stored in secure, encrypted databases with restricted access to authorized personnel only.
3. **Access Controls**: Implement role-based access controls to ensure that only individuals with the necessary permissions can access sensitive information.

Compliance with Data Privacy Laws

Compliance with data privacy laws, such as the General Data Protection Regulation (GDPR) and the California Consumer Privacy Act (CCPA), is essential. These laws mandate specific measures for data protection, user consent, and the right to data deletion. Ensuring that the dataset complies with these regulations is crucial to avoid legal repercussions.

Ethical Use of Data

Ethical considerations must be addressed to ensure the data is used responsibly:

1. **Informed Consent**: Data should be collected with informed consent from individuals captured in the videos, where feasible.
2. **Usage Limitations**: Clearly define and limit the scope of data usage to research and development of autonomous driving technologies.
3. **Transparency**: Maintain transparency about how the data is used, shared, and stored, providing clear information to stakeholders and the public.

Future Concerns

Future reuse of the dataset may raise additional concerns:

1. **Data Retention**: Establish clear policies on data retention, specifying how long data will be kept and under what conditions it will be deleted.
2. **Data Sharing**: Regulate data sharing practices to ensure that third parties adhere to the same data protection standards.
3. **Continuous Monitoring**: Implement continuous monitoring and auditing processes to ensure ongoing compliance with data protection standards and regulations.

Data Quality and Integrity

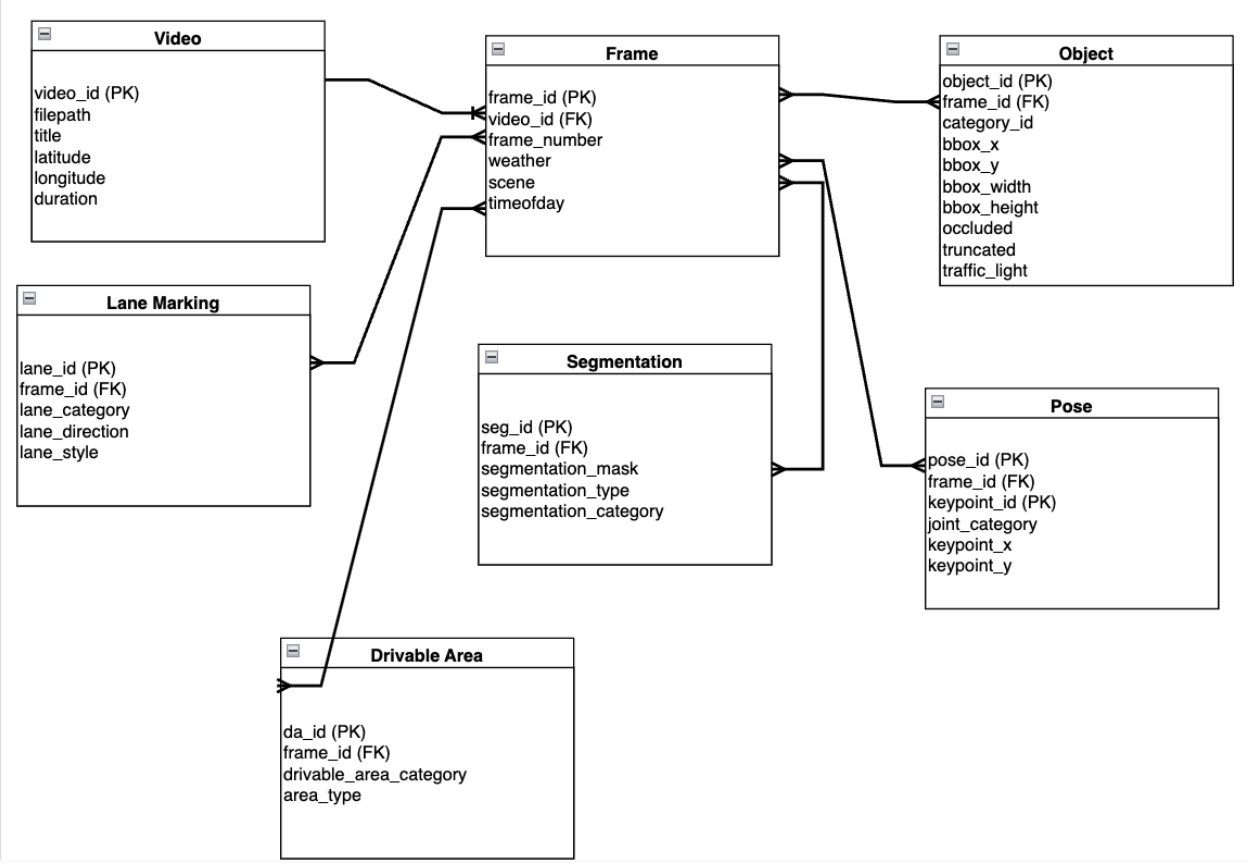
Maintaining high data quality and integrity is critical:

1. **Validation**: Regularly validate and update the data to ensure accuracy and completeness.
2. **Error Handling**: Establish protocols for identifying and correcting data errors or inconsistencies.

By addressing these data governance issues, we can ensure the responsible and ethical use of the BDD100K dataset, protecting individual privacy while supporting valuable research in autonomous driving.

Schema Design

ER Diagram



Use Cases

1. Counting the Number of objects for each frame in a given video

```sql
SELECT f.frame_number, COUNT(o.object_id) AS object_count
FROM Frame f
LEFT JOIN Object o ON f.frame_id = o.frame_id
LEFT JOIN Video v ON f.video_id = v.video_id
WHERE v.title = 'NYC June 1 2019'
GROUP BY f.frame_number;
```

2. Find the number of frames with a crosswalk in each video

```sql
SELECT v.video_id, COUNT(l.lane_id)
FROM Frame f
LEFT JOIN Video v ON f.video_id = v.video_id
LEFT JOIN LaneMarking l ON f.frame_id = l.frame_id
WHERE l.lane_category = 1
GROUP BY v.video_id;
```

3. Finding the number of frames that are available for every time of day and weather combination

```sql
SELECT f.weather, f.timeofday, COUNT(*)
FROM Frame f
LEFT JOIN Video v ON f.video_id = v.video_id
GROUP BY f.weather, f.timeofday;
```

Query Logical Plan Analysis

Use Case 1: Counting the Number of Objects for Each Frame in a Given Video

Naive RA Tree:

1. Selection: Filter `Video` by `title = 'NYC June 1 2019'`
2. Join: Merge `Frame` with filtered `Video` on `video_id`
3. Join: Merge `Frame` with `Object` on `frame_id`
4. Projection: Extract `frame_number` and `object_id`
5. Aggregation: Count `object_id` grouped by `frame_number`

Optimized RA Tree:

1. Selection: Filter `Video` by `title = 'NYC June 1 2019'`
2. Join: Merge `Frame` with filtered `Video` on `video_id`
3. Projection: Extract `frame_id` and `frame_number` from `Frame`
4. Join: Merge `Frame` with `Object` on `frame_id`
5. Aggregation: Count `object_id` grouped by `frame_number`

Performance Differences:
Performing the initial selection and join early reduces the number of rows processed in subsequent steps, enhancing efficiency.

Diagrams included with submission: "*Use Case 1 Naive.pdf*", "*Use Case 1 Optmized.pdf*"

Use Case 2: Find the Number of Frames with a Crosswalk per Video

Naive RA Tree:

1. Selection: Filter `LaneMarking` by `lane_category = 1`
2. Join: Merge `LaneMarking` with `Frame` on `frame_id`
3. Join: Merge `Frame` with `Video` on `video_id`
4. Projection: Extract `video_id` and `lane_id`
5. Aggregation: Count `lane_id` grouped by `video_id`

Optimized RA Tree:

1. Selection: Filter `LaneMarking` by `lane_category = 1`
2. Join: Merge `LaneMarking` with `Frame` on `frame_id`
3. Projection: Extract `frame_id` from `Frame`
4. Join: Merge `Frame` with `Video` on `video_id`
5. Aggregation: Count `lane_id` grouped by `video_id`

Performance Differences:
Early selection and join steps reduce intermediate result sizes, improving query performance.

Diagrams included with submission: "*Use Case 2 Naive.pdf*", "*Use Case 2 Optmized.pdf*"

Use Case 3: Finding the Number of Frames Available for Every Time of Day and Weather Combination

Naive RA Tree:

1. Projection: Extract `weather`, `timeofday`, and `frame_id` from `Frame`
2. Aggregation: Count `frame_id` grouped by `weather` and `timeofday`

Optimized RA Tree:

1. Projection: Extract `weather`, `timeofday`, and `frame_id` from `Frame`
2. Aggregation: Count `frame_id` grouped by `weather` and `timeofday`

Performance Differences:
Both plans are similar, but ensuring the use of indexes on `weather` and `timeofday` can enhance performance.

Diagrams included with submission: "*Use Case 3 Naive.pdf*", "*Use Case 3 Optmized.pdf*"

Indexes

- **Weather and Time of Day Index:** Index on `weather` and `timeofday` columns in the `Frame` table to support the third query.
- **Frame ID Index**: Index on `frame_id` column in the `Object` and `LaneMarking` tables to support the first and second queries.

By incorporating these queries and their logical plans into the analysis, we can further ensure efficient data retrieval and processing within the database schema.

DBMS Choice

PostgreSQL

The ER diagram is well-suited to a relational database because the original nested JSON format made it difficult to access specific annotation types, requiring traversal through individual frames. By extracting annotation data from frame objects, we avoid the need for a JSON database like MongoDB. A relational database is ideal for computer vision tasks, as it allows efficient access to task-specific annotations.

Relational databases are preferred for the BDD100K dataset due to their ability to efficiently manage structured data with clearly defined relationships, enabling complex queries and joins essential for retrieving and analyzing related data across multiple tables. They ensure data integrity and consistency through ACID properties, offer advanced indexing for improved query performance, and support scalability for large datasets. Overall, relational databases provide a robust framework for managing the BDD100K dataset's structured, interconnected data, crucial

for computer vision tasks.

Summary

The final project for DATA 514 involves designing a database schema for the BDD100K dataset, a comprehensive driving video database for autonomous driving research. Our objectives include creating a robust schema, addressing data governance issues, developing and optimizing SQL queries, and selecting an appropriate DBMS. The dataset features diverse annotations for various computer vision tasks. We focus on ethical data usage, compliance with privacy laws, and ensuring data quality. Optimized query plans and appropriate indexing strategies enhance data retrieval efficiency.

PostgreSQL is chosen for its robust relational capabilities and scalability. It efficiently handles complex queries and joins, supports advanced indexing for faster data retrieval, and adheres to ACID properties ensuring data integrity. PostgreSQL's scalability features, such as parallel query execution and partitioning, make it suitable for managing the extensive BDD100K dataset. Additionally, its support for JSON data provides flexibility for future extensions. The strong community and range of extensions further enhance its functionality, making PostgreSQL an ideal choice for this project.