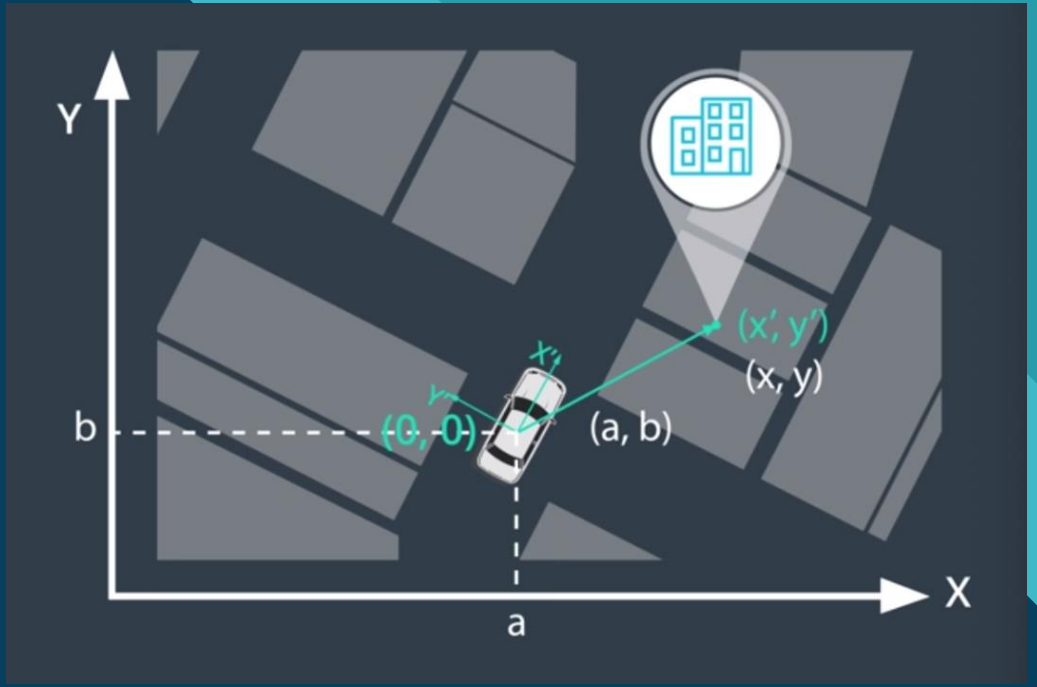
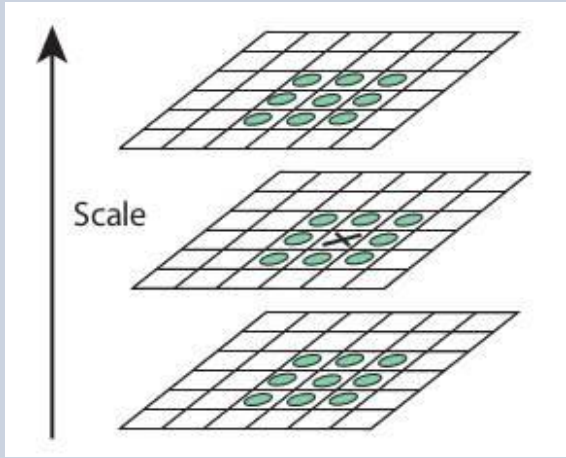
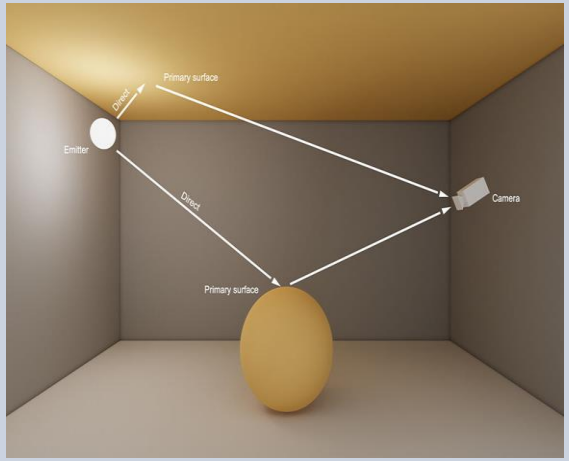
Vehicle Shift Estimation (CPU vs GPU)

Project Goal: 2D Position Estimation

- Accurately determine the 2D position and orientation (pose) of a vehicle from a video sequence.
- Focus on real-time tracking using computer vision techniques.
- Utilize feature-based matching to estimate movement between frames.
- Provide a robust solution that can handle common visual challenges.



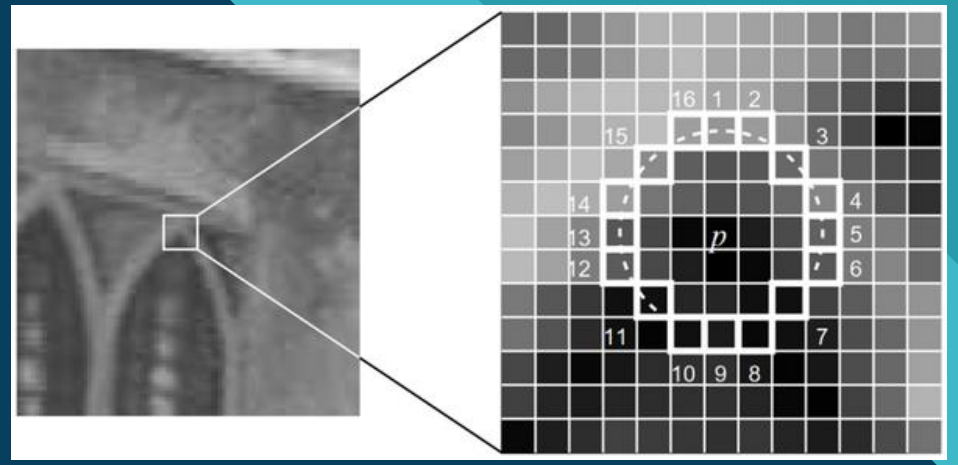
Challenges in Vehicle Tracking

Scale & Rotation	Occlusion	Illumination Changes	Motion Blur
Vehicles appear at different sizes and angles depending on their distance and orientation relative to the camera.	Other objects (pedestrians, other cars, buildings) can partially or fully block the view of the target vehicle.	Variations in lighting, such as shadows, sunlight, and nighttime, can drastically alter the vehicle's appearance.	Rapid vehicle movements can cause motion blur, making feature detection difficult and unreliable.
			

Overview of ORB Feature Matching

ORB (Oriented FAST and Rotated BRIEF) is a high-speed, robust local feature detector and descriptor. It is an efficient, open-source alternative to proprietary algorithms like SIFT and SURF, making it ideal for real-time applications.

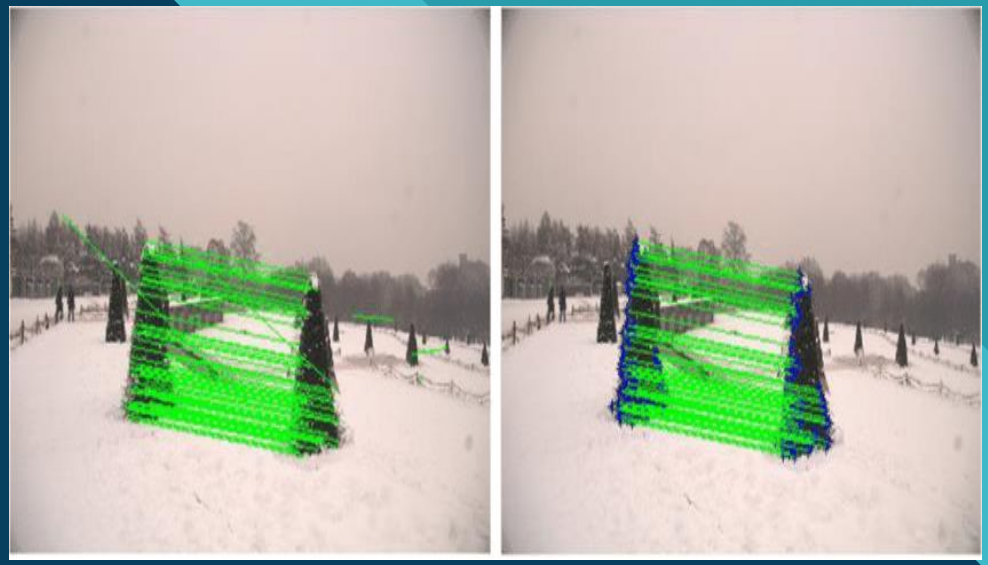
- FAST Keypoint Detector: Identifies points of interest (corners) very quickly.
- BRIEF Descriptor: Creates a compact binary string to describe each feature.
- Rotation & Scale Invariance: Handles changes in vehicle orientation and distance.



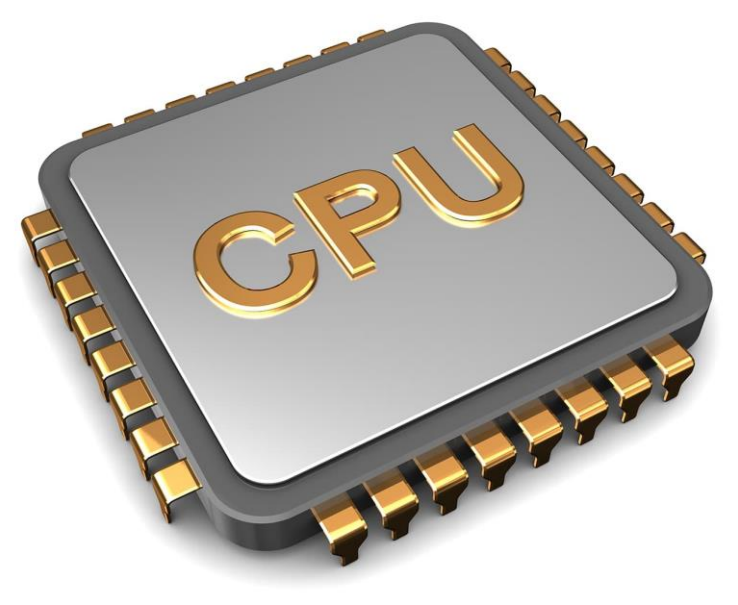
Understanding RANSAC for Robust Estimation

RANSAC (Random Sample Consensus) is an iterative method used to estimate parameters of a model from a dataset containing outliers. When matching features, many matches can be incorrect. RANSAC helps find the true motion by identifying the 'consensus' among the good matches.

- Robust to Outliers: Ignores incorrectly matched feature points.
- Improves Accuracy: Finds the best transformation model that fits the most 'inlier' matches.
- Iterative Process: Randomly samples data to find the optimal solution.



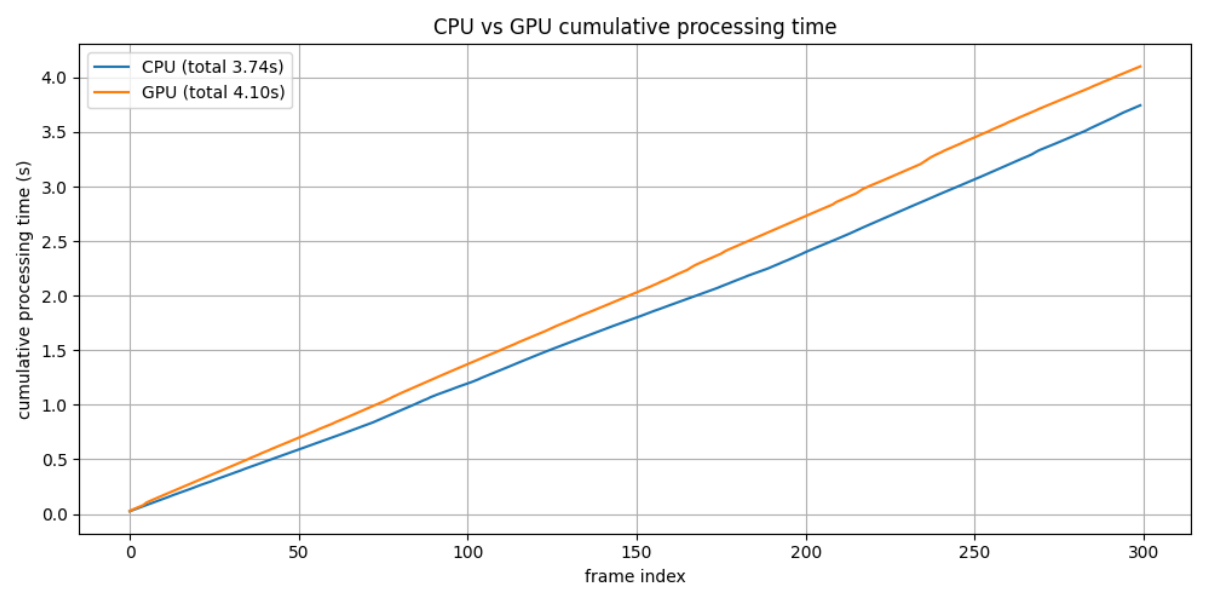
Implémentation Details: CPU vs. GPU

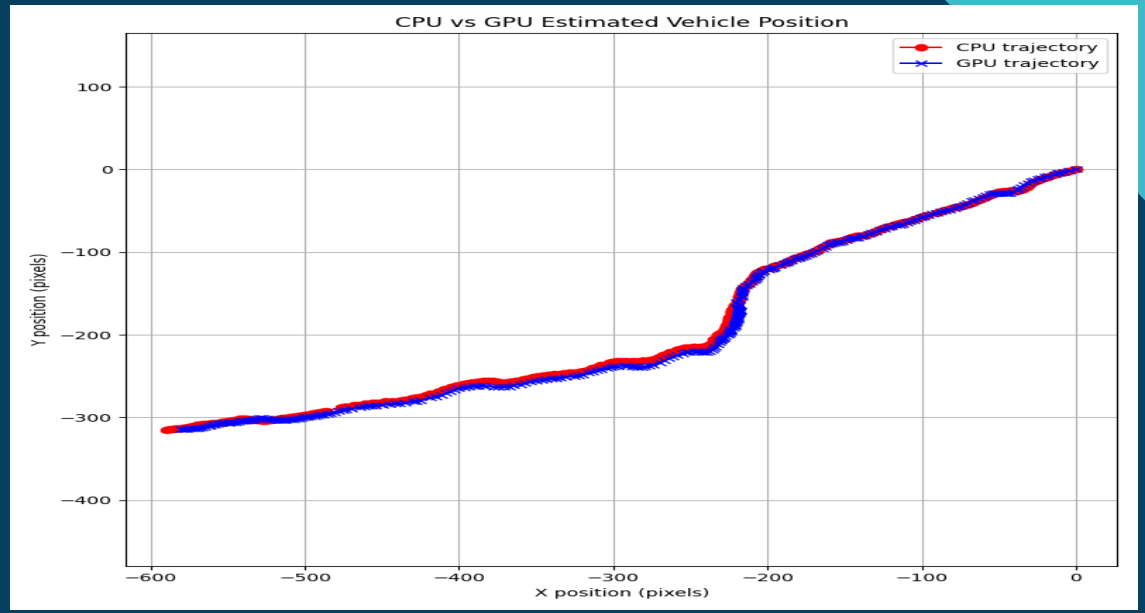
CPU Implementation	GPU Acceleration
The entire pipeline—frame capture, ORB feature detection, RANSAC, and transformation—is executed on the Central Processing Unit. This approach is straightforward but often limited by the CPU's sequential processing nature. 	Key computational tasks, especially feature matching and outlier rejection, are offloaded to the Graphics Processing Unit to leverage its massive parallel processing power. This is essential for achieving real-time performance. 

Performance Comparison: CPU vs. GPU

Although the GPU version processes feature detection and matching in parallel, its total runtime was slightly slower than the CPU version due to data transfer delays between CPU and GPU memory, especially for the non-GPU-accelerated affine estimation. The GPU advantage may be clearer with larger videos, higher-resolution frames, or more features to track.

Both pipelines estimated similar per-frame translations and cumulative positions. The CPU trajectory showed gradual, consistent motion with 800–1200 inlier matches per frame, indicating stable tracking. The GPU trajectory had slightly larger translations and more inlier matches (1200–1450), suggesting the GPU-accelerated ORB detected more features, yielding slightly different but consistent motion estimates.





Thank You