Motion-Aware Vision-Based Drone Navigation Using Weighted Velocity Tracking and Occupancy Prediction

Samrakshyan Chapagain
University of Texas at Arlington
Computer Science Department

June 24, 2026

Abstract

This project presents a real-time monocular vision-based drone navigation system that operates without LiDAR or stereo depth sensors. The system integrates motion detection, centroid-based tracking, weighted velocity estimation, trajectory prediction, and occupancy grid modeling to generate safe navigation corridors in dynamic environments.

A key contribution of this work is a motion-consistency-based object grouping strategy that combines spatial proximity and velocity similarity to maintain stable object identity under noisy and fragmented detections. Additionally, a weighted velocity formulation is introduced to stabilize motion estimation over time and reduce prediction jitter.

1 Introduction

Autonomous navigation is a fundamental problem in robotics and computer vision. Traditional solutions rely heavily on depth sensors such as LiDAR or stereo cameras, or on deep learning-based object detection systems. While effective, these approaches often require significant computational resources or specialized hardware.

This project explores a lightweight alternative using only a monocular RGB camera. The system is designed to perform real-time perception and decision-making using classical computer vision techniques combined with motion-based reasoning.

The primary objective is to construct a pipeline that can detect motion, track objects, predict their future positions, and estimate collision risk in real time.

2 System Overview

The system follows a modular pipeline architecture:

Camera → Motion Detection → Tracking → Velocity Estimation → Prediction → Occupancy Grid → Decision

Each module processes intermediate representations of the scene and passes structured information downstream.

3 Motion Detection

Motion detection is performed using classical frame differencing techniques. Each frame is converted to grayscale and smoothed using a Gaussian filter to reduce noise. The system then computes the absolute difference between consecutive frames to extract regions of motion.

These regions are thresholded and converted into binary masks, from which contours are extracted. Each contour is represented as a bounding box of the form (x, y, w, h).

This approach allows the system to isolate dynamic objects in the scene without requiring pretrained models.

4 Centroid-Based Object Tracking

Each detected bounding box is converted into a centroid representation defined as:

$$C = \left(x + \frac{w}{2}, y + \frac{h}{2} \right)$$

Object identity is assigned by minimizing the Euclidean distance between current and previous centroids:

$$D(i, j) = \|C_i - C_j\|$$

This allows the system to maintain temporal consistency of objects across frames.

However, centroid-based tracking assumes that spatial proximity alone is sufficient to determine identity. This assumption often fails in cases involving occlusion, fast motion, or fragmented detections.

5 Track Memory System

To support temporal reasoning, each object maintains a history of its positions, timestamps, and bounding boxes. This historical data is used for velocity estimation and trajectory prediction.

The track memory enables the system to move beyond frame-by-frame analysis and incorporate temporal dynamics into decision-making.

6 Weighted Velocity Estimation

A major limitation of traditional tracking systems is the instability of velocity estimation when computed using only two consecutive frames. Such approaches are highly sensitive to noise and can result in jittery or inconsistent motion predictions.

To address this, the system introduces a weighted velocity estimation method that incorporates multiple past motion observations.

Instead of relying on a single displacement, velocity is computed as a weighted sum of recent positional differences:

$$v = \sum_{i=1}^{n} w_i \cdot \Delta x_i$$

where Δx_i represents the displacement between consecutive frames and w_i represents the importance weight assigned to each displacement.

The weights are defined as:

$$w = [0.4,\ 0.3,\ 0.2,\ 0.1]$$

This formulation ensures that recent motion has a stronger influence on the final velocity estimate, while older motion provides stabilization rather than direct control.

The key intuition behind this approach is that object motion in real-world environments tends to evolve smoothly over short time intervals. By emphasizing recent motion while still considering historical behavior, the system achieves a balance between responsiveness and stability.

7 Motion-Based Object Grouping

A central contribution of this system is the introduction of motion-consistent object grouping. Traditional tracking systems rely primarily on spatial proximity to associate detections across frames. However, this can lead to fragmentation, especially when parts of the same object move independently or when detections are noisy.

To overcome this limitation, object identity is defined using both spatial and motion similarity. Two detections are considered to belong to the same object if:

$$\|C_1 - C_2\| < \tau_d \quad \text{and} \quad \|V_1 - V_2\| < \tau_v$$

Here, C represents centroid position and V represents velocity. The thresholds τ_d and τ_v control the strictness of grouping.

This formulation allows the system to construct motion-consistent object representations rather than treating each detection independently. As a result, fragmented detections such as limbs, partial occlusions, or motion-blurred regions can be unified into a single coherent object.

This improves tracking stability and produces more reliable trajectories for downstream prediction and risk estimation.

8 Trajectory Prediction

Object trajectories are predicted under the assumption of constant velocity over short time intervals. The future position of an object is computed as:

$$x_t = x_0 + v_x t, \quad y_t = y_0 + v_y t$$

This model is used to estimate short-term motion at multiple time horizons such as 1 second, 2 seconds, and 5 seconds.

9 Occupancy Prediction Model

Each predicted object position generates a dynamic risk zone. The radius of this zone is defined as:

$$r = r_0 + \alpha \cdot speed + \beta \cdot \frac{1}{depth}$$

This formulation captures uncertainty in motion prediction. Faster objects produce larger uncertainty regions, while closer objects contribute higher risk due to increased collision probability.

10 Occupancy Grid Representation

The environment is discretized into a two-dimensional occupancy grid. Each cell represents either free space or occupied space.

$$0 = \text{free space}, \quad 1 = \text{occupied space}$$

This representation enables spatial reasoning and global path analysis.

11 Corridor Generation

Safe navigation corridors are extracted by scanning the occupancy grid and identifying the widest continuous region of free space in the forward direction.

This corridor represents the safest available path for movement, given the current configuration of obstacles and predicted motion.

12 Decision Engine

The system computes a risk score based on occupancy density and corridor width. This score is used to determine the appropriate navigation action:

- Low risk: move forward
- Medium risk: slow down
- High risk: stop

13 Limitations

Despite its effectiveness, the system has several limitations. It does not incorporate true depth sensing and relies on monocular approximations. The centroid-based tracking approach may fail under heavy occlusion, and the constant velocity assumption may not hold under abrupt motion changes. Additionally, the system does not yet include learned object detection models.

14 Future Work

Future improvements include integrating Kalman filter-based tracking, monocular depth estimation using models such as MiDaS, and learning-based trajectory prediction methods. Graph-based motion clustering could further improve object grouping in complex scenes.

15 Conclusion

This project presents a lightweight monocular vision-based drone navigation system that integrates motion detection, weighted velocity estimation, motion-consistent object grouping, trajectory prediction, and occupancy-based decision-making.

The key contribution is a motion-aware identity formulation that improves tracking stability by combining spatial proximity and weighted motion similarity. This enables robust real-time navigation without reliance on expensive sensor systems.