UAV Onboard Dynamic Obstacle Detection & Avoidance System

1. Abstract

This project demonstrates a simulation system that enables Unmanned Aerial Vehicles (UAVs) to detect and autonomously avoid moving obstacles using onboard sensors and real-time computer vision. The UAV leverages cameras, motion prediction, and trajectory planning to reroute itself, eliminating reliance on any central ground controller.

2. Problem Statement

The proliferation of drones in cities and industrial areas increases the risk of mid-air collisions. Traditional centralized control systems introduce latency and can fail with lost connectivity. There is a clear need for fully onboard, real-time obstacle detection and avoidance, empowering UAVs to safely operate in dynamic environments independently.

3. Proposed Solution

The system employs:

- Simulated RGB-D cameras for depth sensing
- Optical flow (Lucas-Kanade method) to estimate movement
- Kalman filtering for object tracking
- Trajectory prediction to foresee potential collisions
- Autonomous rerouting: If a collision is predicted, the UAV adjusts its path without any external command.

4. System Architecture

Component	**Function**

Sensors	Simulated stereo/RGB-D cameras
Vision Processing	Optical Flow (Lucas-Kanade)
Tracking	Kalman Filter for position/velocity estimation
Trajectory Estimation	Predicts object and UAV future positions
Avoidance Logic	Plans and issues reroute maneuvers autonomously

This architecture is closely aligned with current research trends, which highlight the effectiveness of lightweight, vision-based approaches for small UAVs.

5. Methodology

- Optical Flow: Calculates pixel movements between video frames to detect and monitor object motion.
- Kalman Filter: Smooths and predicts the trajectory of detected obstacles, accounting for noise and uncertainty.
- Collision Prediction: Checks if projected UAV and obstacle trajectories will intersect.
- Reroute Planning: If risk is detected, the UAV initiates a vertical avoidance maneuver instantly, without central control intervention[41].

6. Implementation

- Language: Python 3.10
- Libraries: OpenCV, NumPy
- Input: Synthetic drone video file (`test_video.mp4`)
- Output: Real-time visualization with in-frame collision alerts and rerouting indications

7. Simulation Setup

- Scenario: One main (green) drone and a moving obstacle drone (red), simulated in a video file.
- The system predicts both drones' future positions and identifies possible collisions.
- Upon detecting a high collision probability, the main drone autonomously executes a vertical maneuver to avoid impact.

8. Results

- Real-time detection and tracking of dynamic obstacles.
- Instant reroute upon collision risk, with automatic execution.
- Clear visual alerts of avoidance actions.
- Standalone operation requiring no ground control or network connection.

9. GitHub Repository

All code, simulation scripts, flowcharts, the demo video, and documentation are available at:
https://github.com/Vps2905/uav_onboard_avoidance

10. Conclusion

This project confirms that onboard real-time vision-based detection and avoidance is feasible and practical for UAVs, offering a robust foundation for deploying autonomous drones in real dynamic environments.

11. Future steps

- Integration with real-world sensors such as LiDAR and radar.
- Scaling the solution for multi-UAV swarms with decentralized collision avoidance.
- Implementation of deep learning-based object identification.
- Integration with ROS2 for deployment on physical drones.

12. Folder Structure

```
uav_onboard_avoidance/
├── data/
```

```
│   └── test_video.mp4
├── src/
│   ├── optical_flow.py
│   └── kalman_tracker.py
├── utils/
│   ├── trajectory_predictor.py
│   ├── reroute_planner.py
│   └── helpers.py
├── images/
│   └── flowchart.png
├── create_moving_video.py
├── main.py
├── README.md
```