NASA Space Mission AI Project Plan

This presentation outlines a conceptual design for an AI system to optimize autonomous navigation for a NASA Mars rover. The project, led by Andrew Kuruvilla, focuses on real-time path planning and obstacle avoidance using machine learning and sensor data fusion. The goal is to enhance mission efficiency and safety in challenging Martian environments.

by Andrew Kuruvilla

Project Overview & Objectives

Project Description

The project proposes an AI solution for real-time path planning and obstacle avoidance on Mars. It leverages computer vision, reinforcement learning, and sensor fusion to help rovers adapt to unpredictable terrain and environmental changes.

Key Objectives

- Design AI model architecture for path planning
- Create a sensor fusion framework
- Develop a robust testing plan
- Align with NASA mission requirements

Defining the Problem

Autonomous Navigation Challenges

Mars rovers face communication delays and must navigate complex, unpredictable terrains with minimal human input.

Current Limitations

Existing systems rely on pre-programmed routes and lack real-time adaptability, risking inefficiency and mission failure when encountering unexpected obstacles.

Mission Risks

Obstacles like rocks, craters, and sand traps can jeopardize mission success if not detected and avoided in real time.

Proposed AI Solution

Dynamic Path Planning

Utilizes computer vision and reinforcement learning for real-time route optimization.

Sensor Data Fusion

Combines LiDAR, stereo cameras, and IMU data for robust obstacle detection and classification.

Environmental Adaptation

Adapts to changing conditions such as dust storms and lighting variations to ensure safe navigation.

Energy Optimization

Minimizes energy use while achieving mission objectives and reaching scientific targets.

System Architecture Overview

Data Acquisition

Collects and preprocesses LiDAR, camera, and IMU data using Python, OpenCV, and PCL.

Sensor Fusion

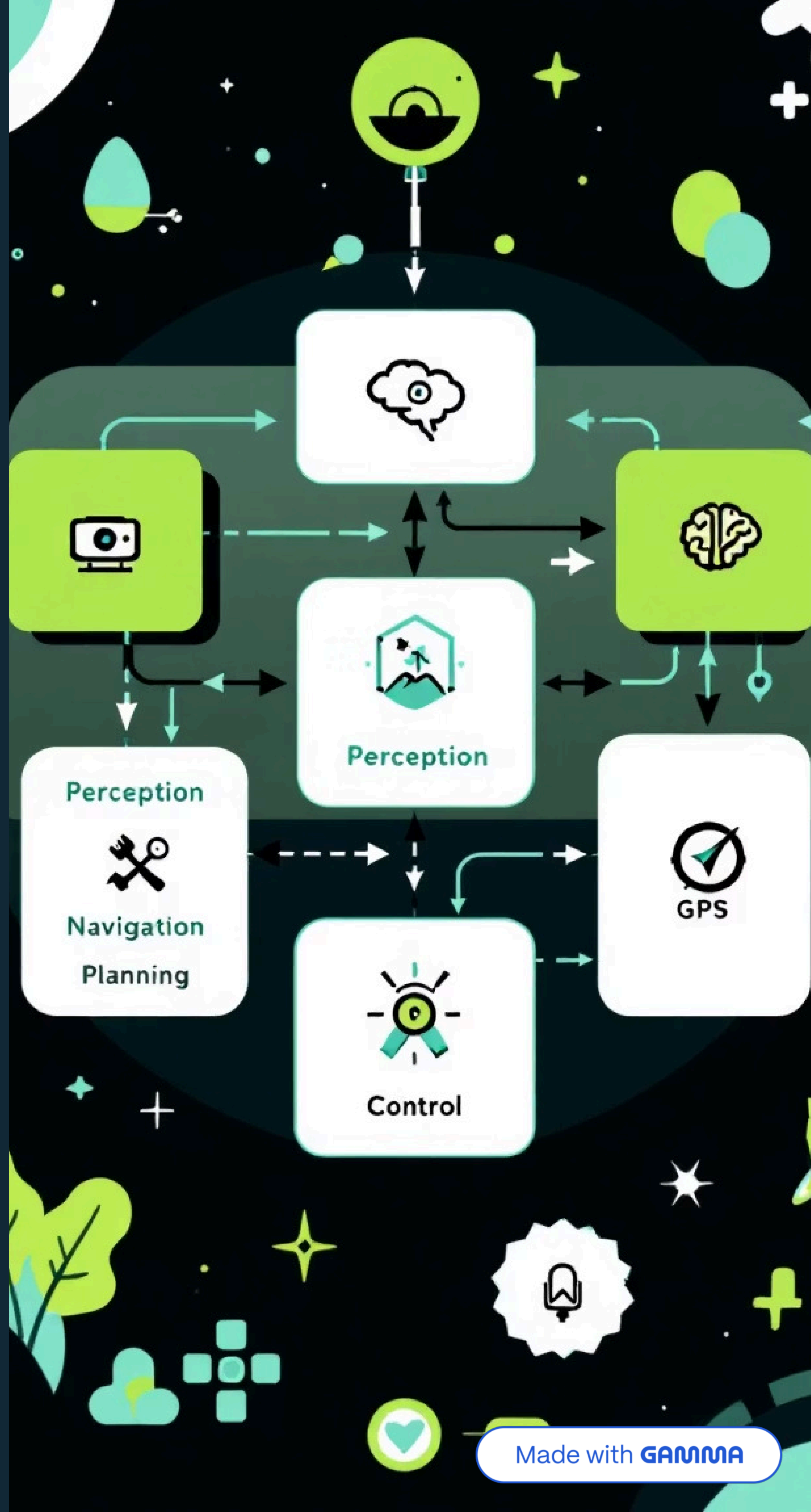
Integrates sensor data with a Kalman filter to create a unified 3D map.

Obstacle Detection

CNN model identifies and classifies obstacles using simulated Mars terrain data.

Path Planning

Reinforcement learning algorithm generates optimal, energy-efficient routes.



Workflow: From Data to Action

1. **Sensor Data Collection**

   LiDAR, cameras, and IMU gather raw environmental data.

2. **Data Fusion & Mapping**

   Sensor fusion creates a 3D map of the rover's surroundings.

3. **Obstacle Classification**

   CNN identifies and labels obstacles by risk level.

4. **Path Planning**

   RL-based planner generates an optimal route to the target.

5. **Motor Commands**

   Control interface translates plans into rover movement.

Assumptions & Risk Mitigation

1

Hardware Support

Assumes onboard GPU for real-time processing and access to simulated Mars environments.

2

Sensor Redundancy

Mitigates sensor noise/failure with fallback strategies and robust fusion.

3

Model Generalization

Prevents overfitting with data augmentation and domain randomization.

4

Efficiency

Optimizes algorithms to address computational limits and prioritize critical tasks.

Testing Plan & Scenarios

1

Static Obstacle Avoidance

Test rover navigation on simulated terrain with rocks and craters; measure collision-free success rate.

2

Dynamic Environments

Assess rover's ability to adapt to changing conditions like dust storms; track adaptation time and avoidance.

3

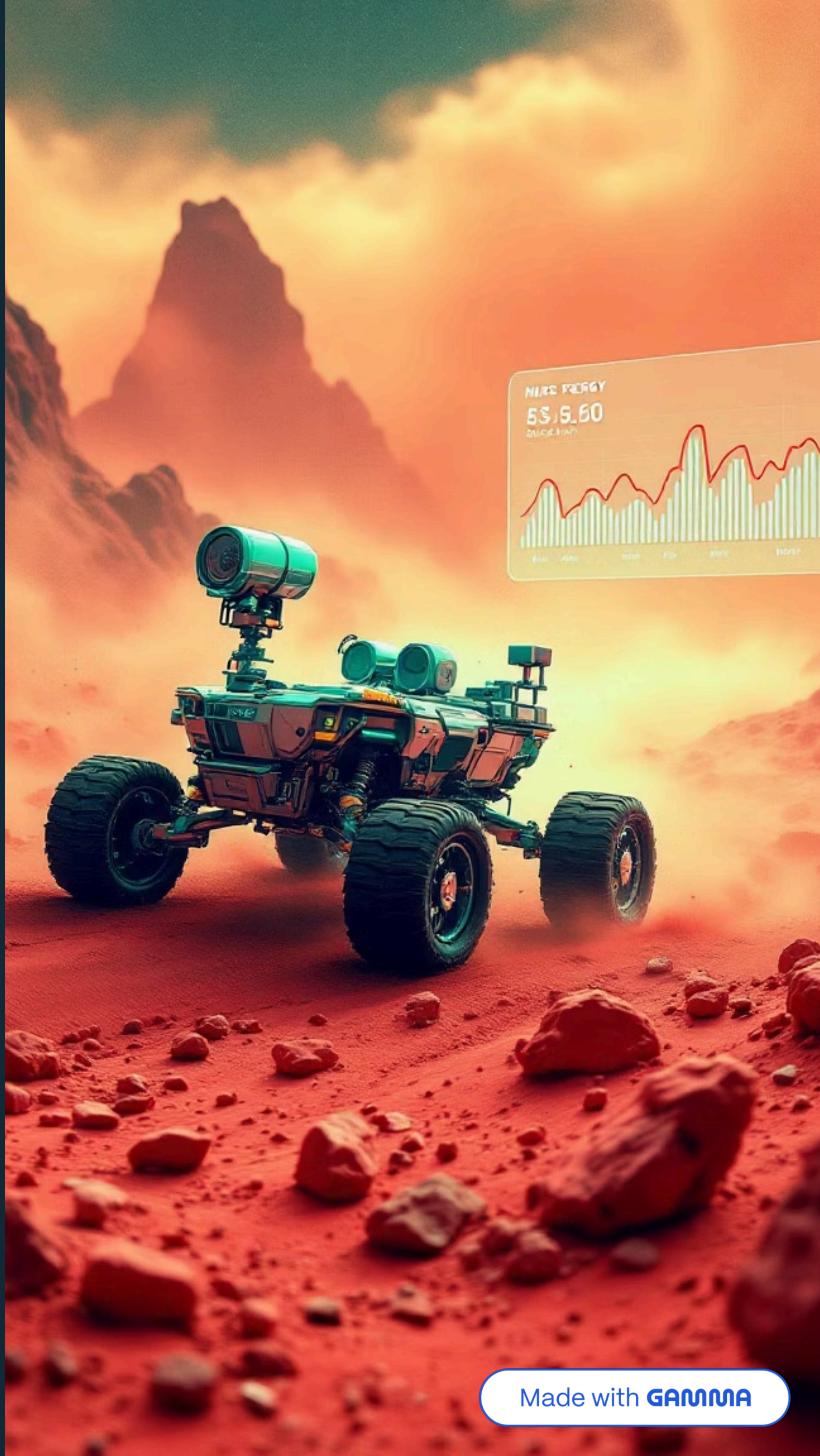
Edge Cases

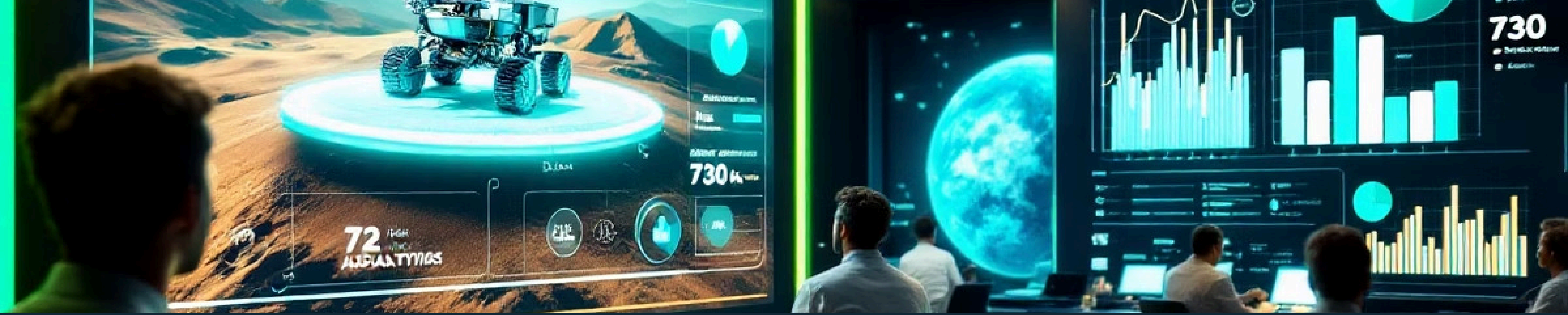
Evaluate performance on extreme slopes, narrow passages, and with sensor noise; monitor failure and recovery rates.

4

Energy Efficiency

Measure energy use during long-distance navigation with obstacles, comparing to baseline paths.





Validation & Success Criteria

90%

Obstacle Detection Accuracy

Precision and recall for identifying obstacles must exceed 90%.

95%

Path Planning Success

Static environment navigation must succeed in over 95% of tests.

20%

Energy Efficiency Gain

System should use at least 20% less energy than shortest-path baseline.

90%

Robustness

No mission failures in more than 90% of edge case tests.