INNOVATORS DAY 2025

24 HOURS HACKATHON

PS ID:1736

AI based frame interpolation, video generation and display system for WMS services

---

TEAM NAME : VISION TECHIES

TEAM MEMBERS:-

1. DHARANI K V
2. JANARTHANAN S
3. PAVITHRA V
4. RAJARAM P

Table of Contents

I Problem Statement

- Title: AI-Powered Frame Interpolation for Satellite Video Visualization on WebGIS
- Overview: A novel solution for generating smooth, high-resolution satellite videos from sparse WMS imagery using frame interpolation techniques.
- Objective: To enable seamless visualization of satellite data on open-source WebGIS libraries like OpenLayers/Leaflet, ensuring compatibility with OGC standards.

Key Challenges:

- Deformable Objects: Interpolating complex, dynamic objects like clouds that deform, appear, and disappear.
- Visualization Integration: Overlaying generated videos on open-source WebGIS platforms like OpenLayers.
- Performance: Ensuring smooth visualization with on-device GPU/NPUs for real-time processing.

II Solution

Key Features:

- Frame interpolation using AI to generate smooth videos from WMS imagery.
- Integration with OpenLayers for interactive WebGIS visualization.

Two System Approaches:

- System 1: Server-based processing with Flask backend.
- System 2: Client-side processing using TensorFlow.js.

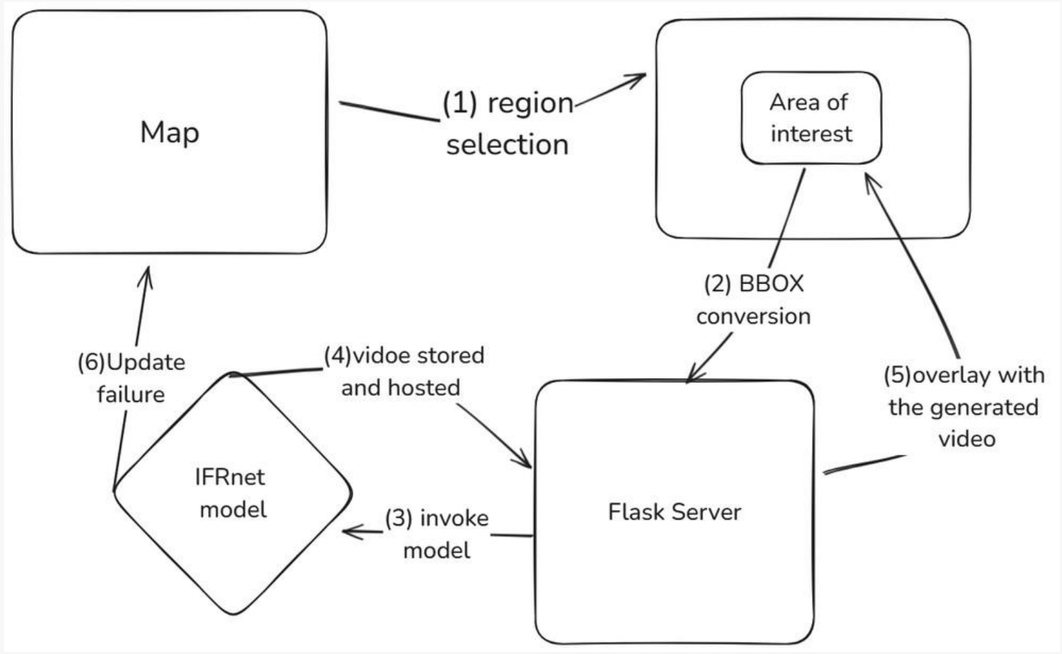
Usage Scenarios: Applications in platforms like

- VEDAS
- MOSDAC
- BHUVAN.

III System - I

- This System uses the IFRnet model for image processing and video generation.
- The system handles bounding box conversion and error updates, ensuring seamless processing.

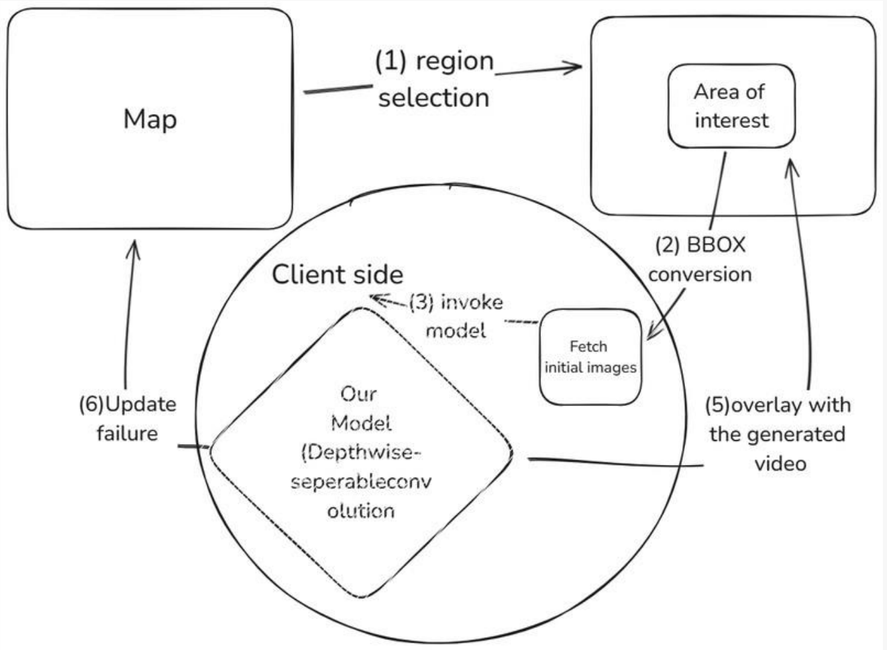
- The core components include the Map, IFRnet model, and Flask Server for video storage and hosting.



IV System - II

- This system takes a user-selected region of interest from a map, processes it using a deep learning model, and overlays the generated video output back onto the map.
- It handles tasks like bounding box conversion and error/failure updates throughout the pipeline.

- The core system is a deep learning model, which uses efficient convolution Model to interpolate the image data.



V Implementation and Feasibility

Implementation Details

- Optimized Depthwise CNN model for mobile and web platforms
- Used TensorFlow JS for client-side inference

Feasibility Considerations

- Computational resources: Ensured sufficient CPU, GPU, and RAM for efficient inference
- Memory constraints: Optimized model to minimize memory usage
- Power consumption: Optimized implementation to reduce power consumption
- Latency and throughput: Met required latency and throughput requirements for real-time applications
- Security and privacy: Implemented measures to protect sensitive user data and prevent model theft or tampering

VI Conslusion

Implementation Details

- Automatic video generation using frame interpolation techniques.
- Compatibility with OGC-compatible WMS services.
- Overlaying videos on an interactive browser-based map using Leaflet.
- Handling deformable objects like clouds

We plan to explore the following improvements when time permits:

- Optimizing our model for low-end devices to improve performance
- Enhancing accuracy by reducing noisy outputs and improving overall model reliability
- Implementing techniques to improve model efficiency and reduce computational requirements