WORK DIVISION FOR PPT

1-PROBLEM STATEMENT+ TECH STACK
2- YOUR SOLUTION
3- EXPLAIN THE WORKING
4- OUTCOME + BUSINESS VALUE

Problem Statement

Marine Protected Areas (MPAs) are ocean regions where fishing and commercial activity are strictly prohibited to preserve marine ecosystems.
However, illegal fishing and maritime trafficking continue to occur inside these zones.

A common tactic used by such vessels is to **disable their AIS (Automatic Identification System) transponders**, making them invisible to conventional tracking systems. These vessels are known as **“dark vessels.”**

Why current systems fail:

- AIS-based monitoring only sees **cooperative ships**
- Manual satellite inspection is **slow and expensive**
- The ocean is vast, making continuous monitoring impractical
- Suspicious behavior often emerges **over time**, not in a single image

As a result, enforcement agencies receive information **too late** or **with insufficient context** to act effectively.

Goal of our System

a **serverless, satellite-driven intelligence platform** that:

- Automatically monitors **only high-value protected zones**
- Detects vessels that are **physically present but digitally silent**
- Analyzes **behavior over time**, not just single detections
- Assigns a **suspicion score** rather than making accusations
- Produces **clear, explainable reports** for human decision-makers

PIPELINE

1. **Geofence Marine Protected Areas**
   Restrict monitoring to legally protected ocean zones using predefined polygon boundaries.

2. **Detect vessels via Sentinel-1 SAR**
   Use radar imagery to identify bright, ship-like objects on the ocean surface, day or night.
3. **Cluster radar signatures (K-means)**
   Group detected radar objects based on size, intensity, and texture to separate vessels from noise.
4. **Cross-check AIS**
   Compare detected vessel locations with AIS broadcasts to identify ships that are physically present but digitally silent.
5. **Validate via CNN**
   Apply a deep-learning classifier on SAR image patches to confirm whether a detected object is truly a vessel.
6. **Build vessel time-series**
   Track each vessel across satellite passes to construct movement and behavior sequences over time.
7. **LSTM anomaly detection**
   Use LSTMs to identify anomalous movement patterns such as loitering or irregular motion inside protected zones.
8. **Assign suspicion score**
   Combine radar, AIS, temporal, and behavioral signals into a continuous risk score for prioritization.
9. **GenAI generates explanation**
   Convert model outputs into clear, human-readable intelligence summaries for analysts and authorities.

STEP 1: Geofence Marine Protected Areas (MPAs)

We restrict surveillance to **Marine Protected Areas**, where any fishing activity is illegal by definition.

How we do it (technically)

1. We ingest **MPA boundary data** as polygons
   - From sources like WDPA (World Database on Protected Areas) (https://data.marine.copernicus.eu/product/EXT_MARINE-PROTECTED-AREAS/description?utm_source=chatgpt.com)

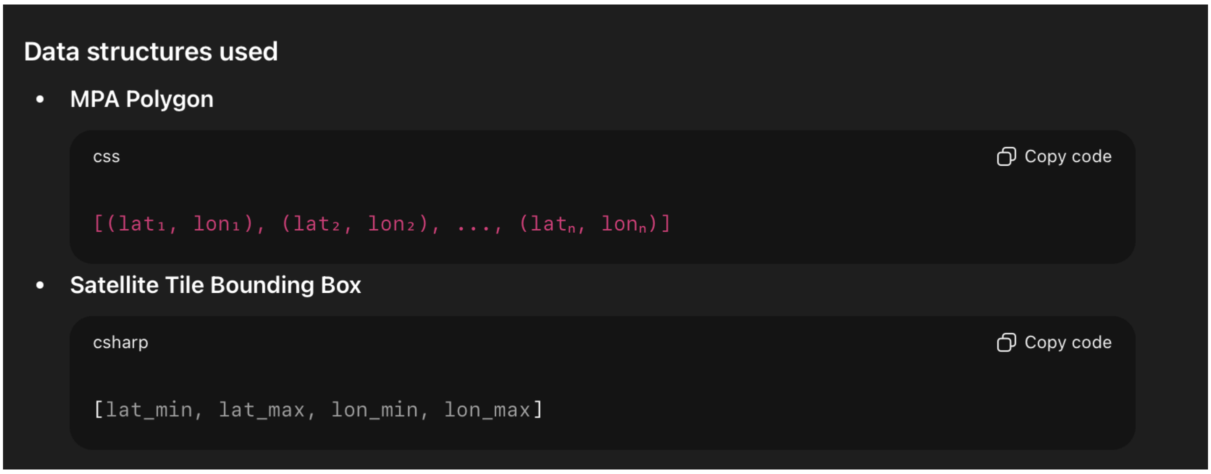
(https://developers.google.com/earth-engine/datasets/catalog/WCMC_WDPA_current_polygons?utm_source=chatgpt.com)

    - Each MPA is represented as a **closed latitude–longitude polygon**

2. For every incoming Sentinel-1 satellite tile:
    - We compute **polygon–tile intersection**
    - Only tiles that overlap an MPA are processed further

Mathematically:

Process tile $\iff \text{Area}(\text{Tile} \cap \text{MPA}) > 0$



Data structures used

- MPA Polygon

```css
[(lat₁, lon₁), (lat₂, lon₂), ..., (latₙ, lonₙ)]
```

- Satellite Tile Bounding Box

```csharp
[lat_min, lat_max, lon_min, lon_max]
```

Why this step matters

- Reduces compute by **~95%**
- Makes the system **economically viable**
- Adds **clear business and legal logic**
- Allows strong framing: *“We guard sanctuaries, not the entire ocean”*

STEP 2: Detect Vessels via Sentinel-1 SAR

What we are doing

*We identify **physical ships on the ocean surface**, independent of any tracking system.*

How Sentinel-1 SAR helps

*Sentinel-1 uses **C-band Synthetic Aperture Radar**, which measures **radar backscatter (σ^0)**.*

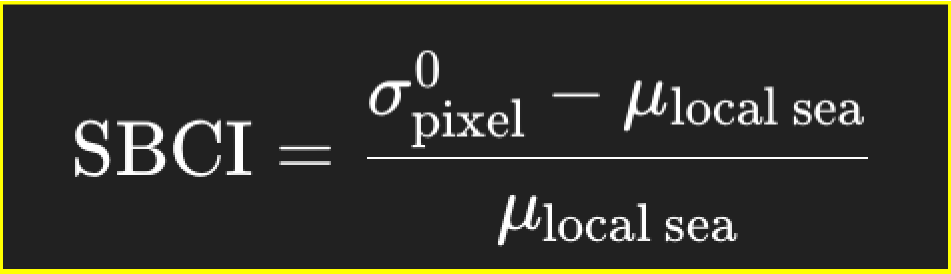
Key physical property:

- *Calm ocean → low backscatter (dark)*
- *Ships (metal, edges) → high backscatter (bright)*

Processing steps

1. ***Radiometric calibration***
   Converts raw SAR signal → σ^0 (physical reflectivity)
2. ***Speckle filtering***
   Reduces radar noise using filters (e.g., Lee filter)
3. ***Local sea normalization***
   Computes local mean backscatter to account for wind/waves

Ship Backscatter Contrast Index (core metric)



$$\mathrm{SBCI} = \frac{\sigma^0_{\text{pixel}} - \mu_{\text{local sea}}}{\mu_{\text{local sea}}}$$

- *Ships → high SBCI*
- *Sea clutter → low SBCI*

Output of this step

*A **binary detection map**:*

- *Bright pixels = possible ship pixels*
- *Dark pixels = sea*

Why this step matters

- *Detects ships **even if AIS is off***
- *Works **day/night and in clouds***
- *Is physics-based → very defensible*

STEP 3: Cluster Radar Signatures (K-means)

What we are doing

*We group detected radar objects to separate **real vessels** from **noise and artifacts**.*

Why clustering is needed

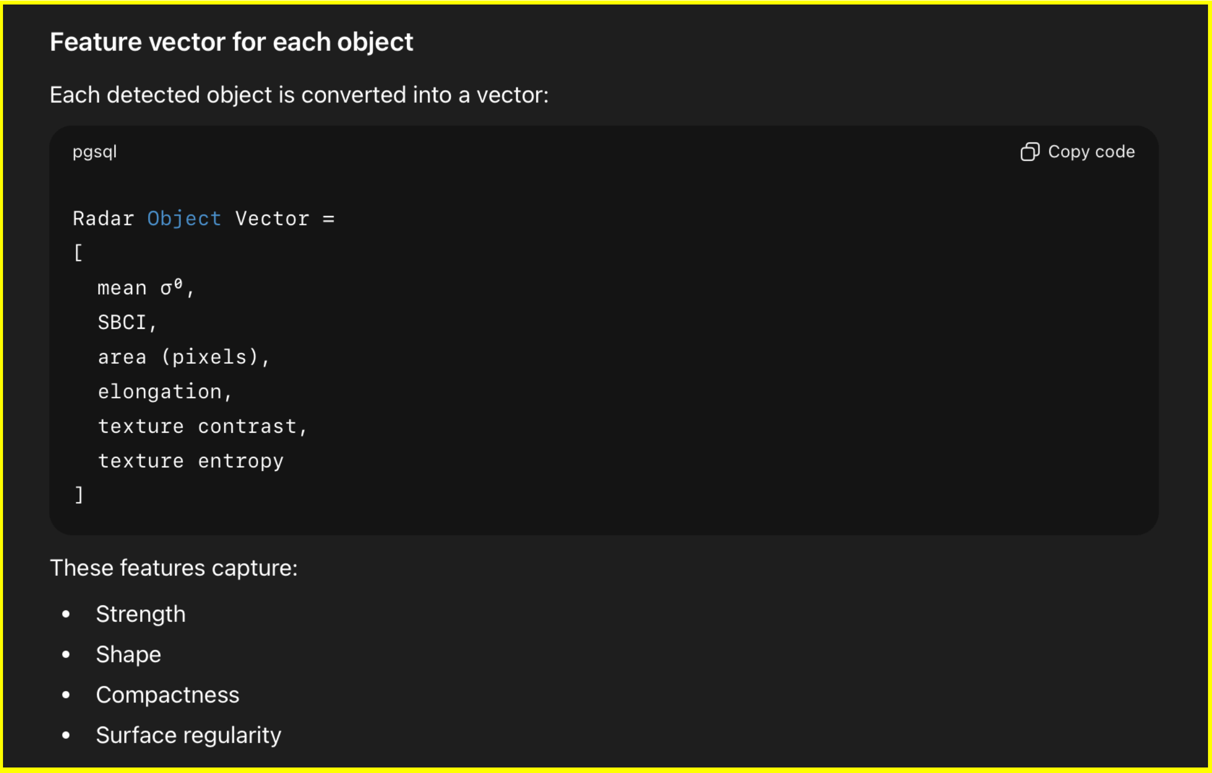
Not every bright spot is a ship:

- *Wave crests*
- *Wind streaks*
- *Radar noise*

*So we **group objects by similarity**, not by labels.*

Object extraction

1. *Threshold SBCI map*
2. *Apply **connected component analysis***
3. *Each connected region = **candidate object***



Feature vector for each object

Each detected object is converted into a vector:

```
Radar Object Vector =
[
  mean σ⁰,
  SBCI,
  area (pixels),
  elongation,
  texture contrast,
  texture entropy
]
```

These features capture:

- Strength
- Shape
- Compactness
- Surface regularity

Clustering with K-means

- Input: object feature vectors
- Distance: Euclidean
- K chosen via **Elbow Method**

Clusters typically correspond to:

- Large vessels
- Small vessels
- Static structures
- Noise

Output of this step

- Each object assigned a **cluster label**
- Only vessel-like clusters proceed

Why this step matters

- Removes false positives early
- Makes CNN validation cheaper
- Shows **classical ML used correctly**, not blindly

STEP 4: Cross-check AIS (The “Dark Vessel” Logic)

What we are doing

We compare **physical presence** (radar) with **digital presence** (AIS).

AIS basics (very important to explain)

AIS broadcasts: [vessel_id, latitude, longitude, timestamp, speed, heading]

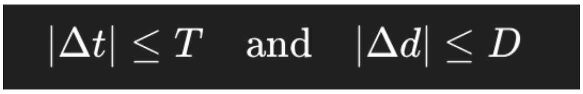
AIS is:

- Voluntary/compliance-based
- Easy to turn off
- Widely used by legal vessels

Matching logic

For each radar-detected vessel object:

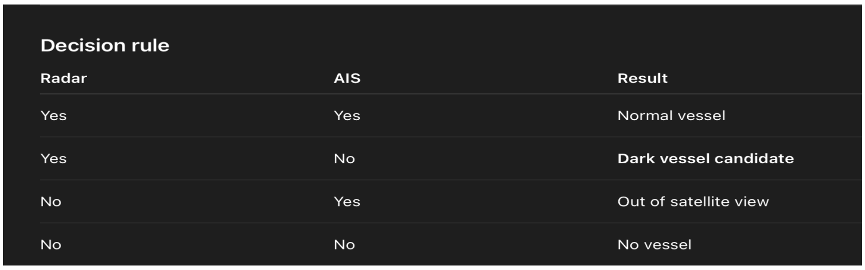
We search AIS records such that:



$$|\Delta t| \leq T \quad \text{and} \quad |\Delta d| \leq D$$

Typical values:

- T ≈ 10–15 minutes
- D ≈ 2–5 km



Decision rule

Radar	AIS	Result
Yes	Yes	Normal vessel
Yes	No	**Dark vessel candidate**
No	Yes	Out of satellite view
No	No	No vessel

Output of this step

*A **Dark Vessel Candidate List***

STEP 5: Validate Vessel Detections using CNN

Why this step is needed

Radar images contain **false positives** such as:

- Small islands
- Wave clutter
- Oil rigs
- Sea state artifacts

Before declaring a “dark vessel”, we must **confirm that the detected object is truly a ship**.

Process

1. For each radar detection from Step 3:
   - Extract a **small SAR image patch** (e.g., 64×64 or 128×128 pixels) centered on the object
2. Pass this patch through a **Convolutional Neural Network (CNN)** trained on:
   - Vessel vs non-vessel SAR imagery
3. The CNN outputs:
   - `P(vessel)` — probability that the object is a real ship

Output

- Detections below a confidence threshold are discarded
- Only **high-confidence vessels** move forward

📌 **Result:**
We now have a **clean, verified list of real vessels physically present inside protected zones**.

Training Data Sources:

- **xView3 Dataset** - Labeled SAR images with vessel/non-vessel annotations
- **Airbus Ship Detection Dataset** - Optical + SAR imagery
- **MaritimeNet** - Marine vessel detection dataset

Why This Matters:

Without CNN validation, you'd send coast guards on wild goose chases investigating waves and debris. This step ensures **95%+ accuracy** before raising alerts.

STEP 6: Build Vessel Time-Series Tracks

Why this step is needed

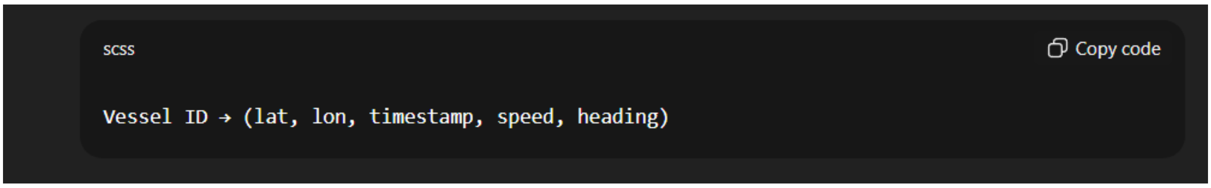
A single satellite image gives a **snapshot**, but illegal activity is revealed by **movement patterns over time**.

We must answer:

- Is the vessel lingering?
- Is it repeatedly entering the protected zone?
- Is it behaving abnormally?

Process

1. For each validated vessel:
   - Assign a **temporary vessel ID**
2. Across multiple Sentinel-1 passes:
   - Match detections using spatial proximity, size, and heading
3. Construct a **time-ordered trajectory**:



```
Vessel ID → (lat, lon, timestamp, speed, heading)
```

Derived Signals

From the trajectory we compute:

- Speed changes
- Directional variance
- Dwell time inside the MPA
- Entry/exit frequency

📌 **Result:**
Each vessel is now represented as a **behavioral time-series**, not just a dot on a map.

STEP 7: LSTM Anomaly Detection (Behavioral Analysis)

Purpose

Not all MPA intrusions are equal. We need to distinguish:

- ✅ **Transit vessel** (passing straight through at 15 knots) - Lower priority
- ⚠️ **Lingering vessel** (slow, irregular movement) - Medium priority
- 🚨 **Fishing vessel** (circling, loitering for hours) - **CRITICAL PRIORITY**

The Problem

How do you automatically detect "suspicious behavior" from a sequence of position points?

The Solution: LSTM Neural Network

LSTM (Long Short-Term Memory) is a type of RNN (Recurrent Neural Network) that's perfect for analyzing sequences over time. It can learn patterns like:

- "Normal vessels move in straight lines at consistent speeds"
- "Fishing vessels circle back on themselves repeatedly"
- "Vessels loitering in one area for >3 hours is abnormal"

Training Data:

You need labeled examples of:

- **Normal behavior** (cargo ships transiting, legal fishing outside MPAs)

- **Anomalous behavior** (documented illegal fishing cases, loitering patterns)

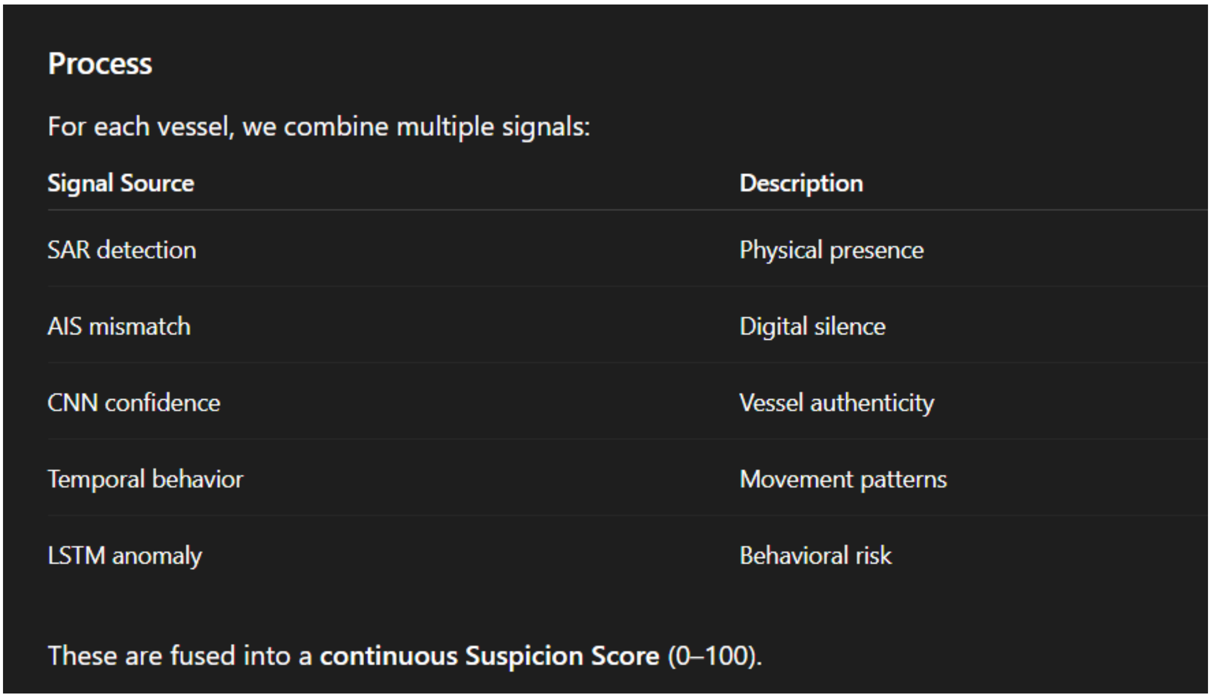
Sources:

- Global Fishing Watch datasets (labeled fishing vs non-fishing)
- Documented illegal fishing incidents from coast guard reports
- Synthesized anomalous patterns (circling, sudden stops, night operations)

STEP 8: Assign Composite Suspicion Score

Why this step is needed

Authorities cannot investigate everything.
They need **prioritized, explainable alerts**.



Process

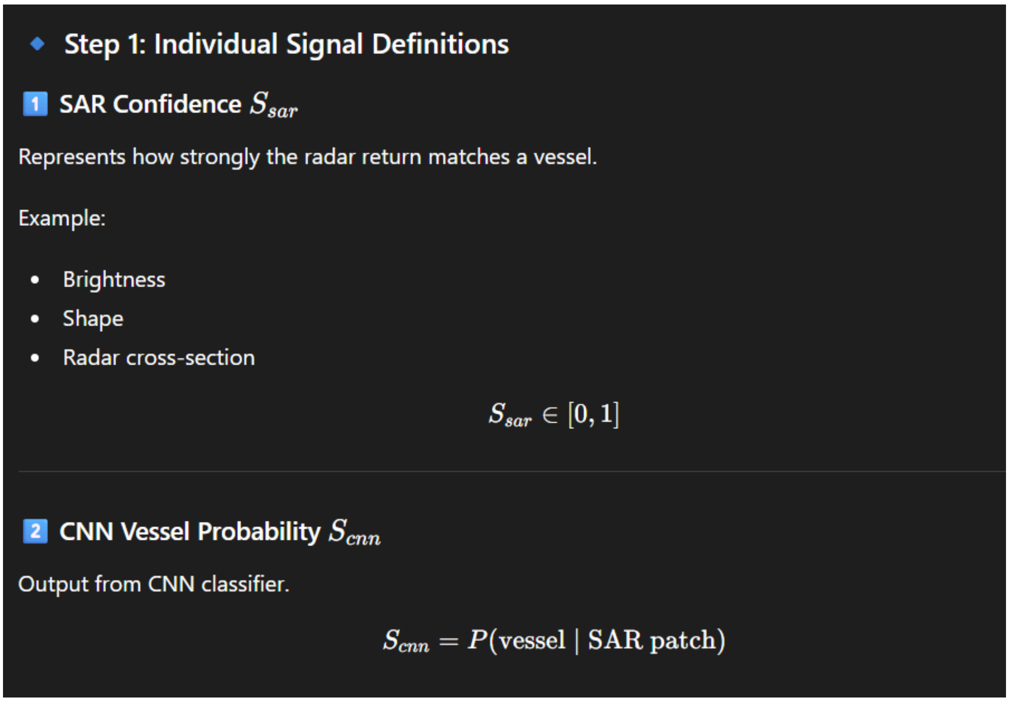
For each vessel, we combine multiple signals:

Signal Source	Description
SAR detection	Physical presence
AIS mismatch	Digital silence
CNN confidence	Vessel authenticity
Temporal behavior	Movement patterns
LSTM anomaly	Behavioral risk

These are fused into a **continuous Suspicion Score** (0–100).

Example Logic

- AIS off + High CNN confidence + High anomaly → 🔴 **Critical**
- AIS off + Normal movement → 🟡 **Medium**
- AIS on + Normal movement → 🟢 **Low**



🔹 Step 1: Individual Signal Definitions

1️⃣ SAR Confidence S_{sar}

Represents how strongly the radar return matches a vessel.

Example:

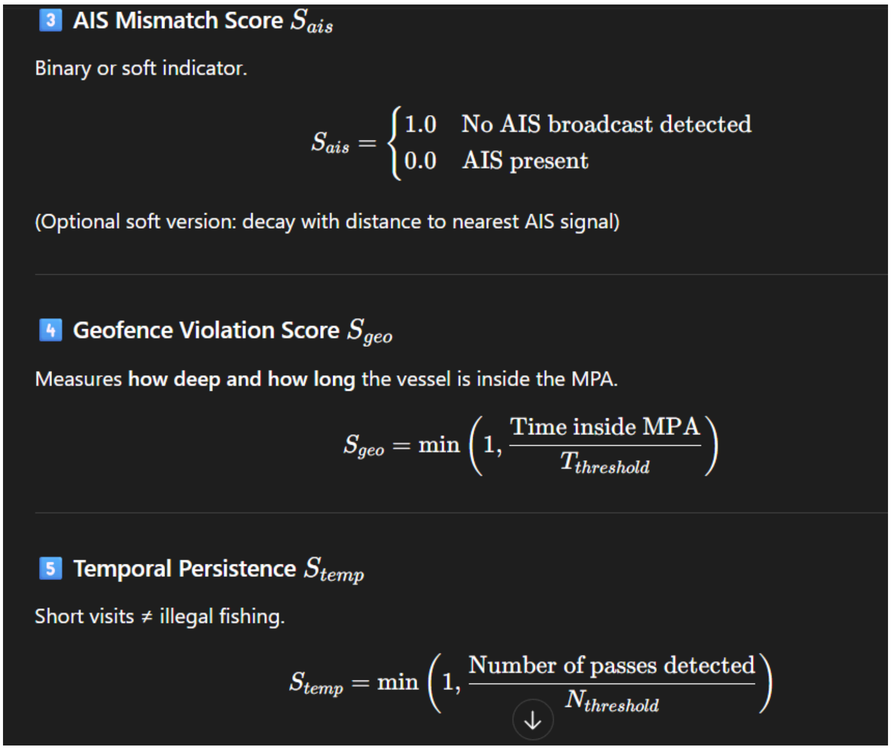
- Brightness
- Shape
- Radar cross-section

$$S_{sar} \in [0, 1]$$

2️⃣ CNN Vessel Probability S_{cnn}

Output from CNN classifier.

$$S_{cnn} = P(\text{vessel} \mid \text{SAR patch})$$



3️⃣ AIS Mismatch Score S_{ais}

Binary or soft indicator.

$$S_{ais} = \begin{cases} 1.0 & \text{No AIS broadcast detected} \\ 0.0 & \text{AIS present} \end{cases}$$

(Optional soft version: decay with distance to nearest AIS signal)

4️⃣ Geofence Violation Score S_{geo}

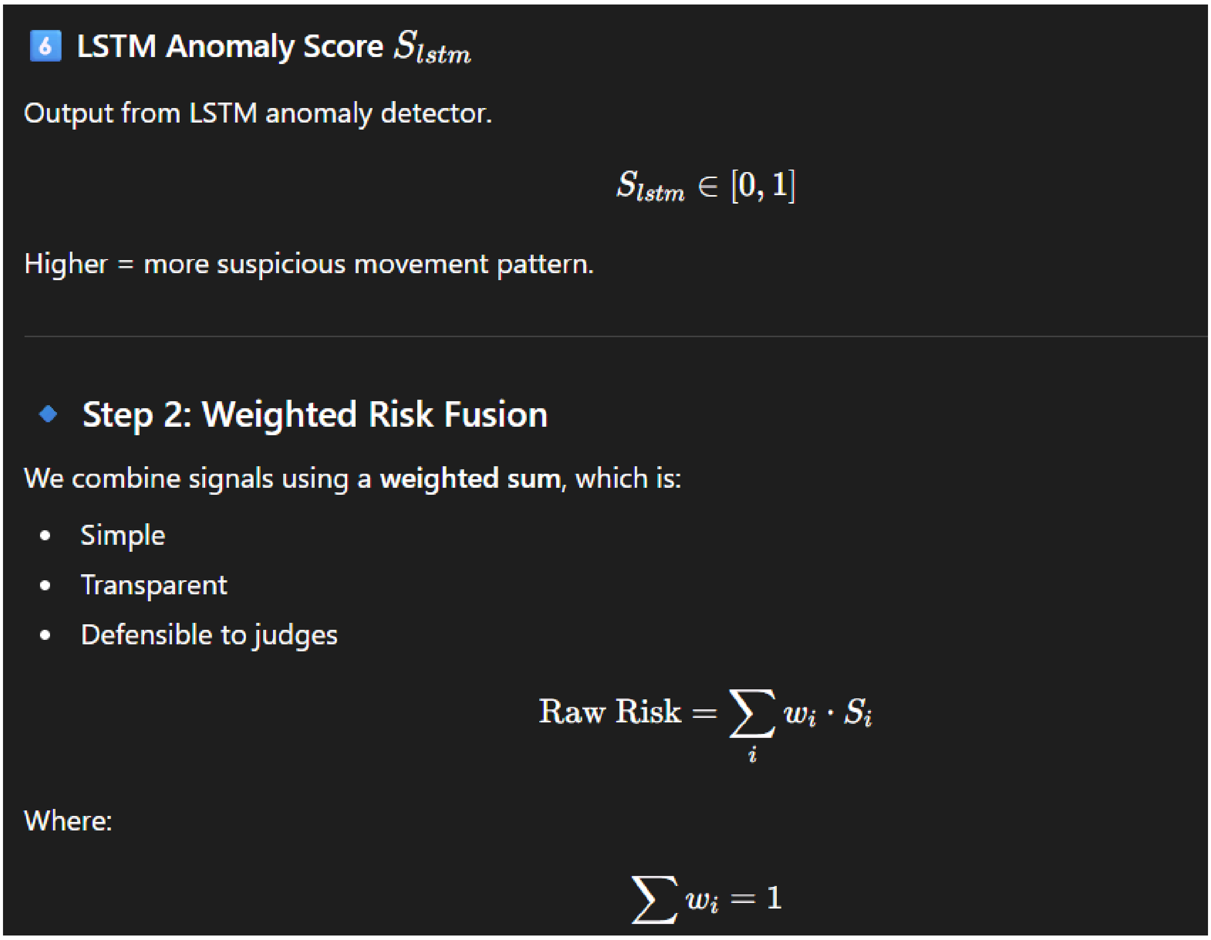
Measures **how deep and how long** the vessel is inside the MPA.

$$S_{geo} = \min\left(1, \frac{\text{Time inside MPA}}{T_{threshold}}\right)$$

5️⃣ Temporal Persistence S_{temp}

Short visits ≠ illegal fishing.

$$S_{temp} = \min\left(1, \frac{\text{Number of passes detected}}{N_{threshold}}\right)$$



6️⃣ LSTM Anomaly Score S_{lstm}

Output from LSTM anomaly detector.

$$S_{lstm} \in [0, 1]$$

Higher = more suspicious movement pattern.

🔹 Step 2: Weighted Risk Fusion

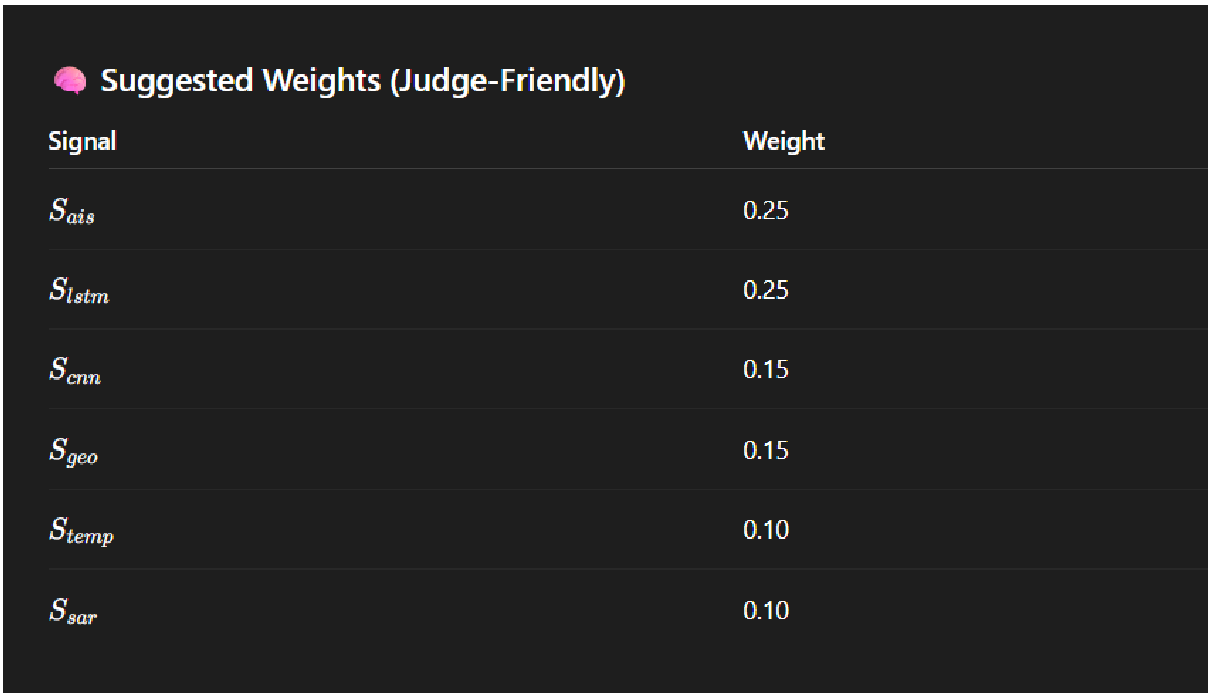
We combine signals using a **weighted sum**, which is:

- Simple
- Transparent
- Defensible to judges

$$\text{Raw Risk} = \sum_i w_i \cdot S_i$$

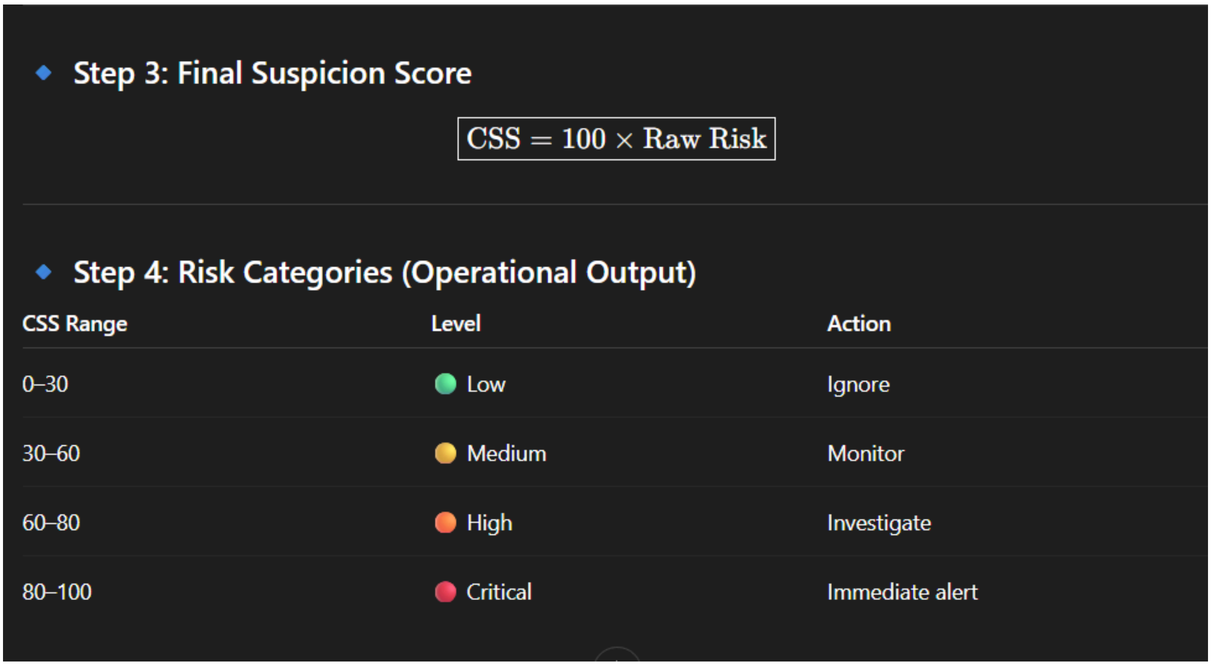
Where:

$$\sum w_i = 1$$



🧠 Suggested Weights (Judge-Friendly)

Signal	Weight
S_{ais}	0.25
S_{lstm}	0.25
S_{cnn}	0.15
S_{geo}	0.15
S_{temp}	0.10
S_{sar}	0.10



🔹 Step 3: Final Suspicion Score

$$\boxed{\text{CSS} = 100 \times \text{Raw Risk}}$$

🔹 Step 4: Risk Categories (Operational Output)

CSS Range	Level	Action
0–30	🟢 Low	Ignore
30–60	🟡 Medium	Monitor
60–80	🟠 High	Investigate
80–100	🔴 Critical	Immediate alert

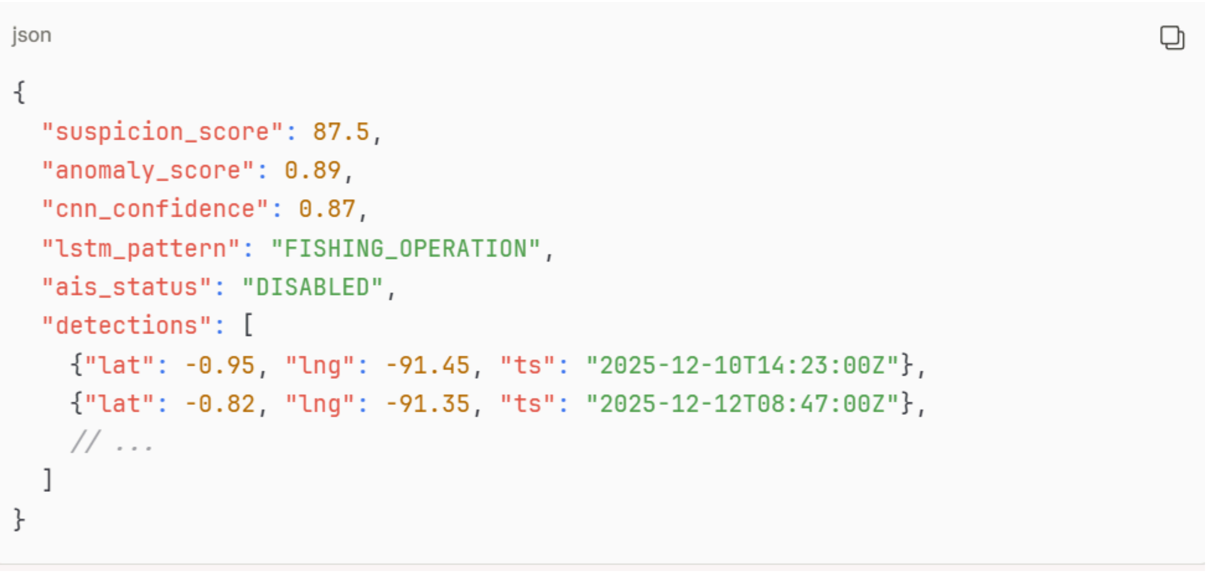
STEP 9: GenAI Generates Explanation (Intelligence Summarization)

Purpose

Transform raw technical outputs into **clear, actionable intelligence reports** that coast guard officers, policy makers, and analysts can understand instantly—without needing to be data scientists.

The Problem:

Your pipeline produces this:



```json
{
  "suspicion_score": 87.5,
  "anomaly_score": 0.89,
  "cnn_confidence": 0.87,
  "lstm_pattern": "FISHING_OPERATION",
  "ais_status": "DISABLED",
  "detections": [
    {"lat": -0.95, "lng": -91.45, "ts": "2025-12-10T14:23:00Z"},
    {"lat": -0.82, "lng": -91.35, "ts": "2025-12-12T08:47:00Z"},
    // ...
  ]
}
```

Use a **Generative AI model** (like GPT-4, Claude, or Llama) to:

1. **Translate** technical data into plain English
2. **Contextualize** the threat (why it matters)
3. **Recommend** specific actions
4. **Generate** court-ready evidence summaries