

Guía de Operación y Uso

Sistema de Posicionamiento Visual para ROVs (PDI-NET)

1 Inicio del Sistema

Una vez instalado el entorno, el operador debe iniciar el software desde la terminal.

```
python script/main.py (Windows) o python3 script/main.py (Linux)
```

2 Flujo de Trabajo Paso a Paso

Paso 1: Selección de Fuente de Video

Al iniciar, se abrirá una ventana de exploración de archivos.

- **Archivos .MP4:** Videos estándar. El sistema procesará la imagen asumiendo una vista monocular o estéreo según la configuración.
- **Archivos .SVO:** Archivos nativos de cámaras ZED. Seleccione esto si necesita datos de profundidad de alta precisión (requiere ZED SDK).

Paso 2: Definición del Punto de Partida

Aparecerá un cuadro de diálogo solicitando el **"Frame Inicial"**.

- Ingrese 0 para analizar el video desde el principio.
- Ingrese un número mayor (ej. 500) si desea saltar la introducción o partes irrelevantes del video para ahorrar tiempo.

3 Interfaz gráfica

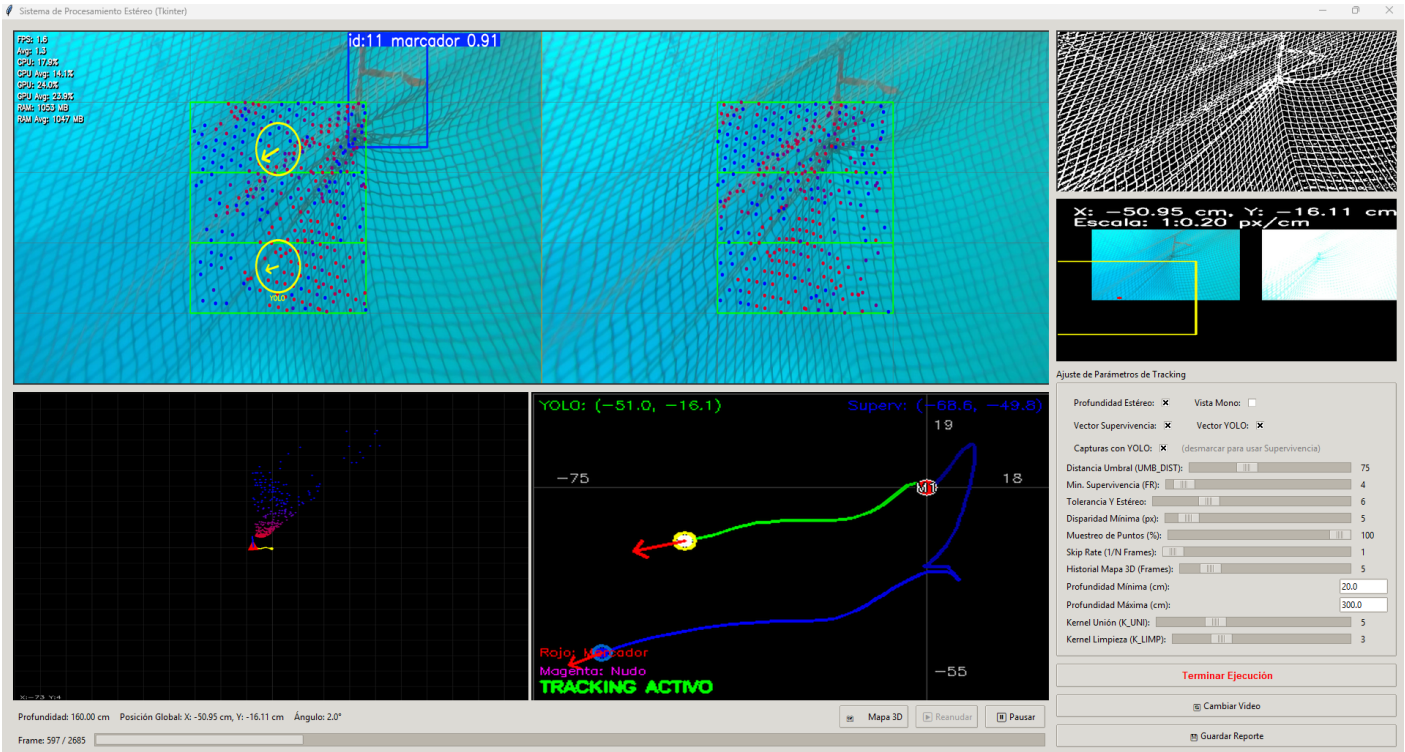


Figura 1: Interfaz de usuario

3.1. Secciones de la interfaz

La interfaz gráfica se puede dividir en 6 zonas principales.

1. **Video stereo:** Permite ver en tiempo real el analisis y procesamiento de los datos. Es estaa zona se muestran los cuadrantes activos, puntos detectados en la malla y flechas de dirección según navegación por puntos y navegación por seguimiento de objetos (Yolo.)
2. **Mascara Threshol:** Muestra la binarización que se está usandoa ctualmente. Esta vista permite al usuario saber si es necesario modificar los valores de tamaño de los kernel de apertura y cierre.
3. **Seguimiento panorámico:** Debbaajo de la imagen binaria, se encuentra un mapa que muestra capturas en puntos específicos según el avance del rovs.
4. **Radar:** En la esquina inferior izquierda tenemos una vista que permite ver en tiempo real los puntos frente a la cámara. El codigo de colores utilizados se muuestra en la sección de barra de profundidad.
5. **Registro historico de posición:** A la derecha del radar se encuentra un mapa que muestra el avance del rovs según cada método.
6. **Ajustes:** Acá se puerder modificar valores necesarios para mejorar el funcionamiento del modelo. Esto se detalla en la sección de ajuste de parámetros.

3.2. A. Ventana Principal (Video Overlay)

Aquí se muestra la realidad aumentada sobre la imagen original.

Simbología de Detección

- **Recuadro VERDE: Objeto Seguro / Estructura Normal.** Indica elementos reconocidos (bordes de malla, nudos) que no presentan anomalías.
- **Recuadro ROJO: ANOMALÍA / DAÑO.** El sistema ha detectado una irregularidad (ruptura, deformación o grieta). El operador debe marcar el minuto del video o revisar el reporte final.
- **Círculos AMARILLO: Marcadores / Balizas.** Puntos de referencia visual detectados automáticamente para corregir la posición en el mapa.
- **Flecha Central: Vector de Movimiento.** Indica la dirección hacia la que se desplaza la cámara (Odometría).

3.3. B. Barra de Profundidad (Lateral)

Una escala de colores vertical indica la distancia de los objetos respecto a la cámara:

- **Rojo Intenso:** Objetos muy cercanos (< 20 cm). ¡Riesgo de colisión!
- **Gradiente:** Transición suave de colores.
- **Azul Oscuro:** Objetos lejanos (> 3 metros). Fondo.

3.4. C. Ventana de Mapa (Grid 2D)

Muestra la trayectoria reconstruida del ROV/Cámara.

- El mapa se construye progresivamente uniendo las imágenes capturadas.
- Permite verificar qué zonas de la estructura ya han sido inspeccionadas.

4 Controles Durante la Operación

El sistema está diseñado para ser automatizado, pero permite intervención básica mediante el teclado sobre la ventana de video activa:

- **Tecla 'q': Detener y Guardar.** Finaliza el análisis inmediatamente, cierra las ventanas y fuerza la generación de los reportes de salida con los datos recolectados hasta el momento.

5 Ajuste de Parámetros

Si el operador nota problemas de rendimiento o detección, puede ajustar valores en el archivo `config.py` con un editor de texto simple.

Problema: El video va muy lento o con "lag"

Solución: Modifique el parámetro `SKIP_RATE`.

- Valor actual: 1 (Procesa todos los frames).
- Cambio recomendado: 2 o 3. Esto hará que el sistema analice 1 de cada 3 imágenes, triplicando la velocidad sin perder demasiada precisión en el seguimiento.

Problema: Se pierden objetos (Tracking inestable)

Solución: Ajuste `UMB_DIST` (Umbral de Distancia).

- Si el ROV se mueve muy rápido, aumente este valor (ej. de 75 a 100) para que el sistema "busque" objetos en un área mayor entre frames.

Problema: Falsos positivos de daño

Solución: Ajuste `DMG_THRESHOLD`.

- Aumente ligeramente el valor (ej. de 2.5 a 3.0) para hacer el sistema menos sensible a variaciones pequeñas de textura.

6 6. Reportes y Salida de Datos

Al finalizar la ejecución (ya sea por fin del video o por tecla 'q'), el sistema genera archivos en la carpeta del proyecto para su análisis posterior:

- Reporte de Daños (.csv):** Lista tabular que incluye: `ID_Anomalía`, `Frame_Video`, `Posición_X`, `Posición_Y`. Ideal para importar en Excel.
- Odometría (.json):**
 - `odometria_yolo.json`: Trayectoria calculada usando inteligencia artificial.
 - `odometria_supervivencia.json`: Trayectoria calculada por método clásico (puntos de control).
- Mapa Final (.png):** Imagen compuesta de alta resolución con todo el recorrido inspeccionado visualmente cosido en un solo plano.