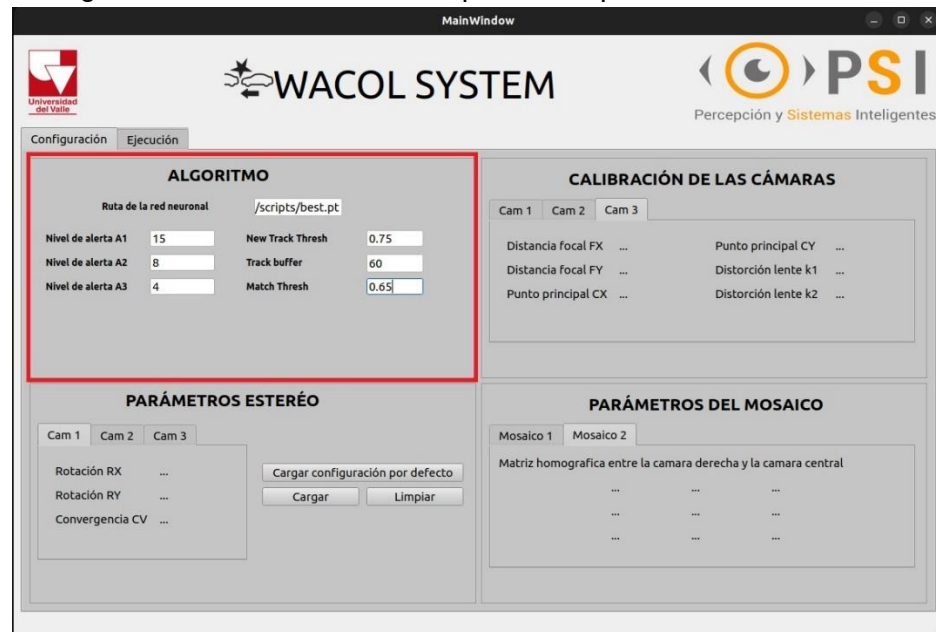


PRUEBA DE INTEGRACIÓN – MÓDULO DE CONFIGURACIÓN	
• Modificar la red neuronal, los niveles de alerta y parámetros del tracker. • Eliminar el “usr_config.yaml” cuando se quiera cargar la configuración. • Visualizar los parámetros extrínsecos e intrínsecos de las cámaras, las homografías y los parámetros estéreo.	
Tipo de prueba	Integración
Hardware Requerido	Portatil: • Procesador (CPU): Intel core i5 10300 H. • Tarjeta gráfica (GPU): Nvidia GTX 1650 Mobile. • Memoria RAM: 16 GB. • Memoria RAM de la GPU: 4 GB. 3 cámaras ZED2
Software requerido	Aplicativo de alerta de colisión con otros vehículos, (Python 3.10.12), ZED SDK, Ubuntu 22.04.5 LTS, Qt designer 5, CUDA, NVCC.
Objetivo	Validar el correcto funcionamiento del módulo de configuración.
Modificar la red neuronal a usar, los niveles de alerta y parámetros del tracker.	
Objetivo	Validar el correcto funcionamiento del aplicativo para generar el archivo “.yaml” que contenga los valores de configuración a usar.
Datos de la prueba	En esta prueba se llenan los datos requeridos en los qlabel en la pestaña configuración para ingresar el path que lleva a la red neuronal a utilizar, así como los valores para el tracker y los niveles de alerta. Yolo path: /home/franklin/ros2_ws/src/yolobot_recognition/scripts/best.pt Nivel de alerta A1: 15 segundos Nivel de alerta A2: 8 segundos Nivel de alerta A3: 4 segundos New Track Thresh: 0.75 Track Buffer: 60 Match Thresh: 0.65
Procedimiento	1. Ingresar el path de la red neuronal que se desea utilizar. 2. Ingresar los valores para los niveles de alerta. 3. Ingresar los parámetros del tracker.
Resultado esperado	1. Se genera el archivo “usr_config.yaml”. 2. Se revisa internamente que los valores se han cargado exitosamente en los nodos correspondientes.

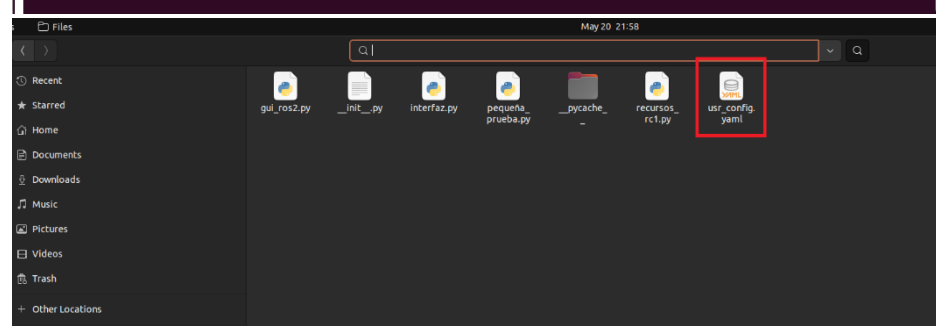
Resultados obtenidos

Se ingresan los valores deseados para cada parámetro.



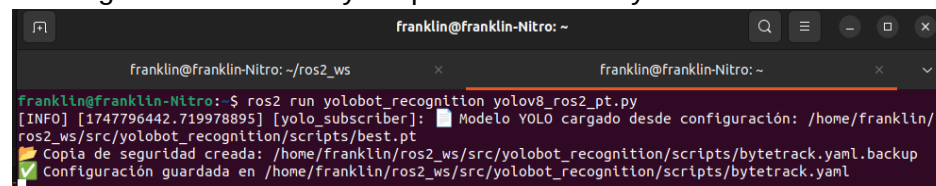
Se genera el archivo "usr_config.yaml" con los valores insertados.

```
franklin@franklin-Nitro:~/ros2_ws$ ros2 run gui_final gui
[INFO] [1747799632.176185720] [gui_node]: Nodo GUI con PyQt5 iniciado.
Warning: Ignoring XDG_SESSION_TYPE=wayland on Gnome. Use QT_QPA_PLATFORM=wayland to run on Wayland anyway.
[INFO] [1747799657.050433561] [gui_node]: Configuración guardada en: /home/franklin/ros2_ws/src/gui_final/gu
i_final/usr_config.yaml
```

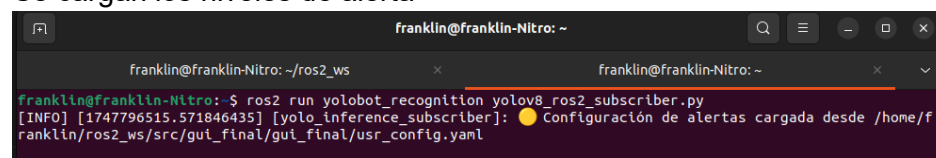


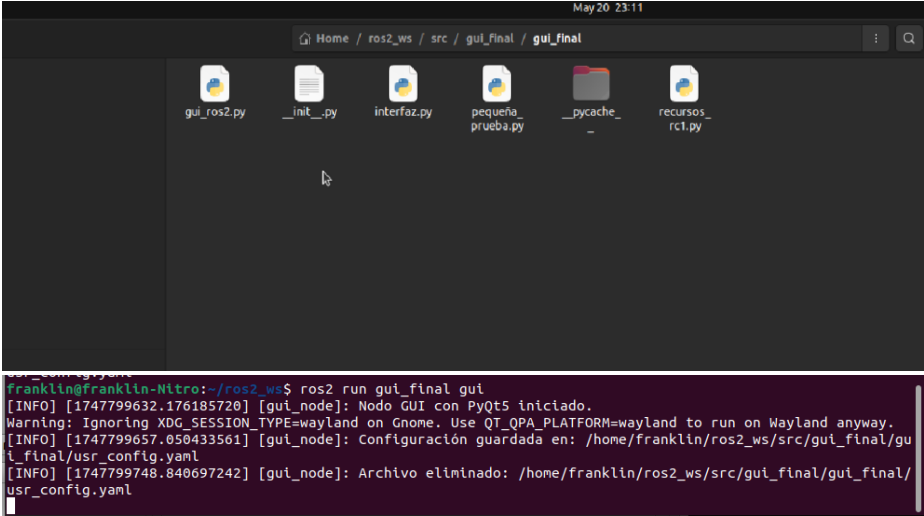
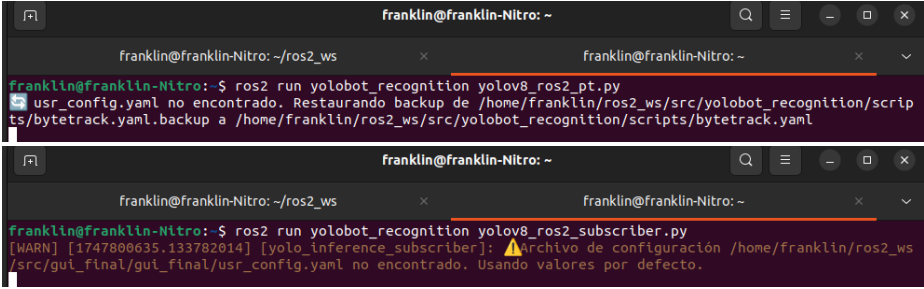
Revisión de uso exitoso del archivo "usr_config.yaml"

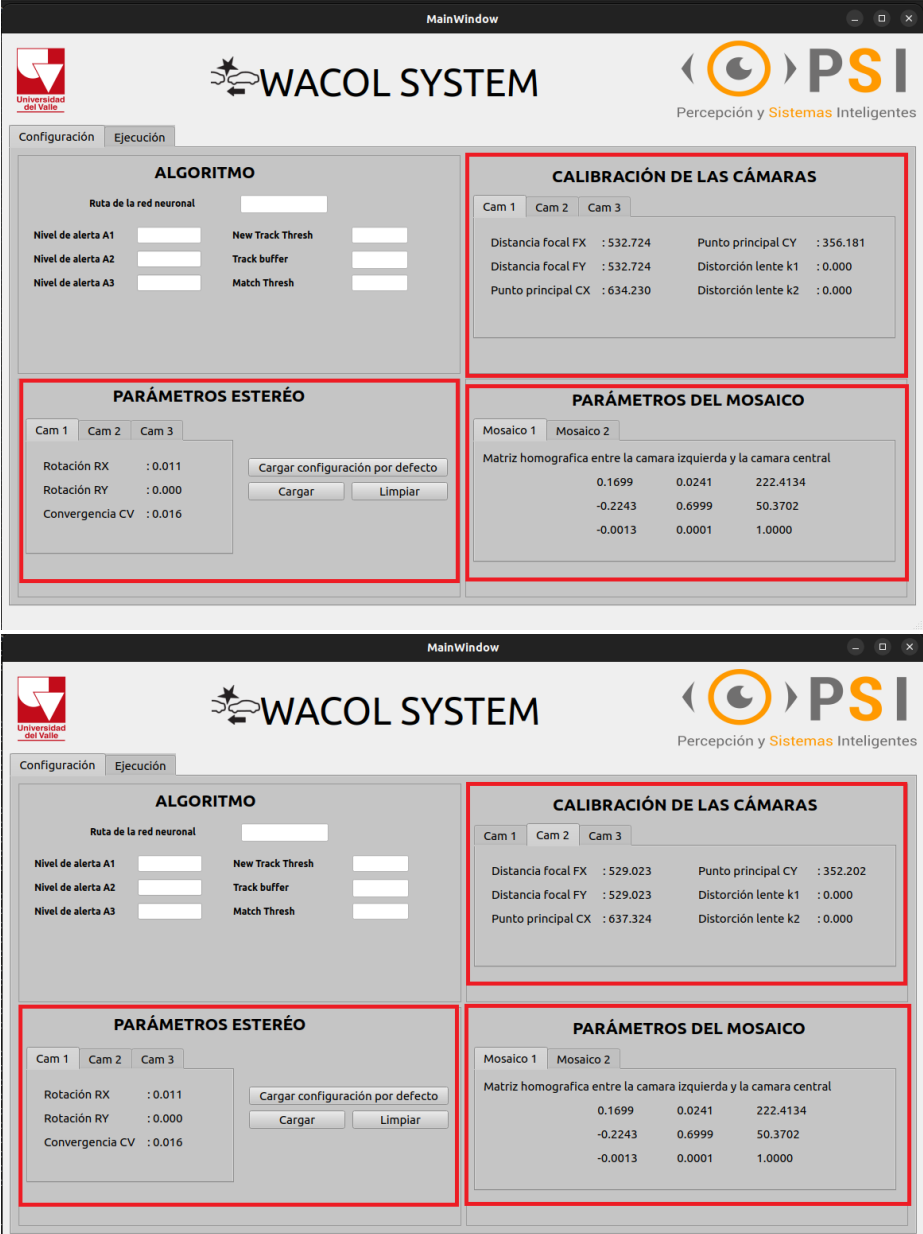
Se carga la red neuronal y los parámetros del bytetrack



Se cargan los niveles de alerta



Eliminar el usr_config.yaml cuando se carga la configuración por defecto.	
Objetivo	Validar que el archivo “usr_config.yaml” se puede eliminar al cargar la configuración por defecto y cada nodo usa su configuración por defecto.
Datos de la prueba	En esta prueba solo se necesita revisar que al presionar el boton de cargar configuración por defecto el archivo “usr_config.yaml” se elimine y los nodos funcionen con su configuración por defecto.
Procedimiento	1. Presionar el botón “cargar configuración por defecto” 2. Revisar que el archivo fue borrado exitosamente. 3. Revisar que los nodos funcionan con la configuración por defecto.
Resultado esperado	1. Se elimina el archivo “usr_config.yaml”. 2. Se revisa internamente que los nodos funcionen con su configuración por defecto.
Resultados obtenidos	Se elimina el archivo “usr_config.yaml”  The screenshot shows a file manager window with the path <code>Home / ros2_ws / src / gui_final / gui_final</code>. It displays several files: <code>gui_ros2.py</code>, <code>_init_.py</code>, <code>interfaz.py</code>, <code>pequeña prueba.py</code>, <code>_pycache_</code>, and <code>recursos_rc1.py</code>. Below the file manager, a terminal window shows the command <code>ros2 run gui_final gui</code> and its output. The logs indicate that the GUI node is initialized, the configuration is saved, and the file <code>/home/franklin/ros2_ws/src/gui_final/gui_final/usr_config.yaml</code> is successfully deleted. Se revisa que los nodos usan la configuracion por defecto.  The screenshot shows two terminal windows. The top window shows the command <code>ros2 run yolobot_recognition yolov8_ros2_pt.py</code> and an error message: <code>usr_config.yaml no encontrado. Restaurando backup de /home/franklin/ros2_ws/src/yolobot_recognition/scripts/bytetrack.yaml.backup a /home/franklin/ros2_ws/src/yolobot_recognition/scripts/bytetrack.yaml</code>. The bottom window shows the command <code>ros2 run yolobot_recognition yolov8_ros2_subscriber.py</code> and a warning message: <code>[WARN] [1747800635.133782014] [yolo_inference_subscriber]: Archivo de configuración /home/franklin/ros2_ws/src/gui_final/gui_final/usr_config.yaml no encontrado. Usando valores por defecto.</code>
Visualizar los parámetros extrínsecos e intrínsecos de las cámaras, las homografías y los parámetros estereo.	

Objetivo	Validar que en la pestaña configuración se pueden observar los parámetros extrínsecos e intrínsecos de las cámaras, las homografías y los parámetros estereo, despues de iniciar la ejecución del aplicativo.
Datos de la prueba	En esta prueba solo se inicializará el aplicativo para funcionar con un rosbag, de modo que se pueda observar todos los parámetros mencionados anteriormente.
Resultado esperado	1. Visualizar los parámetros extrínsecos e intrínsecos de las cámaras, las homografías y los parámetros estereo.
Resultados obtenidos	Se pueden visualizar los datos de calibración de cada camara y los parámetros del mosaico.  The image displays two screenshots of the WACOL SYSTEM software interface, which is used for camera calibration and mosaic generation. The interface is titled 'MainWindow' and features the logos of the Universidad del Valle and PSI (Percepción y Sistemas Inteligentes). It has two tabs: 'Configuración' (Configuration) and 'Ejecución' (Execution). The 'Configuración' tab is active, showing four main sections: 'ALGORITMO' (Algorithm), 'CALIBRACIÓN DE LAS CÁMARAS' (Camera Calibration), 'PARÁMETROS ESTEREO' (Stereo Parameters), and 'PARÁMETROS DEL MOSAICO' (Mosaic Parameters). The 'CALIBRACIÓN DE LAS CÁMARAS' section is highlighted with a red box in both screenshots, showing parameters for three cameras (Cam 1, Cam 2, Cam 3). The 'PARÁMETROS ESTEREO' section is also highlighted with a red box, showing parameters for three cameras (Cam 1, Cam 2, Cam 3) and buttons for 'Cargar configuración por defecto', 'Cargar', and 'Limpiar'. The 'PARÁMETROS DEL MOSAICO' section is highlighted with a red box, showing parameters for two mosaics (Mosaico 1, Mosaico 2) and a table of homography matrices. The 'ALGORITMO' section is not highlighted and contains fields for 'Ruta de la red neuronal', 'Nivel de alerta A1', 'Nivel de alerta A2', 'Nivel de alerta A3', 'New Track Thresh', 'Track buffer', and 'Match Thresh'.

Universidad
del Valle

 WACOL SYSTEM

 PSI
Percepción y Sistemas Inteligentes

Configuración

Ejecución

ALGORITMO

Ruta de la red neuronal

Nivel de alerta A1

Nivel de alerta A2

Nivel de alerta A3

New Track Thresh

Track buffer

Match Thresh

CALIBRACIÓN DE LAS CÁMARAS

Cam 1

Cam 2

Cam 3

Distancia focal FX : 534.464

Distancia focal FY : 534.464

Punto principal CX : 647.290

Punto principal CY : 352.663

Distorsión lente k1 : 0.000

Distorsión lente k2 : 0.000

PARÁMETROS ESTEREO

Cam 1

Cam 2

Cam 3

Rotación RX : 0.011

Rotación RY : 0.000

Convergencia CV : 0.016

Cargar configuración por defecto

Cargar

Limpiar

PARÁMETROS DEL MOSAICO

Mosaico 1

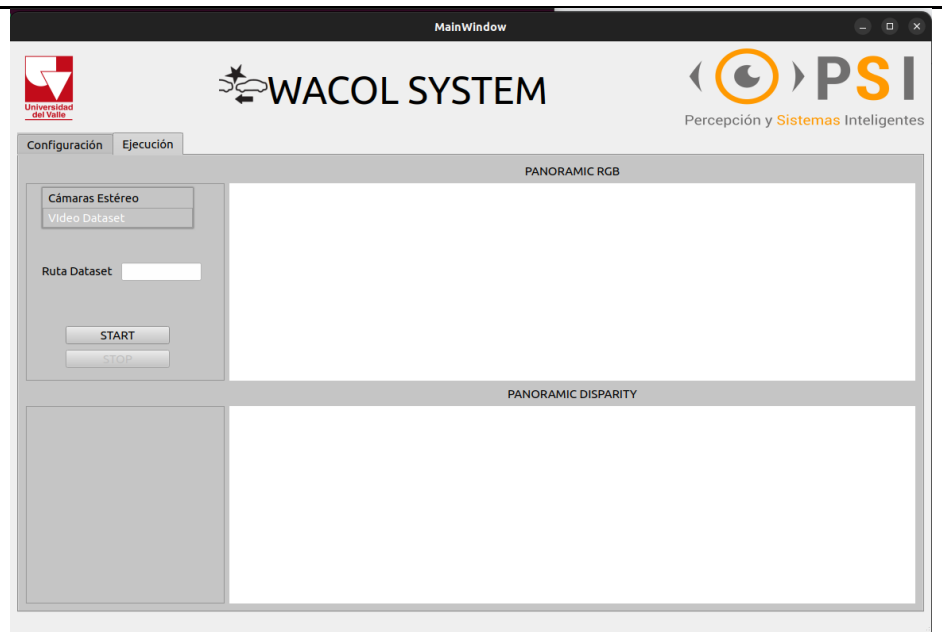
Mosaico 2

Matriz homografica entre la camara derecha y la camara central

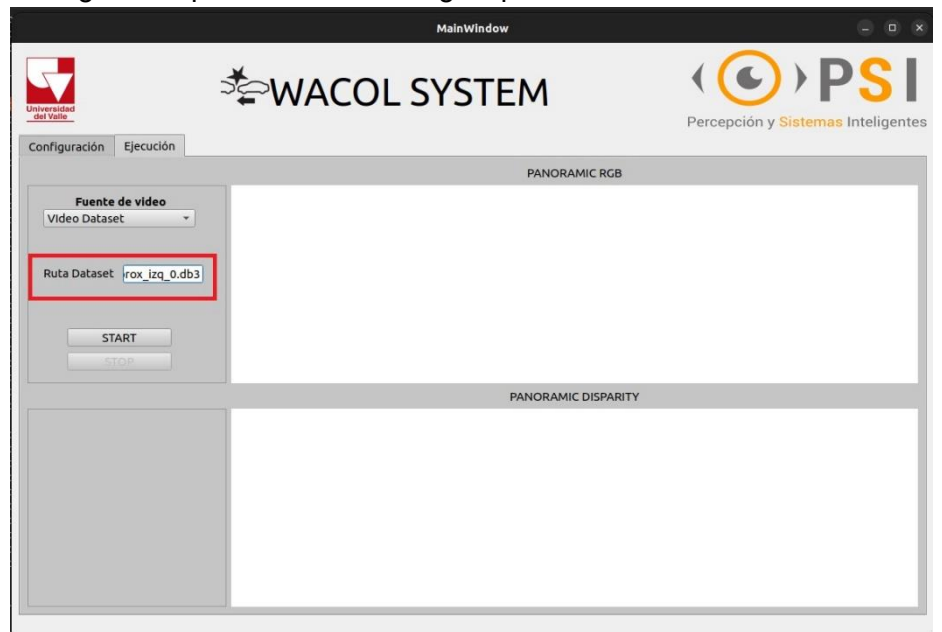
8.9369	0.1666	-2183.2483
1.9610	5.9438	-802.4142
0.0121	0.0006	1.0000

Nota: Cabe mencionar que los parámetros esteréo son iguales para las 3 cámaras.

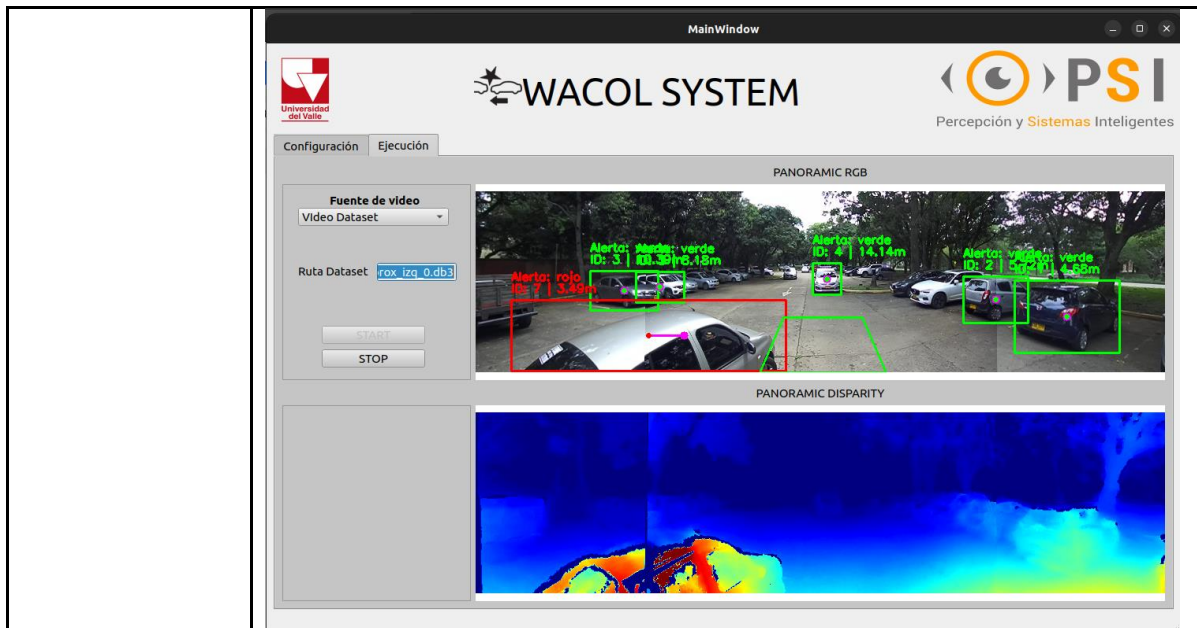
PRUEBA DE INTEGRACIÓN – MÓDULO DE EJECUCIÓN	
• Seleccionar una fuente de información (cámaras o bagfile). • Iniciar y detener el proceso. • Visualizar la panorámica RGB resultante con los datos procesados y la panorámica de disparidad resultante.	
Tipo de prueba	Integración
Hardware Requerido	Portatil: • Procesador (CPU): Intel core i5 10300 H. • Tarjeta gráfica (GPU): Nvidia GTX 1650 Mobile. • Memoria RAM: 16 GB. • Memoria RAM de la GPU: 4 GB. 3 cámaras ZED2
Software requerido	Aplicativo de alerta de colisión con otros vehículos, (Python 3.10.12), ZED SDK, Ubuntu 22.04.5 LTS, Qt designer 5, CUDA, NVCC.
Objetivo	Validar el correcto funcionamiento del módulo de ejecución.
Seleccionar una fuente de información (cámaras o bagfile)	
Objetivo	Validar el correcto funcionamiento del aplicativo para seleccionar un video del tipo bagfile como fuente de información.
Datos de la prueba	Esta prueba se realiza seleccionando la opción de “Video Dataset” o en el comboBox que se encuentra en la pestaña “Ejecución” dentro del módulo de ejecución. Posteriormente ingresar el path al qlabel donde se indica la ruta del data set. Bagfile path: home/franklin/data/vehiculo_stop_1_aprox_izq /vehiculo_stop_1_aprox_izq_0.db3
Procedimiento	1. Seleccionar el comboBox de “Video Dataset” 2. Pegar el path al archivo bagfile que se quiere usar en el qlabel. 3. Presionar el boton “START”
Resultado esperado	1. Se carga el archivo. 2. Se visualiza el video
Resultados obtenidos	Se selecciona “Video Dataset” como fuente de información para la prueba.



Se ingresa el path del archivo bagfile para usarlo.



Se presiona el botón "Start"

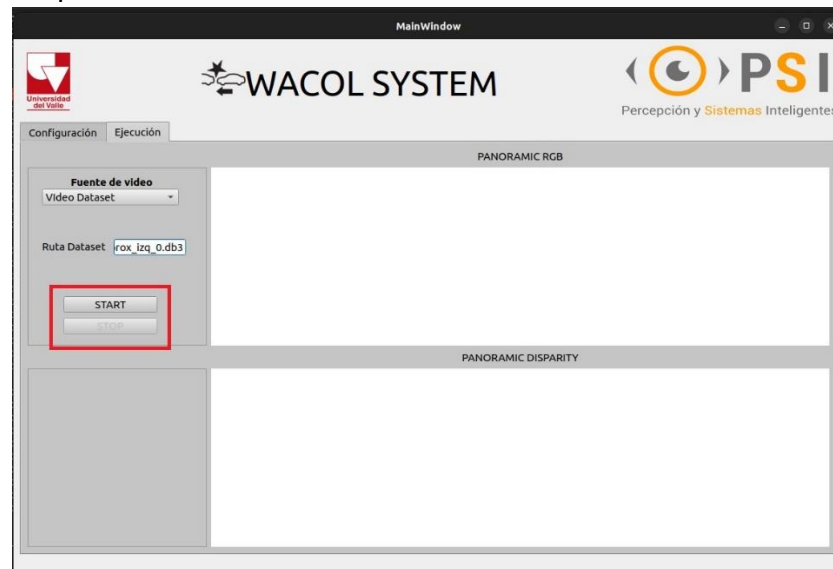


Iniciar y detener el proceso.

Objetivo	Validar el correcto funcionamiento del aplicativo para iniciar o detener el sistema.
Datos de la prueba	Esta prueba se realiza ya con la fuente de información seleccionada y presionando el botón de Start, antes de iniciar el sistema solo tiene habilitado el botón de Start, luego de presionarlo se inhabilita y se habilita el botón Stop para detener el proceso.
Procedimiento	1. Seleccionar la fuente de información. 2. Presionar el botón "START" para empezar. 3. Presionar el botón "STOP" para detener.
Resultado esperado	1. Se carga el archivo. 2. Se visualiza el video presionando el botón de Start. 3. Se deja de visualizar el video presionando el botón Stop.

Resultados obtenidos

Se presiona el botón “Start”



Se visualiza en la terminal que el proceso inicio.

```
franklin@franklin-Nitro: ~  
franklin@franklin-Nitro:~$ ros2 run gui_final gui  
[INFO] [1748097921.574260772] [gui_node]: Nodo GUI con PyQt5 iniciado.  
Warning: Ignoring XDG_SESSION_TYPE=wayland on Gnome. Use QT_QPA_PLATFORM=wayland to run on Wayland anyway.  
[INFO] [1748097930.141516533] [gui_node]: Seleccione Video Dataset.  
[INFO] [1748097930.142160106] [gui_node]: Reproduciendo archivo: /home/franklin/data/bagfiles/vehiculo_stop_1_aprox_izq/vehiculo_stop_1_aprox_izq_0.db3  
[INFO] [1748097930.142465424] [gui_node]: Construyendo panoramica RGB  
[INFO] [1748097930.143248481] [gui_node]: Construyendo panoramica de profundidad  
[INFO] [1748097930.143528449] [gui_node]: Inicializando red neuronal
```

Se presiona el boton “STOP”.

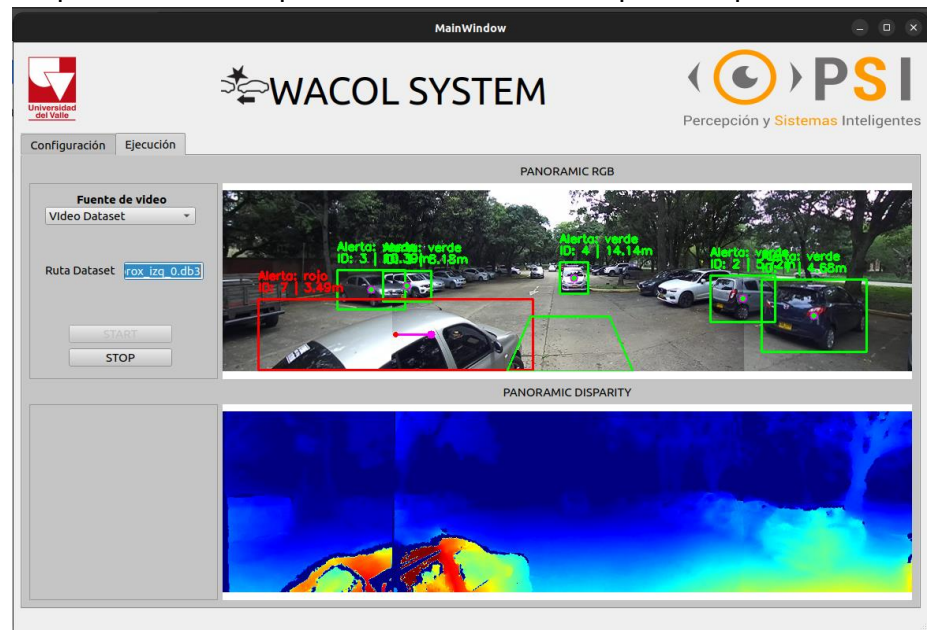


Se visualiza en la terminal que el proceso se detiene.

	<pre>franklin@franklin-Nitro: ~ franklin@franklin-Nitro:~\$ ros2 run gui_final gui [INFO] [1748097921.574260772] [gui_node]: Nodo GUI con PyQt5 iniciado. Warning: Ignoring XDG_SESSION_TYPE=wayland on Gnome. Use QT_QPA_PLATFORM=wayland to run on Wayland anyway. [INFO] [1748097930.141516533] [gui_node]: Selecciono Video Dataset. [INFO] [1748097930.142160106] [gui_node]: Reproduciendo archivo: /home/franklin/data/bagfiles/vehiculo_stop_1_aprox_izq/vehiculo_stop_1_aprox_izq_0.db3 [INFO] [1748097930.142465424] [gui_node]: Construyendo panoramica RGB [INFO] [1748097930.143248481] [gui_node]: Construyendo panoramica de profundidad [INFO] [1748097930.143528449] [gui_node]: Inicializando red neuronal [INFO] [1748097984.953738680] [gui_node]: Deteniendo reproducción del archivo rosbag. [INFO] [1748097984.955400308] [gui_node]: Deteniendo todos los nodos. [INFO] [1748097984.955936694] [gui_node]: ----- ----- []</pre> Se verifica que ya no se muestra nada en la GUI.
Visualizar la panorámica RGB resultante con los datos procesados y la panorámica de disparidad resultante.	
Objetivo	Validar que en la pestaña de ejecucion el aplicativo es capaz de reproducir el video de la panoramica RGB y de disparidad según su fuente de video.
Datos de la prueba	En esta prueba solo se inicializará el aplicativo para funcionar con un rosbag, de modo que se pueda que los qlabels de la panoramica RGB y de disparidad se pueden visualizar sin problema.
Resultado esperado	1. Visualizar la panorámica RGB resultante con los datos procesados y la panorámica de disparidad resultante.

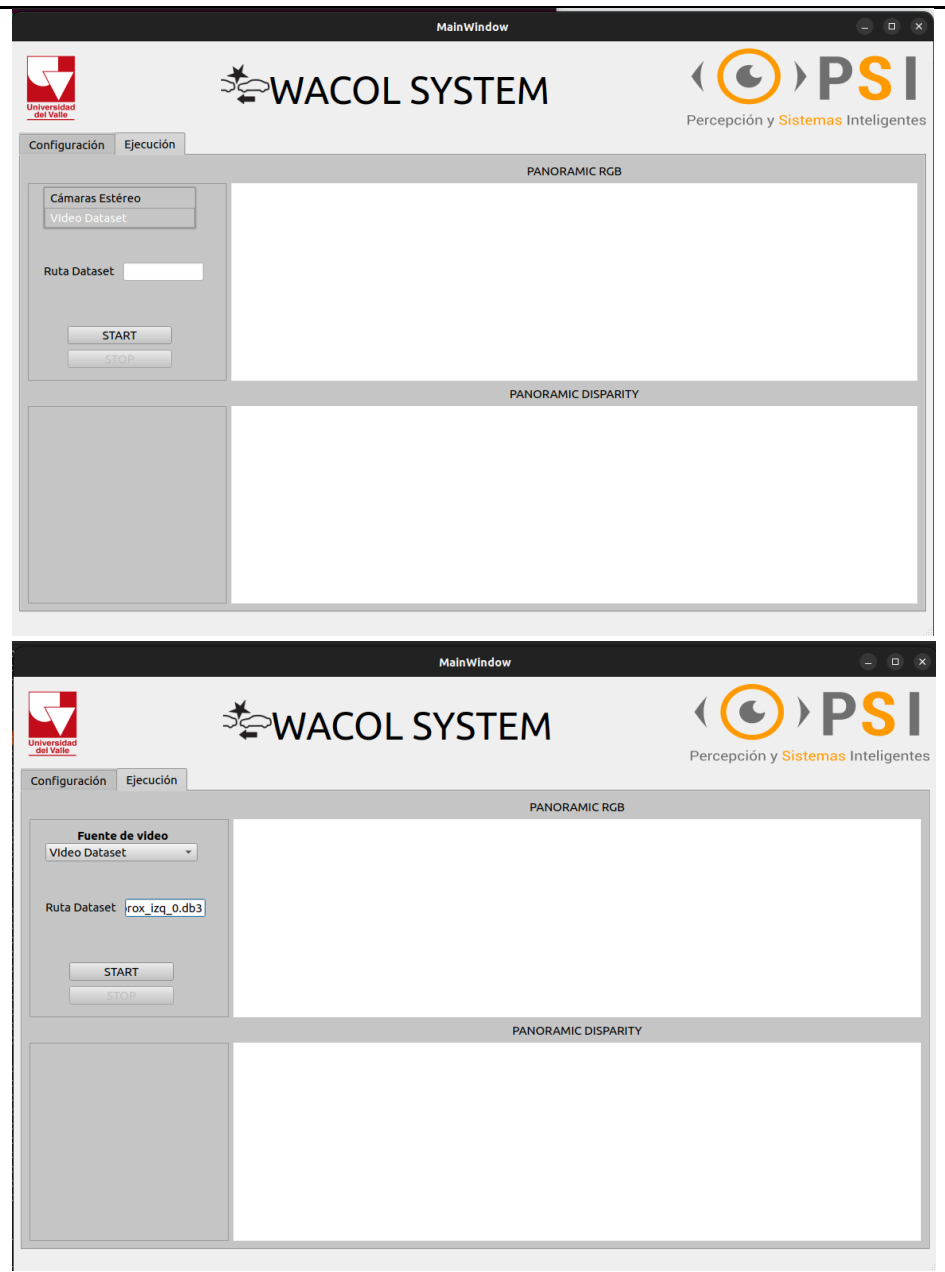
Resultados obtenidos

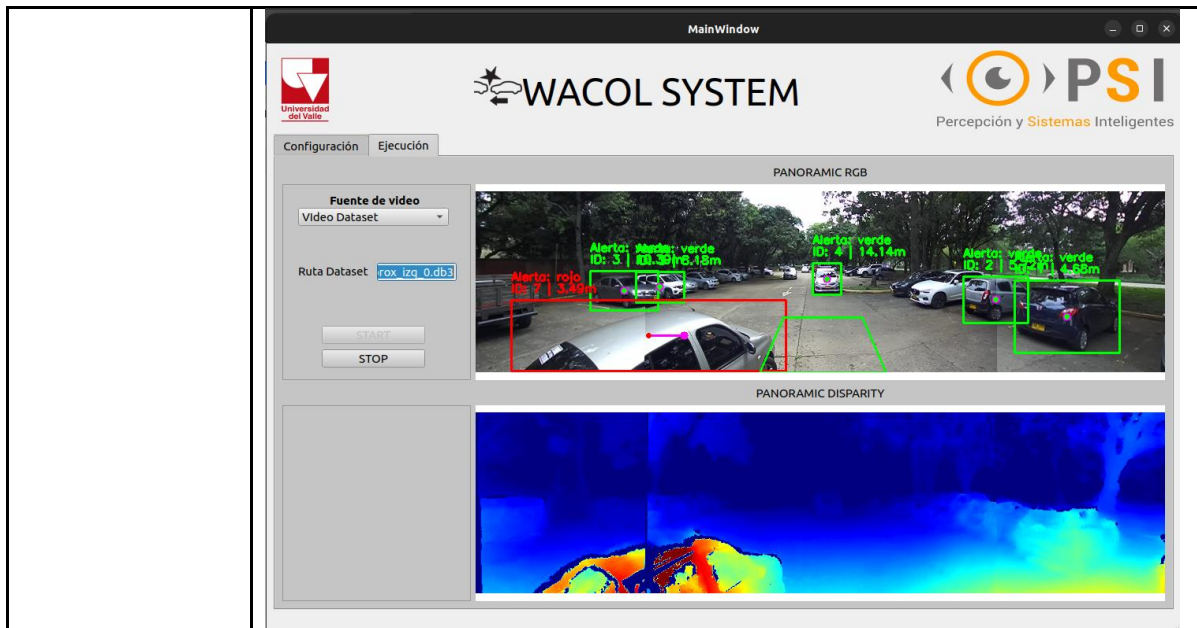
Se puede observar que la GUI muestra las respectivas panorámicas.



PRUEBA DE INTEGRACIÓN – MÓDULO DE EJECUCIÓN	
• Seleccionar una fuente de información (cámaras o bagfile). • Iniciar y detener el proceso. • Visualizar la panorámica RGB resultante con los datos procesados y la panorámica de disparidad resultante.	
Tipo de prueba	Integración
Hardware Requerido	Portatil: • Procesador (CPU): Intel core i5 10300 H. • Tarjeta gráfica (GPU): Nvidia GTX 1650 Mobile. • Memoria RAM: 16 GB. • Memoria RAM de la GPU: 4 GB. 3 cámaras ZED2
Software requerido	Aplicativo de alerta de colisión con otros vehículos, (Python 3.10.12), ZED SDK, Ubuntu 22.04.5 LTS, Qt designer 5, CUDA, NVCC.
Objetivo	Validar el correcto funcionamiento del módulo de ejecución.
Seleccionar una fuente de información (cámaras o bagfile)	
Objetivo	Validar el correcto funcionamiento del aplicativo para seleccionar un video del tipo bagfile como fuente de información.
Datos de la prueba	Esta prueba se realiza seleccionando la opción de “Video Dataset” o en el comboBox que se encuentra en la pestaña “Ejecución” dentro del módulo de ejecución. Posteriormente ingresar el path al qlabel donde se indica la ruta del data set. Bagfile path: home/franklin/data/vehiculo_stop_1_aprox_izq /vehiculo_stop_1_aprox_izq_0.db3
Procedimiento	1. Seleccionar el comboBox de “Video Dataset” 2. Pegar el path al archivo bagfile que se quiere usar en el qlabel. 3. Presionar el boton “START”
Resultado esperado	1. Se carga el archivo. 2. Se visualiza el video

Resultados obtenidos

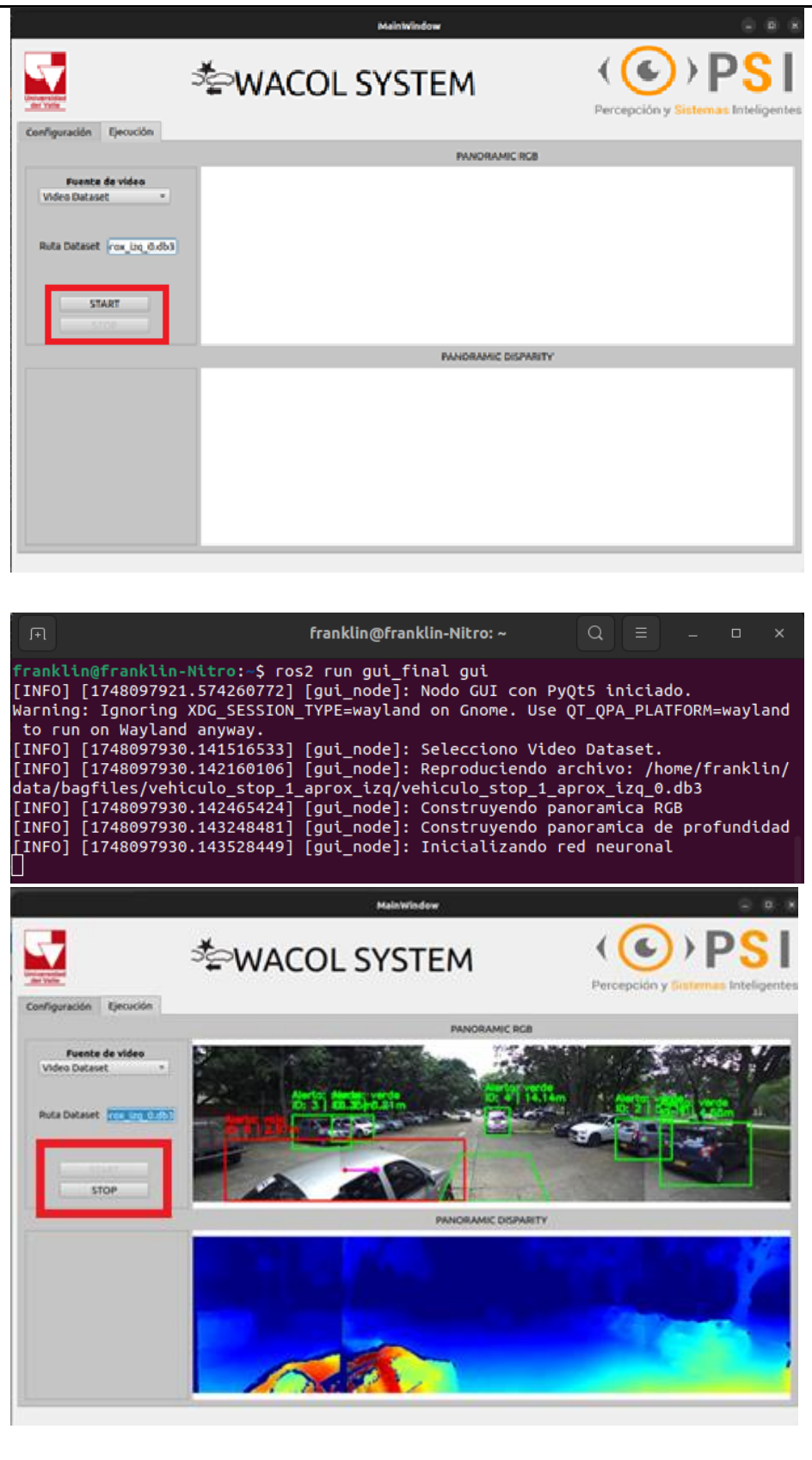


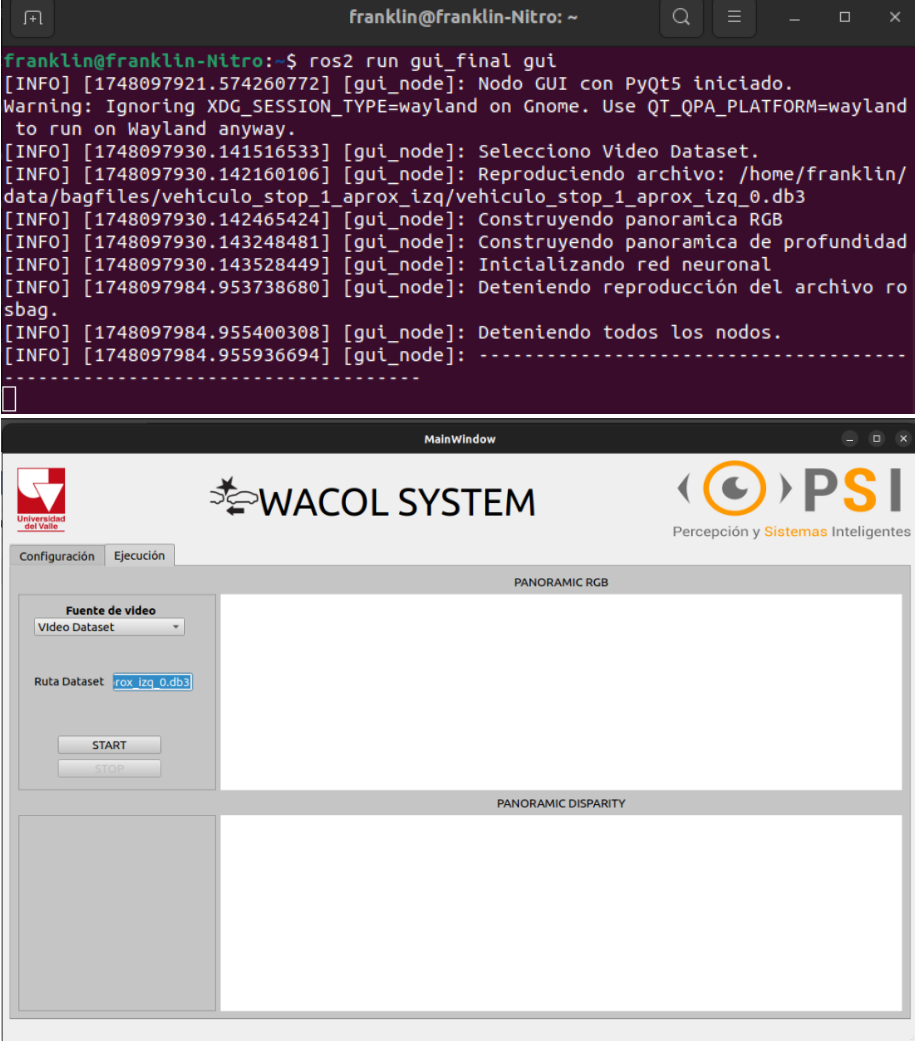


Iniciar y detener el proceso.

Objetivo	Validar el correcto funcionamiento del aplicativo para iniciar o detener el sistema.
Datos de la prueba	Esta prueba se realiza ya con la fuente de información seleccionada y presionando el botón de Start, antes de iniciar el sistema solo tiene habilitado el botón de Start, luego de presionarlo se inhabilita y se habilita el botón Stop para detener el proceso.
Procedimiento	1. Seleccionar la fuente de información. 2. Presionar el botón "START" para empezar. 3. Presionar el botón "STOP" para detener.
Resultado esperado	1. Se carga el archivo. 2. Se visualiza el video presionando el botón de Start. 3. Se deja de visualizar el video presionando el botón Stop.

Resultados obtenidos



	 <pre> franklin@franklin-Nitro: ~ franklin@franklin-Nitro:~\$ ros2 run gui_final gui [INFO] [1748097921.574260772] [gui_node]: Nodo GUI con PyQt5 iniciado. Warning: Ignoring XDG_SESSION_TYPE=wayland on Gnome. Use QT_QPA_PLATFORM=wayland to run on Wayland anyway. [INFO] [1748097930.141516533] [gui_node]: Selecciono Video Dataset. [INFO] [1748097930.142160106] [gui_node]: Reproduciendo archivo: /home/franklin/ data/bagfiles/vehiculo_stop_1_aprox_izq/vehiculo_stop_1_aprox_izq_0.db3 [INFO] [1748097930.142465424] [gui_node]: Construyendo panoramica RGB [INFO] [1748097930.143248481] [gui_node]: Construyendo panoramica de profundidad [INFO] [1748097930.143528449] [gui_node]: Inicializando red neuronal [INFO] [1748097984.953738680] [gui_node]: Deteniendo reproducción del archivo ro sbag. [INFO] [1748097984.955400308] [gui_node]: Deteniendo todos los nodos. [INFO] [1748097984.955936694] [gui_node]: ----- </pre>
Visualizar la panorámica RGB resultante con los datos procesados y la panorámica de disparidad resultante.	
Objetivo	Validar que en la pestaña de ejecucion el aplicativo es capaz de reproducir el video de la panoramica RGB y de disparidad según su fuente de video.
Datos de la prueba	En esta prueba solo se inicializará el aplicativo para funcionar con un rosbag, de modo que se pueda que los qlabels de la panoramica RGB y de disparidad se pueden visualizar sin problema.
Resultado esperado	1. Visualizar la panorámica RGB resultante con los datos procesados y la panorámica de disparidad resultante.

Resultados obtenidos

