

Guía para el uso de SAM2 como *backend* de segmentación en Label Studio

Andrés M. Salas Espinales

1. Introducción

La anotación de imágenes para tareas de segmentación semántica es un proceso costoso y propenso a errores cuando se realiza de manera completamente manual. En este contexto, herramientas de anotación asistida por inteligencia artificial permiten acelerar el proceso, mejorar la consistencia entre anotadores y reducir la carga cognitiva del usuario.

Label Studio es una plataforma de anotación flexible y extensible que permite integrar modelos de aprendizaje profundo como *backends* inteligentes. En esta guía se describe el procedimiento para integrar **Segment Anything Model 2 (SAM2)** como *backend* de segmentación en Label Studio, siguiendo como referencia la guía oficial de Label Studio¹.

2. Objetivos

- Configurar Label Studio mediante Docker.
- Integrar SAM2 como *backend* de segmentación automática.
- Diseñar un template de etiquetado multimodal.
- Configurar el acceso a imágenes almacenadas en un servidor.
- Realizar anotaciones asistidas por IA.

3. Requisitos previos

Antes de iniciar, el estudiante debe contar con:

- Acceso a un servidor Linux.
- Docker correctamente instalado y en ejecución.
- Permisos de usuario para ejecutar contenedores.

Para verificar la instalación de Docker, ejecute:

```
docker --version
```

¹<https://labelstud.io/blog/get-started-using-segment-anything/>

4. Arquitectura general del sistema

El flujo de trabajo se basa en una arquitectura cliente-servidor donde:

- Label Studio se ejecuta en un contenedor Docker.
- SAM2 opera como *backend* externo de inferencia.
- Las imágenes se almacenan y acceden directamente desde el servidor.

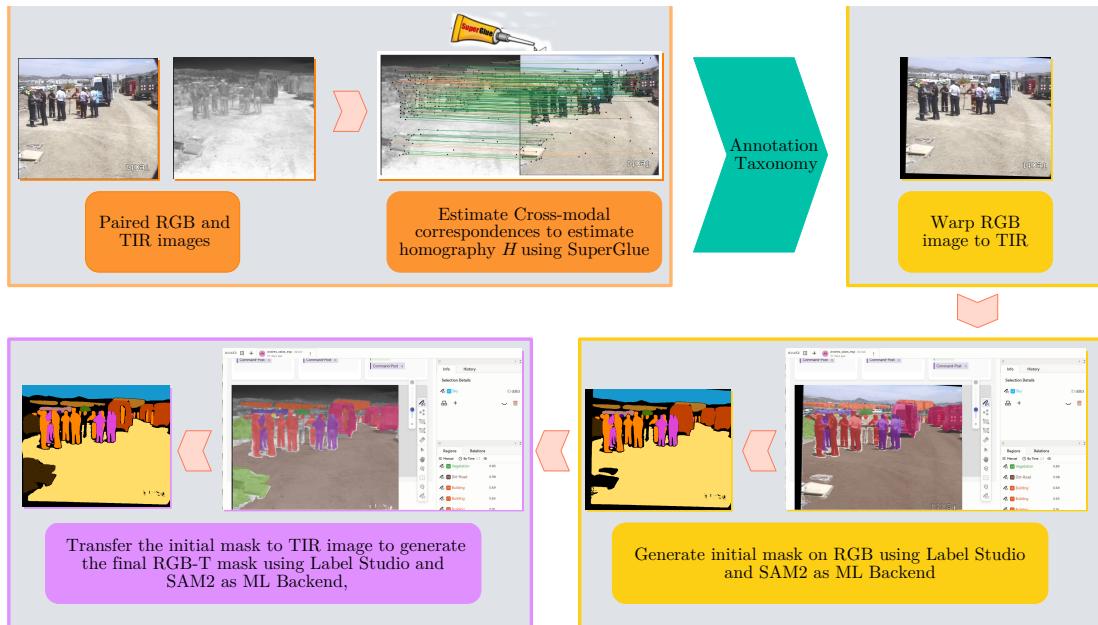


Figura 1: Arquitectura general de Label Studio con SAM2 como backend.

5. Instalación de Label Studio

Label Studio se ejecuta utilizando Docker. Para ello, se utiliza el siguiente comando:

```
docker run -it -p 8080:8080 --user root \
-v /NAS/home/asalas/RGB_T_SAR:/label-studio/data \
--env LABEL_STUDIO_LOCAL_FILES_SERVING_ENABLED=true \
--env LABEL_STUDIO_LOCAL_FILES_DOCUMENT_ROOT=/label-studio/data \
heartexlabs/label-studio:latest
```

Es fundamental que el directorio:

```
/NAS/home/asalas/RGB_T_SAR
```

exista previamente, ya que será el repositorio central de las imágenes utilizadas durante la anotación.

6. Configuración del backend SAM2

El backend SAM2 se encuentra en el repositorio de ejemplos de Label Studio. El directorio de trabajo es:

```
cd label-studio-ml-backend/label_studio_ml/examples/  
segmentAnything_2_image
```

A partir de este directorio se debe modificar el archivo `docker-compose.yml`, ajustando:

- La dirección IP del servidor.
- El *token* de autenticación generado por Label Studio.

```
# Specify the Label Studio URL and API key to access  
# uploaded, local storage and cloud storage files.  
# Do not use 'localhost' as it does not work within Docker containers.  
# Use prefix 'http://' or 'https://' for the URL always.  
# Determine the actual IP using 'ifconfig' (Linux/Mac) or 'ipconfig' (Windows).  
- LABEL_STUDIO_HOST=http://10.0.101.10:8080  
- LABEL_STUDIO_ACCESS_TOKEN=9128fab0ec30016774d8ae41fb84f541a7955817  
ports:  
- "9090:9090"
```

Figura 2: Configuración del backend SAM2 en `docker-compose.yml`.

7. Configuración del template de etiquetado

En la sección *Label Setup* de Label Studio, se debe seleccionar la opción *Custom Template* y pegar el siguiente código XML:

```
<View>  
<Style>  
    .main { font-family: Arial, sans-serif; background:#f5f5f5;  
            margin:0; padding:20px; }  
    .container { display:flex; gap:16px; margin-bottom:20px; }  
    .column { flex:1; padding:10px; background:#fff; border-  
              radius:8px; box-shadow:0 2px 5px rgba(0,0,0,0.08); text-  
              align:center; }  
    .column .title { margin:0; color:#333; font-weight:bold; }  
    .column .label { margin-top:10px; padding:10px; background:#  
                 fafafa; border-radius:6px; }  
    .image-container { width:100%; height:360px; background:#ddd;  
                      border-radius:8px; }  
</Style>  
  
<View className="main">  
  <View className="container">
```

```

<!-- BROCHA (segmentación por pixeles) -->
<View className="column">
    <View className="title">Brush (Segmentation)</View>
    <View className="label">
        <BrushLabels name="brush" toName="image">
            <Label value="First-Responder" background="#E53935"
                hotkey="1"/>
            <Label value="Civilian" background="#8E24AA" hotkey="2"/>
            <Label value="Vegetation" background="#43A047" hotkey="3"
                />
            <Label value="Road" background="#1E88E5" hotkey="4"/>
            <Label value="Dirt-Road" background="#6D4C41" hotkey="5"/
                >
            <Label value="Building" background="#F4511E" hotkey="6"/>
            <Label value="Sky" background="#29B6F6" hotkey="7"/>
            <Label value="Civilian-Car" background="#FDD835" hotkey="8"/>
            <Label value="Responder-Vehicle" background="#D81B60"
                hotkey="9"/>
            <Label value="Debris" background="#7CB342" hotkey="0"/>
            <Label value="Command-Post" background="#5E35B1" hotkey="q"/>
        </BrushLabels>
    </View>
</View>

<!-- PUNTOS (keypoints) -->
<View className="column">
    <View className="title">Keypoints</View>
    <View className="label">
        <KeyPointLabels name="kpts" toName="image" smart="true">
            <Label value="First-Responder" background="#E53935"
                hotkey="1"/>
            <Label value="Civilian" background="#8E24AA" hotkey="2"/>
            <Label value="Vegetation" background="#43A047" hotkey="3"
                />
            <Label value="Road" background="#1E88E5" hotkey="4"/>
            <Label value="Dirt-Road" background="#6D4C41" hotkey="5"/
                >
            <Label value="Building" background="#F4511E" hotkey="6"/>
            <Label value="Sky" background="#29B6F6" hotkey="7"/>
            <Label value="Civilian-Car" background="#FDD835" hotkey="8"/>
            <Label value="Responder-Vehicle" background="#D81B60"
                hotkey="9"/>
            <Label value="Debris" background="#7CB342" hotkey="0"/>
            <Label value="Command-Post" background="#5E35B1" hotkey="q"/>
        </KeyPointLabels>
    </View>
</View>

```

```

        q"/>
    </KeyPointLabels>
</View>
</View>

<!-- RECTANGULOS (bounding boxes) -->
<View className="column">
    <View className="title">Rectangles (Boxes)</View>
    <View className="label">
        <View><Filter toName="rects" minLength="0" name="filter"/><
            RectangleLabels name="rects" toName="image" smart="true"
            >
            <Label value="First-Responder" background="#E53935"
                hotkey="1"/>
            <Label value="Civilian" background="#8E24AA" hotkey="2"/>
            <Label value="Vegetation" background="#43A047" hotkey="3"
                />
            <Label value="Road" background="#1E88E5" hotkey="4"/>
            <Label value="Dirt-Road" background="#6D4C41" hotkey="5"/
                >
            <Label value="Building" background="#F4511E" hotkey="6"/>
            <Label value="Sky" background="#29B6F6" hotkey="7"/>
            <Label value="Civilian-Car" background="#FDD835" hotkey="8"/>
            <Label value="Responder-Vehicle" background="#D81B60"
                hotkey="9"/>
            <Label value="Debris" background="#7CB342" hotkey="0"/>
            <Label value="Command-Post" background="#5E35B1" hotkey="q"/>
        </RectangleLabels></View>
    </View>
</View>

</View>

<View className="image-container">
    <Image name="image" value="$image" zoom="true" zoomControl="true"/>
</View>
</View>
</View>

```

Este template permite combinar:

- Segmentación por píxeles (Brush).
- Puntos clave (Keypoints).
- Cajas delimitadoras (Bounding Boxes).

8. Configuración del almacenamiento de imágenes

Dado que el trabajo se realiza en un servidor, las imágenes no se almacenan localmente. Para habilitar el acceso, se debe configurar el almacenamiento en:

Settings → Cloud Storage

- Storage Type: Local files
- Absolute local path:

```
/label-studio/data/RGB_T_SAR
```

- File Filter Regex:

```
.*png
```

- Activar: *Treat every bucket object as a source file*

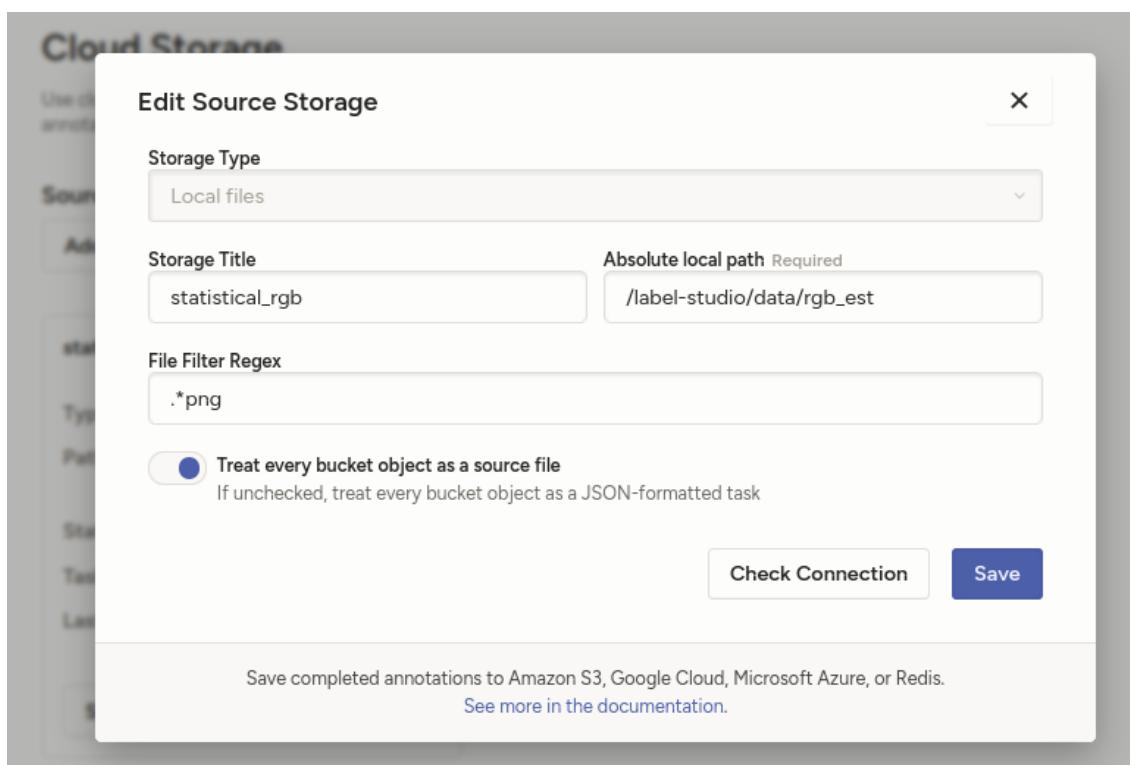


Figura 3: Configuración de almacenamiento local en Label Studio.

9. Flujo de trabajo de anotación

Una vez configurado el sistema, el flujo de anotación es el siguiente:

1. Seleccionar una imagen.
2. Colocar puntos, cajas o pincel sobre el objeto.
3. SAM2 genera automáticamente la máscara.
4. El anotador refina manualmente la segmentación.