Manual de uso X-anylabeling

Página git: https://github.com/CVHub520/X-AnyLabeling

Quick Start Guide

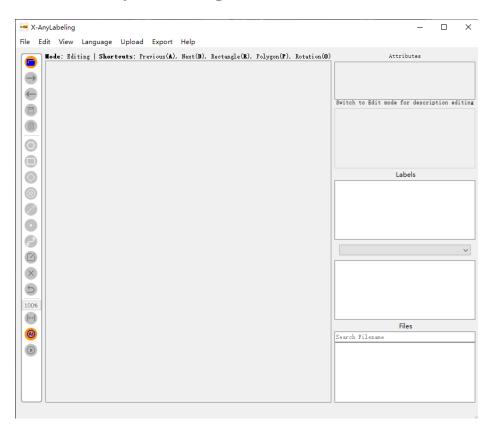
Git page : https://github.com/CVHub520/X- AnyLabeling/blob/main/docs/en/get_started.md

Instalación fácil, https://github.com/CVHub520/X-AnyLabeling/releases/tag/v2.2.0

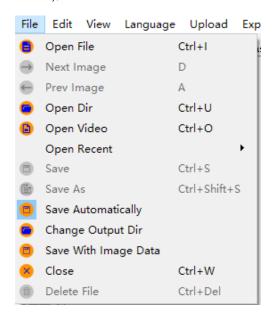
Entrar a la página y descargar una de las opciones adaptables al sistema operativo y si quiere utilizar GPU, tiene también los códigos y ejecutables en Python si quiere ver/modificar algunas funcionalidades y compilar el ejecutable con pyinstaller, tiene todas las informaciones en la página git oficial.

☆X-AnyLabeling-CPU.exe	150 MB	2 weeks ago
⊗ X-AnyLabeling-GPU.exe	873 MB	2 weeks ago
⊗ X-Anylabeling-Linux-CPU	208 MB	2 weeks ago
	347 MB	2 weeks ago

GUI X-AnyLabeling



En el menú 'File', podemos abrir un Directorio de imágenes (Open Dir) o Crear automáticamente el directorio a 30Frames/Segundo a partir de un vídeo dando (Open Video), el directorio crea en la misma ruta donde está el vídeo con el nombre del vídeo.





En el Menú Upload y Export se puede especificar el formato de anotaciones para crear el dataset, en la opción de Upload puede subir un fichero que contiene la informaciones de clases que queremos anotar, por ejemplo classes.txt con nombres de las categorías en la opción YOLO Annotation.



Puede Exportar más de un formato, la opción Upload de clases es opcional, solo se recomienda si tiene muchas clases que quiere etiquetar, ya que se puede crear clases en la herramienta cuando acaba de segmentar/hacer bounxing boxes.

En la barra Izquierda contiene operaciones básicas de una herramienta de etiquetar imágenes, como crear polígonos, rectángulos, líneas o puntos y modificar.



Opción Autolabeling con el icono es la ventaja relevante de esta herramienta, que utiliza los modelos SOTA como YOLO, SAM, DAMO, o cualquier modelo personalizada

que nos ayuda a hacer el proceso de etiquetación de las imágenes.

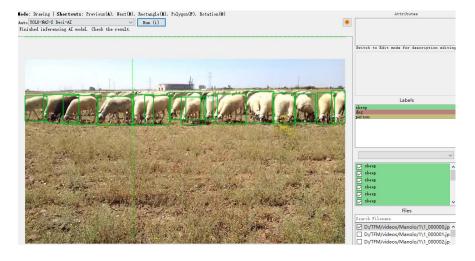
GUI X-AnyLabeling

```
Mode: Editing | Shortcuts: Previous(A), Next(B),
Auto No Model
Read No Model
...Load Custom Model
(User) Segment Anything (ViT-Base)
ch_PP-OCRv4 (PaddleOCR)
CLRNet (tusimple_r18)
DAMO-YOLO-L (Alibaba)
DAMO-YOLO-M (Alibaba)
DAMO-YOLO-S (Alibaba)
DAMO-YOLO-T (Alibaba)
DAMO-YOLO-T (Alibaba)
EdgeSAM
```

En la menú de opciones nos proporciona los modelos reciéntes y útil para hacer AutoLabeling, Al cargar el (User) Segment Anything, aparece opciones como añadir un punto, eliminar el punto, crear un rectángulo, finalizar(guardar datos etiquetadas), y puede elegir el tipo de Output en formá de polígono o en rectángulo.



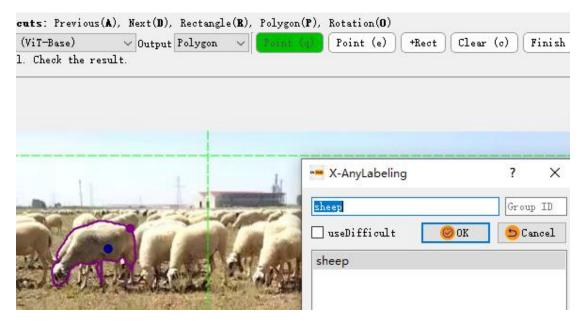
El modelo YOLO puede etiquetar totalmente automático la imagen click el botón Run, que infiere el modelo para hacer la detección en este caso.



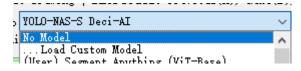


Esta opción puede aplicar a todas las imágenes del directorio, así que encontrar un buen modelo puede ahorrar mucho tiempo de trabajo.

El modelo SAM ayuda mucho para hacer la segmentación, no tiene que dibujar la figura manualmente, slo segmenta automáticamente.



Use Custom Model



X-AnyLabeling tiene la opción de cargar un modelo personalizado, en la página https://github.com/CVHub520/X-AnyLabeling/blob/main/docs/en/model_zoo.md

contiene una lista de modelos SOTA que se puede utilizar para la clasificación, Keypoint Detection, Object Detection, ···etc.

∂ Classification

Name	Description	Configuration	Size	Link
pulc_person_attribute.onnx	PersonAttribute-PULC	pulc_person_attribute.yaml	6.59MB	<u>baidu</u> <u>github</u>
pulc_vehicle_attribute.onnx	VehicleAttribute-PULC	pulc_vehicle_attribute.yaml	6.55MB	baidu github
internimage_l_22kto1k_384.onnx	<u>InternImage</u> -Large	internimage_l_22kto1k_384.yaml	853.16MB	baidu <u>github</u>
yolov5s-cls.onnx	YOLOv5-Cls-ImageNet	yolov5s_cls.yaml	20.81MB	baidu github
yolov8s-cls.onnx	YOLOv8-Cls-ImageNet	yolov8s_cls.yaml	24.28MB	baidu github

Keypoint Detection

• Facial Landmark Detection

Name	Description	Configuration	Size	Link
yolov6lite_l_face.onnx	Facial Landmark Detection	yolov6lite_l_face.yaml	4.16MB	<u>baidu github</u>
yolov6lite_m_face.onnx	Facial Landmark Detection	yolov6lite_m_face.yaml	3.00MB	<u>baidu github</u>
yolov6lite_s_face.onnx	Facial Landmark Detection	yolov6lite_s_face.yaml	2.10MB	<u>baidu github</u>

Para cargar el modelo tiene que tener un fichero .yaml con las informaciones del modelo, por ejemplo en la figura siguiente:



Tiene que espeficicar la ruta del modelo y decir algunos parámetros que quiere usar, en caso de tener el modelo en local, cambiar la ruta de github al directorio donde lo contiene como aparece en la siguiente figura:

NOTA: tiene el fichero .yaml el mismo directorio que los ficheros de codificación y decodificación onnx.