

## Instructions for Dataset Creation using Video Data with UNO Tools and CVAT

### Preparation

1. Install Computer Vision Annotation Tool (CVAT) using the instructions on their github page here: <https://github.com/openvinotoolkit/cvat>
2. CVAT is the only externally developed tool here, the rest of the tools can be found in our google drive folders. Install to a convenient location for command line input. Here are the necessary tools, and other recommended tools to have on hand. Download the most current version of each tool.
  - a. Required: VideoByFrames.py
  - b. Recommended: DatasetResizer.py, FODtoPASCAL.py

### Annotation Process

3. Input video data into the VideoByFrame Tool. The instructions for the use of this tool are contained in the documentation folder contained with the .py file.
4. The VideoByFrames.py tool will output a folder structure and a video that is trimmed to the designated specifications. It is recommended to trim the video into segments where the target object is contained in each frame. VideoByFrames.py will also output each individual frame contained in the video. Follow these steps from here:
  - a. First, create a new task in CVAT.
  - b. Then, create a descriptive name for the task.
  - c. Drag the **trimmed** video (be sure this is the trimmed video, not the original video) that VideoByFrames.py output into CVAT.
  - d. Add any object class labels that you will be annotating.
  - e. Submit the task, this will take a moment to process.
5. Return to the tasks window in CVAT once the task is done processing. The new task will be at the top. Click open in the relevant task, and then click the blue highlighted Job # towards the bottom of the screen. The video will now show up. Use these tips for an efficient annotation process (This is for bounding box annotation on videos. If you need to use a different form of annotation, follow your own steps to annotate in CVAT.):
  - a. Click the draw new rectangle button. Ensure the correct object class is selected. Select the "Track" button and draw the initial rectangle.
  - b. You now only need to annotate every 10<sup>th</sup> frame. Use the go to next step (Default Hotkey [V]) button to skip 10 frames.
  - c. Adjust the initially drawn bounding box to fit the object again, including size and location. CVAT will automatically create the bounding box for the frame in between.

- d. Not every automatically drawn bounding box will be perfectly located and sized. Once the initial bounding boxes are located, watch the video, and adjust any incorrect bounding boxes as necessary.
  - e. Once the entire video is properly annotated, save the task.
  - f. Return to the “Tasks” menu in CVAT.
6. Next, we need export our new annotations to the file structure created by the VideoByFrames.py tool.
  - a. Find the correct task in CVAT
  - b. Click the vertical “...” to open more options for the task.
  - c. Use the dump annotations option to export the annotation (export as Pascal VOC for now) do not use the export as dataset option as this will take significantly longer. This will download a .zip file.
  - d. Once extracted, copy only the “Annotations” folder contained in the extracted zip file to the folder structure created by the VideoByFrame.py tool to be next to the corresponding “frame” folder. Be sure to keep the folder name as “Annotations.”
7. Repeat steps 3-6 until the entire dataset is annotated.

#### Final Dataset Processing

8. The individual trimmed videos can now be removed. These videos can also be left in the dataset if desired.
9. Recommended step: It is likely that the resultant dataset is rather large (takes up many bytes) with ununiform image dimensions. The DatasetResizer.py tool will resize all images and annotations to a specified size. This will also reduce the overall byte size of the dataset drastically. If this is desirable, input the new dataset into the resizer tool using a convenient size. The instructions for use of the tool are contained in a txt file included with the tool.

**NOTE:** It is recommended to create a copy of the original dataset before resizing if the storage space is available. Use the original dataset as a complete backup.
10. Recommend step: To reduce the size of the dataset further and to conform to a common format, input the dataset into the FODtoPASCAL.py tool. This will convert the entire dataset to the Pascal VOC format (the tool exports a new Pascal VOC dataset and leaves the original untouched). The tool will leave out any empty annotation files and corresponding images if these exist. FODtoPascal.py will also create dataset splits based on specified percentages. These split files (as txt files in the resultant ImageSets folder) will also follow the Pascal VOC format. Usage instructions are contained with the tool.
11. If expanding the dataset is necessary, simply follow steps 3-6 using the original dataset. Clean up the expanded dataset using steps 8-10.