LabPics Chemistry Dataset

LabPics Medical Dataset

The dataset contains annotated images for both material and vessels in chemistry labs, medical labs, and any area where liquids and solids are handled within vessels. There are two levels of annotation for each image. One annotation set for vessels and the second for the material phases inside these vessels. Vessels are defined as any container that can carry materials such as Jars, Erlenmayers, Tubes, Funnels, syringes, IV bags, and any other labware or glassware that can contain or carry materials. Material phases are any material contained within or on the vessel. For example, for two-phase separating liquids, each liquid phase is annotated as one instance. If there is foam above the liquid or a chunk of solid inside the liquid, the foam, liquid, and solid will be annotated as different phases. In addition, vessel parts like cork, labels, and valves are annotated as instances. For each instance, there is a list of all the classes it belongs to, and a list of its property. For vessels, the instance classes are the vessel type (Cup, jar, Separatory-funnel…) and the vessel properties (Transparent, Opaque…). For materials, the classes are the material types (Liquid, solid, suspension, foam, powder…) and their properties (Scattered, On vessel surface…), and for parts, the part type (cork/label). In addition, the relations between instances are annotated. This includes which materials instances are inside which vessels, which vessels are linked to each other or are inside each other (for vessels inside other vessels), and which material phase is immersed inside another material phase. In addition to instance segmentation maps, the dataset also includes semantic segmentation maps that give for each pixel in the image all the classes to which it belongs. In other words, for each class (Liquid, Solid, Vessel, Foam), there is a map of all the regions in the image belonging to this class. Note that every pixel and every instance can have several classes. In addition, instances often overlap, like in the case of material inside the vessel, vessel inside the vessel, and material phase immerse inside other material (like solid inside liquid).

File sources and copyright

Creating this dataset was impossible without a community of chemists who take and share high-quality photos of their experiments. Most of the images of this dataset were shared by these sources. Copyright for all images belongs to their contributors. The dataset is shared for academic purposes only. For any other use of the images, inquire with the image source. The source of every image is mention in the source.txt file located next to the image file. Specific copyright for each image also appears in the same file. If no copyright is mentioned, contact the image source for inquiries. See Image sources and copyright file in main folder and source.txt in image folder for detail.

1.Classes/Categories:

The list of categories used in this dataset and their description are available in the Categories folder as text and json files.

2. Reader scripts.

Several example reader python scripts are supplied in the /ReaderScriptsPython/ folder. Examining these readers is probably the best way to start working with the dataset.

2.1 Evaluation scripts

Python scripts for evaluation of prediction vs. data set can be found in /*EvaluationScripts* folder. The folder is divided into vessel content methods to evaluate segmentation of vessel content for each vessel separately, and full image to evaluate segmentation of the full image.

3. Files structure

Each annotated image is assigned a subfolder that contains the image and all the annotation data.

Main Files:

Image.png: (RGB) image of the system.

Ignore.png: 2d map of the regions of the image that were not annotated. Regions in this map that are marked one were not annotated and should be ignored. All other regions marked 0 and were fully annotated. All vessels in the image were either annotated completely, including their content, or were ignored completely, including their content.

Source.txt: Source of the image where the image was taken from.

Data.json: A json data file that contains class properties and relations for all the instances in the image (See Data.json section for detailed description).

4. Instance segmentation general

The region masks for material phases and part instances are saved in "***MaterialsAndPart"*** subfolder. The masks of the vessel instances are saved in the ***"Vessels"*** subfolder. Each mask is saved in a separate .png image file. The name of the instance is the instance index. Information for this instance (classes/relations) can be found by looking for this index in the Data.json file. For example, the data for the instance saved in the file ***"\Vessels\2.png"*** can be found in the ***Data.json->"Vessels"->"2"*** entry. Similarly, data, for instance, stored in the image ***"\MaterialsAndParts\2.png"*** can be found in the ***Data.json->"MaterialsAndParts"->"2"*** entry.

For example use, see scripts in: **/ReaderScriptsPython/InstanceReader_.*.py**

For detailed file structure of the annotation mask and Data.json file, see sections 6,7

5. Semantic Segmentation General

SemanticMaps: subfolder contains the semantic maps that mark the region of the image corresponding to a specific class. This folder is divided into two subfolders: ***FullImage*** and ***PerVessel.***

SemanticMaps\FullImage: subfolder contains the semantic maps for the full image. For each class that appears in the image, there is a map of the region of the image belonging to this class saved as a png image, the name of the image is the name of the class. For example, the file "***SemanticMaps\FullImage\Liquid.png***" will contain the region of the image that contains liquids. Similarly, the file "***SemanticMaps\FullImage\Powder.png"*** will contain all the regions of the image containing Powders.

For use examples see in: ***/ReaderScriptsPython/SemanticMapReader_Full_Image*.py***

SemanticMaps\PerVessel\subfolder: contain subfolders the semantic maps for the content of each vessel instance. The names of the subfolders are the indexes of the specific vessels. For example ***SemanticMaps\PerVessel\3\*** will contain semantic maps for the content of vessel instance 3. For each class that appears in the specific vessel, there is a map of the region of the vessel belonging to this class saved as a png image, the name of the image is the name of the class. For example the file ***SemanticMaps\FullImagePerVessel\2\Foam*.png** will store the regions of vessel 2 that contain foam.

Similarly, the file ***SemanticMaps\FullImagePerVessel\5\Solid.png*** will store the regions of vessel 5 that contain Solids. Note that the file **Vessel.png** is just the region of the specific vessel.

For use examples see in:

/ReaderScriptsPython/SemanticMapReader_Full_ImageSemanticMapReader_PerVessel_*.py

More detail on the files structure see section 8

6. Data.json file

The Data.json file contains full categories and relations information for all instances.

The Data.json dictionary is divided into two main fields: "Vessels" contains data on Vessels instances, and "MaterialsAndParts" contains data on material and parts instances. For example, ***Data.json->"MaterialsAndParts"->"3"*** will contain data on Material/Part instance number 3, which corresponds to the mask saved in the file ***MaterialsAndParts/3.png***. In Addition:

Data.json->"Vessels"->"1" will contain data on Vessel instance number 1, which corresponds to the mask in file ***MaterialsAndParts\1.png.***

Materials/Parts Entries data structure

'Indx': Material instance index

'MaskFilePath': Path to the mask file containing the instance region.

'All_ClassNames': Names of all the instance classes (for example, Liquid, Suspension, Scattered)

'Proprties_ClassNames': Names of instance classes that correspond to properties like on the surface and scattered

'MaterialType_ClassNames': Name of instance classes that correspond to material types like liquid, solid powder…\

'PartsType_ClassNames': Name of instance classes that correspond to parts like cork, labels,

'IsPart': Is the instance part of the vessel (cork/valve/label..)?

'IsOnSurface': True/False, Does the material that covers the vessel surface but does not fill the entire volume.

'IsScattered': True/False, If the material is scattered and does not fill the entire volume.

'IsFullSegmentableMaterialPhase': True if the material is not part, not scattered, or on the surface (if the material fills the volume and is not a part).

'ASegmentableInstance': Is the instance have a distinct boundary and is segmentable, True for parts and materials that are not scattered.

'ContainMaterials_Indx': Indexes of the material phases that are immersed inside this instance. This entry doesn't exist for parts.

'InsideOfMaterials_Indx': Indexes of material phases contain this material phase. This field doesn't exist for parts.

'InsideOfVessel_Indx': Index of the vessel instance that contains this instance.

Vessel Entries Data Structure

'Indx': Vessel instance index.

'MaskFilePath': Path to the mask file containing the instance region.

'All_ClassNames': Names of all the instance classes (for example, Cup, Flask, transparent…)

'Proprties_ClassNames': Names of only classes that correspond to properties like transparent.

'VesselType_ClassNames': Names of only classes that correspond to vessel type (like cup...)

'IsPart': Is the vessel act as part for another vessel (like connector or condenser)

'IsTransparent': Is the vessel transparent.

'IsSemiTransparent': Is the vessel translucent.

'ContainVessels_Indx': Indexes of the vessel instances that the vessel contains.

'InsideOfVessels_Indx': Indexes of the vessel instances that the vessel is located inside of.

'LinkedToVessels_Indx': Indexes of the vessel's instance that the vessel is linked to.

'VesselContentAll_Indx': Indexes of all the materials and parts instances the vessel contains.

'VesselContentParts_Indx': Indexes of all the parts instances the vessel contains.

'VesselContentMaterial_Indx': Indexes of all the materials instances the vessel contains.

7. Instance png masks file structure

MaterialsAndParts folder: contains the regions/masks for all the material phases and part instances saved as .png image. Each material phase and the part region is saved in this folder as a png, image. The name of the file corresponds to the instance index. All the pixels/cells inside the instance region have values larger than zero in this mask, while all cells/pixels not part of the instance will have values of zero. The best practice is to use all the pixels in the mask with values larger than 0 as the instance region.

And more details...

The value of the cells belonging to the instance mask reflects the overlap between this instance and other instances of other material; the values of the cell in the png mask are as follows:

0: Not part of the material instance,

1: Cell is part of the material instance and does not overlap other material instances.

4: Cell is part of the material instance region that is overlapping another material instance and inside it.

5: Cell is part of the material instance region that is overlapping another material instance and contains it.

6: Cell is part of the material instance that is in front of another instance (this relation is partly annotated)

7: Cell is part of the material instance that is beyond other material instance (this relation is partly annotated)

For parts like cork and label, the only values are 1 (part of instance region) and 0 not in the instance regions.

The simplest practice is to use all the pixels in the mask with a value larger than 0 as the instance region.

Vessels folder Contain the regions/masks for all the Vessel instances in the image saved as a one channel .png image. The masks for each vessel instance is saved in this folder as a distinct png image. All the pixels/cells belonging inside the vessel instance region have values larger than zero., while the value of all cells/pixels not part of the instance will have values of zero. The name of the mask is the instance index.

Cells value in the instance masks are:

0: Not part of the vessel instance,

1: Cell is part of the vessel instance and does not overlap other vessel instances.

4: Cell is part of the vessel instance region that is overlapping another vessel instance and inside it.

5: Cell is part of the vessel instance region that is overlapping another vessel instance and contains it.

6: Cell is part of the vessel instance that is in front of other instance

7: Cell is part of the vessel instance that is beyond other vessel instance.

The simplest practice is to use all the pixels in the mask with a value larger than 0 as the instance region.

8. Semantic maps structure

Semantic maps are regions/pixels of the image that correspond to specific classes and are saved as png images. The semantic maps for the full image are saved in the SemanticMaps\FullImage subfolder. The semantic maps for the content of each vessel are saved in ***SemanticMaps\PerVessel\Vesselindx*** subfolder. For example, the file ***SemanticMaps\FullImage\Liquid.png*** cover all the region of the image that contains liquids, while the file ***SemanticMaps\PerVessel\3\Liquid.pn*****g** will cover all the region in vessel 3 that contain liquid.

Semantic maps .png maps detail structure.

The semantic map for each class is saved as an RGB image with 3 channels.

Blue channel: Pixels belonging to the class have values of 1, pixels not belonging to the class have a value of zero (For all practical, purposes this channel should be used as the semantic map).

Green channel: Pixels not belonging to the class have values of 0. Pixels belonging to the class have a value of 1. If the material is inside another material or vessel inside another vessel, the value is 4. If the material belonging to this class is in front of another material or vessel in front of another vessel, the value in this cell is 6.

Red channel: Contain properties. For Vessels and parts, this channel is identical to the blue channel. For materials, if the specific material in this region is scattered, the cell will have a value of 20. If the material in this region is stuck to the vessel surface (not filling volume), the cell will have a value of 30. Finally, if the material is both scattered and on the surface of the vessel, the cell will have a value of 40. If the cell does not contain the class, it will have a value of 0, and if it contains the class with no specific properties (fill volume), it will have a value of 1.

Dataset Annotation

Annotation of the dataset was performed by Mor Bismuth, Naveen Khatkar, Emily Robson.

Annotation was done using the VGG VIA annotator: