

Image Annotator

MANUALLY ANNOTATE IMAGES
TO AID ALGORITHM DEVELOPMENT
AND TRAIN AUTOMATIC OBJECT CLASSIFICATION

Image processing applications are becoming more and more complex: Images are larger than ever, objects of interest become harder to detect, and programmers need the expertise of doctors, professors, researchers, and engineers in a specific application domain ("domain experts") to be able to develop effective algorithms. Image Annotator software facilitates the communication between domain experts and programmers by concurrently displaying the same images and annotations on their computer screens, even if they are remotely located.

The Image Annotator software is a very efficient tool for quickly and accurately annotating images. The user simply outlines an object of interest using the mouse pointer and Image Annotator drawing tools, and then assigns that object to a class from a pull-down list of user defined classes. The object is then automatically added to an object database file. This file is an Aphelion™ ObjectSet, .ASO file.

Image Annotator can be used with any image type or size and for any image domain. Furthermore, its use does not require any programming skills.

Because objects identified in Image Annotator are entered into an ObjectSet, the annotations can serve as input to Aphelion™ Dev image analysis software. This means that all the capability and power of Aphelion™ Dev can be applied to the objects the user has drawn with Image Annotator. This includes the ability to compute a wide range of object measurements and to develop fully automated image processing solutions.

The Image Annotator ObjectSet output can also serve as input for other products of the Aphelion™ Imaging Software Suite. For example, the Classifier Builder software can use the ObjectSet as a training set to generate an automated classifier for the objects defined using Image Annotator. When Image Annotator is combined with Aphelion™ Dev, virtually any image-based application can be developed that includes image capture, object classification and statistical analysis.

As an option, the images annotated and the ObjectSets drawn for those images can be placed on a server machine, from which they can be concurrently accessed by domain experts and programmers located at different sites. This capability creates a virtual lab environment where remotely located project team members can work efficiently and effectively to accomplish project goals.

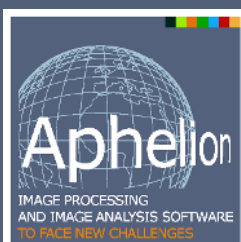
General Software Architecture

Objects of interest can be very different depending on the application. To quickly adapt to the diversity of possible applications, Image Annotator is fully user- configurable. All of its parameters (e.g., class names, color associated with each class) are defined in a user-edited XML file.

Input for Image Annotator is a set of images stored in one or more folders, arranged in a user-defined hierarchy appropriate for the application domain. The folder hierarchy is maintained by Image Annotator so the user can annotate images captured in different modalities. In the case of a medical application, the folder hierarchy might include different levels for: study number, site number, patient ID, visit number, and modalities (see screen capture following).

Image Annotator includes a set of drawing tools to identify objects by drawing a contour around them. If necessary, the user can define the number of control points on the contour to easily edit those objects. An optional comment can be added to the whole image (e.g. image quality).

Image Annotator outputs an object database comprised of Aphelion™ Serial ObjectSet files, which include any object measurements computed with Aphelion™ Dev. A user option also generates a binary image file for each ObjectSet, with the images appropriately saved in the folder hierarchy.



Main benefits of Image Annotator Software:

- Easily adapted for any application domain
- Highly configurable to define class names, colors, and associated drawing tool
- Data concurrently accessible by remote users (e.g., image sets, localizations, etc.)
- Facilitates interactions between programmers and domain experts
- Seamless compatibility with Aphelion Dev and its optional modules
- Complements Classifier Builder for creating training sets for automatic classifiers

Key Software Features

- A browser tree located in the left-side panel of the user interface provides the users with a global view of the folder hierarchy, which includes folders for each image to be annotated, and each of these includes the annotation files for its image. Annotation file names are color coded to identify its author (see screen capture following).
- One or two images can be displayed in the right-side panel of the user interface. Annotations are drawn in one image while the optional, second image is used as a reference to assess the evolution of a process or disease, or as an aid to the annotation.
- An annotation is drawn in the color assigned to its object class. Object class display buttons can be toggled to display any combination of classes. A displayed object class shows all objects in its associated ObjectSet.

- 2D monochrome or color images and movies are supported. For movies, annotations are drawn in a single, user-selected frame. For movies that are perfectly registered, an option enables the annotations to be copied to all frames of the movie.

For an input image, annotations are optionally saved as a set of binary images where each binary image contains the objects of a given ObjectSet (i.e., object class). These binary images are saved in the input image's folder, along with the annotation ASO files.

Using ASO files facilitates interoperability within the Aphelion™ Imaging Software Suite product family. An ASO file can be processed in Aphelion™ Dev to add individual object measurements (more than 50 measurements available). The ASO file can then be imported into Classifier Builder, another Aphelion™ Imaging Software Suite product, to perform automatic classification using Neural Network, Fuzzy Logic, or Multivariate classifiers.

