**Natural Language Processing for PDF/TIFF/Image Documents  
Computer Vision for Image Data  
VISION MODULE  
High Precision Image Processing   
Technical Specification, v1.5**

# 1 Images

## 1.1 Images Overview

The Images CV preprocessor contains the following primary classes, and their relationships:

* Images - This is the base class for the representation of a Computer Vision (CV) preprocessed list of images. The constructor optionally takes as parameters a list of images (paths), and corresponding labels, and flags for CV preprocessing the images.

images = Images([<list of images>], flags …)

* <Category>Image – This is a derived class for the representation of a category specific images, such as medical imaging data. It inherits from the Image base class.
* Image – This is the base class for the representation of a single Computer Vision (CV). The constructor optionally takes as parameters an image (path), corresponding label, and flags for CV preprocessing the image.

**Image**

**Images (Base Class)**

**Image**

**<Category>Images (Derived Class)**

**Image**

**List**

**Fig. 1a High Level view of Images Class Object Relationships**

## 1.2 Images Properties

The Images class contains the following properties:

* dir – Location to store the CV preprocessed image data.
* class – The classification for the list (batch) of images.

## 1.3 Images Overridden Operators

The following operators have their implementations (inherited from the base Object) class overridden:

* There are no overridden operators.

## 1.4 Images Private Methods

The Images class contains the following private methods:

* There are no private methods.

## 1.5 Images Public Methods

The Images class contains the following public methods:

* \_\_init\_\_() - The constructor performs the following:
  + If the images parameter is not None, then an Image object is created for each image in the images parameter.
  + Each image is processed according to settings in the config parameter:
    - grayscale (gray) – convert all RGB and RGBA images to Grayscale (single channel).
    - normalize (norm) – convert all pixel values to be between 0 and 1.
    - resize=(height, width) – change the pixel size of the image.
    - flatten (flat) – after conversions, flatten the pixel data into a 1D vector.
  + When images are being processed as color images (RGB), images with an alpha channel (RGBA) have the alpha channel dropped.
  + The processed image data is converted to a numpy matrix or vector (i.e., flatten)
  + When all the images are preprocessed, the processed images are stored as a batch, with corresponding labels. In a HD5 file.
  + Async - TBD
* load() – TBD
* batch) - TBD

# 2 Image

## 2.1 Image Overview

The Image CV preprocessor contains the following primary classes, and their relationships:

* Images- This is the base class for the representation of a Computer Vision (CV) preprocessed image. The constructor optionally takes as parameters an image (path), and corresponding label, and flags for CV preprocessing of the image.

image = Image([<list of images>], flags …)

## 2.2 Image Properties

The Image class contains the following properties:

* image – The path to the stored image.
* name – The root name of the image (e.g., /mydir/myimage.jpg -> myimage).
* type – The file format of the image (e.g., pdf).
* size – The byte size of the image.
* shape – The shape of the image (e.g., (height, width, channels))
* time - TBD
* dir – Location to store the CV preprocessed image data.
* class – The classification for the image.

## APPENDIX I: Updates

*v1.4*

# APPENDIX II: Anticipated Engineering

The following has been identified as enhancement/issues to be addressed in subsequent update:

1. X

**Proprietary and Confidential Information  
Copyright ©2018, Epipog, All Rights Reserved**