<http://docs.mitk.org/nightly/PAModulePage.html>

Photoacoustic Algorithms Module

**Table of Contents**

* [Overview](http://docs.mitk.org/nightly/PAModulePage.html#PAModulePageOverview)
* [The Photoacoustic Algorithms Module's Classes](http://docs.mitk.org/nightly/PAModulePage.html#PAModuleClasses)

**Overview**

The Photoacoustic Algorithms Module provides a **set of filters for beamforming and post-processing of photoacoustic and ultrasound data**. The main features are:

* Management of all filters through a single class PhotoacousticImage
* Beamforming of ultrasound and photoacoustic image data.
  + Beamforming using the DAS and DMAS Algorithms.
  + Optional real-time beamforming capabilities by the usage of OpenCL GPU computing
* Post/Pre-Processing of any types of images.
  + Crop Filter for removing artifacts in the upper and lower edge of the image.
  + Multiple B-Mode Filter implementations with resampling and logarithmic filter.
  + Bandpass Filter

To use the GPU capabilities of this module, OpenCL needs to be activated in CMAKE. The custom build option "camiPhotoacousticsWorkstation" activates all needed CMAKE options, as well as OpenCL. To build the project using OpenCL, the OpenCL libraries provided by your graphics card vendor need to be installed. The GPU capabilities of this module have only been tested using NVIDIA hardware, but, as OpenCL has been also implemented by AMD, this should work on either one. (Those are included in the CUDA package for NVIDIA graphics card owners.)

The [Photoacoustic Imageprocessing plugin](http://docs.mitk.org/nightly/org_mitk_gui_qt_photoacoustics_imageprocessing.html) provides a GUI to access all of the Photoacoustic Algorithms Module's filters.

**The Photoacoustic Algorithms Module's Classes**

* mitk::PhotoacousticImage: The class where all filters are managed.
* [mitk::BeamformingSettings](http://docs.mitk.org/nightly/classmitk_1_1BeamformingSettings.html): The class used by many filters for storing the configuration to be used when applying them.
* Five filters are currently implemented in the Photoacoustic Algorithms module:
  + [mitk::PhotoacousticBModeFilter](http://docs.mitk.org/nightly/classmitk_1_1PhotoacousticBModeFilter.html)/mitk::PhotoacousticOCLBModeFilter: Two classes for the B-Mode filter on GPU and CPU.
  + Resampling Filter: A resampling filter for post-processing.
  + [mitk::BeamformingFilter](http://docs.mitk.org/nightly/classmitk_1_1BeamformingFilter.html): A filter with a switch for GPU/CPU computation, to compute the beamformed image out of raw ultrasound/photoacoustic data.
  + Crop Filter: A filter for cropping artifacts on the upper and lower edges of the image.
  + Bandpass Filter: A Filter to filter image data in the Fourier domain, using a Tukey window to cut off low or high frequency parts of the image.