Slurry Image Analyzer Version 2

Image Analysis Software

Software Maintenance

|  |  |
| --- | --- |
| Original Author | Mark Polak |
| Company | Xanantec Technologies |
| Client | Syncrude Canada |
| Reviewers |  |
| Document Version | 1.0 |

# Purpose of Document

This concise document includes information relevant to long term maintenance of the Slurry Image Analyzer image processing software and is relevant for long-term software management. It will be useful for anyone responsible for managing and enhancing the software after initial development.

- SIA V2 Image Analysis Software Project Folders:

- src

(source code files)

- tests

(unit and system test files)

- docs

(documentation files)

On SIA V2 deployed machine, miniconda is installed under:

**C:\Users\Xanantec\Miniconda3**

***Saving Environment***

To generate “environment.yml” and “environment\_explicit.txt” files for an existing Conda environment, use the “conda env export” command.

Open Anaconda Prompt or a terminal, activate the environment you want to export, and run:

conda env export --name ImageAnalysisSIA --file environment.yml

The above command will create an environment.yml file in the current directory containing the environment specifications.

If you specifically want a “requirements.txt”-style file with explicit versions of each package, you can use the following:

conda list --name ImageAnalysisSIA --explicit > environment\_explicit.txt

After running either command, you can use the generated file to recreate the environment on another machine or share the environment specifications with others.

***Installing Environment***

To create a Conda environment on a new computer using the **environment.yml** file, you can use the **conda env create** command. Here are the steps:

1. **Transfer the environment.yml file:** Copy the **environment.yml** file from the original computer to the new computer.
2. **Open a Terminal or Anaconda Prompt:** On the new computer, open a terminal or Anaconda Prompt.
3. **Navigate to the directory containing the environment.yml file:** Use the **cd** command to navigate to the directory where the **environment.yml** file is located.
4. **Create the Conda environment:** Run the following command to create the Conda environment based on the specifications in the **environment.yml** file:

conda env create -f environment.yml

The above command will read the **environment.yml** file and create a new Conda environment with the specified packages and their versions.

To activate the Conda environment (after the environment is created) use the following command:

conda activate YourEnvironmentName

Contents of SIAExecSpript.bat:

@echo off

set CONDA\_PATH=C:\Users\Xanantec\miniconda3

set ENV\_NAME=ImageAnalysisSIA

rem Activate the Conda environment

call "%CONDA\_PATH%\Scripts\activate.bat" %ENV\_NAME%

rem Pass the command-line argument to the Python script

python ImageAnalysisSIAMain.py %1

rem Capture the return value of the Python script

set PYTHON\_EXIT\_CODE=%ERRORLEVEL%

rem Return the captured exit code to the calling routine

exit /b %PYTHON\_EXIT\_CODE%

SIAExecSpript.bat "c:\folder\"

echo The exit code is: %ERRORLEVEL%

• Overview: Provides a brief introduction to the SIA image processing software,

including its scope, and its role within the overall system.

• Dependencies: List all external dependencies, libraries, and third-party components

used by the software, along with their versions, and confirmation of licencing

provisions.

• Installation and Deployment: Provide step-by-step instructions for installing,

configuring, and deploying the software.

• Configuration Management: Explain any necessary steps to configure the software,

where configuration files are located, and how configuration changes are handled.

• Testing Procedures: Summarize testing methodologies, including unit testing and

system testing. This documentation enables the efficient retesting of the software

following any modifications or updates to ensure its continued proper functioning.

Includes information on how to run tests and interpret results.

• Version History: Document a version history detailing major and minor releases,

updates, and changes. Include release dates and a summary of changes for each

version.

• Known Issues: List any known issues or limitations of the software and provide

information on their status and workarounds.

• List of error codes and what they mean and how to remedy the error.

• Future Development Plans: Optionally, provide insight into future development

plans, possible additional features, or anticipated changes