

Classes

1. ConfigReader

- **Attributes:**
 - **config_path** (str): The path to the configuration file.
 - **config_parser** (ConfigParser): An instance of a configuration parser.
- **Methods:**
 - **__init__(self, config_path=None, configName=None)**: Constructor to initialize with an optional config path.
 - **getImageName()**: Returns the image name.
 - **getProjectedOverlayAlphaWidth()**: Returns the width of the projected overlay alpha.
 - **getProjectedImageWidth()**: Returns the projected image width.
 - **getTransparencyFactor()**: Returns the transparency factor as a float.
 - **saveConfig(path)**: Saves the configuration to the specified path.
 - **getImageSide()**: Gets the side of the image.
 - **setParameters(parameters)**: Sets parameters for the configuration.
 - **getGamma()**: Returns the gamma value.

2. ProcessImage

- **Methods:**
 - **processImage(image_path, params, main_display)**: Processes the image located at **image_path** with specified **params**, using **MainDisplay** instance for display purposes.

3. MainDisplay

- **Methods:**
 - **readImage(image_path)**: Reads an image from the specified path and returns it as an ndarray.
 - **setImage(image)**: Sets the image to display as an ndarray.

4. MaskCreator

- **Attributes:**
 - **_image** (ndarray): Stores the original image.
 - **_alpha_gradient** (ndarray): Stores the alpha gradient.
 - **_gamma_corrected** (ndarray): Holds the gamma-corrected version of the image.
 - **_mask** (int): Represents the mask value.
 - **transparency_factor** (float): Factor controlling the transparency.
 - **result_image** (ndarray): Holds the result after processing.
- **Methods:**
 - **__init__(_image, transparency_factor)**: Initializes with an image and transparency factor.
 - **createMask(image_side, mask_width, image_width)**: Creates a mask for the specified image side and dimensions.
 - **smoothStep(edge0, edge1, x)**: Returns a smooth transition between two edges.
 - **gammaCorrection(gamma)**: Applies gamma correction.
 - **alphaBlending(image_side)**: Performs alpha blending for the specified side.

5. ImageProcessingApp

○ Attributes:

- **left_image_path** and **right_image_path** (str): Paths to the left and right images.
- **left_params** and **right_params** (dict): Parameters for left and right images.
- **processed_images** (dict): Dictionary to store processed images.
- **notebook** (Notebook), **tab_left**, **tab_right**, **tab_settings**, **tab_preview** (Frame): UI components.
- **process_button** (Button), **progress_bar** (Progressbar): UI elements for processing and progress indication.
- **left_image_label**, **right_image_label**, **original_image_label**, **processed_image_label** (Label): Labels for displaying images.

○ Methods:

- **__init__(root)**: Constructor to initialize the application with a root UI element.
- **setupLeftTab()**, **setupRightTab()**, **setupSettingsTab()**, **setupPreviewTab()**: Methods for setting up respective tabs in the UI.
- **selectLeftImage()**: Allows selection of the left image.
- **saveConfiguration()**, **loadConfiguration()**: Methods to save and load configuration.
- **processAndSave()**: Processes and saves the image.
- **displayPreview()**: Displays a preview of the processed images.
- **displayImage(processedImages)**: Displays an image from processed images.

Relationships

- **ImageProcessingApp** is the main application class and interacts with **ConfigReader**, **ProcessImage**, and **MainDisplay** classes.
- **ConfigReader** provides configuration details to **ImageProcessingApp**.
- **ProcessImage** uses **MainDisplay** to load and display images, possibly utilizing **MaskCreator** for mask and blending operations.
- **MaskCreator** handles the creation of masks and alpha blending, which may be used by **ProcessImage** to prepare images for display in **MainDisplay**.