

Component Specification

1. Software Components

1.1 Feature Extractor

Description: Extracts features from agglutination images using ResNet18.

Inputs:

Image tensor (torch.Tensor)

- Shape: [batch_size, 3, 256, 256]
- Values: Normalized RGB pixels (0-1)
- Format: PyTorch tensor

Outputs:

Feature vector (numpy.ndarray)

- Shape: [batch_size, 512]
- Values: Floating point feature values
- Format: NumPy array

1.2 Concentration Predictor

Description: Predicts concentration values from extracted features using Random Forest.

Inputs:

Feature vector (numpy.ndarray)

- Shape: [batch_size, 512]
- Format: NumPy array

Trained model state

- Model weights
- Scaler parameters

Outputs:

Concentration predictions (numpy.ndarray)

- Shape: [batch_size]
- Values: Scientific notation (e.g., 1e-8)
- Units: Molar concentration (M)

1.3 Dataset Manager

Description: Handles data loading, preprocessing, and augmentation.

Inputs:

- Image directory path (str)
- File naming pattern: "[concentration]_[identifier].[extension]"
Example: "10-8_crop0.jpg"
- Image files

- Formats: JPG, PNG
- Size: Variable (resized to 256x256)

Outputs:

- DataLoader object
 - Batched image tensors
 - Corresponding concentration labels
- Dataset statistics:
 - Unique concentrations
 - Sample counts per class

1.4 GUI Manager

Description: Provides user interface for image analysis and result visualization.

Inputs:

- User uploaded images
- Trained models from 'trained_models' directory

Outputs:

- Displayed images
- Prediction displays
- Saved results in structured format

2. Component Interactions

Interactions to accomplish use cases:

Use case: Predicting concentration from agglutination image

- In this use case, the GUI Manager first handles user interaction by allowing image upload through a simple interface. When an image is uploaded, the Dataset Manager takes over to ensure proper image processing. It standardizes the image format and size, converting it to a 256x256 pixel format and applying necessary normalization to ensure consistent input quality.
- The Feature Extractor component then processes this standardized image using a pre-trained ResNet18 model, extracting a 512-dimensional feature vector that captures the essential characteristics of the agglutination pattern. These features represent various aspects of the image such as particle distribution and clumping patterns.
- The Concentration Predictor receives these features and processes them through a trained Random Forest model. This model has been pre-trained on numerous agglutination patterns with known concentrations, allowing it to accurately predict the concentration level from the extracted features. The predictor outputs a concentration value in scientific notation (e.g., 1e-8 M).
- Finally, the GUI Manager displays the results to the user, showing both the original image and the predicted concentration. It also provides the option to save these results, storing both the image and the predicted concentration value along with a timestamp for future reference.

- Throughout this process, error checking is performed at each step. The Dataset Manager verifies image format and quality, the Feature Extractor ensures proper feature extraction, and the Concentration Predictor validates its predictions within the expected range of $1e-8$ to $1e-1$ M.

3. Preliminary Plan

Phase 1: Core Development (2 weeks)

Week 1 (Current Sprint)

- ☒ Set up project structure
- ☒ Implement Dataset Manager
- ☒ Integrate ResNet Feature Extractor
- ☐ Develop Random Forest Predictor
- ☐ Basic GUI implementation

Week 2

- ☐ Complete GUI development
- ☐ Integration testing
- ☐ Model validation
- ☐ Error handling
- ☐ Basic documentation

Phase 2: Enhancement (2 weeks)

Week 3

- Model optimization
- Advanced error handling
- User feedback implementation
- Extended validation

Week 4

- Performance optimization
- Final testing
- Documentation completion
- Package preparation

Phase 3: Final Delivery (1 week)

- System integration testing
- User acceptance testing
- Documentation finalization
- Deployment preparation