1. Objectives

- Resolve discrepancies between original and fitted image dimensions that affect cropping functionality.
- Align cropping operations with the visible portion of the image within the window to ensure user-intended results.
- Address remaining issues with undo functionality, viewport mapping, and window size dynamics.

---

2. Activities

- **Analysis of Cropping Issues:**
  - Identified that cropping operations used original image data, causing mismatches with user selections on the fitted image.
  - Investigated misalignment between selection rectangles and dynamically visible areas due to viewport and zoom transformations.
- **Attempted Fixes for Cropping Discrepancies:**
  - Updated cropToSelection to operate on fitted view (m_imgData):
    - Adjusted coordinate calculations to align with the window-fitted image.
    - Enhanced processCurrentImage and updateImageFromProcessed to update fitted view data after each operation.
  - Enhanced undo functionality:
    - Stored and restored both fitted and processed image data using deep copies.
  - Updated the status bar to display:
    - Original image size.
    - Interlaced size (if processed).
    - Merged size (processed result dimensions).

        - Fitted size (current window-fitted dimensions).
    - Added getViewportSize in GraphicsView to calculate fitted dimensions dynamically for the window.
    - **Remaining Issues:**
        - Precision of coordinate mapping for cropping still requires refinement.
        - Deep copying in undo history for large images may introduce inefficiencies.
        - Fitted size calculations may not account for dynamic window resizing.
- **Restoration of Previous Cropping Logic:**
    - Reverted cropping operations to use original image dimensions for reference.
    - Removed changes related to fitted size cropping and restored previous GraphicsView behavior.
- **Adjustments for Cropping Alignment with Visible Area:**
    - Modified cropToSelection to:
        - Use zoom factor and visible window area for cropping dimensions.
        - Map window coordinates to image space using mapToScene.
        - Clamp crop dimensions within image bounds.
    - Investigated edge cases, such as extreme zoom levels and small images, for potential inconsistencies.
    - **Remaining Issues:**
        - Cropping may fail if the view matrix is not updated after zooming or panning.
        - Dynamic updates of the visible area during window resizing may lead to inaccuracies.
        - Edge cases still produce unexpected behavior.
    - **Restoration:**

- Reverted to the previous implementation of cropToSelection with static selection rectangles.
- Removed dependencies on window size and viewport calculations.

---

3. Achievements

- Identified and documented key issues with cropping functionality in relation to fitted and visible image dimensions.
- Improved understanding of viewport transformations and dynamic updates for future refinement.
- Enhanced the status bar to provide detailed information about image dimensions in various states.
- Restored stable functionality by reverting to previous implementations for cropping and viewport calculations.

---

4. Problems & Solutions

1. **Problem:** Cropping based on original dimensions caused mismatches with fitted images in the window.
    - **Solution:** Attempted to use fitted view data for cropping but reverted to original dimensions for stability due to precision and dynamic update issues.
2. **Problem:** Cropping alignment with visible window portions was inconsistent.
    - **Solution:** Updated cropping logic to use viewport transformations but reverted changes due to edge case failures and dynamic inconsistencies.
3. **Problem:** Undo functionality introduced inefficiencies with large images.
    - **Solution:** Deep copying was implemented but highlighted for optimization in future updates.
4. **Problem:** Status bar lacked visibility into fitted dimensions.

- **Solution:** Added dynamic updates to display original, interlaced, merged, and fitted dimensions for better debugging.