**Objective**

Automated image stitching involves integrating sequential images to form seamless panoramas. This research focuses on developing a robust stitching system that employs a **sliding window approach** to process large datasets efficiently. The objectives are:

1. **System Development**: Implement an automated pipeline for sequential image stitching with scalability for large datasets. The system is designed with **modular layers**, including pre-processing, feature detection, stitching, and error recovery.
2. **Technical Goals**:
   * Employ advanced algorithms like **SIFT** (Scale-Invariant Feature Transform) for feature detection to ensure robustness against scale, rotation, and illumination changes.
   * Use **CLAHE** (Contrast Limited Adaptive Histogram Equalization) to enhance image quality, particularly in non-uniform lighting.
   * Integrate fallback strategies for error recovery to handle image mismatches or suboptimal stitching results.
3. **Performance Metrics**:
   * Maximize the stitching success rate through adaptive processing.
   * Optimize resource utilization using a sliding window approach, reducing computation overhead for large sequences.
   * Maintain high-quality panoramas by combining advanced blending techniques with systematic quality validation.
4. **Applications**:
   * Real-time video frame stitching for surveillance and environmental monitoring.
   * Automated panoramic creation for photography and virtual tours.
   * Efficient visual data compilation in medical imaging, astronomy, and more.

The research leverages computational techniques, adaptive algorithms, and a modular architecture to ensure high performance in automated stitching, even with challenging datasets.

**Block Diagram**

The stitching system is divided into four key modules, each contributing to the processing pipeline:

**1. Input Processing Layer**

* **Image Validation Module**:
  + Verifies image formats and sequences.
  + Assesses image quality to exclude corrupt data.
* **Batch Formation Module**:
  + Groups images into overlapping batches using the sliding window approach.
  + Ensures smooth transitions by maintaining overlap.

**2. Processing Core**

* **Pre-Processing Module**:
  + Standardizes image resolution (e.g., 1280x720).
  + Enhances visibility using **CLAHE**.
  + Optimizes color spaces for better feature detection.
* **Feature Detection Module**:
  + Extracts keypoints using **SIFT**.
  + Matches descriptors using brute-force or KNN algorithms.
* **Stitching Engine**:
  + Computes transformations via homography estimation.
  + Warps and blends images seamlessly.

**3. Error Handling Layer**

* **Recovery Module**:
  + Adjusts feature matching thresholds.
  + Employs alternative methods for difficult datasets.
  + Processes subsets when batch stitching fails.
* **Quality Control**:
  + Validates output for visible seams or artifacts.
  + Logs errors and reattempts stitching if necessary.

**4. Output Organization**

* **Storage Management**:
  + Organizes results into folders by batch and sequence.
* **Validation**:
  + Conducts systematic checks on panoramas to ensure quality.
  + Generates a report on performance and accuracy.

**Algorithm**

The system implements the following algorithmic steps:

**1. Pre-Processing Algorithm**

* Input: Raw image batch.
* Process:
  1. **Resolution Standardization**: Resize images to a fixed resolution to ensure consistency.
  2. **CLAHE Enhancement**:
     + Convert images to grayscale.
     + Apply CLAHE for localized contrast enhancement, ensuring no over-saturation.
  3. **Color Space Optimization**: Enhance features using LAB color space.

**2. Feature Detection**

* Input: Enhanced images.
* Process:
  1. Use **SIFT** to detect robust keypoints and compute 128-dimensional descriptors.
  2. Match descriptors between images using Lowe’s ratio test to filter out weak matches.

**3. Image Stitching**

* Input: Matched keypoints.
* Process:
  1. Estimate homography using **RANSAC** to eliminate outliers.
  2. Warp images based on the computed transformation.
  3. Blend overlaps using multi-band blending or feathering.

**4. Error Recovery**

* Input: Failed batch.
* Process:
  1. Adjust SIFT thresholds or RANSAC parameters.
  2. Attempt alternative detectors like ORB if SIFT fails.
  3. Process subsets of the batch to recover partial panoramas.

**References**

1. **SIFT**: Lowe, D.G. (2004). "Distinctive Image Features from Scale-Invariant Keypoints". International Journal of Computer Vision, 60(2), 91-110.
   * Key contribution: Introduced scale and rotation-invariant keypoints, forming the basis for robust feature matching.
2. **CLAHE**: Zuiderveld, K. (1994). "Contrast Limited Adaptive Histogram Equalization". Graphics Gems IV, 474-485.
   * Key contribution: Improved localized contrast enhancement without over-saturation, ideal for stitching low-contrast images.
3. **Homography and RANSAC**: Fischler, M.A., & Bolles, R.C. (1981). "Random Sample Consensus: A Paradigm for Model Fitting". Communications of the ACM, 24(6), 381-395.
   * Key contribution: Robustly estimates transformations, tolerating outliers.
4. **Automatic Stitching**: Brown, M., & Lowe, D.G. (2007). "Automatic Panoramic Image Stitching using Invariant Features". International Journal of Computer Vision, 74(1), 59-73.
   * Key contribution: Integrated SIFT-based matching into panoramic stitching workflows.
5. **Camera Calibration**: Zhang, Z. (2000). "A Flexible New Technique for Camera Calibration". IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(11), 1330-1334.
   * Key contribution: Foundation for accurate perspective correction during stitching.