**Lanczos Interpolation Test Cases**

**Test Case L1: Basic Upscaling Verification**

**Description:** Verify that Lanczos interpolation correctly upscales an image by a factor of 2.

**Input:**

* A 100x100 pixel test image with clear, defined edges and patterns
* Scale factor: 2

**Expected Output:**

* A 200x200 pixel image that maintains edge sharpness and minimizes artifacts
* Visual inspection should show smooth edges with minimal ringing artifacts
* Image details should be preserved better than with bicubic or bilinear interpolation

**Test Steps:**

1. Load the 100x100 test image
2. Apply Lanczos interpolation with scale factor 2 using cv2.resize(image, (image.shape[1]\*2, image.shape[0]\*2), interpolation=cv2.INTER\_LANCZOS4)
3. Verify the output dimensions are exactly 200x200 pixels
4. Compare with other interpolation methods (nearest neighbor, bilinear, bicubic) to confirm superior edge preservation
5. Check for aliasing artifacts, particularly around high-contrast edges

**Test Case L2: Preserving Fine Details**

**Description:** Test Lanczos algorithm's ability to preserve fine details like text and thin lines.

**Input:**

* An image containing small text and fine line details
* Scale factor: 3

**Expected Output:**

* Upscaled image with readable text and well-defined lines
* Text edges should remain sharp and not blurry
* Fine lines should maintain consistent thickness and not appear broken

**Test Steps:**

1. Load the test image with text and fine lines
2. Upscale by factor of 3 using Lanczos interpolation
3. Examine the text readability in the output image
4. Measure the consistency of line thickness before and after scaling
5. Calculate PSNR (Peak Signal-to-Noise Ratio) between the original and a downscaled version of the result

**Test Case L3: Handling Color Gradients**

**Description:** Assess how Lanczos interpolation handles smooth color transitions and gradients.

**Input:**

* An image with smooth color gradients (like a sunset or rainbow gradient)
* Scale factor: 4

**Expected Output:**

* Upscaled image with smooth transitions between colors
* No visible banding or posterization in the gradient areas
* Preservation of subtle color variations

**Test Steps:**

1. Load the gradient test image
2. Apply Lanczos upscaling with factor 4
3. Analyze histogram of color channels to verify smooth distribution
4. Perform visual inspection for color banding
5. Calculate color variance in gradient regions before and after upscaling

**Test Case L4: Performance with Noise**

**Description:** Test how Lanczos interpolation performs with noisy images.

**Input:**

* An image with various levels of noise (Gaussian, salt & pepper, etc.)
* Scale factor: 2

**Expected Output:**

* Upscaled image where noise is not significantly amplified
* Overall image clarity despite the presence of noise
* No additional artifacts created by the interpolation process

**Test Steps:**

1. Create test images with different noise types and intensities
2. Apply Lanczos upscaling to each image
3. Compare noise levels before and after scaling (signal-to-noise ratio)
4. Verify that the algorithm doesn't create additional artifacts from the noise
5. Compare with other interpolation methods' handling of noise

**Test Case L5: Edge Cases - Extreme Upscaling**

**Description:** Test the limits of Lanczos interpolation with high scaling factors.

**Input:**

* A small 50x50 pixel image with distinct features
* Scale factor: 8 (resulting in 400x400 output)

**Expected Output:**

* Upscaled image with reasonable quality despite extreme scaling
* Some inevitable detail fabrication, but minimal ringing artifacts
* Preservation of the overall structure and major features

**Test Steps:**

1. Load the small test image
2. Apply Lanczos upscaling with factor 8
3. Verify output dimensions
4. Assess visual quality and artifact presence
5. Compare with other interpolation methods at the same scaling factor

**CLAHE (Contrast Limited Adaptive Histogram Equalization) Test Cases**

**Test Case C1: Basic Contrast Enhancement**

**Description:** Verify that CLAHE properly enhances contrast in an image with poor contrast.

**Input:**

* Low-contrast test image (e.g., foggy or hazy scene)
* clipLimit: 3.0, tileGridSize: (8,8)

**Expected Output:**

* Image with improved contrast and visibility
* Histogram should show wider distribution across the intensity range
* Local details should be more visible without excessive noise amplification

**Test Steps:**

1. Load the low-contrast test image
2. Convert to LAB color space
3. Apply CLAHE to L-channel with specified parameters
4. Reconvert to BGR color space
5. Compare histograms of original and enhanced images
6. Calculate contrast improvement metrics (e.g., standard deviation of pixel intensities)

**Test Case C2: Prevention of Over-enhancement**

**Description:** Test that CLAHE's clip limit effectively prevents over-enhancement and noise amplification.

**Input:**

* Image with both low-contrast and noisy regions
* Test with varying clipLimit values: 1.0, 3.0, and 5.0

**Expected Output:**

* With clipLimit=1.0: Modest contrast enhancement with minimal noise amplification
* With clipLimit=3.0: Better contrast enhancement with controlled noise
* With clipLimit=5.0: Higher contrast but potentially more visible noise

**Test Steps:**

1. Load the test image
2. Apply CLAHE with each clipLimit value while keeping tileGridSize constant at (8,8)
3. Analyze noise levels in dark regions for each result
4. Compare histogram clipping behavior across different settings
5. Calculate signal-to-noise ratio for each result

**Test Case C3: Tile Size Impact**

**Description:** Test how different tile grid sizes affect CLAHE performance.

**Input:**

* Image with varying lighting conditions across different regions
* clipLimit: 3.0
* Test with tileGridSize values: (4,4), (8,8), and (16,16)

**Expected Output:**

* tileGridSize=(4,4): More localized adaptation but potential tile boundary visibility
* tileGridSize=(8,8): Balanced local adaptation and smooth transitions
* tileGridSize=(16,16): Smoother global enhancement but less adaptation to local features

**Test Steps:**

1. Load the test image with varying lighting conditions
2. Apply CLAHE with each tileGridSize while keeping clipLimit constant at 3.0
3. Visually inspect boundary transitions between tiles
4. Measure local contrast improvement in different regions
5. Calculate histogram equalization effectiveness across different regions

**Test Case C4: Performance with Different Image Types**

**Description:** Test CLAHE on different types of images to verify versatility.

**Input:**

* Portrait photo (skin tones)
* Landscape (wide dynamic range)
* Indoor scene (mixed lighting)
* Document scan (text clarity)
* clipLimit: 3.0, tileGridSize: (8,8)

**Expected Output:**

* Portrait: Enhanced features without unnatural skin tones
* Landscape: Improved visibility in both shadows and highlights
* Indoor: Better visibility in dark areas without overexposing bright regions
* Document: Improved text readability without introducing artifacts

**Test Steps:**

1. Load each test image type
2. Apply CLAHE with standard parameters
3. Evaluate appropriate metrics for each image type (e.g., skin tone consistency, shadow detail, text clarity)
4. Conduct user perception tests for aesthetic quality
5. Compare with global histogram equalization to verify advantages

**Test Case C5: Color Preservation**

**Description:** Verify that CLAHE preserves color relationships while enhancing contrast.

**Input:**

* Color-critical image (e.g., color chart, artistic photograph)
* clipLimit: 3.0, tileGridSize: (8,8)

**Expected Output:**

* Enhanced contrast without significant color shifts or hue changes
* Maintained color relationships between image elements
* No introduction of false colors or color artifacts

**Test Steps:**

1. Load the color-critical test image
2. Apply CLAHE enhancement
3. Calculate color differences using Delta-E in original vs. enhanced image
4. Check color histogram distributions for each channel
5. Verify saturation levels aren't inappropriately modified in the LAB color space conversion

**Test Case C6: Processing Edge Cases**

**Description:** Test CLAHE on challenging edge cases.

**Input:**

* Nearly uniform image (e.g., blank wall, clear sky)
* Extremely dark image
* Extremely bright image
* clipLimit: 3.0, tileGridSize: (8,8)

**Expected Output:**

* Uniform image: Minimal change without introducing artificial contrast
* Dark image: Improved visibility without excessive noise amplification
* Bright image: Increased detail visibility in highlight regions without unnatural appearance

**Test Steps:**

1. Load each edge case test image
2. Apply CLAHE with standard parameters
3. Analyze histogram before and after for appropriate transformation
4. Check for introduction of artifacts or noise
5. Verify enhancement effectiveness using objective metrics appropriate to each case

**Integration Test Cases**

**Test Case I1: Combined Lanczos and CLAHE Pipeline**

**Description:** Test the combined effect of Lanczos upscaling followed by CLAHE enhancement.

**Input:**

* Low-resolution, low-contrast image
* Scale factor: 2
* CLAHE parameters: clipLimit=3.0, tileGridSize=(8,8)

**Expected Output:**

* Higher resolution image with improved contrast
* Preserved details from Lanczos without enhancement artifacts from CLAHE
* Natural-looking result without over-processing appearance

**Test Steps:**

1. Load the test image
2. Apply Lanczos upscaling
3. Apply CLAHE enhancement to the upscaled image
4. Evaluate the quality compared to applying each algorithm independently
5. Check for compounding artifacts or issues
6. Calculate overall improvement metrics (resolution, contrast, and detail preservation)