

Lanczos Resizing and Rotation for Image Transformation

Kunovics Dávid Zoltán

December 15, 2024

1 Introduction

Image resizing and rotation are fundamental operations in image processing, commonly used in computer vision and graphics. Achieving high-quality visual results while performing these transformations simultaneously poses a significant challenge. This document describes a method implemented in MATLAB, combining Lanczos resampling for resizing and rotation using a 2D transformation matrix.

2 Mathematical Model

2.1 Lanczos Resampling

Lanczos resampling is a high-quality interpolation method based on the sinc function, defined as:

$$\text{sinc}(x) = \begin{cases} 1 & \text{if } x = 0 \\ \frac{\sin(\pi x)}{\pi x} & \text{if } x \neq 0 \end{cases}$$

The Lanczos kernel is constructed by multiplying two sinc functions:

$$w(x) = \text{sinc}(x) \cdot \text{sinc}\left(\frac{x}{A}\right)$$

where A is the kernel size parameter, controlling the range of influence of each pixel. Larger values of A yield smoother results but increase computational complexity. The kernel is truncated to A to make the computation practical:

$$w(x) = 0 \quad \text{for } |x| \geq A$$

2.2 Rotation Transformation

To rotate an image by an angle θ , we use the 2D rotation matrix:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} x - c_x \\ y - c_y \end{bmatrix} + \begin{bmatrix} c_x \\ c_y \end{bmatrix}$$

where (c_x, c_y) is the image center. The inverse rotation maps the rotated coordinates (x', y') to the original image, enabling interpolation.

3 Methodology

The implementation integrates resizing and rotation in a single sampling loop using Lanczos resampling. The steps are as follows:

1. **Scaling:** The target dimensions are calculated based on the scaling factor.
2. **Inverse Mapping:** For each pixel in the output image, its coordinates are mapped back to the input image using the inverse rotation and scaling transformations.
3. **Lanczos Interpolation:** The surrounding pixels are weighted using the Lanczos kernel, summing their contributions to compute the final pixel value.

The kernel weights depend on the distance between the input and output coordinates, ensuring high-quality interpolation.

4 Test Cases and Results

The function was tested with the following configurations:

1. **Test Case 1:** Rescale by a factor of 4 with no rotation (0 degrees).
2. **Test Case 2:** Rescale by a factor of 2 and rotate by 45 degrees.
3. **Test Case 3:** Downscale by a factor of 0.5 and rotate by 90 degrees.



Figure 1: Test Case 1: Rescaled by a factor of 4 with no rotation.



Figure 2: Test Case 2: Rescaled by a factor of 2 and rotated by 45 degrees.

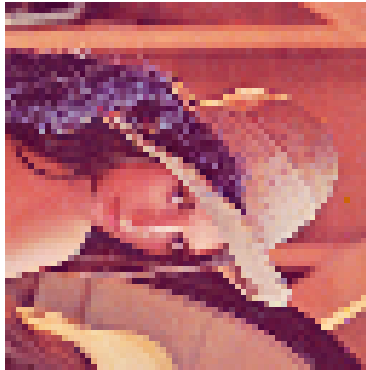


Figure 3: Test Case 3: Downsampled by a factor of 0.5 and rotated by 90 degrees.

5 Conclusion

The described MATLAB implementation demonstrates a robust approach for simultaneous resizing and rotation. By combining Lanczos resampling with inverse mapping, the method ensures high-quality transformations, preserving image details. The test cases highlight its versatility across various scales and angles.