Image Processing

HW4

Lian Natour 207300443

Mohammad Mhneha 315649814

Problem 1:
a)using geometric operations with interpolation versus modifying the Fourier transform and applying an inverse Fourier transform,the best choice often depends on the requirements of the specific application.
Geometric Operations with Interpolation: This method involves directly resizing the image by manipulating pixel values and coordinates. Interpolation techniques (like nearest neighbor, bilinear) are used to estimate pixel values in the scaled image. This approach is typically faster and simpler, especially for applications that require real-time processing. It is also more intuitive and straightforward in terms of implementation.

Fourier Transform Method: This method involves taking the Fourier transform of the image, modifying the frequency components, and then applying an inverse Fourier transform to scale the image. This approach can be advantageous because it allows for operations in the frequency domain, such as filtering or modifying specific frequency components before transforming it back. This method might be preferable when precision or specific frequency manipulation is necessary.

Comparison and Suitability:

- **Efficiency**: Geometric operations with interpolation are generally more efficient in terms of computational resources and time, making them suitable for applications that require quick processing, like interactive applications or real-time systems.
- **Simplicity**: The interpolation method is more straightforward to implement and understand. This simplicity makes it the default choice in many graphics software tools and libraries.
- **Quality and Artifacts**:
    - **Interpolation**: Can introduce artifacts such as blurring, especially with more substantial scaling or with simpler interpolation methods like nearest neighbor.
    - **Fourier Transform**: Can maintain better consistency in handling image frequencies but might introduce artifacts due to edge effects and ringing caused by the manipulation of frequency components.
- **Flexibility**: Fourier transform provides greater flexibility in manipulating the image in the frequency domain, which can be advantageous in specialized applications where such manipulations are necessary.
- **Conclusion:**
  For general image scaling, geometric operations with interpolation are typically preferred for their speed and simplicity, making them ideal for real-time applications. The Fourier transform method is better suited for specialized tasks requiring precise frequency manipulation, due to its complexity and higher computational cost.

b)
To prove mathematically that if F(u,v)=G(u,v), then f(x,y)=g(x,y) , we can use the inverse Fourier transform property:

The inverse Fourier transform of F(u,v) is:

$$f(x,y) = \iint \mathrm{F}(\mathrm{u},\mathrm{v}) e^{i2\pi(ux+vy)}\, du\, dv$$

Similarly, for G(u,v):

$$g(x,y) = \iint \mathrm{G}(\mathrm{u},\mathrm{v}) e^{i2\pi(ux+vy)}\, du\, dv$$

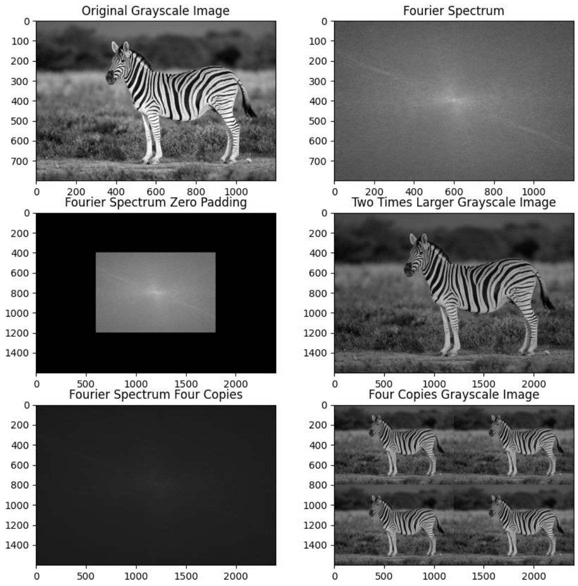
Since F(u,v)=G(u,v), substituting this into the inverse transform gives:

$$f(x,y) = \iint \mathrm{G}(\mathrm{u},\mathrm{v}) e^{i2\pi(ux+vy)}\, du\, dv = g(x,y)$$

This proves that f(x,y)=g(x,y) when their Fourier transforms are equal.

Problem 2:

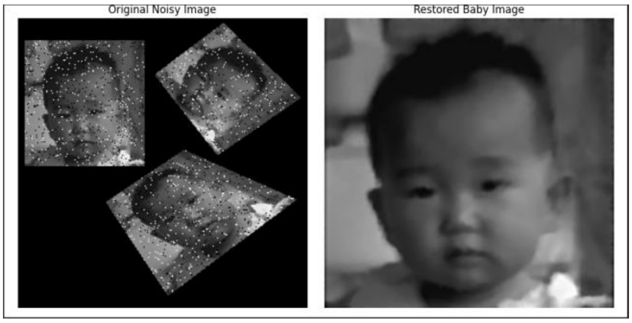
Here is the image that we got :



- B) Zero Padding in the Fourier Domain for Scaling: This approach enlarges the Fourier transform by symmetrically adding zeros around it. This expansion increases the size of the image when transformed back to the spatial domain, creating a larger but identical version of the original image.
- C)Creating Four Copies via Fourier Coefficient Replication: This method spaces out the original Fourier coefficients with zeros, reducing the density of frequency components. When transformed back to the spatial domain, this results in an image with four smaller copies of the original, each in a quadrant of the new image frame.

Problem 3:

baby:

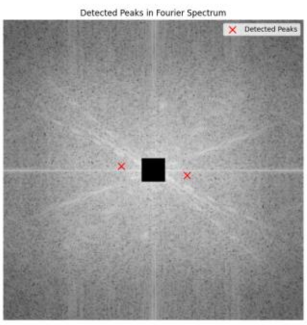


The noisy baby image was restored by first applying median blur to remove salt-and-pepper noise. Then, the three baby faces were cut out and straightened using a perspective transformation. Each face was cleaned again, and finally, all three were blended to create a clear and smooth image.

Windmill:

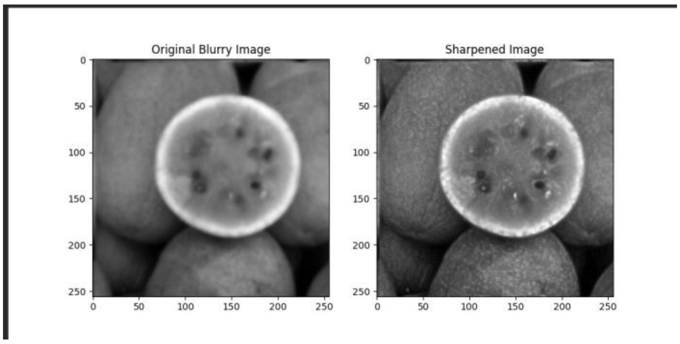
The windmill image was denoised by applying a Fourier transform to shift the image into the frequency domain, where periodic noise appeared as distinct peaks. These noise peaks at coordinates (124, 100) and (132, 156) were set to zero to eliminate the unwanted frequencies. An inverse Fourier transform was then performed to reconstruct the image, resulting in a cleaner version with reduced noise.

Here you can see the peaks that we found :



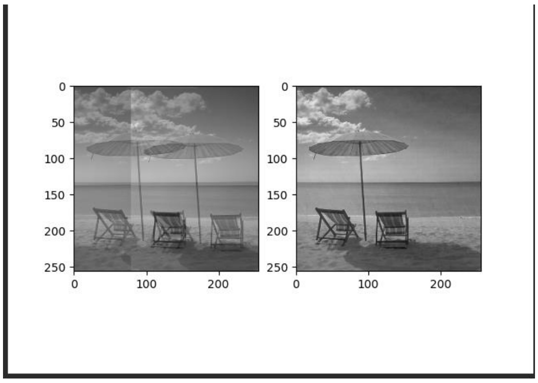

```
C:\Users\liann\Desktop\uni\Sem6_Winter\ImageProcessing\Hws\Hw2\.venv\Scripts
Detected peak coordinates (y, x): [[124 100]
 [132 156]]

Process finished with exit code 0
```

Watermelon:



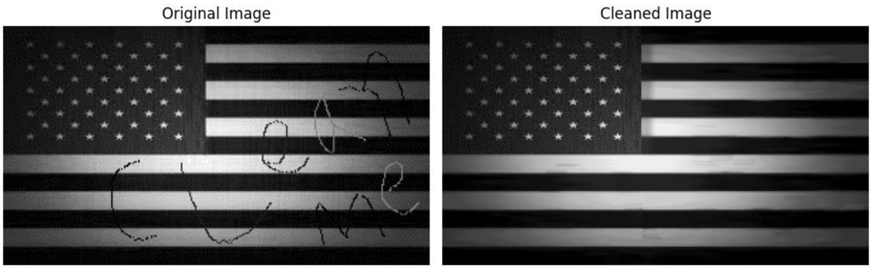
To fix the blurry watermelon image, I applied a sharpening filter using a convolution with a sharpening kernel. The filtered image was then added to the original image using cv2.add() to enhance edges and details. This process effectively emphasized the image's features, resulting in a clearer and sharper image.

Umbrella:



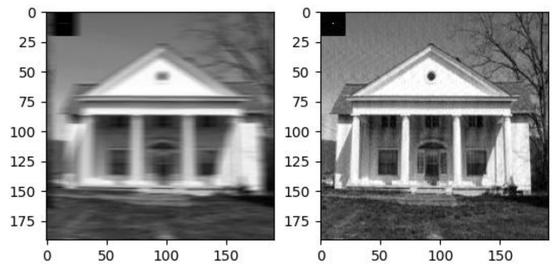
The ghosted umbrella image was fixed by applying a frequency domain filter. We identified the noise frequency and constructed a correction filter to suppress it. The filtered image was then transformed back to the spatial domain, effectively removing the ghosting artifacts.

USAFlag- Clean me:



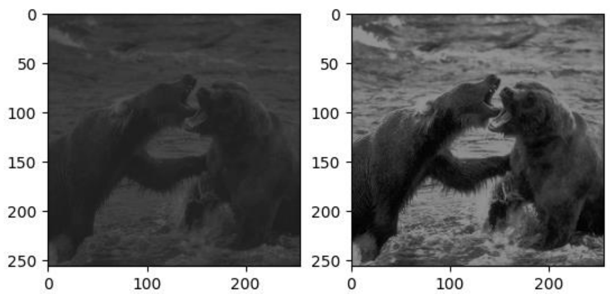
The noisy USA Flag image was cleaned by separating the noisy regions (right side and bottom-left) and applying a combination of median blur and uniform filtering to smooth out the noise. The stars region was preserved to retain important details. The cleaned regions were combined back with the original image to restore the flag.

House:



The motion blur in the house image was corrected using inverse filtering in the frequency domain. A horizontal motion blur kernel was created to simulate the blur effect, and both the image and kernel were transformed using the Fourier Transform. By dividing the image's frequency spectrum by the kernel's spectrum, the blur was effectively removed. Finally, the result was converted back to the spatial domain using the Inverse FFT, restoring the image's clarity.

Bears:



The dark bear image was enhanced using histogram equalization to improve contrast and reveal hidden details. Additionally, gamma correction with a factor of 0.9 slightly brightened the image, enhancing visibility without overexposing brighter regions. This combination effectively balanced the image's contrast and brightness.