7 - 8.7. Polar Transformations

October 7, 2024

Jarrian Vince G. Gojar
Instructor I
College of Information and Communications Technology, Sorsogon State University, Philippines

1 Introduction

`Polar Transformation` is a technique used in image processing to convert the image from Cartesian coordinates to Polar coordinates. The Polar coordinates are represented in terms of radius and angle. The Polar transformation is used in various applications like edge detection, image registration, and image recognition.

There are two types of Polar transformations:

1. **Radial Transformation**: In this transformation, the radius is kept constant, and the angle is varied. The transformation is used to detect the edges in the image.
2. **Log-Polar Transformation**: In this transformation, the radius is varied, and the angle is kept constant. The transformation is used in image registration and image recognition.

Read More:

- [Polar Transformation](#)

2 Setup

```
[ ]: %pip install opencv-python opencv-contrib-python numpy matplotlib
```

3 Initial Setup

```
[97]: # Import Libraries
      import os
      import cv2
      import numpy as np
      import matplotlib.pyplot as plt

      # Asset Root
      asset_root = os.path.join(os.getcwd(), '../../assets')

      # Image Path
```
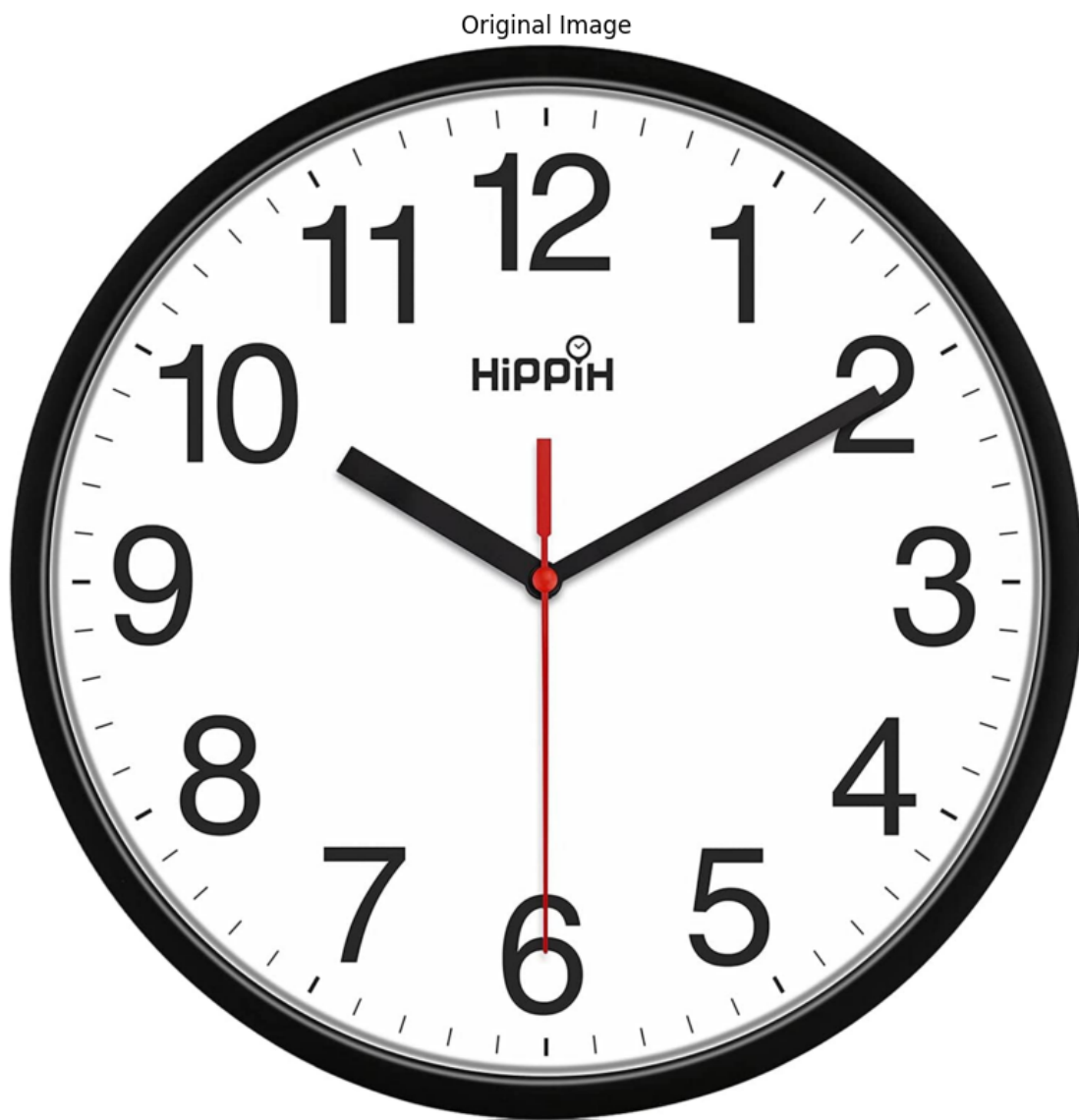
```
image_path = os.path.join(asset_root, 'images', 'clock.jpg')

# Read Image and convert to RGB
input_image = cv2.cvtColor(cv2.imread(image_path), cv2.COLOR_BGR2RGB)

# Display Both Image
plt.figure("Polar Transformation", figsize=(10, 10))

plt.imshow(input_image)
plt.title("Original Image")
plt.axis('off')

plt.show()
```

Original Image

3.1 Radial-Polar Transformation

In `Radial-Polar Transformation`, the radius is kept constant, and the angle is varied. The transformation is used to detect the edges in the image. The transformation is performed using the `cv2.linearPolar()` function in OpenCV.

The function `cv2.linearPolar()` takes the following parameters:

- **src**: The input image.
- **center**: The center of the transformation.
- **maxRadius**: The maximum radius of the transformation.
- **flags**: The interpolation method.

```python
# Radial-Polar Transformation
def radial_polar_transform(image, center, max_radius):
    # Radial-Polar Transformation
    radial_polar_image = cv2.linearPolar(image, center, max_radius, cv2.
 ↪WARP_POLAR_LINEAR)


    return cv2.rotate(radial_polar_image, cv2.ROTATE_90_CLOCKWISE)


# Center of the Image
center = (input_image.shape[1] // 2, input_image.shape[0] // 2)

# Maximum Radius
max_radius = (input_image.shape[1] + input_image.shape[0]) // 2

# Radial-Polar Transformation
radial_polar_image = radial_polar_transform(input_image, center, max_radius)

# Display Both Image
plt.figure("Radial-Polar Transformation", figsize=(20, 10))

plt.subplot(1, 2, 1)
plt.imshow(input_image)
plt.title("Original Image and Center")
plt.scatter(center[0], center[1], color='blue', s=10)
plt.axis('off')

plt.subplot(1, 2, 2)
plt.imshow(radial_polar_image)
plt.title("Radial-Polar Image")
plt.axis('off')

plt.show()
```
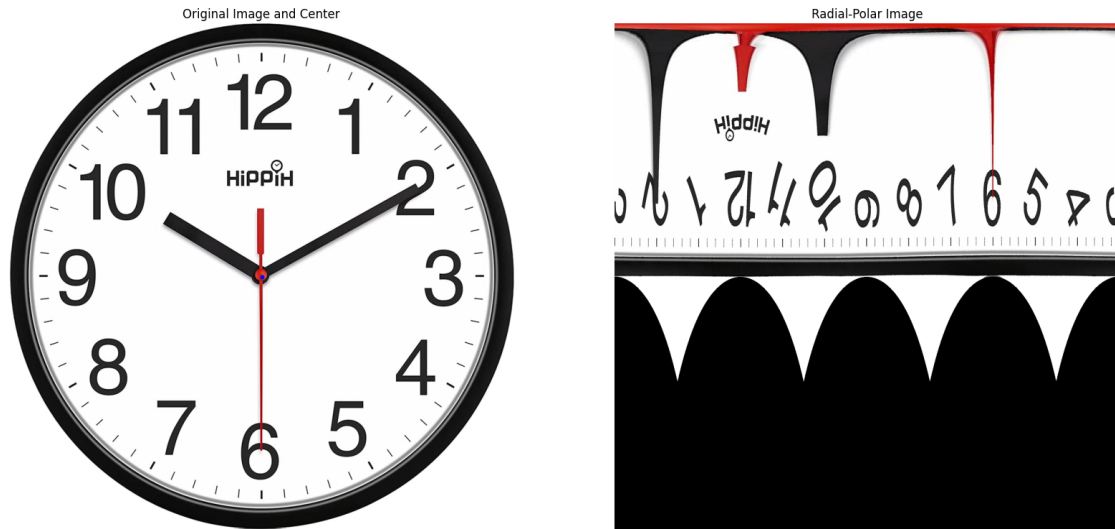
Original Image and Center

Radial-Polar Image

The above code performs the `Radial-Polar Transformation` on the input image. The transformation is used to detect the edges in the image. The center of the image is calculated using the shape of the image. The maximum radius is calculated as the maximum value of the center.

The `cv2.linearPolar()` function is used to perform the transformation. The function takes the input image, center, and maximum radius as input parameters. The transformation is performed using the `cv2.WARP_POLAR_LINEAR` interpolation method.

3.2 Log-Polar Transformation

In `Log-Polar Transformation`, the radius is varied, and the angle is kept constant. The transformation is used in image registration and image recognition. The transformation is performed using the `cv2.logPolar()` function in OpenCV.

The function `cv2.logPolar()` takes the following parameters:

- **src**: The input image.
- **center**: The center of the transformation.
- **maxRadius**: The maximum radius of the transformation.
- **flags**: The interpolation method.

```python
[99]: # Log-Polar Transformation
def log_polar_transform(image, center, max_radius):
    # Log-Polar Transformation
    log_polar_image = cv2.logPolar(image, center, max_radius, cv2.
 ↪WARP_POLAR_LOG)

    return cv2.rotate(log_polar_image, cv2.ROTATE_90_CLOCKWISE)

# Set the Center of the Image
center = (input_image.shape[1] // 2, input_image.shape[0] // 2)
```

```python
# Set the Maximum Radius
max_radius = (input_image.shape[1] + input_image.shape[0]) // 4 // 4

# Log-Polar Transformation
log_polar_image = log_polar_transform(input_image, center, max_radius)

# Display Both Image
plt.figure("Log-Polar Transformation", figsize=(20, 10))

plt.subplot(1, 2, 1)
plt.imshow(input_image)
plt.title("Original Image and Center")
plt.scatter(center[0], center[1], color='blue', s=10)
plt.axis('off')

plt.subplot(1, 2, 2)
plt.imshow(log_polar_image)
plt.title("Log-Polar Image")
plt.axis('off')

plt.show()
```
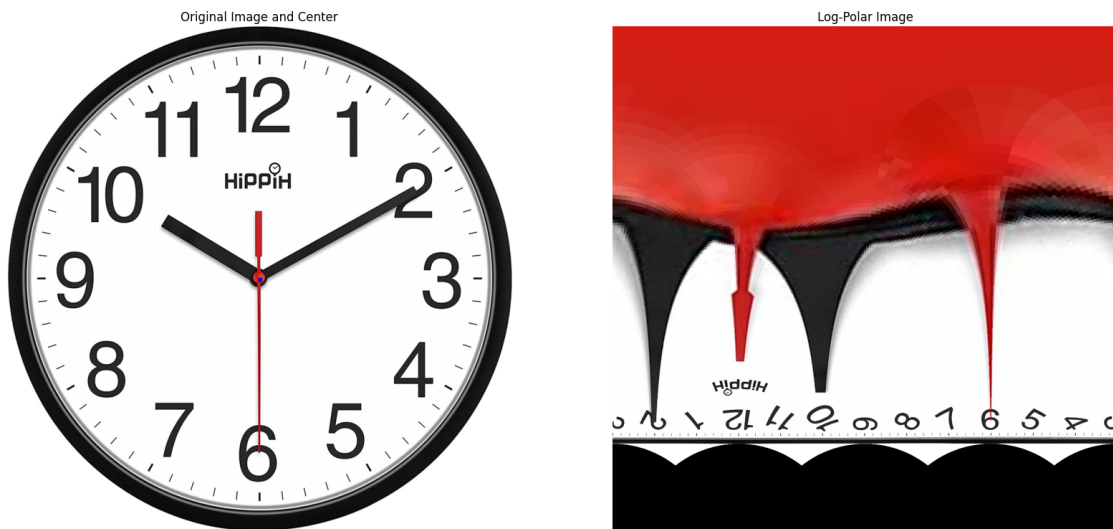

The above code performs the `Log-Polar Transformation` on the input image. The transformation is used in image registration and image recognition. The center of the image is calculated using the shape of the image. The maximum radius is calculated as the maximum value of the center.

The `cv2.logPolar()` function is used to perform the transformation. The function takes the input image, center, and maximum radius as input parameters. The transformation is performed using the `cv2.WARP_POLAR_LOG` interpolation method.

4   Summary

- `Polar Transformation` is a technique used in image processing to convert the image from Cartesian coordinates to Polar coordinates.
- The Polar coordinates are represented in terms of radius and angle.
- There are two types of Polar transformations: Radial Transformation and Log-Polar Transformation.
- `Radial Transformation` is used to detect the edges in the image.
- `Log-Polar Transformation` is used in image registration and image recognition.
- The `cv2.linearPolar()` function is used to perform the Radial-Polar Transformation.
- The `cv2.logPolar()` function is used to perform the Log-Polar Transformation.

Read More:

- Polar Transformation
- cv2.linearPolar()
- cv2.logPolar()

5   References

- Thomas G. (2022). Graphic Designing: A Step-by-Step Guide (Advanced). Larsen & Keller. ISBN: 978-1-64172-536-1
- Singh M. (2022). Computer Graphics and Multimedia. Random Publications LLP. ISBN: 978-93-93884-95-4
- Singh M. (2022). Computer Graphics Science. Random Publications LLP. ISBN: 978-93-93884-03-9
- Singh M. (2022). Computer Graphics Software. Random Publications LLP. ISBN: 9789393884114
- Tyagi, V. (2021). Understanding Digital Image Processing. CRC Press.
- Ikeuchi, K. (Ed.). (2021). Computer Vision: A Reference Guide (2nd ed.). Springer.
- Bhuyan, M. K. (2020). Computer Vision and Image Processing. CRC Press.
- Howse, J., & Minichino, J. (2020). Learning OpenCV 4 Computer Vision with Python 3: Get to grips with tools, techniques, and algorithms for computer vision and machine learning. Packt Publishing Ltd.
- Kinser, J. M. (2019). Image Operators: Image Processing in Python. CRC Press.