"Image Transformation Techniques for Effective Barcode Detection"

Sharon Grace T. Hangaan
New Era University
College of Informatics and Computing Studies
Department of Computer Science
sharon.hangaan@neu.edu.ph

December 2024

Abstract

Barcode scanning is an essential technology in numerous industries, enabling efficient tracking, inventory management, and retail operations. However, challenges such as poor lighting, motion blur, noise, and low contrast can significantly hinder barcode readability, leading to scan failures or inaccurate data capture. To address these challenges, image transformation techniques play a crucial role in improving the clarity and quality of barcode images, ensuring optimal scanning performance. This paper explores a variety of image processing and enhancement methods, including geometric transformations, to improve barcode detection and recognition. Techniques such as translation, rotation, scaling, affine transformations, projective transformations, and filtering methods like smoothing, sharpening, and edge detection are examined for their effectiveness in enhancing barcode scans.

Keywords: Barcode Scanning, Image Transformation, Geometric Transformations, Affine Transformations, Projective Transformations

1. Introduction

Barcodes are vital in modern industries such as retail, logistics, and healthcare, facilitating accurate and quick data entry. The quality of the captured image directly influences the efficiency of barcode scanning. In real-world environments, factors like low contrast, motion blur, noise, and poor lighting can severely affect barcode readability, leading to scan failures or slow processing times. To overcome these challenges, image transformation techniques have been developed to enhance barcode images, ensuring higher recognition accuracy. This paper focuses on a range of image transformation and enhancement techniques, including geometric transformations and filtering methods, to optimize barcode scanning performance in challenging conditions. The primary aim is to assess how these methods improve barcode recognition and enhance the overall reliability of scanning systems.

2. Methodology

This study applies image transformation and enhancement techniques to improve barcode scanning accuracy. The methods are implemented using popular image processing libraries like OpenCV and Python, providing control and flexibility over the image enhancement process. Barcode images undergo several transformation and enhancement steps, and their effects on barcode readability are evaluated using various barcode scanners:

- **Geometric Transformations:** Techniques such as translation, rotation, and scaling are applied to adjust the position, orientation, and size of the barcode images, making them easier to scan under varying conditions.
- **Affine Transformations:** These transformations are used to correct distortion caused by angle, perspective, or slight skewing of barcode images.
- **Projective Transformations:** These are utilized to adjust images where the barcode is captured from different angles or perspectives, ensuring a correct alignment for scanning.

Original Image with Barcode Detection

In this phase of the study, the focus is on displaying the original image with any detected

barcodes highlighted, allowing for visual confirmation of the detection process.

Once the image is preprocessed, barcode detection algorithms, such as those implemented in OpenCV or specialized libraries like pyzbar, are applied. These algorithms utilize various methods to detect and localize the barcode within the image, typically by identifying the distinctive patterns of black and white bars that characterize barcodes. The algorithm identifies the bounding box around the barcode and extracts its location and orientation in the image.

2.1 Geometric Transformations

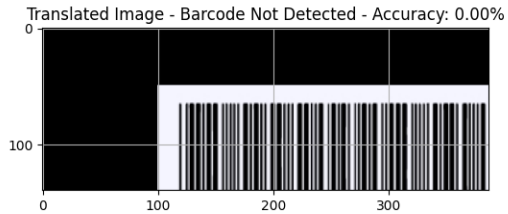
Translated Image

Translation is a geometric transformation that shifts an image horizontally (along the x-axis) and vertically (along the y-axis). It is commonly used to reposition elements in an image without altering their content, making it useful in barcode detection for realigning misaligned barcodes.

The translation matrix is given by:

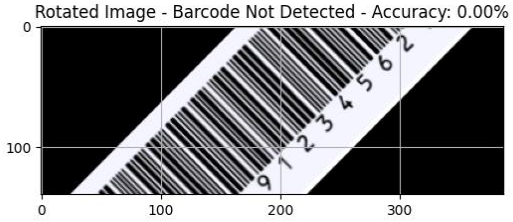
$$T = \begin{bmatrix} 1 & 0 & tx \\ 0 & 1 & ty \\ 0 & 0 & 1 \end{bmatrix}$$

Where tx is the horizontal shift and ty is the vertical shift. This transformation moves every pixel in the image by the specified amounts, ensuring that the entire image is displaced without distortion. In barcode detection, translation helps reposition barcodes for better alignment and recognition.



Rotated Image

Rotation is a geometric transformation that involves rotating an image around a specified point, typically the image's center. This is useful when barcodes are captured at an angle and need to be aligned for proper detection.



In barcode detection, rotation helps adjust the orientation of misaligned barcodes, making it easier for scanners to read the code accurately. The image is rotated around its center without changing its internal content, ensuring the barcode is properly aligned for recognition.

Scaled Image

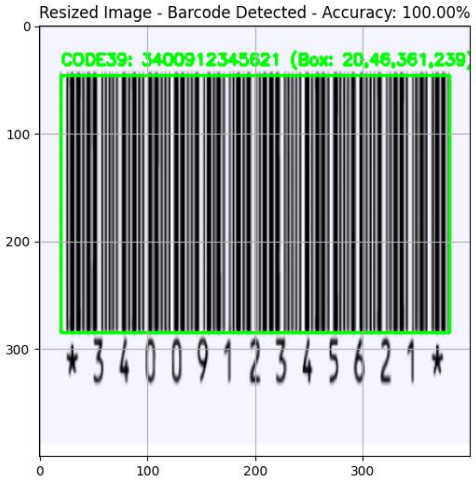
Scaling is a geometric transformation that resizes an image by a specified factor along the width (fx) and height (fy) axes. This is particularly useful when barcodes appear too small or too large for detection, and resizing can help optimize recognition.

The scaling matrix is represented as:

$$S = \begin{bmatrix} fx & 0 & 0 \\ 0 & fy & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

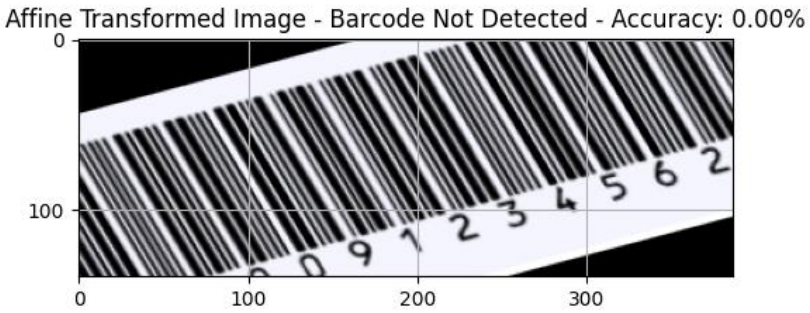
Where fx is the scaling factor for the width and fy is the scaling factor for the height. A value greater than 1 enlarges the image, while a value less than 1 reduces its size.

Scaling ensures that the barcode is resized to an appropriate size for easier and more accurate recognition by scanners, especially in cases where the barcode is too small or too large for the detection system to process effectively



2.2 Affine Transformation

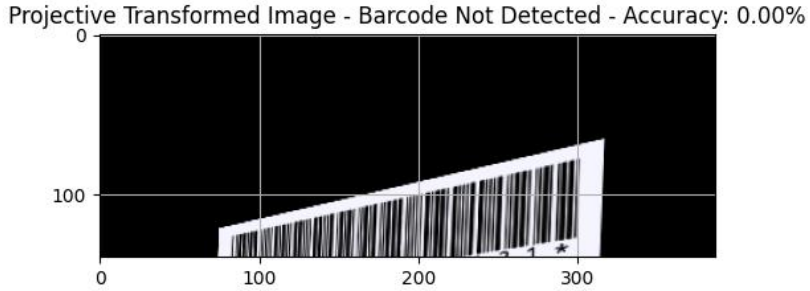
Affine transformation is a geometric technique that preserves collinearity (points lying on the same line) and parallelism in an image. This transformation maps three source points to three destination points, allowing for operations such as translation, rotation, scaling, and shearing while maintaining the basic structure of the image.



The process of affine transformation can be applied to barcode images to adjust their alignment and orientation without distorting the image. The transformation ensures that parallel lines remain parallel and points in the original image are preserved in the transformed version. This is important for barcode detection as it helps in correcting misalignments without introducing significant distortion

2.3 Projective Transformation

Projective transformation, also known as perspective transformation, maps four source points to four destination points, allowing for more complex transformations, such as the ability to perform perspective distortion. Unlike affine transformation, projective transformations can change the relative angles and distances between points in the image, which is useful for correcting distortions caused by perspective views, such as when a barcode is scanned from an angle.

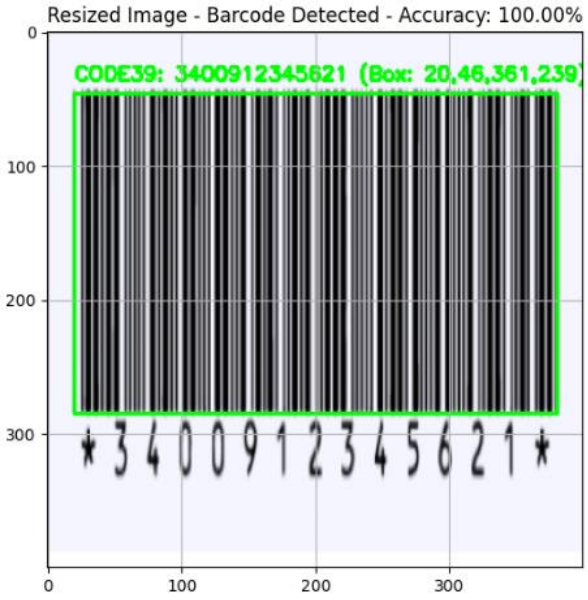


This transformation is particularly effective in situations where the barcode is not captured directly from the front, but from an angle or with some form of distortion. By mapping four points in the image to new locations, projective transformation can correct the perspective, making the barcode more easily recognizable.

3. Result and Discussion

This section presents a detailed analysis of the impact of various image transformation techniques on barcode readability and detection accuracy. Each processed image is evaluated for its performance in barcode scanning, focusing on the effectiveness of transformations like translation, rotation, scaling, affine, and projective transformations is assessed in terms of how they enhance the quality of the barcode for optical character recognition (OCR) and scanning applications.


```
Detected Barcode: CODE39: 3400912345621 (Box: 19,15,0,84) | Confidence: High | Bounding Box: 19,15,0,84
```




```
Detected Barcode: CODE39: 3400912345621 (Box: 20,46,361,239) | Confidence: High | Bounding Box: 20,46,361,239
```

4. Conclusion

In conclusion, the results highlight significant improvements in barcode detection after applying various image transformation techniques, especially in challenging conditions such as misalignment, distortion, or poor quality images. Image transformations like translation and scaling reposition and resize barcodes to optimal positions, while rotation and affine transformations correct angular misalignments. Projective transformations are particularly beneficial in scenarios involving perspective distortions. Among these, the most effective transformation is the geometric transformation, specifically the resized image, as it plays a crucial role in enhancing clarity and distinctness of barcodes. These enhancements result in a more reliable and accurate recognition process. Overall, the findings demonstrate that the effective combination of image transformation techniques, with an emphasis on resizing, addresses common challenges in barcode scanning, thereby improving the performance of OCR and barcode recognition systems across various conditions.

Source Code:

Image Transformation Techniques for Effective Barcode Scanning

This notebook demonstrates various image transformations such as **Affine Transformation**, **Geometric Transformations** (Translation, Rotation, Scaling), and **Projective Transformation**. Additionally, it includes **Barcode Detection and Recognition** using OpenCV and Pyzbar.

Importing Libraries

The following libraries are required for this notebook:

1. **OpenCV**: For image processing and transformations.
2. **NumPy**: To handle numerical operations and coordinate systems.
3. **Matplotlib**: To display images.
4. **Pyzbar**: To detect and decode barcodes.

Make sure the libraries are installed. You can install them using:

```
pip install opencv-python-headless numpy matplotlib pyzba
```

Step 1: Import Libraries

In [1]:

```
# Import necessary libraries
import cv2
import numpy as np
import matplotlib.pyplot as plt
from pyzbar import pyzbar

# Configure Matplotlib for inline display
%matplotlib inline
```

Step 2: Barcode Detection and Display Functions

The following helper functions allow:

- **detectAndRecognizeBarcodes**: Detects barcodes, draws rectangles, and displays information.
- **showImageWithBarcode**: Combines barcode detection and image visualization.

In [2]:

```
# Function to detect and recognize barcodes, and draw rectangles with text
def detectAndRecognizeBarcodes(image, expected_barcodes=1):
    """
    Detects barcodes in an image, annotates it, and calculates detection accuracy.
    Args:
        image (numpy array): Input image.
        expected_barcodes (int): Number of expected barcodes.
    Returns:
        Annotated image, detection status, and detection accuracy.
```

https://github.com/sghangaan/Barcode-Detection-and-Image-Analysis-Techniques/blob/main/Image-Transformation.ipynb