Assingment-1

Siddhesh Deshpande(IMT2022080)

March 2025

Part 1

Preprocessing

The input image undergoes preprocessing to enhance its features and simplify detection. The steps include:

- Conversion to grayscale to reduce computational complexity.
- Resizing such that the longest dimension is 700 pixels.
- Gaussian blur to reduce noise.
- Adaptive thresholding to obtain a binary representation of the image.

Circular Object Detection

Contours are extracted from the thresholded image, and a circularity measure is used to filter valid coin-like objects. The circularity measure is given by:

$$C = \frac{4\pi A}{P^2} \tag{1}$$

where A is the contour area and P is the perimeter. Contours with $0.7 < C < 1.2$ and sufficiently large area are considered as coins.

Segmentation and Saving

For each detected coin, a mask is created, and the region of interest is extracted. The segmented coins are saved as individual images.

Results

The algorithm detected 7 coins in the input image (`coins.jpg`), which matches the actual count of 7 coins in the image. Additionally, an outlined image highlighting detected coins is generated.

What Worked Well

- Adaptive thresholding significantly improved contrast, making contour detection more effective.
- Using the circularity measure helped in accurately filtering out non-circular objects.
- The approach was able to successfully detect well-separated and clearly visible coins.

Challenges and Limitations

- Small or overlapping coins were sometimes missed due to contour approximation errors.
- Lighting variations affected thresholding results, requiring manual tuning of parameters in some cases.
- The method assumes that coins are circular, which might fail if some are partially occluded or deformed.

Final Approach

Based on the observations, the final approach consists of:

- Using adaptive thresholding to improve contrast across varying backgrounds.
- Applying Gaussian blur to minimize noise before contour extraction.
- Filtering objects using a combination of circularity measure and area constraints.
- Ensuring segmentation captures the full shape of each detected coin by refining the mask generation process.

Observations and Conclusion

- The method accurately detects circular objects with good precision.
- Adaptive thresholding helps in improving contrast for better contour detection.
- Some small or overlapping coins may be missed due to contour approximation errors.
- The approach is effective under controlled lighting conditions but may need adjustments for varying backgrounds.
- Further improvements could include using edge detection techniques or machine learning for better segmentation.

Part 2

Feature Detection and Matching

- SIFT (Scale-Invariant Feature Transform) is used to detect keypoints and extract descriptors from each image.
- BFMatcher (Brute-Force Matcher) with Lowe's ratio test is used to find correspondences between keypoints in different images.
- Homography is computed using RANSAC (Random Sample Consensus) to align images.

Image Warping and Stitching

- The second image is warped onto the first using the homography matrix.
- The images are merged, and black borders are removed using contour detection.

Challenges and Observations

What Worked Well

- SIFT proved effective in detecting distinctive features even under variations in lighting and viewpoint.
- RANSAC helped in reducing outlier matches, ensuring accurate alignment.
- Cropping black borders improved the final output quality.

What Didn't Work

- In cases with low feature overlap, feature matching failed, causing poor alignment.
- When images had significant perspective distortions, stitching errors were observed.
- Direct blending without intensity adjustment resulted in noticeable seams.

Final Approach

Based on the observations, the following refinements were applied:

- Image resizing was introduced to ensure a uniform scale before processing.

- A sequential stitching approach was implemented to handle multiple images effectively.
- A visualization step was added to inspect feature matches before final stitching.

Results

The final stitched image successfully merges multiple images into a seamless panorama. The method performs well for images with sufficient overlap and well-defined features. However, improvements could be made in handling images with large perspective differences.

Conclusion

The implemented approach demonstrates effective image stitching using feature-based methods.

NOTE-All the code used for this assingment can be found at Click Here