# ACIP

Activity 2: Automatic CP extraction with Matlab feature functions

Anna, Kimia, Hamza

April 2024

## Introduction

In this study, we explore different features, with a focus on the Scale-Invariant Feature Transform (SIFT) and Binary Robust Invariant Scalable Keypoints (BRISK), to determine their robustness against image transformations. SIFT [1] offers scale and rotation invariance. BRISK [2] provides a computationally efficient alternative with scale and rotation invariance capabilities. This report will demonstrate these feature detectors' suitability for real-world applications by comparing their performance on an original image and its transformation.

## 1 Methodology

### 1.1 Image Acquisition and Feature Extraction

The image of a coin with its rich textural features and distinct edges and contours provides a challenging, but interpretable set of keypoints for detection algorithms. For this task, we checked multiple methods, including SURF, MSER, FAST, MinEigen, Harris, ORB, with a detailed analysis of the SIFT and BRISK features, which showcased the best performance for our image and transformations. SIFT begins by detecting keypoints through the Difference of Gaussians (DoG) approach, isolating potential points of interest that remain consistent across scales. It refines these points through localization, and orientation assignment, and ultimately describes each keypoint using a 128-dimension descriptor, capturing local gradients across subregions. BRISK, a faster alternative, is efficient in binary descriptor computation and invariant to rotations.

### 1.2 Image Transformation

In the first activity, we applied horizontal shearing and rigid transformations to an image. With horizontal shearing, we tilt the image horizontally, without any vertical displacement or scaling. With rigid transformations, we perform a combination of rotation and translation but maintain the distances between points.

### 1.3 Feature Extraction from Transformed Images

Applying the same feature extraction methods to transformed images tests the hypothesis that the selected features are invariant to the applied transformations. A successful feature extraction method would detect a similar distribution of keypoints in both the original and transformed images, despite any distortion or movement.

## 1.4 SIFT

The Scale-Invariant Feature Transform (SIFT) is a technique introduced by David G. Lowe that detects and describes local features in images. The method is designed to be robust against changes in image scale, rotation, and illumination, as well as changes in 3D viewpoint.
SIFT starts by searching for the 3D scale-space extrema in the image using a Difference of Gaussians (DoG) function, which involves subtracting one blurred version of an original image from another, less blurred version of the image. This step identifies potential interest points that are invariant to scale. This scanning and detection process can be sensitive to noise; therefore, it is followed by an interpolation that refines the localization of the extrema, and by filtering to discard unreliable detections. At this point, each keypoint has x, y which are the position of the keypoint and a scale in which the keypoint was found.
Then, we have the orientation assignment. Apart from the scale for each keypoint, SIFT also calculates the main orientation; This is done by taking a square window around the detected feature in the corresponding scale, computing edge orientation or gradient direction for each pixel, creating a weighted direction histogram made of 36 bins (10 degrees per pixel, and the weights are the gradient magnitude), and taking the peak in the histogram as the dominant direction. In the end, each keypoint is made of x, y, scale, and orientation.
Apart from a detector, SIFT is also a descriptor which is considered as one of its strengths. For each keypoint (x, y, scale, orientation) a normalized patch will be used; after undoing the effect of rotation and scaling (orientation and scale invariance). The patch is divided into 16x16 regions (4x4 grid of cells). An 8-bin weighted orientation histogram is calculated for each cell, which will give us the final descriptor: 16 cells* 8 orientations: 128 dimension descriptor.

## 1.5 BRISK

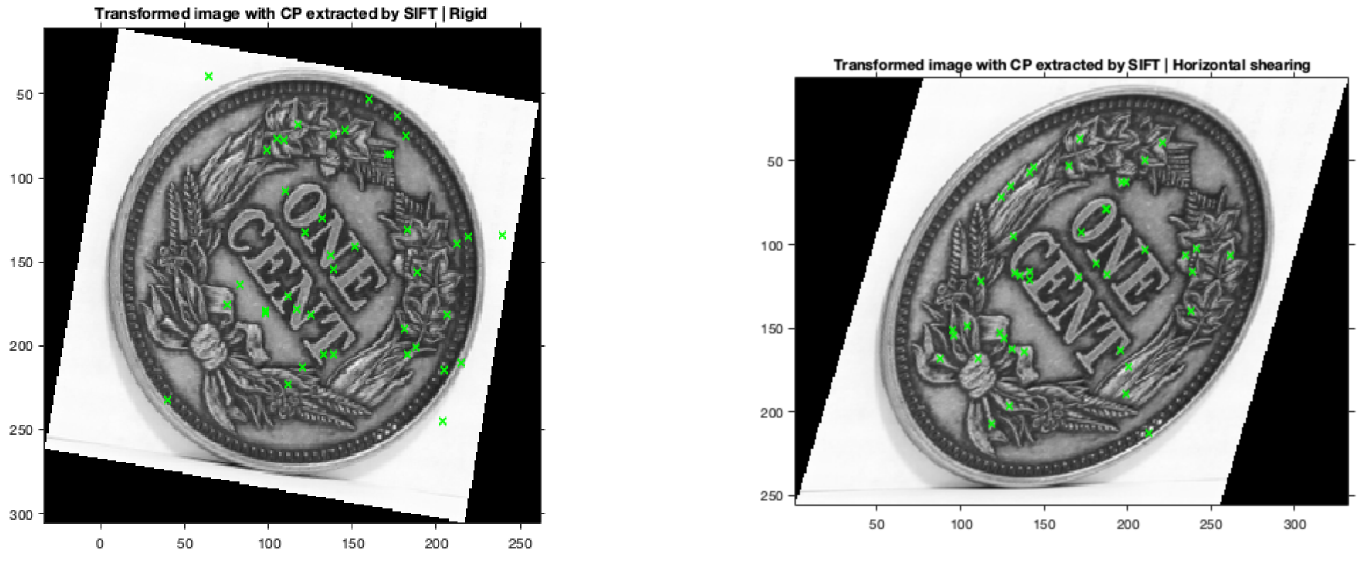
BRISK, short for Binary Robust Invariant Scalable Keypoints, is a feature point detection and description algorithm with scale invariance and rotation invariance, developed in 2011. The BRISK algorithm is designed to be a fast and efficient alternative to other feature detectors and descriptors like SIFT and SURF, with particular emphasis on providing robust performance in real-time applications. BRISK is both a detector and a descriptor; it not only identifies points of interest in an image but also describes the regions around these points in a way that is useful for matching keypoints between different images. Compared with the traditional algorithms, the matching speed of BRISK is faster and the storage memory is lower.
Similar to SIFT, BRISK starts by generating a scale-space representation of the image. At each scale level, the algorithm performs a FAST (Features from Accelerated Segment Test) corner detection and uses the FAST detector score s. It checks for corners by examining a circle of 16 pixels around each candidate pixel and ensures that there are at least 9 contiguous pixels in the circle that are either all darker or all lighter than the central pixel by a certain intensity threshold. These corners are candidates for keypoints. After that, non-maxima supression is performed on each octave and layers between in a way that the s score is maximal within 3x3 neighborhood.
Next, using the points obtained from the previous section, a 2D quadratic function is fit to the 3x3 neighbor of each point, and subpixel maxima are determined (through all the layers). These maxima are later interpolated across scale space, and the local maxima is chosen as the scale that the feature was found in. BRISK generates a descriptor based on a unique sampling pattern that consists of pairs of points at various distances around each keypoint, separating the pairs of pixels into subsets of short-distance and long-distance pairs. Each point in the sampling pattern is smoothed using a Gaussian filter. The standard deviation of the filter is proportional to the distance from the center to avoid aliasing effects in the sampled intensities. Then, for each pair of sampling points, the local gradient is calculated based on the smoothed intensity values. This involves determining the difference in intensities and normalizing by the distance between the points.
BRISK calculates an overall direction for the keypoint using long-distance pairs from the set of all point pairs. This direction is found by averaging the gradients from these pairs, effectively determining a dominant gradient direction which helps in achieving rotation invariance.
The keypoint's sampling pattern is also rotated by an angle, which is derived from the calculated gradient. This is crucial for ensuring that the descriptor is rotation invariant. Finally, a binary string is constructed by performing brightness comparisons between short-distance pairs in the rotated pattern. For each pair, the intensity comparison results in either 0 or 1; if the intensity at the first point of the pair is greater than the second, the result is 1; otherwise, it is 0. This binary string forms the descriptor. BRISK effectively generates invariant descriptors to scale and orientation while maintaining a high degree of distinctiveness and computational efficiency, making it suitable for real-time applications.

# 2 Results

## 2.1 Feature Detection in Original and Transformed Images with SIFT


Figure 1: Original image with CP extracted by SIFT

The SIFT features were analyzed for the original and transformed images, with the results depicted in Figures 1 and 2. Upon examination, we observe that, following horizontal shearing, the CPs largely correspond between the two images, with only minor discrepancies. This suggests a robustness of the SIFT features to the applied shearing transformation. Contrastingly, the CPs from the image subjected to a rigid transformation demonstrate a significant deviation from those in the original image, indicating a reduced stability of the SIFT features under such conditions.




(a) Rigid transformed image with CP by SIFT

(b) Shearing transformed image with CP by SIFT

## 2.2 Feature Detection in Original and Transformed Images with BRISK

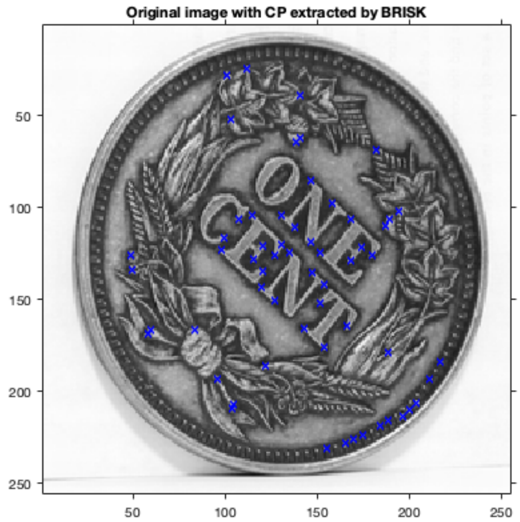
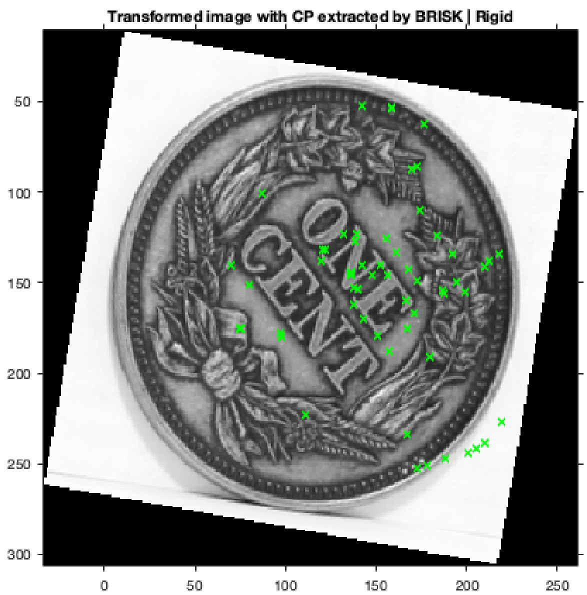
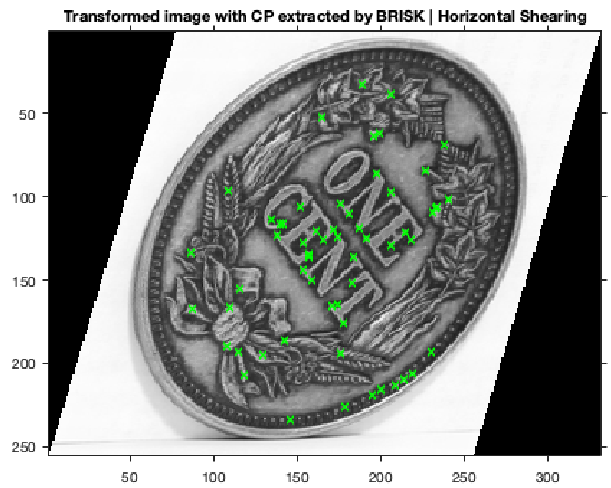



Figure 3: Original image with CP extracted by BRISK

In our experiment with the BRISK feature detection method, we also checked its robustness for both transformations: horizontal shearing and rigid transformation. For the horizontal shearing, the result of which is presented in Figure 4b, the feature performed very well and showcased quite a small error. On the contrary, the results for rigid transform, depicted in Figure 4a show significant differences between the CPs in the original image 3 and those in the transformed image. Notably, the method accurately detected keypoints around distinctive letters like E, C, and N, and on the intricate patterns of the coin in both the original and transformed images. However, it struggled with keypoints around the outer border of the coin, particularly in the transformed image. This discrepancy suggests that while BRISK effectively captures features in areas with high pattern density, it may be less reliable around less structured, peripheral regions. These observations indicate that further investigation is needed to determine when and how the BRISK method is most effective.



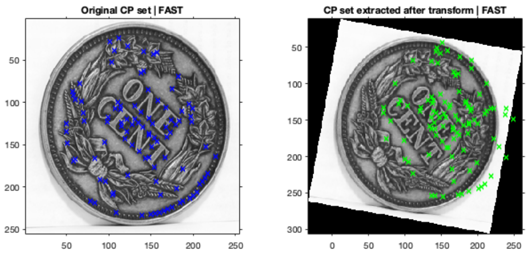

(a) Rigid transformed image with CP by BRISK



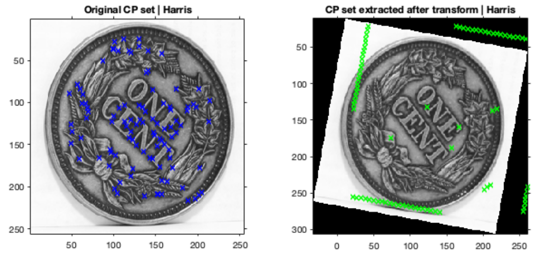

(b) Shearing transformed image with CP by BRISK

## 2.3 Comparison of Additional Feature Detection Methods

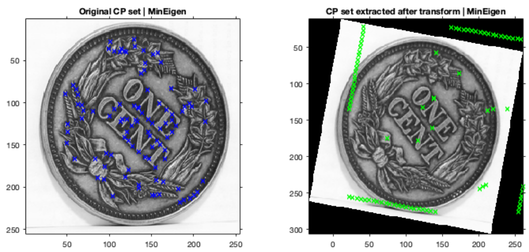
To investigate more, we also experimented with other feature detection methods. In this section we will explore the results of applying the FAST [4], Harris [5], MinEigen [6], MSER [7], ORB [8], and SURF [3] features, to evaluate their performance against SIFT [1] and BRISK [2].



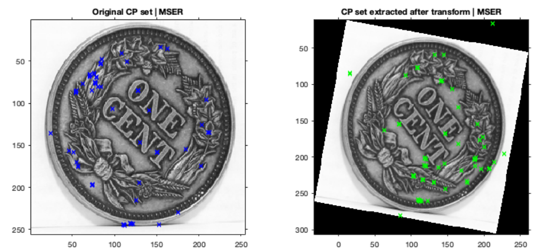

(a) FAST



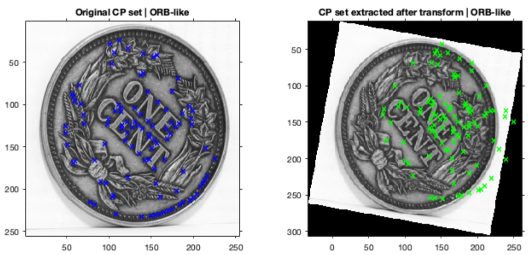

(b) Harris



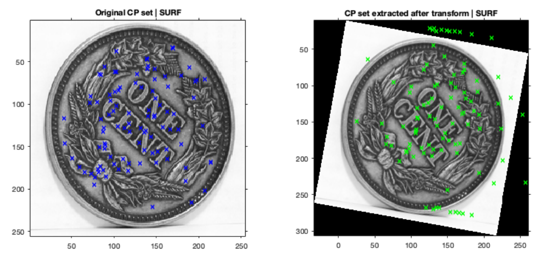

(c) MinEigen




(d) MSER




(e) ORB




(f) SURF

Figure 5: Original and transformed images with CP extracted by various methods

Although each of these methods has distinct advantages in specific scenarios, the overall results were not as satisfactory compared to SIFT and BRISK, particularly in handling the transformations applied in our experiments. However, for completeness, we include these comparisons to demonstrate the variability in feature detection robustness across different methods.

## 2.4 Observations on Other Methods

The tests revealed that methods like SURF and ORB, while effective in certain conditions, did not consistently match the robustness of SIFT and BRISK under the transformations we applied. FAST and Harris provided quick detection but were prone to missing key features following the transformations. MinEigen and MSER, although excellent in detecting stable regions across scale changes, also fell short in maintaining consistency after transformations, compared to SIFT and BRISK.

These results suggest that while SIFT and BRISK remain our primary choices due to their robustness and reliability, other methods might be suitable for specific applications where the nature of image transformations differs from those tested in this study.

# 3 Conclusions

## 3.1 Robustness of SIFT and BRISK

SIFT, with its robust design against scale changes and rotations, generally performed well under horizontal shearing. This performance aligns with its method of detecting keypoints through a scale-space extrema search, optimized for such invariance. However, under rigid transformations, SIFT showed some limitations, highlighting areas for potential enhancement.

At the same time, BRISK, known for its quick binary descriptor system and also designed to be invariant to scale and rotation, faced challenges under rigid transformations. Although it performed well in regions with dense patterns and distinct features, such as letters and intricate coin patterns, its reliability diminished around the outer edges of the coin. These findings underscore the need for further testing and potential refinement of BRISK's approach to ensure consistent performance across different image areas and transformation types.

## 3.2 Comparison of Feature Detection Methods

Our experimental results reveal that both SIFT and BRISK methods demonstrated comparable robustness under the transformations applied, with each showing particular strengths and vulnerabilities. SIFT exhibited consistent performance under horizontal shearing, aligning with its design for scale and rotation invariance. This is reflected by its ability to maintain keypoint information through its sophisticated scale-space extrema detection technique.

Similarly, BRISK, which is optimized for speed and efficiency with its binary descriptor system, also performed well, especially in handling horizontal shearing transformations. It effectively identified keypoints around detailed features such as letters and intricate patterns on the coin, showcasing its capability in areas with high pattern density. However, like SIFT, BRISK showed some limitations under rigid transformations, particularly around the peripheral regions of the coin where fewer distinct features exist.

This indicates that while both methods are capable of handling complex image transformations, their performance can vary depending on the nature of the transformation and the specific regions within the images. This suggests that the choice between using SIFT and BRISK should consider not only the computational efficiency and the detail of the feature detection required but also how these methods perform across different types of transformations and image areas.

# References


[1] Lowe D. G., "Distinctive image features from scale-invariant keypoints," in International journal of computer vision, 60, 91-110., 2004.

[2] Leutenegger, S., Chli, M., Siegwart, R. Y., "BRISK: Binary robust invariant scalable keypoints," in International conference on computer vision, pp. 2548-2555, IEEE, 2011.

[3] Bay, H., Tuytelaars, T., Van Gool, L., "SURF: Speeded up robust features," in Computer Vision and Image Understanding, 110(3), 346-359, 2008.

[4] Rosten, E., Drummond, T., "Machine learning for high-speed corner detection," in European Conference on Computer Vision, pp. 430-443, Springer, 2006.

[5] Harris, C., Stephens, M., "A combined corner and edge detector," in Proceedings of the 4th Alvey Vision Conference, pp. 147-151, 1988.

[6] Shi, J., Tomasi, C., "Good features to track," in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 593-600, 1994.

[7] Matas, J., Chum, O., Urban, M., Pajdla, T., "Robust wide-baseline stereo from maximally stable extremal regions," in Image and Vision Computing, 22(10), pp. 761-767, 2004.

[8] Rublee, E., Rabaud, V., Konolige, K., Bradski, G., "ORB: An efficient alternative to SIFT or SURF," in International Conference on Computer Vision, pp. 2564-2571, IEEE, 2011.