IJWR

Robust Corner Detection Using Local Extrema Differences

Reza Yazdi[a*], Hassan Khotanlou[a], Hosna Khademfar[b]

[a] RIV Lab, Department of Computer Engineering, Faculty of Engineering ,Bu-Ali Sina University, Hamadan, Iran; r.yazdi@eng.basu.ac.ir, Khotanlou@basu.ac.ir

[b] Department of Artificial Intelligence, Shargh Golestan Higher Education Institute, Golestan, Iran; hosnakhadem@gmail.com

ABSTRACT

Corner detection, crucial for many computer vision tasks due to corner's distinct structural properties, often relies on traditional intensity-based detectors developed before 2000. This paper introduces a novel intensity-based corner detector that surpasses existing methods by solely analyzing pixel intensity within a 3×3 neighborhood. Our approach leverages a unique corner response function derived from intensity sorting and difference calculations. We conduct a comprehensive evaluation comparing our detector to seven established algorithms using five benchmark images with ground truth corner locations. The evaluation encompasses detection accuracy, localization error under varying noise levels, and repeatability under transformations and degradations. This assessment utilizes 28 diverse images without ground truth data. Experimental results demonstrate the proposed detector's superior overall performance by 3%. It achieves better accuracy in corner localization and reduces both missed detections and false positives. Furthermore, requiring only one parameter for adjustment, it offers computational efficiency and real-time processing potential. Additionally, the generated corner response map holds promise for integration with deep learning architectures, opening possibilities for further exploration.

Keywords*— *Corner Points Detection, Corner Detection, Interested Points, Corner Points.

1. Introduction

The corners of objects in an image are important local features in many computer vision tasks such as 3-D reconstruction, image matching, motion estimation, object tracking, etc. [1]. For instance, [2] proposed a new camera calibration method based on chessboard corners. [3] extracted corners in aerial images of the city for building detection. Some works used corners integrated with optical flow for tracking [4–6].

One of the fundamental types of interest points are corner points. An interest point is a point in an image that has a well-defined position and is invariant to rotation, translation, intensity, and scale changes [7]. Other potential interest points include isolated regions of local intensity maxima or minima, line intersections on edges, and local maximums of curvature on blobs, etc. [8]. Corner points are still not completely defined mathematically. A point that has two dominant and dissimilar edge directions in its immediate vicinity is referred to as a corner. Another definition of a corner is the point where two or more edge curves meet or the local maximum of curvature on the edge contour [9].

Corners are very effective, because they offer a very simple location map and are more accurate than edges and color maps, especially in some tasks like tracking [3-5]. They also significantly reduce the mnoun of dtt hh eeertt ii ning hh imag’’s faauure information, which is vital for real-time tasks. In many cases, the response of a corner detector is a map of candidate locations, and it is necessary to do a local analysis of detected points to identify which of these are true corners [10].

As a fundamental tool for image processing, corner detection has received a lot of attention. In general, there are three groups of corner detectors: intensity-based techniques, contour-based algorithms, and model-based methods.

Intensity-based approaches can directly detect corners by generating a corner response function that assesses image pixel values in such a way that corners

http://dx.doi.org/10.22133/10.22133/ijwr.2024.458246.1217

Citation R. Yazdi, H. Khotanlou, H. Khademfar, " Robust Corner Detection Using Local Extrema Differences", *International Journal of Web Research*, vol.7, no.1,pp.69-84, 2024, doi: http://dx.doi.org /10.22133/ijwr.2024.458246.1217.

*Coressponding Author

Article History: Received: 18 November 2023 ; Revised: 3 January 2024; Accepted: 7 January 2024.

Copyright © 2022 University of Science and Culture. Published by University of Science and Culture. This work is licensed under a Creative Commons Attribution-Noncommercial 4.0 International license(https://creativecommons.org/licenses/by-nc/4.0/). Noncommercial uses of the work are permitted, provided the original work is properly cited.

are recognized as points with low self-similarity in an image [11]. Moravec [12] discovered that there is little difference in intensity between neighboring pixsss aoong an edge or in nn mmage's uniform araa. Meanwhile, there is a significant intensity variation nnall dircciions tt .he corner. To deeermnne a point's self-similarity, we can use the sum of squared differences (SSD) between corresponding image patches from two images. This is the foundation for many corner detectors. Harris and Stephens [13] built on Morvvcc's idea by using th gradient eggnnvll uss of the local auto-correlation matrix in different directions to detect corners. As stated by Noble [14], the reliability of the Harris corner detector is limited to "L-type corners only. Therffore, Sh and Tomiii [15] enhanced the original detector by adjusting the threshold value used to estimate the smallest eigenvalue values. The smallest uni-value segment with an assimilating nucleus (SUSAN) was developed by Smith and Brady [16] to detect the corners of a gray-level image using a gradient convolution of a circle mask known as the USAN region. Currently, the majority of studies have improved upon the tried-and-true Harris [13], SUSAN [16], and Hessian [17] corner-detection techniques.

Contour-based detectors consist of three major steps: edge detection, contour extraction, and corner classification. Wang and Brady [18] introduced a technique for computing image curvature based on edge strength and the rate of change in direction along the edges. Their method involves taking the inverse of the edge strength multiplied by an additional factor related to the curvature at the edge intersection points. Mokhtarian and Suomela [19] proposed a Curvature Scale Space (CSS) corner detector. In these approaches, the authors first use single- or multi-scale Gaussian filters to smooth the curves. They then calculate the curvature of the smoothed curves at each point. For corner detection, the absolute maximum curvature points at single or multiple scales are integrated. It indicates the three primary issues with the current CSS corner detectors. First, noise or local variations in contours can affect the curvature estimator. Second, there is a problem with Gaussian scale selection. The setting of the threshold is the third step. Zhang et al. [20] suggested corner detection based on the angle difference between the directions of anisotropic Gaussian directional derivatives (ANDDs) on curves. A chord length and an angle are the two factors used by Afrin et al. [21] to create an efficient multi-chord corner identification method. In order to evaluate the response of contour points utilizing Manhattan distance and Euclidean distance, Lin et al. [22] introduced two unique corner detectors based on the second-order difference of contour (SODC) distribution features. To identify corners robustly, Liu et al. [23] suggest measuring the linear fitting error rather than estimating the discrete curve directly. They follow a three-step procedure. Initially, a small curve segment is parameterized into two curves. Next, minimum linear fitting errors with respect to these curves are estimated via the least-squares fitting technique. Finally, the obtained errors serve as the local bending strength for corner detection. Forstner [24] identifies the corners in a given window as the points that are closest to all of the tangent lines of the corner. The algorithm is a least-square solution and is based on the assumption that tangent lines intersect at a single location for an ideal corner.

In general, edge detection had a significant impact on how well edge-based corner detectors performed. Also, edge contours close to junctions are frequently unpredictable, making it challenging for edge-based corner detectors to precisely define the junction [8].

Model-based detectors extract corner points by comparing the image patches to a predefined corner model and estimating their similarity. Rosten et al. [25] used a decision tree to improve the performance of corner detection by introducing the FAST (Features from Accelerated Segment Test) algorithm. Shui et al. [26] introduces a novel corner detector and classifier based on anisotropic directional derivative (ANDD) representations. This method characterizes the local directional grayscale variation at each pixel, leveraging both contour and intensity information for improved accuracy. The detection process involves obtaining an edge map using the Canny detector, calculating normalized ANDD representations for pixels on contours, and applying non-maximum suppression and thresholding to identify corners. Additionally, the method includes a classifier to distinguish between simple corners, Y-type corners, and higher-order corners.

Gao et al. [27] focus on template-based methods, which are computationally efficient and straightforward to implement. They introduce two new algorithms designed to enhance robustness in detecting corners, crucial for applications like robot navigation. The paper developed two template-based corner detection algorithms by taking into account optimal corners with at least two pixels in length on the corner arm directions, which reduces the False-Positive corners.

Xing et al. [28], introduces a novel corner detection method that uses a filled circle and outer ring mask. This method first applies an adaptive threshold to distinguish non-corner regions such as image noise, object edges, corner neighborhoods, and flat regions. The detection process then uses a complex response function that combines the margins of the inner filled circle and the outer ring to identify corner candidates.

The development of a universal corner detection model remains unsolved due to its limited coverage

for various image kinds and distinctive scenery attributes. This challenge hinders adaptability and flexibility in corner detection processes across diverse scenarios. Recently, the more powerful and accurate deep learning models employed for corner detection task [29–32].

Zhang et al. [29], introduces a novel deep learning-based algorithm is introduced to detect the corners of quadrilateral objects in real-time. The proposed method leverages convolutional neural networks (CNNs) to extract features and predict corner positions, achieving high accuracy and speed, which is essential for applications like autonomous driving and augmented reality. In [30], Yoon et al. present a deep learning framework for detecting chessboard corners with high precision. The method utilizes multitask learning to simultaneously address corner detection and classification, enhancing the robustness and accuracy of the detection process. By integrating spatial context and geometric constraints into the learning process, the authors achieve significant improvements in detection performance, which is particularly beneficial for applications in camera calibration and 3D reconstruction. Ercan and Wang [31] explore the application of deep learning techniques for precise corner detection in industrial inspection systems. The proposed method combines CNNs with domain-specific knowledge to accurately identify and localize corners in various inspection tasks. The authors validate their approach through rigorous testing on real-world datasets.

Wang et al. [32] proposes a hybrid approach to count wheat ears using Fully Convolutional Networks (FCN) and Harris corner detection. This method leverages the strengths of FCNs in segmenting wheat ears and the precision of Harris corner detection to accurately count them in field conditions. In their work [33], Wang et al. use Faster R-CNN to first locate the target region. Subsequently, they apply an adaptive version of the Harris corner detection algorithm within this region to detect corner points. The repeatability and light robustness of interest point detecting were improved in [30] using feature pyramids, FCN network architectures, cell boundary drift, and metric learning. Reliable feature points were then filtered away by non-maximum suppression.

Although deep learning models show better performance in many areas of image processing tasks, they require access to a vast amount of data to be effective, and the main problem in corner detection tasks is the lack of a large specific dataset.

1.1. Objective and Contributions

Intensity-based methods, in addition to their simplicity and lower computational cost, can easily be integrated into other corner detection methods like deep learning-based detectors. This work introduces a novel and computationally efficient intensity-based corner detector inspired by the Moravec corner criterion [12]. This theory posits that a corner point exhibits significant intensity variations in all directions. We extend this concept by proposing a novel, simplified function that emphasizes the intensity differences at potential corner locations. This function generates a rich corner response map, facilitating the extraction of accurate corner points. Existing intensity-based detectors, such as Harris [13], Shi-Tomasi [15], and FAST [25], rely on eigenvector analysis of image intensity for corner detection.

The main contributions of our method are:

1. A novel corner response function emphasizing intensity variations at potential corner locations.
2. Superior corner localization accuracy and robustness against noise and local intensity variations.
3. Comprehensive performance evaluation against established algorithms (Moravec [12], Harris [13], Shi-Tomasi [15], SUSAN [16], FAST [25], Förstner [24], and Kitchen-Rosenfeld [34]) using benchmark images and diverse scenarios.

We analyze the properties of our proposed corner representation map, which forms the foundation for our novel corner detection method. We compare the performance of our detector with seven established algorithms: Moravec [12], Harris [13], Shi-Tomasi [15], SUSAN [16], FAST [25], Förstner [24], and Kitchen-Rosenfeld [34]. Our evaluation demonstrates that the proposed technique achieves superior corner localization accuracy and robustness against noise and local intensity variations. This assessment includes detection rate and localization error under varying noise levels (including noise-free conditions) using five benchmark images with ground truth corner locations. Additionally, repeatability is evaluated using 28 diverse images under affine transformations, JPEG compression, and noise degradation. The experimental results confirm the superior overall performance of the proposed detector.

The remaining sections of this article are organized as follows: Section II discusses the issues with current intensity computation algorithms, and Section III presents our cutting-edge corner detection method. The experimental results are presented and discussed in Section IV. Finally, Section V presents our conclusions and feature works.

2. Related Works

Intensity-based detectors are the oldest approach to finding interest points. These approaches look for

areas in images that produce significant changes in intensity when slightly altered horizontally or vertically. Thus, detectors like the first or second derivative are suitable for identifying them. A corner detection method based on a differential operator was proposed by Kitchen and Rosenfeld [34] and performed by computing the first and second partial derivatives of an image and determining corners as local maxima. Intensity-bssed approahh''' seoond-order derivatives are noise-sensitive and rarely employed in the literature [35].

Intensity-based detectors are built upon the fact that there is little difference between neighboring pixels along an edge or in a uniform area of the image. Meanwhile, at the corner, the difference is noticeably large in all directions. As a result, the sum of squared differences (SSD) between their associated image patches in the vertical and horizontal directions can be used ooccccatttt point's sfff-similarity. Allow a window to be centered at a position (x, y). The pixel intensity at this location is I(x,y). The intensity of the pixel at this location will be I(x+u,y+v) if this window is slightly shifted to a new location with displacement (u,v). As a result, the difference in intensities of the window shift will be [I(x+u,yvv)−I(x,y)]. Thss differ. nc w b qu... large for a corner, so we look for windows that have a maximum variation. As a result, we maximize this term by differentiating it along the X and Y axes. Let W(x,y) represent the weights of pixels across a rectangular or Gaussian window. Then E(u,v) is defined as Equ(1):

$$E(u,v) = \sum_{(x,y)} W(x,y)\,[I(x+u,y+v) - I(x,y)]^2 \quad (1)$$

Since (1) is very slow, therefore Taylor series expansion, only the first order, is employed Equ(2):

$$T(x,y) \approx f(u,v) + (x-u)f_x(u,v) + (y-v)f_y(u,v) + \cdots \quad (2)$$

So, rewriting the shifted intensity with Equ(2) yields: (Equ(3))

$$I(x+u,y+v) \approx I(x,y) + \frac{\partial I(x,y)}{\partial x}u + \frac{\partial I(x,y)}{\partial y}v \quad (3)$$

On the other hand, $\frac{\partial I(x,y)}{\partial x} = I_x$ and $\frac{\partial I(x,y)}{\partial y} = I_y$ are image derivatives in the X and Y directions respectively, then: (Equ(4-6))

$$E(u,v) = \sum_{(x,y)} W(x,y)\,[I(x,y) + I_x u + I_y v - I(x,y)]^2 \quad (4)$$

$$E(u,v) = \sum_{(x,y)} W(x,y)\,[I_x u + I_y v]^2 \quad (5)$$

$$E(u,v) = \sum_{(x,y)} W(x,y)\,[I^2{}_x u^2 + I^2{}_y v^2 + 2I_x I_y uv] \quad (6)$$

Now in (6), taking *u,v* out and re-writing in matrix notation gives us Equ(7):

$$E(u,v) = (u,v)M\begin{bmatrix} x \\ y \end{bmatrix} \quad (7)$$

and matrix M is Equ(8):

$$M = W(x,y)\begin{bmatrix} \sum_{(x,y)} I^2{}_x & \sum_{(x,y)} I_x I_y \\ \sum_{(x,y)} I_x I_y & \sum_{(x,y)} I^2{}_y \end{bmatrix} \quad (8)$$

The matrix M is a local auto-correlation matrix of intensity variations, so the difference between corner and non-corner pixels is reflected in M. The corner score is derived from the local auto-correlation mtt rix's ww ii genvll uss. aa rrss [13] defined hhis score for classifying into flat region/edge/corner as Equ(9):

$$R = det(M) - K[trace(M)]^2 \quad (9)$$

Where, dtt (M) = $\lambda_1\lambda_2$, and rrcc((M) = $\lambda_1 + \lambda_2$. the λ_1 and λ_2 are eigenvalues of M, and K is an empirical constant value between 0.04 and 0.06. as shown in Figure 1, depending on the value of R, the window is classified as flat, edge, or corner. A high R-value indicates a corner, while a negative R-value indicates an edge.

Th .. rrss deccoor ssn' invrr ian oo scll nnd works best for L-type corners [36]. Its performance also depends on how the corner response suppression is chosen, impacting overall accuracy. Addressing these shortcomings, Shi and Tomasi [15] improved the Harris detector with better corner scoring using minmrH eggenvll ue vll u R min($\lambda_1\lambda_2$), resulting in higher accuracy. If R is greater than a threshold, ssssll sssifeed ss a corner. Addttoonally, thss mtt hod allows us to locate the top N corners, which may be usefu ff w don't wnnttto idenfffy lll corners.

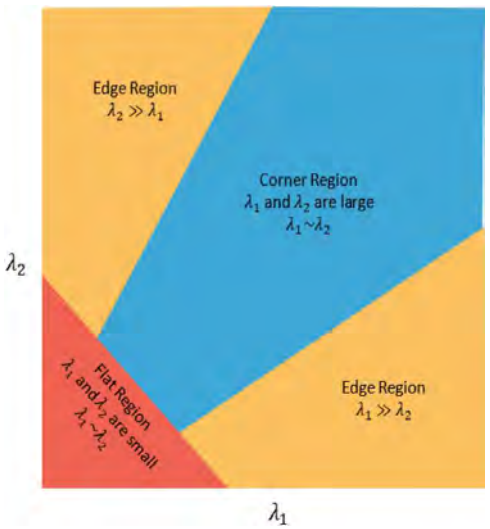

Figure. 1. The Harris Corner Detector detects a corner when both eigenvalues are large.

Foerstner algorithms [24] goal is to find corners defined as crossings of image edges using matrix M as Equ(10-11):

$$Strength = trace(M) \quad (10)$$

$$Roundnes = \frac{trace(M)^2}{4*\det(M)} \quad (11)$$

The roundness of a corner is determined by how similar the gradients that form it are. Only windows having a strength threshold higher than strength and a roundness threshold higher than corner quality are taken into consideration. Then the non-maximum suppression on the window roundness is applied, and the windows candidate for the corner is determined at last. It is accomplished by reducing the square distances, which are weighted by gradient lengths, to all tangent lines within the window that are perpendicular to gradients.

The SUSAN detector proposed by Smith and Brady [16] uses the number of pixels similar to the center pixel in the circular template called USAN, as the corner measure, which fails for some corner types, such as X-shapes. As shown in Figure 2, the USAN area is greatest when the nucleus is in a flat region of the image, but it drops to half of that value very close to a straight edge and even lower when inside a corner. Bae et al. [37] improved the SUSAN detector by replacing the circular mask with a pair of oriented cross operators.

Rosten et al. [25, 38] proposed the FAST (Features from Accelerated Segment Test) algorithm, which first considers an appropriate intensity threshold value. Then it selects candidate interest points, and then it chooses pixel P as the corner based on a circle of 16 pixels around the current candidate pixel. Each pixel in the circle is labeled from 1 to 16 in a clockwise direction, if there is a set of n contiguous pixels in the circle (of 16 pixels) that are all outside of the threshold ± pixel intensity bounds. The case depicted in Figure 3. The FAST corner detector is very suitable for real-time image processing applications because of its high-speed performance. In order to execute the high-speed test for excluding non-corner points, 4 sample pixels 1, 9, 5, and 13 are examined. Since there should be at least 12 contiguous pixels that are either all brighter or all darker than the candidate corner, at least three of these four sample pixels should also be either all brighter or all darker.

All the basic corner detection methods mentioned above received several improvements [39–47]. Gray-scale methods may suffer from noise sensitivity and might not accurately locate the exact corner point position, but they are faster.

3. Proposed Corner Detector

This section provides a comprehensive explanation of our novel intensity-based corner detector. We begin by establishing the theoretical foundation upon which our approach is built. The overall workflow of the proposed corner detection method outlined in Figure 4. We first extract a map highlighting potential corners using our proposed corner detection function. Next, we identify the actual corner points by applying non-maximum suppression to the local maxima in the corner response map. This removes any weak corners and keeps only the most prominent ones.

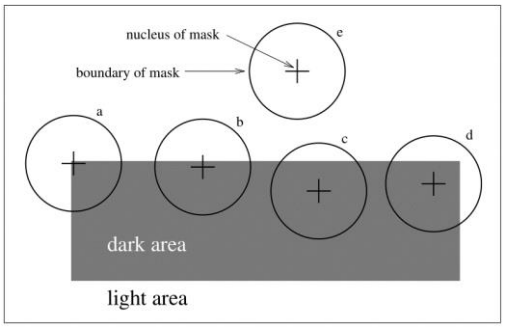

Figure. 2. Showing a dark rectangle on a white background. At five image positions, a circular mask with a center pixel designated as the nucleus is displayed [16].

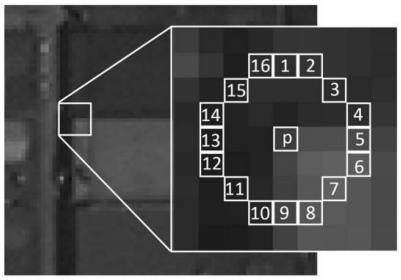

Figure. 3. Corner detection in a 12-point test picture patch using the FAST is displayed [38]. The pixels used for corner detection are shown as highlighted squares. The center of a potential corner is the pixel at position p. The circle is denoted by the line passing through 12 consecutive pixels that are brighter than p in addition to the threshold.

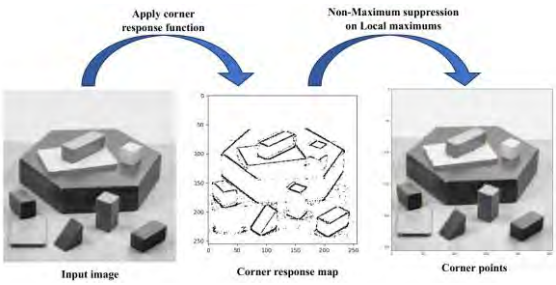

Figure. 4. Corner The overall workflow of our corner detection method. First corner response map extracted using the proposed corner function. Then corner points extracted by applying the non-maximum suppression on Local maximas.

Corner points, as described in [12], exhibit significant variations in intensity across all directions. This characteristic stands in stark contrast to the intensity profiles of neighboring pixels along edges or within uniform regions of an image. Recognizing this principle, researchers have developed various corner detection algorithms, each possessing unique advantages and limitations (e.g., [13, 15, 16, 24, 34, 38]). For instance, the Harris corner detector [13] may exhibit shortcomings in its ability to effectively identify certain corner types compared to the Shi-Tomasi detector [15]. However, the Shi-Tomasi detector might introduce a higher number of false corner detections.

Figure 5 serves as an illustrative example. The image labeled "bedroom light" showcases smooth variations in intensity with distinct corner points. These characteristics make this image particularly suitable for the application of our proposed method.

We applied eight corner detectors to the image and magnified a small region of the image to demonstrate the performance of corner detection methods, as shown in Figure 6. In this region, there are four corner points, one with an angle less than 90 degree, two with angles less than 180, and one with an angle greater than 180. Only one of the corners is detected by FAST [25] (Figure 6(a)), Kitchen-Rosenfeld [34] (Figure 6(c)), and Moravec [12] (Figure 6(d)). Foerstner [24] (Figure 6(b)), Harris [13] (Figure 6(e)), Shi-Tomasi [15] (Figure 6(f)), and SUSAN [16] (Figure 6(g)) all identify two corners, but our method surprisingly identified three of four (Figure 6(h)).

Furthermore, most of them have very high false corners. Because, in addition to the unwanted hidden noise in the image, the border regions do not always have homogeneous values at the pixel level. Also, the corner response function of some methods, like Kitchen-Rosenfeld [34] Figure 6(c) and Moravec [12] Figure 6(d), are unstable to small changes in intensity levels.

In Figure 7, the corner response maps of the fforemeniioned mtt hods ar shonn . It's essy to see why there are so many false corners in some methods, so we need a robust corner response function, while the proposed corner response function generates a very clear corner response map.

Figure. 5. The intensities in the bedroom light image change smoothly in most areas.

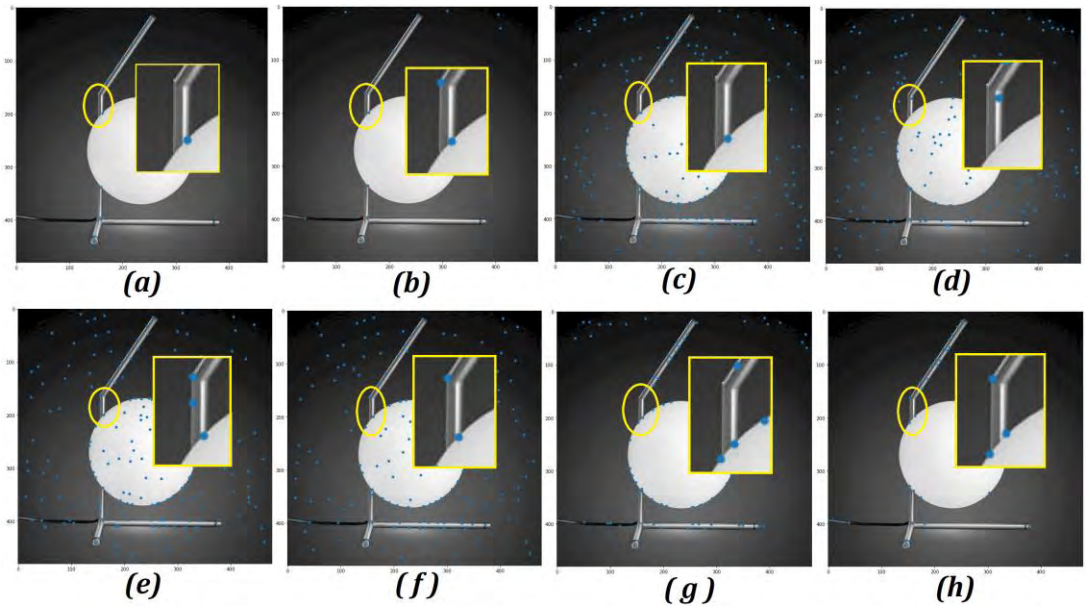

Figure. 6. The result of eight corner detectors on Figure 5. (a) FAST[25], (b) Forstner[24], (c) Kitchen-Rosenfeld[34], (d) Moravec[12], (e) Harris[13], (f) Shi-Tomasi[15], (g) SUSAN[16], and the proposed detector (h).

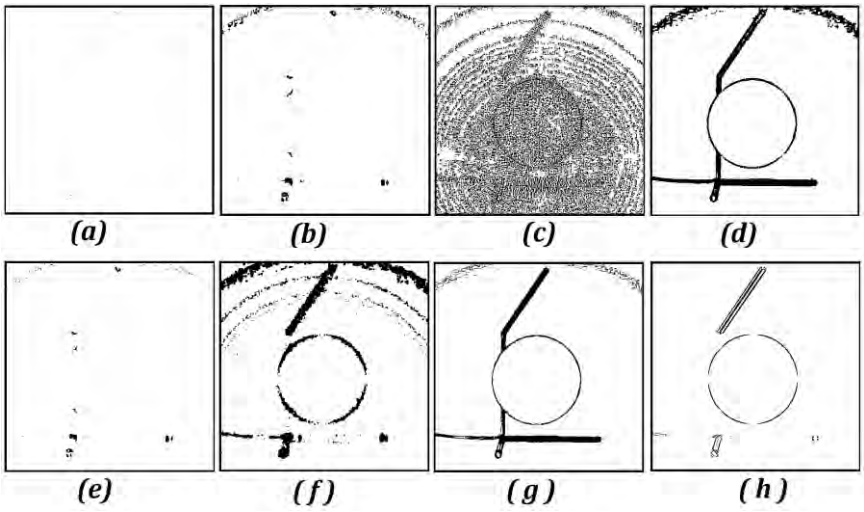

Figure. 7. The corner response map of eight corner detectors on Figure 5. (a) FAST[25], (b) Forstner[24], (c) Kitchen-Rosenfeld[34], (d) Moravec[12], (e) Harris[13], (f) Shi-Tomasi[15], (g) SUSAN[16], and the proposed detector (h).

Since corners are defined by sharp changes in intensity from all directions within a small area, we introduce a new function to identify these regions. This function works by transforming the image's intensity values into a new domain where corners have higher values. As illustrated in Figure 8, for each pixel in the image, we consider its surrounding 3×3 neighborhood and sort the pixel intensities within it. Then, we calculate the difference between the sum of the three brightest and the three darkest pixels. This difference forms the heart of our proposed corner response function.

Figure 8(b) illustrates a key aspect of our corner detection function. If the intensity values of pixels p_7, p_8, and p_9 in the image window are similar, then the absolute difference '*u*' calculated in the function (and similarly '*d*' in Algorithm 1) will be small. This small difference reflects the similarity of intensity values within a non-corner region. Conversely, when the window encompasses a corner, the intensity values will exhibit significant variation. In such cases, relying solely on the maximum and minimum values within the window might be susceptible to noise or unwanted variations in pixel intensities. To address this issue and achieve greater generalization, we propose calculating the difference between the intensity of the upper third of the sorted pixel values and the intensity of the lower third. This approach ensures that even if pixels p_8 and p_9 have similar intensities, the significant difference between p_7 and p_9 can still contribute to the corner detection. Notably, by sorting the pixels based solely on intensity and disregarding their spatial location, we effectively compute the maximum intensity difference across various directions within the window. This strategy enhances the generalization and robustness of the corner definition compared to Moravec's approach [12]. Furthermore, as detailed in Algorithm 1, the

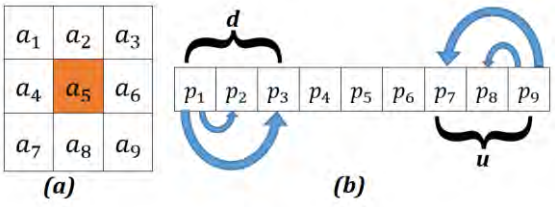

Figure. 8. The proposed corner response function. (a) The 3*3 kernel. (b) The kernel sorted incrementally to get extrema differences.

Algorithm. 1. Proposed Corner Response Function

Input: Gray Scale Image
Output: Corner Response Map
for everyPixel **in** inputImage **do**
1. W = geeee gghboorhood 3 ∗ 3(pixel)
2. p = sortIncremental(W)
3. u = (p_9 p_8) ∗ (p_9 ppp$_7$)
4. d = (p_1 p_2) ∗ (p_1 ppp$_3$)
5. cornerResponseMap[pixel] = u^2 ddd^2

end for
return cornerResponseMap

proposed method is computationally efficient, making it suitable for real-time applications.

Following the generation of the corner response map, we proceed to extract the most prominent corner candidates as the final results. Similar to other corner detection methods, we employ non-maximum suppression with a predefined window size. This window size parameter plays a crucial role in determining the corner resolution, which essentially refers to the minimum separation distance between two detectable corners. It means how precisely two adjacent corners can be distinguished. While it's true that the novel corner point definition we introduced

deviates slightly from the traditional definition, the effectiveness of our approach will be demonstrably validated in the subsequent section.

4. Experiments and Results

The results of a detailed performance evaluation of the suggested corner detector are reported in this section. First, the proposed method is compared to seven basic corner detectors in terms of the number of missed and false corners as well as the localization errors of corners that were correctly recognized, using five images with ground truths. Then, the Average Repeatability (AR) of the detectors during noise degradation, JPEG compression, and image affine transformations is assessed using 28 images.

4.1. Datasets

For corner detection performance evaluation, we used three free noise test images from well-known repositories and two natural test images from the internet containing various scenes and noises. Our GitHub provides access to all codes and data. Figure 9 depicts the ground truths for the five test images.

Th ’Texture’ mnge Figure 9(a) has 33 ground truth corner points, and unlike the other images, its oorners have diffrr en shapss. Th ’Block’ image Figure 9(b) oontii ns 60 cornrr poinss, the ’Boxss’ image Figure 9()) h4446 ll aar oorners, th ’Wooden fenc’’ image Figure 9(d) contains 70 corners, and the ’Bcckyard fnnee’ Figure 9(e) has 69 corner points. For evaluating the AR of our proposed corner detector, we employed 28 images (Figure 10) which gathered from the MSCOCO [48] dataset and [49].

4.2. Average localization errors

According to the [9], the DC = (x_i,y_i), i = 1,2,...M1 and GT = (x_j,y_j), j = 1,2,...M2 represents, respectively, the corners that a corner detector has correctly identified and the true corners in the ground truth images. The shortest distance from set DC is determined for a corner (x_j,y_j) in set GT. The corner (x_j,y_j) is considered to have been appropriately recognized if the shortest distance does not exceed the predefined hlrsshold (here, hlrhhhold 4). Therefore, the detected corner in set DC and the corner (x_j,y_j) in set GT constitute a matched pair. Otherwise, the (x_j,y_j) corner marks as a missed corner. Similarly, we determine the minimum distance from set GT for a corner (x_i,y_i) in set DC. A corner (x_i,y_i) is regarded as a false corner if the minimal distance exceeds the threshold.

The average distance between all matched pairs is used to calculate the localization error as Equ(12):

$$AL_e = \sqrt{\frac{1}{N}\sum_{K=1}^{N}[(\hat{x}_k - x_k)^2 + (\hat{y}_k - y_k)^2]} \qquad (12)$$

The k = 1, 2, ..., N in $AL_e(\hat{x}_k,\hat{y}_k),(x_k,y_k)$ are the matched corner pairs in sets GT and DC. For the other seven detectors, Moravec [12], Harris [13], Shi-Tomasi [15], SUSAN [16], FAST [25], Foerstner [24], and Kitchen-Rosenfeld [34], we used the Scikit-image library in Python. The corner resolution search nnndo siee nnthss paper ss α 7, so seiiing = 4 ss oorrcc ttt ually, th α wss used for loaa pekk selection as initial corner candidates. Also, all of the dtt ccoors’ uunbbee parameeers rr stt to therr default values. For our detector, there is no specific parameter except one for thresholding corner response map, whose default value is 0.001, however we demonstrate the effect of this single parameter on our detector performance in Figure 11.

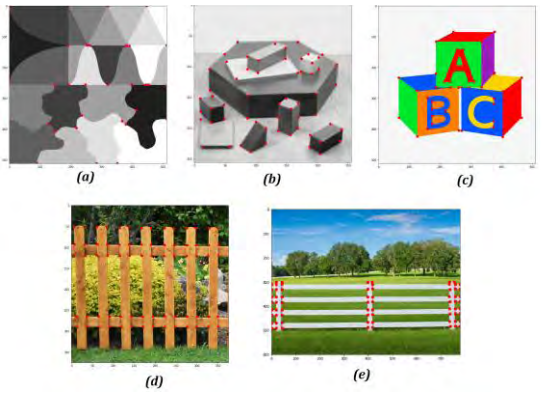

Figure. 9. The five test images with red dots in their true corner ’ee xture’ (a), ’. lock’ (b), ’Boxe’’ (c), ’Wooden fence’ (d), ’Backyard fence’ (e).

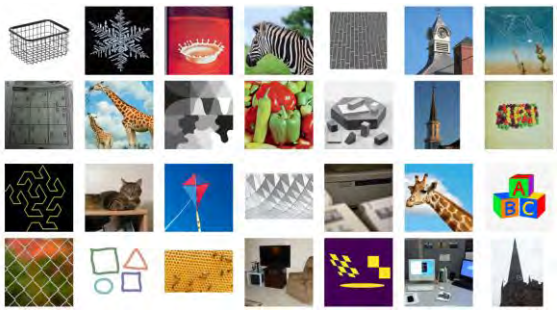

Figure. 10. Twenty-eight test images from various scenes.

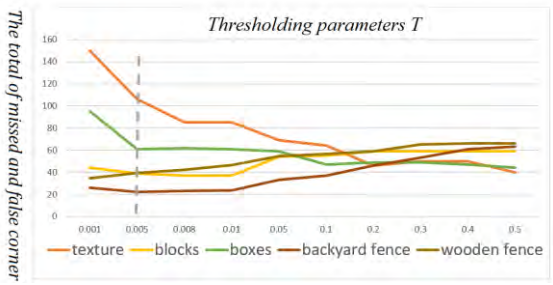

Figure. 11. The impact of adjusting only the thresholding parameters T on the corner detection algorithm.

Because the first three images are noise-free, we consider missed and false corners to have the same effect on detector loss. However, because two fence images are natural and contain noise, they have a higher potential for false corners, so we assign a higher weight on loss to missed corners in these two. It can be seen that the proposed detector with threshold T = 0.005 has the highest overall decrease ratio in terms of missed and false corners.

4.3. Average Repeatability

Since the number of test images with ground truths is limited, evaluating the performance of corner detectors solely on these images is insufficient. For the evaluation of corner detectors, the average repeatability under affine transformations, JPEG compression, and noise degradation was proposed [9, 20, 26]. The average repeatability (13) counts the number of corner points that are consistently detected in the same location between the original and transformed images. Actually, it measures the geometrical stability of the detected corners between the original and transformed images, so a better performance is indicated by a higher average repeatability. Furthermore, because the ground truth of images is not required for the computation of repeatability, evaluation can be done on a large number of images; with this approach, our dataset images exceed 10000.

$$AR = \frac{N_r}{2}\left(\frac{1}{N_o} + \frac{1}{N_t}\right) \quad (13)$$

Where N_o and N_t denote the number of detected corners in the original and transformed images, respectively, and N_r denotes the number of corners that are repeated between them within a threshold δ pixels.

4.4. Detection Evaluation with Ground Truth Images

First, we use five ground truth images as shown in Figure 9 to compare the eight detectors. Figures 12, 13, 14, 15 and 16 show the results of the eight detectors. Table 1 contains the localization error, the number of missed corners, and the number of false corners for each detector. We use the total number of missed and false corners of test images to compare detector performance in Figure 17. The ratio of the total number of missed and false corners in ground truth to the total number of true corners, according to [9], can be used to quantitatively evaluate detection performance. Furthermore, in Table 2, we calculate the F-SCORE measure of detectors for a fair performance comparison.(Equ(14-16))

$$Precision = \frac{TrueConers}{TrueCorners + FalseCorners} \quad (14)$$

$$Recall = \frac{TrueConers}{TrueCorners + MissedCorners} \quad (15)$$

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} \quad (16)$$

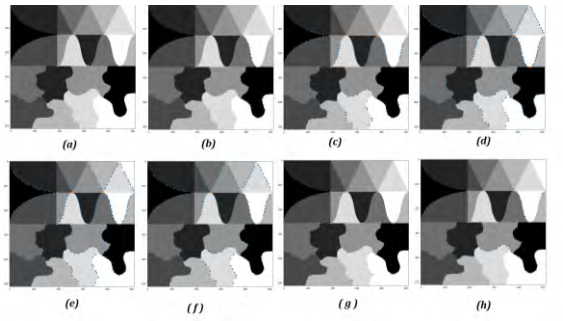

Figure. 12. Detection results on the test image Texture. (a) FAST[25], (b) Forstner[24], (c) Harris[13], (d) Kitchen-Rosenfeld[34], (e) Moravec[12], (f) Shi-Tomasi[15], (g) SUSAN[16], and (h) Proposed.

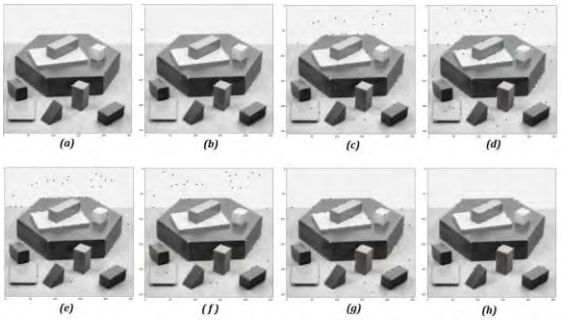

Figure. 13. Detection results on the test image Blocks. (a) FAST[25], (b) Forstner[24], (c) Harris[13], (d) Kitchen-Rosenfeld[34], (e) Moravec[12], (f) Shi-Tomasi[15], (g) SUSAN[16], and (h) Proposed.

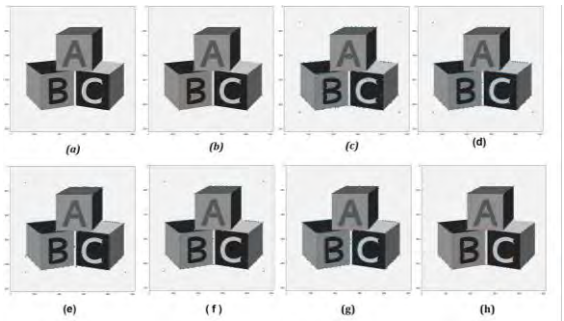

Figure. 14. Detection results on the test image Boxes. (a) FAST[25], (b) Forstner[24], (c) Harris[13], (d) Kitchen-Rosenfeld[34], (e) Moravec[12], (f) Shi-Tomasi[15], (g) SUSAN[16], and (h) Proposed.

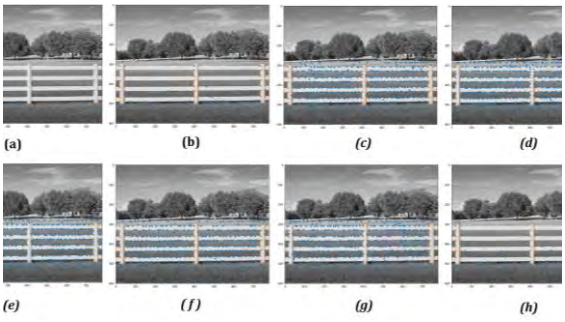

Figure. 15. Detection results on the test image Backyard fence. (a) FAST[25], (b) Forstner[24], (c) Harris[13], (d) Kitchen-Rosenfeld[34], (e) Moravec[12], (f) Shi-Tomasi[15], (g) SUSAN[16], and (h) Proposed.

Table 1. The comparison of the eight detectors' performance for test images which is measured in pixels.

Detector	*Missed conness (FN)*					*Falee conne(((F))*					*oo calization error*				
	Teeeeee	eeeeee	eeeee	e nnnnn nnnnn	nnnnnnnn eeeee	eerrrr e	Bssssss	s eees	Weeee n fecce	Bacaaar fecce	Terrrr e	Bssssss	s eees	Weeee n fecce	Bacaaar fecce
Moravec [.]	33	00	44	00	66	555	44	222	444	555	.. 11	.. 333	.. 999	.. 111	.. 444
Forttner [21]	44	33	66	99	33	00	0	0	2	66	.. 8	.. 777	.. 999	.. 2222	.. 888
Kitchen-Rosenfeld [32]	33	7	7	00	55	777	55	000	000	555	.. 111	.. 777	.. 222	.. 444	.. 444
Harri1110]	55	00	3	11	44	777	99	666	444	222	.. 11	.. 777	.. 111	.. 777	.. 88
Shi-Tosss i [12]	77	6	5	11	00	111	99	155	999	777	.. 555	.. 111	.. 44	.. 333	.. 222
FAST [23]	77	22	55	33	55	1	0	0	11	122	.. 111	.. 555	.. 555	.. 777	.. 666
SUSAN [13]	55	22	33	44	66	99	11	666	666	666	.. 1	.. 888	.. 777	.. 777	.. 88
Propoeed ttt hod	33	66	55	99	99	444	99	22	444	777	.. 555	.. 777	.. 222	.. 555	.. 888

Table 2. F-SCORE, localization error and ratio of them of true corners

Detectoss	*F-SCORE*	*oo calization Eooar*	*F-SCORE/Localization error*
Moravec	0.219	1.74	0.126
Foerttner	0.299	**1.23**	0.243
Kitchen-Rosenfeld	0.284	1.38	0.206
Harris	0.337	1.36	0.248
Shi-Tosss i	0.35	1.28	**0.273**
FAST	0.263	1.69	0.156
SUSAN	0.316	1.44	0.219
Propoeed Method	**0.366**	1.41	0.260

As can be seen, the proposed detector achieves the highest F-SCORE, meaning the best detection performance, in the images. It is worth noticing that two images are not free of noise. Furthermore, in Table 2, corner localization error is an important criterion for evaluating detectors, and while our method's performance on the localization error criterion falls within the average range, it excels in efficiency. In fact, it achieved the second-highest ratio of detection performance to localization error.

4.5. Average Repeatability Performance under Affine transformation

The Average Repeatability (AR) measures detected corner points in the same position between the original and transformed images. For evaluating the AR of our proposed corner detector, we applied the following six different transformations on each original image of Figure 10 images:

- Rotations: The original image was rotated at 18 different nnglss ii hlnn $[-\pi/2, \pi/2]$ at 10 degrees apart.
- Uniform scaling: The original image was scaled with scale factors $S_x = S_y$ in [0.5, 2] with 0.1 apart, excluding 1.

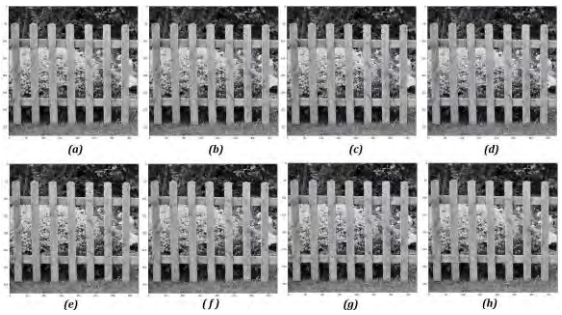

Figure. 16. Detection results on the test image Wooden fence. (a) FAST[25], (b) Forstner[24], (c) Harris[13], (d) Kitchen-Rosenfeld[34], (e) Moravec[12], (f) Shi-Tomasi[15], (g) SUSAN[16], and (h) Proposed.

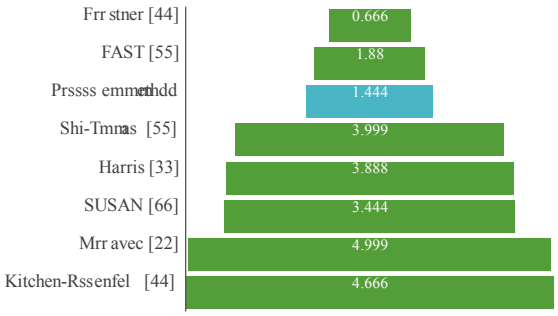

Figure. 17. The ratio of false and missed corners to true corners.

- Non-uniform scaling: The scales S_x chosen by sampling the ranges [0.7, 1.5] and S_y from [0.5, 1.3] with a 0.1 interval.
- Lossy JPEG compression: The original input image is compressed by a compression factor within [5, 100], and the interval is 5.
- Gaussian noise: A white Gaussian noise when the mean is zero and the variances are in [1, 15] with an interval of 1 is added to the original input image.

- hherr rmnsformaiions: The shaar fcctor ’C’ ss in [-1, 1], the interval is 0.1, not including 1, and the transformation rule is Equ(17):

$$\begin{bmatrix} x_{new} \\ y_{new} \end{bmatrix} = \begin{bmatrix} 1 & C \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \quad (17)$$

For this measurement, we first detect the corners of the original image, then transform the image and extract new corners. The original corner points are then transformed, and matched points are collected. Figure 18 depicts the results of AR transformations of eight corner detectors under Gaussian noises, lossy JPEG compression, uniform scaling, non-uniform scaling, rotation, and shear transformation.

Furthermore, the proposed corner detector conforms to previously validated works [9, 20, 26, 50].

According to Figure 19, our method not only does not fail in the transformations, but it also has a higher mean average repeatability rate in the Gaussian noise, uniform scaling, and non-uniform scaling. The proposed approach performs third best when lossy JPEG compression and shear transformation are applied. Additionally, our technique performs comparably to others in rotation transformations.

4.6. Average Repeatability under Pre-processing

Pre-processing refers to fundamental manipulations applied to raw image data before any further processing takes place. Pre-processing aims to enhance image content by removing undesirable noises or modifying certain aspects of the image that are important for specific processing and analysis. In addition to providing clean image data, pre-processing

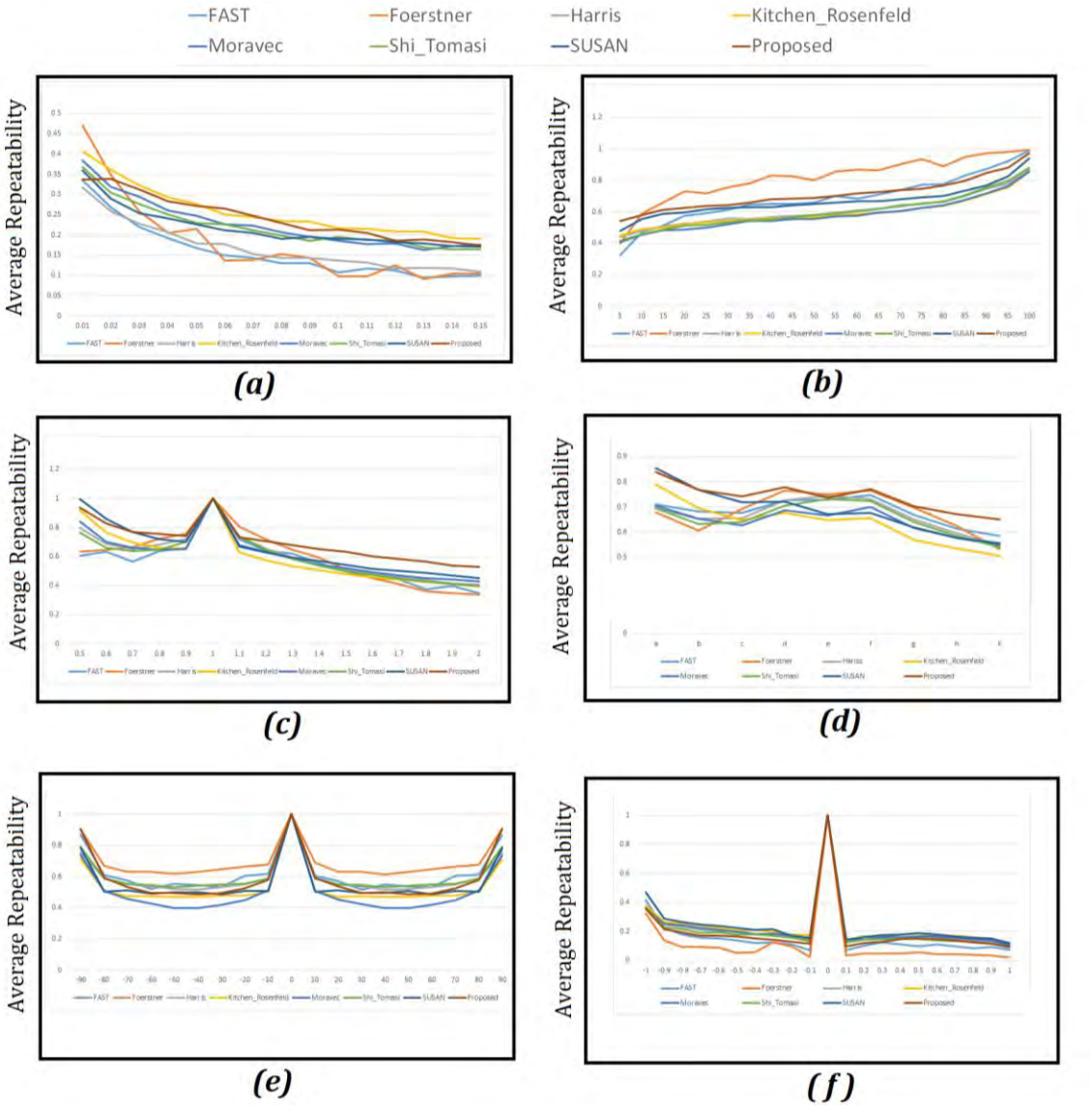

Figure. 18. The Average Repeatability performance results of eight corner detectors under Gaussian noises (a), lossy JPEG compression (b), uniform scaling (c), non-uniform scaling (d), rotation (e), and shear transformation (f).

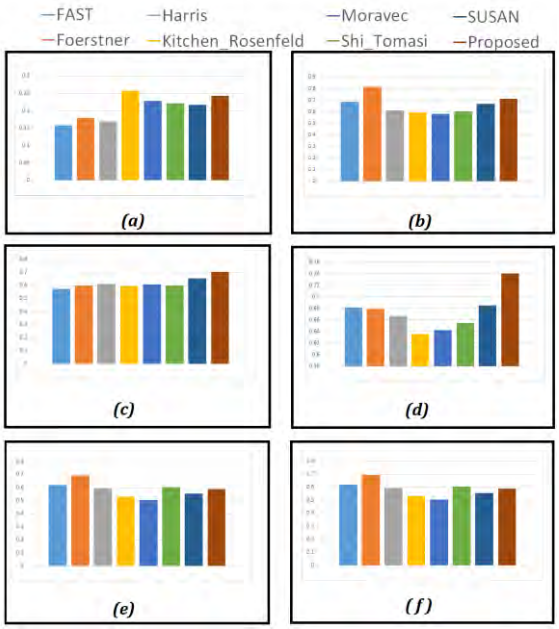

Figure. 19. The Mean Average Repeatability performance of eight corner detectors under (a) Gaussian noises, (b) lossy JPEG compression, (c) uniform scaling, (d) non-uniform scaling, (e) rotation, and (f) shear transformation.

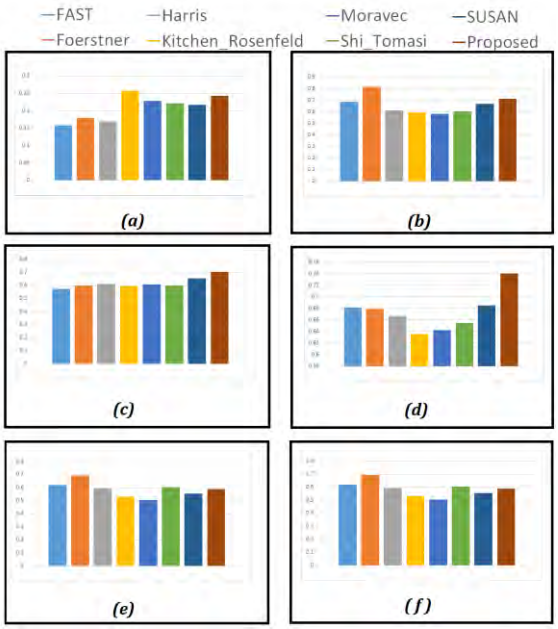

Figure. 20. The Mean Average Repeatability performance of eight corner detectors with pre-processing under (a) Gaussian noises, (b) lossy JPEG compression, (c) uniform scaling, (d) non-uniform scaling, (e) rotation, and (f) shear transformation.

reduces model training time and increases model inference speed. We investigate the effect of two pre-processing steps on corner detectors because pre-processing is an important step in real-world machine vision applications. Because Gaussian blurring and contrast stretching are the most commonly used pre-processing techniques, we first apply Gaussian blurring and then stretch the contrast of images. Table 3 shows that the proposed corner detector is compatible with these pre-processing methods while still outperforming them in terms of localization error and F-SCORE measure. According to Figure 20, with the exception of the shear transformation, the proposed detector has acceptable AR performance in the aforementioned transformations.

Table 3. F-SCORE, localization error and ratio of them of the true corners with pre-processing.

Detectors	F-SCORE	oo calization rr ror	F-Score/Localization error
Moravec	0.196	1.668	0.118
Foerttner	0.437	1.354	0.323
Kitchen-Roeenfeld	0.285	1.386	0.206
Harris	0.425	1.356	0.313
Shi-Tosss i	0.356	1.352	0.263
FAST	0.430	1.694	0.254
SUSAN	0.288	1.411	0.204
Propoeed Method	**0.446**	**1.316**	**0.339**

5. Discussion

While a universally accepted and comprehensive mathematical definition for corner points remains elusive, they are generally understood as image locations where the intensity variations between neighboring pixels reach a maximum simultaneously in all directions. The novel definition presented in this paper similarly identifies regions with significant intensity differences, assigning higher response values to such areas.

However, unlike existing intensity-based corner detectors that rely on first or second-order derivatives [35], our proposed method employs a computationally simpler function. This function operates by considering a 3×3 kernel centered on each image pixel. The pixel intensities within this window are then sorted in ascending order. Finally, the squared difference between these sorted values is assigned as the final response value for the central pixel. Consequently, the proposed corner response function directly amplifies the intensity variations within the 3×3 window. It is important to acknowledge that employing such a small window introduces a slight spatial error, as neighboring pixel values are used to compute the intensity values in the response map. However, it's important to note that using kernel sizes 5 and 7 for this step can significantly degrade the quality of the corner response map. These larger kernels tend to introduce distortions and generate unreliable information, which cannot lead to true corners.

While our proposed method is effective for corner detection, it is important to note that using a small kernel size can limit its ability to handle impulsive noise, such as salt and pepper noise. To address this, we applied blurring and contrast stretching as pre-processing steps for noise removal before using our corner detection method. The results confirm that our corner detector works well with these pre-processing techniques, outperforming them in terms of localization error and F-SCORE. This presents an opportunity for further exploration, possibly incorporating additional pre-processing steps to improve corner detection performance.

Despite the simplicity of the proposed function, the resulting corner response map exhibits excellent detection accuracy and resolution. While derivative-based methods utilize the relative positions of pixels to compute response map values, our method discards this spatial information during the sorting step. This allows for the effective utilization of neighboring pixel information, leading to successful corner detection even in noisy images. Figure 21 compares the F-SCORE to localization error ratio of the proposed method and other intensity-based methods, both with and without pre-processing. The proposed method demonstrates significantly superior overall performance in this metric.

Furthermore, compared to other intensity-based corner detectors, the proposed method offers a significant advantage in terms of parameter efficiency. Our approach requires only a single parameter to configure the algorithm, whereas methods like Forstner [24] and SUSAN [16] necessitate the setting of multiple parameters, and selecting appropriate values for these parameters can significantly impact their performance.

Unfortunately, due to the use of seven standard library corner detectors for performance comparison, a direct comparison of our algorithm's execution time was not feasible. However, as evident from Algorithm 1, the proposed method boasts a low computational cost due to its inherent simplicity.

Finally, the generated corner response map presents the exciting possibility of fusion with response maps produced by other corner detection techniques, such as curvature-based methods [51-52]. This map also holds promise as a feature map within successful deep learning architectures. We envision leveraging this response map as an input for established detection networks like YOLO [50].

6. Conclusion

This paper introduces a novel method for corner detection that leverages a newly developed corner response function. This function offers several key advantages. Firstly, it is remarkably simple, facilitating efficient computation with real-time potential. Secondly, it generates a rich corner response map with high corner resolution. Notably, our method eliminates the need for pre-processing steps beyond grayscale conversion.

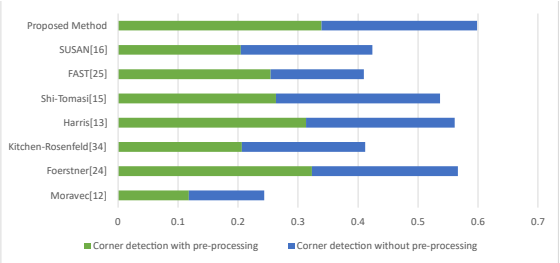

Figure. 21. The F-SCORE to localization error ratio of the proposed method and other intensity-based methods.

Experimental evaluations demonstrate the superiority of our proposed framework compared to seven established corner detectors. Our method achieves in overall more than 3% lower rates of missed corners and incorrect detections while exhibiting superior accuracy in corner localization. Additionally, the proposed method remains compatible with pre-processing tools like contrast stretching and Gaussian blurring, which can further enhance performance in many scenarios.

Looking towards future applications, the generated corner response map presents exciting possibilities. It can be fused with response maps obtained from other corner or edge detection methods. Furthermore, this map holds significant promise as a feature map within successful deep learning architectures. We envision employing this response map as an input to established deep learning models for key-point detection.

Declarations

Funding

This research did not receive any grant from funding agencies in the public, commercial, or non-profit sectors.

Authors' contributions

During the preparation of this work, the authors used Gemini in order to enhance text clarity and writing. After using this tool, the authors reviewed and edited the content as needed. The authors take full responsibility for the content of the publication.

RY: Study design, acquisition of data, interpretation of the results, statistical analysis, drafting the manuscript;

HoK: Study design, interpretation of the results, drafting the manuscript, revision of the manuscript;

HaK: Supervision, drafting the manuscript, revision of the manuscript.

Conflict of interest

The authors declare that no conflicts of interest exist.

References

[1] M. Wang, W. Zhang, C. Sun and A. Sowyy a, A., "Corner detection based on shearlet transform and multi-directional tt ruc.ure tenoor," *Pattern Recognit*, vol. 103, p. 107299, 2020. https://doi.org/10.1016/j.patcog.2020.107299

[2] Y. hh ang, G. ii , X. Xie, .. Wang, "A new algorithm for accurate and automatic chessboard corner detection," In *2017 IEEE International Symposium on Circuits and Systems (ISCAS)*, Baltimore, MD, USA, 2017, pp. 1–4 ,(2017). https://doi.org/10.1109/ISCAS.2017.8050637

[3] W. Song, B. hh ong and X. Sun, "Building corner detection in aerial iaa ges with fully convolutional network"""" *Sensors*, vol. 19, no. 8, p. 1915, 2019. https: //doi.org/10.3390/s19081915

[4] H. hh ang, L. Xiao, G. Xu, "A novel tracking ee thod baeed on improved fast corner detection and pyramid lk optical flow," In *2020 Chinese Control And Decision Conference (CCDC)*, Hefei, China, 2020, pp. 1871–1876. https://doi.org/10.1109/ CCDC49329.2020.9164332

[5] .. Wang and X. Yang, "Moving target detection and tracking based on pyraii d lucaskanade optical flow," In *2018 IEEE 3rd International Conference on Image, Vision and Computing (ICIVC)*, Chongqing, China, 2018, pp. 66–69. https://doi.org/10.1109/ICIVC.2018. 8492786

[6] H. Song and M. I. Shen, "aa rget tracking algorith based on optical flow ttt hod uii ng corner detection," *Multimedia Tools and Applications*, vol. 52, pp. 121–131, 2011. https://doi.org/10.1007/s11042-010-0464-8

[7] S. M. M. Kahaki, M. J. Nordin and A. H. Ahhtari, "Contour-based corner detection and classification by using mean projection tranffor,, " Sensor,, vol. 14, no. 3, pp. 4126–4143, 2014. https://doi.org/10.3390/s140304126

[8] H. Yarui, .. Yunhong and F. Qiaochu, "A survey of image corner detection methods," In Proceeding of the 3rd International Conference on High Performance Compilation, Copp uting and Communication.. HP3C '19, Association for Computing Machinery, New York, NY, USA, 2019, pp. 123–127. https://doi.org/10. 1145/3318265.3318295

[9] W. Zhang, C. Sun, T. Breckon and N. Alshammari, "Dieeeete curvature repreeentations for noiee robutt image corner detection," *IEEE Transactions on Image Processing*, vol. 28, no. 9, pp. 4444–4459, 2019. https://doi.org/10.1109/TIP.2019.2910655

[10] Bao, Jing, W. hh ang, C. Liu and .. Gao, "A corner detection method based on adaptive multi-directional aniootropic diffusion" *Multim. Tools Appl.*, vol. 81, pp. 28729–28754, 2022. https://doi.org/10.1007/s11042-022-12666-w

[11] Wang and W. hh ang, "A survey of corner detection ttt hod,," In *Proceedings of the 2018 2nd International Conference on Electrical Engineering and Automation, ICEEA 2018*, 2018, pp. 214–219. https://doi.org/10.2991/iceea-18.2018.47

[12] H. P. Moravec, "Toward automatic viuual obstacle avoidance," In *Proceedings of the 5th International Joint Conference on Artificial Intelligence*, Cambridge, MA, USA, 1977, p. 584. https://dl.acm.org/doi/abs/10.5555/1622943.1622947

[13] C. G. Harri and M. J. Stephens, "A combined corner and edge detector," In *Proceedings of the 4th Alvey Vision Conference*, 1988.

[14] A. Noble, "Finding corner,," In *Proceedings of the Alvey Vision Conference*, AVC 1987, Cambridge, UK, September, 1987, pp. 1–8. https://doi.org/10. 5244/C.1.37

[15] Shi, "Good feature to track," In *1994 Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, 1994, pp. 593–600. https://doi.org/10.1109/CVPR.1994.323794

[16] S. M. Smith and M., Brady, "Suaan—a new approach to low level image processing," *International Journal of Computer Vision*, vol. 23, pp. 45–78, 1997. https://doi.org/10. 1023/A:1007963824710

[17] D. Chen and G. hh ang, "A new sub-pixel detector for x-corner in camera calibration target"""" In *International Conference in Central Europe on Computer Graphics and Visualization*, 2005, pp. 97–100. http://hdl.handle.net/11025/11219

[18] H. Wang and M. Brady, "Real-time corner detection algorith for oo tion estimation," *Image Vis. Comput.*, vol. 13, no. 9, pp. 695–703, 1995. https://doi.org/10.1016/ 0262-8856(95)98864-P

[19] F. Mokhtarian and R. Suoee la, "Robutt iaa ge corner detection through curvature aaale space," I rr ansactions on Pattern Analysis and Machine Intelligence, vol. 20, no. 12, pp. 1376–1381, 1998. https://doi.org/10.1109/34.735812

[20] W. Zhang and P. Shui, "Contour-based corner detection via angle difference of principal directions of anisotropic gauiii an directional derivatives," Pattern Recognit.,.vol. 48, no. 9, pp. 2785–2797, 2015. https://doi.org/10.1016/j.patcog.2015.03.021

[21] N. Afrin, N. Mohammed and W. aa i, "An ef.ective multi-chord corner detection technique.,"In *2016 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, Gold Coast, QLD, Australia, 2016, pp. 1–8. https://doi.org/10.1109/DICTA. 2016.7797005

[22] X. ii n, C. Zhu, Q. Zhang, . . Huang and Y. Liu, "ff ficient and robust corner detectors based on second-order difference of contour," *IEEE Signal Processing Letters*, vol. 24, no. 9, pp. 1393–1397, 2017. https://doi.org/10.1109/LSP.2017.2724851

[23] S. Liu, Q. Zheng, Y. Li, Z. Zhang, B. Li, B. and S. Zhang, "Robutt corner detection uii ng linear fitting error ettittt ion," *In ICNCC 2020: The 9th International Conference on Networks, Communication and Computing*, Tokyo, Japan, 2020, pp. 87–95. https://doi.org/10.1145/3447654.3447667

[24] W. Förstner and E. Gülch, "A fatt operator for detection and precise location of distinct points, corners and centres of circular feature"""" In *Proceedings of the ISPRS Intercommission Confe-rence on Fast Processing of Phonogrammic Data*, 1987, pp. 281–305, 1987. https://cseweb.ucsd.edu/classes/sp02/cse252/ foerstner/foerstner.pdf

[25] .. Rosten, R. Porter and .. Drummond, "Fatt er and better: A ccc hine learning approach to corner detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 1, pp. 105–119, 2010. https://doi.org/10.1109/TPAMI. 2008.275

[26] P. Shui and W. Zhang, "Corner detection and claiii fication uii ng aniootropic directional derivative repreeentation,," *IEEE Transactions on Image Processing*, vol. 22, pp. 3204–3218, 2013. https://doi.org/10.1109/TIP.2013.2259834

[27] C. Gao, K. Pa. etta a. d S. Agaian, "Robutt template based corner. d. tection algorith for robotic vision," In *2015 IEEE International Conference on Technologies for*

Practical Robot Applications (TePRA), Woburn, MA, USA, 2015, pp. 1–6. https://doi.org/10.1109/TePRA.2015.7219683

[28] Y. Xing, D. hhang, .. Zhao, M. Sun and W. Jia, "Robust fatt corner detector baeed on filled circle and outer rin. sss k," *IET Image Process*, vol. 10, no. 4, pp. 314–324, pp. 2016. https://doi.org/10.1049/iet-ipr.2014.0952

[29] Zhang, .. Jiao, X. An and Y. He, Y. "Real-time quadrilateral object corner detection algorithm based on deep learn.n. ," In *2019 Computing, Communications and IoT Applications (ComComAp)*, Shenzhen, China, 2019, pp. 70–75. https://doi.org/10.1109/ComComAp46287.2019.9018794

[30] H. Yoon, S. ee e, J. Kang and . ee e, "Deep chessboard corner detection uiing multitask learning," In *2021 IEEE 23rd International Workshop on Multimedia Signal Processing (MMSP)*, Tampere, Finland, 2021, pp. 1–6. https://doi.org/10.1109/MMSP53017.2021. 9733619

[31] M. F. rr can and R. B. Wang, "Deep learning for accurate corner detection in computer vision-based inspection," In *Computational Science and Its Applications ICCSA 2021 - 21st International Conference, Cagliari, Italy, September 13-16, 2021, Proceedings, Part II. Lecture Notes in Computer Science*, vol. 12950, 2021, pp. 45–54. https://doi.org/10.1007/978-3-030-86960-1 4

[32] D. Wang *et al*., "Combined use of fcn and harris corner detection for counting wheat ears in field c. nditions," *IEEE Access*, vol. 7, pp. 178930–178941, 2019. https://doi.org/10.1109/ACCESS.2019.2958831

[33] .. Wang, K. Han and H. Sun,."An adaptive corn. r.detection ttt hod based on deep learning," In *2019 Chinese Control Conference (CCC)*, Guangzhou, China, 2019, pp. 8478–8482. https://doi.org/10.23919/ChiCC.2019.8866560

[34] .. J. Kitchen and A. Roeenfeld, "Gray-level corner d. .ection," *Pattern Recognit. Lett*., vol. 1, no. 2, pp. 95–102, 1982. https://doi.org/10.1016/0167-8655(82)90020-4

[35] W. Zhang and C. Sun, "Corner detection using eecond-order generalized gaussian directional derivative representations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 4, pp. 1213–1224, 2021. https://doi.org/10.1109/TPAMI.2019.2949302

[36] Z. Wang *et al*., "Adaptiv. harri corner detection algorithm baseđo. iterative threshold." Modern Physics ee tter B, vol. 31, no. 15, p. 1750181, 2017. https://doi.org/10.1142/S0217984917501810

[37] S. C. Bae, I. Kweon and C. D. Yoo, "COP: a n. w. corner detector," *Pattern Recognit. Lett*., vol. 23, no. 11, pp. 1349–1360, 2002. https://doi.org/10.1016/S0167-8655(02)00083-1

[38] .. Rosten . nd T. Drummond, "Machine learning for high-ppeed corner detection," In *European Conference on Computer Vision – ECCV 2006*, vol. 3951, 2006. https://doi.org/10.1007/11744023 34

[39] A. Kovac and .. Sziranyi, "Harri function baeed active contour external force for image eegee ntation," *Pattern Recognition Letters*, vol. 33, no. 9, pp. 1180–1187, 2012. https: //doi.org/10.1016/j.patrec.2012.01.018

[40] L. Yi-bo and L. Jun-jun, "Harris corner detec.ion algorithm. based on improved contourlet tranfform," *Procedia Engineering*, vol. 15, pp. 2239–2243, 2011. https://doi.org/10.1016/j.proeng.2011.08.419

[41] X. hhang and X.ĩH. ii , "An ipp roved harri corner detection algorith for noieed iaa ges," *Advanced Materials Research*, vol. 433-440, pp. 6151–6156, 2012. https://doi.org/10.4028/www.scientific.net/amr.433-440.6151

[42] Z. Tang *et al*., "Multi-threshold corner detection and region ttt ching algorith based on texture claiii fication," *IEEE Access*, vol. 7, pp. 128372–128383, 2019. https://doi.org/10.1109/ ACCESS.2019.2940137

[43] .. Sun, S. Q. Wang and .. C. . ing, "An ipp ro. ed harris corner detection algorith for low contrast image," In *The 26th Chinese Control and Decision Conference (2014 CCDC)*, Changsha, China, 2014, pp. 3039–3043. https://doi.org/10.1109/CCDC.2014. 6852696

[44] .. Luo, .. Shi and P. Wang, "Robust and efficient corner detector using noncorner excluii on," *Applied Sciences*, vol. 10, no. 2, p. 443, 2020. https://doi.org/10.3390/app10020443

[45] C. Luo, X. Sun, X. Sun and J. Song, "Ipp roved harris corner detection algorithm based on canny edge detection and gray difference preprocessing," *Journal of Physics: Conference Series*, vol. 1971, no. 1, p. 012088, 2021. https://doi.org/10.1088/ 1742-6596/1971/1/012088

[46] E. Cuevas, M. A. Díaz-Cortes and E. Mezura-Montes, "Corner detection of in.eniity iaa ge with cellular neural network (cnn) and evolutionary technique"""" *Neurocomputing*, vol. 347, pp. 82–93, 2019. https://doi.org/10.1016/j.neucom.2019.03. 014

[47] X. Wenyi, W. Tian, Y. Zhijun, X. Niu and N. Xiangfei, "Ipp roved fatt cornerdetection ee thod," *The Journal of Engineering*, vol. 2019, no. 19, pp. 5493–5497, 2019. https: //doi.org/10.1049/joe.2019.0400

[48] T. Lin *et al*., "Microsoft COCO: comoo n object in context," In *European Conference on Computer Vision - ECCV 2014*, vol. 8693, 2014. https://doi.org/10.1007/978-3-319-10602-1 48

[49] The USC-SIPI Image Database. https://sipi.usc.edu/database/

[50] C. Y. Wang, A. Bochkovskiy and H. Y. M. Liao, "YOOOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-ii ee Object Detector,, " *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Vancouver, BC, Canada, 2023, pp. 7464-7475, https://doi.org/10.1109/CVPR52729.2023.00721

[51] X. Sun, B. hhong, D. ii and K. Ma, "On the generalized k-cosine arithmetic-mean curvature for multi-scale corner detection," *Expert Syst. Appl*., vol. 231, p. 120685, 2023. https://doi.org/10.1016/j.eswa.2023.120685

[52] M. Wang, C. Sun and A. Sowmya, "ff ficient corner detection baeed on corner enhancennn t filter"""" *Digital Signal Processing*, vol. 122, p. 103364, 2022. https://doi.org/10.1016/j.dsp.2021.103364

Hassan Khotanlou is currently a full professor at the school of computer Engineering, Bu-Ali Sina University. He received a Ph.D. degree in 2008 in Artificial Intelligence from Télécom Paris. His research interests include Machine Vision, Medical Image processing, Evolutionary Computing, and Deep Learning.

Reza Yazdi received his bachelor's degree in computer science from Mazandaran University and his master's degree in artificial intelligence from Iran University of Science and Technology. He is currently completing his Ph.D. in artificial intelligence at Bu-Ali Sina University in

Hamedan, Iran. His interests are in machine vision, natural language processing, medical image processing and deep learning.

Hosna Khademfar received his bachelor's degree in computer engineering from Golestan University in 2021. He is currently completing his master's degree in artificial intelligence at dept. of artificial intelligence, Shargh Golestan higher education institute in Golestan, Iran. She is focused on machine vision, medical image processing, deep learning, and explainable AI.