Noisy iris smoothing and segmentation scheme based on improved Wildes method

Anchal Kumawat · Sucheta Panda

Received: 1 March 2022 / Revised: 6 August 2022 / Accepted: 4 September 2022 /
Published online: 28 September 2022
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract

In an automated iris recognition system, in order to get higher accuracy, we should have an efficient iris segmentation process. The reliability of accurate “iris recognition” system largely depends on the accuracy of segmentation process. Traditional “iris segmentation” methods are unable to detect the exact boundaries of iris and pupil, which is time consuming and also highly sensitive to noise. To overcome these problems, we have proposed an improved Wildes method (IWM) for segmentation in iris recognition system. The proposed algorithm consists of two major steps before applying Wildes method for segmentation: edge detection of iris and pupil from a noisy eye image with improved Canny with fuzzy logic (ICWFL) and removal of unwanted noise from above step with a hybrid restoration fusion filter (HRFF). A comparative study of various edge detection techniques is performed to prove the efficiency of ICWFL method. Similarly, the proposed method is tested with various noise densities from 10 to 95 dB. Also the working of the proposed HRFF is compared with some existing smoothing filters. Various experiments have been performed with the help of iris database of IIT_Delhi. Both visual and numerical results prove the efficiency of the proposed algorithm.

Keywords Iris segmentation · Edge detection · Fusion filter · CHT · DWT

1 Introduction

In order to discriminate one person from another and to prove the authenticity of a person, biometric security system plays a major role in today’s world. Furthermore, due to the outbreak of covid-19 virus, everyone is interested in contactless security system. The best option for this problem is iris recognition system. An “iris recognition system” is a form of “biometric authentication” that utilises human irises to authenticate individuals. Now a days, in everyday life style, everyone requires a reliable, fast and automatic personal identification
Iris recognition has a number of steps namely: Image acquisition, Image Segmentation, Image Normalization, Feature Extraction and Matching. In recent years, iris recognition has been evaluated as a trending topic because, it requires fewer parameters for “user cooperation” and “imaging conditions”. However, under these imaging conditions, captured iris images can suffer from various factors i.e. “noise”, “gaze deviation”, “iris rotation”, “absence of iris”, “motion/defocus blur”, “occlusions” due to “eyelid/eyelash/hair/glasses”, that makes iris recognition a challenging task. Performance of the “iris recognition” system depends on the “iris segmentation” where iris segmentation separates an “iris region” from entire captured eye image. The whole process of iris segmentation can be divided into two steps i.e. “iris localization” and “noise detection”. The first step separates the iris part from sampled eye image, which can be used by subsequent steps. The second step finds noise in the iris part i.e. eyelashes, eyelids and reflections. Many researchers proposed “iris segmentation” and/or “localization schemes” such as “histogram and thresholding”, “Circular Hough Transform (CHT)”, “Integro-differential operator (IDO)”, “active contours model”, “graph cuts”, “Genetic Algorithm based CHT(GACHT)” or “deep learning”. Many of these methods unable to deal with the eye images containing various noise factors quoted as “eyebrows, eyelashes, contact lenses, non-uniform illumination, defocus and/or eyeglasses”. The functioning of “CHT” and “IDO” is found to be robust against noise, but fails at higher noise densities (NDs). In contrast, the “histogram” and “thresholding” based approaches prove to be fast, but robustness against noise is very low. To address these issues, this study proposes an effective “segmentation” approach.

In order to solve the above issues, in this paper, we focus on two main essential issues i.e., noise removal and iris segmentation. The major contribution of the current work can be summarized as follows.

To obtain an accurate iris segmentation from a noisy iris image, we have modified the first step of the Wildes approach. Now in the improved Wildes method (IWM), in step 1, we obtain the edge map of the input noisy iris image using ICWFL method. A comparative analysis of six standard edge detection techniques with own proposed ICWFL method is carried out. That is Roberts (Bhardwaj and Mittal, 2012), Sobel (El-Khamy et al., 2000), Prewitt (Gonzalez and Woods, 2002), Canny (1986), ICA (Xuan and Hong, 2017), BE3 (Mittal et al., 2019) and ICWFL (Kumawat and Panda, 2021). The results are shown in Tables 2. It is found that ICWFL (Kumawat and Panda, 2021) outperforms the above six edge detection techniques. In order to get a “smooth” and “noise-free” image from the noisy edge map image from the above step, we have applied proposed hybrid restoration fusion filter (HRFF). Varities of Noise Densities(NDs) starting from 10 to 95 dB are added to the edge map image of step1. After obtaining the smooth grey scale edge map image, we apply Wilde’s approach for finding inner and outer boundaries of edge image i.e. iris and pupil. We have compared the accuracy of segmentation with different NDs (10–95 dB). It is found that the proposed filter gives the best iris segmentation accuracy even with high noise in comparison with some standard filters like “Median Filter (MF)” (Kumar et al., 2020), “Hybrid Median Filter (HMF)” (Rakesh et al., 2013), “Novel Adaptive Fuzzy Switching Median (NAFSM)” (Kenny and Nor, 2010), “Based on Pixel Density Filter (BPDF)” (Erkan and Gokrem, 2018), “Different Applied Median Filter (DMAF)” (Erkan et al., 2018).
2 Related work

For the past two years, due to the outbreak of the covid-19 virus, everyone throughout the globe is interested in contactless security system. Due to its contactless features this technology is extensively used in various industries, offices, educational institutions etc. A standard iris biometric system consists of four modules i.e. eye “image acquisition”, “iris segmentation”, “feature extraction”, “matching and recognition”. Here, “iris segmentation” plays a very important role in the overall system’s performance. If iris segmentation is not done accurately, the process of iris recognition fails and the person can not be authenticated properly. Next we describe some “state-of-the-art” schemes on the iris segmentation.

A study of performance metrices for biometric security is carried out in Sivaram et al. (2019). Daugman (2004), presented how iris recognition system works and a comparison is made among 9.1 million users acquired in trails in Britain the USA, Korea and Japan. Iris quality assessment is carried out in Nathan et al. (2006), by analysing 7 quality factors which affects the recognition. Rao et al. (2020), optimise the parameters used in iris recognition system which aims to make the user friendly system, minimize the “time and space” complexities. A deep learning-based iris segmentation approach named Iris Parse Net is proposed (Wang et al., 2020). The proposed approach is a complete “iris segmentation” solution where the “iris mask and parameterized” both boundaries find out together by modelling them into a unified multitask network. Hunny et al. (2012) proposed an iris segmentation algorithm and an adaptive SURF descriptor for iris recognition using an adaptive threshold method. The proposed method can handle all the possible changes in transformation. Rahmani and Narouei (2020) proposed an automatic iris segmentation approach using graphics processing unit (GPU) to detect the border between iris and pupil. In another recent work, Lubos et al. (2021) makes a comprehensive overview of the various iris recognition datasets. They have focused on the quantitative analysis of scholarly publication data, which are utilised for iris recognition. From the web of science online library, they have reviewed 158 different iris datasets, which have been employed in 689 research articles. In another work, Pathak et al. (2019), an effective method for segmentation of iris, sclera and pupil is presented. The input image is pre-processed by using a bilateral filter. An “iris localization” method by using Hough Transformation is proposed (Sunanda and Shikha, 2015). Hough transformation is used to identify the circles, lines and canny edge detection method is used to improve accuracy. An analysis of iris segmentation using IDO and Hough Transform (HT) approach is presented in Zainal et al. (2013), using CASIA database they have studied the performance of HT. To perform “iris recognition”, “iris segmentation” is essential (Wildes, 1997). Daugman’s and Wildes are the two well known iris segmentation methods that are cited in Peihua and Xiaomin (2008), Manchanda et al. (2013) and Jan and Min-Allah (2020) as well as CHT is used to localize the iris as referred in Verma et al. (2012), Kennedy et al. (2018) and Cherabit et al. (2012). An improved version of Wildes method for iris segmentation is used in this paper.

Li et al. (2010) proposed a novel method to segment iris in noisy iris images. They proposed a “limbic boundary localization” algorithm which is a combination of K-mean clustering and an improved version of Hough transform. Also, an upper eyelid detection approach is proposed that combines a parabolic integro-differential operator and a RANSAC (Random Sample Consensus) which utilizes edges detected by a “one-dimensional edge detector”. In the segmented iris images specular highlight is removed.

An innovative iris segmentation algorithm captured in noisy environments is proposed in Labati and Scotti (2010). This method can extract iris from the input eye image in an
uncontrolled environment, where both reflections and occlusions are also presented. There are mainly three steps to perform iris segmentation. In the first step, it locates the centers of both the pupil and iris in the input image. In the second step, two image strips which contains iris boundaries are extracted and linearized. The last step locates the iris boundary points in the strips by performing regularization operation. Different types of occlusions such as reflections and eyelashes are then pointed out and removed from the final area of segmentation. Jeong et al. (2010) proposes an iris segmentation method for non-ideal iris images. Here AdaBoost eye detection method is used in order to reduce the iris detection error, which are caused by two circular edge detection operations. Secondly the method employs a color segmentation for detecting any obstructions caused by ghosts of the visible light. If extracted “corneal specular reflections” in the detected pupil and iris regions is absent, the captured iris image is determined as a closed eye one.

Khan and Kong (2022) developed an iris segmentation approach which is based on Laplacian of Gaussian (LOG) filter in presence of noise. To detect pupil boundary the LOG filter along with region growing is used. In the next step, zero crossing of the LOG filter is employed to mark the inner and outer boundaries. In a recent work Malinowski and Saeed (2022) proposed an iris segmentation method which is insensitive to light reflections and also reflected mirror images. This approach works well even when pupil and Iris are not positioned perpendicularly to the camera eye. The proposed algorithm is effective for noisy and also poor quality eye images, due to the use of edge approximation using the “harmony search algorithm”. A comprehensive review on iris recognition technique is presented in Malgheet et al. (2021). The recognition of iris is divided into seven phases. The authors reviewed the method of iris recognition associated with each phase. Also, two approaches of iris recognition are presented here, the traditional approach and deep learning approach. Abdelwahed et al. (2020) presented a segmentation algorithm for iris recognition by using a hybridization approach between Daugman’s Integro Differential Operator (IDO) with edge-based methods. The proposed algorithm takes the advantages of the good qualities of both the methods to increase the precision and reduce the recognition time. In another research work (Abdulwahid et al., 2020), H.J. Abdulwahid presented an effective method for locating the iris of the eye image. In the first step, A mixture of gamma transform and contrast enhancing mechanisms are used to isolate the iris area. In the next step, the statistical image parameters, that is, mean and standard deviation are employed as a feature for detection of the outer iris boundary. To detect the inner iris boundary, the IDO technique is used.

3 Proposed methodology for segmenting an iris

Figure 1 illustrates flowchart of the proposed methodology for finding accurate segmentation of an iris using IWM approach from an eye image. It includes three modules: Image Acquisition, Image Preprocessing, Image Segmentation. Image preprocessing is divided into two sections i.e. Edge detection using ICWFL and noise removal from iris region using a novel HRFF.

3.1 Image acquisition

In this paper, we have gathered all the images from publicly available database i.e. IIT _Delhi_ database. Table 1 refers the information about this database i.e. from IIT delhi database. It contains 224 samples of eye images and each sample having 10 variations. Therefore, a
total number of 2240 eye images are available. Here, in this paper we consider 5 samples of each person which belongs to 10 variation so, total it contains 50 samples of eye images. All samaples are of size $320 \times 240$ pixels in BMP format along with NIR (Near Infrared) type. To test the working of the introduced HRFF filter, in the proposed IWM segmentation scheme, we have taken all the noisy sample images and the effectiveness of the proposed filter is tested with variaties of Noise Densities (NDs) from 10 to 95 dB but for the sake of convenience in the paper we have given the results of the proposed method with 10 dB, 30 dB, 50 dB, 70 dB and 95 dB.

### 3.2 Image preprocessing

In the first step of image preprocessing, the input RGB eye image from IIT_Delhi database is converted to greyscale (gs) for further processing and segmentation task. Then 10 dB noise is added to the above gs image. The next step is to detect the edges in the noisy input iris images.

#### 3.2.1 Edge detection using ICWFL (improved Canny with fuzzy logic)

The proposed “edge detector” $ICWFL$ is applied to generate the edge map gradients $E(x, y)$ from smooth gray scale eye images $I(x, y)$ which works well for accurate detection of edges and to test the accuracy of the proposed ICWFL edge detector for iris segmentation, we have done a comparative analysis of some existing edge detectors with the proposed method.
The concept behind the ICWFL approach is referred in Kumawat and Panda (2021). The following algorithm steps elaborate the working of ICWFL edge detector:

**Step 1:** “Canny edge detection (CED)” method used “Gaussian filter” for image smoothing. This method unable to detect the edges on “low contrast and noisy images”. To get an improved accuracy of edge detection, a standard edge detection algorithm should impose more smooth effect to noise and less smooth effect to edge points. Keeping this in mind, we have used a “median filter”, which is a “non linear digital filter”. Because, this filter “preserves sharp edges” in the same time removes “noise”. The output of this filter is calculated by the “median of the gray levels” in the “neighborhood of that pixel”.

\[
Op(i, j) = \text{median}_{(p,q) \in Sw_{i,j}}s(p,q) \quad (1)
\]

where \(s(p,q)\) denotes sampled image, \(Sw_{i,j}\) represented as “pixel under the window mask”, \(Op(i, j)\) represents an “output image”.

**Step 2:** In “CED” to calculate “gradient amplitude”, it uses a small “2X2 neighborhood” window to compute the “finite difference mean value”. But using this method one can miss out some “real edges” and also this is very sensitive to “noise”. So, we have calculated the gradients in three directions i.e. (i) “Horizontal gradient in X direction” (ii) “Vertical gradient in Y direction” (iii) “Diagonal gradient in both X and Y direction”. Here, gradients are calculated by using “Prewitt filter”. If we define I as the source image and the \(G_x\) and \(G_y\) are two images, which at each point contain the horizontal and vertical derivative approximations, then

\[
G_x = \begin{pmatrix} +1 & 0 & -1 \\ +1 & 0 & -1 \\ +1 & 0 & -1 \end{pmatrix} * I \quad (2)
\]

\[
G_y = \begin{pmatrix} +1 & +1 & +1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{pmatrix} * I \quad (3)
\]

where * denotes the 2-dimensional convolution operation. In the horizontal mask i.e., \(G_x\) as the center column is zero, it does not include the original values of an image. It calculates the difference of the right and left pixel values around that edge. This results in an increase in edge intensity values and it became enhanced in comparison to the original image. In the second mask that is \(G_y\), as the center row of the mask consist of zeros, it does not include the original values of the edge in the image. It calculates the difference of the above and below pixel intensities and making the edge visually clear. Both the masks have opposite sign in them and sum of both masks equals to zero. At each point in the image, the resulting gradient approximation can be combined to give the gradient magnitude by using the following equation:

\[
G = \sqrt{G_x^2 + G_y^2} \quad (4)
\]

Using the above equation, we can also calculate the gradient direction as given by:

\[
\theta = \arctan \frac{G_y}{G_x} \quad (5)
\]

where \(\theta\) denotes the angle of direction. If the value of \(\theta\) equal to zero, it refers to a vertical edge, which is darker on the right side.
**Step 3:** After “gradient calculated image” we can see “thick” and “thin” edges. “Non-Max Suppression (NMS)” step will help us mitigate the thick ones. It is an “edge-thinning technique” which is applied to find “the largest” edge. After that “gradient calculation”, “edge extracted” from the “gradient value” is “still blurred”. Calculating the “NMS” value, “ICWFL” uses “$3 \times 3$ mask” of pixels, where each pixel consists of “eight neighboring pixels (E, W, N, S, ES, SW, WN, NE)”. Pixel comparison along with its direction as shown in Fig. 2.

Consider the edge in the below Fig. 3, which has three edge points. Assume that the point $(x, y)$ is having largest gradient of edge. Check the edge points in the direction perpendicular to the edge and verify if their gradient is less than $(x, y)$. If the values are less than $(x, y)$ gradient, we can suppress those non-maxima points along the curve as shown in equation 6.

$$M(x, y) = \begin{cases} |\nabla s|(x, y), & \text{if } |\nabla s|(x, y) > |\Delta s|(x', y') \text{ and } |\nabla s|(x, y) > |\Delta s|(x'', y'') \\ 0 & \text{otherwise} \end{cases}$$

(6)

If a sample pixel value is “greater than its adjacent pixels” then do not “replace the value” else “replace the value”.

**Fig. 2** Pixel comparison along with its direction

**Fig. 3** Example of NMS calculation
Step 4: After the NMS step, there are few edge pixels which can be affected by “noise and scalar variation”. In order to account for these “spurious responses”, it is required to filter pixels with a “weak gradient value” and preserve edge pixel with a “high gradient value”. This requirement can be made by selecting “high and low threshold value”. If an edge pixel’s “gradient” value is higher than the “high threshold” value, it is marked as a “strong edge pixel”. If an edge pixel’s “gradient” value is smaller than the “high threshold” value and “larger than the low threshold value”, it is marked as a “weak edge pixel”.

If an edge pixel’s value is smaller than the “low threshold” value, it will be “suppressed”. In the “traditional Canny edge detection algorithm”, there will be two fixed “manual global threshold values” to filter all the “false edges”. But, as the image gets complex, different “local areas” will need “different threshold values” to accurately find the “real edges”. A “threshold set to high” can miss some important information. On the other hand, a “threshold set too low” will falsely identify some “irrelevant information such as noise” as important information. It is difficult to give a “generic threshold” that “works well” on all images. So, the main improvement in this step is to preserve or find all the edges whether it is false or true, without giving the thresholds manually. Both high and low thresholds are obtained by the following equations.

\[
\text{thigh} = 0.5 \times \frac{1}{n} 
\]

Here, “thigh” refers to high threshold and “n” represent the total number of pixels in the given input image.

\[
\text{tlow} = 0.5 \times \text{th} 
\]

Here, “tlow” refers to the low threshold.

Step 5: In the last step, all the “unnecessary edges” are “suppressed” that are either “weak” or those which are “not connected” to “strong edges”.

Step 6: In traditional edge detection procedures there are some drawbacks like “edge thickness” is fixed and the parameters like “threshold” is difficult to implement. The advantage of the “Fuzzy rule based technique” is that “thickness of the edge” can be controlled by “altering the rules” and “output parameters”. Using this, the complexity of the problem can be reduced drastically. In the proposed work, the output image from the “improved version of canny edge detection algorithm” is fed to a “Fuzzy Inference System (FIS)”. A FIS is designed which takes the “process values as input” and these values are then converted into “fuzzy plane”. A “Fuzzy rule base” is defined that determine and show the “edge pixels” in the “output image”. In this step, to “preprocess the image” before “FIS” is applied, the concept of a “window or mask” is carried out as shown in Fig. 4. This mask takes the greyscale sample values \(S_1, S_2, S_3, \ldots, S_8\) of “eight neighborhood pixels” with the “center pixel \(S\)”, as the output pixel as shown in Fig. 4a. Figure 4b demonstrates the “processed window mask”, where \(\Delta S_j = (S_j) - (S)\), for \(j = 1, 2, 3, \ldots, 8\).

Step 8: In “fuzzifier stage” “input membership function” is used where the “grey levels” of the images are “mapped” to new set of “linguistic values”.

Step 9: In “defuzzifier” or “output stage” where the values of the “grey level” are “mapped” to “new crisp values”. In the current work, “defuzzification” is done with the “Centroid of Area (COA)” method.
Algorithm steps for ICWFL edge detector

**Input**: smooth gray scale (gs) eye images obtained from Sect. 3.2

I: “Median filter (MF)” applied on gs image.

II: “gradient magnitude and direction” is calculated on gs MF image.

III: “edge thinning” technique performed on “output image” of step 2.

IV: “double threshold” is used to discard and save the edge pixels with “weak” and “strong” gradient values.

V: “hysteresis” is performed to track the edges which obtain the final “improved Canny edge detected” image.

VI: above output image is “scanned” by “3 × 3 window mask”.

VII: FIS system is designed taken as “eight scanned pixels” as a “crisp input” then these are converted into “linguistic variables” i.e. “Low, Mid and High” by using “Triangular membership function (Tmf)”.

VIII: For the above “3 × 3 window mask” for inputs 24 “fuzzy rules” applied to obtain “fuzzy outputs” i.e. “Weak, Strong or Partial edges” using “Gaussian membership function (Gmf)” based on combination of three “linguistic variables”.

IX: Using “Centroid of Area (COA)” method the above “fuzzy output” is “defuzzified” to get the noisy edge map image.

**Output**: Improved Canny with Fuzzy logic noisy edge map image obtained

End of Algorithm

Figure 5, illustrates flowchart of the proposed ICWFL algorithm.

Figure 6, shows the Edge map gradients of various edge detectors. Here, Fig. 6a represents the smooth grey scale original image, (b, c) shows the horizontal and vertical edge map gradient of (a). Figure 6d represents Vertical edge map gradient of Roberts (Bhardwaj and Mittal, 2012) edge detector. (e–i) shows Vertical edge map gradient of Sobel (El-Khamy et al., 2000), Prewitt (Gonzalez and Woods, 2002), Canny (1986), ICA (Xuan and Hong, 2017) and BE3 (Mittal et al., 2019) edge detector respectively. Figure 6j represents the Vertical edge map gradient of ICWFL edge detector that produces the fine smooth edges as compared to all existing edge detectors.

### 3.2.2 Novel hybrid restoration fusion filter (HRFF)

As the output image of Sect. 3.2.1 contains unwanted noise, we have proposed a Hybrid restoration fusion filter (HRFF) to get a smooth and clear image. So, to the grey noisy edge map \( E(x, y) \) image, HRFF is applied to obtain a clean image \( SI(x,y) \). A novel “HRFF” is proposed with the help of multiresolution concept using “image fusion” combined with
Important features of two restoration filters i.e. DWF (Deconvolution using Wiener Filter) (Trambadia and Dholakia, 2015) and DLR (Deconvolution using Lucy-Richardson Filter) (Al-Taweel et al., 2015). So, motivation behind taking “image fusion based on wavelets” is that of “coefficient combination”. We can merge the coefficient in an appropriate way, which is fit to a particular application, to obtain the best quality in the fused images. The following algorithm steps detail the working of HRFF:

**Algorithm steps for Hybrid Restoration Fusion Filter (HRFF)**

**Input**: Take two noisy edge map eye images obtained from the output of Sect. 3.2.1

I: Apply two non blind deconvolution algorithm i.e. DWF and DLR on input images
II: Perform “DWT decomposition” on above restored images
III: After decomposition, the “approximation and detailed component” can be separated. Here, we modified only approximation coefficient of both restored filters and detail coefficients remain unchanged
IV: obtained approximated coefficient of both DWF and DLR restored filter, then apply DWF and DLR filter again on approximated coefficients which became the modified restored filter i.e. modified DWF (MDWF) and modified DLR (MDLR)
IV: Set MDWF and MDLR image to s1 and s2, then fix the value of “fused factor(ff)” is 0.8. Hence, formulation for the fused image Fs for s1 and s2 is shown below:

\[
F_{s1} = (1 - ff) * s1; 
F_{s2} = ff * s2; 
F_s = F_{s1} + F_{s2};
\]

where the Fs1 and Fs2 matrices are shown as follows:

\[
F_s = \begin{bmatrix} MWAC & WHC \\ WVC & WDC \end{bmatrix} + \begin{bmatrix} MLAC & LHC \\ LVC & LDC \end{bmatrix}
\]

V: After fusion, obtained four coefficients of double hybrid restoration filter image such as Fused Modified Wiener Lucy Approximated Coefficient (FMWLAC), Fused Wiener Lucy Horizontal Coefficient (FWLHC), Fused Wiener Lucy Vertical Coefficient (FWLVC) and Fused Wiener Lucy Detailed Coefficient (FWLDC) where FMWLAC(MWAC,MLAC), FWLHC(WHC,LHC), FWLVC(WVC,LVC) and FWLDC(WDC,LDC).
VI: perform “IDWT(Inverse Discrete Wavelet Transform)” to get a “resultant image”
VII: obtained a fused “double hybrid modified restoration fused filter synthesized” eye image

Output : HRFF eye image obtained

End of Algorithm

Step 1: The process of HRFF inculdes a unified “double hybrid restoration filter” for noise reduction, combined with important features of two restoration filters i.e. DWF (Trambadia and Dholakia, 2015) and DLR (Al-Taweel et al., 2015) as referred in (Trambadia and Dholakia, 2015) and (Al-Taweel et al., 2015). We have applied “DWF filter and DLR filter”, to the two “noisy eye images” which is obtained from 3.2.1 seperately.
Step 2: “restored DWF image and restored DLR image” are decomposed, using “multiresolution approach i.e. DWT”.
Step 3: After “DWT decomposition, obtained 4 bands which transforms the image from the “spatial domain to frequency domain” using “2-D Discrete Wavelet Transformation (DWT)” . The image splits into “vertical and horizontal lines” and represents the “first-order of DWT”, and the image consist of four parts such as, “Approximated coefficient (AC), Horizontal coefficient (HC), Vertical Coefficient (VC) and Diagonal coefficient (DC)”. So, obtained 8 sets of coefficients, where first four sets derived from DWF filter and another four sets from DLR filter. The coefficients are “Wiener Approximated Coefficient(WAC)”, “Wiener Horizontal coefficient (WHC)”, “Wiener Vertical coefficient (WVC)” and “Wiener Diagonal coefficient (WDC)” resulting from “DWF filter”. Similarly, for “DLR filter” we get the coefficients such as: “LAC (Lucy-Richardson Approximated Coefficient), LHC
Lucy-Richardson Horizontal Coefficient), LVC (Lucy-Richardson Vertical Coefficient) and LDC (Lucy-Richardson Diagonal Coefficient)”. This paper deals with the “multiresolution decomposition” which refers to the “discrete two dimensional wavelet transform”, that is proposed before applying the concept of “image fusion”. When “decomposition” is done, the “approximation and detailed component” can be separated. Out of these four bands, the “low-frequency coefficients” based on wavelet transforms retain the “most energy” of the source images. Because of this, this paper is focused on a double hybrid restoration filter to “WAC and LAC” component only and on the other hand, coefficients i.e. “WHC, WVC, WDC and LHC, LVC and LDC” coefficients are remain unaffected.

**Step 4:** After applying the “DWF filter” on “WAC coefficients”, the decomposition of the “modified DWF restoration” will have a coefficients as “MWAC, WHC, WVC and WDC”. Here, only “approximated coefficients” are modified and all other coefficients “remain unchanged”. After applying “DWT” four resulting coefficients can be represented as:

\[
AC = [(s(i, j) \ast \phi - i \phi - j)(2p, 2q)]_{(p,q) \in \mathbb{Z}^2} \\
HC = [(s(i, j) \ast \phi - i \psi - j)(2p, 2q)]_{(p,q) \in \mathbb{Z}^2} \\
VC = [(s(i, j) \ast \psi - i \phi - j)(2p, 2q)]_{(p,q) \in \mathbb{Z}^2} \\
DC = [(s(i, j) \ast \psi - i \psi - j)(2p, 2q)]_{(p,q) \in \mathbb{Z}^2}
\]

where \(AC\), \(HC\), \(VC\) and \(DC\) represents “Approximated coefficient, Horizontal coefficient, Vertical Coefficient and Diagonal coefficient” of given image. \(\phi\) and \(\psi\) represents the “scaling and wavelet function”. \(\mathbb{Z}^2\) represents the “size of the image”, \(p, q\) indicate the “coordinates of the image” and “\(s(i, j)\)” represents the given “sample image” on which we have applied the “level-1 decomposition using DWT”.

**Step 5:** In this step, image fusion is applied to the coefficients of both “MDWF” and “MDLR” filter. This work represents an “image fusion scheme” based on the “wavelet transform”.

**Step 6:** After fusion of both filters, obtained four coefficients of double hybrid restoration filtered image.

**Step 7:** In order to get a fused image having the properties of both modified hybrid restoration filters, “image composition” based on “IDWT” is performed to get resultant image. Formulation of creating “fusion model” for image restoration using DWT can be summarized in algorithm step IV. Figure 7 illustrates the flowchart of a novel HRFF approach.

The efficiency of the proposed HRFF can be tested on the output noisy edge map image of step 3.2.1. We have tested the effect of the algorithm taking the effect of the proposed HRFF filter with various NDs i.e. 10 dB, 30 dB, 50 dB, 70 dB and 95 dB. It can be seen from Fig. 8. So, we can conclude that the “proposed filter” works well even with high NDs i.e. 95 dB, as well as with low NDs i.e. 10 dB, 30 dB, 50 dB and 70 dB. Also, with the help of the new double hybrid restoration filter, combined with multiresolution approach to image fusion, we can obtain a smooth fine image, retaining all the important information from the degraded image.

The proposed HRFF is having the following properties:

1. Generally, noise smoothing filters when applied to noisy images tends to blur the images. But, in addition to this noise is also reduced. HRFF filter is also showing this property i.e., in the process of noise reduction it will result in a blur output image.
2. This filter is good at removing salt and pepper noise from an image. But it also works well for other noises.

3. HRFF Filter preserves sharp edges, in the process of noise reduction. The process results in an image with reduced sharp transitions in intensities that ultimately leads to noise reduction.

4. As HRFF is a combination of two well-known filters that is Wiener filter and Lucy-Richardson filter, this filter works faster than the other filters.

So, this filter is used to suppress image noise, enhance edges and improve edge clarity.

A numerical comparison is shown in Table 2, taking into consideration three image quality assessments parameters i.e. MAE (Mean Absolute Error), RMSE (Root Mean Squared Error) and PSNR (Peak Signal-to-Noise Ratio). In this table a comparison is made between the output HRFF image with the input noisy ICWFL edge map image taking six edge detectors. It is observed that in case of MAE the error is low in all edge detected filtered image. But in proposed ICWFL error is very less i.e. 0.3742 in case of HRFF image. In case of RMSE,
Table 2  Shows the comparison of various edge detectors on noisy and filtered images based on IQA parameters

| Edge detection          | MAE     | RMSE     | PSNR    |
|-------------------------|---------|----------|---------|
|                         | Noisy img | HRFF img | Noisy img | HRFF img | Noisy img | HRFF img |
| Roberts (Bhardwaj and Mittal, 2012) | 0.6960 | 0.7589 | 26.2344 |
|                         | 0.6071 | 0.6672 | 35.1434 |
| Sobel (El-Khamy et al., 2000) | 0.6996 | 0.7617 | 25.8669 |
|                         | 0.6079 | 0.6677 | 35.0764 |
| Prewitt (Gonzalez and Woods, 2002) | 0.6992 | 0.7614 | 25.8964 |
|                         | 0.6075 | 0.6674 | 35.1184 |
| Canny (1986)            | 0.6615 | 0.7326 | 29.7664 |
|                         | 0.5881 | 0.6522 | 37.1156 |
| ICA (Xuan and Hong, 2017) | 0.6560 | 0.7287 | 30.3067 |
|                         | 0.5915 | 0.6553 | 36.7117 |
| BE3 (Mittal et al., 2019) | 0.6883 | 0.7530 | 27.0069 |
|                         | 0.5991 | 0.6611 | 35.9386 |
| ICWFL (Kumawat and Panda, 2021) | 0.4617 | 0.5606 | 57.3341 |
|                         | 0.3742 | 0.4652 | 66.4635 |

MAE, mean absolute error; RMSE, root mean squared error; PSNR, peak signal to noise ratio

HRFF image shows least error for ICWFL edge map image. For the third parameter i.e. PSNR, if the value is high, it produces high quality of image. For the ICWFL edge map image, when we apply HRFF the value of PSNR is 66.4635 which is the highest among all the existing six edge detected images. So, from this numerical result, we can conclude that when we combine ICWFL edge detected image with HRFF, the image quality is very high in comparison to six existing edge detected images like Roberts (Bhardwaj and Mittal, 2012), Sobel (El-Khamy et al., 2000), Prewitt (Gonzalez and Woods, 2002), Canny (1986), ICA (Xuan and Hong, 2017), BE3 (Mittal et al., 2019). Furthermore, there is a difference in numerical values of noisy and filtered images in all the edge detection procedures. That is in case of two image quality assessment parameters MAE, RMSE the amount of error in case of noisy images is more than that of the HRFF image. And for PSNR, the amount of information content is more in HRFF image in comparison to noisy image.

3.3 Image segmentation based on IWM approach

Improved Wilde’s Method (IWM) is a modified version of existing Wildes method. Here, it takes the input from the above HRFF filtered image section 4.2.2 for finding an outer and inner boundary of the circle. The following algorithm illustrates the procedure of finding an iris and pupil from the given eye HRFF smooth image.

**Algorithm steps for IWM approach**

**Input**: HRFF smooth image obtained from section 4.2.2

**Step 1**: Finds outer boundary(iris) from the given image

**Step 2**: Initialize the center coordinates with their radius coordinates for outer circle

**Step 3**: Finds inner boundary(pupil) from the given image

**Step 4**: Initialize the center coordinates with their radius coordinates for inner circle

**Step 5**: compute the circle gradients for both inner and outer

**Step 6**: test the computed gradient is maximum or minimum

**Step 7**: If maximum then construct the circle with their initialized coordinates

**Output**: segmented smooth eye image having iris and pupil boundary

**End of Algorithm**

After obtaining smooth gray scale edge map gradients apply segmentation approach for finding inner and outer boundary of eye image i.e. iris and pupil where inner boundary refers
to pupil and outer boundary of circle refers to iris. Segmenting an iris and pupil together using existing Wildes approach where an input becomes filtered gray scale smooth edge map image. For finding an outer boundary, initialize the center coordinates with their radius and then construct the circle with the help of center and radius coordinates. Three parameters are needed for finding any circle \((x_p, y_p, r_p)\). Here \((x_p, y_p)\) represents the venter coordinates and \(r_p\) denoted the radius of the circle. So the accumulator will be as:

\[
IWM_p = \sum_{q=1}^{m} IWM(x_q, y_q, x_p, y_p, r_p)
\]

where \(m\) is total number of pixles in edge map image and \(IWM(x_q, y_q, x_p, y_p, r_p)\) is the basic circle equation i.e.\([\sqrt{(x_q - x_p)} + \sqrt{(y_q - y_p)} - r_p]\). Hence radius of pupil will be denominated as: \(r_p = \sqrt{(x_q - x_p)} + \sqrt{(y_q - y_p)}\) Here \(q\) refers to 1,2,3...\(m\) and \(IWM_p\) denotes circle range for lower limit and upper limit of radius. Lower limit of radius \(r_l\) will be as \((x_p, y_p, r_p)\) and upper limit of the radius \(r_u\) parameters will be as \((x_p, y_p, r_p)\). For localization of circle, radius range is always required. After that we compute the circle gradients in both ”horizontal and vertical directions”, when the edge map gradient is maximum. If it is maximum, then it is able to detects the outer boundary of the circle in vertical direction, otherwise for constructing a circle, we have to initalize the center coordinates along with their radius again. Repeat this process till we get the maximum edge map gradient for finding outer boundary of the circle i.e. iris and similar steps are followed for pupil also. Figure 9 illustrates a flowchart of the proposed Image segmentation IWM approach.

3.4 Comparison between existing Wildes and improved novel Wildes approach for segmenting an iris

There are a number of shortcomings with the existing Wildes method (WM) and these can be solved using IWM approach. They are:
1: WM requires “threshold values” to be chosen for “edge detection”, and this may result in “critical edge points” being removed, resulting in failure to detect circles/arcs where as Improved novel Wilde’s approach does not require any threshold values for the edge detection.

2: The WM is “computationally intensive” due to its “brute-force approach”, and it is not suitable for “real-time applications” but the proposed algorithm can be applied to real time applications, as it is very fast in execution. This effect can be seen from Table 4, where we have given a comparison of execution time of all the existing filters with the own proposed one.

3: WM approach is highly sensitive to image noise. where as in IWM method it is less sensitive to noise. Even with high ND like 95dB, IWM works well and can segment iris and pupil from the input eye image.

4: When Noise density is high, Wildes method is unable to detect an iris properly where as, in higher noise density IWM is able to detect but, the results leads to blurriness.

5: IWM approach produces more accurate result as compare to existing Wildes approach. Below Fig. 10 shows the results of existing Wildes approach for noisy and segmenting images and Fig. 11 shows result of noisy and segmenting images based on IWM approach.

6: While segmenting an eye image with existing WM method, it contains circle iris, circle pupil, noise with eyelids and eyelashes as shown in Fig. 10a, b shows only iris and pupil of segmented image of Fig. 10a. Figure 11a, IWM method is applied on given sampled eye images which is combination of edge detected ICWFL with HRFF filtered image to show an iris, pupil, noise with eyelids and eyelashes where the noise is less as compare to Fig. 10a. Figure 11b shows only iris and pupil of Fig. 11a sampled eye image where the noise with eyelids and eyelashes are removed. Table 3 shows that IWM approach is better. Because, from Table 3 in case of WM the radius of iris is 99 for sample image S1 and for IWM it is 100 that is, the boundary is detected properly. Similar findings can be drawn in case of pupil.

In this way we can segment an iris and pupil from input eye image using novel IWM segmenting approach. The main difference between the existing and own proposed approach is the edge map and filter and the major contribution of this paper is to present an approach
Fig. 11 Result images of IWM approach a noisy sample images S1, S2, S3; b segmenting sample images S1, S2, S3

Table 3 Numerical analysis of different segmenting approaches

| IRIS Sample | WM | IWM |
|-------------|----|-----|
| X-coordinate | Y-coordinate | Radius | X-coordinate | Y-coordinate | Radius |
| S1          | 138 | 181 | 99   | 140 | 180 | 100 |
| S2          | 142 | 172 | 100  | 140 | 173 | 100 |
| S3          | 123 | 172 | 102  | 125 | 170 | 103 |
| PUPIL       | 135 | 183 | 37   | 137 | 183 | 38 |
| S1          | 139 | 174 | 39   | 140 | 175 | 40 |
| S2          | 120 | 174 | 39   | 122 | 175 | 40 |

that produce effective segmentation of iris to authenticate a people in less time and reduce the complexity and increase the reliability.

The implemented approach was tested with various noise densities on the sampled eye images, such as the images containing noise of low, mid or high density. The test results demonstrated that the "Wildes algorithm" detects the iris efficiently in the lower noise density images with higher accuracy. The functioning of the algorithm on the higher noise density images has been improved by additional preprocessing (filter with ICWFL) approach on these images.

4 Simulation and results

To prove the efficiency of the proposed IWM algorithm we have carried out a performance analysis of different existing restoration filters with the own proposed HRFF with various NDs i.e. 10 dB, 30 dB, 50 dB, 70 dB and 95 dB. Both visual and numerical results shows the accuracy of the proposed HRFF which is applied in IWM algorithm. Figures 12, 13, 14, 15, 16 and 17 shows a comparative analysis of five existing filters i.e. MF (Kumar et al., 2020),
Fig. 12 Five sampled noisy images of each person as input from IITDelhi database with varying noise density (ND): a ND 10dB; b ND 30dB; c ND 50dB; d ND 70dB; e ND 95dB
Fig. 13 Five sampled segmenting noisy images of each person as input from IITDelhi database removing noise density 10 based on different filters using IWM: a median filter (Kumar et al., 2020); b NAFSM filter (Kenny and Nor, 2010); c BPDF filter (Erkan and Gokrem, 2018); d DAMF filter (Erkan et al., 2018); e HMF filter (Rakesh et al., 2013); f Own proposed HRFF filter (Kumawat and Panda, 2021);
Fig. 14 Five sampled segmenting noisy images of each person as input from IITDelhi database removing noise density 30 based on different filters using IWM a median filter (Kumar et al. (2020)); b NAFSM filter (Kenny and Nor, 2010); c BPDF filter (Erkan and Gokrem, 2018); d DAMF filter (Erkan et al., 2018); e HMF filter (Rakesh et al., 2013); f own proposed HRFF filter (Kumawat and Panda, 2021)
Fig. 15 Five sampled segmenting noisy images of each person as input from IITDelhi database removing noise density 50 based on different filters using IWM 

- **a** median filter (Kumar et al., 2020); 
- **b** NAFSM filter (Kenny and Nor, 2010); 
- **c** BPDF filter (Erkan and Gokrem, 2018); 
- **d** DAMF filter (Erkan et al., 2018); 
- **e** HMF filter (Rakesh et al., 2013); 
- **f** own proposed HRFF filter (Kumawat and Panda, 2021)
Fig. 16 Five sampled segmenting noisy images of each person as input from IITDelhi database removing noise density 70 based on different filters using IWM a median filter (Kumar et al., 2020); b NAFSM filter (Kenny and Nor, 2010); c BPDF filter (Erkan and Gokrem, 2018); d DAMF filter (Erkan et al., 2018); e HMF filter (Rakesh et al., 2013); f own proposed HRFF filter (Kumawat and Panda, 2021)
Fig. 17 Five sampled segmenting noisy images of each person as input from IITDelhi database removing noise density 95 based on different filters using IWM a median filter (Kumar et al., 2020); b NAFSM filter (Kenny and Nor, 2010); c BPDF filter (Erkan and Gokrem, 2018); d DAMF filter (Erkan et al., 2018); e HMF filter (Rakesh et al., 2013); f own proposed HRFF filter (Kumawat and Panda, 2021)
Fig. 18 Nine sampled segmented speckle noisy images of each person as input from IITDelhi database with varying noise density (ND) 70 a S1 eye image; b S2 eye image; c S3 eye image; d S4 eye image; e S5 eye image; f S6 eye image; g S7 eye image; h S8 eye image; i S9 eye image

HMF (Rakesh et al., 2013), NAFSM (Kenny and Nor, 2010), DAMF (Erkan et al., 2018), BPDF (Erkan and Gokrem, 2018) with own proposed(op) filter (Kumawat and Panda, 2021). At low NDs all these filters give more or less similar result. But when we keep on increasing noise i.e. 70 dB and 95 dB, all the existing filters fail to restore the original image. But HRFF still retains almost all the information from original image. Figure 18 shows nine sampled segmented images that has been taken from IITDelhi database with 70dB noise density using speckle noise. Here, Fig. 18a shows first sample of eye images i.e. S1 having ten variation of images but due to space constraints this paper presents only 7 variation of images of each sample. Figure 18b shows second sample that is S2 and Fig. 18c–i shows sample from S3 to S9. This paper describes various image quality parameters such as: PSNR (Peak Signal-To-Noise Ratio), SNR (Signal-To-Noise Ratio), Resolution of sampled segmented eye images with noisy image and various accuracy parameters i.e. IR (Iris Ratio), PR (Pupil Ratio), PSIR (Performance of Segmenting Iris Ratio), PSPR (Performance of Segmenting Pupil Ratio) and FAR (False Acceptance Rate), whose equations are given below:

\[
IR = \left[ \frac{\text{accurate iris segmentation}}{\text{accurate iris segmentation} + \text{inaccurate iris segmentation}} \right] \times 100 \tag{18}
\]

\[
PR = \left[ \frac{\text{accurate pupil segmentation}}{\text{accurate pupil segmentation} + \text{inaccurate pupil segmentation}} \right] \times 100 \tag{19}
\]

\[
PSIR = \left[ \frac{\text{accurate pupil segmentation}}{\text{accurate pupil segmentation} + \text{inaccurate iris segmentation}} \right] \times 100 \tag{20}
\]

\[
PSPR = \left[ \frac{\text{accurate iris segmentation}}{\text{accurate iris segmentation} + \text{inaccurate pupil segmentation}} \right] \times 100
\]
Table 4  Filters applied on various sampled 50 images contains salt and pepper noises on different densities i.e. 10 dB, 30 dB, 50 dB, 70 dB, 95 dB

| Filters applied on | acc. seg. | Accuracy% | Improper% | No. of acc. seg. iris | No. of acc. seg. pupil | Improper Iris | Improper pupil | PSIR% | PSPR% | IR% | PR% | Time |
|--------------------|-----------|-----------|-----------|----------------------|----------------------|---------------|---------------|-------|-------|-----|-----|------|
| N D10              |           |           |           |                      |                      |               |               |       |       |     |     |      |
| MF (Kumar et al., 2020) | 41        | 82        | 18        | 48                   | 41                   | 2             | 9             | 95.34883721 | 84.21052632 | 96   | 82  | 12.5179800 |
| HMF (Rakesh et al., 2013) | 42        | 84        | 16        | 46                   | 42                   | 4             | 8             | 91.30434783 | 85.18518519 | 92   | 84  | 10.1072850 |
| DAMF (Erkan et al., 2018) | 35        | 70        | 30        | 45                   | 35                   | 5             | 15            | 87.50000000 | 75.00000000 | 90   | 70  | 26.7017440 |
| BPDF (Erkan and Gokrem, 2018) | 27 | 54        | 46        | 38                   | 34                   | 12            | 16            | 73.91304348 | 70.37037037 | 76   | 68  | 36.4884090 |
| NAFSM (Kenny and Nor, 2010) | 31        | 62        | 38        | 38                   | 40                   | 12            | 10            | 76.92307692 | 79.16666667 | 76   | 80  | 11.5460760 |
| OP (Kumawat and Panda, 2021) | 45        | 90        | 10        | 50                   | 45                   | 0             | 5             | 100.0000000 | 90.90909091 | 100  | 90  | 08.0417000 |
| N D30              |           |           |           |                      |                      |               |               |       |       |     |     |      |
| MF (Kumar et al., 2020) | 31        | 62        | 38        | 43                   | 36                   | 7             | 14            | 83.72093023 | 75.43859649 | 86   | 72  | 08.6981590 |
| HMF (Rakesh et al., 2013) | 33        | 66        | 34        | 40                   | 38                   | 10            | 12            | 79.16666667 | 76.92307692 | 80   | 76  | 11.3607350 |
| DAMF (Erkan et al., 2018) | 34        | 68        | 32        | 36                   | 39                   | 14            | 11            | 73.58490566 | 76.59574468 | 72   | 78  | 48.4974950 |
| BPDF (Erkan and Gokrem, 2018) | 18 | 36        | 64        | 36                   | 25                   | 14            | 25            | 64.10256410 | 59.01639344 | 72   | 50  | 69.9232210 |
| NAFSM (Kenny and Nor, 2010) | 28        | 56        | 44        | 36                   | 34                   | 14            | 16            | 70.83333333 | 69.23076923 | 72   | 68  | 15.0349910 |
| OP (Kumawat and Panda, 2021) | 44        | 88        | 12        | 45                   | 44                   | 5             | 6             | 89.79591837 | 88.23529412 | 90   | 88  | 08.0304740 |
| N D50              |           |           |           |                      |                      |               |               |       |       |     |     |      |
| MF (Kumar et al., 2020) | 30        | 60        | 40        | 38                   | 34                   | 12            | 16            | 73.91304348 | 70.37037037 | 76   | 68  | 07.9216870 |
| HMF (Rakesh et al., 2013) | 25        | 50        | 50        | 30                   | 34                   | 20            | 16            | 62.96296296 | 65.21739130 | 60   | 68  | 10.8463230 |
| DAMF (Erkan et al., 2018) | 32        | 64        | 36        | 34                   | 38                   | 16            | 12            | 70.37037037 | 73.91304348 | 68   | 76  | 66.2846860 |
| BPDF (Erkan and Gokrem, 2018) | 14 | 28        | 72        | 41                   | 14                   | 9             | 36            | 60.86956522 | 53.24675325 | 82   | 28  | 96.4952180 |
| NAFSM (Kenny and Nor, 2010) | 28        | 56        | 44        | 32                   | 36                   | 18            | 14            | 66.66666667 | 69.56217393 | 64   | 72  | 19.6596560 |
| OP (Kumawat and Panda, 2021) | 38        | 76        | 24        | 41                   | 43                   | 9             | 7             | 82.69230769 | 85.41666667 | 82   | 86  | 07.7221410 |
| Filters applied on \(i/p\) | acc. seg. | Accuracy% | Improper% | No. of acc. seg. iris | No. of acc. seg. pupil | Improper Iris | Improper pupil | PSIR% | PSPR% | IR% | PR% | Time |
|-----------------------------|-----------|-----------|-----------|----------------------|----------------------|---------------|---------------|-------|-------|-----|-----|------|
| \(N D70\)                  |           |           |           |                      |                      |               |               |       |       |     |     |      |
| MF (Kumar et al., 2020)     | 10        | 20        | 80        | 11                   | 12                   | 39            | 38            | 23.52941176 | 22.44897959 | 22  | 24  | 07.1620540 |
| HMF (Rakesh et al., 2013)   | 9         | 18        | 82        | 10                   | 9                    | 40            | 41            | 18.36734694 | 19.60784314 | 20  | 18  | 13.3605580 |
| DAMF (Erkan et al., 2018)   | 29        | 58        | 42        | 29                   | 35                   | 21            | 15            | 62.50000000 | 65.90909091 | 58  | 70  | 104.797447 |
| BPDF (Erkan and Gokrem, 2018) | 8         | 16        | 84        | 31                   | 8                    | 19            | 42            | 29.62962963 | 42.46575342 | 62  | 16  | 131.361039 |
| NAFSM (Kenny and Nor, 2010) | 22        | 44        | 56        | 25                   | 30                   | 25            | 20            | 54.54545455 | 55.55555556 | 50  | 60  | 26.9021650 |
| OP (Kumawat and Panda, 2021) | 38        | 76        | 24        | 43                   | 42                   | 7             | 8             | 85.71428571 | 84.31372549 | 86  | 84  | 07.0915439 |
| \(N D95\)                  |           |           |           |                      |                      |               |               |       |       |     |     |      |
| MF (Kumar et al., 2020)     | 0         | 0         | 100       | 0                    | 0                    | 50            | 50            | 0.00000000 | 0.00000000 | 0   | 0   | 07.3340460 |
| HMF (Rakesh et al., 2013)   | 0         | 0         | 100       | 0                    | 0                    | 50            | 50            | 0.00000000 | 0.00000000 | 0   | 0   | 10.8367430 |
| DAMF (Erkan et al., 2018)   | 3         | 6         | 94        | 3                    | 4                    | 47            | 46            | 7.843137255 | 6.12244898 | 6   | 8   | 126.539551 |
| BPDF (Erkan and Gokrem, 2018) | 0         | 0         | 100       | 0                    | 0                    | 50            | 50            | 0.00000000 | 0.00000000 | 0   | 0   | 160.661854 |
| NAFSM (Kenny and Nor, 2010) | 4         | 8         | 92        | 4                    | 4                    | 46            | 46            | 8.00000000 | 8.00000000 | 8   | 8   | 31.6305470 |
| OP (Kumawat and Panda, 2021) | 22        | 44        | 56        | 24                   | 22                   | 26            | 28            | 45.83333333 | 46.15384615 | 48  | 44  | 06.1661920 |
Table 5  PSNR applied on various sampled 90 images contains speckle noises on 70dB noise density

| Filters                        | S1       | S2       | S3       | S4       | S5       | S6       | S7       | S8       | S9       |
|-------------------------------|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| Noise                         | 21.254   | 20.8304  | 21.4105  | 21.7541  | 21.4842  | 20.7709  | 22.0348  | 21.2746  | 21.2480  |
| MF (Kumar et al., 2020)       | 23.5835  | 20.9702  | 21.7386  | 21.3483  | 21.8544  | 21.4450  | 22.3578  | 22.4255  | 22.1717  |
| HMF (Rakesh et al., 2013)     | 22.6384  | 21.2193  | 22.1395  | 21.5657  | 21.6338  | 20.6977  | 21.7301  | 22.0179  | 22.3603  |
| DAMF (Erkan et al., 2018)     | 21.8380  | 21.0934  | 21.8949  | 20.9237  | 21.6494  | 21.1822  | 21.9341  | 21.9089  | 21.8991  |
| BPDF (Erkan and Gokrem, 2018) | 21.7854  | 21.2155  | 22.0367  | 21.2063  | 21.7041  | 20.7744  | 21.6510  | 21.4882  | 21.5649  |
| NAFSM (Kenny and Nor, 2010)   | 22.0736  | 20.9317  | 22.3584  | 21.1576  | 21.6052  | 21.0105  | 21.8615  | 21.8893  | 21.3787  |
| OP (Kumawat and Panda, 2021)  | 24.1616  | 21.7762  | 23.0209  | 21.7458  | 22.5138  | 21.6926  | 22.9074  | 22.9058  | 22.7197  |

Table 6  SNR applied on various sampled 90 images contains speckle noises on 70dB noise density

| Filters                        | S1       | S2       | S3       | S4       | S5       | S6       | S7       | S8       | S9       |
|-------------------------------|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| Noise                         | 7.9052   | 8.2609   | 8.8536   | 8.6944   | 8.4522   | 8.3511   | 8.0974   | 7.9798   | 7.9587   |
| MF (Kumar et al., 2020)       | 9.9232   | 9.8767   | 11.4688  | 10.3287  | 10.3660  | 10.1669  | 10.0831  | 9.7176   | 9.6928   |
| HMF (Rakesh et al., 2013)     | 9.8232   | 9.8185   | 11.3528  | 10.2111  | 10.3137  | 10.1417  | 10.0051  | 9.7115   | 9.5837   |
| DAMF (Erkan et al., 2018)     | 11.091   | 10.8589  | 11.3586  | 11.2138  | 11.2675  | 11.0694  | 11.1843  | 10.9402  | 10.9192  |
| BPDF (Erkan and Gokrem, 2018) | 10.8553  | 10.7660  | 11.0729  | 11.0163  | 10.9624  | 10.8267  | 10.8032  | 10.6477  | 10.7335  |
| NAFSM (Kenny and Nor, 2010)   | 11.7680  | 11.3268  | 11.8316  | 11.6023  | 11.7631  | 11.5405  | 11.7385  | 11.4683  | 11.4968  |
| OP (Kumawat and Panda, 2021)  | 11.7843  | 11.1348  | 14.0761  | 11.7461  | 12.3871  | 11.7643  | 12.1313  | 11.4318  | 11.3525  |

Table 7  Resolution parameter applied on various sampled 90 images contains speckle noises on 70dB noise density

| Filters                        | S1       | S2       | S3       | S4       | S5       | S6       | S7       | S8       | S9       |
|-------------------------------|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| Noise                         | 43.6     | 41.1     | 43.1     | 39.9     | 42.3     | 41.7     | 42.1     | 41.8     | 42.5     |
| MF (Kumar et al., 2020)       | 34.8     | 31.7     | 33.0     | 30.3     | 32.5     | 32.2     | 32.4     | 32.5     | 33.5     |
| HMF (Rakesh et al., 2013)     | 36.3     | 33.6     | 35.4     | 32.1     | 34.6     | 34.4     | 34.5     | 34.4     | 35.4     |
| DAMF (Erkan et al., 2018)     | 37.7     | 36.4     | 38.5     | 36.2     | 37.7     | 36.9     | 37.5     | 36.8     | 37.2     |
| BPDF (Erkan and Gokrem, 2018) | 38.8     | 37.4     | 39.6     | 37.0     | 38.7     | 37.9     | 38.4     | 37.7     | 38.1     |
| NAFSM (Kenny and Nor, 2010)   | 36.2     | 35.4     | 37.3     | 35.3     | 36.5     | 36.0     | 36.4     | 35.8     | 35.7     |
| OP (Kumawat and Panda, 2021)  | 21.6     | 20.4     | 19.1     | 18.4     | 19.7     | 20.3     | 20.6     | 20.7     | 20.8     |
Table 8  False acceptance rate (FAR) parameter applied on various sampled 90 images contains speckle noises on 70 dB noise density

| Filters                        | FA | FAR   |
|-------------------------------|----|-------|
| Noise                         | 75 | 83.33 |
| MF (Kumar et al., 2020)       | 71 | 78.88 |
| HMF (Rakesh et al., 2013)     | 69 | 76.66 |
| DAMF (Erkan et al., 2018)     | 73 | 81.11 |
| BPDF (Erkan and Gokrem, 2018) | 74 | 82.22 |
| NAFSM (Kenny and Nor, 2010)   | 76 | 84.44 |
| OP (Kumawat and Panda, 2021)  | 1  | 11.11 |

\[
FAR = \left[ \frac{\text{Number of Iris False acceptance}}{\text{Total number of Iris Acceptance}} \right] \times 100
\]  

(21)

The percentage value of the parameters are reflected in Table 4 for different filters compared with our proposed filter. For example, if we consider percentage of accuracy of the existing filters with own proposed one, at lower NDs the values are comparable. For example at ND 10dB for own proposed it is 90% and MF it is 92%, HMF-84%, DAMF-70%, BPDF-54% and NAFSM-62%. But when ND increases from 10dB to 95dB the accuracy decreases. At 95 dB MF,HMF,BPDF totally fails. But for op HRFF shows 44% accuracy. Similarly from 50 samples images, number of accurately segmented iris in case of 10 dB with op filter is 50 i.e. 100%, which is highest among all these existing filters. When the noise is 95dB, the number of accurate segmented iris in case of our filter is 24 out of 50 samples, which is also highest among all these existing filters. If we consider PSIR%, for 10 dB it is 100%, in case of op filter. And 89% in case of 30 dB noise for op filter. Similar findings can be drawn for PSPR%, IR% and PR%. PSPR% for 10dB noise, in case of op filter is 90.90909091 which is the maximum in comparison to other filters. And for 95dB noise PSIR% is 45.83333333, while PSPR% is 46.15384615 in case of op filter. For 70 dB noise IR% is 86 and PR% is 84 which is very high in comparison to other filters. So, we can conclude that at higher NDs, our filter outperforms all other filters.

We have also compared the performance of the proposed algorithm with the existing algorithms in terms of execution time and image quality parameters. It can be seen from the Table 4 that at different NDs our op filter takes less time for giving the segmentation result in comparison to all other filters. For example, 95 dB op filter takes 06.166192 seconds which is the lowest among all other filters.

Tables 5 and 6 shows PSNR and SNR value of various filters which contains 70dB speckle noise density on nine sampled of eye images having ten variation of each sample. Here, OP (Kumawat and Panda, 2021) produces higher value of PSNR and SNR as compare to existing filters which shows the result of image quality. Image quality is better when PSNR as well as SNR is high and when it is low, image quality will be degraded. In Table 7 shows the resolution of various filters with noisy images of all nine samples.

From Table 7, we can conclude that noisy segmenting images contains higher resolution and when filters applied on noisy images, resolution got decreased. Here, own proposed filter resolution value is very low as compare to existing filter which shows this own proposed filter is very efficient to remove the noise from the images as compare to others.

In Table 8 shows the false acceptance rate values of each filters with noisy image. Here, the total number of iris acceptance is 90 which is also known as total number of samples of eye images and number of false acceptance represent the term as FA. In this table, OP (Kumawat and Panda, 2021) produces values of FAR is very low as compare to exiting filters. If
Fig. 19 Nine sampled segmented speckle noisy images of each person as input from IITDelhi database with varying noise density (ND)70 based on different filters a PSNR; b SNR; c resolution; d FAR

FAR is low, it ensure that any unauthorized person will not be allowed to access otherwise unauthorized person will be authenticated and data will be loss. All the comparison of image quality parameters such as PSNR, SNR and Resolution along with accuracy parameter i.e. FAR shown in Fig. 19. Here, Fig. 19a plots the PSNR values of all filters, Fig. 19b plots the SNR values of all filters having nine samples, Fig. 19c plots the resolution value of all filters having nine samples and Fig. 19d plots FAR values of all filters using 90 images together. Figure 20 shows nine samples of six different filters where Fig. 20a–f represents filters that is MF (Kumar et al., 2020), HMF (Rakesh et al., 2013), DAMF (Erkan et al., 2018), BPDF
Fig. 20 Nine sampled segmented speckle noisy images of each person as input from IITDelhi database with varying noise density (ND) 70 based on different filters using IWM. a median filter (Kumar et al., 2020); b HMF filter (Rakesh et al., 2013); c DAMF filter (Erkan et al., 2018); d BPDF filter (Erkan and Gokrem, 2018); e NAFSM filter (Kenny and Nor, 2010); f own proposed HRFF filter (Kumawat and Panda, 2021);
(Erkan and Gokrem, 2018), NAFSM (Kenny and Nor, 2010), with own proposed(op) filter (Kumawat and Panda, 2021) applied on all samples that is from S1 to S9. That is the ICWFL edge detection in combination with HRFF can be best suited for an iris segmentation algorithm. The proposed algorithm can be implemented for real-time applications as it takes very less time for performing iris segmentation.

5 Conclusion

In the present paper, an accurate iris segmentation scheme is presented that is robust to noise. The Wilde’s iris segmentation approach is modified to get an accurate segmentation. As mentioned earlier an iris biometric system consists of four modules i.e. “image acquisition”, “iris segmentation”, “feature extraction matching and recognition”. Each module has its own importance and contribution to the accurate and reliable iris recognition. But out of these four modules, iris segmentation functions crucially in the overall system accuracy. Keeping this in mind, this paper focuses on two important aspects i.e. edge detection of iris and pupil using ICWFL method and reduction of unwanted noise with the help of HRFF. The method of edge detection and noise reduction is incorporated in Wilde’s method of iris segmentation. Performance analysis using various parameters for accuracy shows that IWM outperforms Wilde’s method of iris segmentation.

6 Future work

Future work, could be to develop a complete iris recognition system. Here, this paper focus only first two steps of iris recognition i.e. preprocessing and segmentation. In future, an automated scheme for feature extraction and matching can be developed, in order to design a complete iris recognition system. So, a reliable iris recognition system may be developed, which can be best suited to real-time applications.

References

Abdelwahed, H. J., Hashim, A. T., & Hasan, A. M. (2020). Segmentation approach for a noisy Iris images based on hybrid techniques. Engineering and Technology Journal, 38(11), 1684–1691.
Abdulwahid, H. J., Hashim, A. T., & Hassan, A. M. (2020). Segmentation approach for a noisy Iris images based on block statistical parameters. Journal of Physics: Conference Series, 1530, 012021.
Al-Taweel, H. S. R., Daway, G. H., & Kahmees, H. M. (2015). Deblurring average blur by using adaptive Lucy Richardson. Journal of College of Education, 5, 75–90.
Bhardwaj, S., & Mittal, A. (2012). A survey on various edge detector techniques. Procedia Technology, 4, 220–226.
Canny, J. F. (1986). A computational approach to edge detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, 8(6), 679–697.
Cherabit, N., Chelali, Z. F., & Djeradi, A. (2012). Circular Hough transform for Iris localization. Science and Technology, 2(5), 114–121. https://doi.org/10.5923/j.scit.20120205.02.
Daugman, J. (2004). How Iris recognition works. IEEE Transactions on Circuits and Systems for Video Technology, 14(1), 21–30. https://doi.org/10.1109/TCSVT.2003.818350.
El-Khany, E. S., Lotfy, M., & El-Yamany, N. (2000). A modified fuzzy Sobel edge detector (Vol. 17, pp. 1–9).
Erkan, U., & Gokrem, L. (2018). A new method based on pixel density in salt and pepper noise removal. Turkish Journal of Electrical Engineering and Computer Sciences, 26, 162–171. https://doi.org/10.3906/elk-1705-256.
Erkan, U., Gokrem, L., & Enginoglu, S. (2018). Different applied median filter in salt and pepper noise. *Computers and Electrical Engineering, 70*, 789–798. https://doi.org/10.1016/j.compeleceng.2018.01.019.

Gonzalez, R., & Woods, R. (2002). Image segmentation. *Digital Image Processing, 2*(2002), 331–390.

Huny, M., Pankuj, K., & Banshidhar, M. (2012). Fast segmentation and adaptive SURF descriptor for Iris recognition. In *Mathematical and computer modelling* (pp. 1–15).

Jan, F., & Min-Allah, N. (2020). An effective Iris segmentation scheme for noisy images. *Biocybernetics and Biomedical Engineering, 40*, 1064–1080.

Jeong, D. S., Hwang, J. W., Kang, B. J., Park, K. R., Won, C. S., Park, D. K., & Kim, J. (2010). A new Iris segmentation method for non-ideal Iris images. *Image and Vision Computing, 28*(2), 254–260. https://doi.org/10.1016/j.imavis.2009.04.001.

Kennedy, O., Noma-Osaghae, E., John, S., & Ajulibe, A. (2018). An improved Iris segmentation technique using circular Hough transform. *IT Convergence and Security, 2017*, 203–211. https://doi.org/10.1007/978-981-10-6454-8_26.

Kenny, K. V. T., & Nor, A. M. I. (2010). Noise adaptive fuzzy switching median filter for salt-and-pepper noise reduction. *IEEE Signal Processing Letters, 17*(3), 281–284. https://doi.org/10.1109/LSP.2009.2038769.

Khan, T. M., Kong, Y. (2022). A fast and accurate Iris segmentation method using an LoG filter and its zero-crossings. arXiv preprint arXiv:2201.06176

Kumar, N., Dahiya, K. A., & Kumar, K. (2020). Modified median filter for image denoising. *International Journal of Advanced Science and Technology (IJASt), 29*, 1495–1502.

Kumawat, A., & Panda, S. (2021). A robust edge detection algorithm based on feature-based image registration (FBIR) using improved Canny with fuzzy logic (ICWFL). *The Visual Computer*. https://doi.org/10.1007/s00371-021-02196-1.

Labati, R. D., & Scotti, F. (2010). Noisy Iris segmentation with boundary regularization and reflections removal. *Image and Vision Computing, 28*(2), 270–277. https://doi.org/10.1016/j.imavis.2009.05.004.

Li, P., Liu, X., Xiao, L., & Song, Q. (2010). Robust and accurate Iris segmentation in very noisy Iris images. *Image and Vision Computing, 28*(2), 246–253. https://doi.org/10.1016/j.imavis.2009.04.010.

Lubos, O., Jozef, G., Jarmila, P., Milos, O., & Bart, J. (2021). A survey of Iris datasets. *Image and Vision Computing, 108*, 104–109. https://doi.org/10.1016/j.imavis.2021.104109.

Malgheet, J. R., Manshor, N. B., Affendey, L. S., & Abdul Halin, A. B. (2021). Iris recognition development techniques: A comprehensive review. In *Complexity*.

Malinowski, K., & Saeed, K. (2022). An Iris segmentation using harmony search algorithm and fast circle fitting with blob detection. *Biocybernetics and Biomedical Engineering, 42*(1), 391–403.

Manchanda, N., Khan, O., Rehan, R., & Pruthi, J. (2013). A survey: Various segmentation approaches to Iris recognition. *International Journal of Information and Computation Technology, 3*(5), 419–424.

Mittal, M., Verma, A., Kaur, I., Kaur, B., Sharma, M., & Goyal, M. L. (2019). An efficient edge detection approach to provide better edge connectivity for image analysis. *IEEE Access, 7*, 33240.

Nathan, D. K., Jinyu, Z., Natalia, A., & Schmid, B. C. (2006). *Image quality assessment for Iris biometric*. https://doi.org/10.1101/12.666448.

Pathak, M., Srinivasu, N., & Bairagi, V. (2019). Effective segmentation of sclera, Iris and pupil in eye images. *Telecommunication Computing Electronics and Control (TELKOMNIKA), 17*, 101–111.

Peihua, L., & Xiaomin, L. (2008). An incremental method for accurate Iris segmentation. In *International conference on pattern recognition*, Florida, USA.

Rahmani, V., & Narouei, M. A. (2020). Automated Iris segmentation and robust features extraction based on parallel SURF feature model. In *2020 23rd International computer conference, computer society of Iran (CSICC) (Vol. 25, pp. 1–9).* https://doi.org/10.1109/CSICC49403.2020.9050083.

Rakesh, M. R., Ajeya, B., & Mohan, A. R. (2013). Hybrid median filter for impulse noise removal of an image in image restoration. *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Energy, 2*(10), 5117–5124.

Rao, S. S., Shreyas, R., Maske, G., & Choudhury, A. R. (2020). Survey of Iris image segmentation and localization. In *2020 Fourth international conference on computing methodologies and communication (ICCMC) (pp. 539–546).* https://doi.org/10.1109/ICCMC48092.2020.ICCMC-000100.

Sivaram, M., Ahamed, A., Yuvaraj, D., Megala, G., Porkodi, V., & Kandasamy, M. (2019). Biometric security and performance metrics: FAR, FER, CER, FRR. In *2019 International conference on computational intelligence and knowledge economy (ICCIKE) (pp. 770–772).*

Sunanda, S., & Shikha, S. (2015). Iris segmentation along with noise detection using Hough transform. *International Journal of Engineering and Technical Research (IJETR), 3*(5), 441–444.
Trambadia, S., & Dholakia, P. (2015). Design and analysis of an image restoration using Wiener filter with a quality based hybrid algorithms. In 2nd International conference on electronics and communication systems (ICECS) (Vol. 2, pp. 1318–1323). https://doi.org/10.1109/ECS.2015.7124798

Verma, P., Dubey, M., Basu, S., & Verma, P. (2012). Hough transform method for Iris recognition—A biometric approach. International Journal of Engineering and Innovative Technology (IJEIT), 1(6), 43–48.

Wang, C., Muhammad, J., Wang, Y., Zhaofeng, H., & Sun, Z. (2020). Towards complete and accurate Iris segmentation using deep multi-task attention network for non-cooperative Iris recognition. IEEE Transactions on Information Forensics and Security. https://doi.org/10.1109/TIFS.2020.2980791.

Wildes, R. P. (1997). Iris recognition: An emerging biometric technology. Proceedings of IEEE, 85, 1348–1363.

Xuan, L., & Hong, Z. (2017). An improved Canny edge detection algorithm. In 2017 8th IEEE international conference on software engineering and service science (ICSESS) (Vol. 8, pp. 275–278), IEEE.

Zainal, A., Zaheera, M. M., Shibghatullah, A., Yunos, S., Anawar, S., & Ayop, Z. (2013). Iris segmentation analysis using integro-differential operator and Hough transform in biometric system. Journal of Telecommunication Electronic and Computer Engineering (JTEC), 4, 1–8.

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.