Quick Median Filtering Algorithm for Denoising QR Code Images

[1]Obed Appiah, [2]Ezekiel Mensah Martey, [1]Ebenezer Quayson, [1]Eric Opoku
[1]University of Energy and Natural Resource, Sunyani-Ghana
[2]University of Mines and Technology, Tarkwa-Ghana

O. Appiah, E. M. Martey, E. Quayson, E. Opoku (2024), "Quick Median Filtering Algorithm for Denoising QR Code Images", *Ghana Journal of Technology,* Vol. 8, No. 1, pp. 11 - 20.

Abstract

QR Code is a 2D symbology technique that provides a machine-readable optical label with information on the associated item. Its usage is fast becoming ubiquitous in diverse fields of study due to the proliferation of smartphones, which have become the primary devices for scanning these codes. The operation of QR Codes is like information decoding, encompassing activities such as image capturing, image preprocessing, image segmentation, recognition and presentation of information. Issues of poor QR codes, poor camera quality, the introduction of noise into images by image-capturing sensors, communication issues, and errors in image-processing algorithms may affect the whole scanning process. Several techniques have been suggested to address the impact of these issues on a successful scanning task. One such technique is the image denoising technique, which enhances the quality of images to help reduce the chances of failed scans. Integration of image denoising techniques in the preprocessing of QR Codes for effective results by Scanners is a crucial stage in the pipelines of these QR-code scanning applications. However, the computational time complexities of these algorithms sometimes increase the throughput of Scanners. This paper proposes a novel median filtering algorithm for denoising salt and pepper noise in QR codes. Compared to the state-of-the-art median-based filtering algorithms, the proposed algorithm completes the denoising of QR Code images in the shortest possible time.

Keywords: Image denoising, Median Filtering, QR Codes, QR Scanners, Quick Filters

1 Introduction

QR (Quick Response) Code technology has gained momentum in recent times mainly because of the proliferation of smartphones, which have become the primary devices for scanning these codes (Chang, 2014). Its application has no limitations as they are seen almost every day. QR Code is a machine-readable optical label containing information about the associated item or product. The Code is a reasonably 2D symbology developed by Denso Wave Incorporated, Japan (Denso, 2021). The code was initially designed to store data about the various automotive components to facilitate the manufacturing processes at the Toyota company. However, QR Codes have gone beyond such applications and are now being used to encode information such as phone numbers, email addresses, Wi-Fi access codes, URLs, SMS messages, etc. Several QR Codes other than Module 1 can be found today.

The micro QR Codes are a typical example of improving modules 1 and 2 to reduce the space required to print QR Codes. Security QR Codes (SQRC) look the same as regular QR Codes but have a reading restriction function (Ahamed and Mustafa, 2019). A typical application of SQRC is to store private information. High-Capacity Color Barcode (HCCB) created by Microsoft Corporation (Abdullah and Aziz, 2014), High Capacity Colored 2-D Code (HCC2D) and Just Another Barcode (JAB) developed by Fraunhofer Institute SIT (Secure Information Technology) (Fraunhofer, 2020) use the concept of symbology to enhance the performance of QR Codes. Fig. 1 presents a sample QR Code.

Fig. 1: Sample QR Code.

Successful applications of QR codes can be seen in almost every domain. A project by Saumya (2021) demonstrated the use of QR Codes for forest education by tagging trees and plants with QR codes to give interested people more information on these forest products. In academia, the codes have been used extensively in managing resources such as articles, journals, and books (Rikala, 2014; Abas et al., 2015; Shettar, 2016). A study conducted by Durak et al., 2016 concluded that QR codes can enhance learner's experiences. Learners who

*Manuscript received November 5, 2023
Revised version accepted March 22, 2024*

participated in the survey reported that they were aware of QR Codes and could use them and that using QR codes in education was useful. In the healthcare industry, where accuracy and timely information are required, QR codes are also needed (Kari, 2019). However, one predominant application of QR codes is marketing products and services. You are likely to see QR codes on advertisements or posters, which direct the user to a website when scanned with a mobile phone (Thompson and Lee, 2013).

QR Codes, like other symbology, means that decoding information requires image capturing, image preprocessing, image segmentation, recognition, and presentation of information. Errors in any of these stages may result in the presentation of invalid information or the Scanner's inability to decode information. Issues of poor QR codes, poor camera quality, the introduction of noise into images by image-capturing sensors, and errors in image processing algorithms may affect the whole scanning process. To deal with noisy QR codes, several works have proposed techniques to denoise images to help minimize the chances of failed scans (Appiah et al., 2021).

Noisy QR Codes pose challenges for QR code scanners to decode. In their work "Impact of Denoising using Various Filters on QR Codes", Kumawat et al. (2013) demonstrated the impact of Gaussian, Speckle, and Salt-and-pepper noise on QR Codes. Weiner, Mean and Median filtering methods were used to denoise QR Codes to help improve performance. The desire to enhance QR Codes images and reduce the impact of noise on the performance of applications has been studied extensively by these authors (Van-Gennip et al., 2015; Bhatnagar and Bhatnagar, 2018; Wang et al., 2019; Tsai and Hsieh 2019) with exciting outcomes. The works have demonstrated the need to integrate image denoising techniques in the preprocessing of QR Codes for effective results by Scanners. However, the computational time complexities of these algorithms sometimes increase the throughput of Scanners.

In this paper, a novel median filtering algorithm is designed to denoise salt-and-pepper noise in QR Code images. Compared to existing state-of-the-art median-based filtering algorithms, this new approach achieves the denoising process in the shortest possible time.

2 Related Review

Median filtering algorithms exploit the kernel's rank-order information of pixel intensities and replace the central pixel with median values. It is one of the most robust filtering approaches for handling impulse or salt-and-pepper noise (Senthil and Sukumar, 2019; Mafi et al., 2019; Liang et al., 2021). The filter has seen various variations over the last four decades. However, one of its main challenges is selecting the median value (Aranda et al., 2017; Erkan et al., 2018; Qi et al., 2021). The time-consuming selection process contributes to median-based filtering algorithms' high time complexity (Appiah et al., 2016).

Suomela (2014) demonstrated that the median filtering problem is tantamount to the sorting problem theoretically and practically. The repetitive comparisons of pixel values while estimating the median value contribute to the high time complexity of sort-based median filtering algorithms. Over the past three decades, alternative approaches have been proposed to enhance the runtime efficiency of median filtering. One such approach proposed by Singh (2011) as an Alternative Algorithm for 3 x 3 median filtering of digital images, demonstrated a reduction in excessive comparisons, resulting in a decreased running time of median selection to $O(nlogn)$. This algorithm showcases superior performance compared to classical 3 x 3 kernel size for median filtering. Two paradigms have been observed in the literature aiming to enhance the running time of median filtering for digital images. Approximation techniques typically reduce the steps or number of pixel values utilized to estimate median values, achieving lower running times, while histogram-based approaches utilize histograms for median estimation.

A Directional Median Filter (DMF), a stick median filter, functions by dividing the filtering kernel or window into 1D stick components. The estimated median values from these 1D components are combined to produce the final result. Several researchers have suggested different methods for integrating the results from the sticks to generate the final value used to replace the central value of the window or kernel. Equation 1 was utilised to approximate the median value for the window (Czerwinski et al., 1995).

$$S(i,j) = max\ \{Median(k,l) \epsilon\ W\theta\ \{D(i+k, j+l)\} \quad (1)$$

W: Sliding window
D: Direction in a window
k: row number
l: column number
i: row index of the active window
j: column number of active windows

Lu et al. (2010) introduced the Sort Optimization Algorithm of Median Filtering (SOAMF). Their approach involves employing various methods to handle each stick and amalgamating their outcomes to estimate the median for the central value of a window, aiming for effective results.. The SOAMF method is delineated by Equation 2.

$$Fmed = med\ (Amin,\ Bmed,\ Cmax) \quad (2)$$

$Amin$: Minimum value in the first column
$Bmed$: Median value in the second Column
$CMax$: Maximum value in the third Column
$Fmed$: Median of Amin, Bmed, and Cmax.

Marcus and Ward (2013) proposed a non-discrete algorithm that rapidly approximates a median filter. This algorithm's effectiveness stems from a technique prohibiting overlapping columns for selecting median pixel values as the window or kernel slides. Fig. 2 illustrates Marcus and Wards' Approximated algorithm. While this algorithm effectively denoises images with a lower density of impulse noise, its performance diminishes as the image density increases. Nonetheless, its running time is significantly better than that of the classical median filtering algorithm.

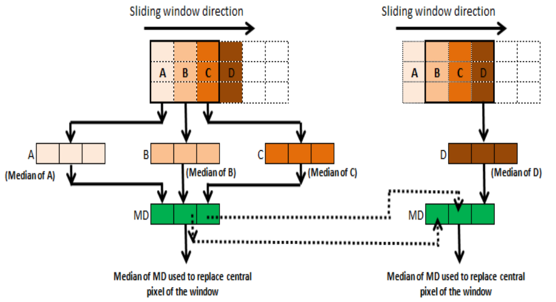


Fig. 2: An illustration of DP: a Fast Median Filter Approximation

Appiah et al. (2022) proposed an improved approximated median filter algorithm for real-time computer vision applications, which tackles the challenge of the DP: a Fast Median Filter Approximation. In the work, the concept of mid-Value Decision-Median (MVDM) is proposed to help improve the quality of the denoised images as the image's noise density increases.

The approximation techniques also use several comparisons of pixel values to estimate median values. These comparisons and exchange of values on a list make computational time complexity of the approximated median filtering relatedly expensive. The histogram-based approach has been proposed as an alternative to achieve real-time image denoising of impulse noise. The method works by simply constructing a histogram for a window and using the generated histogram to estimate the median.

Initial work by Ahmad and Sundararajan (1987) proposed a fast two-dimensional median filtering algorithm. The algorithm constructs a histogram for the left-most kernel and estimates the median using a histogram. Subsequent estimation of the median values relies on an initially constructed histogram. The leftmost pixel values of the last processed window are deleted from the histogram, while the rightmost values of the current window are added to the histogram. The resulting histogram is used to estimate the median value for the denoising. Weiss (2006) and Perreault & Hébert (2007) improved the method further to reduce the running time of Huang's algorithm. Fig. 3 presents a pictorial presentation of how the kernel histogram is updated by adding the modified column histogram and subtracting the leftmost one.

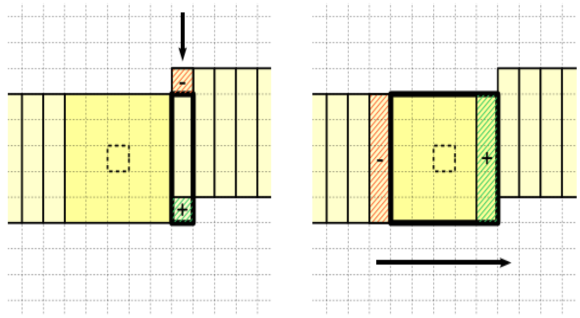


Fig. 3: Two steps of Perreault et al. (2007) algorithm. (a) The column histogram to the right is moved to one row by adding one pixel and subtracting another. (b) The kernel histogram is updated by adding the modified column histogram and subtracting the leftmost one (Appiah et al., 2022).

Zhu and Huang (2012) presented a histogram-based median filtering approach capable of achieving O(n) time complexity by utilizing a statistical histogram to speed up selecting a median. Zhang et al. (2014) proposed a joint-histogram technique for estimating median values in median filtering. This algorithm operates with a time complexity of O(r), where 'r' is associated with the radius of the window used for the filtering process.

Bae and Yoo (2018) suggested enhancements to histogram-based approaches for median filtering tasks. Unlike Zhu's method, which scans through the histogram from 0 to k-1 (where k is the number of bins on the histogram) to identify the median value, typically resulting in O(n + k) running times, their approach utilizes the location of the previously identified median as the starting point for the search of current median values on the histogram. This modification significantly reduces the running time.

Indeed, all histogram-based median filtering techniques require additional space to manage the histogram. Moreover, generating location addresses for the source and destination indexes can significantly increase the time the algorithm needs to complete the denoising task. This challenge, coupled with the relatively high number of comparison operations in the approximated method, adversely affects real-time performance, particularly in denoising QR Codes. Prompt processing of QR Code images is essential for users to quickly derive correct information. Therefore, an algorithm that operates swiftly ensures a practical user experience. To address this need, we propose a Quick Median Filtering Algorithm that circumvents excessive comparisons or histograms to estimate median values, resulting in improved computational time complexity for denoising QR Code images. This technique enables QR Codes to be decoded in real-time, fulfilling Quick Response's purpose.

3 Proposed Algorithm

In this section, we present the proposed algorithm. The proposed algorithm (Quick Median Filter) is designed to achieve real-time QR Code denoising. The fundamental principles of the algorithm are presented as:

- It is a median filter
- The algorithm works on only binary images
- Does not use rank order in the selection of median values
- Does not use counting or frequency estimation to determine median values
- Rely on estimated median values from adjacent windows to estimate the median value of an active window as the filtering kernel slides from the left side of the window to the right.

Binary images usually have two intensity levels in the image. Let α and β represent the intensity values in a binary image, such that $\alpha < \beta$. To determine the median value of a given window of n elements, the rank-order (sort-based) method will arrange the intensity values as:

$$\text{Sort} = [x_1 \leq x_2 \leq x_3 \leq x_4 \leq \ldots \leq x_n]$$

The value situated in the middle is selected as the median for the window. Given $\alpha < \beta$, if x_i is β, then x_{i+1} to x_n will all be β, while x_1 to x_{i-1} can be α or β. Estimating the median after the ordering can be done using equation 3.0.

$$m = \left(\frac{n+1}{2}\right) \tag{3}$$

Median = x_m
n: number of elements in the window
m: median value index

Binary images present an interesting means of estimating the median of a window. We can use the counting of intensity values to estimate the median of a window. By this, the intensity value with the highest frequency is used as the median value. This process is simple and has low computational complexity. Equation (4) is used to estimate the median.

$$\text{Median} = \begin{cases} \beta, & {}_n\beta \geq m \\ \alpha, & {}_n\alpha \geq m \end{cases} \tag{4}$$

${}_n\alpha$: Number of α values in the window
${}_n\beta$: Number of β values in the window

The above concept of estimating the median value forms the basis for the proposed Quick Median filtering algorithm. The proposed method estimates a value *Pr* (Parameter) for each window using equation (5).

$$Pr = {}_n\beta \,.\, \beta \tag{5}$$

Assuming that α is zero (0), then *Pr*, which is the number of β in a window multiplied by β value, can also be estimated by adding all the β values in the window. If the assumption that α is zero (0), then summing all the elements in a window will be the same as adding all β elements. Equation (6) presents the summation technique of estimating *Pr*.

$$\text{Pr} = \sum_{\substack{1<i<w \\ 1<j<w}} K(i,j) \tag{6}$$

Pr: Parameter defined in equation 5.0
K: Sliding window
w: sliding window width or height

Given all the conditions defined above, equation (7) can estimate the median for a given window.

$$\text{Median} = \begin{cases} \beta, & \text{Pr} \geq \left(\frac{n+1}{2}\right)\beta \\ \alpha, & \text{Pr} < \left(\frac{n+1}{2}\right)\beta \end{cases} \tag{7}$$

The Quick Median Algorithm is presented next.

Proposed Algorithm – Quick Median Algorithm

Input: Image X of size M x N, kernel size of n x n, and radius of r

Output: Image Y of the same size as X

Initialize *Pr* to zero (0)
Pad the image on all the sides of X with columns and rows as r
for *i* = r to *M* +1 – r **do**

$$\Pr \leftarrow \sum_{\substack{-r<x<r \\ -r<y<r}} I(x+i+1, y+r+1)$$

if $\Pr \geq \left(\frac{n+1}{2}\right)\beta$

$Y_{i,j} \leftarrow \beta$

else

$Y_{i,j} \leftarrow \alpha$

for j = r to N +1 – r **do**

for k = -r to r **do**

$\Pr \leftarrow \Pr - X_{i+r,\ j-r-1}$

$\Pr \leftarrow \Pr + X_{i+r,\ j+r}$

end for

if $\Pr \geq \left(\frac{n+1}{2}\right)\beta$

$Y_{i,j} \leftarrow \beta$

else

$Y_{i,j} \leftarrow \alpha$

end for

end for

The proposed algorithm uses previously computed summation values as the window slides from the image's left to right. This technique helps significantly reduce the computational time required to estimate the median. Theoretically, the computational time complexity of the algorithm can be estimated as less than O(n).

3.1 Evaluation of Proposed Algorithm

3.1.1 Count-based algorithm

This technique of selecting median values for denoising images estimates the pixel values with the highest occurrence in a window or kernel. Binary images have 2 possible pixel intensity values that can be found in a given window. The technique counts the number of times each pixel value occurs and selects the pixel value with the highest frequency as the median value for the denoising task.

3.1.2 Histogram-I (Hist-I)

This technique is technically the same as the Count technique previously described; however, the method uses a histogram to help estimate the pixel value with the highest frequency in a window.

3.1.3 Histogram-II (Hist-II)

HIST-II is an improvement on Hist-I. It is the binary image version of the denoising algorithm proposed by these authors [19, 27].

3.1.4 *Summation (Sum)*

The summation approach is another approach to the Count technique described earlier. This technique sums the elements in a window and evaluates the result to determine which elements occur more. This technique avoids excessive comparisons in the median selection process, as seen in the Count approach. It is similar to the proposed algorithm but does not rely on adjacent window or kernel median value for estimation of the median value of an active window.

4 Results and Discussions

In this section, we present the results of the running times of the various median selection approaches for denoising salt & pepper noise in QR Code images. Count, Sum, Hist-I, Hist-II, and Proposed Algorithm running times estimated using MatLab. Images of sizes 550 x 550 were used for the experiment, and the MatLab tic and toc functions were used to estimate the running times of the algorithms. The measurements were done 1000 times for each algorithm, and the average running times in Seconds were presented. The relative running times to help determine the performance of the proposed algorithm against the existing ones are also presented. The Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Metric (SSIM) determine which window size will be ideal for denoising QR code images.

4.1 Running Times Results

Fig. 3 presents the average running time results for denoising images using a window size of 3 x 3. The proposed algorithm outperformed all the other techniques. It could run 1.7 times faster than Hist-II for the window size of 3 x 3.

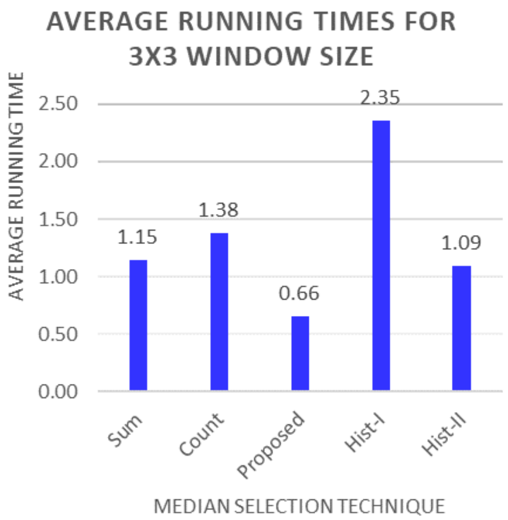


Fig. 3: Average running times of denoising algorithms with 3 x 3 window size

Fig. 4 presents the results of using a window size of 11 x 11. Again, the proposed algorithm outperformed the other four techniques, running 1.8 times faster than Hist-II.

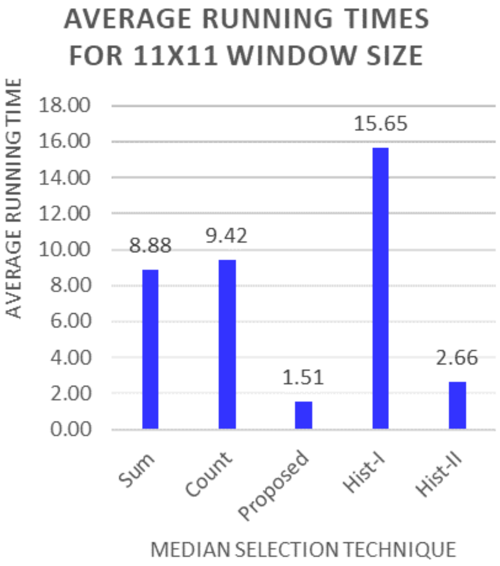


Fig. 4: Average running times of denoising algorithms with 11 x 11 window size

Fig. 5 presents the running times of the various algorithms with 5 window sizes. The proposed algorithm outperforms all the algorithms. Again, the results show that the performance of Hist-I, Sum, and Count degrades with increasing window size. Hist-I has the worst observed running time, largely because creating histograms involving access arrays increases the algorithm's running time.

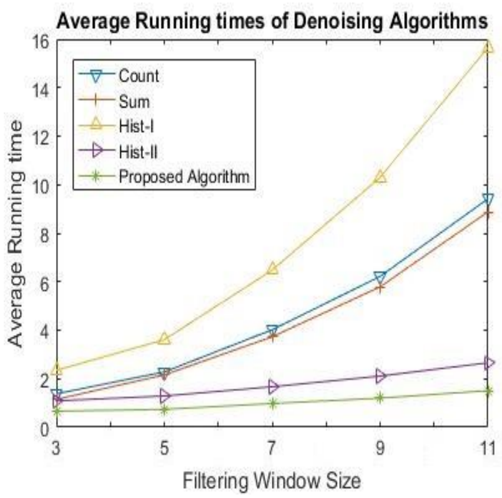


Fig. 5: Average running times of denoising algorithms using windows widths of [3, 5, 7, 9, and 11]

Fig. 6 presents the relative running times of the four (4) algorithms compared with the proposed algorithm.

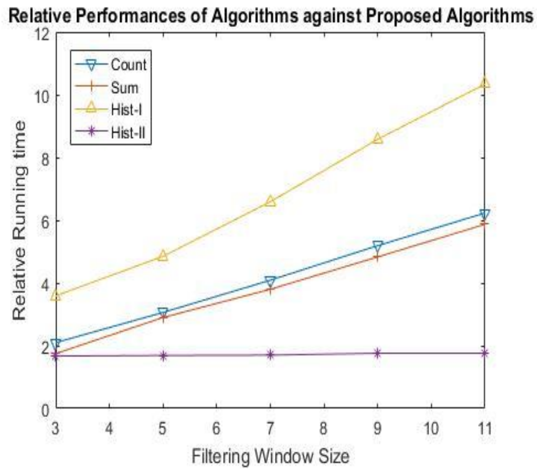


Fig. 6: Relative running times of 4 denoising algorithms against the proposed using windows widths of [3, 5, 7, 9, and 11]

From the relative times, Hist-II has the closest running time to the proposed algorithm. However, it takes about twice the time of the proposed algorithm for filtering QR Codes. The time is higher because it uses histograms to estimate the median.

4.2 Algorithm's Output Performance with Different Kernel Sizes

The Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Metric (SSIM) are used to quantitatively evaluate the output generated by the proposed image processing algorithm. The metrics measure how close the denoised images are to the original image.

4.2.1 Peak Signal-to-Noise Ratio (PSNR)

The peak signal-to-noise error measurement technique is used to evaluate the performance of the proposed filtering algorithm. The Peak Signal-to-Noise Ratio (PSNR) is illustrated in Equation 8). The MSE in the equation is the Mean Square Error, presented in equation (9).

$$PSNR = 10\log_{10}\left(\frac{255^2}{MSE}\right) \quad (8)$$

$$MSE = \frac{\sum_{M,N}[I_1(m,n) - I_2(m,n)]^2}{M * N} \quad (9)$$

M :- is the number of pixel rows in the image.

N: - is the number of pixel columns in an image.

I_1 :- is the noisy image while

I_2: - is the filtered image.

m: - current row number

n: - current column number

4.2.2 Structural Similarity Index Metric (SSIM)

It is designed to enhance classical image comparison methods such as the Mean Squared Error (MSE) and Peak Signal-to-Noise Ratio (PSNR). The SSIM index ranges between -1 and 1, where a value of 1 indicates perfect structural similarity between images. Equation 10 presents the SSIM formula.

$$SSIM(I, I) = \frac{(2\mu\mu_W + C_1)(2\sigma(2\mu\mu_W + C_2))}{(\mu^2 + \mu_W^2 + C_1)(\sigma(I)^2 + \sigma(I_W)^2 + C_2)} \qquad (10)$$

Fig. 7 presents the average PSNR results for various denoised QR code images with 10 window sizes. Nineteen (19) QR codes were generated, each with a different number of characters or numeral encoded. The number of characters or numerals started from 100 to 1000 with a step of 50. As the numerals or characters encoded in QR Codes increase, it becomes challenging to denoise the images. PSNR values decrease with an increasing number of characters with the 10 window sizes used. A window size of 5 x 5 performs best among all the other window sizes in most cases.

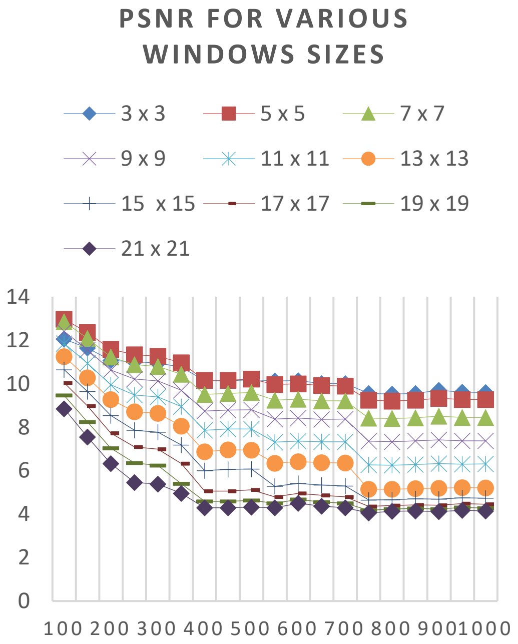


Fig. 7: PSNR results for denoising QR Codes

SSIM results for the 10 window sizes and the 19 different QR Code image types are presented in Fig. 8. Again, it is observed that the SSIM values decrease with the increasing number of characters or numerals in the QR Code. The average performance of window size of 5 x 5 outperformed all the sizes.

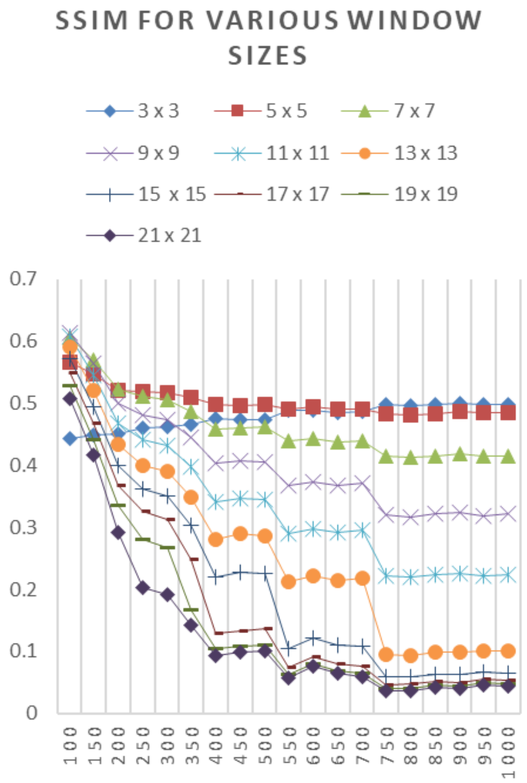
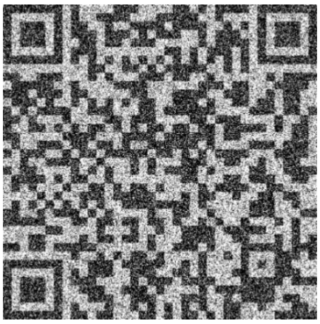


Fig. 8: SSIM results for denoising QR Codes

The summation method avoids excessive comparisons in selecting the median value to aid denoising. Again, avoiding using histograms reduces the space and time complexity required to estimate the median value. This resulted in the running time of the proposed algorithm performing better than the rest of the techniques. The work did not compare with any sort-based ones because they generally have high time and space complexities, whether classical or approximated, as a result of comparing and exchanging values during the estimation of median values. The proposed method can run better than these types of median filters. PSNR and SSIM values confirmed the accuracy of the proposed algorithm since it generated the same output as the standard median filter. From the PSNR and SSIM results in Figs 7 and 8, a window size of 5 x 5 outperforms all the window sizes used in the experiment. Fig. 8 presents sample QR Codes.

(a)



(b)

(c)

Fig. 8: (a) Original QR Code, (b) Corrupted QR Code (c) Denoise QR Code using Quick Median

5.0 Conclusion

In this work, a novel image-denoising algorithm is proposed to denoise QR Codes corrupted with salt & pepper noise. The Quick Median Filter takes advantage of the binary nature of QR Code images to accurately estimate median values for denoising images. The summation approach used to estimate the median value makes it possible for the algorithm to complete faster than the sort approximation or histogram approaches. Experimental results of this method compared with non-sort methods for estimating the median indicate that the proposal is best for running time among the state-of-the-art median-based image filters. The closest filter, Hist-II, has a running time of about twice that of the proposed method. The Quick Median algorithm can therefore be used to effectively denoise QR codes without impeding the running time of QR code scanners.

References

Abas, H., Yahya, F. H. and Kamaruddin, M. (2015), “User Readiness Evaluation of Qr Codes In Mobile Learning (m-learning)”. *International Conference on Information Technology and Society*. Lugano, Switzerland. pp. 126-136.

Abdullah, A. M. and Aziz, R. H. A. (2014), “Evaluating the use of Quick Response (QR) code at Sulaimani University libraries”. *International Journal of Advanced Research in Computer Science and Software Engineering*, Vol. 4, No. 11, pp. 62-72.

Ahamed, M. S. and Mustafa, H. A. (2019), “A Secure QR Code System for Sharing Personal Confidential Information*”. In 2019 International Conference on Computer, Communication, Chemical, Materials and Electronic Engineering* (IC4ME2), Rajshahi, Bangladesh, pp. 1-4. IEEE.

Ahmad, M., and Sundararajan, D. (1987), “A Fast Algorithm for Two-Dimensional Median Filtering”. *IEEE Transactions on Circuits and Systems*, Vol 34, No. 11, pp. 1364-1374.

Appiah, O., Asante, M. and Hayfron-Acquah, J. B. (2016), “Adaptive Approximated Median Filtering Algorithm for Impulse Noise Reduction”. *Asian Journal of Mathematics and Computer Research*, Vol 12, No. 2, pp. 134-144.

Appiah, O., Asante, M. and Hayfron-Acquah, J. B. (2022), “Improved Approximated Median Filter Algorithm for Real-Time Computer Vision Applications”. *Journal of King Saud University-Computer and Information Sciences*, Vol 34, No. 3, pp.782-792

Appiah, O., Martey, E. M., Ninfaakanga, C. B., Awarayi, N. S. and Opoku, E., (2021), “Performance Evaluation of Corrupted QR Code Scanners”. In 8th International Conference on Adaptive Science and Technology (ICAST), Accra, Ghana, pp. 1-6. doi: 10.1109/ ICAST52759.2021.9682029

Aranda, L. A., Reviriego, P. and Maestro, J. A. (2017), “Error Detection Technique for a Median Filter”. *IEEE Transactions on Nuclear Science*, Vol. 64, No 8, pp. 2219-2226.

Bae, J. and Yoo, H. (2018), “Fast Median Filtering by Use of Fast Localization of Median Value”. *International Journal of Applied Engineering Research*, Vol. 13, No. 12, pp. 10882-10885.

Bhatnagar, I. and Bhatnagar, S. (2018), “QR Code Denoising using parallel Hopfield Networks”. *arXiv preprint*, doi: https://doi.org/ 10.48550 /arXiv.1812.01065.

Chang, J. H. (2014), “An Introduction to using QR Codes in Scholarly Journals”. *Sci Ed*, Vol. 1, No. 2, pp. 113-117.

Czerwinski, R. N., Jones, D. L. and O'Brien, W. D. (1995), “Ultrasound Speckle Reduction by Directional Median Filtering”. *In Proceedings., International Conference on Image Processing*, Vol. 1, pp. 358-361.

Denso W. (2021), “Denso Wave, the inventor of QR Code. QRcode.com”. https://www.qrcode. com/en/ind-ex.html. Accessed: 24 September 2021.

Durak, G., Ozkeskin, E. E. and Ataizi, M. (2016), “QR Codes in Education and Communication”. *Turkish Online Journal of Distance Education*, Vol. 17, No. 2. doi:https://doi.org/10.17718 /tojde.89156.

Erkan, U., Gökrem, L. and Enginoğlu, S. (2018), “Different Applied Median Filter in Salt and

Pepper Noise". *Computers and Electrical Engineering*, Vol. 70, pp. 789-798.
Fraunhofer Sit. (2020), "JAB Code". https://jabcode.org/, Accessed: 24 September 2021.
Karia, C. T., Hughes, A. and Carr, S. (2019), "Uses of quick response codes in healthcare education: a scoping review". *BMC Medical Education*, Vol. 19, No. 1, pp. 1-14.
Kumawat, D., Singh, R. K., Gupta, D. and Gupta, S. (2013), "Impact of denoising using various filters on QR code". *International Journal of Computer Applications*, vol. 63, No. 5, pp. 0975-8887.
Liang, H., Li, N., and Zhao, S. (2021), "Salt and Pepper Noise Removal Method Based on a Detail-Aware Filter". *Symmetry*, Vol 13, No. 3, pp. 515.
Lu, Y., Dai, M., Jiang, L. and Li, S. (2010), "Sort Optimization Algorithm of Median Filtering Based On FPGA". *In 2010 International Conference on Machine Vision and Human-machine Interface*, Kaifeng, China, pp. 250-253. doi: 10.1109/ MVHI.2010.145
Mafi, M., Martin, H., Cabrerizo, M., Andrian, J., Barreto, A. and Adjouadi, M. (2019), "A Comprehensive Survey on Impulse and Gaussian Denoising Filters For Digital Images". *Signal Processing*, Vol. 157, pp. 236-260.
Marcus, R. C. and Ward, W. C. (2013), "DP: A Fast Median Filter Approximation". *Tech Rep* Los Alamos National Lab, pp. 1-11.
Perreault, S. and Hébert, P. (2007), "Median Filtering in Constant Time". *IEEE transactions on image processing*, Vol 16, No. 9, pp. 2389-2394.
Qi, Y., Yang, Z., Sun, W., Lou, M., Lian, J., Zhao, W., Deng, X. and Ma, Y. (2021), "A Comprehensive Overview of Image Enhancement Techniques". *Archives of Computational Methods in Engineering*, pp. 1-25.
Rikala, J. (2014), "Evaluating QR Code Case Studies Using a Mobile Learning Framework". *International Association for the Development of the Information Society*.
Saumya B. S. B. (2021), "Use QR codes on nature trails to make them more engaging". QR Batch Blog. https://scanova.io/blog/qrbatch/qr-codes-on-nature-trails/. Accessed on 24 September 2021.
Senthil Selvi, A. and Sukumar, R. (2019), "Removal of Salt and Pepper Noise from Images Using Hybrid Filter (Hf) and Fuzzy Logic Noise Detector (FLND)". *Concurrency and Computation: Practice and Experience*, Vol. 31, No 12, pp. 4501.
Shettar, I. M. (2016), "Quick Response (QR) Codes in Libraries: A case study on the use of QR codes in the Central Library, NITK". *In Proc. TIFR-BOSLA National Conference on Future Librarianship* (pp. 129-134).
Singh, S. (2011), "An Alternate Algorithm for (3x3) Median Filtering of Digital Images". *International Journal of Computers & Technology*, Vol. 1, No. 1, pp. 7-9.
Suomela, J. (2014), "Median filtering is equivalent to sorting". *arXiv preprint*. doi: https://doi.org/10.48550/arXiv.1406.1717
Thompson, N. and Lee, K. (2013), "Information security challenge of QR codes". *Journal of Digital Forensics, Security and Law*, Vol. 8, No. 2, pp. 2.
Tsai, M. J. and Hsieh, C. Y. (2019), "The visual color QR code algorithm (DWT-QR) based on wavelet transform and human vision system". *Multimedia Tools and Applications*, Vol 78, pp. 21423-21454.
Van-Gennip, Y., Athavale, P., Gilles, J. and Choksi, R. (2015), "A Regularization Approach to Blind Deblurring and Denoising of Qr Barcodes". *IEEE Transactions on Image Processing*, Vol 24, No. 9, pp. 2864-2873.
Wang, W., Wen, S., Wu, C. and Deng, J. (2019), "Denoising Piecewise Constant Images with Selective Averaging and Outlier Removal". *Numerical Mathematics: Theory, Methods & Applications*, Vol 12, No. 2, pp. 467. doi:10.4208/nmtma.OA-2017-0130.
Wang, W., Wu, C., & Deng, J. (2019), "Piecewise Constant Signal and Image Denoising Using a Selective Averaging Method with Multiple Neighbors". *Inverse Problems and Imaging*, Vol 13, No. 5, pp. 903.
Weiss, B. (2006), "Fast median and bilateral filtering". *In ACM SIGGRAPH* Vol. 2006, pp. 519-526.
Zhang, Q., Xu, L. and Jia, J. (2014), "100+ times Faster Weighted Median Filter (WMF)". *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2830-2837.
Zhu, Y. and Huang, C. (2012), "An Improved Median Filtering Algorithm for Image Noise Reduction". *Physics Procedia*, Vol 25, pp. 609-616.

Author

O. Appiah is a Senior Lecturer at the University of Energy and Natural Resources (UENR) - Sunyani. He is also a research scientist with the West African Science Service Centre on Climate Change and Adapted Land Use (WASCAL). He holds a PhD in Computer Science from KNUST. His research interest focuses on Computer Vision, Machine learning and Digitalisation.

E. M. Martey holds the position of lecturer in the Department of Railways and Allied Engineering Programmes at the School of Railways and Infrastructure Development, Essikado. He holds Doctor of Engineering in Software Engineering, having earned his degree from the University of Electronic Science and Technology of China, PR China. His research expertise spans various domains, including information retrieval, pattern recognition, computer graphics, image processing, and computer vision. His contributions to these areas reflect a commitment to advancing knowledge and innovation within the field of software engineering.

E. Opoku received his MPhil. Degree in Computer Science from Kwame Nkrumah University of Science and Technology, Ghana, in 2022. He is currently working as a Complaint Administrator with Savana Signatures in Ghana. His research interests include machine learning and engineering, primarily deep learning, generative models, computer vision, and image processing. Recently, he has been focused on the reasoning through relational dependencies in healthcare and agricultural data for recommendations and diagnosis.

E. Quayson is a lecturer at the Department of Computer Science and Informatics, School of Sciences, University of Energy and Natural Resources (UENR). He holds MPhil. Computer Science from the KNUST. His research interest includes data Science, Internet of Things, Deep Learning (Optimizing image classification with Capsule Network)