A Modified Inexact Arithmetic Median Filter for Removing Salt-and-Pepper Noise From Gray-Level Images

M. Monajati and E. Kabir

***Abstract*—We have recently proposed approximate median filters (APMFs). They are based on the sorting network and achieve acceptable image quality under low-cost hardware. In this brief, we develop a specific comparator to improve the capabilities of those filters in noise elimination. The architecture of our inexact median filters (IMFs) is regular and modular. Also, we introduce the histogram-based error dispersion plot as a new error evaluation method to have a better assessment of IMF performance. Simulation results show that the proposed filter is effectively low cost in power, area, and speed. Despite the tradeoff between the filtering accuracy and circuit characteristics, the output quality of the filter is largely similar to that of the precise one. Also, the degradation is almost not noticeable to the human eye.**

***Index Terms*—Gray-level image, high speed, inexact arithmetic, low cost, low power, magnitude comparator, median filter, noise suppression, salt-and-pepper noise.**

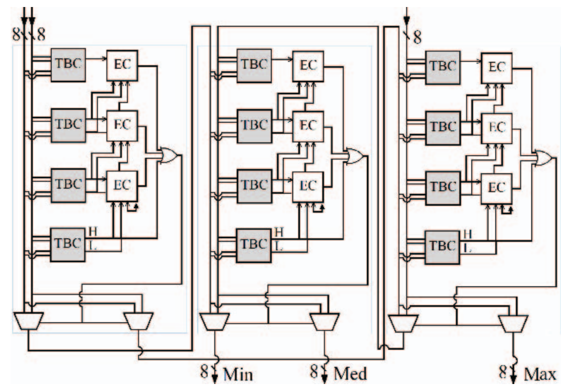


Fig. 1. Schematic of Ternary Data Sorter (TDS), TBC and EC are two-bit magnitude comparator, and equality checker, respectively.

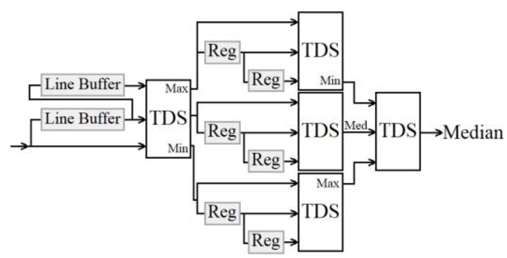


Fig. 2. Block diagram of a TDS-based median filter.

I. Introduction

NOISE Suppression is widely used to remove unwanted noise while preserving the image details. Salt and pepper noise is a special kind of impulse noise, where the noisy pixels can take only the maximum and the minimum values in the dynamic range. It appears as randomly scattered black-and-white dots over gray-level images [1].

The median filter is considered as the utmost preference for removing salt and pepper noise. It can eliminate low density impulse noise effectively [2] and is considered as a state-of-the-art method for removing salt-and-pepper noise in signal processing applications such as video and audio processing; Thus, its realization must provide high-speed processing; Moreover, low-power design is crucial in many computer vision applications.

Although the basic implementation of median filter is easy, its hardware complexity is high. Given that these filters are intrinsically error-tolerant, using approximate calculations can lessen hardware resources, in exchange for a decrease in accuracy [3], [4], [5], [6].

We have recently proposed the approximate Median Filters (APMFs) [5]. They are based on sorting networks using imprecise magnitude comparators in a pipeline manner. The main part of a sorting-network-based median filter is the triple data sorter (TDS). Fig. 1. shows the internal block diagram of TDS. In this figure, TBC is a two-bit comparator unit. The first top TBCs compare the least significant bits. Each EC block checks the equality of the previous more significant bits [5]. Fig. 2 represents the architecture of a pipeline median filter, based on a 3×3 TDS, which can process a 3×3 window elements all together [5]. According to Figs. 1 and 2, there are 12 comparators in the filter structure. Therefore, the physical characteristics of the median filter are significantly affected by those of TBC.

APMFs achieve acceptable image quality under low-cost hardware requirements on salt-and-pepper noise removal. The output qualities of different APMFs depend on the approximation as well as the image [5]. Error dispersion Plot (EDP) is an important qualitative measure in evaluating approximate comparators, representing the error characteristics in a more informative way [5]. However, the quality of an approximate filter also depends on the image, that cannot be inferred from

Manuscript received February 11, 2019; revised April 24, 2019; accepted May 25, 2019. Date of publication May 29, 2019; date of current version April 1, 2020. This brief was recommended by Associate Editor L.-P. Chau. *(Corresponding author: M. Monajati.)*

The authors are with the Department of Electrical and Computer Engineering, Tarbiat Modares University, Tehran 14115111, Iran (e-mail: mehrnaz.monajati@modares.ac.ir; kabir@modares.ac.ir).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSII.2019.2919446

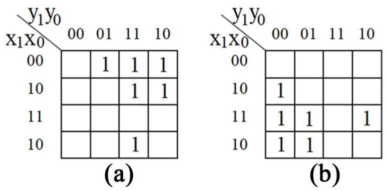


Fig. 3. K-maps of TBC for: a) L and b) H.

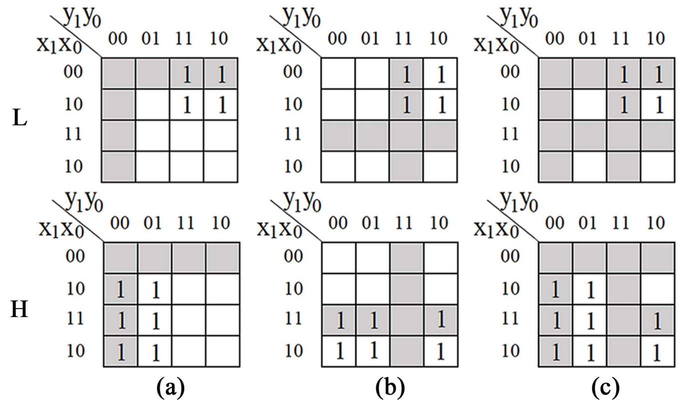


Fig. 4. Imprecise K-maps of TBC for removing: a) peppers, b) salts, c) salts and peppers.

EDP. Furthermore, none of the APMFs can remove salt-and-pepper noise as good as the precise one, especially in full approximation. That they leave a significant amount of noisy pixels in the image, compared to the exact filter.

In this brief, we extend the imprecise arithmetic used in [5] to design a new specific magnitude comparator set in a sorting network-based median filter. We apply the approximation such that the filter can remove salt and pepper noise as much as possible.

The proposed inexact median filter (IMF) achieves largely acceptable image quality under low-cost hardware requirements. an important advantage of IMF compared with APMFs is its precise noise cancellation. Also, we specifically propose IMFS to remove salts, and IMFP to eliminate peppers, with further hardware complexity reduction. Furthermore, by involving the histogram of the image in EDP, we introduce a new error evaluation method called histogram-based-EDP (HEDP) to have a better view of IMF performance. Also, we present a formula to pre-characterize the IMF. It helps the designer to find out the proper inexact filter for the desired application.

II. Proposed Imprecise Arithmetic Median Filter (IMF)

As explained in the introduction, a simpler TBC leads to a less complex filter. When the logic function is less complicated, fewer resources are required to be implemented, which reduces switching activity and consequently decreases the power consumption [5]. Similar to [5], we simplify the logic implementation of TBC by inserting minor error in its truth table. The main difference of our method with [5] is that we apply the approximation in a conscious way so that there is no error in the comparison of 8-bit numbers with salts and peppers. Although, [5] proposes several simple magnitude comparator units, APMFs cannot eliminates the noise as good as the exact filter. Compared with APMF, our inexact filter can remove salt-and-pepper noise exactly like the precise one.

We employ imprecision to achieve simplified arithmetic units with low complexity to save more power than using conventional low-power design techniques. However, the IMF image quality is less than that of the exact filter, but it should be acceptable from the perspective of the viewer.

A. Inexact Arithmetic for Removing Peppers

As stated before, simplifying the hardware of the TBC simplifies the median filter hardware. We consciously introduce errors to some cells of TBC's K-maps in such a way that it does not affect the removal of peppers in median filter. Fig. 3 gives the K-maps of the precise TBC.

Peppers are the black dots with zero intensities in the image. So, we focus on K-map cells associated with two-bit zeros. We call them critical cells involved with peppers, CCPs, that should be free of errors. They are shown as colored cells in Fig. 4.a.

Either the error should not occur in the critical cells, or, if occurring, should be covered. As H plays a key role in deciding the circuit's output, the error is only allowed in the critical cells of L, where their corresponding cell in H contains zero. Note that the error must not occur in any of the critical cells of H. In this case, the errors in L do not affect the TBC.

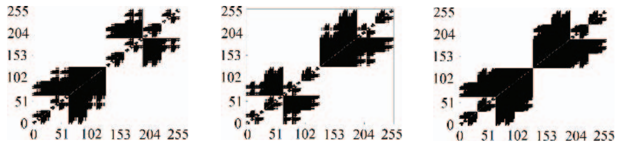
Considering the experiences of [4], we choose the imprecise L as shown in Fig. 4, and use it for all of the approximations. According to Fig. 4. a, just a CCP matched to the cell labeled as 0001 of L is problematic.

We explain this case by an example. Consider two inputs as X and Y. Since our focus is on peppers, one of them is zero. Suppose that X= 00 00 00 00 and Y = 01 $\times\times$ $\times\times$ $\times\times$, in the binary format. Where, $\times$ symbol denotes to "don't care" that can be zero or one. Due to the explanations, for two MSBs of inputs, we have L=H=0. Thus, the comparator detects them as equal. Therefore, the comparison shifted to the next lower significant couples. According to Fig. 4. b, if these couples of Y contain everything except 01, the comparator correctly detects X smaller than Y. However, if Y = 01 01 01 01, then the comparator sets Y equal to X. Therefore; we have H=0, Max=Y, and Min=X. So, the comparator acts accurately and the filter implemented using this imprecise TBC removes the peppers without error. Since this filter acts accurately for pepper, we call it IMFP.

B. Inexact Arithmetic for Removing Salts

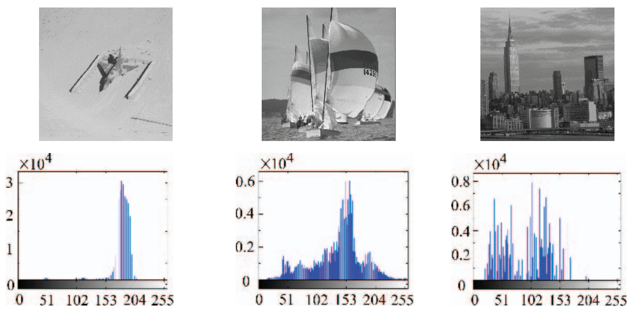
Salts are the white points with 255 intensities in the image. Since 255 is equal to 11111111 in binary, we have to focus on K-map cells associated with 11 (i.e., x1x0=11 or y1y0=11). We call them critical cells involved with salts, CCSs, that should be free of errors. They are shown as colored cells in Fig. 4.b.

For the purpose of this section, '11' must properly compare with other two-bit numbers. Imprecise L and H have to satisfy the conditions explained in the previous section. We suggest H as shown in Fig. 4.b. According to that figure, there is no error in CCSs of H. In addition, the cell labeled by 1011 of H, corresponds to problematic CCs of L, contains zero. Therefore, the inexact median filter removes salts such as the precise one. Because of the exact function of filter on salts, we call it IMFS.



(a) (b) (c)

Fig. 5. EDPs of a) EBC-P8, b) EBC-S8, c) EBC-SP8.



(a) (b) (c)

Fig. 6. Images and histograms of a) airplane, b) boats, and c) buildings.

C. Inexact Arithmetic for Removing Salts and Peppers

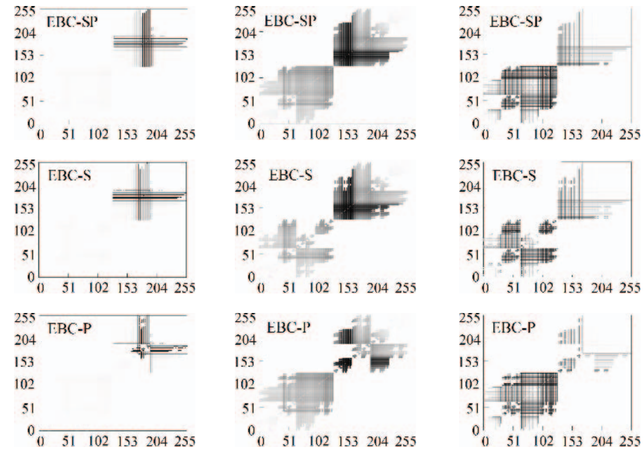
To design an inexact median filter that removes salts and peppers correctly, we need to combine the methods described in the previous two subsections. We suggest L and H as shown in Fig. 4.c. As explained in Section II-A, there is not any error in CCs of H. Also, two cells of H labeled as 0001 and 1011, corresponded to the problematic CCs of L, contain zero. Therefore, the filter works wrong in neither salts nor peppers. However, because of the inexact calculations, some parts of the image may contain errors.

Note that the proposed imprecise arithmetic can be applied to one TBC or more. Clearly, although increasing the number of imprecise blocks reduces the circuit area and power consumption, it diminishes the image quality.

III. Error Evaluation Method

As explained in [5], both of the approximation method and image features affect the quality of the IMF. To select the proper imprecision method, one should know about error characteristics as well as physical properties of the design. Error dispersion plot, EDP, is a good qualitative representation to show all possible error occurrences of imprecise comparator within its input range [5]. EDPs of our proposed eight-bit comparators (EBC) are represented in Fig. 5. EBC-P8 and EBC-S8 denote an EBC composed of eight imprecise TBCs shown in Fig. 4. a, and Fig. 4. b respectively. EBC-SP8 uses eight imprecise TBCs as shown in Fig. 4. c.

As mentioned earlier in the Introduction section, the quality of an inexact filter depends on the image that cannot be inferred from EDP. To better understand the IMF error characteristics, we introduce histogram based EDP (HEDP) as an extended version of EDP. In HEDP, the brightness intensity of each point indicates its error occurrence probability, i.e., the darker the spot, the greater the chance of error. Obviously, the HEDP of each imprecise EBC depends on its logic function, as well as the image histogram. Fig. 6 shows three sample images and their histograms. Fig. 7 represents the HEDPs of imprecise EBCs for three sample images. These maps get



(Airplane) (Boats) (Buildings)

Fig. 7. HEDPs of full-imprecise EBC-SP, EBC-S, and EBC-P, for three sample images.

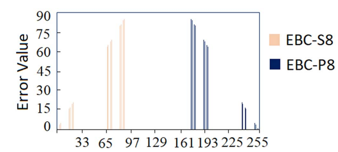


Fig. 8. The range of numbers susceptible to error during salt and pepper removal using EBC-P8 and EBC-S8.

some information about the image degradation using inexact filters, before designing them. According to Fig. 7, the quality of IMFs for airplane should be better than the others. Of course, the IMF ability in noise suppression is as important as the image degradation. It is illustrated in Fig. 8, giving the regions with high risk of error and also error-free regions for ECB-P8 and ECB-S8 in comparing numbers with 255 (salt) and zero (pepper). The IMFP8 may not remove pepper noise from any part of the image with intensity values less than 86. The same is true for IMFS8 in removing salt spots and the image intensity values more than 169. In the other words, IMFP8 correctly eliminates salts and peppers in dark images with a brightness greater than 86. Also, IMFS8 is suitable for removing salt and pepper from bright image with intensity values more than 169. It is worth noting that the IMF, regardless of the image brightness, removes the salt-and-pepper noise correctly.

Using HEDP information and errors of the imprecise EBC in comparing 255 and zero with image intensity values, we calculate the mean error dispersion (ε_d) of imprecise EBSs to pre-characterize the IMF.

$$\varepsilon_d = \frac{1}{255}\left(\frac{\sum_{i=1}^{N_{sp}} E_{sp}(i)}{N_{sp}} + \frac{\sum_{i=1}^{N} E(i)}{N}\right) \tag{1}$$

where, N is the number of points in HEDP of an imprecise EBC, and E is the error value. Nsp denotes the number of errors occurred in comparison of zero or 255 with pixel intensities, and Esp is the related error value. Note that the denominators are the maximum possible error value.

Fig. 9 illustrates the ε_d of imprecise EBCs for three sample images. It allows us to compare the ability of inexact filters in noise suppression along with the image degradation, before designing them. According to Fig. 9, for airplane image,

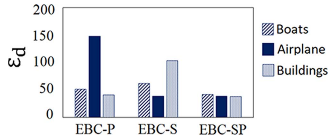


Fig. 9. Mean error dispersion (ε_d) of EBC-P, EBC-S, and EBC-SP for images of boats, airplane, and buildings.

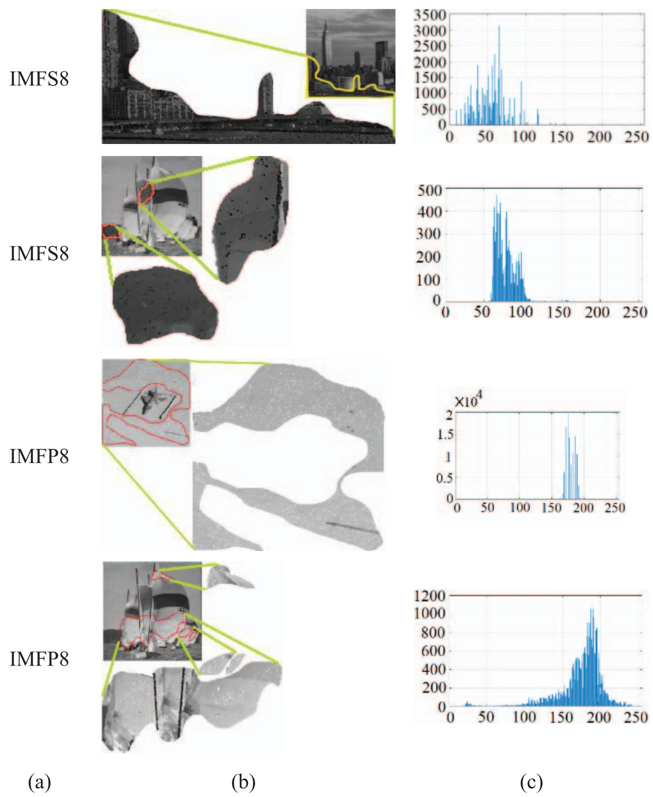


Fig. 10. a) Inexact filter, b) magnified regions of image containing noise after filtering. The input image currupted with 15% noise, c) histograms of noisy regions.

IMF-S is best and IMF-P is the worst. whereas for building, which is darker, IMF-P is the best and IMF-S is the worst. Of course, in any case, the IMF is quite acceptable. The most striking is that Fig. 8 and ε_d are sufficient to determine the proper inexact filter for the desired application.

We tested the filtering capabilities of our inexact filters on three 512×512 gray-level images corrupted by salt-and-pepper noise, where 15 % of pixels are affected. In Fig. 10, we magnify the noisy regions in the outputs of filters, to make them well visible. As we expected from Fig. 9, IMFP perform the worst for airplane image. Because according to Fig. 6, a and Fig. 8, large part of the airplane image is in the high-risk area of IMFP. The same is true for IMFS and image of buildings.

We quantify the performance of output images using the well-known metrics. PSNR objectively measures the intensity of the error signal. MSE is Mean Square Error. MSSIM is a semi-subjective metric, which measures the structural similarity between two images, based on the human visual perception [7].

As shown in Fig. 11, The qualities and also the PSNR values of output images using 2 to 6-bit imprecision in the

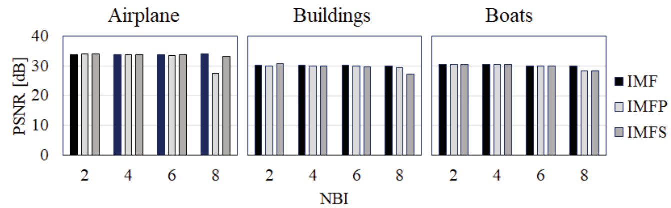


Fig. 11. PSNR of inexact filters for different NBI. Noise density = 10%.

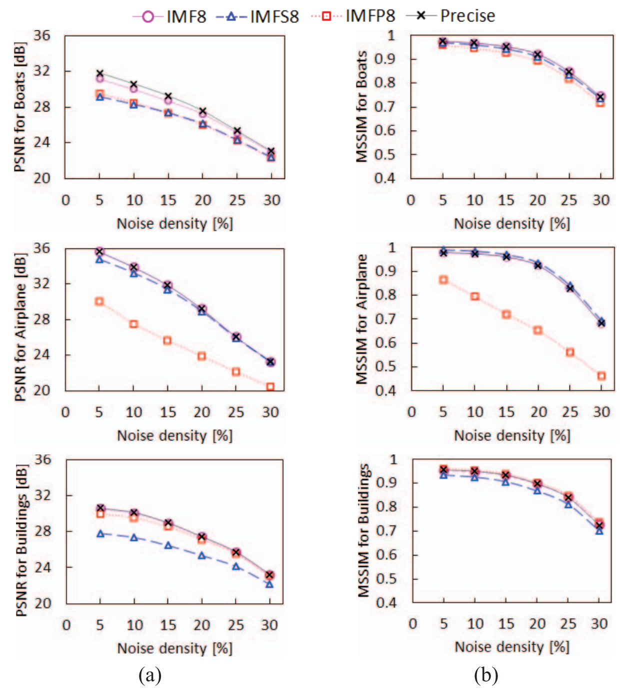


Fig. 12. a) PSNR, b) MSSIM against noise percentage for three sample images.

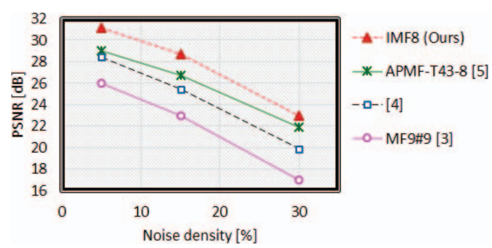


Fig. 13. Comparison on PSNR of different inexact filters for image of boats. The values of the MF9#9 are the mean PSNR on 30 test images [3].

inexact filters are very close to those of the precise one. It means that IMFN, IMFPN, and IMFSN in which, N=2,4,6 can be used safely instead of the precise filter and achieve the desired results. The rightmost digit in the filter name shows the number of the least significant bits calculated imprecisely. Fig. 12 gives the PSNR and MSSIM of images using fully inexact filters. According to Fig. 12, expected from Fig. 9, IMFS8 and IMFP8 has the worst results for images of buildings and airplane, respectively. The rest of the outcomes are very close to each other.

We compare the quality of our best fully inexact filter (IMF8) with three state-of-the-art approximate median filters operating on 3×3 kernel [3], [4], [5]. Fig. 13, shows the PSNR of the filters for image of boats for different noise densities.

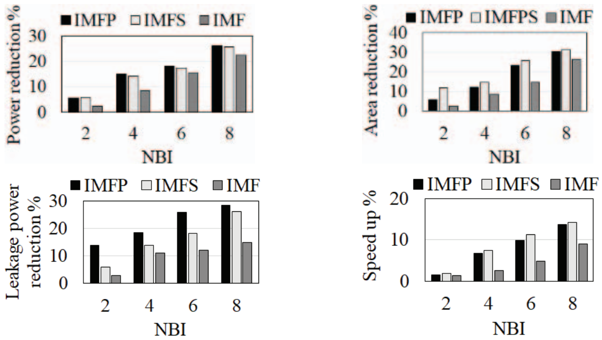


Fig. 14. Synthesis results for different types of APMFs with respect to precise one. NBI shows the numbers of bits computed imprecisely.

TABLE I
COMPARISON ON PHYSICAL PROPERTIES

Median filter	Power (uW)	Delay (ns)	PDP (pJ)
IMFS8	92.56	10.83	1.00
IMFP8	94.95	10.76	1.02
IMF8	103.77	11.43	1.19
APMF-T43 [5]	97.02	10.73	1.04
[8]	129.93	14.70	1.91
[9]	134.08	14.91	2.00
MF9#9 [3]	119.70	13.06	1.56

Note that as reported in [3], the values of the MF9#9 are the mean PSNR on 30 test images. According to Fig. 13, IMF has the best quality among all others.

We also tested on "Lena" image, corrupted with 10% noise. The results of using exact filter (PSNR=33.25[dB], MSSIM=0.98) and IMF8 (PSNR=32.52[dB], MSSIM=0.98) were very close together.

IV. SYNTHESIS RESULTS

In order to evaluate the physical properties of our approximate median filter, we synthesize it with Nangate 45nm Open Cell Library using Synopsys Design Compiler. Power consumption is estimated using Power analysis tool of Synopsys with VCD file produced in post synthesis simulation by applying 100,000 random inputs. Since all window-based spatial filters need line buffers (see Fig. 2), and their size is dependent on the size of the input image, we do not consider them in synthesis results. Fig. 14 illustrates the synthesis results of our design. NBI indicates the number of bits computed imprecisely. As NBI increases, the circuit become much more efficient in terms of area and power consumption. As shown in Fig. 14, using IMF with 8-bit approximation, we can save more than 20% in power and area consumption. Leakage power is a top concern in deep submicron process technologies (65nm and below). According to Fig. 14, using IMF, we can reduce leakage power up to 20% with respect to the precise one. In accordance with Fig. 14, our proposed filters are more efficient in terms of the switching energy than the exact one.

We compare our filters with several state-of-the-art implementations, [3], [5], [8], [9], aiming similar objectives to ours (i.e., high-speed operation and power savings). For this purpose, we synthesize them with the technology used in this brief. Simulation results are shown in TABLE I. The method of [4] is in software-level and is different from those in TABLE I. The Power-Delay-Product (PDP) is the product of average power and worst case delay. Our filters save more power and are faster than [3], [8], [9]. According to TABLE I, the physical properties of IMFS and IMFP is near those of [5]. However, the IMFS and IMFP have better performance than that of IMF, but as previously mentioned, in most cases, the quality of the output image using IMF is much better than those of using IMFS and IMFP. As stated before, there is a trade-off between physical properties and the output quality of inexact filters. We note that all comparisons are made at post-synthesis level for all designs at block level with a similar loading, as ASIC designers perform for their building blocks, being fully characterized for inclusion in a future chip.

V. CONCLUSION

In this brief, we proposed a method to design three imprecise two-bit magnitude comparators that can effectively utilize to tradeoff between precision and physical properties (area, power and speed) of a median filter. These filters are different in the noise that remains in the images. Simulation results show that although the quality of output images is slightly less than that of the precise filter, but the loss is such that the human eye hardly perceives it. The physical properties of our inexact median filters are much better than those of the precise one. Simulations show that the implementations of inexact filters are effectively low-cost and can provide savings up to 26% and 30% in terms of power and area consumption, respectively and get 15% speed up in relation to the conventional accurate one.

ACKNOWLEDGMENT

M. Monajati would like to acknowledge her joint supervisor, the late Prof. S. M. Fakhraie, for his very kind support and valuable advice in the early stages of this brief.

REFERENCES

[1] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 4th ed. Upper Saddle River, NJ, USA: Pearson, 2017.
[2] U. Erkan, L. Gökrem, and S. Enginoğlu, "Different applied median filter in salt and pepper noise," *Comput. Elect. Eng.*, vol. 70, pp. 789–798, Aug. 2018.
[3] L. Sekanina, Z. Vasicek, and V. Mrazek, "Automated search-based functional approximation for digital circuits," in *Approximate Circuits*, vol. 26. Cham, Switzerland: Springer, 2019, pp. 175–203.
[4] Z. Vasicek and V. Mrazek, "Trading between quality and non-functional properties of median filter in embedded systems," *Genet. Program. Evol. Mach.*, vol. 18, no. 1, pp. 45–82, 2017.
[5] M. Monajati, S. M. Fakhraie, and E. Kabir, "Approximate arithmetic for low-power image median filtering," *Circuits Syst. Signal Process.*, vol. 34, no. 10, pp. 3191–3219, 2015.
[6] Z. Vasicek, V. Mrazek, and L. Sekanina, "Evolutionary functional approximation of circuits implemented into FPGAs," in *Proc. IEEE Symp. Series Comput. Intell. (SSCI)*, Athens, Greece, 2016, pp. 1–8.
[7] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
[8] Z. Hosseinkhani *et al.*, "Adaptive real-time removal of impulse noise in medical images," *J. Med. Syst.*, vol. 42, no. 11, pp. 216–225, 2018.
[9] D. Prokin and M. Prokin, "Low hardware complexity pipelined rank filter," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 57, no. 6, pp. 446–450, Jun. 2010.