

A Review of Image Histogram Computation Architectures On FPGA

Bonagiri koteswar rao¹, Dr.Giribabu kande², Dr.P.ChandrasekharReddy³

¹Assistant Professor, ECE, Marri Laxman Reddy Institute of Technology and Management, Hyderabad, Telangana.

¹ Research scholar, Jawaharlal Nehru Technological University Hyderabad, T.S.

¹bonagirikoteswarrao@gmail.com

² Professor & Dean of studies, ECE, VVIT, Guntur, A.P.

²giribabukande@gmail.com

³Professor, ECE, JNTUH, Telangana.

³drpcsreddy@gmail.com

Article History: Received: 11 January 2021; Revised: 12 February 2021; Accepted: 27 March 2021; Published online: 16 April 2021

Abstract. In recent years, most of the research work is done statistical image description as histogram estimation. In this paper we discussed the histogram of a grayscale image. Because of effective hardware utilization of histogram structures, it is well suited to generate histograms for various applications like military and medical applications which is not possible in MATLAB software. We discussed about various histogram structures which are implemented in FPGA platform. These histogram estimation values generated by the modelsim are similar to the histogram estimation values of MATLAB output with 100% accuracy. We discussed the Experimental results of various existed architectures for different size of images.

Keywords: Probability Density Function (PDF), Histogram Estimation, Grayscale Image, Field Programmable Gate Arrays(FPGA).

Keywords: Feature selection, LASSO, Boruta, Recursive Feature Elimination, Regularised Random Forest

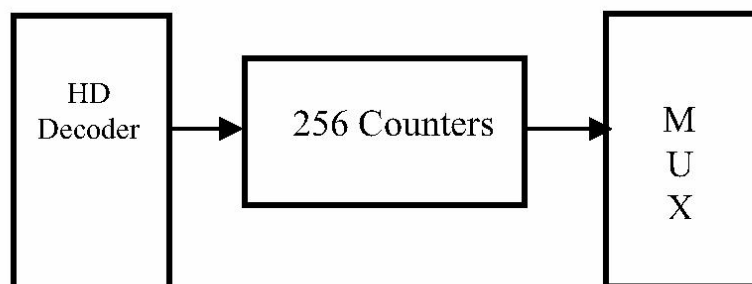
INTRODUCTION

Based on a variable data, estimation of a probability density function (PDF) is very important issue arising in various domains like image processing applications, machine learning, pattern recognition, telecommunications etc. Estimation of a random variable can be done using semi parametric, parametric and non parametric techniques. Histogram is the most fundamental non parametric estimator and is the simplest PDF estimator, has numerous applications in signal and image processing. These estimators are good in image segmentation and can enhance the contrast quality and brightness of grayscale image. Several histogram estimators are proposed to estimate the PDF and to monitor the behaviour of optical channels in wideband applications. In general histogram of an image can be plotted using MATLAB simulation tool but it is not feasible to generate histogram architecture. In this paper development of various histogram architecture for different sizes of grayscale image has been addressed.

LITERATURE SURVEY

ALSUWAILEM A.M., ALSHEBEILI S.A.: 'A new approach for real-time histogram equalization using FPGA'. Proc. 2005 Int.Symp. Intelligent Signal Processing and Communication Systems. ISPACS, 2005, pp. 397-400

This paper presents a new design for real-time histogram computation based on field programmable gate arrays (FPGAs). The architecture is implemented using nonconventional schemes to evaluate the histogram statistics and equalization in parallel. Counters are used in concurrence with a decoder designed for this purpose. The hardware is fast, simple, and plasticity with appropriate development cost. The proposed design is implemented using Stratix II family chip type EP2S 15F484C3. [1].



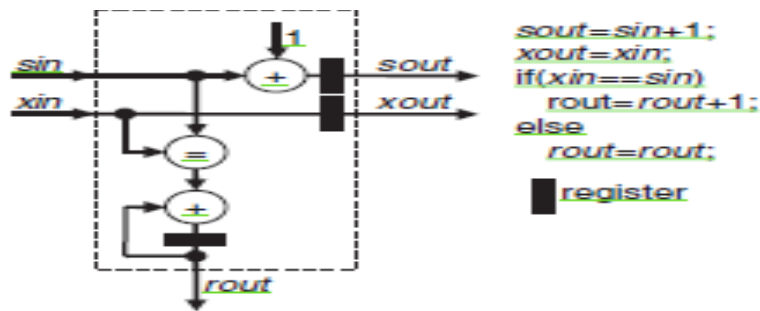
Shahbahrami, A., Hur, J.Y., Juulink, B., and Wong, S.: 'FPGA implementation of parallel histogram computation'. 2nd HiPEAC Workshop on Reconfigurable Computing, Goteborg, Sweden, 2008, pp. 63–72.

Parallelization of the histogram evaluation is a challenging task. This is because if there are various occurrences of a pixel value, there are various writes to the same memory location. Such a situation is called as a memory collision. In image and video processing collisions are common because there are so many occurrences of a pixel value.

We introduce a hardware technique for Parallel Histogram Calculation (PHC). Our proposed method leverages a dual-ported on-chip memory in order to avoid memory collisions. Considering serially processing hardware and software implementation as a reference, a comparative study is conducted. The presented hardware module design is implemented in an FPGA. The results show that the PHC technique performs nearly 2_ better with moderate area overheads, compared to the sequential hardware [2].

JoseO.cadenas,R.Simonsherratt,PabloHuerta,WenChungKao,2011.Parallel Pipelined Array Architectures for Realtime Histogram Computation in Consumer Devices. IEEETransactionsonConsumerElectronics,57(4),pp.1460-1464.

Huerta Proposed a unique cell histogram design which will execute k data items in parallel to calculate $2q$ histogram bins per time step. An array of $m/2q$ cells executes an m -bin histogram with a speed-up factor of k ; $k \geq 2$ makes it faster than current dual-ported memory designs. Further more, simple methods for dispute-free storing of the histogram bins into an external memory array are discussed



A unique pipelined array of cells for the evaluation of a histogram with a speed-up factor of k is discussed. This cell architecture is able to process k data items per time step while evaluating $2q$ histogram bins; k and q are design parameters. Significantly, no memory arrays are used for parallel histogram evaluation, thus no write address disputes to such memory arrays can occur, yet the histogram bins can still be eventually stored in external memory arrays. It is the careful choice of k that offers a good speed-up factor than the factor of two obtained from common histogram results based on dual-port memories.[3]

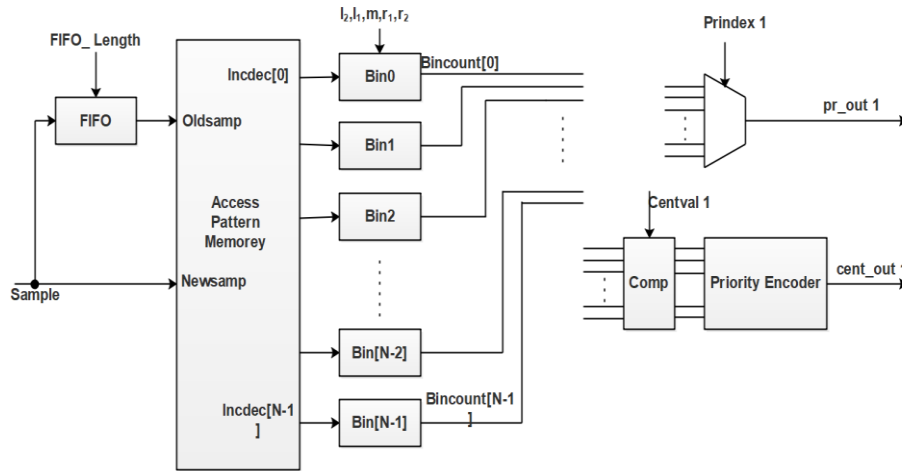
Methodology

Parallel computation of Histogram using array of cells. The result of histogram can be stored in memory block in a pipelined manner

Research gap

More processing cells are required to compute the histogram. More hardware is required because of more processing elements are required. Collisions may occur. Accuracy is low.

Fahmy,S.A.andMohan,A.R.,2012.Architectureforrealtimeonparametricprobabilitydensityfunctionestimation.IEEETransactionsonVeryLargeScaleIntegration(VLSI)Systems,21(5),pp.910-920.



The architecture presented here accepts image data at its input and calculates the cumulative histogram of that data, which is stored internally. By normalizing, based on the window size, which can be modified to any power of 2, the resultant information contained in the circuit gives the cumulative density function. This enables us to extract different PDF statistics in a highly efficient manner. Two design variations are presented in this paper. The first calculates the bare histogram, while the second uses a kernel-based method that results in smoother histograms from less data. Maintaining the full histogram within the design has the advantage of allowing the designer to extract multiple statistics of interest, as required for a specific application. The first half of the circuit evaluates the histogram. This is done by instantiating a bank of counters that keep a tally of the number of occurrences for each input value. This histogram unit is built such that it is updated as each new sample value enters the system. The design is heavily pipelined to allow for highest performance. The second half of the circuit consists of the statistical units that revive information of interest from the cumulative histogram, for use in the target application. We decouple these two parts so that the application designer can extract as many statistics as needed without impacting the performance of the histogram component.

We introduced a novel architecture for real-time computation of PDF estimates based on the histogram and kernel density estimation methods. It makes extensive use of FPGA resources to parallelize and accelerate the algorithm. We showed how a cumulative histogram can be constructed in parallel, how statistical properties can be extracted in real-time, and how priority encoders can be used to extract further statistics. We showed an extended architecture for kernel-based PDF estimation capable of changing kernel widths at run-time, without loss in performance. The architecture can process data streams at 250 million samples per second. We also presented simulation results that help illustrate the trade-off in selecting between raw histogram-based PDF estimation and kernel-based estimation.[5]

Methodology

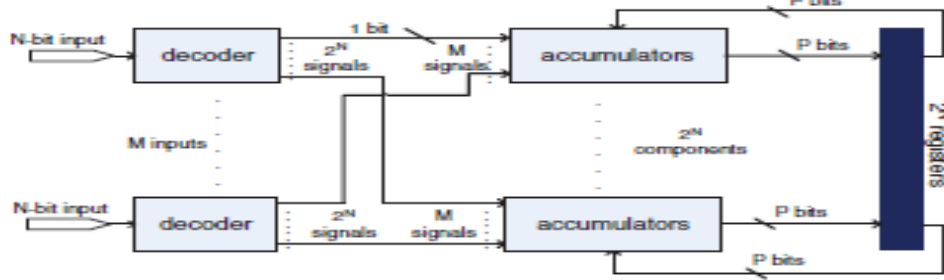
Memory and counter based architecture is used to estimate the histogram. A general architecture for non parametric PDF estimation is presented. It uses both histogram and kernel based methods. It is designed for integration into streaming applications on FPGA.

Research gap

Complexity high because of two ROMs. Bin node processor complexity is high. Memory collisions may occur because of FIFO, if FIFO delay time is not matched with consecutive samples delay time.

Q. Gan, J. M. P. Langlois, and Y. Savaria, "Parallel array histogram architecture for embedded implementations," IET Electron. Lett., vol. 49, no. 2, pp. 99–101, Jan. 2013. doi:10.1049/el.2012.2701.

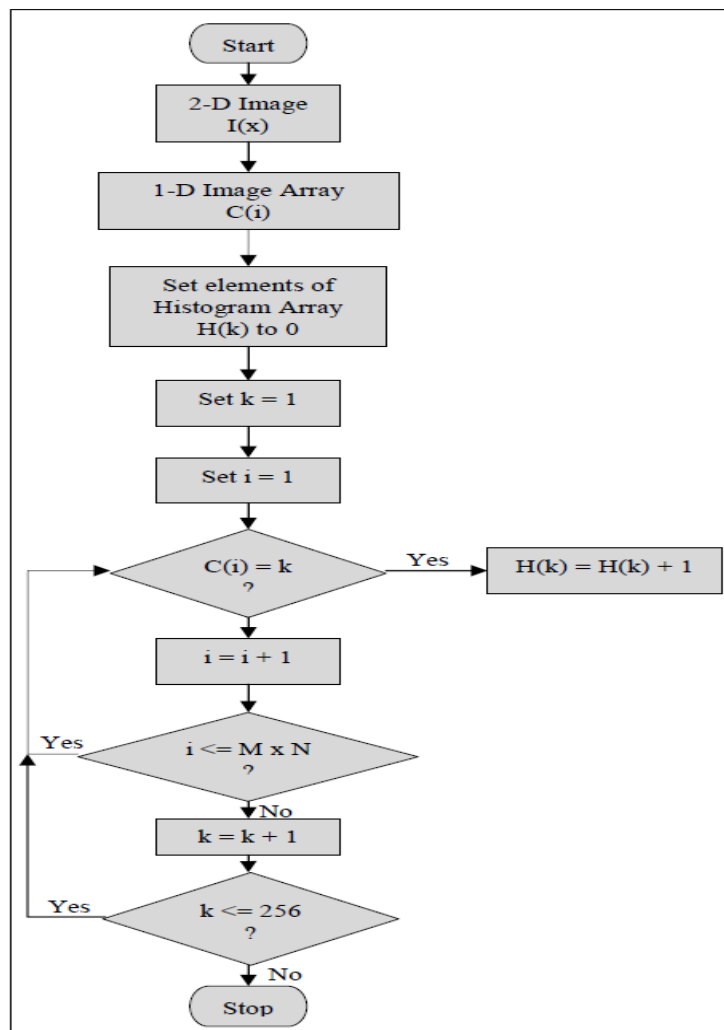
Authors have proposed parallel array histogram design for embedded implementations, in this a register array is used to overcome the limited speed up of memory access. Parallel array histogram architecture (PAHA) is suitable for embedded implementations. The PAHA uses a register array instead of a memory array to store the histogram values. In each step, M inputs can be processed in parallel to renovate the histogram bins without any additional dormancy. Also described is a second version of the PAHA with a flexible number of inputs, potentially neglecting the need for multiple PAHAs in a single application. Implementation results show that the design can achieve a super-linear speed-up of 43.75× for a 16-way PAHA when compared to a software implementation in a general purpose processor.



Conclusion: Above diagram is a parallel array histogram architecture for embedded implementations. A register array is used to overcome the limited speed-up due to memory access. The proposed architecture achieves better performance in terms of throughput when compared to previous work. Implementation results show the speed-up can attain 3.2×, 13.1× and 43.7× for 1-way, 4-way and 16-way PAHAs, respectively, when compared to a traditional software implementation.[7]

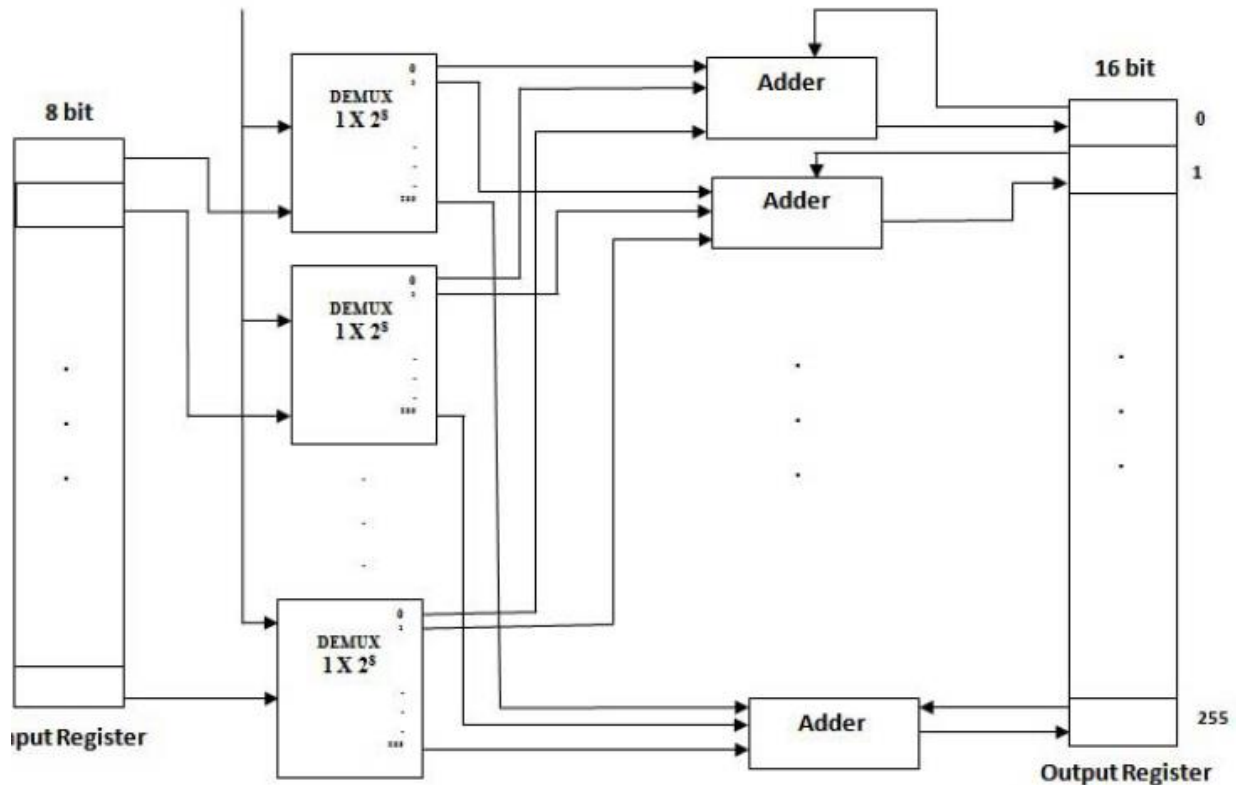
Ghosh S, Hazra S, Maity SP, Rahaman H. A New Algorithm for Grayscale Image Histogram Computation. 12th IEEE India International Conference (INDICON) 2015; 1-6.

A novel histogram development algorithm, which can produce the histogram of any type of grayscale images, is presented in this paper. The algorithm avoids use of any inbuilt functions to develop the histogram. The histogram thus obtained looks exactly similar to that produced by inbuilt functions available in tools like Matlab. Moreover, the algorithm is simple and can be effectively applied to different types of grayscale images. Another important characteristic of the algorithm is that its time complexity is very less. Experimental results have shown that the execution time of generating histograms the proposed algorithm is rather small.[10]



K. S. Gautam, "Parallel Histogram Calculation for FPGA: Histogram Calculation," 2016 IEEE 6th International Conference on Advanced Computing (IACC), Bhimavaram, 2016, pp. 774-777.

In this paper, an architecture is proposed to calculate the histogram of image. Which is faster than the previous serial methods, this architecture achieves the parallelism but needs the enough resources and gives the better performance. If, resources is not an issue then this is one of the best method for histogram calculation in FPGA (Field Programmable GateArray). Some other methods are also proposed to use the same architecture with less number of resources which cost some reduction in speed.



In this paper, we proposed a new architectures 256-ways histogram calculation and its generalized version i.e. n -ways histogram calculation to calculate the histogram in FPGA. Analysis and result verified that the proposed architectures i.e. 256-ways or n -ways are better than the previous serial method in terms of performance by nearly 256 or n times but needs more resources by 256 or n times. For further improvement pipelining can be used.[16]

Hazra,S.,Ghosh,S.,Maity,S.P.andRahaman,H.,2016. A new FPGA and programmable soc based VLSI architecture for histogram generation of grayscale images for image processing applications.ProcediaComputerScience,93,pp.139-145

Methodology

Histogram generation hardware architecture is presented. It develops histogram for all types of grayscale images of 256x256. Histogram generator block consists of adders, decoders, comparators, counters and logic gates.

Research gap

Complexity is high because more number of adders, counters and decoders are used.Image size is restricted to 256x256 only[18]

Yang,Y.,Liu,Y.X.andDong,Q.F.,2017. Sliced integral histogram: an efficient histogram computing algorithm and its FPGA implementation. MultimediaToolsandApplications,76(12),pp.14327-14344.

MethodologyEfficient integral histogram evaluationis done using a sliced integral histogram algorithm. It is used for all available target regions and widely used in computer vision task. Hardware architecture is implemented on FPGA.

Research gap

More number of processing elements are required to estimate local histograms so hardware structure is complex. Less accuracy when compared to software approach. [19]

Mondal, P. and Banerjee, S., 2019. A Reconfigurable Memory Based Fast VLSI Architecture for Computation of the Histogram. IEEE Transactions on Consumer Electronics, 65(2), pp. 128-133.

Methodology

Memory based parallel and pipelined architecture for the computation of joint histogram is proposed. With the help of increasing number of processing blocks in the array and increasing the size of fetching data from memory. This method becomes more pipelined and parallelized. Fast computation of histogram is done with less hardware

Research gap

Memory reconfiguration is not tolerable and considered as a bottle neck (if grayscale image size changes). If image size increases more storage required to store histogram estimation values. [27]

Problem identification

- Parallelization of the histogram computation is a challenging task because of memory collisions.
- For storing the old sample and new sample operation in APM, two ROMs were used in the existed architectures.
- Complex Bin node processors were used in the existed architectures.
- The hardware utilization of the Histogram Equalization architecture is high due to the usage of more logical elements, ROMs and bin counters.
- Most of the histogram computation research works were shown in the SOFTWARE platform only.
- Only few research works were developed in FPGA platform to visualize the histogram count in the simulation waveform.
- ASIC Performance analysis was not provided in the literature.

Objectives

- To develop a Modified Histogram Estimation architecture with low area.
- To evaluate FPGA and ASIC performance parameters for the existed and proposed architectures.
- To propose a parallel and pipelined memory based architecture for histogram estimation.

Conclusion

Enhancement of image is an important feature in the area of image processing. This study has discussed an overview of the background and related work in the area of image computation using FPGA architectures. Recently many modifications on histogram computation were presented in order to find the best optimization architecture. In this paper, we studied the histogram estimation structures for grayscale images. Histogram bin values obtained from both MATLAB and modelsim for various architectures are studied in this paper. FPGA performance of various histogram estimation architectures have been studied. The research gap of this existed structure is hardware utilization is more and frequency i.e., the speed is less, still we can improve the speed of operation and reduce the complexity of the architecture by using parallel pipelined memory based architecture which requires less hardware and speed of operation is very high. In future we can analyze the ASIC performance for both histogram estimation and equalization architectures

REFERENCES

1. ALSUWAILEM A.M., ALSHEBEILI S.A.: 'A new approach for real-time histogram equalization using FPGA'. Proc. 2005 Int. Symp. Intelligent Signal Processing and Communication Systems. ISPACS, 2005, pp. 397–400
2. Shahbahrani, A., Hur, J.Y., Juulink, B., and Wong, S.: 'FPGA implementation of parallel histogram computation'. 2nd HiPEAC Workshop on Reconfigurable Computing, Goteborg, Sweden, 2008, pp. 63–72.
3. Jose O. Cadenas, R. Simonsherratt, Pablo Huerta, Wen Chung Kao, 2011. Parallel Pipelined Array Architectures for Realtime Histogram Computation in Consumer Devices. IEEE Transactions on Consumer Electronics, 57(4), pp. 1460-1464.
4. H. Medeiros, G. Holguin, P. J. Shin, and J. Park, "A parallel histogram-based particle filter for object tracking on SIMD-based smart cameras," *Comput. Vis. Image Understand.*, vol. 114, no. 11, pp. 1264–1272, Nov. 2010. doi:10.1016/j.cviu.2010.03.020.

5. Fahmy,S.A.andMohan,A.R.,2012.Architecture for real time non parametric probability density function estimation. *IEEETransactionsonVeryLargeScaleIntegration(VLSI)Systems*,21(5),pp.910-920.
6. Poostchi M, Palaniappan K, Bunyak F, Becchi M, Seetharaman G (2012) Efficient GPU implementationof the integral histogram. In: International conference on computer vision, pp 266–278
7. Q. Gan, J. M. P. Langlois, and Y. Savaria, “Parallel array histogramarchitecture for embedded implementations,”*IET Electron. Lett.*, vol. 49,no. 2, pp. 99–101, Jan. 2013. doi:10.1049/el.2012.2701.
8. Sanny A, Yang YH, Prasanna VK. Energy-efficient histogram on FPGA. In*ReConFigurable Computing and FPGAs (ReConFig)*, 2014 International Conference on 2014 Dec 8 (pp. 1-6). IEEE
9. Maggiani, L., Salvadori, C., Petracca, M., Pagano, P., & Saletti, R. (2014, June). Reconfigurablearchitecture for computing histograms in real-time tailored to FPGA-based smart camera. In *Industrial Electronics (ISIE)*, 2014 IEEE 23rd International Symposium on (pp. 1042-1046). IEEE.
10. Ghosh S, Hazra S, Maity SP, Rahaman H. A New Algorithm for Grayscale Image Histogram Computation. *12th IEEE India International Conference (INDICON) 2015*; 1-6.
11. Tsai YW, Cheng FC, Ruan SJ (2015) An efficient dynamic window size selection method for 2-D histogram construction in contextual and variational contrast enhancement. *Multimed Tools Appl*:1–17
12. Yadav, A.K., Naskar, R.: A tamper localization approach forreversible watermarking based on histogram bin shifting. In:*IEEE Power, Communication and Information TechnologyConference*, pp. 721–726 (2015). doi:10.1109/PCITC.2015.7438091
13. Yuan, L. Cai-nian, X. Xiao-liang, J. Mei, and Z. Jian-guo, “A two-stage hog feature extraction processor embedded with SVM for pedestriandetection,” in*Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2015,pp. 3452–3455
14. Zhou, Z. Chen, and X. Huang, “A pipeline architecture for trafficsign classification on an FPGA,” in*Proc. IEEE Int. Symp. Circuits Syst.(ISCAS)*, May 2015, pp. 950–953.
15. Y. Hsiao, S.-Y. Lin, and S.-S. Huang, “An FPGA based human detec-tion system with embedded platform,”*Microelectron. Eng.*, vol. 138,pp. 42–46, Apr. 2015.
16. K. S. Gautam, "Parallel Histogram Calculation for FPGA: Histogram Calculation," 2016 IEEE 6th International Conference on Advanced Computing (IACC), Bhimavaram, 2016, pp. 774-777.
17. ang P, Wang Q, Zhang J (2016) Parallel design and implementation of error diffusion algorithm and IPcore for FPGA. *Multimed Tools Appl* 75(8):4723–4733
18. Hazra,S.,Ghosh,S.,Maity,S.P.andRahaman,H.,2016.A new FPGA and programmable soc based VLSI architecture for histogram generation of grayscale images for image processing applications. *Procedia Computer Science*,93,pp.139-145.
19. Yang,Y.,Liu,Y.X.andDong,Q.F.,2017.Slicedintegralhistogram:an efficient histogram computing algorithm and itsFPGA implementation.*Multimedia Tools and Applications*,76(12),pp.14327-14344.
20. Chen, J. Xu, and Z. Yu, “A fast and energy efficient FPGA-based systemfor real-time object tracking,” in*Proc. Asia–Pacific Signal Inf. Process.Assoc. Annu. Summit Conf. (APSIPA ASC)*, Dec. 2017, pp. 965–968
21. E. Ilas, “HOG algorithm simplification and its impact on FPGA imple-mentation: With applications in car detection,” in*Proc. 9th Int. Conf.Electron., Comput. Artif. Intell. (ECAI)*, Jun. 2017, pp. 1–6.
22. J. Rettkowski, A. Boutros, and D. Göhringer, “HW/SW co-design of theHOG algorithm on a xilinx zynq SoC,”*J. Parallel Distrib. Comput.*,vol. 109, pp. 50–62, Nov. 2017
23. B. K., V. Venkatraman, A. R. Kumar, and S. D. S., “Accelerating real-time computer vision applications using HW/SW co-design,” in*Proc. Int.Conf. Comput., Commun. Electron. (Comptelix)*, Jul. 2017, pp. 458–463.
24. B. Meus, T. Kryjak, and M. Gorgon, “Embedded vision system forpedestrian detection based on HOG+SVM and use of motion informationimplemented in zynq heterogeneous device,” in*Proc. Signal Process.,Algorithms, Archit., Arrangements, Appl. (SPA)*, Sep. 2017, pp. 406–411
25. Sledevie, A. Serackis, and D. Plonis, “FPGA-based selected objecttracking using LBP, HOG and motion detection,” in*Proc. IEEE 6thWorkshop Adv. Inf., Electron. Electr. Eng. (AIEEE)*, Nov. 2018, pp. 1–5.
26. M. Qasaimeh, J. Zambreno, and P. H. Jones, “A runtime configurablehardware architecture for computing histogram-based feature descriptors,”in*Proc. 28th Int. Conf. Field Program. Log. Appl. (FPL)*, Aug. 2018,pp. 351–3513.
27. M.-S. Wang and Z.-R. Zhang, “FPGA implementation of HOG basedmulti-scale pedestrian detection,” in*Proc. IEEE Int. Conf. Appl. Syst.Invention (ICASI)*, Apr. 2018, pp. 1099–1102.
28. Mondal,P.andBanerjee,S.,2019.AReconfigurable Memory Based Fast VLSI Architecture for Computation of the Histogram.*IEEETransactionsonConsumerElectronics*,65(2),pp.128-133.