Pakistan Journal of Engineering and Technology, PakJET
Multidisciplinary | Peer Reviewed | Open Access

Volume: 1, Number: 1, Pages: 1- 6, Year: 2018

Implementation and Hardware Optimization for Real Time Target Tracking

Omer Khan[*,1], Umair Tahir[1], Raheel Muzzammel[1] and Nayab Saeed[1]

[1]Electrical Engineering Department, The University of Lahore, 1-KM Defence Road, Lahore, 54000, Pakistan
Corresponding author: Omer Khan (Email: omer.khan@ee.uol.edu.pk)

Abstract— **Real time target tracking algorithms provides challenging environment in the field of computer vision. These challenges arise due to chaotic environment, irregular motion of objects, noise of both camera and video sequence, changes in target object and scene, etc. There are many optimized techniques used for target tracking, however to develop a system for real time target tracking these algorithms are to be implemented on some hardware. In this paper Fast- Normalized Cross-Correlation is carefully chosen for the purpose of real time target tracking. Usually these algorithms require enormous computations in real time which makes it hard to accomplish. To implement such algorithms high performance embedded hardware is required. In this research "TMS320DM642 evaluation module with TVP video decoders" digital signal processor embedded board is used to implement real time tracking algorithm. Digital signal processor hardware has limited recourse, so to achieve real time target tracking some hardware optimization techniques are also implemented.**

Index Terms— Normalized Cross-Correlation (NCC); Real time Tracking (RTT); Region of Interest (ROI); Sum of Squared Distance (SSD); Digital Signal Processor (DSP); Phase Alternation Line (PAL); Computational time; Frame rate; Throughput.

I. INTRODUCTION

Target tracking is used widely in both government and business organizations, such as airports, post offices, convenient stores, banks and public streets. It requires target tracking in real time to provide real time target tracking we need fast and reliable algorithm, with the help of intelligent tracking system this problem can be improved. The aim of this research is to design a target tracking algorithm that can used to detect and classify the events in real time. In most of the systems hardware optimization is also required for real time tracking for automated systems. Dedicated hardware can be designed to perform application's specific tasks but it will be much expensive to use redundant hardware. Optimizing the code will not provide same results as of hardware optimization but hardware optimization is not cost effective and fusible.

To get the optimized target tracking algorithm for TMS320DM642 Evaluation Module with TVP Video Decoders that provides output with the maximum frame rate, which is compatible with chaotic environment and deal more efficiently with the contrast of the video.

II. RELATED WORK

The phenomenon of analyzing video sequences is known as video surveillance [1]. Video surveillance is a demanding and vigorous region in the field of computer vision and has been proved vital in data storing and displaying. Automated surveillance systems are required to provide target tracking and feature extraction [2]. To provide video surveillance for a long time by a human operator is not possible [3]. Statistical method is introduced to overcome the inadequacy of the shortcomings in background subtraction using alpha methods.

In correlation based methods velocity is calculated by estimating the shifts, between two consecutive images of an image sequence. The shift describes the movement of region between images [4], [5]. Some of the existing correlation based methods include method proposed by Sutton, Walters, Peters, Anson and McNeil [6] where optical flow is estimated by iterative minimization of SSD of deformation parameter, obtained by linear deformation of local image patches. A 3x3 spatial operator is used by Kories and Zimmerman [7] to classify central pixel intensity by counting the low intensity values of the neighboring pixels. In [8], [9] Cauchy-Schwartz inequality is used to calculate NCC for medical image registration by template matching. MATLAB is used for the implementation purpose of NCC template matching.

In [10], [11] for efficient search a winner update scheme is introduced by combining multilevel adaptive partition for NCC. The literature for fast-

NCC can be studied from [6], [12] which provide significant basic knowledge of our research.
Hardware optimization is also a part of this research. Different techniques for memory optimizations are implemented [13] to enhance results. These techniques also improve; CPU utilization, reduce power consumption, etc [14]. A novel cluster-based register optimization technique called Power Islands Synthesis (PIS) is introduced by Deniz et al. [15]. Hardware optimization techniques are the main area of research these days to provide best performance for general purpose embedded hardware. Keoncheol Shin et al. devised a technique that can be used effectively to the synthesis of arithmetic circuits aiming power minimization while satisfying the timing constraint of the circuit [16], [17]. By implementing both tracking and hardware optimizations better system can be provided for the purpose of Real time target tracking.
In this research cross correlation target tracking algorithm technique is implemented. After implementation results are recorded in terms of

- Processing time
- Frame rate
- Percentage error
- Throughput
- Power consumption

Target tracking using cross correlation is then modified to Normalized Cross-Correlation (NCC) and Fast-NCC and then results are compared. Additional hardware optimization algorithm are applied which are discussed below, kernel technique is also applied to further improve the results. All techniques provide some improvement and decaying in results but kernel technique showed promising improvements in results. Results of all the hardware techniques are discuss in later sections.

III. PROPOSED SYSTEM

The proposed system is developed by merging different techniques to provide best suited real time tracking algorithm for embedded system (TMS320DM642 evaluation module with TVP video decoders) with hardware optimization as shown in figure 1.

IV. IMPLEMENTATION OF TARGET TRACKING

A standard and most commonly used approach for feature matching is correlation between two frames, some sophisticated techniques are also used but they are much expensive regarding computations and power consumption.

A. Template Matching By Cross-Correlation

Euclidean distance is considered as an origin for template matching using cress-correlation.

$$\boldsymbol{d^2}_{\boldsymbol{f,t}}(\boldsymbol{u},\boldsymbol{v}) = \sum_{\boldsymbol{x,y}}[\boldsymbol{f}(\boldsymbol{x},\boldsymbol{y}) - \boldsymbol{t}(\boldsymbol{x}-\boldsymbol{u},\boldsymbol{y}-\boldsymbol{v})]^{\boldsymbol{2}}) \qquad (1)$$

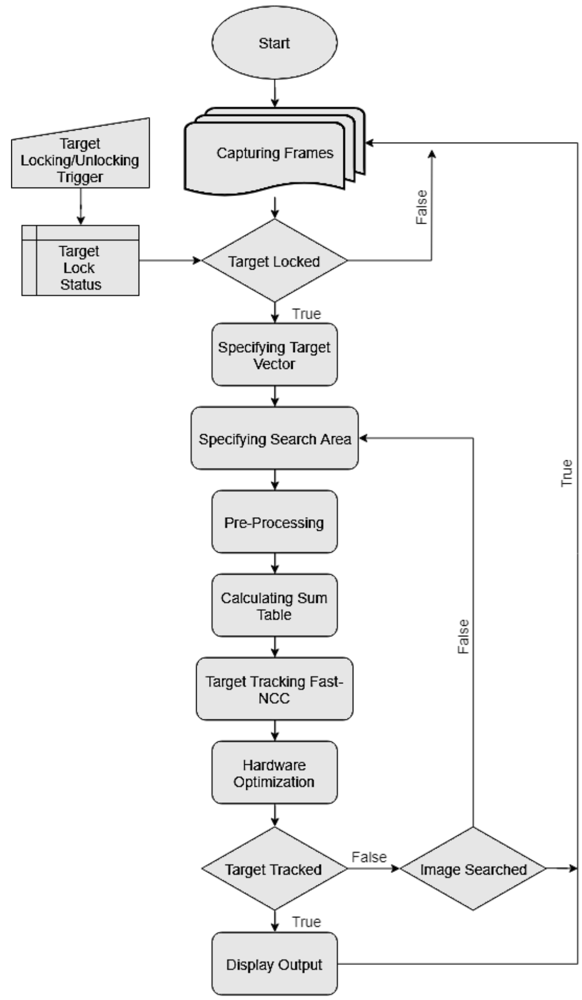

Figure 1 Flow Chat

In (1) f is the search area extracted from the video input. If sum is provided over x, y under the window containing the target vector t positioned at u, v. Expansion of equation:

$$\boldsymbol{d^2}_{\boldsymbol{f,t}}(\boldsymbol{u},\boldsymbol{v}) = \sum_{\boldsymbol{x,y}} \boldsymbol{f^2}(\boldsymbol{x},\boldsymbol{y}) - \boldsymbol{2f}(\boldsymbol{x},\boldsymbol{y})\boldsymbol{t}(\boldsymbol{x}-\boldsymbol{u}, \boldsymbol{y}-\boldsymbol{v}) + \boldsymbol{t^2}(\boldsymbol{x}-\boldsymbol{u},\boldsymbol{y}-\boldsymbol{v}) \qquad (2)$$

In above equation term $\sum t^2(x-u, y-v)$ is a constant value, the term $\sum f^2(x, y)$ is approximately constant. By considering above two terms constant we can say that above equation can be written as follows:

$$\boldsymbol{d^2}_{\boldsymbol{f,t}}(\boldsymbol{u},\boldsymbol{v}) = \sum_{\boldsymbol{x,y}} \boldsymbol{f}(\boldsymbol{x},\boldsymbol{y})\boldsymbol{t}(\boldsymbol{x}-\boldsymbol{u},\boldsymbol{y}-\boldsymbol{v}) \qquad (3)$$

Above equation represents the cross-correlation equation, it is used to measures the similarity between the search area and the target vector.

i. Correlation coefficient

Cross-correlation faces problems due to changes in amplitude of values, which is caused by changes, occur in brightness with the passage of time. These difficulties in correlation are overcome by normalizing the search area and target vectors to unit length with the help of correlation coefficient, yielding a cosine-like correlation coefficient.

B. Normalized Cross-Correlation

The problem mentioned in above section is handled by using normalized cross-correlation. To determine the location of target vector in search area of two-dimensional frame is by computing the normalized cross-correlation coefficient γ for every point of (u, v) for f and t, which has been shifted by u pixels along x direction and v pixels along y direction. Equation below represents the basic equation of normalized cross-correlation coefficient.

$$\gamma(u,v) = \frac{\sum_{x,y}\left[f(x,y) - \bar{f}_{u,v}\right]\left[t(x-u,y-v) - \bar{t}\right]}{\sqrt{\left\{\sum_{x,y}\left[f(x,y) - \bar{f}_{u,v}\right]^2 \sum_{x,y}\left[t(x-u,y-v) - \bar{t}\right]^2\right\}}} \quad (4)$$

Where $\bar{t}$ represents the mean value target vector and $\bar{f}_{u,v}$ signifies the mean value of search area $f(x, y)$ extracted from video input. The equations for determining mean of search area and target vector are below

$$\bar{f}_{u,v} = \frac{1}{N_x N_y} \sum_{x=u}^{u+N_x-1} \sum_{y=v}^{v+N_y-1} f(x,y) \quad (5)$$

$$\bar{t}_{u,v} = \frac{1}{M_x M_y} \sum_{x=u}^{u+M_x-1} \sum_{y=v}^{v+M_y-1} t(x-u,y-v) \quad (6)$$

Where N_x *and* N_y determine the dimensions of search area frame, M_x *and* M_y are the limits defined for the target vector.

C. FAST NORMALIZED CROSS-CORRELATION

Image function f and search area energy f^2 are processed to compute their sum tables to provide fast calculation of the normalized cross-correlation, and these are the pre-computed integrals used for Fast-NCC.

i. Calculation for Denominator

To reduce the computations of denominator of normalized cross-correlation coefficient, the procedure used is by maintaining two sun tables $s(u, v)$ and $s^2(u, v)$ for the search area $f(x, y)$ and its energy $f^2(x, y)$. The sum table can be expressed by

$$s(u,v) = f(u,v) + s(u-1,v) + s(u,v-1) - s(u-1,v-1) \quad (7)$$

Similar recursive function for search area energy is given by

$$s^2(u,v) = f^2(u,v) + s^2(u-1,v) + s^2(u,v-1) - s^2(u-1,v-1) \quad (8)$$

Using these sum tables mean for search area from equation 5 can be very efficiently calculated independent of the size of target vector.

ii. Calculation of Nominator

Algorithm presented in last subsection efficiently reduces the denominator equation of normalized cross-correlation, but calculation for numerator for NCC-coefficient is still comparatively high. If it is implemented in frequency domain using fast Fourier transform algorithm. Therefore, to avoid many computations nominator is to be simplified, numerator can be represented as:

$$N(u,v) = \sum_x \sum_y f(x,y) t'(x-u,y-v) - \bar{f}_{u,v} \sum_x \sum_y t'(x-u,y-v) \quad (9)$$

Where, in above equation $t'(x-u, y-v)$ is known as zero mean template function defined by

$$t'(x-u,y-v) = t(x-u,y-v) - \bar{t}(x,y) \quad (10)$$

Now, in above equation $\bar{t}(x, y)$ has zero mean value and thus also zero sum, the term $\bar{f}_{u,v} \sum_x \sum_y t'(x-u, y-v)$ will also approaches to zero as well. Now numerator of equation 4.6 can be expressed as:

$$N(u,v) = \sum_x \sum_y f(x,y) t'(x-u,y-v) \quad (11)$$

Where (2) provides the simplified term for nominator of normalized cross-correlation coefficient.

V. IMPLEMENTATION OF HARDWARE OPTIMIZATION TECHNIQUES

Various techniques are followed by researchers to optimize their algorithm to improve hardware utilization. Some techniques are so simple and easy to implement; however, they provide enhancements in CPU utilization, power consumption, response time, and throughput of the system. There are few constraints in embedded hardware that can be referred as:

- Time: Time allowed to complete a task requires optimization of machine cycles to complete a task, in real time systems time is a major constraint.
- Computational Platform: Embedded system usually power consumption, cost, size, memory are major factors to develop a system.

A. CPU Optimization Techniques

Central processing unit (CPU) is referred as brain of any system, as all processing is done by CPU such as addition, multiplication, decision making and input/output controls. To maximize the through put one must increase the CPU utilization, but it will cause more power consumption. There are some CPU optimization techniques that increase the throughput without increasing power consumption.

- Pass object by reference instead by value
- Avoid using library functions
- Use of inline function

B. Memory Optimization Technique

In the field of embedded systems memory optimization has its own importance. Embedded systems have limited memory which provide challenging environment to provide best suited solutions for such scenarios. External memory can be interfaced with embedded systems with input/output ports but it introduces additional overhead to the systems.

- Loop unrolling
- Reuse of memory spaces
- Direct Memory Access

C. Minimization of Code

Compact code is a key to develop optimized and efficient embedded system. Line of code can be minimized by many techniques without disturbing the actual functionality of the algorithm. A very effective and efficient technique which improves performance of our system exponentially is Kernel technique.

In target tracking by cross-correlation we have to compare target vector in search area to find the desired location. Time to process a frame depends upon two things size of target vector and search area and displacement of object from its original location. It is impossible to control displacement of the object as it moves with its own free will but we can reduce the size of vectors in such a manner that efficiency of the system is not reduces. By reducing the size of vectors we actually reduced number of additions and multiplication in to process of tracking algorithm. After locating kernel, whole target vector is matched to verify whether the correct object is tracked or not.

In this research we only reduce the size of target vector to one fourth of the actual template size. By this we reduce the response time exponentially which allows the system to process more frame with respect to time, so frame rate is automatically increased.

VI. Real Time Target Tracking And Hardware Optimization Results

In order to examine our implemented real time target tracking algorithm some experiments are to be made to obtain some results. We take four set of video sequences of time interval two minutes; normal video sequence, chaotic video sequence, low contrast video sequence and dark video sequence to compute the results.

A. Experimental Parameters

Parameters selected to investigate the efficiency of this implemented algorithm are; average time to process a frame, frame rate, accuracy by percentage of throughput and error and power consumption.

i. Process Time

A hardware pin was set high when processing starts and set zero when process is done; this pin is then monitored by using logic analyzer to calculate the average time duration to process a frame.

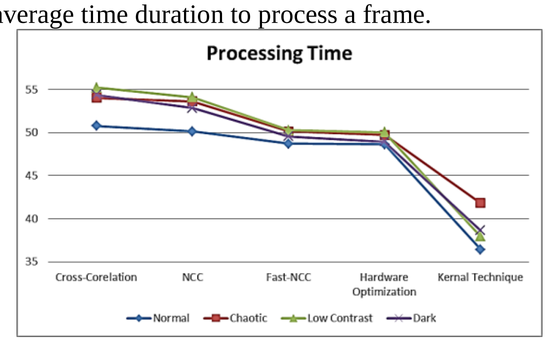

Figure 2 Comparison of Process Time in ($\boldsymbol{ms}$)

It clear from above graph that there is a small improvement in processing time between cross-correlation and NCC, but fast-NCC provides way enhanced results from both of the techniques. Hardware optimization techniques showed exponential progress in processing time especially by Kernel method.

ii. Frame Rate

IEEE standard for efficient PAL standard video sequence is 25 frames per second (FPS). As mentioned in previous sub-section we control an output pin during process, we can also calculate no. of frames processed in one second.

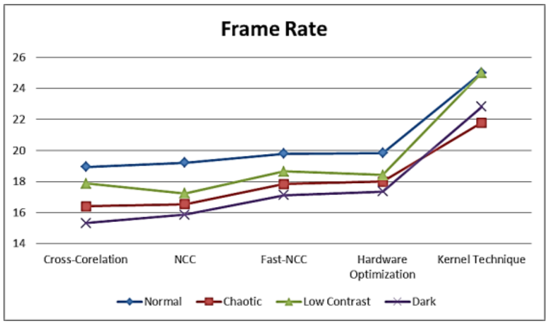

Figure 3 Comparison of Frame Rate in FPS

Comparison of frame rate is showen in figure 3, cross-correlation, NCC, fast-NCC, and hardware optimization showed liner improvement. Kernal basesed technique provide maximum frame rate, nearly equal to input frame rate.

iii. *Percentage Throughput*

Throughput can be computed for a target tracking algorithm in terms of number of frames with target object located. It can be computed by this formula.

$$Throughput\ \% = \frac{frame\ recived - target\ lost}{frame\ recived} \times 100$$

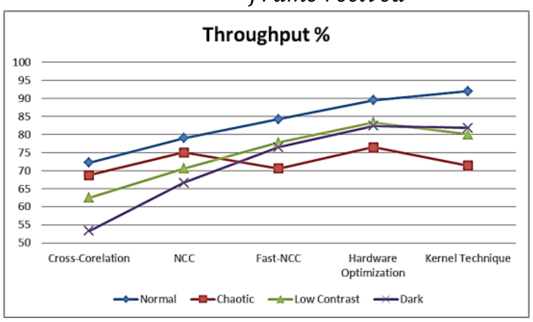

Figure 4 Comparison of Percentage Throughput

Throughputs of all the implemented techniques are liner except for the video sequence of chaotic environment. Fast-NCC and Kernel method is not best suited for chaotic scenes.

iv. *Percentage Error*

Error can be computed in terms of number of frames with target object fail. Frames with number of target object fail can further be categorized into two categories object lost and false object located. But in this research we only consider whether target is found or lost. It can be calculated by using this formula.

$$Error\ \% = \frac{frame\ recived - target\ located}{frame\ recived} \times 100$$

Below figure shows percentage error of our research. Error ratio for all the techniques are improved but results for chaotic environment does not change or improved.

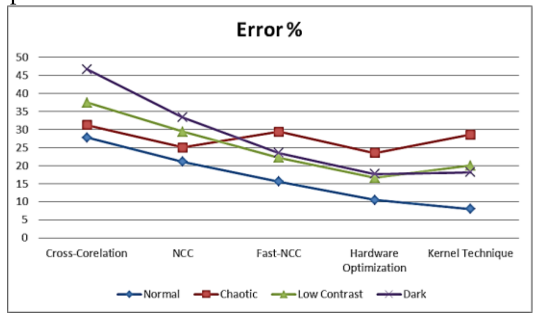

Figure 5 Comparison of Percentage Error

v. *Power Consumption*

Power consumption is one of the important parameter in the field of embedded system. In our research we are implemented our algorithm on TMS320DM642 evaluation module with TVP video decoders as it is patched hardware so it is impossible to calculate power consumption of individual modules so we calculate power consumption of the whole system. Maximum power that can be utilized by TMS320DM642 evaluation module with TVP video decoders is 25 watt.

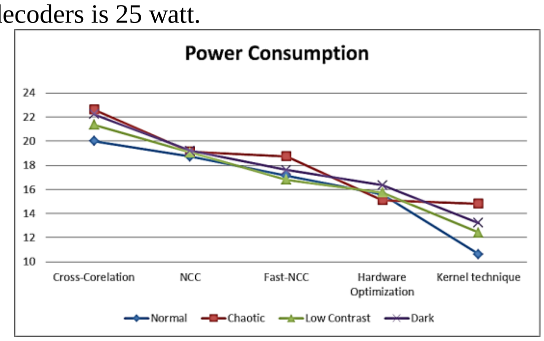

Figure 6 Comparison of Power Consumption in Watts

Power consumption is reduced in a regular manner as showed in figure 6. There are no irregular behaviors observed in this research.

VII. CONCLUSION AND FUTURE WORK

This paper presents simple techniques to improve the results of Real Time Target Tracking by two means; optimizing target tracking algorithm, and some hardware optimization techniques. The results showed that Fast-NCC provide significant improvement in all the experimental parameters. Our proposed algorithm provides enhanced results in different complex environment; chaotic, low contrast, and dark scenes. Congregated results from fast-NCC provide satisfactory results but to further enhance the throughput of the system we use hardware optimization techniques, but kernel-based technique provides exponential improvement in all the results parameters. Results showed significant improvement in performance, and object tracking.

In future further tracking optimization can be implemented, hardware optimizations algorithm that can utilize provided resources for maximum throughput can be implemented as well. Our algorithm does not deal with situations like; target object is scaled, tilted, and object rotates from its original location.

REFERENCES

[1] R. Venkatesan, and A. B. Ganesh, "Supervised and Unsupervised Learning Approaches for Tracking Moving Vehicles", Proceedings of the 2014 International Conference on Interdisciplinary Advances in Applied Computing, article 6, Kerala, Oct., 2014.

[2] H. Sui, Z. Song, D. Gao and L. Hua, "Automatic Image Registration Based on Shape Features and Multi-scale Image Segmentation," 2017 2nd International Conference on Multimedia and Image Processing (ICMIP), Wuhan, 2017, pp. 118-122.

[3] I. S. Kim, H.S. Choi, K. M. Yi, J. Y. Choi and S.G. Kong. "Intelligent visual surveillance — A survey", International Journal of Control Automation and Systems, vol. 8, no. 5, pp. 926-939, 2010.

[4] J. L. Barron, D. J. Fleet and S. S. Beauchemin, “Performance of optical flow techniques”, International Journal of Computer Vision, vol. 12, no. 1, pp. 43-77, 1994.

[5] L. Wang, X. Wang, M. Tornatore, K. J. Kim, S. M. Kim, D. U. Kim, K. E. Han and B. Mukherjee, "Scheduling with machine-learning-based flow detection for packet-switched optical data center networks," IEEE/OSA Journal of Optical Communications and Networking, vol. 10, no. 4, pp. 365-375, April, 2018.

[6] A. Bruhn and J. Weickert, "Towards Ultimate Motion Estimation: Combining Highest Accuracy with Real-Time Performance," Tenth IEEE International Conference on Computer Vision (ICCV'05), vol. 1, Beijing, pp. 749-755, 2005.

[7] P. Kruizinga, F. Mastik, J. G. Bosch, N. de Jong, G. van Soest and A. F. W. van der Steen, "High frame rate ultrasound displacement vector imaging", 2014 IEEE International Ultrasonics Symposium, Chicago, IL, 2014, pp. 707-709.

[8] J. N. Sarvaiya, S. Patnaik and S. Bombaywala, "Image Registration by Template Matching Using Normalized Cross-Correlation," 2009 International Conference on Advances in Computing, Control, and Telecommunication Technologies, Trivandrum, Kerala, 2009, pp. 819-822.

[9] F. E. T. Munsayac, L. M. B. Alonzo, D. E. G. Lindo, R. G. Baldovino and N. T. Bugtai, "Implementation of a normalized cross-correlation coefficient-based template matching algorithm in number system conversion", IEEE 9th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM), Manila, 2017, pp. 1-4.

[10] S. Wei and S. Lai, “Fast template matching algorithm based on normalized cross correlation with adaptive multilevel winner update‖”, IEEE Transactions on Image Processing, Vol. 17, No. 11, Nov. 2008.

[11] Z. Yang, "Fast Template Matching Based on Normalized Cross Correlation with Centroid Bounding," 2010 International Conference on Measuring Technology and Mechatronics Automation, Changsha City, 2010, pp. 224-227.

[12] C. Zach, T. Pock, and H. Bischof, “A duality-based approach for real time TV-L1 optical flow”, 29th DAGM conference on Pattern recognition, Heidelberg, Germany, 2007, pp. 214-223.

[13] Panda, Dutt and Nicolau, "Exploiting off-chip memory access modes in high-level synthesis," 1997 Proceedings of IEEE International Conference on Computer Aided Design (ICCAD), San Jose, CA, USA, 1997, pp. 333-340.

[14] Xiaoyong Tang, Tianyi Jiang, A. Jones and P. Banerjee, “Compiler optimizations in the PACT HDL behavioral synthesis tool for ASICs and FPGAs”, IEEE International Systems-on-Chip SOC Conference, Portland, OR, USA, 2003, pp. 189-192.

[15] D. Dal and N. Mansouri, “Power Optimization with Power Islands Synthesis”, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 28, no. 7, pp. 1025-1037, July 2009.

[16] K. Shin and T. Kim. “Leakage power minimization for the synthesis of parallel multiplier circuits”, Proceedings of the 14th ACM Great Lakes symposium on VLSI (GLSVLSI 04), Boston, 2004, pp. 166-169.

[17] C. Wang, X. Chang, Y. Zhang, L. Zhang and X. Chen, “Failure Analysis of Composite Structures Based on Digital Image Correlation Method”, International Conference on Sensing, Diagnostics, Prognostics, and Control (SDPC), Shanghai, 2017, pp. 473-476