Egyptian Computer Science Journal (ISSN-1110-2586)
Volume 40 – Issue1 January 2016

Symmetric Multi-Processing 3D Object Categorization Model Using a Spin-Point Curvature Selection Strategy

Doaa Hegazy
Scientific Computing Department, Faculty of Computer and Information Sciences,
Ain Shams University, Cairo, Egypt.
doaa.hegazy@cis.asu.edu.eg

Abstract

Due to significant improve in High Performance Computing (HPC) techniques, many researchers are motivated to use such HPC software and hardware platforms to enhance their computation models. Nowadays, HPC is used in many research fields. Both of object recognition and object categorization are of the best examples. In this paper, I introduce a novel scalable categorization system for Symmetric Multi- Processing (SMP) architectures. The proposed system is based on spin- image algorithm. It uses generated spin-images at certain spin-points as features to train a Support Vector Machine classifier (SVM). The new idea is to select certain spin-points based on maximum and minimum curvature of the 3D object, Also with the Bag of Features (BOF) [1] concept, I change the feature vector representation to increase SVM classifier performance. I compare between Mediod and Mean techniques in selecting the prototypes for BOF. Finally, I evaluate the new categorization system across two different datasets and promising results are obtained.

Keywords: *Spin-images, HPC, SMP, SVM, MPI, TPL, BOF, K-Means clustering, K-Mediod clustering.*

1. Introduction

The goal of computer vision is giving the human ability of seeing and understanding to the computer. Computer vision includes many branches to do such job. Both of object recognition and object categorization are important fields of computer vision. However, there is a difference between object categorization and recognition. While object categorization means making sense of shape concepts, object Recognition means identifying concrete shapes [2]. Object categorization can be considered as a superset of object recognition. Many applications depend on object categorization, even that it is used as a preparatory step before the recognition to enhance the recognition performance [3] [4] [5]. This idea is derived from the fact that human ability to recognize increases if they have information about object category. For example, to recognize a group of human faces, it is better to categorize them first as males, females, young, old, known person, unknown person ...etc.

Both object categorization and object recognition includes a heavy processing tasks. Therefore, there are two research trends to enhance the processing time. The first one is to reduce such needed tasks, not only without affecting recognition or categorization ability, but also trying to increase it. The second trend is applying High Performance Computing (HPC) techniques. In last decade, this trend has been encouraged due to rapid development in HPC software frameworks and hardware architectures. Some of HPC architectures are PCs clusters, Symmetric Multi-Processing (SMP), Grid, Cloud and GPUs computing. Some of HPC software frameworks are Message Passing Interface (MPI) for PCs cluster computing, Open Multi-Processing (Open MP) for SMP (multi-core architecture) and OpenCl for GPU computing.

Object Categorization systems include four main stages: pre-processing stage, feature extraction stage, classification, post-processing stage. Depending on feature selection and type of classifier, the quality of entire categorization system is determined. I mean by quality both of categorization performance and total system runtime.

In this paper, I present a 3D object categorization system based on SMP for multi-core architecture using Microsoft Task Parallel Library (TPL). The proposed categorization system is based on Spin-image features and Support Vector Machines (SVM) classifier. In the categorization system, I developed a new approach based on Bag of Features (BOF) technique with K-Mean and K-Medoid clustering. I have evaluated both of the different clustering techniques in the approach and promising results are obtained.

The paper is organized as follows: Section 2 discusses the related work. The proposed 3D categorization system is presented in section 3. In Section 4, I discuss datasets used in experimental evaluation. Moreover, I discuss the hardware and software platform specifications. The conclusions and future work are given in section 5.

2. Related work

The related work is organized in four categories. The first category focuses on using HPC techniques in object categorization and recognition. In this category, I discuss [6] [7] [8]. In [6] a parallel technique for extraction of features is introduced. The technique uses CPU and GPU by hybrid model of OpenMP and CUDA programming and it is employed by a mobile robot for object recognition purposes. The process for extracting key points is parallelized. It depends on parallel version of SIFT key point descriptor [9] while maintaining robust performance.

In [7], a real-time recognition and tracking system based on a parallel implementation for using OpenMP for ORB point descriptor [10] is presented. The result shows an improved processing speed.

In [8], a method to efficiently face detection and recognition is presented. It focuses on the use of Symmetric Multiprocessor (SMP) architectures to enhance detection and recognition time. Authors show that their work can be extended to any object detection and recognition algorithms. They gain in 2X speed up on dual core systems and aimed to prove their algorithm scalability across different SMP architectures. Although all of such categorization systems showed accepted results in terms of recognition accuracy and more better in terms of system run time, they do not use any depth information of objects. They depend on 2d images.

The second category is closer to our work. It uses the spin-image algorithm in object recognition and categorization. However, the majority of researches in this category do not consider time performance measurements and analysis. In this category we find [11] [12].

In [11], authors present a face recognition algorithm based on surface shape information inferred from image brightness using a Lambertian shape-from- shading scheme. Moreover, they used Spin-image Algorithm to perform sur- face representation. They tried to reduce the Spin-image computation effort by computing local spin-images on image patches. Although, their approach succeeded to reduce the number of extracted spin-images, the extraction process of one spin-image remains with same complexity.

In [12], authors introduced 3D object categorization approach based on spin-images and bag of features classification instead of surface matching. This approach shows success categorization rate of 65%.

The third category includes efforts to enhance spin-image (features enhancement) to increase recognition performance. In this category, I discuss [3] [13]. In [3], authors present a new enhanced algorithm called spin-image signatures. The new algorithm shows better results than the original one. It is more sensitive for 3D deformations. Although the new enhanced object recognition technique enhances recognition, it adds more computations to the original one to get signature of spin-images.

In [13], another enhanced the original spin-image algorithm. The enhanced version depends on 3D photometric information. Such enhancement leads to a new algorithm called textured spin-image, and used for 3D registration. It is more reliable than the original spin-image algorithm in terms of accuracy and more resilience for noise. Moreover, this new algorithm involves the same spin-image extraction procedure in addition to calculation related for photometric information.

The fourth category is more related to my work. This category includes researches using HPC techniques to build categorization system using spin- images and SVM classifier. In this category, I discuss [14] where a parallel version of spin-image extraction is introduced. The parallel implementation of such extraction is based on MPI. This work uses spin- images as categorization feature. Moreover, it overcomes the heavy feature extraction computation of Spin-image and produces a system with 6.36 seconds for single object categorization.

In the presented work, I provide an enhancement of the approached presented in [14] with a clear enhancement in terms of both success categorization rate and system runtime.

3. Proposed Categorization System

Figure 1 shows that my proposed categorization system consists of five stages. In first stage, I prepare the point cloud input to be in a form of connected mesh. While this process is being performed, the second stage done independently. The 3D point cloud is scanned to calculate a number of repetitions and corresponding locations for each depth in the image. The two stages are concurrent stages. Inside the second stage, there is another dimension for concurrency. Instead of single point scan with all other points to calculate depth repetitions (The Histogram Complexity is $O(n^2)$), I convert all depth values from floating point representation into integer representation by multi plying depth value 10^M where M represents maximum number of digits after floating point. I gained two advantages from this step: The first one is the floating point operations reduction. The second advantage is that all depth values (V) will be $0 <= V <=$ Max (depth) $* 10^M$. This means that each depth in this new domain can be attached to one worker to scan all points for its appearance. The histogram complexity is translated into O(n) when number of workers equal to depths set size.

3.1 Spin-image Extraction

In the third stage, spin-images are generated at points selected from the second stage as object features. Spin-image algorithm is a 3D shape descriptor introduced in [15]. Although it shows best results in 3D object retrieval, it is also used in recognition and categorization system [14] [12]. Spin-image is 2D image calculated for each point with known normal (oriented point). Figure 2 shows the process of Spin-image generation. Equations 1 and 2 show how 2D image [i,j] is computed based on two values α and β, where X is the oriented point and P belongs to 3D mesh oriented points. Also W and b represent image-width and bin-size respectively for the generated spin image.

$$(\alpha,\beta) = \left(\sqrt{\|x-p\|^2 - (n.(x-p))^2}, (n.(x-p))\right) \qquad (1)$$

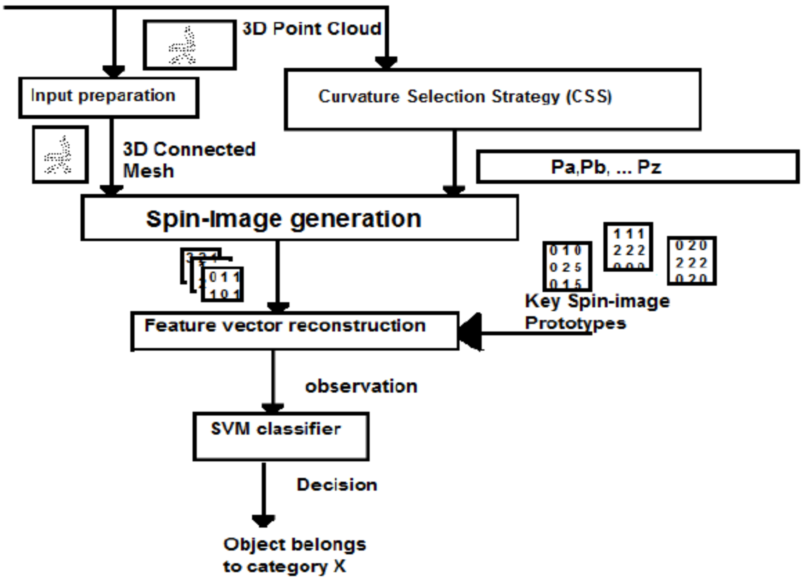

Figure 1: Proposed Categorization Model.

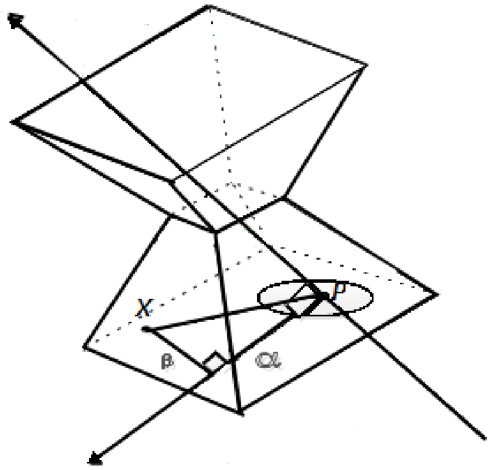

Figure 2: The generation process of spin-image at Point P with normal n for 3D object. α and β are calculation parameters, X is the oriented point and P belongs to 3D mesh oriented points

$$(i, j) = \left(\frac{\left(\frac{W}{2}\right) - \beta}{b}, \frac{\alpha}{b}\right) \tag{2}$$

Equation 1 shows that if we have two Point X and Y with known normals, the calculation of each Spin-image of them is independent from other one.

3.2 Spin-image Extraction Parameters

Extracting the Spin-images is affected by three main parameters, image-width, bin-size and support-angle. Image-width represents the number of rows or columns inside Spin-image, bin-size means size of storage for the Spin-image that reduces the effect of individual point positions. Support-angle represents the maximum allowed angle between the oriented point normal and the surface normal. According to [15], there is no specific values for these parameters. Depending on application, these parameters should be tuned. I have used an empirical approach to decide this value as in Figure 2.

3.3 Curvature Selection Strategy (CSS)

Unlike [14] that uses the random selection strategy (RSS) for generating Spin-images, which means a random selection for oriented, points where system will generate Spin-images. I use a proposed Curvature Selection Strategy (CSS). The main idea is to estimate some of landmarks for the 3D object. I assume that the maximum and minimum shape curvature can be used as these landmarks. I do not mean dividing 3D object into regions to local minimum and maximum, but I mean using global maximum and minimum curvature in 3D object. A depth histogram data structure is built to provide point depth repetition (histogram), point depth value and location (point x,y coordinates) of repetitions. Points with minimum histogram and maximum or minimum depth value are the target Spin-point. In order to reduce features number, I use the bag of features technique with some modifications.

Bag of Features(BoF) aims to simplify the object feature representation [1]. In BoF technique, object descriptors are computed first. Then a quantization technique is used to cluster these descriptors resulting in clusters centroids which represents the prototypes. The prototypes are known as the codebook. Feature vector are computed as histogram, each histogram bin is associated to one of the prototypes as shown in [12].

Instead of that and to generate the prototypes, If we have training samples S_0, S_1, ..., S_n , all Spin-images generated from selected point P1 (Based on CSS) from all samples belong to one cluster. We only have to pick this cluster centroid. I evaluate two different technique:

- Mean Centroid Technique (MCT): Assume we have Spin-images M_0 , M_1, ..., M_n ,so the target Spin-image (prototype) (T_p)is calculated according to Equation 3

$$T_p = \frac{1}{N} * M_i \quad \textbf{(3)}$$

- Mediod Centroid Technique (MDCT): Due to the fact that, MCT select the average Spin-image, the result spin-image may not belong to the spin-images domain. In Equation 4, MDCT picks the spin-image centroid based on minimum distance (L) from all other Spin-images.

$$L_x = \frac{1}{N} * \sum_{0<i<N} D(M_i, x)^2 \quad \textbf{(4)}$$

Where D is a distance difference function based on Euclidean distance

In the fourth stage, the feature vector is computed. The feature vector contains key Spin- images (prototypes bins) for each category respectively. Generated Spin-image for first point is compared to feature vector bins that represent the first key Spin-point of all categories. According to MCT or MDC one of feature vector bins is set to be one while others set to be zero. After feature vector reconstruction stage, classification stage is performed (final stage) where the SVM classifier decides for the object category.

Figure 3: Some samples of RGB-D objects

3.4 SVM classifier

The robust classification performance of SVM [16] classifier, is the main reason behind choosing it in the proposed categorization system. It has been used successfully in different classification applications (e.g. [17] and [18]) However, I used the SVM implementation for

LIBSVM. Although SVM is a binary classifier, LIBSVM provides an implementation for many class classifications. It is simple and easy-to-use support vector machines tool for classification (C-SVC, nu- SVC), regression (epsilon-SVR, nu-SVR), and distribution estimation.

4. Experimental Evaluation

4.1 Data

In order to compare the proposed categorization system with the categorization system presented in [14], I used the same Princeton Shape Bench- mark(PSB) dataset. It is public and available dataset that contains graphically 3D models. However, some concerns are taken on this dataset as it is synthetic, the images do not suffer from noise, as in the case sensory captured images. Therefore, to investigate the robustness of the proposed model more experiments using another dataset, RGB-D [20], are conducted. RGB-D is one of most common datasets to evaluate object categorization systems [21] [22]. The RGB-D is a dataset that consists of 300 different objects. This dataset was recorded using a Kinect camera. It has 51 different categories.

Figure 3 shows some RGB-D object samples. RGB-D has a subversion which contains manually segmented 3D point cloud. The depth image only contains one 3D object point cloud. Due to size of 83 GB for RGB-D, we only use 16 different classes. These classes are organized into three main categories (Inanimate Things (IT),Electronic Devices (ED) and Fruit and vegetables(FV)). Table 1 shows the organization of the used three classes.

Table 1: Distribution of RGB-D classes into our three categories.

Category Name	Classes
Inanimate Things (IT)	Binder Food bag Food Can Cereal box Kleenex
Electronic Devices (ED)	Calculator Camera Cell phone Flash light keyboard
Fruit and vegetables(FV)	Apple Lemon Orange Peach Garlic Potato

4.2 Platform Specification

I select SMP as our High Performance Computing (HPC) development methodology [23]. Processor became more powerful and contains more than one core connected throw bus or cross bar switches, each core has its own ALU, registers and caches, but all cores have a shared memory. Like PCs cluster development methodology (MPI), SMP has many different software implementations that facilitate the development. The proposed categorization systems is built using C# language, with the Task Parallel Library (TPL) provided in dotNet Framework 4. The TPLa set of APIs is for thread man- agreement. TPL aims to hide complexity of thread management to perform concurrent operations. The TPL automatically scales the code use all the processors that are available on the target machine, developers do not have to concern about available number of processors (workers).

The system implementation is run on a powerful server (http://msdn.microsoft.com/en-us/library/dd460717(v=vs.110).asp) that has Intel Quad Core x5570 Xeon CPU 2.93ghz, RAM of 24 GB and Windows 64 bit operating system.

4.3 Results

There is a significant effect for changing RSS to CSS in Spin-point selection as revealed by the experimental results discussed below. Table 2 shows a comparison between RSS technique in [14] and the new categorization system based on either MCT or MDCT in terms of mean success categorization.

Moreover, I give a comparison among the three techniques based on run-time over the hardware and software specification mentioned before. Table 3 shows that system runtime for the two techniques of (CSS with MCT or CSS with MDCT) are close to each other, however their performance outperforms that is of the RSS technique.

Table 2: Comparison between Random Selection Strategy (RSS) and Curvature Selection Strategy (CSS) for both Medoid and Mean Key spin point, spin- image generation parameters: bin-size=0.1,image-width=20 and support- angle=60,SVM parameters: RBF kernel function with σ = 8 and C = 32 for PSB, σ = 512 and C = 4 for RGB-D, the number of spin-points=100

Table 3: System Runtime for RSS -as in [14]- technique, CSS with MCT and CSS with MDCT

Technique	Success categorization rate	Dataset
RSS[14]	65 %	PSB
CSS with MCT	70.65%	PSB
CSS with MDCT	76.11 %	PSB
CSS with MCT	69.991 %	RGB-D
CSS with MDCT	77.235 %	RGB-D

Table 4: Success categorization rates for three different categorization techniques with and varying the no. number of spin- points

Technique	Overall system runtime (seconds)
RSS	0.596511
CSS with MCT	0.655895
CSS with MDCT	0.70021

No. of Spin points	**CSS with MCT**	**CSS with MDCT**	**RSS**
20	37.22	37.22	46.70
40	50.00	53.88	50.60
60	53.32	62.77	47.10
80	63.88	65.00	56.70
100	70.55	76.11	65.00

Also Table 4 shows the success categorization rate of the three techniques at different number of spin-points. This comparison is performed over PSB dataset. Although CSS with MDCT technique and CSS with MCT technique show a close categorization performance at low number of Spin points (40 points), CSS with MDCT technique shows better performance as the number of Spin-points increases. However, both curvature based selection techniques yield superior performance to RSS technique.

5. Conclusions and Future Work

Due to the absence of recursive methods in Spin-images generation and few memory needs for such generation, the performance of SMP architectures is better than PCs Cluster. The problem of heavy computation needed to generate Spin-images is overcame much better with HPC technique of SM development. Moreover, I can conclude that transforming extracted features vector to simplified discrete form using Mean or Mediod selection -to construct feature vector bins- has a significant impact on success categorization rate. However, the experimental results have showed that Mediod selection yields better results than the Mean selection. Although the proposed new categorization system have shown promising results for different datasets, the overall system runtime is 0.6 seconds which means 1.6 frames per second (fps). In order to reach the real-time categorization constraint [24] we need to decrease system runtime. I intend to use another HPC development techniques such hybrid platforms of CPUs and GPUs. Moreover, I aim in the future researchto measure our categorization system against cluttered and occluded scenes.

6. Acknowledgments

I appreciate and acknowledge the triangulation technique and its Matlab Implement-tation for Prof. Dr. Ajmal Saeed Mian Computer Science. Univ of Western Australia. Moreover, I give thanks to Dr. Mohamed Hamdy from the information network at Ain Shams University for access such serve their intel quad core x5570 Xeon CPU 2.93ghz, RAM of 24 GB and Windows 64 bit operating system.

References

[1]. G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray, “Visual Categorization with Bag of key points,” in Proc of Workshop on Statistical Learning in Computer Vision, ECCV, 2004, pp. 1–22.

[2]. S. Dickenson, A. Leonardis, B. Schiele, and M. J. Tarr, Object Categorization: Computer and Human Vision Perspectives. Cambridge University Press, September 2009.

[3]. K. Veropoulos, G. Bebis, and M. A. Webster, "Investigating the Impact of Face Categorization on Recognition Performance." In ISVC, ser. Lecture Notes in Computer Science, G. Bebis, R. D. Boyle, D. Koracin, and B. Parvin, Eds., vol. 3804. Springer, 2005, pp. 207–218.

[4]. A. J. O'Toole, J. Peterson, and K. A. Deffenbacher, "An Other-race Effect' for Categorizing Faces by Sex," Perception, vol. 25, 1996, pp. 669 – 676.

[5]. Y. D. Cheng, A. J. O'Toole, and H. Abdi, "Classifying Adults' and Children's Faces by Sex: Computational Investigations of Sub-Categorical Feature Encoding," Cognitive Science, vol. 25, no. 5, 2001, pp. 819–838.

[6]. J. Kim, E. Park, X. Cui, H. Kim, and W. Gruver, "A Fast Feature Extraction in Object Recognition Using Parallel Processing on CPU and GPU," in Proc of IEEE International Conference on Systems, Man and Cybernetics (SMC 2009),2009, pp. 3842–3847.

[7]. J. C. Kwak, T. R. Park, Y. S. Koo, and K. Y. Lee, "Implementation of Object Recognition and Tracking Algorithm on Real-time Basis," in EU- ROCON, 2013 IEEE, July 2013, pp. 2000–2004.

[8]. T. Deepak Shekhar and K. Varaganti, "Parallelization of Face Detection Engine," in Proc of 39th International Conference on Parallel Processing Workshops (ICPPW), 2010, pp. 113–117.

[9]. D. G. Lowe, "Distinctive Image Features From Scale-Invariant Keypoints," Int. J. Computer. Vision, vol. 60, no. 2, pp. 91–110, 2004.

[10]. E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An Efficient Alternative to SIFT or SURF," in Proc of IEEE International Conference on Computer Vision (ICCV), 2011, pp. 2564–2571.

[11]. M. Bock, G. Cortelazzo, C. Ferrari, and C. Guerra, "Identifying Similar Surface Patches on Proteins Using a Spin-image Surface Representation," in Combinatorial Pattern Matching, ser. Lecture Notes in Computer Science, A. Apostolico, M. Crochemore, and K. Park, Eds. Springer Berlin Heidelberg, 2005, vol. 3537, pp. 417–428. [Online]. Available: http://dx.doi.org/10.1007/11496656

[12]. M. Baum, D. Agnes, and D. Sven, "Using Spin Images for 3D Object Classification," Master's thesis, Applied Informatics Group,Bielefeld, 2011.

[13]. N. Brusco, M. Andreetto, A. Giorgi, and G. Cortelazzo, "3D Registration by Textured Spin-Images," Fifth International Conference in 3-D Digital Imaging and Modeling, 2005 (3DIM 2005), pp. 262–269.

[14]. A. Eleliemy, D. Hegazy, and W. Elkilani, "MPI Parallel Implementation of 3D Object Categorization Using Spin-images," in Proc of 9th International conference in Computer Engineering (ICENCO), 2013, pp. 25–31.

[15]. A. Johnson, "Spin-images: A representation for 3-D Surface Matching," Ph.D. dissertation, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, August 1997.

[16]. C. Cortes and V. Vapnik, "Support-vector Networks," Machine Learning, vol. 20, no. 3, pp. 273–297, 1995.

[17]. N. O. S. Noha, A. Hikal and F. W. Zak, "Real Time Facial Expression Recognition Based On Hierarchal SVM," Egyptian Computer Science Journal, vol38, issue no. 1, 2014, pp. 30-38.

[18]. A. B. Adeyemo, A. O. Abiodun and I. Nigeria, "Adaptive SIFT/SURF Algorithm for Off-line signature Recognition," Egyptian Computer Science Journal, vol39, issue no. 1, 2015, pp. 50-56.

[19]. H. Song, K. K. Choi, I. Lee, L. Zhao, and D. Lamb, "Adaptive Virtual Support Vector Machine for Reliability Analysis of High-dimensional Problems." Structural and Multidisciplinary Optimization. vol. 47, pp. 479–491, 2013.

[20]. L. Bo, X. Ren, and D. Fox, "Hierarchical Matching Pursuit for Image Classification: Architecture and Fast Algorithms," in Advances in Neural Information Processing Systems 24, 2011, pp. 2115–2123.

[21]. K. Lai, L. Bo, X. Ren, and D. Fox, "A large-Scale Hierarchical Multi-view RGB-D Object Dataset," In Proc of IEEE International Conference on Robotics and Automation (ICRA), 2011, pp. 1817–1824.

[22]. J. M. Gomez, M. Cazorla, I.G. Varea and C. Gonzalez, " Object Categorization From RGB-D Local Features and Bag of Word," in Proc of 2nd Iberian Robotics Conference (Robot 2015): Advances in Robotics, vol. 2, 2015, pp. 635-644.

[23]. L. Bo, Z. Zhenliu, and W. Xiangfeng, "A survey of HPC Development," In Proc. of International Conference in Computer Science and Electronics Engineering (ICCSEE), 2012, vol. 2, March 2012, pp. 103–106.

[24]. J. Oh, G. Kim, I. Hong, J. Park, S. Lee, J.-Y. Kim, J.-H. Woo, and H.- J. Yoo, "Low-power, Real-Time Object Recognition Processors for Mobile Vision Systems" Micro, IEEE, vol. 32, no. 6, pp. 38–50, 2012