ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 3, Issue 1, pp: (442-449), Month: January - March 2015, Available at: www.researchpublish.com

Acceleration of Image Retrieval System Using CUDA Based Parallel Computing On GPU

[1]Shingade Suraj, [2]Patil Ganesh, [3]Bhamare Sameer, [4]Kumar Saurabh

[1,2,3,4] Computer Engineering in Savitribai Phule University

Abstract: **Image Retrieval instruments can aid individuals in making productive utilization of computerized picture accumulations; likewise it has gotten to be basic to discover productive techniques for the retrieval of these pictures. Most picture preparing calculations are innately parallel, so multithreading processors are suitable in such applications. In huge picture databases, picture preparing takes long time for run on a solitary(single core processor) center processor due to single string execution of calculations. GPU is more basic in most picture transforming applications because of multithread execution of calculations, programmability and minimal effort. In this paper we execute shading minutes and surface based picture Retrieve (entropy, standard deviation and neighborhood range) in parallel utilizing CUDA programming model to run on GPUs. These features are connected to inquiry pictures from a database which is like an inquiry picture. We assessed our image retrieve framework utilizing review, exactness, and normal accuracy measures. Exploratory results demonstrated that parallel execution prompted a normal accelerate of 144.67×over the serial execution when running on a NVIDIA GPU GeForce GT610M. Additionally the normal exactness and the normal review of proposed strategy are 61.968% and 55% individually.**

Keywords**: CBIR, Color moments, CUDA, GPU.**

I. INTRODUCTION

Advances in the internet and digital imaging have resulted in an exponential increase in the volume of digital images. The need to find a desired image from a collection of databases has wide applications, such as, in crime prevention by automatic face detection, finger print, medical diagnosis, to name a few. These days for such applications, Images are broadly utilized and picture Retrieval as one of the fascinating utilizations of picture preparing is a suitable case to be executed in parallel. Since in this application, pictures are generally partitioned into a few sections and every part is prepared independently and also to one another. The picture retrieve frameworks utilize low-level visual features like shape, texture and color. Ordinarily, picture retrieve frameworks remove this low level features in the logged off stage and utilize these features for coordinating as a part of the online stage.

For this, the visual contents of the pictures in the database are separated and it is depicted by multidimensional feature vectors, which are more packed and simpler to process. Content based image retrieve (CBIR), a method which utilizes visual substance to look pictures furthermore for separating comparative pictures from extensive scale picture databases. It files visual qualities of a picture, for example, its shading, shape and composition to search for a picture in a picture database.

In CBIR system, it is usual to group the image features in three main classes: color, texture and shape. Ideally, these features should be integrated to provide better discrimination in the comparison process. Object shape features can also provide powerful information for image retrieval, because humans can easily recognize objects solely from their shapes. Shape representation strategies incorporate Fourier descriptors, polygonal estimate, invariant moment, B-splines, deformable formats, and bend scale space. Area based and boundary based strategies (otherwise called form based methodologies) are two essential routines for the shape depiction methods. The area based strategies consider the entire zone of the article and frequently utilize moment descriptors, for example, geometrical moments, Zernike moments, pseudo-Zernike moments, and Legendre moments to depict shapes. Indeed, in area based techniques, shape descriptors use data from both limits and inside locales of the shape. This technique is worldwide in nature and can be connected to

bland shapes, yet they neglect to recognize comparable items. The limit based shape descriptors have a tendency to be more proficient for taking care of shapes that are depicted by their item forms. The most well-known boundary based shape descriptors are chain codes, Fourier descriptors, wavelet descriptors, and form dislodging. Shape features are less created than their color and texture partners in view of the intrinsic unpredictability of speaking to shapes.

The color features are the most widely used visual features in image retrieval because they are easier to extract compared with texture and shape information. Color feature is relatively robust to background complication and independent of image size and orientation. The color distribution, the mean worth and the standard deviation, to speak to the worldwide attributes of the picture, and the picture bitmap is utilized to speak to the neighborhood qualities of the picture for expanding the exactness of the Image retrieve framework. The most widely recognized technique for shading based picture retrieve is the shading histogram that portrays the worldwide shading conveyance in a picture. The primary disadvantage of shading histogram is that it just gives a general evidence of the shading substance of a picture yet does not give data about the spatial conveyance of the hues. Hence two separate pictures with comparable shading appropriation can create comparative histograms. As tigers and cheetahs have same hues however diverse examples, likewise utilizing shading feature alone can't recognize them.

Texture is an essential feature of characteristic pictures. An assortment of methods has been created for measuring texture likeness. The greater part of the systems depends upon looking at estimations of what are known as second-arrange insights ascertained from inquiry and put away pictures. These techniques ascertain measures of picture surface, for example, the level of complexity, coarseness, directionality and consistency; or periodicity, directionality and arbitrariness. Elective techniques for texture investigation for picture retrieve incorporate the utilization of Gabor channels and fractals. Gabor channel (or Gabor wavelet) is broadly embraced to concentrate texture features from the pictures for picture Retrieve, and has been demonstrated to be exceptionally effective. Picture retrieve utilizing Gabor features beats that utilizing Pyramid-organized wavelet change (PWT) feature, tree-organized wavelet change (TWT) feature and multi determination synchronous autoregressive model (MRSAR) features. Thus, in our proposed technique, Gabor channel is utilized for extraction of texture features.

Entropy, standard deviation and nearby range are methodologies for extraction of the texture offers in pictures that can be productive in picture indexing and to concentrate the color features, the color moments incorporate mean, standard deviation, and skewness is utilized. In this work, to concentrate the visual features, the picture is partitioned into little non-covering picture squares and every picture square is handled like other picture pieces. So in every picture obstruct, the visual features are calculated independently. Indeed, the same direction is executed by various processors utilizing diverse information streams (SIMD). Thus these methodologies can be performed in parallel. To quicken visual feature extraction, multi center processors can be valuable on account of multithreading execution of projects.

As of late, GPUs have gotten to be progressively more attractive for broadly useful parallel computations. Parallel codes exploiting GPU equipment may yield results identical to several traditional CPUs, at a small amount of the expense. Graphical Processors Units (GPU) assume essential part for speedup the preparing of database pictures coordinating calculations in light of the fact that it has more inbuilt execution cores. Numerous scientists have as of now been connected GPUs to execute numerous calculations in different regions, for example, picture transforming, computational geometry, and logical calculation, and also PC design. Parallel executions on GPUs have been connected to different numerical issues to lessen the calculation time without sacrificing the level of exactness. Figure Unified Device Architecture (CUDA) programming model discharged by NVIDIA gives an arrangement of negligible expansions to the C programming dialect that permit the software engineer to compose bits executed in parallel on the GPU. In this paper, parallel usage is performed for color moments, entropy, and standard deviation and nearby range utilizing CUDA to run on the GPU. Additionally our work utilizes broad use of exceptionally multithreaded construction modeling of multi-core GPU.

II. THE GPU ARCHITECTURE

As of late, GPUs have gotten to be progressively more alluring for broadly useful parallel calculation. Fig.1. Shows Basic structure of NVIDIA GPU which directly with CUDA drivers. A GPU has a progression of Multiprocessors (MP), and every multiprocessor contains 8 Scalar Processors (SP). Every last MP has its own particular memory bank and The memory bank of every MP can be gotten to by all its SPs and worldwide memory of GPU is imparted over all the MPs.

ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 3, Issue 1, pp: (442-449), Month: January - March 2015, Available at: www.researchpublish.com

CUDA is one of the parallel processing building design grew by NVIDIA to actualize calculations with utilization of GPUs. A CUDA Program basically comprise of two sorts of the code. Initially sort is Host Code and second is Device code, where CPU is the host and GPU is the gadget. The hosts i.e. CPU performs all the nonparallel operations and passes information to GPU in the worldwide memory and dispatches a part and information parallel shares of an application are actualized as bits.

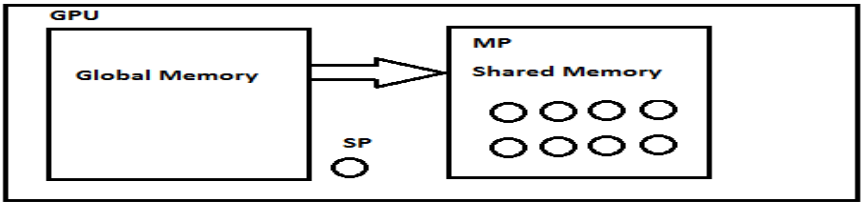

Fig.1. the NVIDIA GPU Architecture

The bit executes the processing utilizing parallel threads on the SPs. As indicated by CUDA, information parallel parcels of an application are actualized as bits. The primary CPU, going about as the host, can start one part at once. Every portion is executed in parallel by a few threads. These threads are gathering into blocks and blocks are gathered further into grid. A thread block is a 3, 2 or 1-dimensional gathering of tread. Threads inside a block can participate or communicate among themselves by sharing information through some shared memory and synchronizing their execution to facilitate memory gets to thread in diverse pieces can't coordinate. The quantity of threads for every block is obliged by the constrained memory assets of a processor core.

CUDA Coding Product:

The CUDA parallel programming model stresses two key outline objectives. To start with, it means to develop a standard successive programming dialect, particularly C/C++, with a moderate arrangement of deliberations for communicating parallelism. Extensively talking, this lets the software engineer concentrate on the imperative issues of parallelism—how to specialty effective parallel calculations as opposed to thinking about the mechanics of a new and confounded dialect. Second, it is intended for composing exceptionally versatile parallel code that can run crosswise over countless simultaneous threads and many processor cores. This is essential in light of the fact that the physical parallelism of current NVIDIA GPUs ranges from eight processor centers and 768 string connections up to 240 processor centers and 30,720 string settings. The CUDA demonstrate characteristically manages the developer to compose parallel projects that straightforwardly and productively scale over these diverse levels of parallelism.

A CUDA project is sorted out into a host system, comprising of one or more successive threads running on the host CPU, and one or more parallel kernels that are suitable for execution on a parallel transforming gadget like the GPU. A kernel executes a scalar successive program on an arrangement of parallel strings. The software engineer arranges these threads into a lattice of thread blocks. The threads of a single thread block are permitted to synchronize with one another by means of obstructions and have admittance to a high velocity, every square imparted on-chip memory for inter thread communication. Threads from different blocks in the same grid can arrange just by means of operations in a shared worldwide memory space unmistakable to all threads. Fig. 2. Shows the multithread Execution using CUDA.

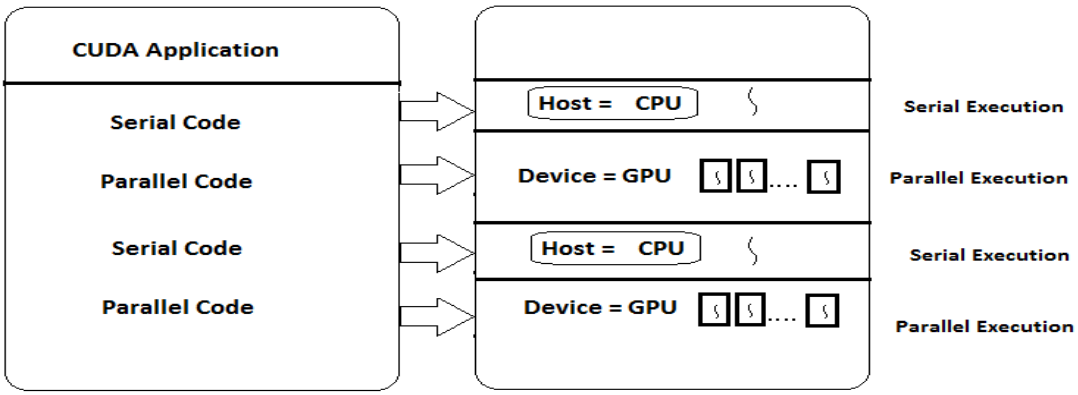

Fig.2. Multithreaded Execution using CUDA

CUDA Architecture Consist of

- CUDA Libraries
- Application/MATLAB
- CUDA Runtime
- CUDA Driver

CUDA presents the idea of grid, block and thread to manage them, Because of the large measure of threads. In a network, the blocks are overseen in x, y and z bearings. The measurements for a framework can be given as the variable gridDim, whose segments can be gridDim.x, gridDim.y and gridDim.z. The measurements for a block are given by the variable blockDim, whose segments can be given as blockDim.x, blockDim.y and blockDim.z. The list for block is given by blockIdx, which may contain the parts of blockIdx.x, blockIdex.y and blockIdx.z. So threads are overseen in a 3D manner in a block and the thread list is given by threadIdx (threadIdx.x, threadIdx.y, threadIdx.z). Some of capacities are as given underneath:

/Commands for duplicate the inputs to gadget or device

cudaMemcpy(dev_a, a, size, cudaMemcpyHostToDevice);

cudaMemcpy(dev_b, b, size, cudaMemcpyHostToDevice);

/ Charge for dispatching include () portion with N and N pieces

add<<<N, N >>> (dev_a, dev_b, dev_c);

/ Commands for duplicate the consequence of device to host duplicate of c

cudaMemcpy(c, dev_c, size, cudaMemcpyDeviceToHost);

When all is said in done the variables are incorporated as a parameter of the capacity as takes after:

Dim3 grid(x, y, 1), block (m, n, p)

func_gpu <<<grid, block>>> (a, b, c)

Here a, b, and c are the working parameters and m, n, and p are the quantity of strings in x, y and z headings in a block. x, y and 1 are the quantity of pieces in x, y, and z bearings in a framework. In capacity fun_gpu the piece of <<grid, block>>> is included for GPU-capacity call.

III. THE PROPOSED IMAGE RETRIEVAL PROCEDURE

This Section displays some information about the key thoughts of CBIR space and the proposed picture retrieve framework.

A. The Basic Concepts of CBIR Systems

Fig.3. Shows the CBIR Architecture. The methodology of picture retrieval comprises of two errands incorporate indexing and retrieval. Features are the agents of the pictures. Indexing means portrayal of pictures in view of picture properties. The rates of retrieval productivity completely depend on determination of fitting picture characteristics. Two primary prerequisites of picture retrieval are high retrieval precision and less computational many-sided quality. The Image retrieval process comprises of computing a feature vector for describes some picture properties, and put away in the picture characteristic database. The client gives an inquiry picture, and the picture retrieval framework processes the feature vector for it, and contrasts it and the specific picture characteristic database pictures. The pertinence examination is performed by utilizing some separation estimation systems and the base or admissible separations are the measurements for the coordinated or comparative pictures.

B. Extraction of the Color Features in Images

Content based picture retrieval is viewed as one of the most proficient routines for getting to visual information and is to investigate picture data by low-level peculiarities of a picture, which incorporate color, texture and shape. There are different techniques which can be utilized for retrieval of the image that one of them is color moments. The sub-picture is

characterized by isolating the picture space into 4×4 non covering squares. Accordingly, the picture segment dependably yields little equivalent estimated sub-pictures notwithstanding of the extent of the first picture. To portray the sub- picture, we then create color moments for every sub-picture. The primary request (mean), the second (standard deviation), and the third-arrange (skewness) color moments have been ended up being effective and successful in speaking to shading conveyances of pictures. In the event that the estimation of the i^{th} color shading channel at the j^{th} picture pixel is P_{ij}, then the shading minutes or color moments are as per the following.

Minute 1: Mean

$$\mu_{i=}\frac{1}{N}\sum_{j=1}^{N} P_{ij}$$

Minute 2: Standard deviation

$$\sigma_{i=}\sqrt{\sum_{j=1}^{N}\left(P_{ij}-\mu_i\right)^2/N}$$

Minute 3: Skewness

$$S_{i=3}\sqrt{\sum_{j=1}^{N}\left(P_{ij}-\mu_i\right)^3/N}$$

For making of peculiarity database above strategy is rehashed for all the pictures of the picture database and postulations characteristic vectors are put away in peculiarity database.

C. Extraction of the Texture Features in Images

To concentrate the composition characteristics, entropy, nearby range and standard deviation measures are utilized as execution parameters.

Texture = (Entropy + Standard deviation + Local Range)

Entropy is a factual measure of arbitrariness that can be used to describe the surface of the info picture. The worth of entropy can be ascertained as:

$$ENT=\sum_{k=1}^{M}\left(\mathrm{P_k}\log\frac{1}{P_k}\right)$$

Where, ENT, M, and P are entropy, aggregate number of tests, and likelihood of events separately. Standard deviation can be ascertained as:

$$S=\sqrt{\sum_{i=1}^{n}(X_i-\bar{X})^2/n}\quad \bar{X}=\frac{\sum_{i=1}^{n}X_i}{n}$$

Where, n is number of components in the example. Neighborhood range is greatest estimation of picked pixel-least estimation of picked pixel. On the off chance that x and y are two d-dimensional peculiarity vector database picture and question picture individually, the Euclidean is

$$d_{E\,(x,y)=}\sqrt{\sum_{i=1}^{d}(x_i-y_i)^2}$$

At the point when a question picture is submitted for Image retrieval, its visual feature is extricated and coordinating operation is performed between question picture features and the picture peculiarities put away in database, the outcomes closes to the question picture is retrieved from the database. Every Image is two-dimensional space that can be mapped to CUDA threads. There are 16 blocks in network furthermore 64 threads in every block. Thus, for concentrate visual peculiarities, every sub- picture is mapped to one block and every picture square is handled by one thread. In this way offer extraction is performed in all picture obstructs in parallel by CUDA threads.

ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 3, Issue 1, pp: (442-449), Month: January - March 2015, Available at: www.researchpublish.com

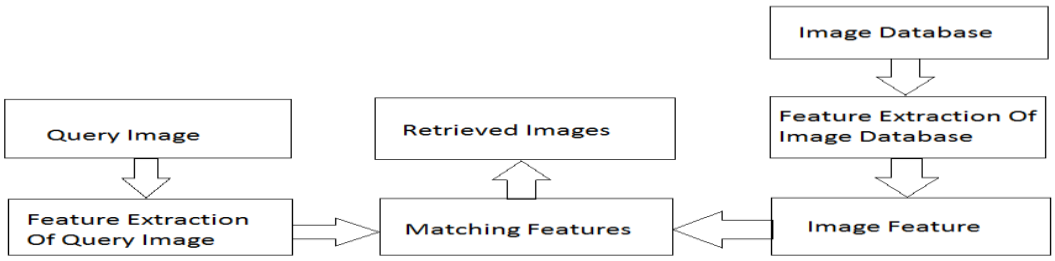

Fig.3. Architecture of CBIR system

IV. EXPERIMENTAL RESULTS

To assess the strength and effectiveness of the proposed technique, we utilize two elements, specifically exactness i.e. precision and review i.e. recall. Exactness (Precision) measures how well the retrieved things coordinate the referred query and recall uncovers the rate of significant archives that are images retrieved by the module.

$$Precision = \frac{\text{Number of relevant images retrieved}}{\text{Total number of images retrieved}}$$

$$Recall = \frac{\text{Number of relevant images retrieved}}{\text{Total number of relevant images}}$$

The average precision for the pictures that fits in with the qth class (Aq) has been figured by:

$$AP = \sum_{k \in A_q} \frac{p(i_k)}{|A_q|}$$

Serial usage of the picture retrieve system is carried out in C dialect utilizing a PC with Intel Core i5 Pentium Processor (2.5GHz) and 4GB RAM. Parallel execution acquired a normal accelerate of 144.67×over the serial usage when running o n a GPU named NVIDIA GeForce GT610M. The GeForce GT610M is a section level card with 48 CUDA cores and 900MHz core clock speed. Fig. 4 demonstrates the outcomes created from our proposed framework that demonstrate the effectiveness of our proposed approach and have a normal image retrieval time as 7.6 seconds. These outcomes demonstrate that the execution of the proposed system is superior to alternate techniques. The normal exactness and the normal review of our proposed system are 61.968% and 55% individually.

In the proposed technique, the retrieval precision is displayed in term of Precision-Recall bend. It gives an important result when the database is known and has been utilized be some prior frameworks. We select all pictures from every class in the database to utilize them as queries to calculate the precision and recall. For every picture, the exactness of the retrieval result is gotten by expanding the quantity of retrieved pictures. Fig.4. Demonstrates the Precision-Recall chart for the proposed technique. From the Figure, we notice that the framework has great exactness comes about over the distinctive estimations of recall. For the proposed technique, the greatest normal precision of 100% at recall quality is 10%, and the precision worth reductions to 49.94% at 100% of review.

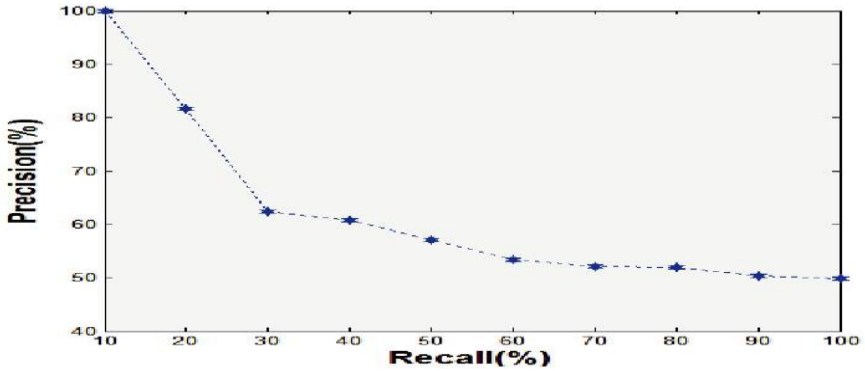

Fig.4. The Average precision / recall chart of the presented system

ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 3, Issue 1, pp: (442-449), Month: January - March 2015, Available at: www.researchpublish.com

Content based image retrieve is a system utilized for extracting comparative pictures from a picture database with a downside of a ton of time utilization. To make speedier the system, we parallelized it on CUDA. GPU parallel processing can give a speedup of 144.67 over CPU consecutive handling. Along these lines, it shows that parallel processing utilizing a GPU can accomplish great execution.

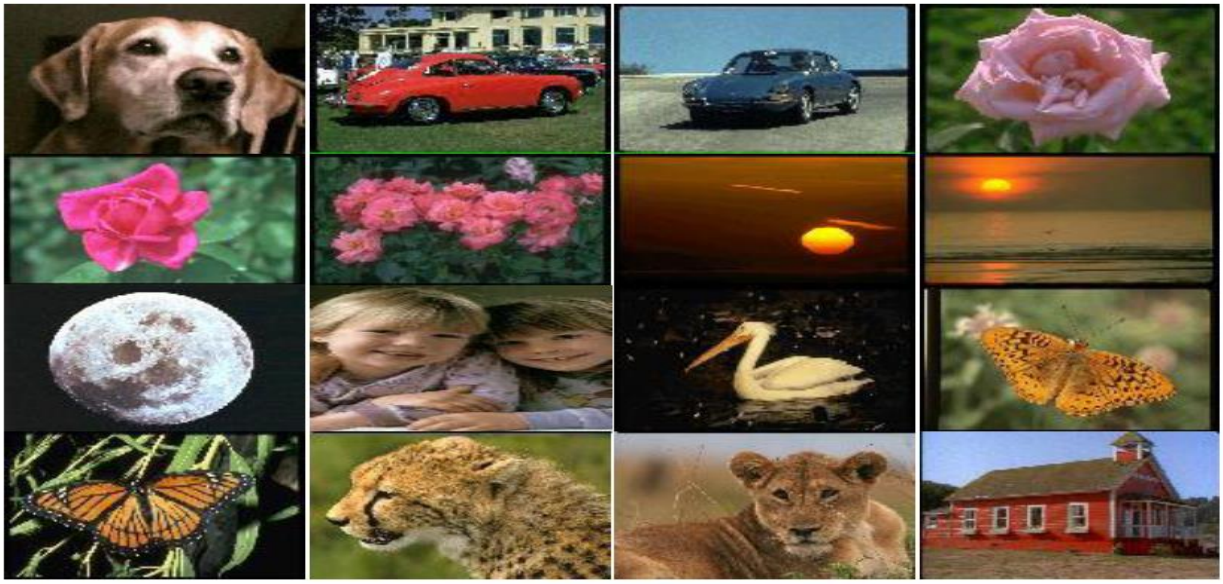

Fig.5. Sample Image Database

(a) House Query, the top 4 retrieved images

(b) Butterfly, the top 4 retrieved images

V. CONCLUSION AND FUTURE WORK

We have introduced a GPU parallel figuring calculation for Content based picture Retrieve utilizing color and texture features. Actually, in this paper, parallel execution was performed for color moments and texture based picture retrieval (entropy, standard deviation and neighborhood extent) utilizing CUDA programming model to run on GPU. In the investigations on a substantial database, it is demonstrated that, utilizing a parallel usage on the GeForce GT610M GPU is 144.67 times quicker than a CPU form on the Core i5, while giving literally the same numerical results. The accuracy, review and normal exactness estimations were utilized to depict the execution of picture retrieval process.

In Future, we can build up a framework that joins the composition, shape, and spatial features with the color feature to speak to the picture, so that the retrieval proficiency can be further increments. Also, the parallel usage on current GPU gets ten times speedup; thusly, it can be normal that the pace up in transforming time will be progressed.

ACKNOWLEDGEMENT

This exploration has been empowered by the utilization of registering assets gave by Computer Engineering Department of Karmaveer Kakasaheb Wagh College of Engineering and Research, Nashik, India.

REFERENCES

[1] H. Kekre, S. Thepade, and A. Maloo, “Image Retrieval Using Fractional Coefficients of Transformed Image Using DCT and Walsh Transform”, International Journal of Engineering Science and Technology, Vol. 2, No. 4, pp. 362-371, 2010.

[2] J. Vogel and B. Schiele, “Performance evaluation optimization for content based image retrieval”, Pattern Recognition 39, pp. 897-909, 2006.

[3] D. Tralic, J. Bozek, and S. Grgic, “Shape analysis and classification of masses in mammographic images using neural networks”, 18th International Conference on Signal and Image Processing, pp. 1-5, 2011.

[4] T. Lu and C. Chang, “Color image retrieval technique based on color features and image bitmap”, Information Processing and Management 43, pp. 461-472, 2007.

[5] M. Kokare, B. Chatterji, and P. Biswas, “Cosine-modulated wavelet based texture features for content-based image retrieval”, Pattern Recognition Letters 25, pp. 391-398, 2004.

[6] D. Zhang, A. Wong, M. Indrawan, and G. Lu, “Content-based Image Retrieval Using Gabor Texture Features”, IEEE Transactions PAMI, pp. 1-4, 2000.

[7] I. Andreou and N. Sgouros, “Computing, explaining and visualizing shape similarity in content-based image retrieval”, Information Processing and Management 41, pp. 1121-1139, 2005.

[8] D. Donno, A. Esposito, L. Tarricone, and L. Catarinucci, “Introduction to GPU Computing and CUDA Programming: A Case Study on FDTD”, IEEE Antennas and Propagation Magazine, Vol. 52, No. 52, pp. 116- 122, 2010.

[9] S. Arivazhagan, L. Ganesan, and S. Selvanidhyananthan, “Image Retrieval using Shape Feature”, International Journal of Imaging Science and Engineering (IJISE), Vol. 1, No. 3, pp. 101-103, 2007.