Multithreaded Approach for Registration of Medical Images using Mutual Information in Multicore Environment and its Applications in Medical Imaging

Sanjay Saxena
School of Biomedical Engineering
Indian Institute of Technology

Neeraj Sharma, Ph.D.
School of Biomedical Engineering
Indian Institute of Technology

Shiru Sharma, Ph.D.
School of Biomedical Engineering
Indian Institute of Technology

ABSTRACT
Image Registration plays very crucial role in case of medical imaging to register different modalities of images like CT (Computed Tomography) and PET (Positron Emission Tomography) registration. CT is essential for structural information of anatomic and PET (Positron Emission Tomography) is for functional information. Basically it is the procedure of transforming dissimilar sets of data into one coordinate system. These sets of data can be acquired from multiple image modalities, different viewpoints, similar or dissimilar sensors. MI based image registration has been found to be reasonably useful methods of image registration. However, it is found to be quite computationally intensive and time consuming process for enormous size images and for different data sets of images. It involves steps for computation of joint histogram, marginal entropies, calculation and probability distribution. Main motive of this paper is to provide an intelligent method for image registration based on Mutual Information using multi core environment with maintaining the synchronization between different activated cores and processors. Proposed Method has been able to execute with different number of threads to achieve all the remuneration of the processors and gives significant speedup working with verity of images like gray scale, RGB and Dicom images with different size. Finally the designed algorithm has been used to register medical images of different modalities.

Keywords
Image registration, Parallel computing, Mutual Information, Medical Images, Multithreading, CT, PET.

1. INTRODUCTION
Image registration (IR) is extremely essential task of Image Processing as it is the procedure of aligning two images so that the point in one image corresponds to the same anatomical position in the other. It is a key part in the medical imaging analysis. Medical images are often taken at different time and places, resulting in varying frame of references for the same part of the human body in the images [1]. Image registration plays vital role in the registration of different modalities of medical images like CT – PET registration and many more. MI based image registration is very valuable as it is based on the statistical relationship between two images and most important thing is that segmentation is not necessary prior to MI based image registration. Apart from the importance of MI based image registration it is quite computationally expensive [2]. Parallel computing seems to be the only solution for fast and efficient computing as by using this we can use non local resources very efficiently with suitable multithreading. Importantly it removes the limits of serial computing [3]. Now day's multicore processors include several processing elements within an integrated circuit and this can be considered as parallel solution which removes the limitation of sequential computing [4-7].

For implementing MI based Image Registration we need to consider following basic concepts

1.1 Entropy
It was introduced by Shannon in 1948 [8], and is defined as

$$H(X) = \sum_{x \in X} p(x) \log \frac{1}{p(x)} \tag{1}$$

Here p represents the function of probability mass of the random variable X. Shannon entropy measures the degree of uncertainty of a random variable by scoring less likely outcomes higher than the more likely ones. To define a Joint Probability distribution for both images to we have to generate a 2D histogram where each axis represents the number of possible gray scale values in each image and each histogram cell is incremented each time occurs in the pair of images. If the images are perfectly aligned then the histogram is highly focused if it is not than the dispersion grows.

1.2 Mutual Information
Mutual Information is the amount that the uncertainty in N(or M) is minimized when M (or N) is known and is defined as [9-16]

$$I(M;N) = H(N) - H(M|N) = H(M) - H(M|N) \tag{2}$$

MI based image registration is very much important as it consists individual entropy and works superior than simply joint entropy in regions of image background (low contrast) where there will be stumpy joint entropy but this is offset by low individual entropies as well so the overall mutual information will be low.

To make best use of the mutual information we have to minimize the joint entropy so the equation is

$$I(M;N) = H(M) + H(N) - H(M,N) \tag{3}$$

One more way of representing MI is by the following equation

$$I(M,N) = \sum_{a,b} p(m,n) \cdot \log\left(\frac{p(m,n)}{p(m)\,p(n)}\right) \tag{4}$$

This definition is correlated to the Kullback-Leibler distance between two distributions. It measures the dependence of the two distributions. In image registration I (M,N) will be maximized when the images are aligned.

It is having some properties that are given below

a. MI is symmetric: I(M;N) = I(N,M)

b. I(M,M) = H(M)

c. I(M,N) <= H(M), I(M,N) <= H(N) : info every image contains concerning the other cannot be larger than the info they themselves include.

d. I(M,N) >= 0 : Cannot amplify uncertainty in M by knowing N

e. If M,N are independent then I(M,N) = 0

f. If M,N are Gaussian then:

$$I(M,N) = -\tfrac{1}{2}\log(1-\rho^2) \quad (5)$$

Following figures describes the basic steps of Image Registration and MI based Image Registration describe by [11].

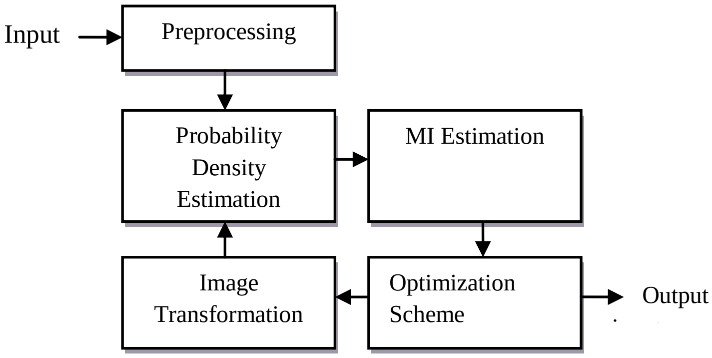


Figure 1: Basic Steps of MI Based Image Registration [11]

Above figure describes about the basic steps of MI based Image Registration we have parallelized this method as per suitability in our research to gain best result.

There are several techniques developed using parallel computing [17-21]for fastest Image Registration has been developed. Some of them are "A Parallel GPU Algorithm for Mutual Information based 3D Non rigid Image Registration" [22], "A Parallel Mutual Information Based Image Registration Algorithm for Applications in Remote Sensing" [23], "GPU Accelerated Medical Image Registration Techniques"[24], "Acceleration of Genetic Algorithm with Parallel Processing with Application in Medical Image Registration" [25], "Parallel Image Registration Using Bio-Inspired Computing"[26], "Multimodal Image Registration Using GPU Parallel Computing Technology"[27], "Fast Parallel Image Registration On CPU And GPU For Diagnostic Classification Of Alzheimer's Disease" [28], "Histogram-Based Image Registration Using Parallel Computing"[29]. There are some limitations of these algorithms like some of them approaches need GPU and good knowledge of CUDA programming models. Some of them need to be implemented on Medical Dicom images some approaches need to be tested on variety of images and so on. One of our papers based on review of existing image registration techniques using parallel computing has been accepted for publications and is in press[30]. There is the comparison of different techniques, limitations and benefits. This work is basically implemented to overcome the limitations of existing technique.

For this we organize this research paper into seven sections. Next section II will provide an overview of the concepts of parallel computing, parallel programming models and environments available for parallel computing. Section III describes about the proposed method for MI based image registration using parallel computing. Evaluation of experiment with different image data sets and results obtained is discussed section IV. Section V contains the discussion of this method and results obtained. Applications in medical imaging is described in section VI followed by conclusion in Section VII.

Now we are going to discuss briefly about the basic concepts of parallel computing and the parallel programming models which we have implemented in our work.

2. PARALLEL COMPUTING, ENVIRONMENT AND PROGRAMMING MODELS AVAILABLE FOR PARALLEL COMPUTING

It is a variety of computation in which various calculations are carried out concomitantly as it is having the principle that a large problem or a multifarious problem can be separated into smaller one. There can be numerous kind of parallel machine like a cluster of computers that contains multiple PCs combined together with a elevated speed network, a shared memory multiprocessor by connecting multiple processors to a single memory system, a Chip Multi-Processor (CMP) contains multiple processors/cores on a solitary chip [31 - 39]. Major fundamentals of Parallel computing are given below. To find maximum parallelism there is

Amdahl's Law: It states that if s be the fraction of work done sequentially, so (1-s) is fraction parallelizable [40]

P = number of processors

Speedup (P) = Time (1)/Time(P)

= 1/(s + (1-s)/P)

= 1/s

There are some keys of parallelism which we have considered at the time of parallelism of this research.

Granularity: It is defined as the numeral of basic units. Sometime it is coarse grained (Few Tasks of more powerful computing) and some time it is Fine Grain (Large Number of Small parts and less powerful computing)[41].

Implicit and Explicit Parallelism: We can defined parallel processing as two type First is Explicit in which algorithms includes instructions to specify which processes are built and executed in parallel way and Second is Implicit in which compiler has the task of inserting the necessary instructions to run the program on a parallel computer[41].

Synchronization is also necessary for making an algorithm from sequential to parallel as it prevents from the overlapping of two or more processors [41].

Latency: it is defined as the time conversion of information from request to accept [41].

Scalability: If we talk about scalability than It is defined as the capability of an algorithm to preserve its effectiveness by escalating the number of processors and the size of the problem in the same percentage [42]. Initially processors were developed with using only one core. As per the requirement multicore processors were developed later. Multi core processors may contain 2 cores as present in dual core Central processing units, for example Intel Core Duo and AMD Phenom II X2 , four cores are present as in quad core CPUs, e.g. Intel i5 and i7 processors and AMD Phenom II X4, six cores in hexa core CPUs, e.g. AMD Phenom II X6 and Intel Core i7 Extreme version 980X , eight cores are there in octa core processors, like Intel Xeon E7 2820 and AMD FX 8350 and ten cores e.g. Intel Xeon E7 2850, or more[43].

Performance of the multi core processors is basically dependent on the designed algorithm and their implementation. As per Amdahl's law possible gains are restricted by the portion of the software which can be executed in parallel concurrently on various cores. Basic architecture of dual core system is given below

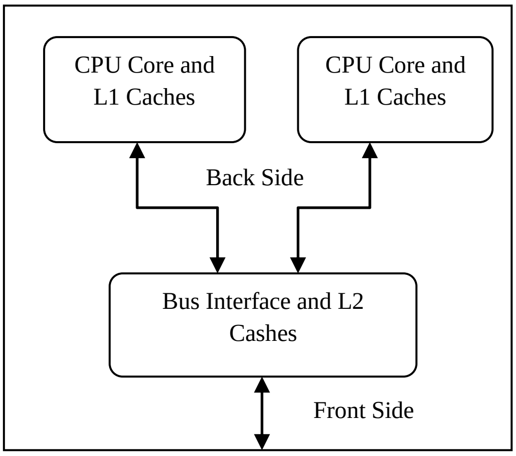


Figure 2: General Dual Core Structure [44]

For developing environment of parallel computing there are several tools available as HPF, Message Passing, Active Messages, Fortran, Parallel C, Parallel C++ , Data Structure Libraries (PETSc 2.0 for MPI, Multipol, LPARX), Numeric Libraries (LAPACK) and Many More. MATLAB plays a very vital role for developing parallel programming model it is extensively used for prototyping an algorithm. It is having parallel computing tool box which can be effectively used in many applications. Teng-Yi Huang*, Yu-Wei Tang and Shiun-Ying Ju uses MATLAB through C for parallel computation [45]. NVIDIA finds near about 15X speed up using MATLAB [46]. Simi V.R. Justin Joseph and Praveer Sihota analyzed performance measurement by MATLAB[47]. So we can see that there are so many authors developed their tool by using MATLAB.

Java is also very important tool for multithreading with the following reason we have used this as the major tool with the following reason. Java uses better resources utilization. Designing of the programs are very simple. Performance on multiple processors is very good. Programs are very responsive in this case.

Apart from this there are many disadvantage of multithreading in java but by synchronization it with MATLAB we can overcome all the drawbacks of this as we have used this in our research.

3. PROPOSED METHOD: IMAGE REGISTRATION USING MUTUAL INFORMATION IN PARALLEL COMPUTING ENVIRONMENT

Followings are the description of our proposed method for MI based parallel image registration. Here we are going to discuss developed framework, Computation of Mutual Information on Individual core, Joint Histogram Calculation.

Framework for Parallel Image Registration Using Mutual Information (MI) Method in Multicore Architecture

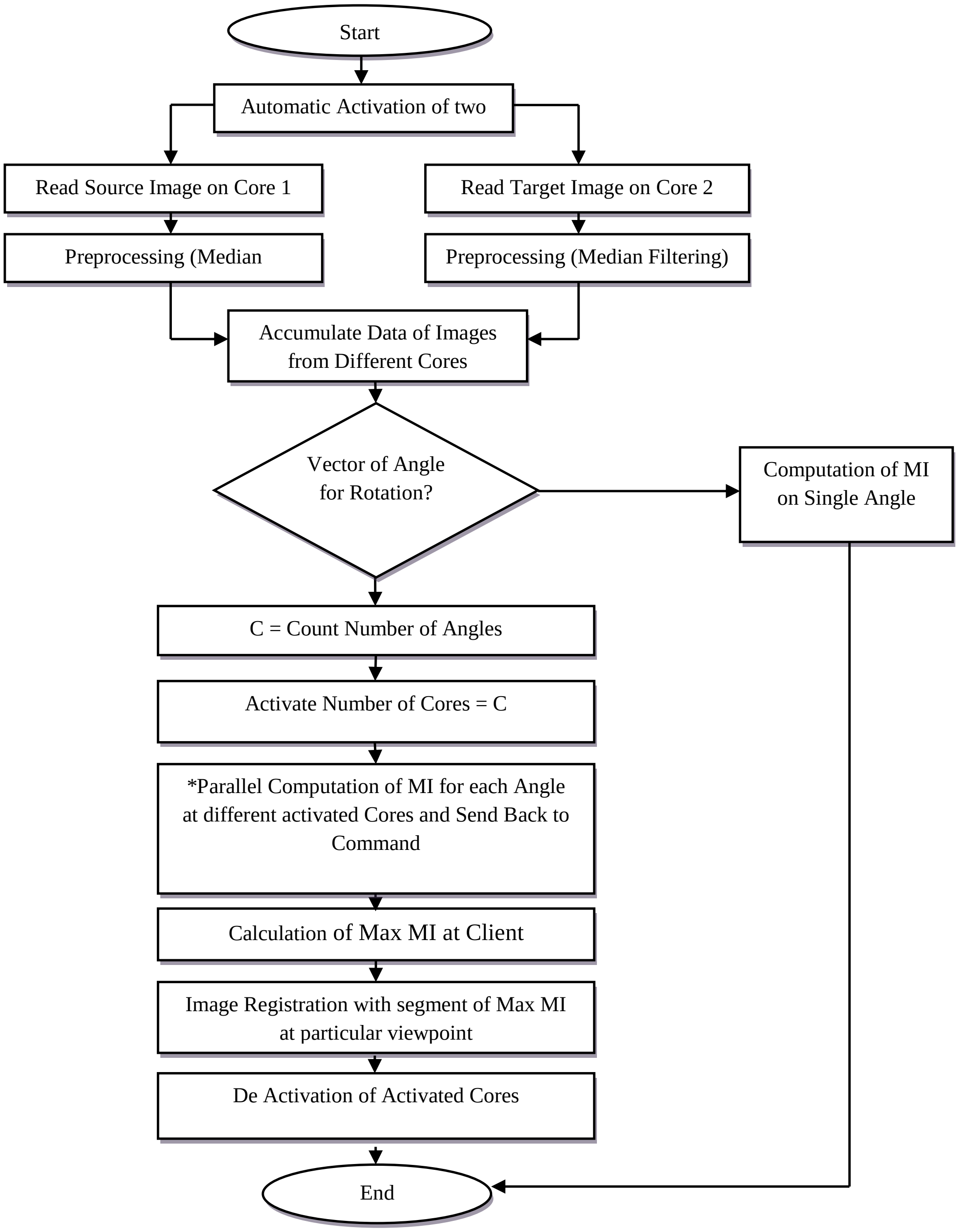


Figure 3: Framework of the Proposed Method

*** Steps for Parallel Computation of MI (Mutual Information)**

STEP 1: Reading of two images on different cores.

STEP 2: Computation of Joint Histogram

STEP 3: Normalization of the joint histogram

STEP 4: Find Sum of the rows and columns of joint histogram

STEP 5: Partition the array of rows and columns into the number of logical cores available.

STEP 6: Distribute the part of the images on the logical cores available.

STEP 7: Calculate marginal entropies of the individual part of image by maintaining synchronization between activated cores.

STEP 8: Accumulate the data of marginal entropy from activated individual logical cores.

STEP 9: Calculation of the joint entropy and Mutual Information (MI) on client.

<u>Steps for Calculation of Joint Probability Distribution on logical core</u>

STEP 1: Start

STEP 2: Calculate R = No of rows of Image1, C = No of Columns of Image2

STEP 3: N = R.

STEP 4: Zero Matrix H of Size N X N.

STEP 5: For I = 1: R

STEP 6: For J = 1: C

STEP 7: H(Image1(I,J) + 1, Image2(I,J) + 1) = H(Image1(I,J) + 1, Image2(I,J) + 1) + 1.

STEP 8: End

STEP 9: End

STEP 10: End

4. EXPERIMENTAL SETUP AND RESULTS OBTAINED

This Section describes about the Experimental Setup developed to implement the proposed intelligent method. We have implemented 60 Experiments with the following experimental setup, plot of the result obtained, registration of images and working of core i7 processor.

Setup for Experiments:
Hardware Configuration:
Processor: Intel(R) Core(TM) i7 -3770 CPU @340 GHz

RAM (Random Access Memory) : 4 GB

Hard Disk Drive: 320 GB

Software Environment:
System Type: 64 Bit operating System, x – 64- based processor.

Development Tools: MATLAB, Intel Compiler 9.1, jdk – 6 – windows – i586

Parallelization Scheme in MATLAB [20]
Multithreaded Parallelism and Explicit Parallelism

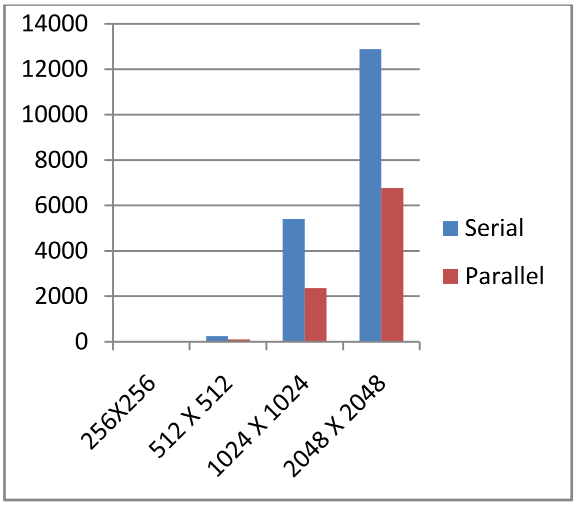
Following Table illustrates the Comparative study of execution time of sequential and parallel implementation. We have mark out the speedup with significant result

4.1 Image Set: Twenty Gray Scale Images (Different Size – 256 X 256, 512 X 512, 1024 X 1024, 2048 X 2048)

Table 1: Comparison of Sequential and Parallel Time Execution and Speed UP for Gray Scale Image

Images	Size	Sequentially Execution with Implicit Multithreading	Parallely Execution with Explicit Multithreading	Speedup (%)
Image 1	256 X 256	12.1	6.7	180.59
Image 1	512 X 512	207.8	132.6	156.71
Image 1	**1024 X 1024**	**5402.6**	**2360.2**	**228.90**
Image 1	2048 X 2048	12436.2	7184.6	173.09
Image 2	256 X 256	11.8	6.9	171.01
Image 2	512 X 512	192.1	100.6	190.95
Image 2	1024 X 1024	5112.4	2728.4	183.37
Image 2	2048 X 2048	11233.5	6225.1	180.45
Image 3	256 X 256	13.23	6.92	191.18
Image 3	**512 X 512**	**235.1**	**111.3**	**211.23**
Image 3	1024 X 1024	5638.7	2911	193.70
Image 3	2048 X 2048	12963.1	6829.9	189.80
Image 4	256 X 256	12.6	7.2	175
Image 4	512 X 512	214.4	109.6	195.62
Image 4	1024 X 1024	5723.8	2899.8	197.38
Image 4	2048 X 2048	13000.1	6914.5	188.01
Image 5	**256 X 256**	**12.26**	**6.08**	**201.64**
Image 5	512 X 512	219.5	112.1	195.80
Image 5	1024 X 1024	6123.4	3498.3	175.03
Image 5	**2048 X 2048**	**12887.7**	**6772.2**	**190.30**

Now we are going to show the plot of best time comparison and maximum speed up from different images and speed up plot for this



s

Figure 4: Plot of Comparison of Time

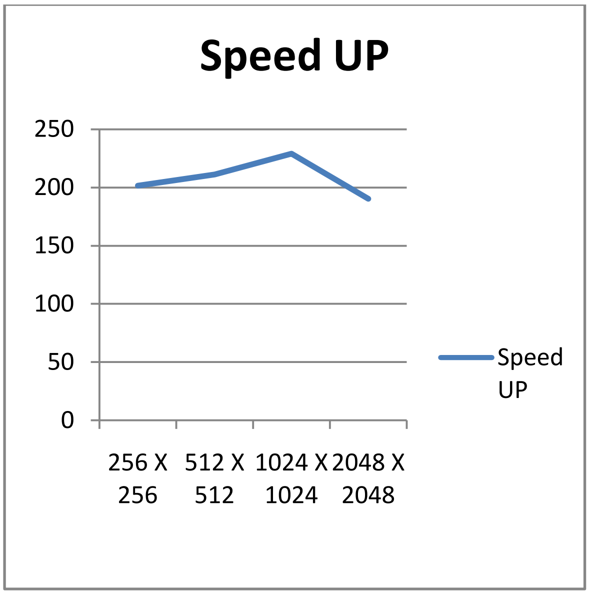


Figure 5: Speed UP Plot

Model Image Source Image Registered Image

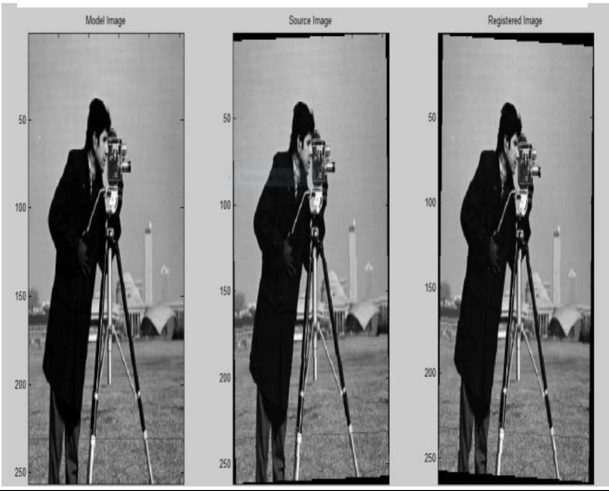


Figure 6: Average Result of Image Registration

4.2 Image Set: Twenty RGB Images (Different Size – 256 X 256, 512 X 512, 1024 X 1024, 2048 X 20148)

Table 2: Comparison of sequential and parallel execution of MI based IR for RGB images

Images	Size	Sequentially Execution with Implicit Multithreading	Parallely Execution with Explicit Multithreading	Speedup (%)
Image 1	256 X 256	14.5	7.8	185.89
Image 1	512 X 512	235.2	120.62	194.99
Image 1	1024 X 1024	5922.4	3370.8	175.69
Image 1	2048 X 2048	13126.1	7344.5	178.72
Image 2	256 X 256	13.7	7.8	175.64
Image 2	512 X 512	200.4	109.7	182.68
Image 2	1024 X 1024	5721.3	2805.3	203.94
Image 2	2048 X 2048	12248.8	6315.4	193.95
Image 3	256 X 256	14.9	8.13	183.27
Image 3	512 X 512	291.4	150.2	194.00
Image 3	**1024 X 1024**	**6123.8**	**2889.9**	**211.90**
Image 3	2048 X 2048	13912.4	7124.3	195.28
Image 4	**256 X 256**	**13.5**	**6.8**	**198.529**
Image 4	512 X 512	301.5	182.43	165.27
Image 4	1024 X 1024	6219.6	3191.4	194.88
Image 4	2048 X 2048	13978.6	7426.2	188.23
Image 5	256 X 256	12.91	8.12	158.99
Image 5	**512 X 512**	**339.4**	**169.9**	**199.76**
Image 5	1024 X 1024	6871.5	3517.5	195.35
Image 5	**2048 X 2048**	**13456.01**	**6571.6**	**204.76**

Execution of MI based IR for RGB Images

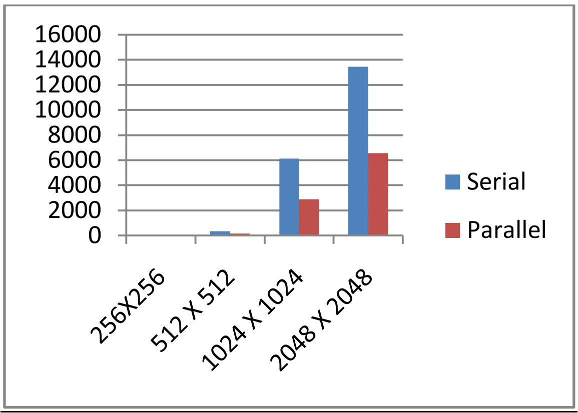


Figure 7 : Plot of Comparison of Time

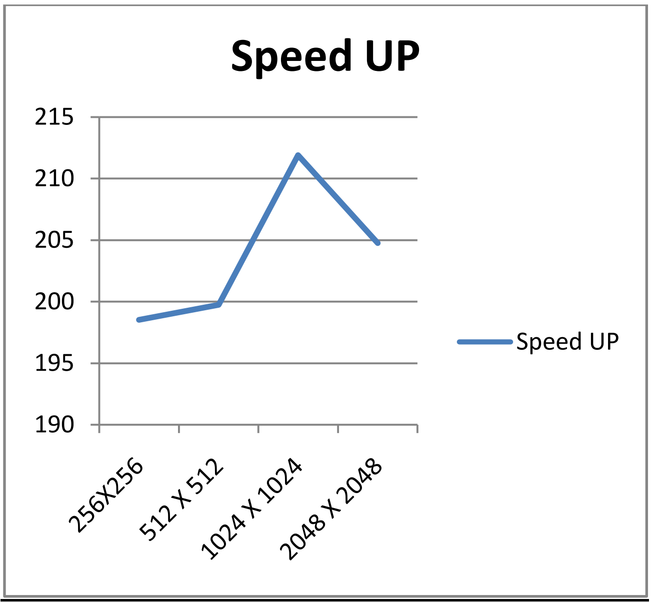
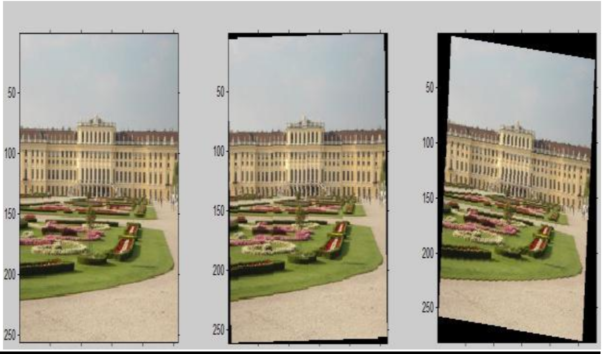


Figure 8: Speed UP Plot for RGB Images

Model Image Source Image Registered Image



4.3 Image Set: Twenty different Brain Dicom Images of size 256 X 256

Table 3: Comparison of sequential and parallel execution of MI based IR for Dicom images

Images	Size	Sequentially Execution with Implicit Multithreading	Parallely Execution with Explicit Multithreading	Speedup (%)
Image 1	256 X 256	13.55	5.01	270.45
Image 2	256 X 256	13.98	4.98	280.72
Image 3	256 X 256	12.60	5.22	241.37
Image 4	256 X 256	14.2	5.17	274.66
Image 5	256 X 256	13.56	5.32	254.88
Image 6	256 X 256	14.02	4.77	293.92
Image 7	256 X 256	13.24	4.23	313.01
Image 8	256 X 256	14.16	5.19	272.83
Image 9	256 X 256	13.43	4.76	282.14
Image 10	256 X 256	12.80	4.38	292.23
Image 11	256 X 256	14.03	5.98	234.61
Image 12	256 X 256	14.98	6.01	249.25
Image 13	256 X 256	13.78	5.34	258.05
Image 14	256 X 256	13.93	4.68	297.64
Image 15	256 X 256	14.45	5.38	268.58
Image 16	256 X 256	15.01	6.10	246.06
Image 17	256 X 256	13.04	4.39	297.04
Image 18	256 X 256	13.77	5.49	250.82
Image 19	256 X 256	14.11	5.12	275.59
Image 20	256 X 256	13.20	4.80	275

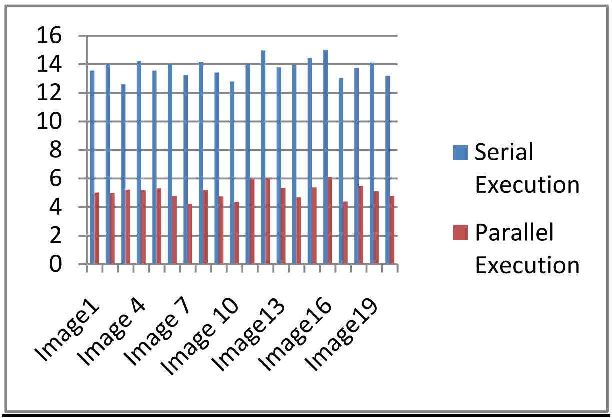


Figure 10 : Plot of Comparison of Time

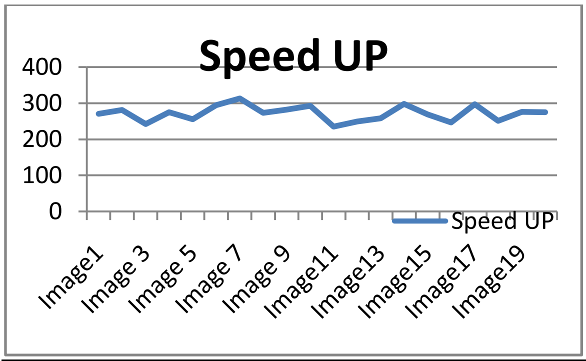


Figure 11: Speed UP Plot for Dicom Images

Model Image Source Image Registered Image

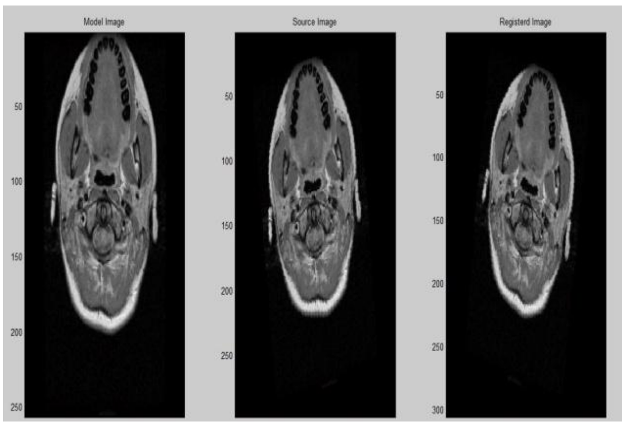


Figure 12 : Average Result of Image Registration

Following figure represents copious consumption of the cores in i7 processor

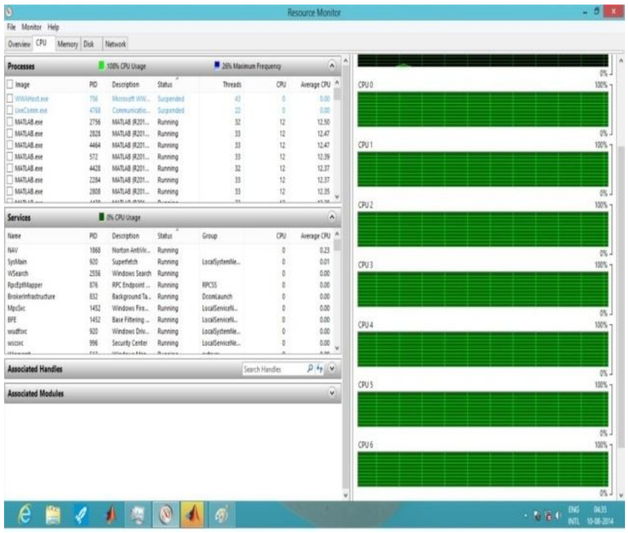


Figure 13: Shows all activated logical core and also shows fully utilization of the resources available

5. DISCUSSION

We can glimpse that in the above table that our approach is drastically able to work with diverse images such as Gray Scale, RGB and Dicom images and got very excellent results while implementing MI based Image registration parallelly rather than implementing sequentially. Almost all experiments are having twice of speed up. We parallelize the most time consuming steps of MI based image registration and assigned considerable tasks such as probability density estimation and MI calculation at different angles on individual cores and got maximum match. Multithreading is done in this way that synchronization is maintained at each assigned core and others are available to execute other tasks. Main advantage of this work is that this implementation is carried out in the processors having multiple cores with almost whole consumption of cores and got significant result. To evaluate the registration quality we focused on accuracy and efficiency. Accuracy defined as ratio between correct and all registrations and Efficiency is defined by the mean number of function evaluations for a correct registration these parameters are shown in above section having significant results. One thing is more several existing techniques as discussed above need GPU and good knowledge of CUDA programming model. This effort gives significant outcomes without using GPU and CUDA programming models.

6. APPLICATIONS IN MEDICAL IMAGING

There are several applications of Medical Image Registration as it increases the information content after register the medical images of different modalities like CT PET and CT MR registration and many more. There are lots of research based on the applications of parallel implementation of registration of neuro imaging as statistical group comparisons, tissues segmentation, mathematical morphological operations and many more discussion of medical imaging in [48 - 55]. This fastest result can be very useful for the radiologist for the treatment planning.

7. CONCLUSION

This paper explores the implementation of MI based image registration using parallel computing. Here an explicit multithreaded approach was developed for MI based image registration for prompt and proficient result. This research work can be easily implemented on multi core processor which are easily available in the market and produce tremendous result. Image Registration based on Mutual Information is time consuming processes as it is having several task to be executed. In this paper, we have assigned individual task to each activated cores after maintaining the synchronization between them. After execution of the particular task assigned core/processor is free to execute another existing task in queue. Main advantage of this work is that we have parallelized all the time consuming steps of MI based image registration as a result we can see that the performance in terms of speed up of the developed approach is extensively excellent. Another major advantage of this approach is that this is able to work on multi core processor having no GPUs. It is tested on the variety of Gray Scale, RGB and Dicom images as discussed above and able to give result in minimum execution time compared with existing algorithm of parallel image registration. For the future work's point of view we can test this work on cluster of computers and the computer having GPU. Apart from this we can add the concept of memory sharing between cores/processors/computers in this work.

8. REFERENCES

[1] Derek L G Hill, Philipp G Batchelor, Mark Holden and David J Hawkes, "Medical image registration", PHYSICS IN MEDICINE AND BIOLOGY, 2001, , R1–R45.

[2] Moriyoshi Ohara, Hangu Yeo, Frank Savino, Giridharan Iyengar, Leiguang Gong, Hiroshi Inoue, Hideaki Komatsu, Vadim Sheinin, Shahrokh Daijavad, and Bradley Erickson, "Real-time mutual-information-based linear registration on the cell Broadband engine processor" in IEEE, ISBI 2007.

[3] Sanjay Saxena, Neeraj Sharma and Shiru Sharma, "Image Processing Tasks using Parallel Computing in Multi core Architecture and its Applications in Medical Imaging" International Journal of Advanced Research in Computer and Communication Engineering, Vol. 2, Issue 4, April 2013.

[4] T. W. Burger, "Intel Multi-Core Processors: Quick Reference Guide", http://cachewww.intel.com/cd/00/00/23/19/231912_231912.pdf

[5] M. C. Michael, "Programming models for scalable multicore programming", http://www.hpcwire.com/features/17902939.html, 2007

[6] L. Chai, Q. Gao, D. K. Panda, "Understanding the Impact of Multi-Core Architecture in Cluster Computing: A Case Study with Intel Dual-Core" .

[7] Accelerating non-linear image registration with GPUs Paul Ross Academic year 2010/2011

[8] C.E. Shannon, A mathematical theory of communication, Bell Syst. Tech . J. 27 1948, 379–423, 623–656.

[9] Josien P. W. Pluim, J. B. Antoine Maintz and Max A. Viergever "Mutual information based registration of medical images : A survey " in IEEE TRANSACTIONS ON MEDICAL IMAGING, VOL. XX, NO. Y, MONTH 2003.

[10] Elisabeth Hedlund, Jan-Erik Karlsson, Sven-Åke Starck, "Automatic and manual image fusion of In-pentetreotide SPECT and diagnostic CT in neuroendocrine tumor

imaging - An evaluation" in Journal of Medical Physics, Vol. 35. No 4 , 2014, pp. 223 – 228.

[11] "Mutual Information for Image Registration and Feature Selection" by M. Farmer.

[12] Sanjay Saxena , Shiru Sharma and Neeraj Sharma "Image Registration Techniques using Parallel Computing in Multicore Environment and Its Applications in Medical Imaging : An Overview" accepted in ICCCT, IEEE Xplore database(in press), 2014.

[13] Sanjay Saxena , Shiru Sharma and Neeraj Sharma "Parallel Computation of Mutual Information for Image Registration in Multicore Environment & Its Applications in Medical Image Registration" accepted in MEDCOM 2014(in press), 2014.

[14] Collignon, F. Maes, D. Delaere, D. Vandermeulen, P. Suetens, and G. Marchal. "Automated multi-modality image registration based on information theory Information Process" Med. Imaging, 1995 , pages 263–274.

[15] F. Maes, A. Collignon, D. Vandermeulen, G. Marchal, and P. Suetens , "Multimodality image registration by maximization of mutual information". IEEE Trans. Med. Imaging.,1997, 16(2):187–198.

[16] W. M. WellIII, P. Viola, H. Atsumi , and R. Kikinis, "Multi-modal volume registration by maximization of mutual information" in Medical Image Analysis., 1(1): 1996, 35–51.

[17] C.R. Castro-Pareja, J.M. Jagadeesh, and R. Shekhar, "FAIR: A Hardware Architecture for Real-Time 3-D Image Registration," IEEE Transactions on Information Technology in Biomedicine, vol.7, no.4, Dec. 2003, pp. 426-434.

[18] L. Jianchun, R. Shekhar, and C. Papachristou, "A "brick" caching scheme for 3D medical imaging," IEEE International Symposium on Biomedical Imaging: Macro to Nano vol. 1, Apr. 2004, pp. 563-566.

[19] T. Rohlfing, C.R. Maurer, "Nonrigid image registration in shared-memory multiprocessor environments with application to brains, breasts, and bees ," IEEE Transactions on Information Technology in Biomedicine , vol.7, no.1,Mar. 2003, pp. 16- 25.

[20] S. K.Warfield, F. Jolesz, and R. Kikinis, "A high performance approach to the registration of medical imaging data ," Parallel Computing, vol. 24, no. 9–10, 1998 pp. 1345–1368.

[21] Jian Shen,Yurong Chen, He Li, Yimin Zhang, Yinlong Xu, "Accelerating Mutual-Information-Based Registration om Multi-Core Systems" in IEEE 2010.

[22] Vaibhav Saxena, Jonathan Rohrer, Leiguang Gong, "A Parallel GPU Algorithm for Mutual Information Based 3D Nonrigid Image Registration" in Lecture Notes in Computer Science Volume 6272,2010, pp 223-234.

[23] Y unfei Du, Haifang Zhou, Panfeng Wang, Xuejun Yang, Hengzhu Liu , "A Parallel Mutual Information Based Image Registration Algorithm for Applications in Remote Sensing" in Parallel and Distributed Processing and Applications Volume 4330, 2006, pp 464-473.

[24] S.Mahalakshmi, "GPU Accelerated Medical Image Registration Techniques International Journal of Innovative Research in Science & Engineering".

[25] B. Laksanapanai, W. Withayachumnankul , C. Pintavirooj and P.Tosranon , "Acceleration of Genetic Algorithm with Parallel Processing with Application in Medical Image Registration"in Conference proceedings ISBN 80-903100-9-5 WSCG'2005.

[26] Silviu-Ioan Bejinariu, Florin Rotaru, Cristina Nita, Ramona Luca, Hariton Costin, "Parallel Image Registration using Bio-inspired Computing" in 4th IEEE International Conference on E-Health and Bioengineering - EHB 2013

[27] Xiaolu Sun, "Multimodal Image Registration Using GPU Parallel Computing Technology" Research Project in Computational mathematics, Waterloo, Ontario, Canada, 2010.

[28] Denis P.Shamonin, EstherE.Bron, BoudewijnP.F.Lelieveldt, MarionSmits, StefanKlein and MariusStaring,"Fast parallel image registration on CPU and GPU for diagnostic classification of Alzheimer's disease", Frontiers in Neuroinformatics Volume 7, January 2014

[29] Benjamin Guthier, Stephan Kopf, Matthias Wichtlhuber, Wolfgang Effelsberg, "Parallel algorithms for histogram-based image registration", IWSSIP , Vienna, Austria, 2012.

[30] Sanjay Saxena , Shiru Sharma and Neeraj Sharma "Image Registration Techniques using Parallel Computing in Multicore Environment and Its Applications in Medical Imaging : An Overview" Accepted in ICCCT, IEEE Xplore database (in press), 2014.

[31] Help, MATLAB Software.

[32] Sanjay Saxena, Neeraj Sharma, Shiru Sharma , "An intelligent System for Segmenting an Abdominal Image in Multicore Architecture", IEEE Explore in CEWIT, New York , USA, 2013.

[33] J. Fung and S. Mann, "Using graphics devices in reverse: GPU – based image processing and computer vision", in IEEE International Conference on Multimedia and Expo. IEEE, June 2008, pp. 9-12.

[34] Edelman, P. Husbands, S. Leibman, "Interactive Supercomputing" s Star – P Platform: Parallel MATLAB and MPI Homework Classroom Study on High Level Language Productivity," HPEC, 2006.

[35] Blaise Barney, Lawrence Livermore National Laboratory, "Introduction to Parallel Computing"(https://computing.llnl.gov/tutorials/parallel_comp/) August 10 , 2014.

[36] Slabaugh, Richard Boyes,om Xiaoyun Yang, "Multicore Image Processing with OpenMP".

[37] Teng-Yi Huang*, Yu-Wei Tang, Shiun-Ying Ju , "Accelerating image registration of MRI by GPU-based parallel computation" in Magnetic Resonance Imaging 29 , 2011, 712–716.

[38] Concepts of Parallel Computing by NVIDIA

[39] Simi V. R., Justin Joseph and Praveer Sihota, “An Analysis of the performance of Brightness Preserving Bi-Histogram Equalization on Medical Images”.

[40] Mark D. Hill and Michael R. Marty , “Amdahl’s Law in the Multicore Era” IBM, 2007.

[41] Sanjay Saxena, Neeraj Sharma and Shiru Sharma, “Image Processing Tasks using Parallel Computing in Multi core Architecture and its Applications in Medical Imaging” International Journal of Advanced Research in Computer and Communication Engineering, Vol. 2, Issue 4, April 2013.

[42] R. T. Rasúa, “Algoritmos paralelos para la solución deproblemas de optimización discretos aplicados a ladecodificación de señales,” Ph.D. dissertation, Departamento de Sistemas Informáticos y Computación. Universidad Politécnica de Valencia, Valencia, España, 2009.

[43] http://en.wikipedia.org/wiki/Multi-core_processor(August 17, 2014)

[44] http://en.wikipedia.org/wiki/Multi core_processor#mediaviewer/File:Dual_Core_Generic.svg(August 18, 2014)

[45] Teng-Yi Huang*, Yu-Wei Tang, Shiun-Ying Ju , “Accelerating image registration of MRI by GPU-based parallel computation” in Magnetic Resonance Imaging 29 , 2011, 712–716.

[46] YasserM. Kadah, Khaled Z. Abd-Elmoniem, and Aly A. Farag , ”Parallel Computation inMedical Imaging Applications” in International Journal of Biomedical Imaging Volume 2011.

[47] Friston, K., Holmes, A., Worsley, K., Poline, J.-P., Frith, C., and Frackowiak, R. “Statistical parametric maps in functional imaging: a general linear approach” Hum. Brain Mapp. 2, 1994, 189–210.

[48] Ashburner, J., and Friston, K. “Voxel-based morphometry–the methods”. Neuroimage 11, 805–821, 2000.

[49] Fischl, B., Salat, D., Busa, E., Albert, M., Dieterich, M., Haselgrove, C., et al. “Whole brain segmentation: automated labeling of neuroanatomical structures in the human brain” Neuron 33, 2002,341–355.

[50] Ashburner, J., and Friston, K. “Unified segmentation”, Neuroimage 26, 2005, 839–851.

[51] Klöoppel, S., Stonnington, C., Chu, C., Draganski, B., Scahill, R., Rohrer, J., et al. “Automatic classification of MR scans in Alzheimer's disease” Brain 131,2008, 681–689.

[52] Magnin, B., Mesrob, L., Kinkingnéhun, S., Pélégrini-Issac, M., Colliot, O., Sarazin, M., et al.“Support vector machine-based classification of Alzheimer's disease from whole-brain anatomical MRI”, Neuroradiology 51, 2009, 73–83.

[53] Cuingnet, R., Gerardin, E., Tessieras, J., Auzias, G., Lehéricy, S., Habert, M., et al. “Automatic classification of patients with Alzheimer's disease from structural MRI: a comparison of ten methods using the ADNI database.” Neuroimage 56, 2011, 766–781.

[54] Sanjay Saxena, Neeraj Sharma and Shiru Sharma, “Region wise processing of an image using multithreading in multicore environment & its application in Medical Imaging” in IJCET Volume:4,Issue:4, 2013, Pages:20-30.

[55] Josien P. W. Pluim, J. B. Antoine Maintz and Max A. Viergever “Mutual information based registration of medical images : A survey ” in IEEE TRANSACTIONS ON MEDICAL IMAGING, VOL. XX, NO. Y, MONTH 2003