ACKNOWLEDGEMENT

It is my foremost responsibility to express my profound gratitude and sincere respect to my Research Guide **Dr.S.Subha** for her continuous support throughout my work. Her talents, dedication and enthusiasm for research will be a continuing source of inspiration

I wish to express my heartfelt gratitude to **Dr.G.Viswanathan**, Chancellor, VIT , Vellore, for providing facilities for the research work.

I am highly grateful to our Vice Presidents, **Shri. Sankar Viswanathan, Dr. Sekar Viswanathan and Shri. G.V.Selvam**, Vice Chancellor **Dr. Anand A. Samuel**, Pro-Vice Chancellors **Dr. S. Narayanan** and **Dr. V. Raju** for providing necessary resources.

I wish to express my heart-felt gratitude to the Doctoral Committee members, Dr.Dhavachelvan **& Dr.S.Venkatesan.**

I wish to place on record my gratitude to **Dr.Buvaneswari,** Dean (Academic Research), VIT for her kind words of support and encouragement during my research work.

My sincere gratitude to **Dr. Aswani Kumar Cherukuri**, Dean, School of Information Technology and Engineering for giving me the opportunity to undertake the research work.

I thank the management of VIT for permitting me to use the library resources. I also thank all the faculty members of VIT for giving me the courage and strength I needed to complete my goals. This acknowledgement would be incomplete without expressing my whole hearted thanks to my family and friends who motivated me during the course of the work.

Finally, and most importantly, I would like to thank the Almighty God, for it is under his grace that I live, learn and flourish

Place : Vellore

Date : **MANIKANDAN N**

CONTENTS

ABSTRACT

In the evolution of modern processors, multi core based CPUs play a vital role. The current world of computing is getting more and more powered by latest processors. Multimedia and Big Data based applications frequently need series of calculations and features for supporting parallelism. With the help of SIMD functions and instruction set, the research can enhance the speed of the multi core based processing. Tools and techniques used for exploiting SIMD parallelism will have high impact results. Now commercial compilers are coming with automatic vectorization of programs but code performance of such compilers are not common for all type of applications. Compiler level vectorization has some difficulties like pointer reference, data and control dependency. These problems can be addressed only by parallel programmers. There are some parallel programming tools like OpenMP, OpenCL, CUDA, MATLAB and thread based programming which supports vectorization. This thesis has developed a detailed analysis and implementation on vectorization methods for multi core processing. This thesis includes dependency analysis methods, choosing correct parallel algorithm, parallel execution tools and manipulation of data. Implementations focus on remodelling the existing applications and algorithms that act as the performance booster with the support of code analyser, parallel algorithm and tool analyser. To test the performance of methodologies, computer intensive applications like matrix computing, big data processing, data compression and encryption algorithms are used. The results are obtained with Intel based multi core processors. The same work is implemented and compared with multicore processors and graphics processing unit combination. As a result of vectorising different applications with varying scale, minimum of 3X and maximum of 140X performance improvement has been achieved.

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

AES	Advanced Encryption Standard
ALU	Arithmetic and Logical Unit
ANN	Artificial Neural Network
CCUMA	Cache Coherent UMA
CDB	Common Data Bus
CPU	Central Processing Unit
CUDA	Compute Unified Device
DCT	Discrete Cosine Transform
DES	Data Encryption Standard
DSP	Digital Signal Processing
EBPN	Error Back Propagation Network
GPGPU	General Purpose Graphics Processing Units
GPU	Graphics Processing Units
ICU	Intensive Care Unit
ILP	Instruction Level Parallelism
IoT	Internet of Things
IPC	Instructions per cycle
MEDUSA	Medical Distributed Utilization of Services and Applications
MIMD	Multiple Instruction stream and Multiple Data stream
MISD	Multiple Instruction stream and Single Data stream
NUMA	Non Uniform Memory Access
OLAP	Online Analytical Processing
PANN	Parallel Artificial Neural Networks
PRAM	Programmable Random Access Memory
SIMD	Single Instruction stream and Multiple Data stream
SIMT	Single Instruction Multiple Threads
SISD	Single Instruction stream and Single Data stream
SMP	Symmetric Multiprocessor
UMA	Uniform Memory Access
IOT	Internet of Things