

PROSIDING SEMINAR KEBANGSAAN SAINS, TEKNOLOGI & SAINS SOSIAL

27 ~ 28 MEI 2002

HOTEL VISTANA, KUANTAN, PAHANG

Anjuran :



**Universiti Teknologi MARA
Cawangan Pahang**

Dengan Kerjasama



**Kerajaan
Negeri Pahang Darul Makmur**

JILID 1



IMPLIMENTASI 'TREE SEARCHED VECTOR QUANTIZER' UNTUK PEMAMPATAN IMEJ DENGAN MENGGUNAKAN KAEDAH FPGA

M B Abu Bakar, Z Mohd Yusof (UTM), S Sulaiman
Jabatan Sains Kejuruteraan, Fakulti Sains dan Teknologi,
Kolej Universiti Sains dan Teknologi (KUSTEM),
21030 Mengabang Telipot, Kuala Terengganu, Terengganu.
email : bakhtiar@kustem.edu.my
Tel : 609-6683332

Abstrak – *Tree Searched Vector Quantization* atau TSVQ merupakan salah satu kaedah didalam teknik *quantization* didalam sesuatu sistem pemampatan imej. Algoritma ini diterjemahkan kepada peringkat rekabentuk logik digital sebelum diaturcarakan kepada cip FPGA. Cip FPGA merupakan cip boleh aturcara berulang kali dalam masa yang singkat dan amat sesuai digunakan didalam bidang penyelidikan dan pembangunan (R&D) terutamanya di pusat pengajian tinggi ataupun syarikat berorientasikan kepada penyelidikan dan pembangunan. Di peringkat permulaan, teknologi ini dipelopori oleh dua syarikat utama iaitu Altera dan Xilinx. Teknologi ini mempercepatkan kitar masa untuk penyelidikan dan pembangunan sesuatu sistem aplikasi kepada bentuk litar bersepadu. Di dalam kertas kerja ini, algoritma TSVQ akan dibangunkan dan diaplikasikan kedalam cip Altera Flex 10K100 bagi proses pemampatan imej. Perantaraan papan litar FPGA yang telah direkabentuk dan antaramuka talian selari pada komputer peribadi akan digunakan untuk proses pengujian. Data masukan adalah diambil dari imej disimpan didalam ROM. Platform sistem dibangunkan dengan menggunakan perisian MaxplusII yang dibekalkan oleh syarikat Altera dan bagi pengujian akhir sistem ini menggunakan cip Altera Flex EPF-10k100ARC240-3 dengan kapasitinya 100,000 get logik.

Kata Utama : 'Tree Searched Vector Quantizer', Pemampatan Imej, FPGA

PENGENALAN

Pemampatan imej adalah salah satu kunci utama dalam bidang yang berkaitan dengan sistem multimedia dan dunia ICT. Pemampatan membolehkan penghantaran maklumat yang bersaiz besar dapat dihantar dengan cepat dan pantas. Contoh maklumat atau data yang bersaiz besar seperti imej/video akan dapat dihantar dengan menggunakan talian internet. Talian internet pada kebiasaannya mempunyai jalur lebar yang kecil. Pemampatan menyebabkan data yang memerlukan jalur lebar yang besar seperti imej atau isyarat video akan dapat menggunakan talian internet. Matlamat pemampatan imej adalah untuk mengurangkan kadar bit data bagi penghantaran melalui talian penghantaran dan mengurangkan saiz untuk mengecilkan ruang simpanan tetapi mengekalkan kualiti imej tersebut.

Piawaian pemampatan imej dan video yang sediaada termasuklah H.261, MPEG1, MPEG2, JPEG and H.263. Algoritma pemampatan boleh diimplimentasikan dengan beberapa cara; penyelesaian perisian, kaedah pemprosesan isyarat digit dan implimentasi perkakasan. Untuk pemprosesan masa sebetar bagi isyarat video (pemampatan dan nyahpemampatan), ia memerlukan ciri-ciri prestasi pemprosesan yang pantas hanya terdapat pada implimentasi perkakasan.

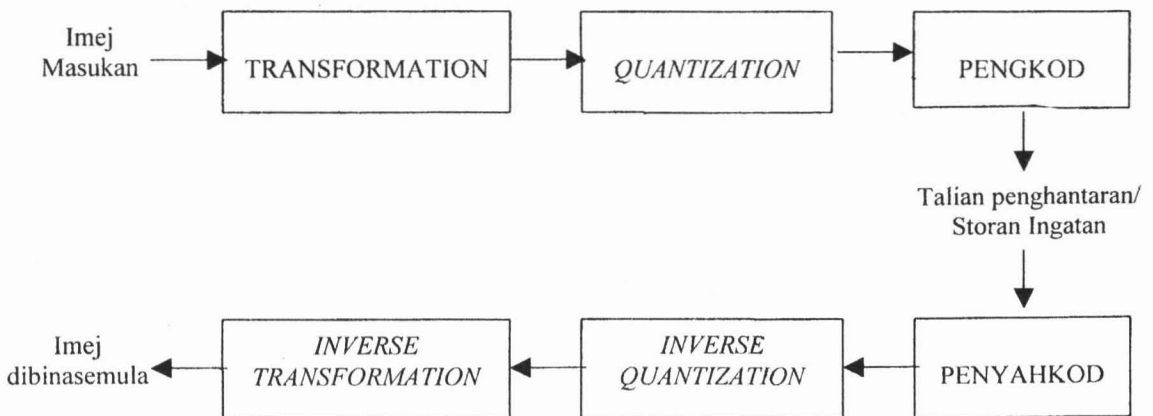
Implimentasi perkakasan boleh dicapai dengan menggunakan kaedah *Application Specific Integrated Circuit (ASIC)* dan FPGA. Implikasi menggunakan kaedah ASIC memerlukan kos yang tinggi, kitar masa pembangunannya yang lama dan mengambilkira kos pengeluaran. Ia tidak sesuai dalam persekitaran R&D. Untuk *prototyping*, implimentasi FPGA adalah amat sesuai sekali. Teknologi FPGA semakin maju dengan ketumpatan get dan prestasi kelajuan pemprosesan yang tinggi berbanding dengan ASIC. Tambahan pula teknologi FPGA berupaya untuk boleh aturcara dalam sistem pengatucaraan, kitar rekabentuk yang singkat serta kos pembangunan yang rendah. Atas motivasi inilah, penyelidikan dan pembangunan implimentasi perkakasan untuk algoritma pemampatan imej ini menggunakan kaedah FPGA.

PEMAMPATAN DATA

Masalah utama berkaitan dengan imej digital atau video digital sistem ialah memerlukan saiz jalur lebar yang besar untuk talian penghantaran atau ruang simpanan yang besar. Untuk mengatasi masalah ini, banyak algoritma pemampatan dibangunkan. Pemampatan boleh dikategorikan kepada 2 kumpulan iaitu Pemampatan Tak Kehilangan dan Pemampatan Kehilangan. Pemampatan Tak Kehilangan bermaksud selepas proses pemampatan berlaku, data yang seperti mana data sebelum pemampatan akan diperolehi. Pemampatan ini digunakan untuk pemampatan huruf dan fail perduaan komputer. Walaubagaimanapun bagi Pemampatan Kehilangan, sedikit maklumat mungkin akan hilang semasa proses pemampatan dan nyahpemampatan berlaku.

Kebanyakan pemampatan kehilangan berupaya menentukan kehilangan maklumat berlaku pada faktor pemampatan yang tinggi supaya ia tidak akan menjejaskan kualiti imej tersebut. Teknik pemampatan imej/video secara umumnya adalah pemampatan kehilangan dimana ia mengesplotasikan ciri kelebihan statistik dalam data dan sistem penglihatan manusia. Sistem penglihatan manusia kurang sensitif pada frekuensi maklumat tinggi. Sekiranya terdapat sebarang kehilangan pada frekuensi maklumat tinggi, manusia tidak dapat mengesan. Dengan membuang frekuensi maklumat tinggi, teknik pemampatan akan tercapai.

Sistem pemampatan imej terbahagi kepada 3 komponen utama iaitu *Transformation*, *Quantization* dan Pengkodan. *Transformation* adalah proses untuk penjelmaan isyarat atau imej dari domain ruang kepada domain frekuensi untuk memperolehi kesan yang maksima dalam proses pemampatan. Algoritma yang digunakan seperti Discrete Fourier Transform (DFT), Haar, Hadamard dan Discrete Cosine Transform (DCT).



Gambarajah 1 : Komponen Asas Sistem Pemampatan Gambar

Quantization adalah seperti proses pengelasan kepada titik berdekatan atau titik kumpulan. Proses ini akan kehilangan kumpulan frekuensi maklumat tinggi. Hasil data yang diperolehi tidak semestinya sama tetapi mestilah serupa. *Quantization* adalah pemampatan kehilangan dimana sebahagian maklumat data itu akan hilang. Terdapat 2 jenis *quantization* iaitu *scalar quantization* (SQ) dan *vector quantization* (VQ). Penyelesaian SQ adalah mudah dan ia lebih sesuai bagi penyelesaian yang asas. Bagi penyelesaian VQ, sekumpulan sample data akan di *quantized*. Berasaskan pada teori, VQ mempunyai prestasi yang lebih baik daripada SQ. Terdapat 2 kaedah dalam VQ iaitu *Full Search VQ* dan *Structured VQ*. Didalam *Structured VQ* terdapat beberapa teknik untuk mengurangkan kekompleksan VQ iaitu termasuk *Tree Search VQ*, *Multi-Stage VQ*, *Product VQ* dan *Lattice VQ*. Didalam kertas kerja ini, teknik *Tree Search VQ* akan di implimentasikan dalam sistem pemampatan imej.

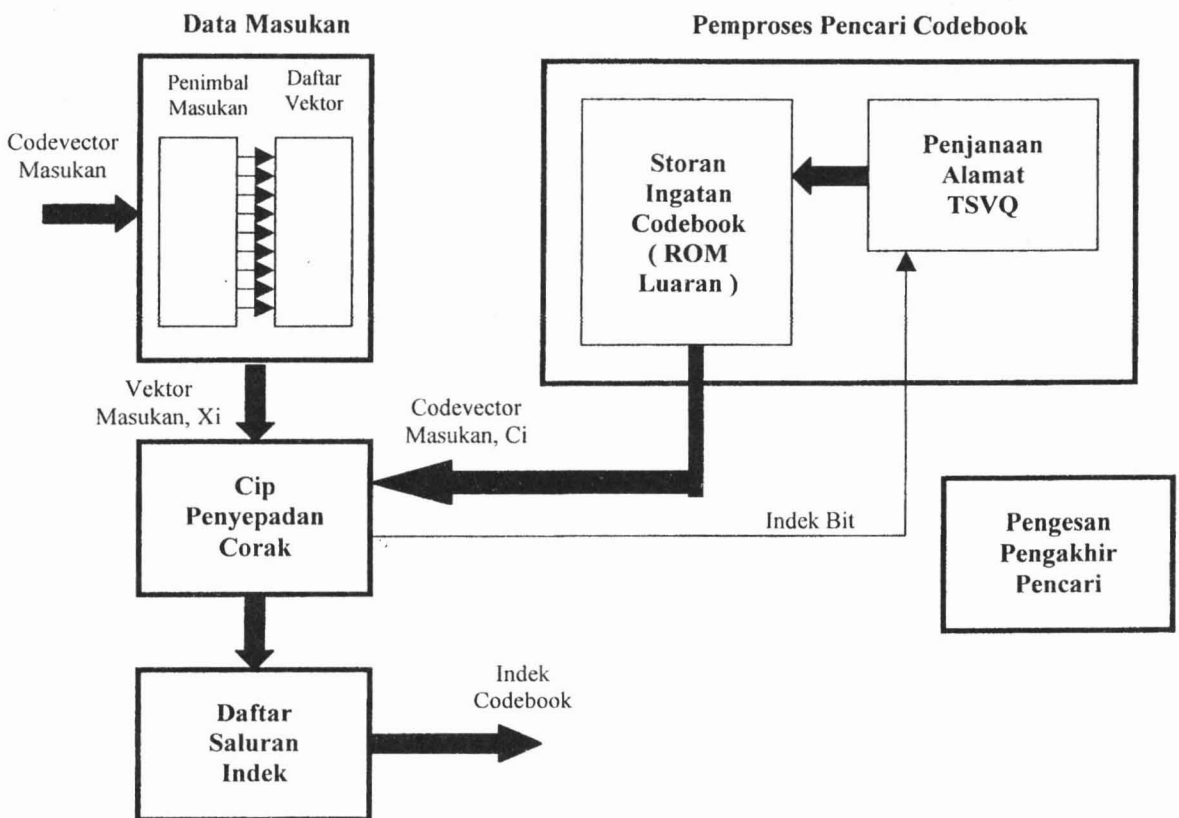
Pengkodan ialah proses untuk menentukan *raw* data dalam keadaan kod yang lebih efisien. Ini bermakna akan berlaku mengurangan bilangan bit data yang mewakili imej atau data tersebut.

Pengkodan akan mengecilkan saiz data tetapi masih mengekalkan kualiti imej. Terdapat dua kaedah pengkodan iaitu *Fixed Length Code* dan *Variable Length Code*. Algoritma yang digunakan seperti *Huffman Coding* dan *Aritmetic Coding*.

Sebelum imej pemampatan digunakan semula, imej tersebut perlu dinyahpemampatan. Proses penyahpemampatan ialah proses terbalik atau proses pembalikan semula imej pemampatan. Dimulai dengan proses penyahkod, *inverse quantization* dan *inverse transformation*.

SENI BINA PENGKODAN *TREE SEARCH VQ*

Seni bina pemprosesan Pengkodan TSVQ ini di rekabentuk secara berasingan mengikut modul-modul tertentu seperti yang ditunjukkan pada gambarajah 2. Setiap modul-modul ini mempunyai fungsi tersendiri. Seni bina TSVQ ini terbahagi kepada beberapa modul utama yang terdiri daripada modul Penerima Data, Pemproses Pencari Codebook, modul Penyepadan Corak, modul Pendaftar Indeks Saluran dan modul Pengesan Pengakhir Pencari. Antara modul-modul ini, modul Pemproses Pencari Codebook adalah yang sukar dan kompleks senibinanya.



Gambarajah 2 : Seni Bina *Tree Search Vector Quantizer*

Modul yang mempunyai peranan yang penting dalam seni bina TSVQ ini adalah modul Pemproses Pencari Codebook. Modul ini direkabentuk untuk mencapai indeks atau alamat codevector dimana indeks ini adalah nilai yang paling hampir dengan nilai vector masukan. Proses ini akan menerima vector masukan pada sesuatu tempoh masa tertentu dan ia akan di proses untuk mendapatkan sesuatu nilai indeks dari codevector yang merupakan nilai yang paling sepadan setelah tamat tempoh masa pencarian.

Codebook disimpan dalam ruang simpanan ingatan ataupun lebih dikenali sebagai ROM (Read Only Memory) dimana semua kandungan codevector di simpan didalamnya berdasarkan kepada talian alamat. Modul Penjana Alamat TSVQ akan menjanakan alamat-alamat yang sepatutnya untuk capaian

kandungan di dalam ROM. Sekiranya modul penjanaan alamat ini dijana seperti biasa, iaitu melakukan proses pembilang, ia akan berfungsi seperti seni bina penjana alamat untuk algoritma *Full Search VQ*. Untuk merekabentuk modul pemprosesan pencarian codebook bagi FSVQ ia adalah mudah jika dibandingkan dengan TSVQ. Walaubagaimanapun bagi rekabentuk seni bina TSVQ, ia memerlukan satu pin masukan sebagai pin kawalan bagi membolehkan proses pencarian bercabang di laksanakan.

Bit kawalan ia akan menunjukkan arah cabang mana yang akan dituju mengikut kaedah senibina perduaan struktur pokok. Arah cabang mana akan di pilih samaada ke kiri atau ke kanan bergantung kepada nilai perbezaan yang terendah yang diperolehi daripada perbandingan antara nilai vektor masukan dengan codevector. Di dalam kertas kerja ini, pin kawalan ini di wakili oleh nilai '0' bagi arah cabang ke kiri dan nilai '1' bagi arah cabang ke kanan.

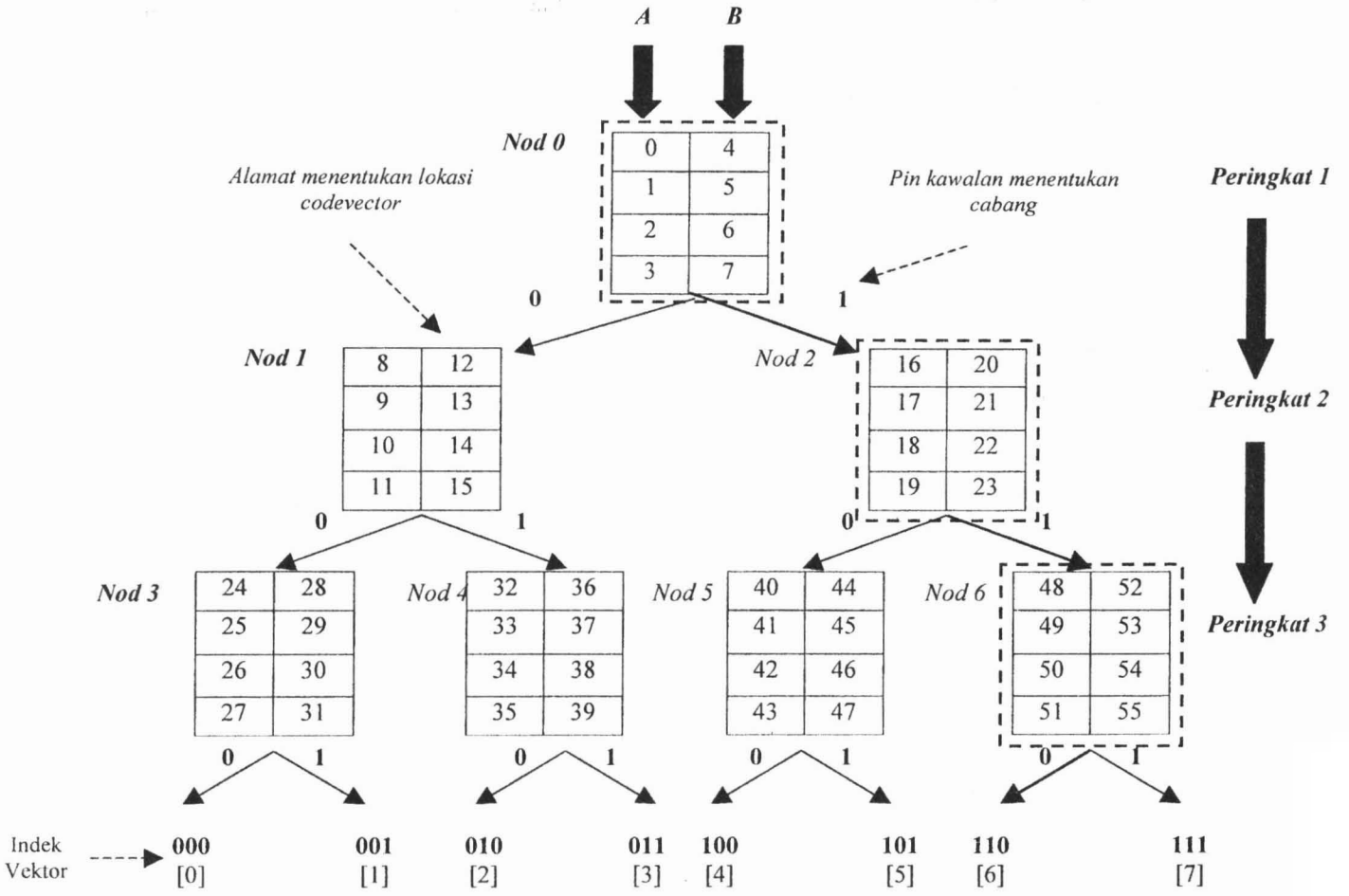
Untuk bahagian data masukan, data yang di ambil dari pixel gambar bagi satu kerangka akan di susun mengikut blok 4 x 4 pixel. Bilangan blok ini bergantung kepada saiz gambar yang digunakan. Susunan blok-blok ini akan dijadikan sebagai vektor-vektor masukan. Dimana vector-vector masukan ini akan dibandingkan dengan codevector untuk memperolehi nilai perbezaan yang minimum. Nilai perbezaan ini diperolehi mengikut kaedah ralat minimum mutlak dimana kaedah ini direkabentuk didalam modul penyepadanan corak. Indek vector ini akan dipegang oleh pendaftar saluran indek sehingga proses ke cabang terakhir tercapai. Indek vector yang dikeluarkan dari blok Pendaftar Saluran Indek inilah yang akan digunakan dalam proses penyahkod TSVQ.

ALGORITMA TREE SEARCH VQ

Modul penjana alamat memainkan peranan yang penting untuk pemilihan kawasan alamat codebook mana yang akan digunakan dalam proses pengiraan nilai perbezaan yang minimum diantara vektor masukan dengan kandungan codebook. Kaedah pengiraan ini dinamakan Pengiraan Ralat Minimum Mutlak, $MAE = \sum |C_i - X_i|$ dimana C_i ialah codevector didalam kandungan codebook di dalam ruang simpanan ingatan dan X_i pula ialah vector masukan. Di dalam kerja kerja ini, algoritma TSVQ direkabentuk berdasarkan kepada saiz vektor masuk ialah 4 dan saiz codebook ialah 8 bit iaitu proses pencarian sehingga ke peringkat 3.

Semasa proses pencarian codebook dilaksanakan, alamat permulaan bermula dari 0 pada peringkat 0 dan ia perlu mendapatkan blok mana yang paling hampir nilai dengan menggunakan persamaan MAE. Satu bit kawalan akan dijanakan untuk menentukan arah cabang mana akan dipilih untuk proses peringkat seterusnya. Jika nilai dari persamaan MAE pada blok A lebih kecil dari blok B, ia akan memilih cabang ke arah kiri dan bit kawalan ini akan di wakili dengan nilai "0". Seterusnya, jika nilai persamaan MAE pada blok B lebih kecil dari blok A, ia akan memilih cabang ke arah kanan dengan bit kawalan akan di wakili oleh nilai "1". Dan proses ini akan berterusan sehingga ke akar struktur perduaan pokok.

Sebagai contoh pada gambarajah 3, pada peringkat 0 jika persamaan MAE pada blok B lebih kecil dari blok A, maka cabang di sebelah kanan akan dipilih. Bit kawalan akan diwakili dengan nilai 1 pada peringkat 0 ini. Alamat pada storan ingatan codebook akan bermula pada alamat 16 sehingga ke alamat 23. Bagi peringkat '1' ini kawasan storan ingatan codebook yang bermula alamat 7 sehingga 15 tidak perlu lagi digunakan dalam proses ini. Bagi pengiraan seterusnya, jika persamaan blok A lebih kecil dari blok B, maka cabang di sebelah kiri akan dipilih dan bit kawalannya akan diwakili oleh nilai "0". Bagi peringkat "2" ini, alamat storan ingatan codebook yang bermula dari alamat 40 hingga 47 sahaja yang akan digunakan dalam proses pengiraan persamaan MAE. Pada peringkat ini, sekiranya blok B mempunyai nilai perbezaan yang minimum yang lebih kecil dari blok A, maka cabang di sebelah kiri akan dipilih dan bit kawalannya akan diwakili oleh nilai "1".



Gambarajah .3 : Struktur Perduaan Pokok bersaiz vektor masukan 4 dan saiz codebook 8 bit

Apabila proses pencarian pengkodan TSVQ dicapai ke cabang terakhir dalam struktur perduaan pokok ini, maka indek vector bagi mewakili proses Pengkodan TSVQ di perolehi. Indek vector inilah yang akan di hantar untuk proses penyahkodan untuk mendapatkan semula data yang hampir serupa dengan data asal. Gambarajah 4 menunjukkan alamat-alamat yang dicapai dalam proses pengkodan TSVQ seperti yang diterangkan dalam contoh diatas.

0	16	40
1	17	41
2	18	42
3	19	43
4	20	44
5	21	45
6	22	46
7	23	47

Peringkat "0" Peringkat "1" Peringkat "2"

Gambarajah 4 : Susunan Alamat yang dicapai dalam proses pencarian struktur perduaan pokok

KOMEN DAN KESIMPULAN

Pembangunan algoritma Pengkodan TSVQ kedalam bentuk logik digital dapat dilaksanakan. Algoritma yang telah diterjemahkan kedalam bentuk logik digital, direkabentuk dengan menggunakan perisian Maxplus II. Rekabentuk ini dipecahkan kepada beberapa modul dan setiap modul itu telah dilakukan pengujian melalui proses simulasi. Semua modul-modul ini akan digabungkan menjadi satu modul Pengkodan TSVQ. Untuk pengujian keseluruhan, modul penyahkodan perlu dibina. Modul penyahkodan agak mudah kerana ia hanyalah proses untuk membaca balik kandungan dalam codebook berdasarkan indekvector yang diperolehi daripada pengkodan TSVQ.

Semasa fasa implimentasi, modul pengkodan akan di aturcarakan kedalam chip FPGA manakala modul penyahkodan dibina melalui bahasa aturcara C++. Untuk proses penyahkodan dalam aplikasi masa sebenar, bahasa aturcara C++ terpaksa dibuat untuk membaca hasil pengkodan melalui antaramuka talian selari yang dilakukan oleh chip FPGA. Chip yang digunakan untuk proses pengujian ini adalah Altera Flex EPF-10k100ARC240-3. Antaramuka talian selari pada komputer peribadi digunakan di antara papan litar FPGA. Penggunaan komputer peribadi diperlukan untuk memaparkan semula imej selepas proses pemampatan berlaku dengan menggunakan bahasa aturcara C++.

Pembangunan algoritma pengkodan TSVQ ini mempunyai beberapa kelemahan dari segi sistem unit pemprosesan. Kaedah pendekatan rekabentuk seni bina pengkodan TSVQ yang lebih efisein perlu dikaji semula. Teknologi yang berasaskan pemampatan imej/video mempunyai potensi untuk terus berkembang memandangkan teknologi ini adalah *core* kepada teknologi yang berteraskan kepada ICT dan MSC. Penglibatan dengan lebih ramai para penyelidik tempatan terutamanya diperingkat institusi pengajian tinggi dan penglibatan sektor swasta perlu untuk merealisasikan Malaysia menjadi negara yang maju.

RUJUKAN

1. R. Kolagotla, S. S Yu, J. F. Jaja, (1993). *VLSI Implementation of a Tree Searched Vector Quantizer*, IEEE Trans. On Signal Proc., vol. 41, no 2 February.
2. H. Park, V. K. Prasanna, C.L. Wang (1992). *An Architecture for Tree Search Based Vector Quantization for Single Chip Implementation*. IEEE Inter. Conf. On App. Spec. Array Processors, pp. 385-398
3. J. V. McCanny, M. Yan. (1998). *A VQ Tree Search System Based On Bit Level Systolic Array*. Belfast, N. Ireland, Maple Press.
4. A. Gersho , B. Ramamurthi.(1982). *Image Coding Using Vector Quantization*. IEEE Conf. Acoust., Speech, Signal Processing, vol. 1, pp. 428-431, May 1982.
5. S. M. S. Mustafa. (1999). *Field Programmable Gate Array (FPGA) Based Vector Quantizer for Real Time Data Compression*. Universiti Teknologi Malaysia: Degree Thesis.
6. David B. Lidsky, Jan M. Rabaey. *Low Power Design of Memory Intensive Functions. Case Study: Vector Quantization*. California, USA.
7. R. Kolagotla, S. S Yu, J. F. Jaja, (1993). *VLSI Implementation of a Tree Searched Vector Quantizer*. University of Maryland, College Park: Technical Research Report.
8. Z. M. Yusof. (1995). *Lattice Vector Quantizer for Transform and Subband Image and Video Coding*. Washington State University: Ph. D Thesis.