

Perbandingan PSNR, *Bitrate*, dan MOS pada Pengkodean H.264 Menggunakan Metode Prediksi Temporal

Ari Haryadi*¹, Yohanes Suyanto²

¹Program Studi Elektronika dan Instrumentasi; Sekip Utara BLS 21, (0274) 513339

²Jurusan Ilmu Komputer dan Elektronika, FMIPA UGM, Yogyakarta

e-mail: *ari.haryadi28@gmail.com, yanto@ugm.ac.id

Abstrak

Standar pengkodean H.264/AVC merupakan hasil perumusan Joint Video Team (JVT), H.264/AVC didesain untuk menjawab kebutuhan akan tingkat kompresi yang tinggi maupun untuk dapat diimplementasikan pada berbagai aplikasi. Pada tugas akhir dilakukan perbandingan nilai PSNR, *bitrate*, dan MOS untuk masing-masing video dengan karakteristik yang berbeda.. Penelitian ini dilakukan menggunakan software referensi pengkodean video JM18.3. Hasil pengujian video *foreman*, *hall*, *news*, *waterski*, *carphone*, dan *lobby*, *bitrate* yang dihasilkan untuk setiap *sequence* pada setiap *Quantization Parameter (QP)* dipengaruhi oleh karakteristik *sequence*. Untuk hasil pengujian PSNR, diperoleh kesimpulan bahwa semakin besar nilai *Quantization Parameter* akan menghasilkan PSNR yang semakin kecil. Berdasarkan penilaian ITU-T, untuk dapat mencapai kualitas excellent (>37 dB), rata-rata nilai parameter kuantisasi yang memenuhi untuk keenam video tersebut berada pada QP 28.

Kata kunci— H.264/AVC, *bitrate*, PSNR, *prediksi temporal*, *interframe*

Abstract

H264 / AVC coding standard is developed by Joint Video Team (JVT), H.264/AVC was designed, either to meet the needs of high compression level, or to be implemented on various application. This final paper compares Peak-to-peak Signal to Noise Ratio (PSNR), *bitrate* and Mean Opinion Score (MOS) for each videos with different characteristics using library JM 18.3. Tests result on *foreman*, *hall*, *news*, *waterski*, *carphone* and *lobby* videos show that the *bitrate* produced in each *sequence* for every *Quantization Parameter (QP)* is influenced by the *sequence* characteristics. As for PSNR, it is concluded that higher QP produces smaller PSNR. Based on ITU-T scoring for excellent quality (PSNR >37 dB), the quantization parameters of the evaluated videos that meet the standard are 28.

Keywords—H.264/AVC, *bitrate*, PSNR, *temporal prediction*, *interframe*

1. PENDAHULUAN

Tujuan utama dibalik proyek H.264 adalah untuk mengembangkan standar *video coding* yang menghasilkan performa yang tinggi. H.264/AVC merupakan standar yang dapat menghasilkan tingkat kompresi tinggi sekaligus merupakan standar yang “*network-friendly*” untuk berbagai macam layanan baik “*conversational*” (*video telephony*) dan “*non-conversational*” (*storage*, *broadcast*, *streaming*). Standarisasi H.264 memiliki sejumlah tantangan yang membedakan dari standarisasi yang ada saat ini, dan disaat yang sama, H.264 saling membagi spesifikasi yang sama dengan standarisasi lain yang ada saat ini. Untuk mengetahui performansi encoder dan decoder H.264 maka perlu dianalisis kinerjanya terhadap video yang memiliki karakteristik yang berbeda-beda.

Teknologi kompresi bertujuan untuk memproses data video menjadi data yang lebih kecil, namun tetap menjaga kualitas video dan audio yang dihasilkan. H-264, MPEG-4 dan MJPEG adalah tiga teknologi kompresi DVR CCTV yang ada saat ini [1]. Ketiganya memiliki kelebihan dan kekurangan masing-masing. MPEG4 pada awalnya dikenal dengan MPEG-1, diikuti oleh MPEG-2. MPEG-4 merupakan teknologi kompresi yang di-*upgrade* dari MPEG-2 dimana konsentrasi inovasi ini ditujukan untuk menghasilkan kompresi data yang lebih baik. Teknologi kompresi MPEG-4 merupakan standar kompresi yang secara efektif dapat mengompresi suara dan data video untuk tujuan *streaming* sistem CCTV (akses CCTV via internet). H-264 dikenal sebagai teknologi kompresi MPEG-4 generasi ke-10 dimana teknologi kompresi ini menggunakan format MPEG-4, namun memiliki sejumlah perbaikan dimana teknologi kompresi H-264 memiliki kemampuan untuk memprediksikan perpindahan antar gambar video hingga sampai 32 kemungkinan. Kelebihan inilah yang membuat teknologi kompresi H-264 menghasilkan kompresi data yang lebih baik daripada MPEG-4 sehingga membuat teknologi kompresi ini paling ideal dalam *streaming* CCTV. H.264 ini menggunakan inovasi terbaru dalam teknologi kompresi video untuk menyediakan kualitas video resolusi tinggi dari jumlah terkecil yang diperlukan data video. File yang lebih kecil akan menghemat bandwidth dan biaya penyimpanan lebih dari generasi sebelumnya *video codec*. H.264 memberikan kualitas yang sama seperti MPEG-2 pada ketiga setengah *data rate* dan sampai empat kali ukuran frame MPEG-4 Bagian 2 di rate data yang sama. H.264 digunakan dalam perangkat video terbaru untuk *High Definition* perangkat video, BluRay DVD dan harus dipertimbangkan untuk setiap proyek CCTV profesional.

Teknologi telekomunikasi mengalami perkembangan yang sangat pesat [2]. Kebutuhan manusia akan sarana telekomunikasi yang canggih, cepat, dan handal, menjadi kebutuhan yang tidak dapat ditawar-tawar lagi. Salah satu sarana telekomunikasi yang banyak dikembangkan adalah transmisi video digital. Sarana ini memungkinkan terjadinya komunikasi dalam dua bentuk, yaitu gambar dan suara. Salah satu standar yang banyak digunakan dalam transmisi video digital adalah MPEG4/H.264. Dalam standar ini, terdapat salah satu modul yang bernama *Deblocking Filter*. Modul ini berfungsi untuk menghilangkan efek distorsi yang timbul dari kompresi yang dilakukan dalam standar ini.

Perkembangan dunia teknologi informasi saat ini makin pesat, sehingga informasi tidak lagi hanya dalam bentuk tulisan dan gambar diam saja, tetapi bahkan menggunakan gambar bergerak yang dilengkapi suara [3]. Perkembangan tersebut di atas juga didukung dengan perkembangan teknologi digital, sehingga saat ini telah terjadi peralihan besar-besaran dari sistem analog ke sistem digital. Proses digitalisasi ini juga berpengaruh terhadap sistem pemrosesan gambar bergerak. Sehingga format-format gambar analog pun mulai ditinggalkan dan beralih ke format gambar digital. Salah satu format gambar digital yang banyak dipakai adalah format MPEG, yang telah berkembang dari format MPEG-1 sampai dengan MPEG-7 yang masih dalam penyusunan. Format MPEG-1 menjadi salah satu standard yang dipakai untuk berbagai aplikasi seperti format MPEG-1 audio, format MPEG-1 video, dan juga *teleconference*. Format ini memiliki rasio kompresi yang cukup besar sehingga file dengan format ini memiliki ukuran yang sangat kecil dibanding dengan data asalnya. Format MPEG-1 video sepanjang durasi 60 menit dapat disimpan dalam media penyimpanan *Compact Disc* yang disebut *Video CD* atau VCD, sehingga sangat praktis bila dibandingkan dengan media penyimpanan lain seperti pita kaset video.

Kompresi video adalah bentuk kompresi data yang berhubungan dengan data video digital [4]. Kompresi video diperlukan agar penulisan data video dalam file menjadi lebih efisien. Kompresi juga diperlukan dalam *streaming* video agar transmisi data menjadi lebih cepat dan tidak memakan terlalu banyak *bandwidth*. Kompresi adalah pengubahan data kedalam bentuk yang memerlukan bit yang lebih sedikit, biasanya dilakukan agar data dapat disimpan atau dikirimkan dengan lebih efisien. Jika kebalikan dari proses ini, yaitu dekompresi, menghasilkan data yang sama persis dengan data aslinya, maka kompresi tersebut disebut *lossless compression*. Sebaliknya, dekompresi tersebut menghilangkan sebagian data, maka disebut *loosy compression*. *Loosy compression* biasanya diterapkan dalam kompresi data

berupa gambar. Walaupun tidak dapat menghasilkan data yang sama persis dengan aslinya, namun dianggap lebih efisien. Video pada dasarnya merupakan array tiga dimensi. Dua dimensi digunakan untuk menggambarkan ruang pergerakan gambar, dan satu dimensi menggambarkan waktu. Sebuah frame adalah kumpulan *pixel* pada suatu waktu. Pada dasarnya, frame sama dengan gambar. Data video mengandung *redundancy* (pengulangan). Kesamaan tersebut dapat dikodekan dengan mencatat perbedaan dalam sebuah frame atau antara frame. Kompresi video pada umumnya mengurangi pengulangan tersebut dengan *loosy compression*. Pada saat ini, hampir semua kompresi video menerapkan standar MPEG (*Motion Picture Expert Group*). Salah satu teknik yang paling kuat untuk video mengompresi adalah kompresi *interframe*. Kompresi *Interframe* menggunakan satu atau lebih atau yang lebih baru frame sebelumnya dalam urutan untuk kompres frame lancar, sedangkan kompresi *intraframe* hanya menggunakan frame saat ini, yang efektif kompresi gambar. Metode yang paling umum digunakan bekerja dengan membandingkan setiap frame dalam video dengan yang sebelumnya. Jika jendela memiliki wilayah di mana tidak ada yang bergerak, sistem hanya mengeluarkan perintah pendek yang salinan yang bagian dari frame sebelumnya, bit-untuk-bit, ke yang berikutnya. Jika bagian dari memindahkan bingkai dengan cara sederhana, kompresor memancarkan perintah (sedikit lebih panjang) yang menceritakan *decompresser* bergeser, memutar, meringankan, atau menggelapkan copy- perintah lagi, tapi masih jauh lebih pendek daripada kompresi *intraframe*. kompresi *Interframe* bekerja dengan baik untuk program yang hanya akan diputar kembali oleh penonton, tetapi dapat menyebabkan masalah jika urutan video perlu diedit. Karena data kompresi *interframe* salinan dari satu frame ke yang lain, jika frame asli hanya dipotong (atau hilang di transmisi), frame berikut ini tidak dapat direkonstruksi dengan benar. format video Beberapa, seperti DV, kompres setiap frame secara independen dengan menggunakan kompresi *intraframe*. 'Memotong' Pembuatan dalam *intraframe*-video terkompresi hampir semudah mengedit video tidak terkompresi - satu menemukan awal dan akhir setiap frame, dan hanya salinan-bit-bit untuk setiap frame yang satu ingin tetap, dan membuang frame satu yang tidak diinginkan. Perbedaan lain antara *intraframe* dan kompresi *interframe* adalah bahwa dengan sistem *intraframe*, setiap frame menggunakan jumlah yang sama data. Dalam sistem *interframe* kebanyakan, frame tertentu tidak diizinkan untuk menyalin data dari frame lain, dan memerlukan lebih banyak data dari frame lain di sekitarnya.

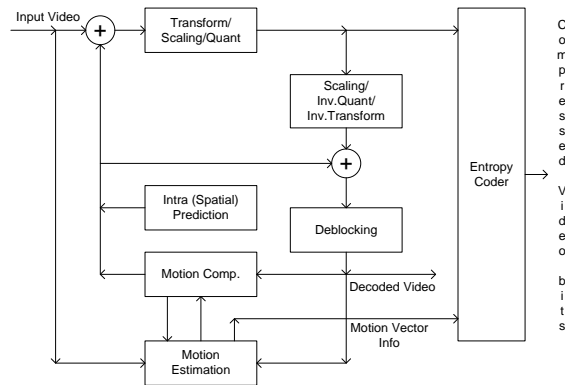
2. METODE PENELITIAN

2.1 Metode Analisa

Metode analisa yang dipakai dalam penelitian ini adalah melalui analisis nilai PSNR, *bitrate*, dan melakukan penilaian baik secara subjektif maupun objektif. Teknik kompresi yang digunakan yaitu H.264 dengan menggunakan metode prediksi temporal /*interframe*. Penelitian ini menganalisa 6 buah video yaitu foreman, hall, news, waterski, carphone, dan lobby.

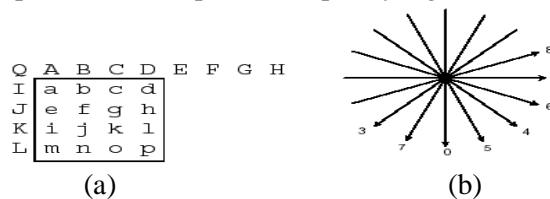
2.2 Arsitektur Penelitian

Arsitektur Penelitian yang digunakan yaitu pengkodean H.264 yang dapat dilihat pada Gambar 1. Pada penelitian ini video masukan dibagi menjadi makroblok-makroblok dan akan melalui proses-proses seperti prediksi, transformasi, kuantisasi dan pengkodean entropi. Berikut proses yang akan dilalui oleh tiap-tiap frame dalam video.



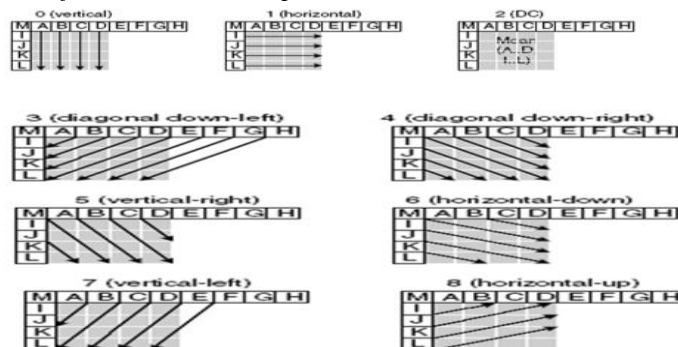
Gambar 1 Kompresi H.264

Proses pertama yang dilalui oleh makroblok adalah prediksi. Pengkodean *intraframe* dibagi menjadi beberapa tipe yaitu mode 4x4 , 16x16 serta PCM. Mode pengkodean 4x4 melakukan prediksi pada setiap blok *luminance* berukuran 4x4 dan sesuai untuk pengkodean pada bagian citra yang memiliki detail signifikan. Berbeda dengan mode 4x4, mode pengkodean 16x16 melakukan prediksi pada blok *luminance* berukuran 16x16 dan sesuai untuk pengkodean pada bagian citra yang halus. Mode terakhir adalah PCM. Mode PCM ini tidak melakukan proses prediksi dan langsung melakukan proses transformasi pada nilai sampel. Pada mode 4x4, setiap blok 4x4 diprediksi dari sampel tetangga seperti yang ditunjukkan gambar 2 (a). 16 sampel pada blok 4x4 yang dinotasikan sebagai a-p diprediksi dengan menggunakan sampel yang dinotasikan sebagai A-Q. Untuk setiap blok 4x4 dapat memilih satu dari sembilan mode prediksi. Selain prediksi “DC” (dimana satu nilai digunakan untuk memprediksikan keseluruhan blok 4x4), terdapat pula delapan arah mode prediksi seperti yang diberikan pada gambar 2 (b).



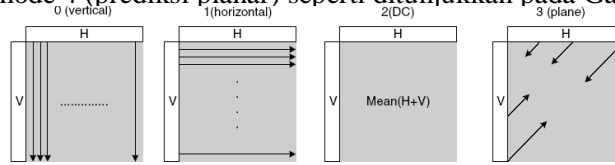
Gambar 2 (a) Prediksi Intra 4x4 untuk sampel a-p menggunakan sampel A-Q
(b) Delapan “Arah prediksi” untuk prediksi intra 4x4

Gambar 3 menunjukkan sembilan mode prediksi intra 4x4. Untuk mode 0 (prediksi vertikal) sampel-sampel diatas blok 4x4 dicopi ke dalam blok yang ditunjukkan oleh panah. Mode 1 (prediksi horizontal) bekerja dengan cara yang sama namun sampel yang dicopi adalah yang berada di sebelah kiri. Mode 2 adalah prediksi DC sedangkan 6 Mode lainnya merupakan mode prediksi diagonal yaitu *diagonal-down-left*, *diagonal-down-right*, *vertical-right*, *horizontal-down*, *vertical-left* dan *horizontal-up*.



Gambar 3 Mode Prediksi Intra 4 x 4

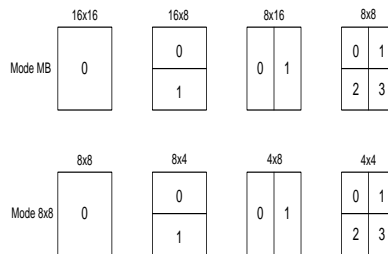
Pada mode 16x16, seluruh komponen *luminance* dari makroblok diprediksikan. Terdapat empat mode prediksi yaitu mode 0 (prediksi vertikal), mode 1 (prediksi horizontal), mode 2 (prediksi DC) dan mode 4 (prediksi planar) seperti ditunjukkan pada Gambar 4.



Gambar 4 empat mode prediksi

2.2.1 Prediksi Inter-frame

P-slice menggunakan prediksi *interframe* dengan melakukan *motion estimation*. *Motion estimation* dapat dilakukan pada makroblok 16x16 atau partisi sub-makroblok berukuran 16x8, 8x16, 8x8, 8x4, 4x8 dan 4x4 seperti yang ditunjukkan oleh Gambar 5. Ketika dipilih sampel 8x8, satu elemen *syntax* tambahan harus dikirimkan untuk menunjukkan apakah partisi ini lebih jauh akan dipartisi lagi menjadi 8x4, 4x8 atau 4x4.



Gambar 5 Segmentasi makroblok untuk *motion estimation*

Sinyal prediksi dari setiap *predictive-coded* MxN blok ditentukan oleh *translational motion vector* dan indeks dari *frame* referensi. Satu partisi sub-makroblok hanya diizinkan memiliki satu *motion vector* (MV). Sehingga, apabila makroblok dikodekan menggunakan 4 buah partisi 8x8 dan setiap partisi tersebut dibagi lagi menjadi partisi 4 buah 4x4, 16 *motion vector* akan dikirimkan untuk satu makroblok P dan ini merupakan jumlah maksimum.

Akurasi dari *motion compensation* adalah 1/4 dari jarak antara sampel. *Motion* dapat diestimasi dari *frame-frame* sebelum maupun sesudahnya. Untuk mengestimasi *motion*, nilai piksel diinterpolasi untuk mencapai akurasi 1/4 piksel. Pada kasus dimana *motion vector* menunjukkan posisi sampel integer, sinyal prediksi akan mengandung sampel pada citra referensi.

Pada B-slice dapat digunakan dua *motion vector* untuk merepresentasikan dua motion untuk setiap partisi sub-makroblok. Kedua *motion vector* ini dapat berasal dari citra referensi sebelum maupun sesudahnya. Harga rata-rata piksel pada citra referensi digunakan sebagai prediktor.

2.2.2 Transformasi, Kuantisasi Dan Scanning

Transformasi menggunakan 16 bit dan bukan 32 bit. Transformasi dilakukan dengan penjumlahan dan *bit-shifts*, Transformasi dilakukan pada ukuran 4x4 dan bukan 8x8, Karena ukuran makroblok komponen *luminance* adalah 16x16 dan komponen *chrominance* adalah 8x8, maka harus terlebih dahulu dibagi menjadi blok 4x4 untuk diaplikasikan pada matriks transformasi T yang merupakan matriks integer yang mirip dengan DCT.

$$T = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{bmatrix}, H = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix}, C = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

Matriks transformasi T diaplikasikan baik pada komponen *luminance* maupun *chrominance*. Sebagai tambahan, ketika mode prediksi intra 16x16 digunakan, koefisien DC

dari 16 buah blok 4x4 dipilih dan ditransformasikan dengan menggunakan transformasi Hadamard (H) dan blok 2x2 dari komponen *chrominance* DC ditransformasikan dengan matriks C yang merupakan transformasi Hadamard 2x2.

Koefisien-koefisien setelah transformasi dikuantisasi menggunakan *Quantization Parameter* (QP) yang dapat berubah untuk setiap makroblok. Nilai ini disusun sehingga peningkatan 1 pada parameter memiliki arti peningkatan step kuantisasi sekitar 12% yang berarti pula penurunan *bit rate* sebesar 12%.

Koefisien transformasi yang telah dikuantisasi kemudian *discanning* secara *zig-zag fashion* dan diteruskan dengan pengkodean entropi. Dan koefisien DC 2x2 dari komponen *chrominance* *discanning* secara *raster-scan order*.

2.2.3 Pengkodean Entropi

Pengkodean Entropi merupakan teknik pengkodean *lossless* yang menggantikan elemen data dengan kode yang dengan kombinasinya dengan transformasi sebelumnya akan menghasilkan penurunan ukuran data secara signifikan. Dua mode pengkodean entropi pada standar ini adalah *Variable Length Coding* (VLC), misalnya *Huffman Coding* dan *Binary Arithmetic Coding* (BAC). Saat ini, H.264/AVC telah mendesain pengkodean entropi adaptif yaitu *Context-Adaptive Variable Length Coding* (CAVLC) dan *Context-Adaptive Binary Arithmetic Coding* (CABAC).

2.2.3 Deblocking Filter

H.264/AVC akan mengalami *blocking artifacts* yang disebabkan oleh *block based transform* pada prediksi *intraframe* dan *interframe* serta proses kuantisasi koefisien transformasi. Penggunaan *deblocking filter* akan mengurangi efek “*blockiness*” pada *block boundary* dan mencegah propagasi noise. H.264/AVC menggunakan *deblocking filter* untuk performansi pengkodean yang lebih baik walaupun harus dibayar dengan implementasi yang kompleks. Filter ini digunakan setelah *inverse transform* dan diaplikasikan pada sisi horizontal dan vertikal dari *block* berukuran 4 x 4 dalam sebuah *macroblock*. Citra yang telah difilter digunakan untuk memprediksi *motion* untuk citra lain. *Deblocking filter* merupakan filter adaptif yang bekerja sesuai dengan mode kompresi makroblok (*intra* atau *inter*), parameter kuantisasi, *motion vector*, pengkodean *frame/field* dan sebagainya. Untuk ukuran kuantisasi yang kecil, filter akan secara langsung *off*.

2. 3 Kriteria Penilaian

Proses kompresi akan mengakibatkan terjadinya distorsi pada data hasil rekonstruksi. Dalam sebuah sistem kompresi, distorsi yang terjadi sebaiknya masih dapat diterima oleh sistem visual manusia. Penilaian dapat dilakukan baik secara objektif (berdasarkan perhitungan parameter-parameter matematis tertentu) atau secara subjektif (berdasarkan sistem visual manusia). Penilaian ini akan menentukan performansi suatu sistem kompresi.

2. 3.1 Kriteria Penilaian Objektif

Penilaian data hasil rekonstruksi dapat dinilai dengan parameter *Peak-to-peak Signal-to-Noise Ratio* (PSNR). Untuk *frame* berukuran MxN, parameter penilaian secara objektif dapat diuraikan sebagai berikut.

$$PSNR = 10 \log \frac{255^2}{\frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} |f(x, y) - g(x, y)|^2} \quad (1)$$

Dengan :

M = jumlah piksel per baris
 N = jumlah baris pada *image*

x,y = posisi piksel dalam *image*
 $f(x,y)$ = intensitas *luminance* piksel pada *original image*
 $g(x,y)$ = intensitas *luminance* piksel pada *reconstructed image*

2.3.2 Kriteria Penilaian Subjektif

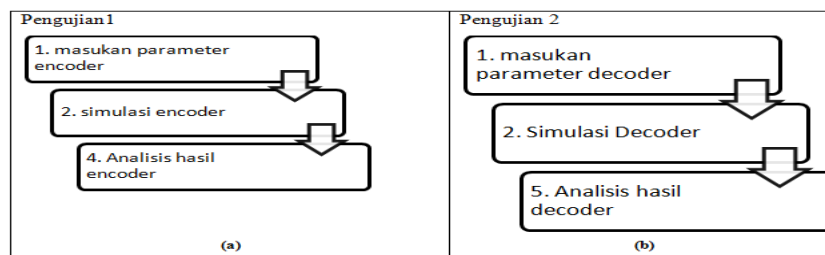
Penilaian secara subjektif dilakukan dengan membandingkan deretan *frame* video asli dengan hasil rekonstruksinya. Nilai MOS juga bisa didasarkan pada hasil perhitungan nilai PSNR. Pedoman penilaian didasarkan pada standar ITU yang ditunjukkan pada Tabel 1.

Tabel 1 ITU-T P.910 *quality and impairment scale*

Scale	Quality	Impairment	PSNR (dB)
5	Excellent	Imperceptible	>37
4	Good	Perceptible, not annoying	31-37
3	Fair	Slightly annoying	25-31
2	Poor	Annoying	20-25
1	Bad	Very annoying	<20

2.4 Implementasi Program

Program yang digunakan dalam penelitian ini adalah *Microsoft Visual C++ 2008* dengan menggunakan *library JM 18.3*. Pada *library JM 18.3* terdapat dua buah file konfigurasi yaitu *encoder.cfg* dan *decoder.cfg*. Masing-masing input video akan dikodekan oleh *encoder* dengan output berupa file kompresi H.264 dan direkonstruksi berupa file YUV oleh *decoder* sesuai dengan nilai parameter kuantisasi masing-masing. File video yang akan dijadikan input pada program yaitu *foreman*, *hall*, *news*, *waterski*, *carphone*, dan *lobby*. Gambaran implementasi program yang akan digunakan dapat dilihat pada Gambar 6.



Gambar 6 Flowchart pengujian sistem (a) simulasi *encoder* (b) simulasi *decoder*

2.4.1 Implementasi Encoder

Pada Gambar 6 (a), pengujian file konfigurasi *encoder.cfg* dilakukan dengan memasukkan *input* video berekstensi YUV ke dalam *encoder*. Penelitian ini dilakukan pada level main profile. Parameter yang digunakan dalam *encoder.cfg* yaitu *FramesToBeEncoded* (banyaknya frame yang akan dikodekan), *SourceWidth* (lebar frame), *SourceHeight* (tinggi frame), *OutputWidth* (lebar frame *output*), *OutputHeight* (tinggi file *output*), *profileIDC* (profil yang digunakan), nilai parameter kuantisasi untuk *I – P – B frame*, mode pencarian, parameter *weight prediction*, parameter *deblocking filter*, parameter *fast mode decision*, parameter *entropy coding*, dan penempatan file *output*.

2.3.1 Implementasi Decoder

Proses pengujian *decoder* pada Gambar 6 (b) dilakukan dengan menggunakan program *Microsoft Visual C++ 2008* dengan menggunakan *library JM 18.3*. Pada program JM, terdapat file konfigurasi *decoder.cfg*. file ini diubah parameter-parameter yang akan digunakan, kemudian melakukan perintah *build ldecod* agar memperoleh file *executable* dengan nama *ldecod.exe*.

3. HASIL DAN PEMBAHASAN

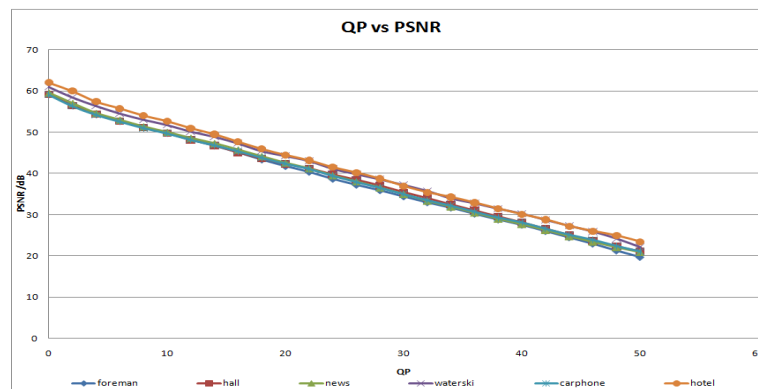
File video foreman, hall, news, waterski, carphone, dan lobby dilakukan kompresi dengan variasi nilai kuantisasi, dan hasilnya dicatat sesuai nilai kuantisasinya dan hasil output program disimpan pada foldernya masing-masing.

3.1 Perbandingan QP VS PSNR

Untuk dapat melihat pengaruh QP terhadap nilai PSNR dilakukan dengan cara memasukan nilai QP dari 0 - 50, kemudian diamati dan dibandingkan nilai PSNR yang dihasilkan tiap-tiap QP pada video Foreman , Hall , Dan News, Waterski, Carphone, Dan Lobby, maka didapat data pada tabel 2. Pada Gambar 7 ditunjukkan perbandingan QP VS PSNR antara video Foreman , Hall , Dan News, Waterski, Carphone, Dan Lobby dapat dilihat bahwa ketiganya memiliki karakteristik PSNR yang tidak terlalu jauh beda.

Tabel 2 Nilai PSNR tertinggi dan terendah foreman, hall, news, waterski, carphone dan Lobby

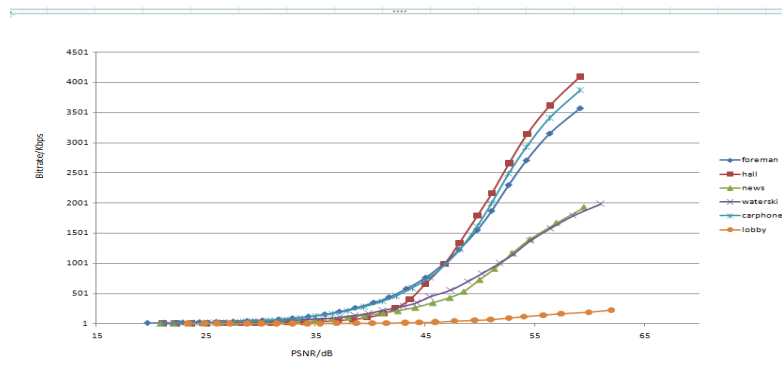
QP	PSNR (dB)					
	foreman	hall	news	waterski	carphone	lobby
0	59.162	59.161	59.526	61.022	59.148	62.009
50	19.654	21.069	20.875	22.113	21.113	23.405



Gambar 7 Perbandingan QP VS PSNR video Foreman, hall, news, waterski, carphone, dan lobby

3.2 Perbandingan PSNR VS Bitrate

Untuk dapat melihat pengaruh PSNR terhadap nilai *Bitrate* dilakukan dengan cara mengamati dan membandingkan hasil nilai PSNR dan *bitrate* yang kemudian dibandingkan dari masing-masing video, maka diperoleh hasil yang dapat dilihat pada Gambar 8.



Gambar 8 Perbandingan PSNR VS *Bitrate* video Foreman, hall, news, waterski, carphone, dan lobby

Jika dilihat nilai *bitrate* dari keenam video tersebut memiliki perbedaan yang cukup signifikan, hal ini dikarenakan karakteristik masing-masing video yang sangat berbeda sehingga untuk proses kompresi menjadi bentuk .264 memiliki pemrosesan yang berbeda pula sesuai karakteristik masing-masing. Untuk hall memiliki karakteristik perubahan pergerakan yang sedikit pada waktu tertentu. Sementara news memiliki karakteristik perubahan secara berkelanjutan (continuous) sepanjang waktu dari awal hingga akhir video pada bagian-bagian tertentu.

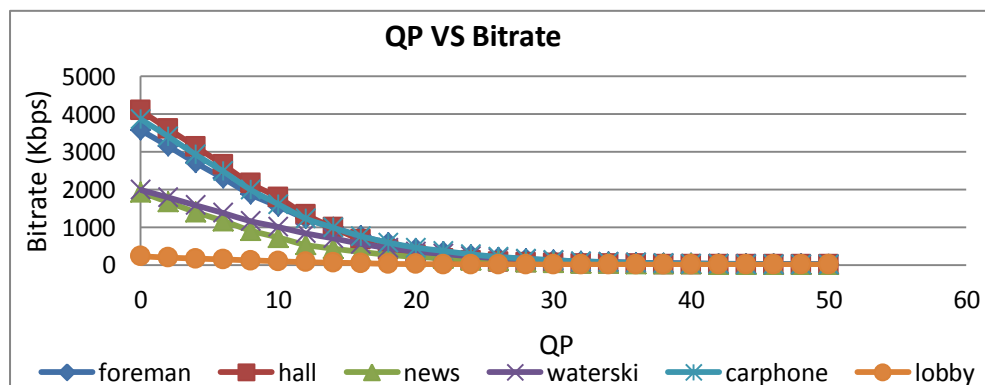
Foreman memiliki nilai awal *bitrate* yang lebih besar karena seluruh frame bergerak dan pergerakannya tidak stabil, hal ini menyebabkan prediksi temporal dilakukan pada seluruh bagian dan terus menerus dari waktu awal hingga akhir video. waterski memiliki grafik yang hampir sama seperti news, hanya saja pada nilai PSNR kecil, nilai *bitrate* waterski lebih besar daripada news. Hal ini dikarenakan pada bagian akhir video, waterski memiliki adegan yang dominan pemandangan laut dan para pemain ski terlihat kecil sehingga menghasilkan *bitrate* yang hampir sama seperti news yang pergerakannya hanya pada bagian tertentu.

3.3 Perbandingan QP VS Bitrate

Untuk dapat melihat pengaruh QP terhadap nilai *Bitrate* dilakukan dengan cara memasukan nilai QP dari 0 - 50, kemudian diamati dan dibandingkan nilai *Bitrate* yang dihasilkan untuk tiap-tiap QP, maka didapat data sebagai berikut. Dari tabel 3 dan Gambar 9, dapat dilihat bahwa secara keseluruhan semakin besar nilai kuantisasi yang diberikan akan menghasilkan nilai *bitrate* yang semakin kecil, namun apabila dilihat dari masing-masing video, hall memiliki nilai *bitrate* terbesar pada QP = 0 yaitu 4096.22 Kbps, dan lobby memiliki *bitrate* terkecil untuk QP = 50 sebesar 2.28 Kbps. News memiliki nilai *bitrate* terkecil pada QP = 0 yaitu 1928.24 Kbps, dan pada QP 50 akan menghasilkan *bitrate* 5.98 Kbps.

Tabel 3 Perbandingan nilai tertinggi (QP 0) dan terendah (QP 50) Foreman, Hall, News, Waterski, Carphone, Dan Lobby

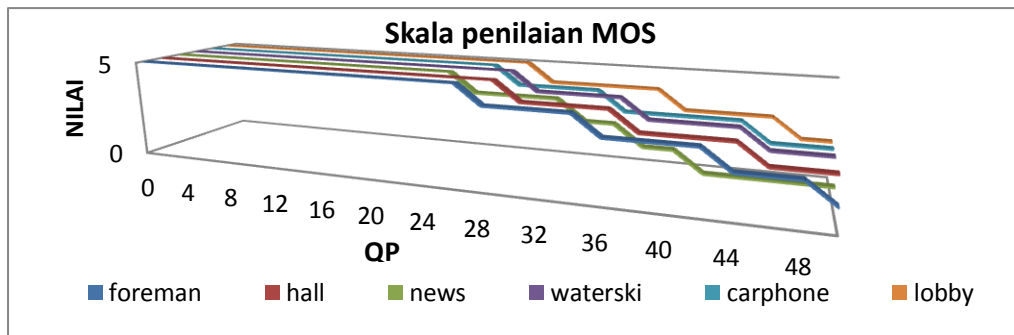
QP	Bitrate/Kbps					
	foreman	hall	news	waterski	carphone	lobby
0	3576.67	4096.22	1928.2	1989.2	3872.2	227.54
50	13.9	3.82	5.98	12.05	6.93	2.28



Gambar 9 Perbandingan QP VS *Bitrate* video Foreman, hall, news, waterski, carphone, dan lobby

3.4 Perbandingan QP VS MOS

Berdasarkan pedoman penilaian yang didasarkan pada standar ITU-T yang ditunjukkan pada Tabel 1, maka diperoleh nilai MOS untuk masing – masing *sequence* yang ditampilkan pada Gambar 10.



Gambar 10 Skala Penilaian MOS

Dari hasil nilai penilaian MOS diatas ditunjukkan bahwa untuk hasil pengujian memiliki nilai skala MOS yang bervariasi untuk masing- masing video. Sehingga dapat dikatakan hasilnya memiliki kualitas *Bad, Poor, Fair, Good, Excellent*. Hal ini didasarkan pada hasil perhitungan PSNR yang diadaptasi kedalam nilai MOS (*ITU-T P.800 quality and impairment scale*). Pada Gambar 5.22 menunjukan semakin bertambahnya QP maka semakin menurun kualitasnya. Rata-rata QP yang termasuk kategori excellent dari keenam video tersebut berada pada QP 28.

4. KESIMPULAN

1. Semakin besar nilai *Quantization Parameter* akan menghasilkan PSNR yang semakin kecil. Nilai PSNR pada QP 0 dari keenam video memiliki *range* antara 59.148 dB hingga 62.009 dB. Nilai PSNR pada QP 50 memiliki *range* antara 19.654 dB hingga 23.405 dB.
2. Semakin besar nilai *Quantization Parameter* akan menghasilkan *bitrate* yang semakin kecil. Nilai *bitrate* pada QP 0 dari keenam video memiliki *range* antara 227.54 Kbps hingga 4096.22 Kbps. Nilai *bitrate* pada QP 50 dari keenam video memiliki *range* antara 2.28 Kbps hingga 13.9 Kbps.
3. Untuk dapat mencapai PSNR yang *excellent* (>37 dB), rata-rata nilai parameter kuantisasi yang memenuhi untuk keenam video tersebut berada QP 28.

5. SARAN

Perlu dikembangkan untuk simulasi pengkodean H.264 dengan menggunakan intraframe dan profile yang lebih tinggi.

DAFTAR PUSTAKA

- [1] Ariawan, P.S., 2011, *Kompresi Video H.264 / MPEG4 Digunakan Untuk Streaming CCTV*, Jurusan Teknik Elektro fakultas Teknik Universitas Udayana, Bali.
- [2] Hutomo, D.B., 2011, *Deblocking Filter Untuk kompresi Video Menggunakan Standar MPEG4/H.264*, Jurusan Teknik Elektro fakultas Teknik Universitas Udayana, Bali.
- [3] Candra, H, 2002, *Video MPEG-1*, Dosen Jurusan Teknik Elektro-FTI, Universitas Trisakti, Jakarta.
- [4] Firmansah, 2011, *Kompresi Video Menggunakan Standar MPEG*, Jurusan Teknik Elektro fakultas Teknik Universitas Udayana, Bali.