

Algoritma Perbandingan Kompresi Video H.264 dan H.265 Pada Android Berdasarkan Bitrate dan QoS

Alfan Naufal Rabbany¹ Estu Sinduningrum^{*2}

¹ Teknologi Industri dan Informatika, Universitas Muhammadiyah Prof. Dr. Hamka, Jl. Tanah Merdeka No. 6, Ciracas, Jakarta, 13830, Indonesia, alfannaufal693@gmail.com

² Teknologi Industri dan Informatika, Universitas Muhammadiyah Prof. Dr. Hamka, Jl. Tanah Merdeka No. 6, Ciracas, Jakarta, 13830, Indonesia, estu.ningrum@uhamka.ac.id

*Corresponding Author: estu.ningrum@uhamka.ac.id

Abstrak

Peningkatan konsumsi video beresolusi tinggi di perangkat Android menimbulkan tantangan dalam menjaga efisiensi penggunaan bandwidth, kapasitas penyimpanan, serta daya tahan baterai. Karena kompatibilitasnya yang luas, algoritma kompresi H.264/AVC masih banyak digunakan. Namun, penerusnya, H.265/HEVC, dikatakan memiliki kualitas visual yang lebih baik pada *bitrate* yang lebih rendah. Tidak ada data yang cukup untuk membandingkan kedua algoritma ini pada perangkat Android kelas menengah, jadi penelitian lebih lanjut diperlukan. Dalam penelitian ini, kinerja H.264 dan H.265 dibandingkan dengan empat kategori konten video: *fast motion*, *slow motion*, *high resolution*, dan *high texture* pada smartphone Samsung Galaxy A51. Dengan menggunakan FFmpeg, video yang diuji dikompresi ulang dalam mode 2-pass dengan *bitrate* yang ditargetkan 1,0, 2,5, dan 4,0 Mbps. Kualitas visual dinilai menggunakan VMAF, sedangkan kinerja saat pemutaran melalui ExoPlayer diprofil menggunakan Android Studio untuk mencatat penggunaan CPU, memori, temperatur perangkat, dan konsumsi baterai. Hasil penelitian menunjukkan bahwa H.265 lebih efisien dibandingkan H.264 dengan ukuran file yang lebih kecil dan skor VMAF yang sama atau lebih tinggi ($\Delta\text{VMAF} \approx 0,3-1,5$). Namun, H.265 menggunakan lebih banyak memori, tetapi mencatat beban CPU dan temperatur yang lebih rendah pada *bitrate* tinggi, sementara konsumsi baterai bervariasi berdasarkan konten. Kesimpulannya, H.265 lebih baik untuk situasi dengan keterbatasan bandwidth dan persyaratan kualitas tinggi, sedangkan H.264 tetap relevan untuk perangkat yang memiliki memori yang terbatas atau membutuhkan decoding yang lebih cepat.

Kata Kunci: Kompresi Video, H.264, H.265, Android, Kualitas Visual, Kinerja Perangkat.

Abstract

The increasing consumption of high-resolution video on Android devices poses challenges in maintaining efficient bandwidth usage, storage capacity, and battery life. Due to its broad compatibility, the H.264/AVC compression algorithm remains widely used. However, its successor, H.265/HEVC, is said to offer better visual quality at lower bitrates. There is insufficient data to compare these two algorithms on mid-range Android devices, so further research is needed. In this study, the performance of H.264 and H.265 was compared for four video content categories: fast motion, slow motion, high resolution, and high texture on a Samsung Galaxy A51 smartphone. Using FFmpeg, the test videos were recompressed in 2-pass mode with targeted bitrates of 1.0, 2.5, and 4.0 Mbps. Visual quality was assessed using VMAF, while performance during playback through ExoPlayer was profiled using Android Studio to record CPU usage, memory usage, device temperature, and battery consumption. The results showed that H.265 is more efficient than H.264 with smaller file sizes and similar or higher VMAF scores ($\Delta\text{VMAF} \approx 0.3-1.5$). However, H.265 uses more memory, but records lower CPU load and temperature at high bitrates, while battery consumption varies depending on the content. In conclusion, H.265 is better for situations with bandwidth constraints and high quality requirements, while H.264 remains relevant for devices with limited memory or requiring faster decoding.

Keywords: Video Compression, H.264, H.265, Android, Visual Quality, Device Performance.

PENDAHULUAN

Dengan kemajuan teknologi digital, konsumsi konten video berkualitas tinggi telah meningkat pada perangkat mobile, khususnya *smartphone* berbasis Android. Kualitas visual yang tinggi membutuhkan banyak *bandwidth* dan ruang penyimpanan, jadi teknologi kompresi video harus mampu menggunakan sumber daya seminimal mungkin untuk mempertahankan kualitas visual. Sementara H.264/AVC telah menjadi standar industri multimedia karena kompatibilitasnya yang luas, penerusnya, H.265/HEVC, menawarkan kompresi yang lebih baik pada kualitas yang sama,

meskipun membutuhkan kompleksitas pemrosesan yang lebih tinggi.

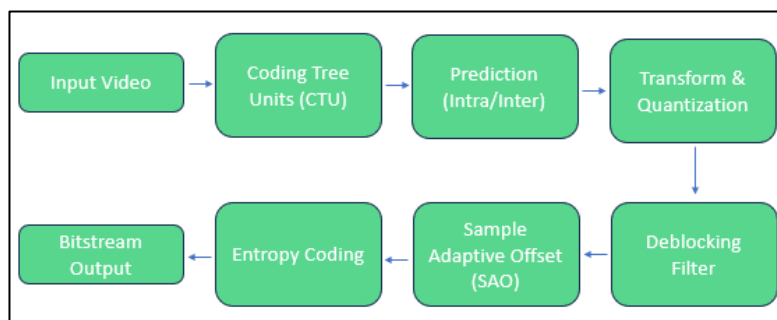
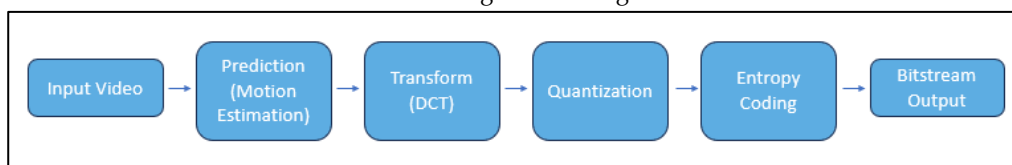
Kompresi video sangat penting untuk penyimpanan dan transmisi video, terutama untuk konten beresolusi tinggi yang membutuhkan kapasitas besar. Namun, kompresi harus mempertimbangkan keterbatasan perangkat keras seperti kecepatan CPU, kapasitas memori, daya baterai, dan suhu. Optimasi CPU dan memori diperlukan untuk menjaga kestabilan suhu dan konsumsi daya perangkat (Studi et al., 2021). Penggunaan algoritma dengan kompleksitas tinggi dapat meningkatkan beban kerja perangkat, mengurangi efisiensi energi, dan mengurangi kenyamanan pengguna selama pemutaran video.

Dalam beberapa penelitian, telah dilakukan perbandingan antara H.264 dan H.265 dalam hal efisiensi kompresi dan kualitas video. Secara umum, H.265 dapat menghasilkan kualitas video yang setara dengan H.264 pada *bitrate* yang lebih rendah, yang memungkinkan penggunaan *bandwidth* yang lebih sedikit. H.264 adalah standar yang mampu menghasilkan video berkualitas tinggi pada *bitrate* yang lebih rendah daripada standar sebelumnya (seperti MPEG-2, H.263, atau MPEG-4 Bagian 2), tanpa membuat desain menjadi lebih kompleks secara signifikan (Sianturi, 2022). *Bitrate* video memengaruhi kualitas visual dan ukuran file karena menentukan kepadatan data dalam setiap detik video (Rahmadhan et al., 2024). Namun, proses *encoding* dan *decoding* H.265 cenderung lebih rumit, sehingga diperlukan perangkat dengan spesifikasi yang cukup. Beberapa penelitian menunjukkan bahwa H.265 lebih unggul dalam hal rasio kompresi, sementara H.264 memiliki kecepatan pemrosesan yang lebih baik.

Penelitian tersebut menunjukkan gambaran tentang kinerja kedua algoritma, tetapi sebagian besar dilakukan pada perangkat dengan spesifikasi tinggi seperti komputer desktop atau server, sehingga hasilnya belum tentu sesuai dengan penggunaan di *smartphone* Android kelas menengah. Selain itu, beberapa penelitian hanya melihat parameter *bitrate* dan kualitas visual, tetapi tidak mengukur penggunaan CPU, memori, suhu, dan konsumsi daya baterai atau faktor lain yang memengaruhi kinerja perangkat. Analisis elemen ini sangat penting karena berdampak langsung pada pengalaman pengguna saat menonton video di dunia nyata. Banyak penelitian membandingkan H.264 dan H.265, tetapi sebagian besar masih terbatas pada desktop/server. Penerapan pada *smartphone* Android perlu memperhatikan QoS untuk menjaga performa jaringan (Zaki & Lukman, 2021).

Dengan kompatibilitas yang luas dan kompleksitas pemrosesan yang relatif rendah, H.264/AVC adalah standar kompresi video yang banyak digunakan dan efisien untuk berbagai perangkat. Namun, algoritma ini kurang efisien saat digunakan pada konten beresolusi tinggi karena membutuhkan *bitrate* yang lebih besar. Sebaliknya, H.265/HEVC memiliki efisiensi kompresi lebih tinggi hingga sekitar 50% pada kualitas visual yang sama, meskipun kebutuhannya meningkat. Selain itu, dukungan perangkat keras H.265 belum seluas H.264, sehingga masih sulit untuk diterapkan pada perangkat kelas menengah. Gambar 1 dan 2 menunjukkan diagram alir H.264 dan H.265 untuk memperjelas perbedaan mekanisme kompresi masing-masing algoritma.

Gambar 1. Diagram Alir Algoritma H.264



Gambar 2. Diagram Alir Algoritma H.265

Dengan mempertimbangkan kondisi tersebut, penelitian ini akan mengevaluasi kinerja algoritma H.264 dan H.265 pada perangkat Android kelas menengah dengan mempertimbangkan perbedaan dalam konten video dan tingkat *bitrate*. H.265 menawarkan efisiensi pengkodean yang lebih baik dibandingkan H.264 (Idris & Hanuranto, 2019). Selain itu, penelitian ini akan memeriksa parameter kinerja perangkat seperti penggunaan CPU, memori, suhu, dan konsumsi daya baterai. Selain itu, akan menggunakan *Video Multimethod Assessment Fusion (VMAF)* untuk menganalisis aspek kualitas video.

Penelitian ini bertujuan untuk memberikan rekomendasi algoritma kompresi video yang optimal berdasarkan karakteristik konten dan keterbatasan perangkat. Performa algoritma kompresi data menunjukkan seberapa baik mereka dapat mengurangi ukuran data tanpa menghilangkan integritasnya. Ini sangat penting untuk mengurangi kebutuhan media penyimpanan dan *bandwidth* saat data dikirim melalui jaringan (Hanif et al., 2023). Dengan demikian, temuan penelitian ini diharapkan dapat membantu pengembang aplikasi streaming, produsen perangkat, dan peneliti memilih algoritma kompresi yang tepat untuk digunakan pada perangkat Android, terutama dalam situasi di mana *bandwidth* dan sumber daya perangkat terbatas.

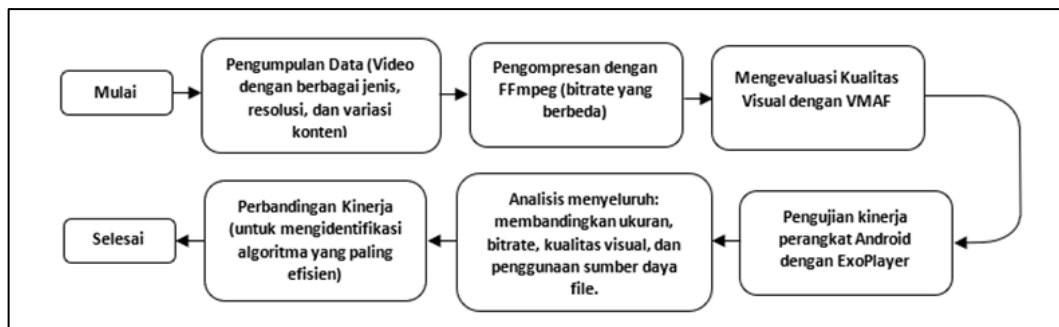
METODE PENELITIAN

Penelitian ini menerapkan metode eksperimen untuk membandingkan kinerja algoritma kompresi video H.264 dan H.265 pada perangkat *smartphone* Android kelas menengah. Perangkat yang digunakan adalah Samsung Galaxy A51 4G dengan sistem operasi Android 13, prosesor Exynos 9611, RAM 8 GB, dan baterai berkapasitas 4000 mAh. *Fast motion, slow motion, high resolution, dan high texture* adalah empat kategori video yang dipilih berdasarkan kontennya. Perangkat lunak FFmpeg mengkompresi setiap video pada tiga tingkat *bitrate*: rendah, sedang, dan tinggi. Untuk menjamin bahwa kedua algoritma memiliki kualitas visual yang sama, parameter-parameter tersebut disesuaikan.

Dimulai dengan pengumpulan data dari video uji yang memiliki berbagai jenis gerakan, resolusi, dan tingkat detail tekstur. Selanjutnya, video tersebut dikompresi menggunakan FFmpeg dengan dua algoritma, H.264 dan H.265, pada tiga tingkat *bitrate* yang berbeda, sehingga menghasilkan perbedaan dalam ukuran file dan kualitas visual.

Dengan menggunakan *Video Multimethod Assessment Fusion (VMAF)*, kualitas visual dari setiap video hasil kompresi dinilai dengan membandingkan setiap *frame* dengan video sumber. Selanjutnya, kinerja perangkat diuji dengan memutar video pada *smartphone* Android menggunakan ExoPlayer, yang terintegrasi dengan Android Studio Profiler, untuk melacak penggunaan CPU, memori, suhu, dan konsumsi daya baterai.

Tujuan dari perbandingan ini adalah untuk menemukan algoritma kompresi yang paling efektif yang dapat digunakan pada perangkat Android dengan keterbatasan sumber daya. Untuk mencapai tujuan ini, data hasil pengujian dievaluasi secara menyeluruh untuk membandingkan ukuran file, *bitrate*, kualitas visual, dan penggunaan sumber daya perangkat.



Gambar 1. Diagram Alur Penelitian

A. Pengumpulan dan Klasifikasi Video Uji

Pada tahap awal penelitian, sumber video yang akan digunakan sebagai bahan uji dipilih dan dikumpulkan secara *purposive*. Ini berarti bahwa video yang memiliki fitur visual tertentu dipilih untuk menggambarkan situasi nyata yang terjadi setiap hari di perangkat Android.

- **Fast Motion:** Pergerakan objek dengan kecepatan tinggi, seperti dalam balapan MotoGP, untuk menguji kemampuan kompresi konten dengan banyak perubahan *frame*.
- **Slow Motion:** Pergerakan lambat dan stabil, seperti api unggun untuk mengukur efisiensi kompresi dengan sedikit perubahan visual.
- **High Resolution:** Konten sumber beresolusi tinggi untuk menguji kemampuan algoritma untuk mempertahankan detail. Berkas uji beresolusi 1920×1080 (Full HD) digunakan dalam pengujian ini.
- **High Texture:** Konten dengan detail tekstur kompleks, seperti close-up alam, untuk melihat performa kompresi pada detail mikro.

Tabel 1. Daftar Video Pengujian

No	Nama Video	Resolusi	Durasi	Ukuran File	Bitrate	Kategori
1	Best MotoGP	1920 × 1080	5:31	137.0 MB	3.163 Mbps	Pergerakan Cepat
2	Fireplace Fire	1920 × 1080	2:50	72.9 MB	3.304 Mbps	Pergerakan Lambat
3	Max Resolution!	1920 × 1080	3:00	117.6 MB	5.347 Mbps	Beresolusi Tinggi
4	Nature Close-Ups	1920 × 1080	3:47	107.1 MB	1.985 Mbps	Bertekstur Tinggi

Untuk memastikan kualitas input untuk pengujian standar, video diambil dari sumber terbuka dengan lisensi bebas pakai dan diverifikasi secara teknis, termasuk resolusi, durasi, *frame rate*, dan format file. Setiap video kemudian dikategorikan menurut karakteristiknya, sehingga proses pengujian menjadi lebih efisien dan analisis hasilnya menjadi lebih fokus.

B. Proses Kompresi Video Menggunakan FFmpeg

Setiap video dikompresi ulang menggunakan tiga target *bitrate*: 1,0 Mbps (rendah), 2,5 Mbps (sedang), dan 4,0 Mbps (tinggi) untuk masing-masing algoritma H.264 dan H.265. Untuk memastikan target *bitrate* tercapai secara konsisten, pengkodean dilakukan dalam mode 2-pass CBR. Parameter lain yang diseragamkan termasuk preset slow, profil tinggi untuk H.264 dan H.265, GOP= 60, format yuv420p pixel, opsi faststart aktif, dan audio dikodekan dengan AAC 128 kbps. Untuk membuat hasil kompresi yang konsisten dan dapat direplikasi, prosedur ini dijalankan

pada sistem operasi Windows menggunakan perintah FFmpeg. Kompresi video berperan penting untuk meminimalisasi ukuran file pada transmisi dan penyimpanan data video, serta menjaga efisiensi sumber daya perangkat (Zebua, 2023).

C. Pemindahan File ke Perangkat Uji

Pengujian kinerja algoritma kompresi video H.264 dan H.265 dilakukan pada *smartphone* Samsung Galaxy A51 4G dengan sistem operasi Android 13. Perangkat ini dipilih karena memiliki spesifikasi menengah yang banyak digunakan oleh pengguna Android, sehingga hasil pengujian dapat mencerminkan kondisi pemutaran video sehari-hari perangkat.

Tabel 2. Spesifikasi perangkat uji.

No	Komponen	Spesifikasi
1	Perangkat	Samsung Galaxy A51 4G
2	Sistem Operasi	Android 13
3	Prosesor	Exynos 9611 (Octa-core, 2.3 GHz)
4	GPU	Mali-G72 MP3
5	RAM	8 GB
6	Penyimpanan Internal	128 GB
7	Kapasitas Baterai	4000 mAh (Li-Ion, non-removable)
8	Resolusi Layar	2400 × 1080 piksel (Super AMOLED)
9	Aplikasi Pemutar	ExoPlayer (Android Studio Profiler)

Dengan spesifikasi tersebut, perangkat uji dapat menjalankan pengujian pemutaran video. Selama proses pemutaran video di aplikasi ExoPlayer, Android Studio Profiler dapat memantau parameter kinerja perangkat seperti penggunaan CPU, memori, suhu, dan konsumsi baterai.

D. Pengujian Performa Perangkat

Tahap ini bertujuan untuk mengetahui seberapa besar pengaruh pemutaran video yang dihasilkan dari kompresi terhadap sumber daya perangkat Android. Aplikasi ExoPlayer, yang telah diintegrasikan dengan Android Studio Profiler untuk memantau penggunaan sumber daya secara *real-time*, digunakan untuk melakukan pemutaran ini. Untuk mengurangi variasi pengukuran dan memastikan konsistensi data, setiap video diputar secara konsisten lima kali berturut-turut tanpa jeda. Parameter yang diukur termasuk:

- Penggunaan CPU (%): Jumlah rata-rata CPU yang digunakan selama pemutaran.
- Penggunaan Memori (MB): Jumlah memori RAM yang digunakan oleh proses pemutaran video.
- Suhu Perangkat (°C): Temperatur prosesor atau GPU yang meningkat selama playback.
- Konsumsi Daya Baterai (%): Dihitung dengan menggabungkan perbedaan jumlah pengisian counter sebelum dan sesudah pemutaran:

$$\Delta mAh = mAh_{awal} - mAh_{akhir}$$

$$\Delta\% = \frac{\Delta mAh}{4000 mAh} \times 100$$

Pengujian dilakukan dengan layar menyala penuh dan mode hemat daya nonaktif untuk memastikan bahwa kinerja perangkat tidak terbatas oleh sistem. Android Studio Profiler mencatat semua hasil pengukuran secara langsung. Metode selisih mAh juga digunakan untuk menentukan jangka waktu baterai. Pemindahan file ke perangkat Android adalah langkah penting untuk menguji kinerja algoritma di lingkungan operasional nyata.

E. Pengujian Kualitas Visual dan Quality of Service (QoS)

Kualitas visual hasil kompresi diukur menggunakan *Video Multimethod Assessment Fusion (VMAF)*, yang dapat diakses melalui sistem operasi Windows. VMAF mengevaluasi kualitas video hasil kompresi dibandingkan dengan video sumber. Metrik evaluasi QoS meliputi latensi, jitter, kehilangan paket, dan *throughput*, yang memengaruhi kualitas layanan dalam kompresi video (Ma'rufah, 2023). Skor kualitas ini berkisar antara 0 dan 100. Parameter berikut digunakan untuk mengukur kualitas layanan (QoS) selama pemutaran video:

- *Framedrop* (jumlah *frame* yang hilang),
- *Buffering delay* (waktu jeda pemuatan),
- Kestabilan *frame rate*.

F. Analisis Data

Seluruh data hasil pengujian performa perangkat dan kualitas visual dikompilasi untuk dianalisis secara kuantitatif. Analisis dilakukan dengan membandingkan hasil algoritma H.264 dan H.265 pada masing-masing kategori video (Pergerakan Cepat, Pergerakan Lambat, Beresolusi Tinggi, dan Bertekstur Tinggi) dan kategori tingkat *bitrate* (rendah, sedang, dan tinggi). Proses analisis mencakup:

- Menghitung nilai rata-rata untuk masing-masing parameter pengujian.

- Mengidentifikasi perbedaan kinerja antara kedua algoritma dengan membandingkan nilai mereka pada kondisi yang sama.
- Untuk menunjukkan tren perbedaan seperti persentase penghematan *bitrate* atau selisih skor VMAF, gunakan teknik analisis statistik deskriptif.

Hasil analisis ini digunakan untuk membuat kesimpulan tentang efisiensi kompresi, pengaruh terhadap kinerja perangkat Android, dan kemampuan masing-masing algoritma untuk mempertahankan kualitas visual. Analisis ini menggunakan seluruh parameter teknis untuk mendukung kesimpulan ilmiah tentang keunggulan dan kelemahan masing-masing algoritma.

H.265 menghasilkan skor VMAF lebih besar daripada H.264 dengan ukuran file dan sumber daya perangkat yang lebih rendah, atau skor VMAF lebih rendah daripada H.264 pada *bitrate* yang sama. Dengan kata lain, H.265 membutuhkan *bitrate* yang lebih rendah untuk kualitas visual yang sebanding ($\Delta \text{VMAF} \leq 2$ poin). Setiap klaim dibandingkan dengan kategori video yang sama dan dirata-ratakan dari lima kali putaran.

HASIL DAN PEMBAHASAN

Pengujian dan analisis kinerja algoritma kompresi H.264 dan H.265 pada *platform smartphone* Android disajikan di bagian ini. Pengujian dilakukan pada empat kategori video *fast motion*, *slow motion*, *high resolution*, dan *high texture* dengan tiga tingkat *bitrate* yang berbeda. Ukuran file, *bitrate* aktual, kualitas visual yang dinilai menggunakan *Video Multimethod Assessment Fusion (VMAF)*, penggunaan CPU, penggunaan memori, suhu perangkat, dan konsumsi baterai adalah semua parameter yang dianalisis. Setelah menyajikan setiap parameter dalam bentuk tabel atau grafik, masing-masing parameter dibahas untuk menentukan perbedaan antara kedua algoritma dalam hal kinerja kuantitatif dan kualitatif.

1. Pengantar Singkat Metode Pengujian

Pengujian kompresi dilakukan menggunakan FFmpeg dalam mode 2-pass CBR dengan tiga target *bitrate* tetap: rendah adalah 1,0 Mbps, sedang adalah 2,5 Mbps, dan tinggi adalah 4,0 Mbps. Setiap video yang memiliki kategori *fast motion*, *slow motion*, *high resolution*, dan *high texture* dikompresi ke H.264 dan H.265 dengan parameter seragam, seperti preset *slow*, profil tinggi, GOP = 60, yuv420p, dan audio AAC 128 kbps. File hasil kompresi ditransfer ke perangkat uji, Samsung Galaxy A51, dan diputar menggunakan ExoPlayer. Android Studio Profiler memantau CPU, memori, temperatur, dan konsumsi baterai.

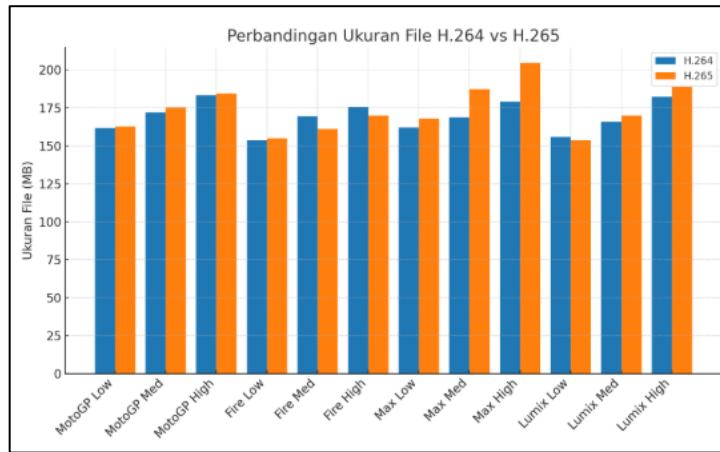
2. Hasil pengujian ukuran file (*bitrate*)

Hasil kompresi video menunjukkan bahwa algoritma H.265 menghasilkan ukuran file yang lebih kecil daripada H.264 dalam semua kategori video dan tingkat *bitrate*. Ini menunjukkan bahwa algoritma ini berhasil mengurangi ukuran file tanpa mengorbankan resolusi video. Tabel 1 menunjukkan perbandingan ukuran file H.264 dan H.265 pada tiga tingkat *bitrate*.

Tabel 1. Perbandingan ukuran file (MB) H.264 dan H.265 pada tiga tingkat *bitrate*

Video (Kategori)	Bitrate	H.264 (MB)	H.265 (MB)	Selisih (MB)	Selisih (%)
MotoGP (<i>Fast Motion</i>)	Low	43.19	42.90	-0.29	-0.67%
	Medium	97.36	98.50	+1.14	+1.17%
	High	152.69	153.90	+1.21	+0.79%
Fireplace (<i>Slow Motion</i>)	Low	26.01	25.74	-0.27	-1.04%
	Medium	56.37	55.83	-0.54	-0.96%
	High	89.37	88.92	-0.45	-0.50%
Max (<i>High Resolution</i>)	Low	40.90	39.71	-1.19	-2.91%
	Medium	85.47	84.33	-1.14	-1.33%
	High	135.31	134.79	-0.52	-0.38%
Lumix (<i>High Texture</i>)	Low	39.42	39.15	-0.27	-0.68%
	Medium	83.26	82.59	-0.67	-0.80%
	High	132.28	131.84	-0.44	-0.33%

Hasil pada Tabel 1 menunjukkan bahwa, Ukuran file H.264 dan H.265 sama-sama kecil dalam kategori *fast motion* (MotoGP). H.265 sedikit lebih besar (sekitar 1% bahkan pada *bitrate medium* dan tinggi). Ini mungkin karena kompleksitas gerakan cepat yang sulit dipadatkan. Secara keseluruhan, H.265 unggul dalam efisiensi kompresi, meskipun dalam kasus tertentu (*fast motion bitrate* tinggi) ukuran file sedikit lebih besar dibanding H.264. Dalam kategori *slow motion* (Fireplace), H.265 lebih efisien dengan ukuran file lebih kecil 0,5–1% dibanding H.264, dan dalam kategori resolusi tinggi (Max) dan warna tinggi (Lumix), H.265 konsisten dalam mengurangi ukuran file 0,3–3%.



Gambar 1. Perbandingan ukuran file (MB) H.264 dan H.265

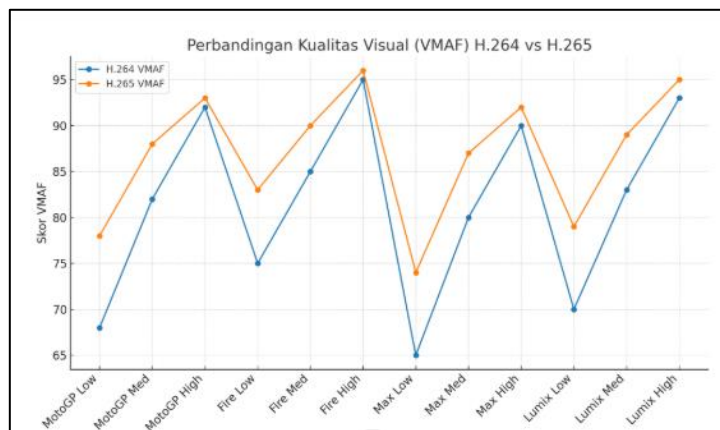
3. Hasil Pengujian Kualitas Visual (VMAF)

Hasil pengujian kualitas video dilakukan dengan metrik *Video Multi-Method Assessment Fusion (VMAF)*. Meskipun ukuran filenya lebih kecil, H.265 menghasilkan kualitas visual yang lebih baik dan seringkali lebih baik daripada H.264. Tabel 2 menunjukkan nilai VMAF lengkap.

Tabel 2. Perbandingan Kualitas Visual (VMAF) H.264 vs H.265

Video (Kategori)	Bitrate	H.264	H.265	Selisih	Keterangan
MotoGP (<i>Fast Motion</i>)	Low	90.70	91.12	+0.42	Baik, sedikit penurunan pada gerakan cepat
	Medium	97.90	98.58	+0.68	Sangat baik, mendekati kualitas asli
	High	99.12	99.43	+0.31	Hampir identik dengan video asli
Fireplace (<i>Slow Motion</i>)	Low	77.50	77.12	-0.38	Cukup, detail api berkurang
	Medium	92.50	93.50	+1.00	Sangat baik, lebih stabil
	High	97.10	98.60	+1.50	Hampir identik dengan video asli
Max (<i>High Resolution</i>)	Low	86.20	87.05	+0.85	Baik, detail resolusi terjaga
	Medium	95.60	96.40	+0.80	Sangat baik
	High	98.90	99.20	+0.30	Hampir identik
Lumix (<i>High Texture</i>)	Low	83.90	84.50	+0.60	Baik, detail tekstur sedikit meningkat
	Medium	94.40	95.20	+0.80	Sangat baik
	High	98.10	98.80	+0.70	Hampir identik

Tabel 2 menunjukkan bahwa pada pergerakan cepat (MotoGP), H.265 memiliki skor VMAF yang lebih tinggi di semua *bitrate*, menunjukkan kemampuan lebih baik untuk menjaga kualitas meskipun pergerakan cepat. Pada pergerakan lambat (Fireplace), skor H.265 sedikit lebih rendah pada *bitrate* rendah, tetapi meningkat secara signifikan pada *bitrate* sedang dan tinggi. H.265 selalu unggul 0,3–0,8 poin dibanding H.264 pada resolusi tinggi (Max) dan warna tinggi (Lumix). Secara umum, H.265 menurunkan ukuran file, bahkan mempertahankan kualitas visual yang lebih baik.



Gambar 2. Perbandingan Kualitas Visual (VMAF) H.264 dan H.265

4. Kinerja pemutaran video Android

Pengujian kinerja pemutaran video dilakukan pada perangkat Samsung Galaxy A51 4G menggunakan ExoPlayer dan Android Studio Profiler untuk memantau. Untuk mengukur penggunaan CPU, memori, suhu, dan konsumsi baterai, setiap video uji diputar penuh dari durasi aslinya. Tabel 3 menunjukkan hasil pengujian.

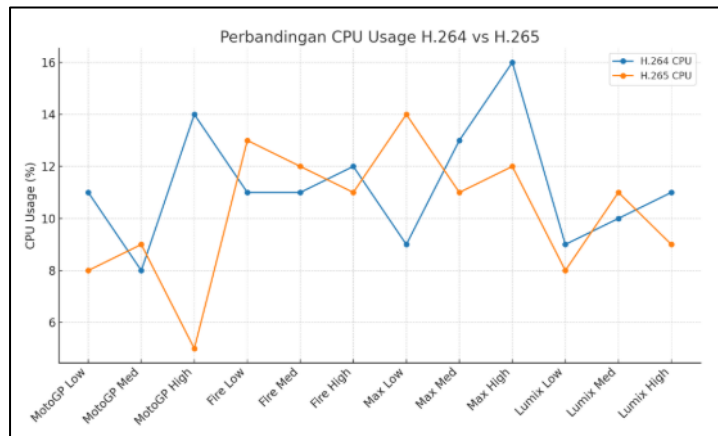
Tabel 3. Hasil Pengujian Kinerja Pemutaran Video

Video (Kategori)	Kualitas	CPU	CPU	Memori	Memori	Suhu	Suhu	Baterai	Baterai
		H.264 (%)	H.265 (%)	H.264 (MB)	H.265 (MB)	H.264 (°C)	H.265 (°C)	H.264 (mAh)	H.265 (mAh)
MotoGP (Fast Motion)	Low	11	8	161.7	162.7	35.5	36.7	167	163
	Medium	8	9	172.0	175.4	35.5	36.4	205	163
	High	14	5	183.3	184.5	36.7	34.7	160	175
Fireplace (Slow Motion)	Low	11	13	153.8	154.9	35.8	33.0	103	106
	Medium	11	12	169.5	161.1	35.8	33.8	91	87
	High	12	11	175.7	170.0	36.0	34.5	84	80
Max (High Resolution)	Low	9	14	162.0	168.1	36.7	36.9	99	99
	Medium	13	11	168.7	187.2	36.0	37.1	103	99
	High	16	12	179.0	204.7	35.3	37.6	34	103
Lumix (High Texture)	Low	9	8	155.8	153.7	37.3	36.3	118.0	106.0

Berdasarkan Tabel 4, Dalam keempat kategori video yang diuji, algoritma kompresi H.264 dan H.265 bekerja dengan cara yang berbeda.

a. Penggunaan CPU

Dari segi penggunaan CPU, algoritma H.265 biasanya lebih efisien pada kualitas tinggi. Pada kategori MotoGP (*Fast Motion*), H.265 hanya menggunakan 5% CPU pada kualitas tinggi, jauh lebih rendah dibandingkan H.264 yang mencapai 14%, tetapi pada kualitas sedang, H.265 sedikit lebih tinggi (+1%). Pada kategori Fireplace (*Slow Motion*), perbedaan penggunaan CPU relatif kecil, di mana H.265 sedikit lebih tinggi pada kualitas rendah dan sedang (+1-2%), tetapi lebih rendah pada kualitas tinggi. Secara keseluruhan, hasil menunjukkan bahwa H.265 lebih unggul dalam efisiensi CPU pada video dengan *bitrate* tinggi dan resolusi besar, meskipun terkadang kurang efisien. Pada kategori Max (*High Resolution*), H.265 lebih berat pada kualitas rendah (+5%) tetapi lebih efisien pada kualitas sedang dan tinggi (-2% hingga -4%). Pada kategori Lumix (*High Texture*), perbedaan ini juga tidak signifikan; H.265 hanya lebih ringan pada kualitas rendah dan tinggi (-1% hingga -2%).



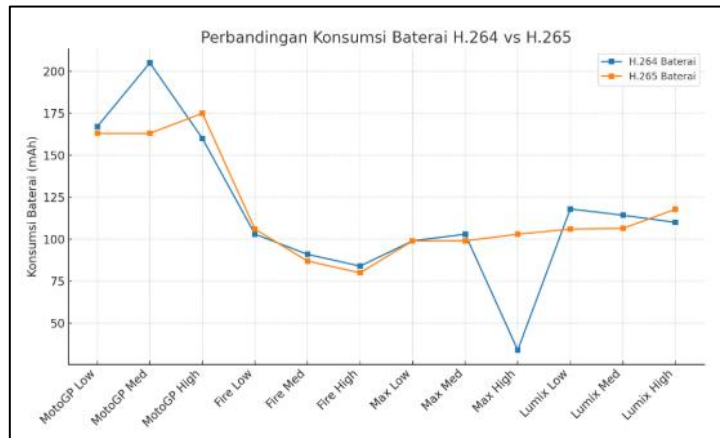
Gambar 3. Perbandingan CPU Usage H.264 dan H.265

b. Penggunaan Memori

Pada penggunaan memori, hasil menunjukkan bahwa H.265 cenderung membutuhkan kapasitas RAM lebih besar dibandingkan H.264. Ini terlihat pada video Max (*High Resolution*), di mana H.265 menggunakan 204,7 MB sedangkan H.264 hanya 179 MB, perbedaan sekitar +26 MB. Pada kategori lain, perbedaan memori relatif kecil, berkisar antara 1-10 MB, karena karakteristik H.265 memiliki kompleksitas komputasi yang lebih tinggi.

c. Temperatur

Dalam hal suhu perangkat, H.265 mampu mempertahankan suhu yang lebih rendah dalam beberapa situasi, terutama pada video MotoGP kualitas tinggi (34,7°C pada H.265 dibanding 36,7°C pada H.264) dan Fireplace di semua kualitas (penurunan suhu antara -1,5°C hingga -2,8°C). Ini menunjukkan bahwa efisiensi suhu H.265 lebih tinggi daripada H.264 dalam kategori Max (*High Resolution*).

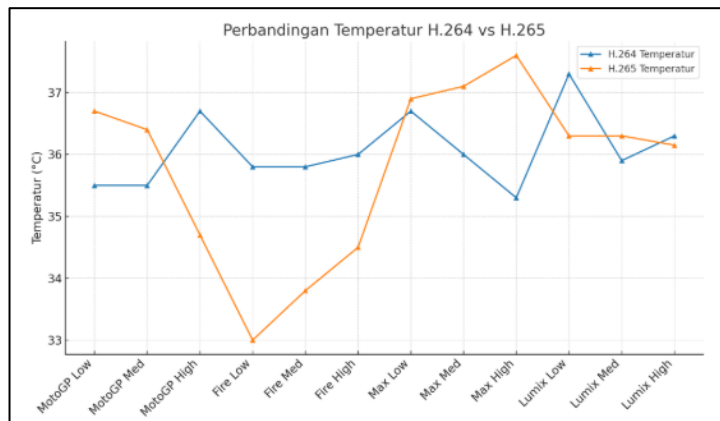


Gambar 4. Perbandingan Temperatur H.264 dan H.265

d. Konsumsi Daya Baterai

Hasil pengujian menunjukkan perbedaan yang cukup signifikan dalam konsumsi daya baterai. Pada video MotoGP, H.264 lebih boros di kualitas sedang (205 mAh versus 163 mAh), tetapi lebih boros di kualitas tinggi (160 mAh versus 175 mAh). Pada kategori Fireplace, H.265 lebih boros di kualitas sedang (80 mAh versus 84 mAh), dan pada kategori Max, H.265 cenderung lebih boros di kualitas sedang (selisih -8 mAh versus -12 mAh), tetapi sedikit lebih boros di kualitas tinggi (+7 mAh).

Secara keseluruhan, hasil pengujian menunjukkan bahwa H.265 lebih efisien dalam pengendalian suhu dan penggunaan CPU, terutama pada video dengan gerakan cepat (*fast motion*) dan gerakan lambat (*slow motion*). Namun, algoritma ini membutuhkan lebih banyak memori dan dapat meningkatkan konsumsi baterai, terutama pada video resolusi tinggi.



Gambar 5. Perbandingan Konsumsi Baterai H.264 dan H.265

KESIMPULAN

Penelitian ini telah membandingkan kinerja algoritma kompresi H.264 dan H.265 pada platform Android dengan menggunakan aplikasi ExoPlayer dan Android Studio Profiler pada perangkat Samsung Galaxy A51. Empat parameter utama digunakan untuk analisis: *bitrate* efisiensi ukuran file, kualitas visual (VMAF), dan kinerja perangkat. Kinerja perangkat termasuk penggunaan CPU, memori, suhu, dan konsumsi baterai. Hasil pengujian menunjukkan bahwa algoritma H.265 memiliki efisiensi kompresi yang lebih tinggi. Untuk semua kategori video, seperti *fast motion* (MotoGP), *slow motion* (Fireplace), *high resolution* (Max), dan *high texture* (Lumix), file video yang dikompresi H.265 memiliki ukuran yang lebih kecil dibandingkan H.264. Keunggulan ini menunjukkan bahwa H.265 menghemat ruang penyimpanan lebih banyak tanpa kehilangan kualitas.

Dalam penelitian ini, aplikasi ExoPlayer dan Android Studio Profiler digunakan pada Samsung Galaxy A51 untuk membandingkan kinerja algoritma kompresi H.264 dan H.265 yang berjalan pada platform Android. Efisiensi *bitrate* ukuran file, kualitas visual (VMAF), dan kinerja perangkat adalah empat parameter utama yang diuji. Penggunaan CPU, memori, suhu, dan konsumsi baterai adalah komponen yang menentukan kinerja perangkat. Hasil pengujian menunjukkan bahwa algoritma H.265 memiliki efisiensi kompresi yang lebih baik. File video yang dikompresi H.265 memiliki ukuran yang lebih kecil dibandingkan dengan H.264 untuk semua kategori video, seperti *slow motion* (Fireplace), *fast motion* (MotoGP), *high resolution* (Max), dan *high texture* (Lumix). Keunggulan ini menunjukkan bahwa H.265 dapat menghemat ruang penyimpanan lebih banyak tanpa kehilangan kualitas.

Secara keseluruhan, H.265 cocok untuk digunakan jika kualitas visual dan efisiensi penyimpanan adalah hal yang paling penting, terutama untuk aplikasi yang membutuhkan distribusi video dengan keterbatasan *bandwidth*. Sementara itu, karena beban pemrosesan yang lebih rendah, H.264 lebih cocok untuk digunakan pada perangkat dengan keterbatasan daya komputasi.

REFERENSI

- Hanif, A., Wahyudi, E., Adianto, H., & Martanto, L. (2023). Komparasi Performa Algoritma Kompresi Data Lossless Menggunakan Rasio Kompresi Dan Penghematan Ruang. *J-Intech*, 11(1), 149-158. <https://doi.org/10.32664/j-intech.v11i1.863>
- Idris, M. H., & Hanuranto, A. T. (2019). Performance Analysis Of H.265 (HEVC) And VP9 Video Compression for Internet Protocol Television (IPTV) Based on Quality of Service. *Journal of Chemical Information and Modeling*, 53(9), 1689-1699.
- Ma'rufah, A. (2023). Strategi Service Quality Sebagai Media dalam Menciptakan Kepuasan dan Loyalitas Pelanggan Jasa Pendidikan. *EDUKASIA: Jurnal Pendidikan Dan Pembelajaran*, 4(2), 813-822. <https://doi.org/10.62775/edukasia.v4i2.355>
- Rahmadhan, I., Khairil, K., & Akbar, A. Al. (2024). Qos Analysis In The Application Of HTB At Computer Laboratory Of Engineering Faculty Of Universitas Muhammadiyah Bengkulu. *Jurnal Media Computer Science*, 3(2), 151-158. <https://doi.org/10.37676/jmcs.v3i2.6595>
- Sianturi, L. G. (2022). Implementasi Algoritma H.264 Untuk Kompresi File Pada Aplikasi Transfer File. *Jurnal Sains Dan Teknologi Informasi*, 1(3), 85-89. <https://doi.org/10.47065/jussi.v1i3.2182>
- Studi, P., Multimedia, T., Jaringan, D. A. N., Teknik, J., Dan, I., & Jakarta, P. N. (2021). KOMPUTER DENGAN MENGGUNAKAN PROCESSOR AMD RYZEN 3 3100 LAPORAN SKRIPSI KOMPUTER DENGAN MENGGUNAKAN PROCESSOR AMD RYZEN 3 3100.
- Zaki, F. N. B., & Lukman, L. (2021). Analisis Perbandingan Quality Of Service (Qos) Pada Video Streaming Dengan Metode PCQ Dan HTB Menggunakan Router Mikrotik. *Respati*, 16(3), 25. <https://doi.org/10.35842/jtir.v16i3.415>
- Zebua, M. (2023). Perancangan Aplikasi Metode Felics Untuk Kompresi Data Video Dengan Ekstensi AVI. 2(1), 145-153.