International Journal of Computational and Experimental Science and ENgineering
(IJCESEN)
Vol. 12-No.1 (2026) pp. 603-611
http://www.ijcesen.com

ISSN: 2149-9144

Research Article

Image Processing, Bitrate Optimization, and Mobile Upload Efficiency

Farooq Abdulla Mulla*

Microsoft, USA
* **Corresponding Author Email:** reachfarooqmulla@gmail.com - **ORCID:** 0000-0002-0047-4880

Article Info:

DOI: 10.22399/ijcesen.4870
Received : 02 December 2025
Revised : 30 January 2026
Accepted : 02 February 2026

Keywords

Mobile Media Optimization,
Bitrate Reduction,
Video Transcoding,
Compression Algorithms,
Adaptive Quality Control

Abstract:

Media transmission efficiency is a growingly important issue in the modern mobile ecosystem as device imaging capabilities grow significantly beyond the network infrastructure limitations. High megapixel sensors and high frame rate video capture on Smartphones create large files of media that often surpass realistic transmission capabilities in cellular and wireless network settings. The use of advanced compression algorithms, smart bitrate control, and adaptive encoding schemes allows for reducing the file size significantly without compromising the visual quality that is not noticeable to human viewers in a wide range of content types. The format conversion, chroma subsampling, adaptive transcoding, and asynchronous processing architecture all minimize bandwidth usage and speed up the completion of media transmission under limited network conditions. Client-side processing deployment is a strategic distribution of computational load to mobile devices, which reduces the requirements of server infrastructure and improves user experience by reducing the transmission time. Quality evaluation systems based on a combination of complementary perceptual measures inform the choice of optimization parameters, which ensures uniform visual fidelity to a wide range of content and heterogeneous device platforms. New codec technologies, such as VVC and AV1, are expected to achieve significant efficiency gains to enable smooth media exchange at scale in resource-limited mobile computing devices.

1. Introduction

The mobile imaging technology has developed exponentially, causing a fundamental disconnect between the content generation capabilities and the network transmission infrastructure. The camera sensors in smartphones have dramatically improved from 5 megapixels in the phones released in 2010 to the present flagship phones with sensors of over 200 megapixels, which is a forty-fold improvement in image resolution compared to just fifteen years ago [1]. This violent sensor evolution is an indication of competition among manufacturers and consumer desire for better quality photography, but cellular and wireless network technologies have not kept pace, and this has resulted in an unequal bandwidth bottleneck in the worldwide media distribution.

The most common infrastructure of media distribution in the world is wireless multimedia sensor networks (WMSNs) and mobile devices, but these systems are subjected to extreme efficiency and energy saving requirements that are fundamentally dissimilar to desktop or server-based systems. Studies on image compression algorithms in wireless multimedia sensor networks have shown that uncompressed images need to be transmitted at 800 kilobits to 12 megabits per frame, depending on the resolution and color depth specifications [1]. This is the unoptimized transmission of raw data with no compression applied to it, which is the theoretical maximum bandwidth of any image content at the given resolution parameters.

Cellular communication networks are faced with two simultaneous limitations that jointly impair the media transmission efficiency: harsh bandwidth constraints and the extreme need to conserve battery power. A detailed examination of bandwidth and power constraints in cellular communication systems shows that current 4G/LTE cellular infrastructure offers an average upload throughput of 5.2 to 14.8 megabits per second in urban metropolitan cellular infrastructure where cell tower density is greatest, and falls precipitously to 1.3 to 3.5 megabits per second in rural and remote deployments where cell tower density and network investment are low [2]. At the same time, battery energy limitations are also limiting, with sustained media transmission using 2.1 to 4.8 watts of power based on device hardware generation and transmission technology used, draining the typical mobile device batteries by 8.5 to 15.3% per hour of sustained use [2]. These two constraints are multiplicative: increasing upload time by increasing bandwidth limits directly increases battery use, and battery saving measures that decrease transmission power indirectly limit the bandwidth that can be achieved by modulation scheme constraints.

The real-world expression of this capacity gap generates significant operational tension across mobile ecosystems. One 48 megapixel image in the standard JPEG format takes 8.2 to 12.5 megabytes of storage space, and 5.6 to 24 seconds to transmit in full under different network conditions, assuming uninterrupted connectivity [1]. This is just a still image, a very basic form of media. Video recording makes this difficult many times over: 4K resolution video capture at 30 frames per second produces 28.3 to 34.5 megabytes per second of uncompressed video data, which is 1.7 to 2.1 gigabytes per minute of uninterrupted recording without any compression applied [2]. These file sizes exceed realistic transmission limits in a typical mobile user environment, especially in a setting with intermittent connectivity, fluctuating signal strength, or data usage restrictions due to cellular service provider policies. Users regularly experience upload errors, timeouts, and high battery consumption when they are trying to share multimedia content without optimization. Optimization techniques have become key infrastructure elements in modern mobile platforms, resolving this capacity gap with algorithmic and architectural advances. A large body of research has shown that compression methods, bitrate optimization techniques, and adaptive encoding methods can be used strategically to reduce the size of media files by 52 to 78% without compromising visual quality within imperceptible limits to human viewers [1]. This enhancement tackles three key operational issues at once: minimizing upload time, which can be several hours to manageable minutes by saving bandwidth, cutting energy use by 45 to 68% by reducing transmission time, and saving bandwidth in line with cellular provider policies and user data plan restrictions [2]. The proportional benefits are experienced in server-side infrastructure, where less storage requirements reduce capital expenditures by 35 to 52% per year and proportional savings in bandwidth delivery costs.

Compression optimization on mobile platforms has unique engineering problems that are not found in desktop or server-based environments. Mobile processing capabilities are grossly limited compared to desktop systems, and mid-range mobile processors provide only 15 to 28% of the same desktop processing throughput and run under stringent thermal and power constraints that require aggressive thermal management and frequency throttling under sustained loads [2]. Battery-aware optimization architecture entails making prudent engineering choices between compression efficiency and energy consumption, which makes processing costs worth the money by proportionate transmission time savings and system-wide energy savings. Network variability requires dynamic optimization techniques that react to changes in bandwidth, signal quality, and connectivity between cellular and wireless network modes [1]. Mobile media optimization can overcome these limitations by using a smart algorithm choice tuned to the capabilities of the device: a progressive processing architecture that allows background tasks to be run, and network-aware parameter tuning that adapts continuously to the varying transmission conditions [2].

2. Media Optimization Fundamentals

2.1 The Challenge of Raw Media

Raw media files produced directly by mobile device sensors include detailed pixel data of full spatial resolution and full color depth specifications. Mobile phone camera sensors generally capture 24 bits of color data (8 bits each in red, green, and blue channels) per pixel of standard color capture, which means that a 12-megapixel image (4096 x 3072 pixels) needs at least 36 megabits to capture all color data, or at least 4.5 megabytes of required storage space [3]. Longer color space specifications, like Adobe RGB or ProPhoto RGB, expand the per-pixel color depth to 32 or 48 bits, correspondingly raising storage requirements. Smartphone photographs are regularly full of large amounts of metadata overhead, such as EXIF data capturing capture parameters, color profiles that specify color space properties, and embedded preview thumbnails to quickly display them, all of which add 200 to 850 kilobytes of overhead to typical photographic captures [3].

Video content generation adds time dimension and frame rate considerations, which generate significantly larger storage needs compared to a static image capture. Compression schemes that examine 1080p video recording at 30 frames per second produce 5.18 megabytes of data per second in uncompressed representation, which needs 311 megabytes per minute of constant recording [4]. This scaling relationship is multiplicative with increases in resolution and frame rate: 4K video at 60 frames per second increases uncompressed data requirements to 20.74 megabytes per second or 1.24 gigabytes per minute [4]. The storage of raw video longer than a few minutes is fundamentally impractical on any consumer device and requires real-time or near-real-time compression when capturing video. This is a fundamental difference between video and still image processing since video compression needs to be done at capture time with a minimum of latency and not as an optimization step after capture.

2.2 Bitrate as Optimization Variable

Measurements of bitrate are used to measure the density of information in media files, in bits per pixel in the case of image data or bits per second in the case of video data. The minimum number of bits per pixel needed to represent an uncompressed image is 24 bits per pixel, which is the minimum number of bits needed to represent the standard RGB color representation, and is the theoretical lowest possible number of bits needed to represent full color information. Good compression algorithms can compress the bitrate by a factor of 4 to 8 to levels of 0.15 to 0.35 bits per pixel at quality ratings of 75 to 85 on standardized perceptual rating scales [3]. This compression performance is 93 to 98% bitrate reduction compared to uncompressed representations, which is an exceptional compression efficiency that can be attained with current codec technology. Video bitrate reduction shows similar efficiency potential by optimizing codecs. H.264 compression of uncompressed 1080p 30fps video at 5.18 megabytes per second is compressed to 0.25 to 0.75 megabytes per second, with quality goals, an 85 to 95% reduction in bitrate [4]. This compression allows full-resolution video to be transmitted over bandwidth-limited networks that could previously only transmit heavily downsampled alternative representations.

Perceptual quality is logarithmic, not linear, with bitrate reduction, which is a key principle of an effective optimization strategy. In logarithmic compression quality curves, the initial bitrate reduction can be aggressive without causing any noticeable quality difference because of the properties of human visual perception, and additional reduction will eventually reach perceptibility limits beyond which quality deterioration will be apparent to the viewer. Experimental studies of codec efficiency have shown that bitrate cuts of 2.0 megabits per second to 1.2 megabits per second cause imperceptible quality loss in standard observer testing, and bitrate cuts of 0.8 to 0.3 megabits per second cause more and more visible compression artifacts in most observer populations [4]. Knowledge of this perceptual property can be used to optimize the bitrate allocation of image regions and video frames by the optimization algorithm, giving priority to those areas of the image and video frame where human observers are most sensitive to compression artifacts, and aggressively compressing those areas where perceptual sensitivity is low.

2.3 Preservation of Resolution by Intelligent Encoding

Traditional resolution reduction algorithms downsample images of original size to smaller pixel representations, reducing the data requirements directly by reducing the number of pixels. The downsampling of an image by 3000 pixels to 2000 pixels reduces the amount of data needed by 75%, but introduces noticeable blur and loss of detail, which is not tolerable to quality-conscious users [3]. Contemporary optimization methods preserve the original resolution entirely, but decrease the information accuracy by selective bitrate optimization, with similar size reduction at significantly higher perceived quality than downsampling options.

The intelligent encoding processes examine the properties of the content and assign bitrate resources based on the perceptual significance instead of even distribution across the image areas. In portrait photographs, bitrate allocation is made to facial regions at 35 to 45% above baseline average bitrate allocation, with background regions allocated 25 to 35% less bitrate than the baseline because human observers show significantly less sensitivity to background artifacts [3][4]. This dynamic bitrate assignment under constant overall limits allows preservation of resolution and attainment of target file size goals by perceptually-optimized compression allocation. Such content-aware allocation implementation will need complex image analysis algorithms that can detect semantic content regions and modify codec parameters to suit them.

3. Core Compression Techniques and Methods

3.1 Format Conversion and Advanced Codecs

The historical standard by which modern compression advances are judged is the baseline JPEG compression, which was developed based on discrete cosine transform (DCT) mathematical underpinnings. JPEG compression can compress photographic material at 8:1 to 12:1 compression ratios due to effective DCT coefficient quantization and entropy coding [5]. The JPEG bitrate needs between 0.25 and 1.2 bits per pixel, depending on the quality settings of 60 to 95 on standardized quality scales [5]. Although JPEG compression has been used for decades and decoders are widely available, the underlying mathematical basis and entropy coding methods inherently limit its performance, allowing newer formats to significantly outperform older standards.

The modern codec alternatives significantly enhance the performance of JPEG with an advanced mathematical basis and better entropy coding schemes. WebP format, a derivative of VP8 video codec principles scaled to still image compression, has the same quality as JPEG with 18 to 35% lower bitrate demands, and bitrates of 0.18 to 0.82 bits per pixel to achieve quality levels similar to JPEG at the same quality settings [5]. This enhancement indicates a better inter-block compression, a more advanced entropy encoding, and a better prediction algorithm that is not available in the old JPEG implementations.

High-Efficiency Image Format (HEIF) technology is based on HEVC video codec foundations, scaled to still image compression, with compression ratios of 15:1 to 20:1 on photographic content when coded at quality levels equivalent to JPEG 85 quality settings [5]. HEIF bitrate needs are 0.15 to 0.68 bits per pixel, which is 22 to 43% better than JPEG at the same perceived quality [5]. The choice of format is a complexity-quality tradeoff: WebP has better compression and wider decoder support on heterogeneous platforms, whereas HEIF has more advanced features, such as support of animation sequences and alpha channels in transparent areas not supported by JPEG architecture. The case of animated image sequences is a special compression case whose optimization needs are fundamentally different from those of a static image. The standard animated GIF format stores animation as a series of individual frames with LZ77 lossless compression of palette-indexed frames, which can be 12 to 35 megabytes in size depending on the length of the animation sequence [5]. WebP animation delivery is equivalent in visual quality to 2.8-8.2 megabytes using inter-frame delta encoding and the use of VP8 codecs, which is 70 to 92% smaller than GIF baseline [5]. HEIF animation formats further cut down on requirements to 2.1 to 6.5 megabytes, providing 82 to 94% of the reduction over GIF storage, and are especially useful in bandwidth-restricted distribution cases [5].

3.2 Chroma Subsampling and Color Space Optimization

Human visual perception has asymmetric sensitivity to various visual dimensions, and spatial resolution in color perception is significantly lower than the sensitivity of luminance (brightness) perception. Chroma subsampling takes advantage of this perceptual property by sacrificing the spatial resolution of color information but maintaining the full luminance resolution. Chroma subsampling 4:2:0 chroma subsampling halves the spatial resolution of color planes to the luminance plane resolution, halving the amount of color data needed, but causing no perceptible quality loss on natural images [6]. Application of 4:2:0 subsampling in baseline JPEG compression adds 38 to 48% of overall compression ratio improvement, showing the significant role of this perceptual optimization method [6]. Advanced subsampling plans use variable subsampling patterns depending on content characteristics, in contrast to uniform methods. Photographs with smooth transitions in skin tones and smooth color changes in the portrait are better served by 4:2:2 subsampling (one-half color resolution in horizontal direction, full resolution in vertical direction), which has less compression advantage than aggressive 4:2:0 but maintains important color fidelity to facial reproduction [6]. Landscape images with large areas of uniform color are good candidates for aggressive 4:2:0 application with little perceived quality effects. Adaptive subsampling selection algorithms examine the properties of source content and automatically choose an optimal subsampling pattern for a given image content, with average bitrate gains of 15 to 28% over uniform subsampling application [6]. This content-based method is the most efficient in compression because it does not use the same strategies on all content but instead uses subsampling intensity that is matched to the real content properties.

Progressive image encoding sends a low-quality representation of the image and then sends successive refinement data to improve the visual quality progressively, allowing the partial display of the image immediately during transmission. The implementation of progressive JPEG incurs 18 to 26% more encoding overhead than baseline sequential JPEG because of the use of multiple encoding passes and improved error correction structures, but can display images perceptibly in 1.2 to 2.8 seconds after the transmission has started [6]. This progressive encoding significantly enhances user experience measures over the baseline sequential encoding, where the viewers are subjected to blank screens or slow image display during transmission.

3.3 Video Transcoding and Codec Selection

Video transcoding is the process of re-encoding video files to optimized target formats to apply modern codec technology to older video files. The application of H.264 video codec can compress video content by 50:1 to 200:1, based on quality goals and scene properties, and the compression ratio can vary significantly based on content-dependent encoding efficiency [6]. More recent H.265/HEVC codec implementation achieves 40-50% bitrate savings over H.264 at the same quality perception, allowing 1080p quality video delivery at bitrates that previously had to be used to deliver 720p content, significantly enhancing delivery efficiency [6]. Rate control algorithms are used to optimize the allocation of bitrate between temporal video sequences, and to change the allocation between frames based on the complexity of the frame content and temporal properties. Constant bitrate (CBR) mode uses a constant bitrate during the encoding duration, and assigns the same amount of data to each video second, irrespective of the complexity of the scene, leading to uniform but possibly inefficient bitrate allocation. Variable bitrate (VBR) mode focuses bitrate distribution on temporally active scenes with significant motion or fine detail, and less on stationary scenes with little temporal variation, enhancing the overall compression efficiency [6]. Two-pass VBR encoding is 12 to 24% more efficient in bitrate than single-pass constant bitrate methods by content analysis during the first encoding pass, and then optimized allocation during the second pass [6]. This two-pass algorithm is a radical improvement in the efficiency of the encoding process, allowing the codec to make allocation decisions based on full analysis of the video, as opposed to real-time single-pass encoding decisions.

4. Mobile Application Implementation Strategies

4.1 Client-Side Processing Architecture

The effectiveness of client-side processing architecture in enhancing the efficiency of media transmission is evidenced by comparative measurements of video conferencing application performance with various types of network backhaul [7]. The processing throughput of mobile devices ranges between 15 and 45 gigabits per second, depending on the processor generation, the number of cores, and the hardware acceleration features it has [7]. Mid-range mobile processors are characterized by performance differentiation according to the intensity of optimization: JPEG recompression processing takes 45 to 185 milliseconds per 4-megapixel photograph, and WebP encoding operations take 125 to 380 milliseconds on the same source material [7]. Video transcoding of 1-minute source material to H.264 takes 1.8 to 4.2 seconds with hardware video encoding acceleration, and up to 22 to 58 seconds on processors without specialized video hardware support [7]. Thermal management becomes a major issue when there is a prolonged client-side processing operation. Constant media encoding at peak processor power produces thermal power of 8.2 to 14.5 watts on mobile platforms, elevating the surface temperature of devices 12 to 18 degrees Celsius above ambient temperatures due to sustained heat dissipation [7]. Long processing times can cause thermal throttling, slowing down the processor frequency by 20 to 35% when device temperature limits are reached, and increasing processing time proportionally to performance loss, and performance gains are canceled [7]. Strategic scheduling spreads processing over longer durations or halts processing under high device temperature conditions, so that thermal constraints do not undermine processing efficiency. Advanced thermal management software continuously measures the temperature of the device and changes the processor scheduling to balance between compression efficiency and thermal limits [7].

4.2 Asynchronous Optimization Pipelines

The effectiveness of a queue-based optimization architecture in the sequential processing of multiple media items without blocking user interface operations is demonstrated by efficient asynchronous federated evaluation research with strategy similarity awareness [8]. The typical throughput of normal mobile platform processing is between 8.5 and 16.3 megabytes per second, depending on the nature of the content and the level of optimization options applied [8]. Processing queue depth control keeps 3-12 items at parallel processing conditions with throughput rates rising linearly to thermal or power constraints that initiate throttling controls to reduce processing efficiency [8]. This queue depth control trades off responsive user experience with available computational resources, avoiding excessive queue buildup that would slow optimization completion or use up too much memory.

Network-aware pipeline adaptation adjusts the intensity of optimization according to the available bandwidth conditions, which is the basic innovation in mobile optimization architecture. Network conditions with high speeds (more than 12 megabits per second upload capacity) activate relaxed optimization with retention targets of 85 to 92% of original quality, and quality preservation is prioritized when bandwidth is available [8].

Standard optimization profiles are triggered by medium-speed conditions of 3 to 8 megabits per second, which preserve 70 to 82% of original quality, trade quality versus transmission time [8]. Slow speeds under 2 megabits per second enable aggressive optimization settings that leave 45 to 62% of original quality, and focus on the time to complete an upload rather than the quality of the upload [8]. This dynamic adaptation guarantees the best user experience in a wide range of network conditions, automatically varying the intensity of optimization to suit the available transmission capacity.

4.3 Tiered Quality Strategies

Multi-tier quality models define different optimization profiles that can be used in different applications and network conditions with particular quality and bandwidth trade-offs. Standard quality tier applies optimization to achieve 55 to 68% size reduction, which is appropriate in the common cellular network environment where bandwidth is constrained [8]. Tier 1 limits reduce to 18 to 32%, which is suitable for WiFi network delivery or users who value quality preservation over transmission speed [8]. Archive quality tier applies minimum optimization of 8 to 15% reduction to local device storage or high-end cloud storage services, and focuses on quality retention rather than bandwidth efficiency [8].
Automatic tier selection systems consider various factors, such as the availability of network bandwidth, the battery charge of the device, the amount of storage space left, and the user preference, to suggest or automatically set a suitable quality tier [8]. The network conditions, battery state, user preference history, and device thermal state are weighted by 40%, 25%, 20%, and 15%, respectively, and the resulting tier recommendations are based on cumulative optimization goals [8]. Manual override features allow user tier selection regardless of automatic suggestions of specialized use cases where users have particular quality or transmission duration needs that are not the same as algorithmic suggestions [8].

5. Best Practices and Future Considerations

5.1 Quality Metrics and Performance Assessment

A number of complementary quality measurement methodologies are available, which provide different perspectives on the quality of compressed media, and each of the metrics represents a different aspect of quality. Peak Signal-to-Noise Ratio (PSNR) is a measurement of the difference between the magnitude of both original and compressed images in decibels, and the difference is typically acceptable between 28 and 38 decibels on subjective quality scales [9]. The Structural Similarity Index Metric (SSIM) measurement is used to measure perceptual similarity based on luminance, contrast, and structure preservation, with values of 0.85 to 0.95 indicating no quality degradation [9].
The visual quality assessment methodology uses a variety of metrics that cover different dimensions of quality using complementary measurement strategies. Perceptual quality measures, such as Video Multimethod Assessment Fusion (VMAF), are a combination of various assessment methods, with correlation coefficients of 0.92 to 0.96 with subjective quality ratings of various observer groups [9]. Mean Opinion Score (MOS) assessment methodology is a survey of representative populations of observers rating quality on a scale of 1 to 5, where a score of 3.8to 4.5 represents acceptable quality with no visible compression artifacts [9]. Constant tracking of quality indicators during optimization pipelines helps to identify the misconfiguration of parameters or malfunction of algorithms early enough before distributing suboptimal content to final users [9].

5.2 Adaptive Control of the Model

The use of machine learning technology in video compression shows significant efficiency gains compared to traditional manual parameter tuning methods. The neural network models that are trained on a variety of image and video content datasets estimate the best compression parameters, reducing the bitrate by 8 to 15% over traditional algorithms with the same quality measures [10]. The size of training datasets is usually 50,000 to 200,000 reference images of various types, such as portraits, landscapes, action scenes, low-light photography, and synthetic content [10]. Content-aware parameter selection examines the properties of source material and chooses compression parameters that are optimal for a particular type of content. Facial recognition algorithms identify portrait photographs, which results in quality allocation whereby facial parts are allocated at bitrate premiums of 28 to 45% over baseline and background parts are reduced by 22 to 38% [10]. Motion detection determines high-motion sequences that need bitrate increments of 18 to 32% over the static sequence baseline [10]. Scene complexity analysis modulates the compression parameters according to the density of the edges and the variety of the color palette, and the complex scenes are allocated a bitrate 15 to 25% higher than the simple scene baseline [10].

5.3 Future Evolution and Adoption of Codecs

New codec standards, such as AV1 video codec and VVC (Versatile Video Coding), are expected to be more efficient in compression than the existing H.265 baseline due to better mathematical underpinnings. The AV1 codec development is 25-35% lower in bitrate than H.265 in a variety of video content types [10]. The implementation of the VVC standard aims at 30-40% bitrate reduction over H.265, and commercial implementation is expected to start in the 2025 to 2027 timeframe when hardware acceleration support is more common [10]. Availability of hardware acceleration is a key adoption consideration, and it is currently being implemented at 15 to 25% of the installed base of mobile devices, which is a significant obstacle to adoption in the near term [10]. Backward compatibility plans allow a gradual format migration without the need to leave support of legacy devices. Dual-format transmission supports WebP and JPEG versions to heterogeneous device groups, with selective delivery of advanced formats as device market penetration reaches 60 to 75% targets, to ensure wide compatibility in transitional phases [10]. The format upgrade paths between 3 and 5 year deployment windows support the lifecycle properties of devices and user upgrade cycles, noting that the installed base of devices changes at a relatively slow rate relative to the rate of codec capability development [10].

Table 1: *Network and Power Constraints in Mobile Media Transmission [1,2]*

Network/Power Parameter	Performance Specification
4G/LTE Urban Upload Speed	5.2 to 14.8 Megabits per second
4G/LTE Rural Upload Speed	1.3 to 3.5 Megabits per second
Battery Depletion per Hour	8.5 to 15.3%
48-Megapixel Photo File Size	8.2 to 12.5 Megabytes
Photo Upload Duration	5.6 to 24 Seconds
4K Video Data Generation	28.3 to 34.5 Megabytes per second
Compression Size Reduction	52 to 78%
Energy Savings from Optimization	45 to 68%

Table 2: *Codec Compression Performance and Format Comparison [5,6]*

Codec Format	Compression Metric
JPEG Bitrate Requirement	0.25 to 1.2 Bits per Pixel
WebP Bitrate Improvement	18 to 35% Lower than JPEG
WebP Bitrate Range	0.18 to 0.82 Bits per Pixel
HEIF Bitrate Requirement	0.15 to 0.68 Bits per Pixel
Animated GIF Size (10-second)	12 to 35 Megabytes
HEIF Animation Size	2.1 to 6.5 Megabytes
H.265/HEVC Bitrate Advantage	40 to 50% versus H.264

Table 3: *Mobile Processing Capabilities and Quality Tier Specifications [7,8]*

Processing Parameter	Specification Range
Mobile Processor Throughput	15 to 45 Gigabytes per second
JPEG Recompression Time (4MB)	45 to 185 Milliseconds
WebP Encoding Time (4MB)	125 to 380 Milliseconds
H.264 Transcoding (with acceleration)	1.8 to 4.2 Seconds per minute
H.264 Transcoding (without acceleration)	22 to 58 Seconds per minute
Processing Queue Throughput	8.5 to 16.3 Megabytes per second
Standard Quality Tier Reduction	55 to 68%
High-Quality Tier Reduction	18 to 32%
Archive Tier Reduction	8 to 15%
Optimal Queue Depth	3 to 12 Concurrent Items

Table 4: *Quality Metrics and Emerging Codec Technologies [9,10]*

Quality/Technology Parameter	Measurement/Capability
Peak Signal-to-Noise Ratio (PSNR)	28 to 38 Decibels
Structural Similarity Index Metric (SSIM)	0.85 to 0.95

Visual Quality Assessment (VMAF) Correlation	0.92 to 0.96
Mean Opinion Score (5-point scale)	3.8 to 4.5
ML Training Dataset Size	50,000 to 200,000 Reference Images
AV1 Codec Bitrate Reduction	25 to 35% vs. H.265
VVC Standard Bitrate Improvement	30 to 40% vs. H.265
Hardware Acceleration Coverage	15 to 25% of Mobile Devices
Format Migration Timeline	3 to 5 Years

6. Conclusions

The optimization of mobile media must be done with a balanced approach to various competing goals, such as minimization of file size, maintenance of visual quality, conservation of computational resources, and battery energy efficiency in a wide range of deployment conditions. The addition of modern compression codecs like WebP, HEIF, H.265, and new standards allows reducing the file size by a dramatic margin without any noticeable quality loss when the implementation parameters are set correctly and tested against standardized quality metrics in a systematic way. The architecture of client-side optimization allocates processing load in a strategic manner, minimizing the bandwidth needs of the server and supporting network variability with adaptive parameter selection in response to the available transmission bandwidth and device capabilities. Quality tier implementation offers deployment flexibility to support a wide range of applications, including cellular transmission applications to high-end cloud storage preservation applications. The ongoing development of codec technology, machine learning-based parameter optimization, and hardware acceleration integration will guarantee a significant reduction in operational costs and an increase in user satisfaction rates. Organizations that implement end-to-end optimization pipelines that cover format selection, content-aware parameter tuning, and network-aware processing strategies will see a significant decrease in operational costs and an increase in user satisfaction rates. The future codec adoption directions based on the new technology standards must be carefully managed with compatibility, gradual platform migration, heterogeneous device ecosystems, and long device lifecycle features common in the global mobile computing implementations.

Author Statements:

- **Ethical approval:** The conducted research is not related to either human or animal use.
- **Conflict of interest:** The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper

- **Acknowledgement:** The authors declare that they have nobody or no-company to acknowledge.
- **Author contributions:** The authors declare that they have equal right on this paper.
- **Funding information:** The authors declare that there is no funding to be acknowledged.

- **Data availability statement:** The data that support the findings of this study are available on request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.
- **Use of AI Tools:** The author(s) declare that no generative AI or AI-assisted technologies were used in the writing process of this manuscript.

References

[1] Hanaa ZainEldin et al., "Image compression algorithms in wireless multimedia sensor networks: A survey", ScienceDirect, 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2090447914001567

[2] Ikechukwu H. Ezeh and Isaac A. Ezenugu, "Challenges of Bandwidth and Power Limitations in Cellular Communication: A Review," IOSR-JMCA, 2020. [Online]. Available: https://www.iosrjournals.org/iosr-jmca/papers/Vol7-Issue4/Series-1/A07040111.pdf

[3] Shuo Han et al., "High-Quality Image Compression Algorithm Design Based on Unsupervised Learning," MDPI, 2024. [Online]. Available: https://www.mdpi.com/1424-8220/24/20/6503

[4] Evgeny Belyaev, "An Efficient Compressive Sensed Video Codec with Inter-Frame Decoding and Low-Complexity Intra-Frame Encoding," MDPI, 2023. [Online]. Available: https://www.mdpi.com/1424-8220/23/3/1368

[5] Tomáš Chlubna and Pavel Zemčík, "Comparative survey of image compression methods across different pixel formats and bit depths", Springer Nature, Sep. 2025. [Online]. Available: https://link.springer.com/article/10.1007/s11760-025-04579-6

[6] Grzegorz Pastuszak, "Subsampling of 3D Pixel Blocks as a Video Compression Method for Analog Transmission," MDPI, 2023. [Online]. Available: https://www.mdpi.com/2079-9292/12/12/2641

[7] Rohan Kumar et al., "Comparison of Popular Video Conferencing Apps Using Client-side Measurements on Different Backhaul Networks," ACM, 2022. [Online]. Available: https://dl.acm.org/doi/epdf/10.1145/3492866.3557734
[8] Shaowen Qin et al., "Efficient Asynchronous Federated Evaluation with Strategy Similarity Awareness for Intent-Based Networking in Industrial Internet of Things", arXiv, Nov. 2025. [Online]. Available: https://arxiv.org/pdf/2512.20627
[9] Krzysztof Okarma and Mateusz Kopytek, "Improved Combined Metric for Automatic Quality Assessment of Stitched Images," MDPI, 2022. [Online]. Available: https://www.mdpi.com/2076-3417/12/20/10284
[10] Lesia Mochurad, "A Comparison of Machine Learning-Based and Conventional Technologies for Video Compression," MDPI, 2024. [Online]. Available: https://www.mdpi.com/2227-7080/12/4/52