

BAB 1 PENDAHULUAN

1.1 Latar Belakang

Voice merupakan media komunikasi verbal yang digunakan oleh manusia. *Voice* dihasilkan oleh interaksi pita suara, sistem pernafasan, dan saluran vokal yang menghasilkan bunyi yang berbeda-beda [1]. Walaupun terkadang terdapat kemiripan, namun secara parameter dari bentuk *voice* tersebut berbeda-beda (jenis suara, nada suara, dan volume) [2]. Perbedaan yang dihasilkan oleh bentuk *voice* saat proses perekaman menyebabkan beberapa informasi tidak dapat diproses dengan sempurna. Hal ini tentunya akan mempengaruhi kualitas *voice*, sehingga informasi tidak dapat tersampaikan dengan baik.

Munculnya teknologi 5G atau bahkan 6G, membuka lebih jauh kemungkinan untuk inovasi berbasis *voice*, *voice* spasial melalui *headphone* akan menjadi langkah besar menuju komunikasi realistis yang memungkinkan pengguna mendengar audio seperti dikehidupan nyata [3]. Untuk menghasilkan *voice* spasial dapat dilakukan dengan beberapa cara, salah satunya dengan perekaman secara *binaural*. Perekaman secara *binaural* dapat menghasilkan pengalaman mendengarkan yang lebih realistis dan *imersif* [4]. Hal ini dikarenakan pada *binaural* memanfaatkan fitur *psikoakustik* pendengaran manusia untuk menghasilkan suara 3D [5]. Namun, untuk menghasilkan suara yang realistis dan *imersif*, maka dibutuhkan metode pengkodean yang efisien untuk memaksimalkan hasil suara [6].

Pengkodean suara memiliki tahapan yang disebut *codec*. *Codec* merupakan singkatan dari *compresi-decompresi* yang mengindikasikan bahwa suara telah mengalami kompresi atau pemampatan. Penggunaan *codec* pada *voice* bertujuan untuk mengatasi tantangan komunikasi suara *real-time* yang berfokus pada efisiensi, latensi rendah, dan menghasilkan kualitas yang baik. Hal ini berbeda dari audio *codec* yang lebih terfokus pada pengalaman mendengar [7].

Voice codec dirancang untuk mencapai efisiensi maksimal dengan mengompresi data suara sehingga memungkinkan pengiriman melalui jaringan dengan penggunaan *bandwidth* minimal [8]. Selain itu, *voice codec* bekerja untuk mengurangi latensi sebanyak mungkin, memastikan bahwa suara dapat dikirimkan dan diterima *real-time* tanpa gangguan, serta memberikan pengalaman komunikasi yang responsif. Meskipun fokus pada efisiensi dan latensi rendah, *voice codec* tetap menghasilkan kualitas suara yang baik, mengidentifikasi elemen-elemen penting suara yang harus dipertahankan untuk memastikan bahwa informasi dan emosi dalam percakapan tetap terjaga, walaupun dengan pengurangan data [9].

Pengurangan data ini terjadi dikarenakan menggunakan metode kompresi *lossy*. Metode *lossy* adalah teknik kompresi data dengan cara menghilangkan sebagian data dengan tetap mempertahankan rasio dan perkiraan data aslinya,

dimana data hasil dekompresi mengalami perubahan dari data aslinya [10]. Perubahan pada hasil dekompresi pada metode *lossy* dapat mengakibatkan berkurangnya daya tahan data, sehingga rentan terhadap distorsi yang membuat kejelasan informasi dalam sinyal tersebut menjadi berkurang bahkan hilang [11]. Proses kompresi secara *lossy* dapat menurunkan kualitas dari data yang dikompresi, sehingga akan mempengaruhi pengalaman yang dirasakan oleh pendengar, namun belum ada penelitian yang menguji bagaimana pengaruh perubahan hasil kompresi dengan metode *lossy* berpengaruh terhadap *audio binaural* yang menggunakan *voice codec*.

Voice Codec yang banyak digunakan pada saat sekarang ini adalah Opus, hal ini dikarenakan Opus sebagai *codec open source*, sehingga banyak diaplikasikan pada media sosial, *teleconference*, dan *game online* [7]. Selain itu, Opus adalah *codec* yang efisien dalam kompresi dan transmisi data secara *real-time* [12]. Opus dapat menghasilkan kualitas *voice* yang baik pada *bitrate* yang rendah, serta dapat mengkodekan *voice* pada *bandwidth* yang dibutuhkan sehingga *voice* dapat ditransmisikan secara efisien. Selain itu, Opus dirancang dengan penundaan rendah yang berarti suara dapat dikirim secara *real-time*. Hal ini sangat penting dalam aplikasi VoIP (*Voice over Internet Protocol*) [13].

Penelitian yang dilakukan oleh Pasi Ojala pada tahun 2010 yang berjudul “*Parametric binaural audio coding*”, didapat hasil bahwa *voice* yang direkam secara *binaural* menggunakan *codec* AAC dapat menghasilkan suara yang lebih baik dari AMR-WB [14]. Namun, *codec* AAC merupakan pengkodean audio yang tidak spesifik untuk digunakan pada *voice*, oleh karena itu dibutuhkan penelitian lebih lanjut terhadap *voice binaural* menggunakan pengkodean audio yang spesifik untuk digunakan pada *voice*. Pada penelitian yang dilakukan oleh Jean-Marc Valin pada tahun 2013 “*High-quality, low-delay music coding in the Opus codec*” tentang evaluasi kualitas *audio* pada *codec* Opus, AAC, dan vorbis, didapatkan hasil bahwa *codec* Opus lebih baik dari *codec* AAC dan vorbis [15]. Hal ini menunjukkan bahwa Opus sebagai *codec voice* dapat menghasilkan kualitas yang lebih baik dari AAC. Namun, belum ada penelitian yang dilakukan pada *voice* yang direkam secara *binaural* menggunakan *codec* Opus.

Oleh karena itu penulis ingin menguji kualitas *voice* yang memiliki efek *binaural* pada kompresi menggunakan *codec* Opus dengan judul penelitian “*Analisis Kualitas Binaural Audio pada Codec Opus*”. Penelitian ini dilakukan secara subjektif dengan mengikuti standar ITU-R BS.1116-3, objektif dengan mengikuti standar ITU-R BS.1387-1 dan *Signal to Noise Ratio* (SNR). Format *codec lossy* yang akan diuji adalah Opus dan sampel yang diuji merupakan *audio voice* yang memiliki efek *binaural*.

1.2 Rumusan Masalah

Rumusan masalah pada penelitian ini adalah bagaimana kualitas *voice* pada *binaural* audio yang dihasilkan dari variasi *bitrate* dari proses kompresi pada *codec* Opus.

1.3 Tujuan Penelitian

Tujuan dari penelitian ini adalah untuk mengetahui kualitas *voice* pada *binaural* audio dengan variasi *bitrate* yang dihasilkan dari proses kompresi *lossy* pada *codec* Opus, dengan menggunakan *Subjective Difference Grade* (SDG), *Objective Difference Grade* (ODG) dan *Signal-to-Noise Ratio* (SNR) sebagai parameter acuan untuk melihat kualitasnya.

1.4 Batasan Masalah

Batasan – batasan yang diperlukan bagi penelitian ini :

1. Pengujian dilakukan menggunakan perangkat lunak yang telah tersedia.
2. Pengujian difokuskan pada pengujian kualitas *voice* dengan efek *binaural*.
3. Sampel audio asli yang digunakan adalah sampel suara dengan metode perekaman secara *binaural* dan terdiri dari 6 sampel.
4. Penelitian menggunakan metode kompresi *lossy* dengan *codec* Opus
5. Penilaian penilaian kualitas audio dilakukan dengan metode *subjective* terhadap 20 orang pendengar.
6. Pendengar akan mendengarkan audio uji melalui *headphones*.
7. Penilaian yang dilakukan menggunakan parameter *subjektive* untuk menghasilkan nilai *Subjective Difference Grade* (SDG), *objektive* untuk nilai *Objective Difference Grade* (ODG), dan SNR.

1.5 Manfaat Penelitian

Studi ini diharapkan dapat memberi pengetahuan mengenai pengaruh kompresi audio yang dihasilkan pada suara audio *binaural* dengan metode *lossy* sehingga nantinya dapat digunakan untuk aplikasi yang sesuai.

1.6 Sistematika Penulisan

Pada tugas akhir ini, disusun dalam beberapa bab dengan sistematika tertentu, sistematika laporan ini adalah sebagai berikut:

BAB I PENDAHULUAN

Bab ini membahas tentang latar belakang dari tugas akhir, rumusan masalah, tujuan penelitian, batasan masalah, manfaat penelitian dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Bab ini membahas tentang landasan teori yang berkaitan dengan penulisan tugas akhir ini.

BAB III METODOLOGI PENELITIAN

Bab ini berisikan penjelasan mengenai metodologi penelitian yang digunakan berupa tinjauan sistem, prinsip kerja sistem, gambaran sistem, perancangan sistem, dan teknik pengujian yang dilakukan pada penelitian ini.

BAB IV HASIL DAN PEMBAHASAN

Pada bab ini akan dibahas mengenai hasil yang didapatkan selama penelitian.

BAB V PENUTUP

Penutup menuturkan mengenai kesimpulan yang didapatkan dari penelitian yang telah dilakukan, pada bab ini juga akan diberikan saran mengenai penelitian dan apa yang baik dilakukan untuk penelitian selanjutnya.

