



## **Penerapan Metode Two ided Side Match Untuk Penyisipan Gambar Pada File Audio**

**Fiderman Lase**

Program Studi Teknik Informatika, Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Budi Darma,  
Jalan Sisingamaganraja No. 338, Medan, Sumatera Utara, Indonesia  
Email: [fiderlase12@gmail.com](mailto:fiderlase12@gmail.com)

**Abstrak**—Keamanan data pesan pada saat ini sangat penting dilakukan teknik pengamanan, karena peningkatan pengguna teknologi media komunikasi yang semakin meningkat sehingga bisa saja pihak lain dapat mengetahui isi pesan. Teknik steganografi merupakan teknik yang dapat digunakan untuk menyisipkan pesan gambar pada file audio. Metode Two Sided Side Match adalah salah satu metode Steganografi yang menggunakan dua sisi nilai pixel, yaitu sisi atas dan sisi kanan atas. Untuk menentukan seberapa banyak biner gambar yang akan disisipkan kedalam sebuah pixel. Metode Steganografi ini juga merupakan seni menyembunyikan pesan rahasia dengan suatu teknik yang tidak menimbulkan kecurigaan pihak lain, sehingga pesan hanya dapat diketahui oleh sipengirim dan sipenerima pesan yang dibangun dengan bahasa pemrograman Microsoft Visual Studio 2010. Pemanfaatan teknik steganografi dapat meningkatkan kesulitan pihak lain untuk mengetahui keberadaan pesan penting atau pesan rahasia karena teknik steganografi melakukan penyembunyian pesan penting pada objek tertentu. Dalam teknik steganografi ini sangat efisien digunakan untuk menyembunyikan pesan tanpa mengurangi kualitas objek pesan yang disisipkan.

**Kata Kunci:** Steganografi; Gambar; Audio; Two Sided Side Match

**Abstract**— Message data security at this time is very important to do security techniques, because of the increasing number of users of communication media technology that is increasing so that other parties can find out the contents of the message. Steganography technique is a technique that can be used to insert picture messages in audio files. The Two Sided Side Match method is a Steganography method that uses two sides of the pixel value, namely the top side and the top right side. To determine how many binary images to insert into a pixel. This Steganography method is also the art of hiding secret messages with a technique that does not arouse the suspicion of the other party, so that the message can only be known by the sender and recipient of the message which was built using the Microsoft Visual Studio 2010 programming language. important messages or secret messages because the steganography technique hides important messages on certain objects. In this very efficient steganography technique, it is used to hide messages without reducing the quality of the inserted message object.

**Keywords:** Steganografi; Picture; Audio; Two Sided Side Match

### **1. PENDAHULUAN**

Perkembangan teknologi pada saat ini yang semakin berkembang pesat. Kebutuhan untuk mengirimkan informasi dari satu ke tempat lain menjadi sangat mudah dilakukan pada masa sekarang ini. Hal ini menyebabkan proses pengiriman informasi menjadi cepat. Jenis informasi yang bisa dikirimkan juga beranekaragam, seperti gambar, audio, pesan, dan video. Dengan perkembangan teknologi saat ini, audio, gambar, pesan, dan video dapat disembunyikan dengan cara steganografi yang berbasis komputer.

Steganografi adalah teknik penyisipan pesan rahasia kedalam sebuah informasi seperti gambar dan audio dengan menerapkan metode *Two sided side match*. Teknik steganografi bertujuan untuk mengamankan sebuah file pesan sehingga tidak dapat menimbulkan kecurigaan pada pihak lainnya.

Berbicara mengenai penyisipan gambar pada file audio sering mengalami masalah, sering terjadinya pencurian data, penyadapan data, dan pengubahan data oleh pihak yang tidak bertanggung jawab. Oleh karena itu, diperlukan cara untuk mengatasi masalah tersebut supaya dalam melakukan penerapan tidak ada pihak lain yang mengetahui.

Berdasarkan penelitian yang dilakukan oleh Nurul Khairina Tahun 2016 yang dipublikasikan pada *jurnal CESS Journal Of Computer Engineering, System And Science* volume 1 nomor 2 yang berjudul “Analisis Steganografi Metode Sided Side Match” menyimpulkan bahwa dari percobaan penyisipan 5 huruf yang dilakukan pada citra asli yang berukuran 5x5, pada metode Two Sided Side Match hanya 2 huruf yang dapat kembali dengan normal pada saat ekstraksi. Hal ini bisa terjadi karena kemungkinan adanya perbedaan pembulatan nilai desimal saat proses penyisipan pesan dan saat proses ekstraksi pesan sehingga banyak bilangan biner yang tidak dapat diperoleh kembali [1].

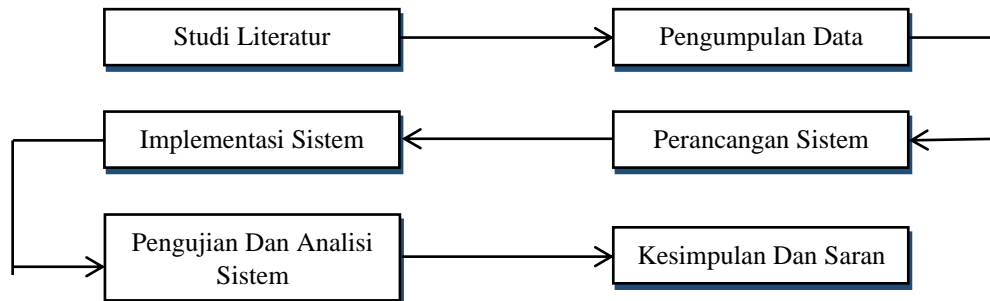
Pada penelitian sebelumnya yang dilakukan oleh G. Swain Dkk Tahun 2013 yang dipublikasikan pada jurnal *CSI Transaction on ITC* volume 1 nomor 2 yang berjudul “*Steganography using two sided, three sided, and four sided side match methods*” menyimpulkan bahwa jumlah bit yang tertanam dalam piksel target ditentukan tergantung pada korelasi piksel target dengan piksel tetangganya. Kapasitas persembunyian dari metode yang diusulkan relatif lebih baik. Setelah informasi disematkan, perubahan kualitas gambar tidak terlihat. Nilai PSNR yang diamati juga baik. Distorsi lebih rendah dalam metode pencocokkan sisi empat sisi dibandingkan dengan metode pencocokkan sisi dua dan tiga sisi. Metode-metode ini dapat digunakan dalam berbagai kesempatan dimana komunikasi berdifat rahasia [2].

Penelitian yang sama pernah dilakukan oleh A. Ratnasari, F. Dwiyanto pada tahun 2020 yang berjudul Metode Steganografi Citra Digital (Ratnasari And Dwiyanto 2020). Metode *Two Sided Side Match* menyembunyikan pesan pada media digital dengan mempertimbangkan nilai dari dua *piksel* tetangganya yaitu nilai *piksel* sisi atas, dan sisi kanan atas. Sementara itu metode *four sided side match* yang merupakan pengembangan dari metode *Two Sided Side Match* mempertimbangkan nilai dari empat piksel tetangganya, yaitu nilai *piksel* sisi atas dan sisi kiri [3].

Penelitian yang dilakukan oleh Swain and Lenka 2013 menganalisis metode *Two Sided side*, *Three Sided Side*, dan *Four Sided Side Macth*. Masing-masing metode ini memprediksi berapa banyak pesan yang mampu ditambah pada sebuah *pixel* tetangganya. Hasil penelitian ini menunjukkan bahwa metode-metode tersebut memiliki kapasitas menyembunyikan pesan yang besar dan memiliki tingkat keamanan pesan yang cukup baik [4].

## 2. METODOLOGI PENELITIAN

### 2.1 Kerangka Kerja Penelitian



**Gambar 1.** Kerangka Kerja Penelitian

Berdasarkan tahap penelitian yang telah digambarkan diatas, maka dapat diuraikan pembahasan masing-masing tahap dalam penelitian adalah sebagai berikut.

- Study Literatur**  
Pada tahapan ini peneliti melakukan apa yang disebut dengan kajian pustaka, yaitu mempelajari buku-buku referensi dan hasil penelitian sejenis sebelumnya yang pernah dilakukan oleh orang lain.
- Pengumpulan data**  
Teknik pengumpulan data adalah cara yang digunakan untuk mengumpulkan informasi atau fakta-fakta yang ada di lapangan sesuai dengan objek yang akan diteliti.
- Perancangan Sistem**  
Pada tahap ini penulis menggunakan bahasa pemrograman Visual Studio 2008 dengan aplikasi *MYSQL* untuk merancang atau mendesain sistem yang berupa langkah-langkah operasi dalam proses pengolahan data dan prosedur-prosedur untuk mendukung sistem dan juga untuk merancang *flowchart*, *input* dan *output*.
- Implementasi Sistem**  
Tahapan implementasi adalah tahap penerapan dan sekaligus pengujian bagi sistem berdasarkan hasil analisa dan perancangan yang telah dilakukan.
- Hasil Pengujian Dan Analisis Sistem**  
Hasil pengujian dan analisis sistem yang diperoleh peneliti yaitu dari hasil perhitungan dan perancangan yang dilakukan, dari hasil ini dapat diketahui hasil yang didapatkan apakah sesuai dengan maksud dan tujuan penelitian dalam menyelesaikan permasalahan.
- Kesimpulan Dan Saran**  
Tahapan ini berisi kesimpulan mengenai hasil evaluasi dari seluruh kegiatan yang dilakukan dalam melakukan penelitian terhadap penyisipan gambar pada file audio melalui metode *bayes*. Pada tahap ini juga diberikan saran-saran pengembangan dan pengelolaan sistem lebih lanjut.

### 2.2 Steganografi

Steganografi adalah ilmu seni dan seni menyembunyikan pesan rahasia didalam pesan lain sehingga keberadaan pesan rahasia tersebut tidak dapat diketahui. Steganografi membutuhkan dua properti, yaitu media penampung dan pesan rahasia. Media penampung yang umum digunakan adalah gambar, suara, video atau teks. Pesan yang disembunyikan dapat berupa sebuah artikel, kode program, atau pesan lain. Proses penyisipan pesan kedalam media *covertext* dinamakan *encoding*, sedangkan ekstaksi pesan dari *stegotext* dinamakan *decoding*.

### 2.3 Metode Two Sided Match

Metode *Two Sided Side Match* menggunakan dua nilai *pixel* tetangganya untuk memprediksi beberapa banyak pesan yang dapat disisipkan pada sebuah *pixel*. Metode ini menggunakan nilai *pixel* tetangganya pada sisi atas dan sisi kanan atas.

#### 2.3.1 Penyisipan Metode Two Sided Side Match

Dalam menyisipkan pesan dengan metode *Two Sided Side Match* ini, harus terlebih dahulu menghitung perbandingan nilai dantar nilai *pixel* tetangga dengan rumus seperti persamaan (1) di bawah ini.

$$d = (g_u + g_{ur}) / 2 - g_x \tag{1}$$



Pada sebuah citra, terdapat pixel target  $P_x$  dengan nilai 72, dan memiliki *pixel* tetangga  $P_u$ , dan  $P_{ur}$  dengan nilai masing-masing *pixel*  $g_u$ , dan  $g_{ur}$  adalah 80 dan 76, sehingga diperoleh nilai  $d = (80+76) / 2 - 72 = 6$ . Pesan dapat disisipkan ke dalam sebuah *pixel* citra apabila memiliki nilai  $d \geq 2$  dan  $d \leq -2$ , dan *pixel* akan diabaikan dalam arti tidak akan disisipkan pesan apabila nilai  $d = -1, 0$  dan  $1$ . Karena pada perhitungan sebelumnya diperoleh nilai  $d = 6$ , maka pesan yang dapat disisipkan akan lebih dari 1 bit. Selanjutnya, akan dihitung berapa banyak bit pesan ( $n$ ) yang dapat disisipkan pada sebuah *pixel* dengan rumus seperti persamaan (2) di bawah ini.

$$n = \log_2|d|, \text{ if } |d| > 1 \tag{2}$$

Dari hasil perhitungan  $d$  yang telah dilakukan sebelumnya, diperoleh  $d = 6$ . Kemudian dilakukan perhitungan nilai  $n = \log_2 |6| = 2,585$ , kemudian nilai  $n$  dibulatkan kebawah (*floor*) sehingga diperoleh  $n = 2$ . Setelah mendapatkan nilai  $n$ , maka selanjutnya bit pesan akan dikonversikan ke bilangan *integer*  $b$  dan diperoleh nilai  $d'$  yang baru dengan rumus seperti persamaan (3) di bawah ini.

$$d' = \begin{cases} 2^n + b, \text{ if } d > 1 \\ -(2^n + b), \text{ if } d < 1 \end{cases} \tag{3}$$

Dari perhitungan nilai  $d$ , kita misalkan hasil konversi nilai  $b$  adalah 3, maka dapat dilakukan perhitungan  $d' = 2^2 + 3 = 7$ . Kemudian diperoleh nilai *pixel*  $g_x'$  yang baru, yang merupakan hasil dari penyisipan pesan pada *pixel*  $P_x$  dengan rumus seperti persamaan (4) di bawah ini.

$$g_x' = (g_u + g_{ur}) / 2 - d' \tag{4}$$

Tahapan terakhir adalah perhitungan nilai  $g_x'$  yang baru yang merupakan nilai *pixel* hasil penyisipan pesan. Perhitungan  $g_x'$  dilakukan dengan melibatkan nilai *pixel* tetangga sebelumnya, yaitu

$$g_x' = (80+76) / 2 - 6 = 72. \tag{5}$$

### 2.3.2 Ekstraksi Metode Two Sided Side Match

Dalam ekstraksi pesan dengan metode *Two Sided Side Match* ini, diasumsikan bahwa *pixel* yang telah disisipkan pesan adalah  $P_x^*$  dengan nilai *pixel*  $g_x^*$ , *pixel* tetangga sisi atas  $P_u^*$  dengan nilai *pixel*  $g_u^*$ , dan *pixel* tetangga sisi kanan atas  $P_{ur}^*$  dengan nilai *pixel*  $g_{ur}^*$ . Untuk mengetahui berapa banyak bit pesan yang telah disisipkan, terlebih dahulu dihitung perbedaan nilai  $d^*$  antar nilai *pixel* tetangga dengan rumus seperti persamaan dibawah ini [8]:

$$d^* = (g_u^* + g_{ur}^*) / 2 - g_x^* \tag{6}$$

Dari perubahan nilai *pixel* pada proses penyisipan sebelumnya, dapat kita hitung kembali perbedaan nilai  $d^* = (80 + 76) / 2 - 72 = 6$ . Dari setiap *pixel* akan dihitung berapa banyak bit pesan ( $n$ ) yang telah disisipkan dengan rumus dibawah ini :

$$n = \log_2|d^*|, \text{ if } |d^*| > 1 \tag{7}$$

Sebelumnya telah diperoleh nilai  $d^* = 6$ , maka selanjutnya dapat dihitung nilai  $n = \log_2 |6| = 2,585$ , kemudian nilai  $n$  dibulatkan kebawah (*floor*) sehingga diperoleh  $n = 2$ . Setelah mendapatkan nilai  $n$ , maka selanjutnya dapat dihitung nilai  $b$  dengan rumus (7) dibawah ini :

$$b = \begin{cases} d^* - 2n, \text{ if } d^* > 1 \\ -d^* - 2n, \text{ if } d^* < 1 \end{cases} \tag{8}$$

Tahapan terakhir adalah menghitung nilai  $b$  dengan mempertimbangkan nilai  $d^*$  dan nilai  $n$ , maka  $b$  dapat dihitung dengan  $b = 6 - 2^2 = 2$ . Pada akhir proses ekstraksi, nilai  $b$  dikonversi ke nilai biner untuk mendapatkan pesan yang telah disisipkan

## 2.4 Citra

Citra menurut kamus *Webster* adalah suatu representasi atau gambaran, kemiripan, atau imitasi dari suatu objek atau benda, contohnya yaitu audio seseorang dari kamera yang mewakili orang tersebut, foto sinar *X-thorax* yang mewakili gambar bagian tubuh seseorang dan lain sebagainya. Citra sebagai salah satu komponen multimedia memegang peranan penting sebagai bentuk informasi *visual*. Citra mempunyai karakteristik yang tidak dimiliki oleh data teks, yaitu citra kaya dengan informasi [7].

## 2.5 Audio

Audio merupakan suara atau bunyi yang dihasilkan ketika molekul di udara berubah karena suatu gerakan yang ditimbulkan oleh sebuah objek yang menghasilkan sebuah getaran. Objek ini berupa senar gitar, vocal manusia, atau sebuah kaleng yang bergerak ketika ada energi yang menggetarkannya *Cycle persecond* atau *CPS*. Satuan ini bisa dikenal dengan sebutan *Hertz (Hz)*. Kadangkala getaran yang dihasilkan sangat cepat, dan perbandingan gerakan yang dihasilkan adalah seribu, maka satuan digunakan adalah *kilohertz (KHz)* [9]:.

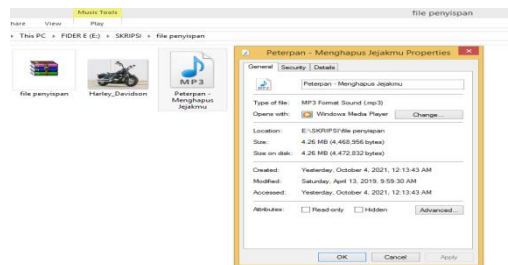
### 3. HASIL DAN PEMBAHASAN

#### 3.1 Analisa Penerapan Metode Two Sided Side Match Dalam Penyisipan Pesan Gambar Pada File Audio

Analisa yang dilakukan dalam penelitian ini adalah menyisipkan pesan gambar pada *file* audio dengan menerapkan Metode *Two Sided Side Match*, dalam penerapan metode ini dilakukan dengan mengubah pesan gambar yang akan disisipkan menjadi bentuk biner dan mengubah *file* audio yang menjadi penampungnya ke bentuk desimal. Proses penyisipan bergantung pada seberapa besar selisih antar desimal pada *file* audio dan dari selisih tersebut menentukan banyaknya biner pesan yang bisa disisipkan, proses tersebut terus berlangsung hingga semua biner pesan tersisipkan pada desimal file audio. Ada beberapa sampel data yang digunakan untuk dijadikan sebagai bahan perancangan dalam melakukan penyisipan *file* gambar pada *file* audio sebagai berikut.



Gambar 2. File Gambar



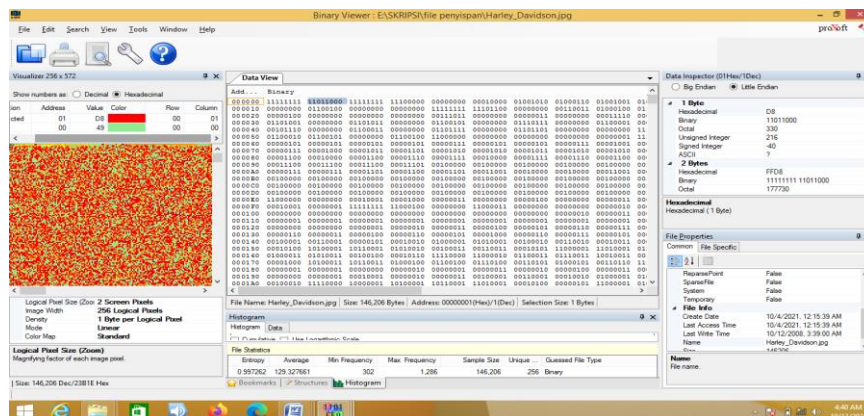
Gambar 3. File Audio

##### 3.1.1 Penerapan Metode Two Sided Side Match

Berikut ini merupakan analisa penyisipan file gambar pada file audio menerapkan metode *Two sided side match* untuk penyisipan pesan pada *file audio* dengan sampel *file gambar pada file audio*. Dengan menggunakan tools binary viewer maka diperoleh nilai decimal dari file sampel gambar Harley Davidson.Jpg dengan stego audio Peterpan-Menghapus Jejakmu.Mp3 sebagai berikut:

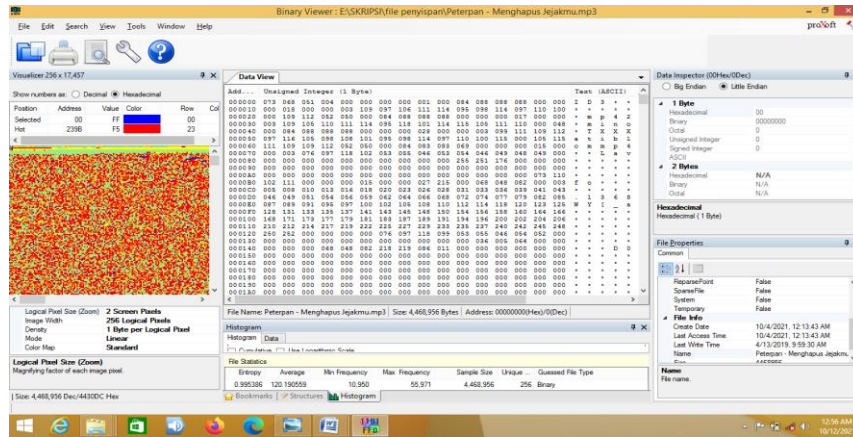
##### 3.1.2 Penerapan Algoritma Nearest Neighbour

Data yang dianalisa adalah data pemilihan bibit unggul kakao pada tahun 2020, data dikonversikan terlebih dahulu ke dalam bentuk angka. Atribut data yang dikonversikan ke dalam angka yaitu atribut perkembangbiakan generatif.



Gambar 4. Nilai Sampel Biner pesan Gambar

Dengan menggunakan bantuan *binary viewer* sehingga diperoleh nilai biner pesan yang akan disisipkan adalah “11011000” ke dalam file audio. Selanjutnya dengan menggunakan *binary viewer* maka diperoleh nilai desimal sampel audio sebagai media penampung pesan gambar berupa sampel biner di atas seperti berikut ini:



**Gambar 5.** Nilai Sampel Desimal file Audio

Dari gambar di atas, didapat data decimal yang tertera pada Proses *Embedding* (Penyisipan).

a. Proses *Embedding* (Penyisipan).

penyisipan pesan dimulai dari koordinat  $x,y = (2,2)$ . Pixel tersebut diasumsikan sebagai  $P_x$  yang mempunyai nilai  $g_x = 173$ . Dimana nilai pixel tetangga atas  $P_u$  yang mempunyai nilai  $g_u = 133$ , dan nilai pixel tetangga kanan atas  $P_{ur}$  yang mempunyai nilai  $g_{ur} = 135$ . Maka proses penyisipan dapat dilakukan seperti berikut :

**Tabel 1.** Hasil Desimal Audio

131	133	135	137	141	143	145	148	150	154
171	173	177	179	181	183	187	189	191	194
212	214	217	219	222	225	227	229	233	235

1. Hitung nilai  $d$  dengan  $d = \frac{g_u + g_{ur}}{2} - g_x$   
 $d = \frac{133 + 135}{2} - 173$

$d = -39 = 39$  (Negatif tidak berlaku)

2. Hitung nilai  $n$  dengan  $n = \log_2 |d| = \log_2 |39| = 5.28 = 5$  (dibulatkan ke bawah)

3. Karena nilai  $n = 5$ , maka ambil 5 bit pesan pertama dari "11011000", yaitu "11011" sehingga sisa pesan "000"

4. Konversikan 5 bit pesan yang akan disisipkan tersebut ke dalam integer, maka  $11011 = 27$

5. Hitung nilai  $d'$  baru dengan  $d' = (2^n + b) = -(2^5 + 27) = 59$

6. Hitung nilai  $g_x'$  baru dengan  $g_x' = \frac{g_u + g_{ur}}{2} - d'$

$G_x' = \frac{133 + 135}{2} - 59$

$G_x' = 75$

Sehingga nilai  $g_x$  yang awalnya adalah 173, sekarang nilai  $g_x$  menjadi 75.

**Tabel 2.** Penyisipan pesan pada piksel koordinat (2,2)

131	133	135	137	141	143	145	148	150	154
171	75	177	179	181	183	187	189	191	194
212	214	217	219	222	225	227	229	233	235

Selanjutnya dilakukan hal yang sama dengan memperhatikan koordinat  $x,y$ , hingga mendapatkan nilai Sehingga nilai  $g_x$  yang awalnya adalah 177, sekarang nilai  $g_x$  menjadi 104.

**Tabel 3.** Hasil Penyisipan pesan pada piksel koordinat (2,3)

131	133	137	137	141	143	145	148	150	154
171	104	177	179	181	183	187	189	191	194
212	214	217	219	222	225	227	229	233	235

b. Proses *Extraction* (Ekstraksi)

Ekstraksi pesan dimulai dari koordinat  $x, y = (2,2)$ . Pixel tersebut diasumsikan sebagai  $P_x$  yang mempunyai nilai  $g_x = 75$ . Dimana nilai pixel tetangga atas  $P_u$  yang mempunyai nilai  $g_u = 133$ , nilai pixel tetangga kanan atas  $P_{ur}$  yang mempunyai nilai  $g_{ur} = 135$ . Maka proses ekstraksi dapat dilakukan seperti berikut.

**Tabel 4.** Ekstraksi pada piksel koordinat (2,3)

131	133	135	137	141	143	145	148	150	154
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

171	75	177	179	181	183	187	189	191	194
212	214	217	219	222	225	227	229	233	235

1. Hitung nilai d dengan  $d = \frac{g_u + g_{ur}}{2} - g_x$   

$$d = \frac{133 + 135}{2} - 75$$

$$d = 59$$
2. Hitung nilai n dengan  $n = \log_2 |d| = \log_2 |59| = 5,88 = 5$  (dibulatkan ke bawah)
3. Hitung nilai b dengan  $b = d^* - 2^n = 59 - 2^5 = 27$
4. Konversikan nilai b ke bilangan biner,  $b = 27 = 11011$
5. Dari hasil ekstraksi yang telah dilakukan sebelumnya, diperoleh bit pesan rahasia yang telah disisipkan, yaitu "11011".

**Tabel 5.** Ekstraksi pada piksel koordinat (2,3)

131	133	135	137	141	143	145	148	150	154
171	75	177	179	181	183	187	189	191	194
212	214	217	219	222	225	227	229	233	235

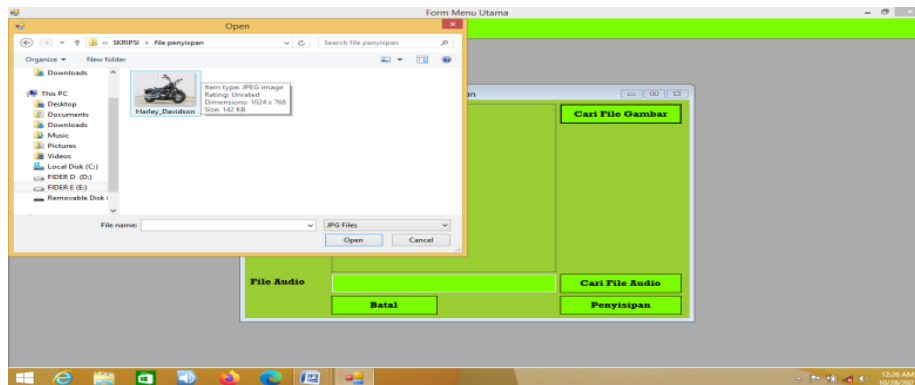
### 3.2 Hasil Pengujian

#### 3.2.1 Pengujian Metode Two Sided Side Match

Untuk mendapatkan hasil dari penelitian ini maka dilakukan perhitungan menggunakan alat bantu yaitu *software Visual basic* Pada bagian ini merupakan hasil dari pengujian rancangan berupa tampilan layar aplikasi yang sedang berjalan

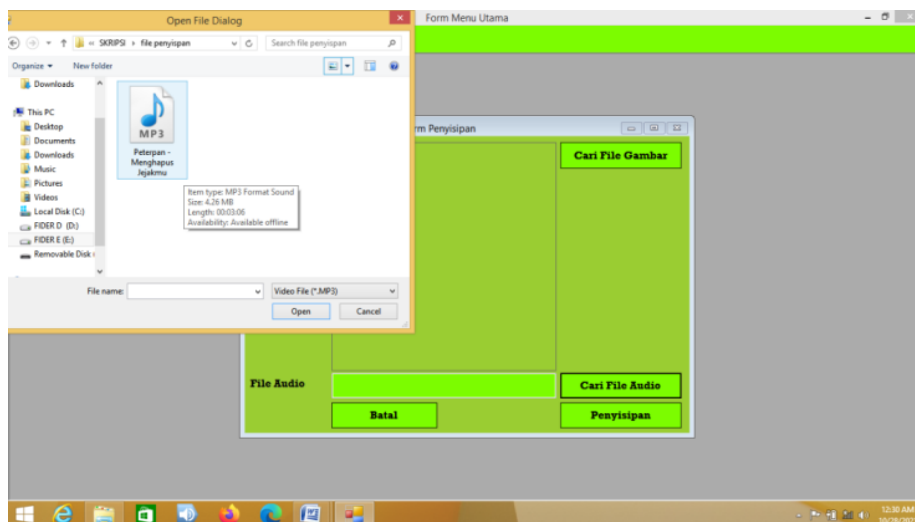
##### a. Form Embedding

Adapun tampilannya hasil pengujian pada *form embedding* dapat dilihat pada gambar dibawah ini



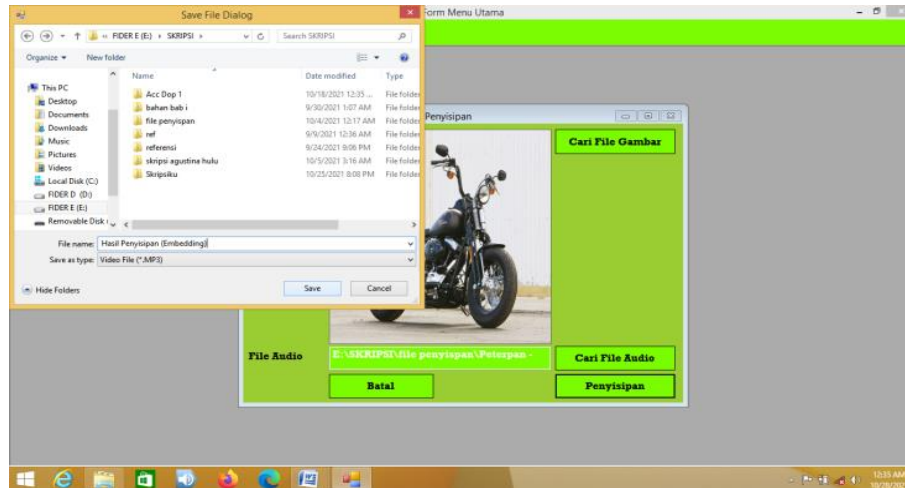
**Gambar 6.** Tampilan cari file gambar

Kemudian input audio yang akan disisipkan pada tampilan gambar dibawah ini:



**Gambar 7.** Tampilan input audio yang akan disisipkan

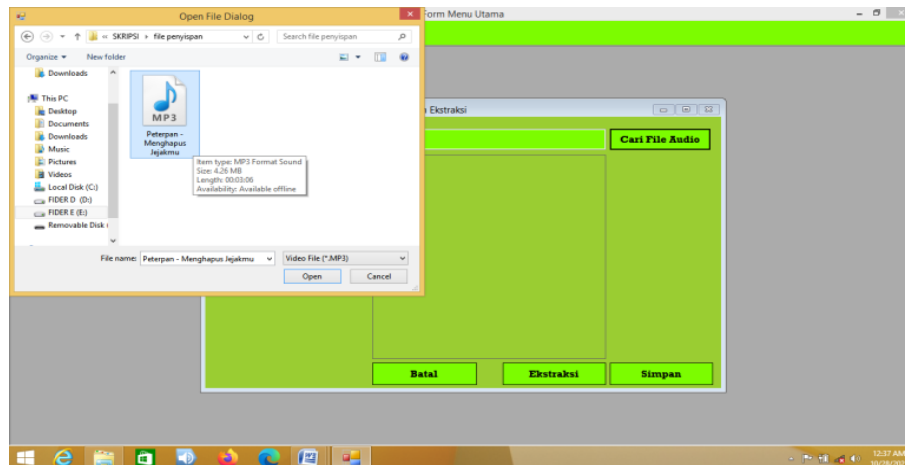
Kemudian setelah proses embedding maka simpan file dari hasil embedding seperti gambar dibawah ini:



**Gambar 8.** Tampilan *embedding* dan simpan *file* hasil *embedding*

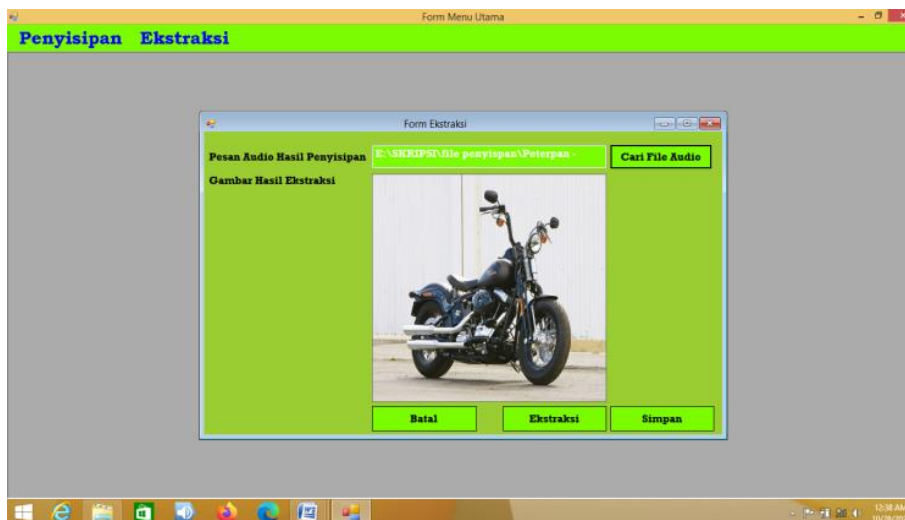
b. *Form* Ekstraksi

Adapun tampilannya hasil pengujian pada form ekstraksi dapat dilihat pada gambar dibawah ini:



**Gambar 9.** Tampilan pencarian *file* audio hasil *embedding*

Kemudian dari hasil pengujian ekstraksi maka didapatkan hasil tampilan seperti gambar dibawah ini:



**Gambar 10.** Tampilan yang akan diekstraksi

Hasil pengujian juga dilakukan terhadap dua *file* yang ada pada sampel data sebelumnya. Hasil yang didapatkan dalam pengujian penyisipan (*embedding*) dan ekstraksi pada sampel data tersebut dapat dilihat pada tabel berikut.



**Tabel 3.6.** Hasil Pengujian

No	Nama File	Pesan yang disisipkan	Proses <i>Embedding</i>	Hasil Proses Ekstraksi
1	Harley_Davidson. JPG	√	√	√
2	Peterpan - Menghapus Jejakmu.	√	√	√

Dari hasil pengujian di atas, maka disimpulkan bahwa penyisipan gambar pada *file* audio dengan menerapkan metode *Two Sided Side Match* dapat dilakukan dan memperoleh pesan hasil ekstraksi yang sama dengan gambar ketika disisipkan ke dalam *file* audio. Metode *Two Sided Side Match* merupakan salah satu metode yang bisa menjadi alternatif solusi untuk penyisipan gambar pada *file* audio.

#### 4. KESIMPULAN

Berdasarkan pembahasan dan hasil penelitian yang dilakukan oleh peneliti dari bab-bab sebelumnya maka dapat diambil kesimpulan. Penyisipan sebuah gambar dengan format Jpg dengan maksimal resolusi 2048x1536 kedalam file audio yaitu mengekstraksi pesan gambar kedalam audio dengan format Mp3 berukuran maksimal 15 mb menggunakan teknik steganografi. Penerapan metode *Two Sided Side Match* dapat dilakukan dalam dua proses yaitu proses penyisipan (*embedding*) dan proses ekstraksi dengan menggunakan teknik steganografi. Aplikasi penyisipan gambar pada file audio dapat dirancang menggunakan bahasa pemograman *visual studio 2010* dan metode *Two Sided Side Match* sehingga dapat mempermudah proses penyisipan dan proses ekstraksi.

#### REFERENCES

- [1] Cole, E. 2003. *Hiding in Plain Sight : Steganography and the Art of Covert Communication*. Wiley : Canada. IEEE : Kolkata, West Bengal.
- [2] Cummins, J., Diskin, P., Lau, S., & Parlett, R.2004. *Steganography and Digital Watermarking*. The University of Birmingham.
- [3] Hermawati, F. A. 2013. *Pengolahan Citra Digital*. Andi : Yogyakarta
- [4] Khairina, Nurul. 2016. *Analisis Perbandingan Metode Steganografi Two Sided Side Match Dengan Four Sided Side Match Pada Citra Multilayer TIFF*. Tesis. Medan : USU.
- [5] Sutoyo, T., Mulyanto, E., Suhartono, V., Nurhayati, O.D. & Wijanarto. 2009. *Teori Pengolahan Citra Digital*. Andi : Yogyakarta
- [6] Swain, G. 2013. *Steganography in Digital Image Using Maximum Difference of Neighboring Pixel Values*. *International Journal of Security and its Applications* 7(6) : 285-294.
- [7] Yunus, M., & Harjoko, Agus. 2014. *Penyembunyian Data pada File Video Menggunakan Metode LSB dan DCT*. *IJCCS* 8 (1) : 81-90.