**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI**

**INSTITUT TEKNOLOGI SEPULUH NOPEMBER**

USULAN TUGAS AKHIR

# IDENTITAS PENGUSUL

Nama : **Claracia Dinasty**

NRP : **5109 100 109**

Dosen Wali : **Diana Purwitasari, S.Kom., M.Sc.**

# JUDUL TUGAS AKHIR

***“Implementasi Steganografi Citra Berdasarkan Metode Penyisipan Informasi Tinggi dan Deteksi Tepi”***

# LATAR BELAKANG

Keamanan adalah hal yang sangat penting, termasuk dalam bidang teknologi informasi. Keamanan data menjadi sesuatu yang sangat dibutuhkan terutama jika pesan yang dikirimkan bersifat sangat rahasia. Berbagai usaha dilakukan untuk menjamin agar pesan rahasia yang dikirimkan tersebut tidak bisa diakses oleh pihak lain. Metode yang cukup umum digunakan kriptografi, yaitu teknik menyamarkan arti dari suatu pesan, misalnya dengan mengacak atau menyandikan pesan tersebut, sehingga bila pesan tersebut jatuh ke tangan yang salah, diharapkan orang tersebut tidak akan mendapatkan informasi yang diinginkan. Akan tetapi, kriptografi tidak menyembunyikan keberadaan pesan tersebut. Untuk itulah Steganografi dibutuhkan.

Steganografi adalah seni dan ilmu menulis pesan tersembunyi atau menyembunyikan pesan dengan suatu cara sehingga selain si pengirim dan si penerima, tidak ada seorangpun yang mengetahui atau menyadari bahwa ada suatu pesan rahasia. Pada era informasi digital, steganografi merupakan teknik dan seni menyembunyikan informasi dan data digital dibalik informasi digital lain, sehingga informasi digital yang sesungguhnya tidak kelihatan. Steganografi pada media digital digunakan untuk mengeksploitasi keterbatasan kekuatan sistem indera penglihatan dan pendengaran manusia, sehingga dengan keterbatasan tersebut sulit menemukan perubahan pada berkas yang telah disisipi pesan rahasia.

Hampir semua jenis berkas digital dapat digunakan sebagai media penampung pesan, baik berkas berkas citra, berkas *audio*, maupun berkas *video*. Semua berkas dapat dijadikan tempat bersembunyi, asalkan berkas tersebut memiliki bit-bit data redundan yang dapat dimodifikasi. Berkascitra merupakan media yang sering digunakan dalam dunia internet maupun dunia citra digital, ukuran *berkas*nya relatif kecil apabila dibandingkan dengan berkas *audio* atau *video*. Beberapa studi membuktikan bahwa *sharp area* dapat menampung pesan tanpa terdekteksi lebih banyak daripada *smooth area*, oleh karena itu diberlakukan deteksi tepi untuk menentukan apakah pixel tersebut terletak di dibagian tepi atau tidak. Selain itu yang perlu dibahas adalah bagaimana dapat menyisipkan pesan rahasia dengan kapasitas sebanyak mungkin ke dalam media penampung tanpa terdekteksi.

# RUMUSAN MASALAH

Rumusan masalah yang diangkat dalam Tugas Akhir ini adalah sebagai berikut :

* + 1. Bagaimanakah penggunaan metode penyisipan tinggi dalam steganografi?
    2. Bagaimanakah deteksi tepi dapat mempengaruhi banyaknya pesan rahasia yang disisipkan?
    3. Bagaimanakah mendesain algoritma dalam MATLAB agar diketahui kinerjanya?
    4. Bagaimanakah hasil uji coba perbandingan dengan metode lain?

# BATASAN MASALAH

Permasalahan yang dibahas pada Tugas Akhir ini memiliki beberapa batasan, yaitu sebagai berikut:

1. Implementasi menggunakan MATLAB 7.6.0.
2. Media penampung bertipe citra.
3. Media yang disisipkan bertipe *text* atau citra.
4. Uji coba metode dilakukan menggunakan menggunakan nilai PNSR (Peak Signal to Noise Ratio).

# TUJUAN TUGAS AKHIR

Tujuan pada Tugas Akhir ini adalah:

1. Menyembunyikan pesan dengan kapasitas sebesar mungkin tanpa terdeteksi.
2. Mengimlementasikan metode penyisipan informasi tinggi dan deteksi tepi pada MATLAB sehingga dapat dilihat kinerjanya.
3. Mengetahui hasil uji coba metode.

# MANFAAT TUGAS AKHIR

Hasil dari Tugas Akhir ini diharapkan dapat memberi manfaat pada bidang teknologi informasi terutama dalam bidang keamanan data.

# RINGKASAN TUGAS AKHIR

Dua hal penting dalam melakukan steganografi adalah proses penyembunyian pesan rahasia dan proses ekstraksi pesan rahasia.

## Penyembunyian pesan rahasia

Langkah – langkah untuk melakukan penyisipan pesan rahasia kedalam citra adalah sebagai berikut :

* + - 1. Memilih *host image*, dalam tahap ini, tidak ada kriteria khusus. *Host image* bisa berupa citra apapun yang diinginkan.
      2. Membuat versi *grayscale* dari host image dengan formula : I = 0.299\*R + 0.587\*G + 0.114\*B
      3. Mengaplikasikan teknik deteksi tepi *hybrid* pada versi *grayscale* dari *host image*. Teknik deteksi tepi *hybrid* yang digunakan dalam tugas akhir ini adalah perpaduan *fuzzy edge detector* dengan *sobel filter* yang menghasilkan citra Sobel OR Fuzzy dan *fuzzy edge detector* dengan *laplacian filter* yang menghasilkan citra Laplacian OR Fuzzy.



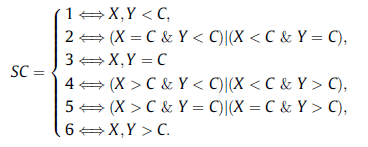
Gambar Alur Diagram Deteksi Tepi Fuzzy

* + - 1. Melakukan perhitungan jumlah bits yangs bisa disisipkan pada host image untuk setiap *channel* warna merah, hijau, dan biru pada setiap pixel menggunakan metode *penyisipan informasi tinggi*. Pada metoda ini pertama – tama ditentukan nilai dari MC (Main Case) untuk setiap pixel. MC ditentukan bernilai antara 1 hingga 16. Untuk menghitungnya menggunakan rumus berikut :



Kemudian menentukan MCcolor = index, dengan 1 ≤ index ≤ 16, dan .

Selanjutnya, dengan mengasumsikan C sebagai nilai MC warna saat ini, dan X,Y adalah nilai MC untuk warna lainnya, didapatkan index dari SC (Sub Case) :



Dengan CP sebagai pixel saat ini yang berisikan set warna {R,G, dan B}, dan NP sebagai pixel selanjutnya (setelah CP).

Warna yang sedang dipilih :



MC untuk warna yang sedang dipilih :



Sedangkan RCi adalah warna lain dari pixel (warna yang sedang tidak dipilih)

Pseudo-code untuk menentukan jumlah bit yang dapat disisipkan adalah sebagai berikut :

If (Cmc>=2 AND SC = 1) {

//Check the MC of rest colors

If (RC = 1) {

//Check if all current pixel property is equal to

the next pixel property

If (CP.property = NP.property) {

//Hide 2 data bits in the selected color

from the current pixel

Hide (CP.C, DataBits (2));

//Hide 2 data bits in the selected color byte

from the next pixel

Hide (NP.C, DataBits (2));

}

}

}

Else if (Cmc>=1 AND Cmc<=16 AND SC = 3) {

//Hide 1 data bit in the Red color byte

Hide (CP.R, DataBits (1));

//Hide 1 data bit in the Green color byte

Hide (CP.G, DataBits (1));

//Hide 2 data bits in the Blue color byte

Hide (CP.B, DataBits (2));

}

Else if (Cmc>=1 AND Cmc<=15 AND SC = 5) {

//Hide 2 data bits in the selected color byte from

the current pixel

Hide (CP.C, DataBits (2));

//Hide data bits in the greater or equal MC byte for

the selected color min case from the

//rest pixel bytes

Hide (CP.E, DataBits (2));

}

Else if (Cmc>=1 AND CP.Cmc<=15 AND SC = 6) {

//Hide 2 data bits in the selected color byte from

the current pixel

Hide (CP.C, DataBits (2));

//Hide data bits in the greater or equal MC byte for

the selected color min case from the

//rest pixel bytes

Hide (CP.H, DataBits (2)); }

* + - 1. Jika menurut proses deteksi tepi pixel tersebut merupakan pixel tepi, maka jumlah bits yang disisipkan untuk setiap *channel* warna merah, hijau, dan biru akan ditambahkan 1 bit ekstra.
      2. Untuk keamanan lebih, tidak semua pixel disisipi pesan rahasia, melainkan menggunakan *pseudorandom number generator* untuk menentukan pixel mana yang akan disisipi selanjutnya (bisa 1 pixel berikut, 2 pixel berikut, atau 3 pixel berikutnya, sehingga untuk *best case* adalah semua pixel digunakan, sedangkan untuk *worst case*, hanya 30% pixel yang disisipi pesan rahasia)
      3. Memilih pesan rahasia yang akan disisipkan, dengan ketentuan, ukurannya tidak boleh melebihi batas ukuran yang sudah dihitung pada step 4-6.
      4. Menyisipkan pesan rahasia ke dalam *host image*, sehingga menghasilkan 2 *stego image* (masing – masing untuk citra Sobel OR fuzzy dan Laplacian OR Fuzzy).
      5. Membuat berkas – berkas pelengkap yang digunakan untuk proses ekstraksi (1 berkas untuk citra Sobel OR fuzzy dan 1 berkas Laplacian OR Fuzzy). Setiap berkas berisi informasi ukuran panjang dan lebar citra, jumlah bits yang dimasukkan yang diubah dalam setiap *channel* warna merah, hijau, dan biru pada setiap pixel dari *host image*.
      6. Mengenkripsi berkas pelengkap menggunakan algoritma Triple-DES untuk keamanan.

Secara umum proses penyembunyian data dapat digambarkan dalam alur diagram sebagai berikut :



Gambar Alur Diagram Proses Penyembunyian Pesan Rahasia

## Ekstraksi pesan rahasia

Langkah – langkah untuk melakukan proses ekstraksi pesan rahasia dari stego image adalah sebagai berikut ;

1. Memuat *stego image* yang akan diekstraksi.
2. Memasukkan kunci rahasia yang digunakan mendekripsi berkas pelengkap yang berisi data – data penyembunyian data yang sebelumnya telah dienkripsi menggunakan algoritma Triple-DES.
3. Mengekstrak pesan rahasia dari *stego image* berdasarkan informasi pada berkas.

# METODOLOGI

Metodologi yang akan dilakukan dalam Tugas Akhir ini memiliki beberapa tahapan, diantaranya sebagai berikut :

1. Penyusunan Proposal Tugas Akhir

Tahap awal untuk memulai pengerjaan Tugas Akhir adalah penyusunan Proposal Tugas Akhir. Pada proposal ini, penulis mengajukan gagasanImplementasi Steganografi Citra Berdasarkan Metode Penyisipan Informasi Tinggi dan Deteksi Tepi

1. Studi Literatur

Pada tahap ini dilakukan pencarian informasi dan studi literatur yang diperlukan untuk pengumpulan data dan desain sistem yang akan dibuat. Informasi didapatkan dari buku dan materi-materi lain yang berhubungan dengan algoritma metode yang digunakan dalam pengerjaan tugas akhir ini, yang didapat dari *internet* maupun buku acuan.

1. Implementasi

Implementasi merupakan tahap untuk membangun algoritma tersebut. Untuk membangun algoritma yang telah dirancang sebelumnya, diimplementasikan dengan menggunakan MATLAB.

1. Pengujian dan Evaluasi

Pada tahap ini dilakukan uji coba dengan menggunakan 15 fungsi numerik untuk mencoba jalannya aplikasi apakah telah sesuai dengan rancangan dan desain implementasi yang dibuat, juga untuk mencari kesalahan – kesalahan program yang mungkin terjadi untuk selanjutnya dapat dilakukan penyempurnaan.

1. Penyusunan Buku Tugas Akhir

Tahap terakhir merupakan penyusunan laporan yang memuat dokumentasi mengenai pembuatan serta hasil dari implementasi perancangan dan prototyping yang telah dibuat. Secara garis besar, buku laporan tugas akhir ini terdiri atas beberapa bagian yaitu:

1. Pendahuluan
   1. Latar Belakang
   2. Permasalahan
   3. Batasan Tugas Akhir
   4. Tujuan
   5. Metodologi
   6. Sistematika Penulisan
2. Tinjauan Pustaka
3. Desain dan Implementasi
4. Uji Coba dan Evaluasi
5. Kesimpulan dan Saran
6. Daftar Pustaka

# JADWAL KEGIATAN TUGAS AKHIR

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Tahapan** | **Bulan** | | | | | | | | | | | | | | | | | | | | | | | |
| **September** | | | | | **Oktober** | | | | | **November** | | | | **Desember** | | | | | | **Januari** | | | |
| 1 | Penyusunan Proposal |  |  |  |  |  | |  |  |  |  | |  |  |  | |  |  |  |  |  | |  |  |  |
| 2 | Studi Literatur |  |  |  |  |  | |  |  |  |  | |  |  |  | |  |  |  |  |  | |  |  |  |
| 3 | Implementasi |  |  |  |  |  | |  |  |  |  | |  |  |  | |  |  |  |  |  | |  |  |  |
| 4 | Pengujian dan Evaluasi |  |  |  |  |  | |  |  |  |  | |  |  |  | |  |  |  |  |  | |  |  |  |
| 5 | Penyusunan Buku Tugas Akhir |  |  |  |  |  | |  |  |  |  | |  |  |  | |  |  |  |  |  | |  |  |  |

# DAFTAR PUSTAKA

* 1. Loannidou, Anastasia, Halkidis, Spyros T., & Stephanides, George (2012). A novel technique for image steganography based on penyisipan informasi tinggi method and edge detection. expert systems with applications, 39(4), 11517-11524.
  2. Gonzales, Rafael C., & Woods, Richard E. (2008). Digital Image Processing : Third Edition.
  3. Chen, Wen-Jan, Chang, Chin-Chen, & Hoang Ngan Le, T. (2010). Penyisipan informasi tinggi steganography mechanism using hybrid edge detector. Expert Systems with Applications, 37(4), 3292–3301.
  4. EL-Emam, Nameer N. (2007). Hiding a large amount of data with high security using steganography algorithm. Journal of Computer Science, 3(4), 223–232.