**DEPARTEMEN INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI DAN KOMUNIKASI**

**INSTITUT TEKNLOGI SEPULUH NOPEMBER**

**USULAN TUGAS AKHIR**

# IDENTITAS PENGUSUL

**NAMA : Alek Nur Fatman**

**NRP : 05111440000018**

**DOSEN WALI : Dr. Agus Zainal Arifin, S.Kom., M.Kom.**

**DOSEN PEMBIMBING : 1. Tohari Ahmad, S.Kom., MIT., Ph.D.  
 2.**

# JUDUL TUGAS AKHIR

“Modifikasi Metode *Histogram Shifting* dan *Neighbor Similarity* untuk *Steganografi* pada *Video Digital*.”

# LATAR BELAKANG

Seiring dengan perkembangan zaman dimana perkembangn teknologi berkembang pesat, semakin besar pula kebutuhan yang diperlukan manusia untuk saling bertukar data. Dalam pertukaran data, keamanan data merupakan salah satu hal yang menjadi perhatian penting.

*Steganography* merupakan salah satu metode yang digunakan untuk menyembunyikan informasi. Dengan s*teganography* informasi akan disembunyikan dengan cara menyisipkan informasi tersebut kedalam data lain [1]. *Steganography* dibagi menjadi dua, yaitu *reversible* *data hiding* dan *irreversible data hiding.* *Reversible* *data hiding* adalah metode dalam *data hiding* yang dimana hasil dari penyisipan data bisa dikembalikan atau dipulihkan sama seperti data sebelum disisipi. Sedangkan *irreversible data hiding* adalah metode dalam *data hiding* yang tidak dapat dipulihkan data dari hasil penyisipan kembali ke data sebelum dilakukan penyisipan.

Contoh dari *reversible* *data hiding* adalah *histogram shifting* dan *neighbouring similarity*. Dalam tugas akhir ini akan dilakukan penelitian untuk meningkatkan kemiripan data hasil penyisipan pada media video digital dengan modifikasi metode *neighbouring similarity.*

# RUMUSAN MASALAH

Berikut beberapa hal yang menjadi rumusan masalah dalam tugas akhir ini:

1. Bagaimana mengembangkan tingkat kemiripan dan jumlah input dalam penyisipan dibandingkan dengan menggunakan metode *neighbouring similarity*?
2. Bagaimana mengatasi masalah yang muncul akibat modifikasi dalam pembangkitan *prediction error*?

# BATASAN MASALAH

Batasan masalah pada tugas akhir ini adalah sebagai berikut:

1. Video yang digunakan untuk data uji berasal dari xiph.org kategori *derf collection* [2]
2. Video yang digunakan sebanyak 20 video dengan sebagian besar dipilih sama dengan video yang digunakan pada penelitian sebelumnya.
3. Data masukan untuk penyisipan adalah teks dari *Lorem Ipsum* [3]

# TUJUAN PEMBUATAN TUGAS AKHIR

Tujuan dari pembuatan tugas akhir ini adalah megusulkan sebuah metode *steganography* untuk video digital yang bersifat *reversible* (bisa dikembalikan). Maksud dari *reversible* adalah video yang sudah disisipi dapat dikembalikan ke keadaan semula seperti sebelum disisipi pesan rahasia.

# MANFAAT TUGAS AKHIR

Manfaat dari tugas akhir ini adalah sebagai berikut:

1. Mengembangkan metode *neighbouring similarity* pada videosehingga hasil penyisipan semakin meningkat kemiripannya dengan video sebelum disisipi dan juga meningkatkan jumlah inputan.
2. Memberikan sumbangasih dalam bidang teknologi terutama dalam bidang steganografi

# TINJAUAN PUSTAKA

1. **Video Digital**

Video Digital adalah representasi dari adegan video yang berbentuk digital, dimana setiap sampel diwakili oleh angka-angka yang menunjukkan nilai dari pencahayaan dan warna sampel tersebut [4]. Sebuah video terdiri dari beberapa frame yang hampir mirip antar satu frame dengan frame selanjutnya.

Dengan pergantian frame yang hampir mirip tersebut, objek dalam video tersebut terlihat bergerak. Semakin banyak frame yang digunakan dalam suatu waktu, maka akan terlihat lebih nyata pergerakan objek dalam video.

1. **Steganography**

*Steganography* berasal dari kata “stegos” yang berarti tertutup dan “grafia” yang berarti tulisan. *Steganography* dan *cryptography* adalah metode yang digunakan dalam *data hiding* [1]. Steganography adalah metode yang digunakan untuk menyembunyikan data kedalam data lain, sedangkan *cryptography* digunakan untuk mengubah bentuk dari sebuah data menggunakan algoritma dan kunci tertentu.

1. **Histogram Shifting**

Histogram Shifting adalah salah satu algoritma *reversible steganography* [5]*.* Tahap-tahap algoritma ini adalah:

* 1. Memindai seluruh piksel pada citra digital secara vertikal dan horizontal.
  2. Menentukan *peak point* dari nilai yang memiliki frekuensi paling tinggi, dan *zero point* dari nilai yang memiliki frekuensi paling rendah.
  3. Menggeser seluruh piksel yang berada diantara *peak point* dan *zero point.* Tujuan dari pergeseran adalah untuk mengosongkan nilai disebelah *peak point*. Pergeseran dilakukan menggunakan persamaan (1).

|  |  |
| --- | --- |
|  |  |
|  | (1) |

Di mana α adalah piksel, x dan y adalah koordinat dari piksel α, p adalah peak point dan z adalah zero point.

* 1. Menyisipkan pesan rahasia pada piksel yang memiliki nilai sama dengan peak point. Proses penyisipan menggunakan persamaan (2).

|  |  |
| --- | --- |
|  |  |
|  | (2) |

Di mana nilai adalah bit dari pesan rahasia ke-n.

* 1. Untuk mengekstrak gunakan persamaan (3).

|  |  |
| --- | --- |
|  |  |
|  |  |
|  | (3) |

* 1. Untuk mengembalikan ke keadaan semula gunakan persamaan (4).

|  |  |
| --- | --- |
|  |  |
|  | (4) |
|  |  |

1. **Neighbouring Similarity**

Neighbouring Similarity adalah metode *reversible steganography* untuk cover data berupa video digital [6]. N*eighbouring similarity* adalah pengembangan dari *histogram shifting.* Jika diterapkan pada video digital *histogram shifting* membangkitkan *histogram* berdasarkan nilai warna piksel pada citra digital atau pada suatu frame, sedangkan neighbouring similarity membangkitkan histogram berdasarkan nilai selisih antar frame pada video digital yang disebut dengan prediction error. Tahapan dalam penyisipan N*eighbouring similarity* adalah sebagai berikut:

* 1. Membangkitkan *prediction error* pada frame berdasarkan nilai piksel dari frame tersebut. Prediction error adalah nilai absolut dari selisih antar piksel pada frame yang saling berdekatan tetapi tidak saling overlap.
  2. Memindai seluruh nilai prediction error secara vertikal dan horizontal.
  3. Menggeser semua nilai setelah *peak point* ke kanan. Perbedaan N*eighbouring similarity* dengan *histogram shifting* adalah n*eighbouring similarity* tidak memakai nilai *zero point* sebagai sasaran pergeseran *histogram* tetapi memakai nilai paling kanan *histogram*(nilai 255). Persamaan (5) digunakan untuk proses pergeseran:

|  |  |
| --- | --- |
|  | (5) |

* 1. Menyisipkan payload pada *prediction error* yang memiliki nilai sama dengan *peak point* menggunakan persamaan (6).

|  |  |
| --- | --- |
|  | (6) |

Di mana βi adalah payload ke-i, δ adalah informasi *overhead* dan m adalah *secret message*.

* 1. Untuk mengekstrak gunakan persamaan (3).
  2. Untuk mengembalikan ke keadaan semula gunakan (4).

1. **Python**

Python adalah salah satu bahasa pemrograman gratis yang mudah dipakai baik untuk programmer awam maupun berpengalaman. Python memilki banyak *modul* dan *library* yang dapat digunakan [7].

1. **Peak to Signal Noise Ratio (PSNR)**

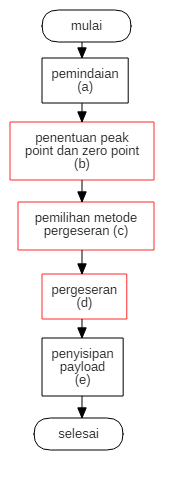
*Peak to Signal Noise Ratio* (PSNR) adalah tolak ukur untuk mengevaluasi metrik kualitas dari sebuah video [8]. PSNR menggunakan formula Mean Square Error (MSE) untuk membandingkan nilai piksel pada cover data dengan stego data. Nilai PSNR dapat dihitung mneggunakan persamaan (7).

|  |  |
| --- | --- |
|  |  |
|  | (7) |

Di mana w dan h adalah dimensi frame, x dan y adalah koordinat nilai piksel, s adalah stego data, c adalah cover data, dan Cmax adalah nilai maksimum dari piksel.

# RINGKASAN ISI TUGAS AKHIR

Pada tugas akhir ini dilakukan modifikasi terhadap algoritma *neighbouring similarity* pada video[6]*.* Modifikasi dilakukan untuk meningkatkan kemiripan hasil dari penyisipan dengan video asli sebelum dilakukan penyisipan. Untuk meningkatkan kemiripan, maka diperlukan metode yang dapat mempersedikit perubahan yang dilakukan terhadap piksel-piksel dalam frame selama dilakukan pergeseran nilai. Pada *histogram shifting* pergeseran dilakukan dari *nilai peak point* ke *zero point.* Sedangkan pada *neighbouring similarity* dilakukan pergeseran pada prediction error dari *peak point* ke arah kanan(nilai 255). Pada tugas akhir ini akan dilakukan pemilihan metode yang akan digunakan untuk melakukan pergeseran.



Gambar 1 Diagram Alur

Tahapan yang dilakukan dalam modifikasi algoritma adalah:

* 1. Memindai seluruh piksel dari citra digital secara vertikal dan horizontal pada frame 1 dan 2.
  2. Menentukan *peak point 1* dari nilai yang memiliki frekuensi paling tinggi pada frame 2, *zero point 1* dari nilai yang memiliki frekuensi paling rendah pada frame 2, *zero point 2* adalah nilai 0*,* dan *zero point 3* adalah nilai 255. Menentukan *peak point 2* dari nilai yang memiliki frekuensi tertinggi dari nilai selisih antara frame 1 dan frame 2, *zero point 4* dari nilai yang memiliki frekuensi paling rendah dari nilai selisih antara frame 1 dan frame 2, *zero point 5* adalah selisih paling kiri (nilai 0)*,* dan *zero point 6* adalah selisih paling kanan (nilai 255).
  3. Mencari nilai pemilihan (S) terbesar untuk menentukan cara pergeseran yang akan dilakukan. Pemilihan dilakukan dengan membandingkan jumlah frekuensi *peak point* ke-i () dan jumlah frekuensi semua nilai yang berada antara *peak point* ke-i dan *zero point* ke-j (). Pemilihan menggunakan persamaan (8)

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  | (8) |

* 1. Melakukan pergeseran nilai yang berada diantara *peak point* dan *zero point.* Pergeseran dilakukan berdasarkan nilai S yang paling besar. Pergeseran dilakukan sebagai berikut:
     1. Jika S1 adalah nilai terbesar, lakukan pergeseran sesuai dengan persamaan (1).
     2. Jika S2 adalah nilai terbesar, lakukan pergeseran sesuai persamaan (9).

|  |  |
| --- | --- |
|  | (9) |

* + 1. Jika S3 adalah nilai terbesar, lakukan pergeseran sesuai persamaan (10).

|  |  |
| --- | --- |
|  | (10) |

* + 1. Jika S4 adalah nilai terbesar, lakukan pergeseran sesuai persamaan (11).

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  | (11) |

* + 1. Jika S5 adalah nilai terbesar, buat *prediction error* lalu lakukan pergeseran pada *prediction error* menggunakan persamaan (12).

|  |
| --- |
|  |
|  | (12) |

* + 1. Jika S6 adalah nilai terbesar, buat *prediction error* lalu lakukan pergeseran pada *prediction error* menggunakan persamaan (13)

|  |
| --- |
|  |
|  | | (13) |

* 1. Menyisipkan payload ke dalam *peak point.* Penyisipan dilakukan berdasarkan persamaan (2).
  2. Untuk mengekstrak gunakan persamaan (3) jika S1, S2 atau S3 yang terpilih. Jika yang terpilih S4, S5 atau S6 gunakan persamaan (14).

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  | (14) |

* 1. Untuk mengembalikan ke keadaan semula gunakan (4) jika S1, S2 atau S3 yang terpilih. Jika yang terpilih S4, S5 atau S6 gunakan persamaan (15).

|  |  |
| --- | --- |
|  |  |
|  | (15) |

Perbedaan metode yang diajukan dengan metode yang sebelumnya bisa dilihat di Tabel 1 Perbedaan Metode yang Diajukan.

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Histogram Shifting | Neighbouring similarity | Metode yang diajukan |
| Letak penyisipan | Di setiap frame | Di prediction error | Di frame genap |
| Arah pergeseran | Ke arah zero point | Ke arah kanan (nilai 255) | Tergantung nilai S yang terbesar. |
| Jumlah bit yang bisa dimasukkan | Jumlah peak point pada frame | Jumlah peak point pada prediction error | Dipilih berdasarkan persamaan (8) |
| Adanya prediction error | Tidak | Iya | Tidak |

Tabel 1 Perbedaan Metode yang Diajukan

# METODOLOGI

## Penyusunan proposal tugas akhir

Proposal Tugas Akhir ini berisikan mengenai apa saja yang dibutuhkan, serta rumusan masalah yang ada dalam “Modifikasi Metode *Histogram Shifting* dan *Neighbor Similarity* untuk *Steganografi* pada *Video Digital*”.

## Studi literatur

Tugas akhir ini menggunakan literatur *thesis* dengan judul “An improved neighbouring similarity method for video steganography” [9] yang merupakan acuan utama dan dasar dalam pengerjaan tugas akhir ini.

## Analisis dan desain perangkat lunak

Pada tahap ini akan dilakukan analisis dan design perancangan aplikasi sesuai dengan tujuan yang dijabarkan. Kemudian disesuaikan dengan metode yang tepat, hal ini dimaksudkan agar nantinya ketika diimplementasikan ke dalam aplikasi dapat berjalan sesuai yang diharapkan.

## Implementasi perangkat lunak

Implementasi merupakan tahap untuk membangun metode tersebut. Untuk mengimplementasikan metode tersebut menggunakan bahasa pemrograman *python*.

## Pengujian dan evaluasi

Pada tahap ini dilakukan uji coba dengan menggunakan bermacam citra masukan untuk menguji aplikasi apakah telah berjalan sesuai dengan rancangan dan desain implementasi yang dibuat, mengamati kinerja sistem, serta mengidentifikasi kendala yang mungkin muncul.

## Penyusunan Buku Tugas Akhir

Pada tahap ini dilakukan penyusunan laporan yang menjelaskan dasar teori dan metode yang digunakan dalam tugas akhir ini serta hasil dari implementasi aplikasi perangkat lunak yang telah dibuat. Sistematika penulisan buku tugas akhir secara garis besar antara lain:

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

# JADWAL KEGIATAN

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Tahapan | 2017 | | | | | 2018 | | | | | | | | | | | | | | | |
| Desember | | | | Januari | | | | | Februari | | | | Maret | | | | April | | | |
| Penyusunan Proposal |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Studi Literatur |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Perancangan Sistem |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Implementasi |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Pengujian dan Evaluasi |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Penyusunan Buku |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

# DAFTAR PUSTAKA

|  |  |
| --- | --- |
| [1] | M. S. Subhedar and V. H. Mankar, "Current status and key issues in image steganography: A survey," *Computer Science Review,* 2014. |
| [2] | "Xiph.org :: Derf's Test Media Collection," [Online]. Available: https://media.xiph.org/video/derf/. [Accessed 30 December 2017]. |
| [3] | "Lorem ipsum - download text and word document examples," Signale, [Online]. Available: http://www.loremipsum.de/download\_lorem\_ipsum.html. [Accessed 3 1 2018]. |
| [4] | I. E. G. Richardson, H. 264 and MPEG-4 video compression: video coding for, John Wiley & Sons, 2003. |
| [5] | Z. Ni, Y.-Q. Shi, N. Ansari and W. Su, "Reversible data hiding," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 16, no. 3, pp. 354 - 362, 2006. |
| [6] | H.-L. Yeh, S.-T. Gue, P. Tsai and W.-K. Shih, "Reversible video data hiding using neighbouring similarity," *IET Signal Processing,* vol. 8, no. 6, pp. 579 - 587, 2014. |
| [7] | "About Python | Python.org," python.org, [Online]. Available: https://www.python.org/about/. [Accessed 4 January 2018]. |
| [8] | S. Wolf and M. Pinson, "Reference Algorithm for Computing Peak Signal to Noise Ratio (PSNR) of a Video," 2009. |
| [9] | D. M. Firmansyah and T. Ahmad, "An improved neighbouring similarity method for video steganography," in *Cyber and IT Service Management, International Conference on*, Bandung, 2016. |