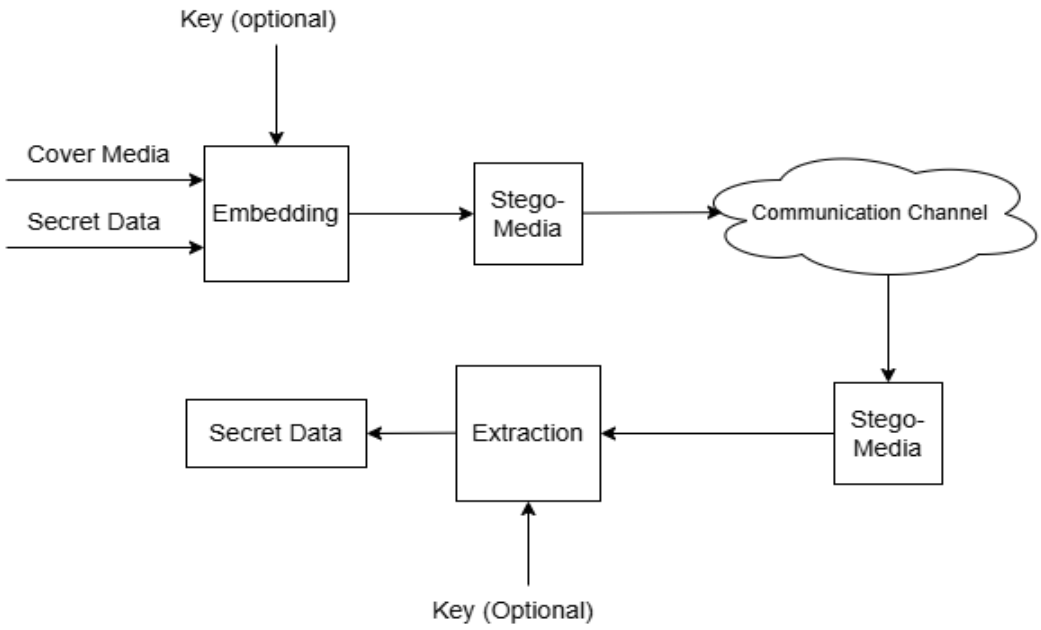
BAB II
TINJAUAN PUSTAKA

2.1. Landasan Teori

2.1.1. Steganografi

Steganografi berasal dari bahasa yunani yaitu ”*Stegano*” artinya (tertutup) dan ”*Graphy*” artinya (menulis), yang berarti ”penulisan tersembunyi”. Steganografi adalah teknik menyembunyikan sebuah informasi rahasia atau pribadi dalam media digital, seperti gambar, video, atau audio, tanpa diketahui oleh pihak yang tidak berwenang. Berbeda dengan kriptografi yang menyamarkan isi pesan agar tidak dapat dibaca, steganografi bertujuan untuk menyembunyikan keberadaan pesan itu sendiri (Şahin et al., 2021). Dalam steganografi terdapat beberapa istilah penting, yaitu data rahasia (*Secret Data*) sebagai informasi yang akan disembunyikan, media penampung (*Cover Media*) sebagai tempat penyisipan informasi tersebut, *stego-media* yang merupakan media yang telah disisipi informasi rahasia, serta kunci steganografi yang bersifat opsional dan digunakan sebagai elemen tambahan dalam proses penyisipan maupun ekstraksi pesan (Kadhim et al., 2019).

Steganografi bekerja dengan cara menyisipkan informasi ke dalam media penampung (*Cover Media*) sehingga hasilnya disebut (*Stego-Media*). *Stego-Media* dapat dibagikan ke penerima untuk di ekstraksi kembali. Proses steganografi melibatkan media penampung (*Cover Media*), data rahasia (*Secret Data*), *Stego-Media* dan kunci steganografi.



Gambar 2.1 Proses Steganografi (Kadhim et al., 2019)

Gambar 2.1 merupakan proses steganografi yang terdiri dari dua tahap utama yaitu penyisipan (*embedding*) dan ekstraksi (*extraction*). Pada tahap *embedding*, media penampung (*Cover Media*) dan data rahasia (*Secret Data*) digabungkan menggunakan teknik steganografi, dengan opsi penggunaan kunci (*key*) sebagai elemen tambahan dalam proses penyisipan dan ekstraksi, sehingga menghasilkan *Stego-Media* (media yang sudah berisi data tersembunyi). *Stego-Media* ini kemudian dikirim melalui *communication Channel* ke penerima. Pada tahap *extraction*, *Stego-Media* diterima dan diproses untuk mengekstrak kembali *Secret Data*, dengan kemungkinan memerlukan *key* (opsional) untuk membuka informasi tersembunyi. Proses ini memungkinkan penyampaian informasi secara rahasia tanpa menarik perhatian pihak ketiga (Kadhim et al., 2019).

Steganografi dapat dikatakan baik apabila memenuhi beberapa faktor, yaitu *imperceptibility* (ketidakterlihatan), *Fidelity* (Keutuhan Media), *Robustness* (Ketahanan terhadap Modifikasi), dan *Recovery* (Kemampuan Pemulihan Pesan) (Minarni et al., 2023).

1. *Imperceptibility* (Ketidakterlihatan)

Imperceptibility adalah tingkat di mana steganografi tidak dapat dideteksi baik secara visual maupun melalui analisis statistik (Minarni et al., 2023).

2. *Fidelity* (Keutuhan Media)

Fidelity menunjukkan sejauh mana kualitas media stego tetap baik setelah penyisipan pesan. Meskipun ada perubahan akibat steganografi, gambar tetap memiliki tampilan yang tajam dan tidak mengalami distorsi visual yang signifikan (Minarni et al., 2023).

3. *Robustness* (Ketahanan terhadap Modifikasi)

Ketahanan terhadap modifikasi mencerminkan kemampuan steganografi untuk mempertahankan integritas data yang disisipkan meskipun terjadi perubahan pada media, seperti kompresi, *cropping*, atau serangan steganalisis (Minarni et al., 2023).

4. *Recovery* (Kemampuan Pemulihan Pesan)

Pesan yang disisipkan harus dapat diekstrak kembali dengan akurasi tinggi (Minarni et al., 2023).

2.1.2. *AVL TREE*

AVL Tree adalah pohon pencarian biner (*Binary Search Tree* - BST) yang seimbang, di mana selisih tinggi antara sub-pohon kiri dan sub-pohon kanan dari setiap simpul tidak boleh lebih dari 1. Jika perbedaan tinggi lebih dari 1, maka pohon harus dilakukan rotasi agar tetap seimbang. *AVL Tree* pertama kali diperkenalkan oleh Adelson-Velsky dan Landis pada tahun 1962 dan merupakan salah satu struktur data yang efisien dalam pencarian, penyisipan, dan penghapusan elemen (Brown, 2024).

Adapun karakteristik dari *AVL Tree* (Kushwah et al., 2021).

1. Keseimbangan Tinggi (*Balance Factor*)

Pada *AVL Tree*, selisih tinggi antara subpohon kiri dan subpohon kanan di setiap simpul tidak boleh lebih dari 1. Perbedaan tinggi ini direpresentasikan dalam suatu atribut keseimbangan (*Balance Factor*) dengan nilai -1, 0, atau +1, yang dihitung sebagai:

Balance Factor = tinggi subpohon kiri simpul − tinggi subpohon kanan simpul

Jika setelah operasi penyisipan atau penghapusan terdapat simpul dengan *Balance Factor* -2 atau +2, maka dilakukan operasi rotasi untuk mengembalikan keseimbangan pohon.

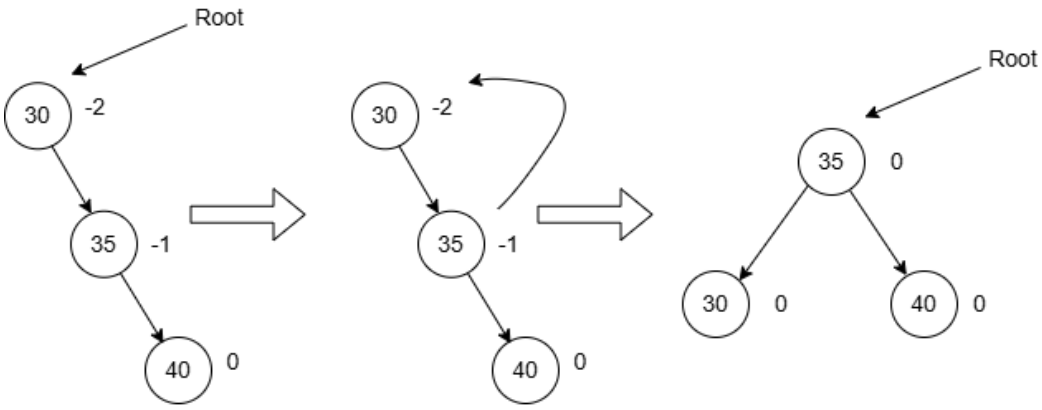
2. Rotasi

Terdapat empat jenis rotasi yang dapat dilakukan :

a. Rotasi Tunggal (*Single Rotation*)

1) Rotasi Kiri (*Left Rotation*)

Rotasi kiri dilakukan jika penambahan terjadi di sub-pohon kanan dari simpul kanan seperti pada gambar 2.2.

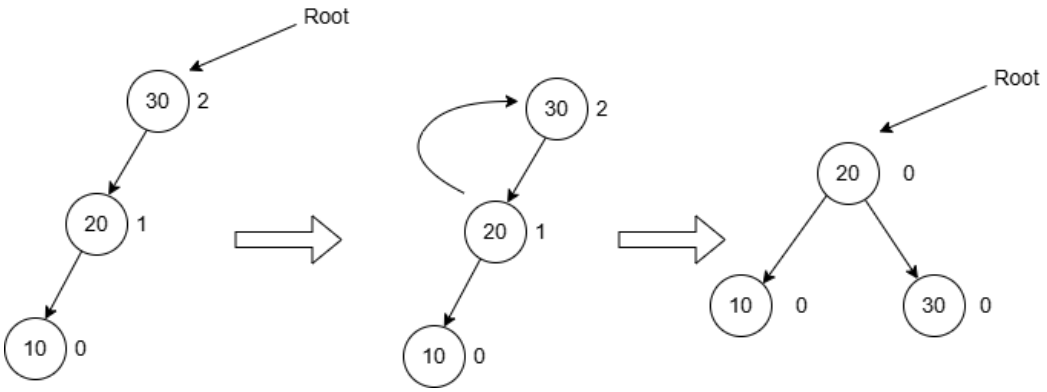


Gambar 2.2 Proses Rotasi Kiri (Brown, 2024)

Gambar 2.2 menunjukkan rotasi kiri (*Left Rotation*) untuk menangani ketidakseimbangan *Right-Right* (RR) *Imbalance* setelah penyisipan simpul 40. Sebelum rotasi, simpul 30 memiliki faktor keseimbangan -2, menandakan bahwa subpohon kanan terlalu tinggi, sementara simpul 35 memiliki faktor keseimbangan -1. Untuk menyeimbangkan pohon, dilakukan rotasi kiri pada simpul 30, sehingga 35 menjadi akar baru, 30 menjadi anak kiri 35, dan 40 tetap sebagai anak kanan 35. Setelah rotasi, faktor keseimbangan di semua simpul menjadi 0, mengembalikan keseimbangan *AVL Tree*.

2) Rotasi Kanan (*Right Rotation*)

Rotasi kanan dilakukan Jika penambahan terjadi di sub-pohon kiri dari simpul kiri seperti pada gambar 2.3.



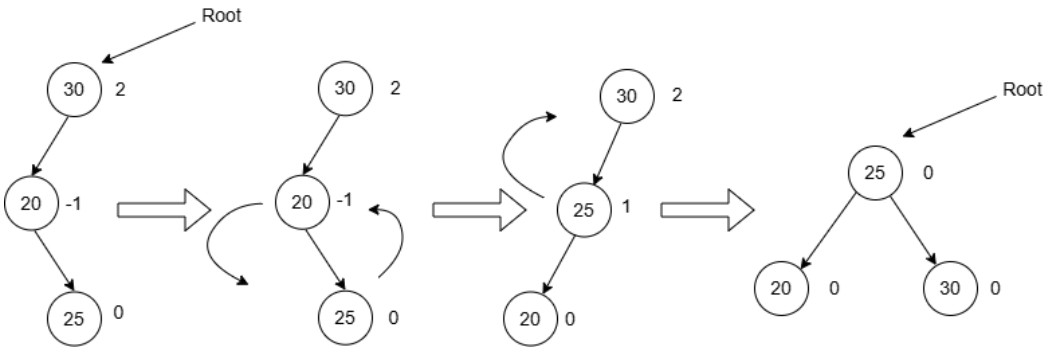
Gambar 2.3 Proses Rotasi Kanan (Brown, 2024)

Gambar 2.3 menunjukkan rotasi kanan (*Right Rotation*) untuk menangani ketidakseimbangan *Left-Left* (LL) *Imbalance* setelah penyisipan simpul 10. Sebelum rotasi, simpul 30 memiliki faktor keseimbangan 2, menandakan bahwa subpohon kiri terlalu tinggi dibandingkan dengan subpohon kanan, sementara simpul 20 memiliki faktor keseimbangan 1, yang menunjukkan bahwa ketidakseimbangan terjadi di subpohon kiri. Untuk menyeimbangkan pohon, dilakukan rotasi kanan pada simpul 30, sehingga 20 menjadi akar baru, 10 tetap sebagai anak kiri 20, dan 30 menjadi anak kanan 20. Setelah rotasi, faktor keseimbangan di semua simpul menjadi 0, mengembalikan keseimbangan *AVL Tree*.

b. Rotasi Ganda (*Double Rotation*)

1) Rotasi Kiri-Kanan (*Right-Left Rotation* / RL *Rotation*)

Rotasi Kiri-Kanan dilakukan Jika penambahan terjadi di sub-pohon kanan dari simpul kiri seperti pada gambar 2.4.

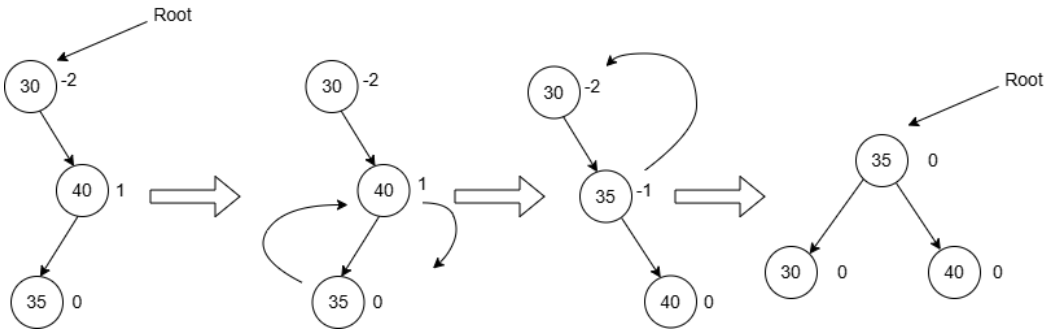


Gambar 2.4 Proses Rotasi Kiri-Kanan (Brown, 2024)

Gambar 2.4 menunjukkan proses rotasi kiri-kanan (*Left-Right Rotation*) untuk menangani ketidakseimbangan *Left-Right* (LR) *Imbalance* setelah penyisipan simpul 25. Awalnya, simpul 30 sebagai akar memiliki faktor keseimbangan 2, yang menunjukkan bahwa subpohon kiri lebih tinggi daripada subpohon kanan, sementara simpul 20 memiliki faktor keseimbangan -1, menandakan bahwa subpohon kanan lebih tinggi dibanding subpohon kiri. Karena pola ketidakseimbangan *Left-Right* (LR), pertama dilakukan rotasi kiri pada simpul 20, sehingga 25 menjadi akar dari subpohon kiri dengan 20 sebagai anak kirinya. Selanjutnya, dilakukan rotasi kanan pada simpul 30, sehingga 25 menjadi akar baru pohon, dengan 20 sebagai anak kiri dan 30 sebagai anak kanan. Setelah kedua rotasi ini, faktor keseimbangan semua simpul menjadi 0, mengembalikan keseimbangan *AVL Tree*.

2) Rotasi Kanan-Kiri (*Right-Left Rotation* / RL *Rotation*)

Rotasi Kanan-Kiri dilakukan Jika penambahan terjadi di sub-pohon kiri dari simpul kanan seperti pada gambar 2.5.



Gambar 2.5 Proses Rotasi Kanan-Kiri (Brown, 2024)

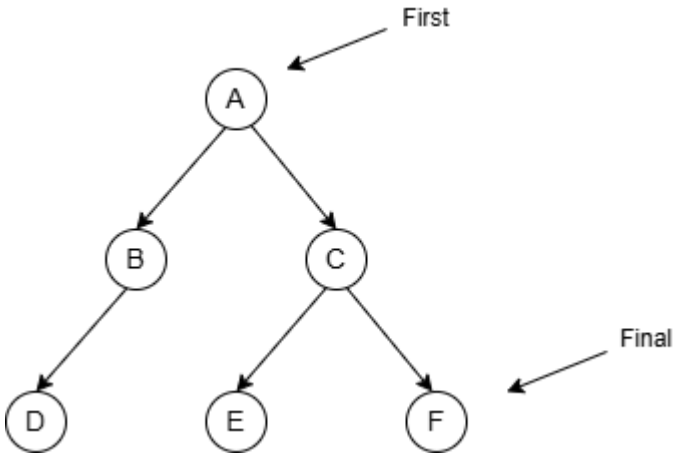
Gambar 2.5 menunjukkan proses rotasi kanan-kiri (*Right-Left Rotation*) untuk mengatasi ketidakseimbangan *Right-Left* (RL) *Imbalance* setelah penyisipan simpul 35. Awalnya, simpul 30 sebagai akar memiliki faktor keseimbangan -2, menunjukkan bahwa subpohon kanan lebih tinggi daripada subpohon kiri, sementara simpul 40 memiliki faktor keseimbangan 1, yang berarti subpohon kirinya lebih tinggi dibanding subpohon kanannya. Karena ini adalah pola *Right-Left* (RL) *Imbalance*, pertama dilakukan rotasi kanan pada simpul 40, sehingga 35 menjadi akar dari subpohon kanan dengan 40 sebagai anak kanannya. Selanjutnya, dilakukan rotasi kiri pada simpul 30, sehingga 35 menjadi akar baru pohon, dengan 30 sebagai anak kirinya dan 40 sebagai anak kanannya. Setelah kedua rotasi ini, faktor keseimbangan semua simpul menjadi 0, mengembalikan keseimbangan *AVL Tree*.

2.1.3. *Breadth First Search* (BFS)

Breadth First Search (BFS) adalah salah satu algoritma pencarian graf yang bekerja mengeksplorasi semua simpul pada satu level sebelum melanjutkan ke level berikutnya. BFS memulai pencarian dari simpul awal (*root*) dan mengunjungi

semua simpul yang bertetangga sebelum berpindah ke simpul yang lebih dalam (Refan Rahmat Fauzi et al., 2023).

BFS menggunakan struktur data antrian untuk menyimpan simpul yang akan dieksplorasi berikutnya. Gambar 2.6 merupakan cara kerja algoritma BFS (Qulub & Shanti Bhuana, 2024).



Gambar 2.6 Cara Kerja BFS (Qulub & Shanti Bhuana, 2024)

Berdasarkan gambar 2.6, BFS dimulai dari simpul akar A dan memasukkannya ke dalam antrean. Kemudian, BFS mengunjungi simpul-simpul pada level pertama, yaitu B dan C, dan menambahkannya ke dalam antrean. Setelah itu, BFS mengambil B dari antrean, memprosesnya, lalu menambahkan anaknya D ke antrean. Selanjutnya, BFS mengambil C, memprosesnya, dan menambahkan anaknya E dan F ke antrean. BFS kemudian mengunjungi simpul-simpul berikutnya, yaitu D, E, dan F, secara berurutan. Karena tidak ada anak tambahan, pencariannya berakhir. Hasil *traversal* BFS dalam urutan yang dikunjungi adalah A,B,C,D,E,F.

2.1.4. *PIXEL INDICATOR TECHNIQUE* (PIT)

Pixel Indicator Technique (PIT) adalah metode steganografi yang menggunakan nilai bit dari satu saluran warna (RGB) sebagai indikator untuk menentukan di mana data rahasia akan disisipkan dalam saluran warna lainnya. Teknik ini bertujuan untuk meningkatkan kapasitas dan keamanan (Pandey et al., 2019). Pixel Indicator Technique (PIT) terdiri dari dua bagian, yaitu *Indicator Channel* yang berfungsi menentukan saluran penyisipan data, dan *Data Channel* yang merupakan saluran tempat informasi rahasia disisipkan (Rahmani & Mohammadpour, 2017).

Pada penelitian (Rahmani & Mohammadpour, 2017) Metode PIT bekerja dengan menganalisis 3 *Most Significant Bit*s (MSB) dari saluran merah untuk menentukan saluran mana yang akan digunakan dalam penyisipan data. Aturan penyisipannya seperti pada Tabel 2.1.

Tabel 2.1 PIT 3-bit MSB (Rahmani & Mohammadpour, 2017)

3 MSB *Red Channel*	*Red*	*Green*	*Blue*
000	*No data*	*No data*	*No data*
001	*No data*	*No data*	2 Bit
010	*No data*	2 Bit	*No data*
011	*No data*	2 Bit	2 Bit
100	2 Bit	*No data*	*No data*
101	2 Bit	*No data*	2 Bit
110	2 Bit	2 Bit	*No data*
111	2 Bit	2 Bit	2 Bit

Berdasarkan Tabel 2.1 perubahan maksimal yang terjadi pada setiap piksel adalah 6 bit, karena data rahasia disisipkan menggunakan 2 *Least significant bits* (LSB) di satu atau lebih saluran warna (*Red*, *Green*, *Blue*) berdasarkan 3 *Most*

*Significant Bit*s (MSB) dari saluran merah. Jika 3 MSB = 111, maka ketiga saluran digunakan, sehingga total perubahan dalam satu piksel adalah 2 bit di *Red*, 2 bit di *Green*, dan 2 bit di *Blue*, menghasilkan perubahan maksimal 6 bit dari 24 bit per piksel.

Pada penelitian (Gutub et al., 2008) Metode PIT bekerja dengan menganalisis 2 *Least Significant Bit*s (LSB) dari saluran hijau untuk menentukan saluran mana yang akan digunakan dalam penyisipan data. Aturan penyisipannya seperti pada Tabel 2.2.

Tabel 2.2 *Pixel Indicator Technique* 2-bit LSB (Gutub et al., 2008)

2 LSB *Green Channel*	*Red*	*Blue*
00	*No data*	*No data*
01	*No data*	2 Bit
10	2 Bit	*No data*
11	2 Bit	2 Bit

Berdasarkan Tabel 2.2 menunjukan bahwa 2 *Least significant bits* (LSB) dari saluran hijau (*Green Channel*) digunakan sebagai *Indicator Channel* yang bertujuan untuk menentukan lokasi penyisipan data dalam saluran merah (*Red*) atau biru (*Blue*). Jika 2 bit terakhir dari saluran hijau (G) memiliki nilai bit 00, maka tidak ada bit pesan yang disisipkan. Jika 2 bit terakhir dari saluran hijau (G) memiliki nilai bit 01, maka 2 bit pesan akan disisipkan ke saluran biru (B). Jika 2 bit terakhir dari saluran hijau (G) memiliki nilai bit 10, maka 2 bit pesan akan disisipkan ke saluran merah (R). Sementara itu, Jika 2 bit terakhir dari saluran hijau (G) memiliki nilai bit 11, maka 2 bit pesan akan disisipkan ke saluran merah (R) dan biru (B).

Pada penelitian (Njoum et al., 2024) Metode PIT bekerja dengan menganalisis 2 *Least Significant Bits* (LSB) dari saluran hijau untuk menentukan saluran mana yang akan digunakan dalam penyisipan data. Aturan penyisipannya seperti pada Tabel 2.3.

Tabel 2.3 Pixel Indicator Technique 2-bit LSB (Njoum et al., 2024)

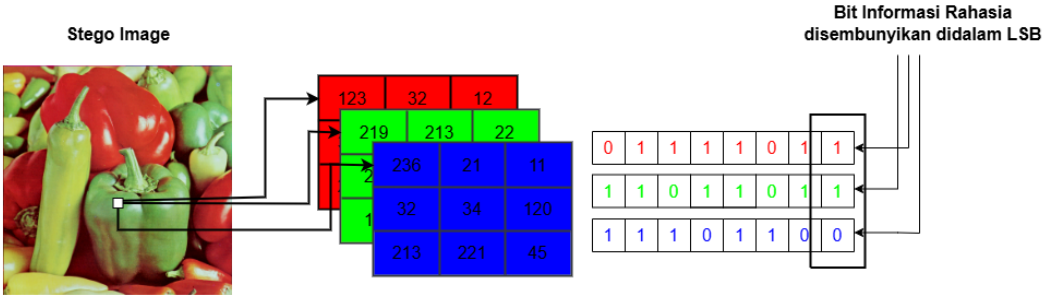
2 LSB *Green Channel*	*Red*	*Blue*
00 atau 11	2 Bit	*No data*
10 atau 01	*No data*	2 Bit

Berdasarkan Tabel 2.2 menunjukan bahwa 2 *Least significant bits* (LSB) dari saluran hijau (*Green Channel*) digunakan sebagai *Indicator Channel* yang bertujuan untuk menentukan lokasi penyisipan data dalam saluran merah (*Red*) atau biru (*Blue*), dengan perubahan maksimal 2 bit per piksel. Jika LSB = 00 atau 11, maka 2 bit data disisipkan dalam saluran merah, sementara saluran biru tetap tidak berubah. Sebaliknya, jika LSB = 10 atau 01, maka 2 bit data disisipkan dalam saluran biru, sedangkan saluran merah tetap utuh. Pendekatan ini membatasi perubahan dalam satu piksel agar tetap minimal, sehingga mengurangi dampak pada kualitas gambar dan menjaga nilai *Peak Signal-to-Noise Ratio* (PSNR) tetap tinggi. Dengan hanya mengubah 2 bit dari 24 bit per piksel.

2.1.5. *Least significant bit* (LSB)

Least significant bit (LSB) merupakan teknik steganografi sederhana yang digunakan untuk menyembunyikan informasi rahasia dalam berbagai media digital, seperti gambar, audio atau video dengan mengganti bit paling tidak signifikan dari suatu unit data dalam media digital seperti piksel dengan bit-bit informasi rahasia

yang akan disisipkan. Bit paling tidak signifikan memiliki pengaruh minimal terhadap representasi visual dari media tersebut, perubahan yang terjadi hampir tidak terdeteksi oleh mata manusia, sehingga informasi tersembunyi dapat disisipkan tanpa mengubah perubahan tampilan secara mencolok (Bansal et al., 2020). Pada citra digital berbasis RGB, informasi rahasia disisipkan dengan menggantikan bit paling tidak signifikan dari setiap saluran warna dalam piksel seperti pada gambar 2.7.



Gambar 2.7 Least Signifikan Bit (LSB) (Sugiarto et al., 2020)

Gambar 2.7 menunjukan bahwa setiap piksel terdiri dari tiga komponen warna diantaranya yaitu Merah (R), Hijau (G) dan Biru (B) yang direpresentasikan dalam format 8 bit biner. Teknik ini bekerja dengan mengambil bit terakhir dari setiap piksel dan menggantikannya dengan bit informasi rahasia. Perubahan ini sangat kecil, sehingga sulit dideteksi oleh mata manusia (Sugiarto et al., 2020).

2.1.6. Pengukuran Kualitas *Stego-image*

Berdasarkan penelitian dari (Humayrah et al., 2022) mengenai pengujian terhadap dua buah citra menyatakan bahwa setiap cover citra memiliki nilai MSE dan PSNR yang berbeda. Berdasarkan hasil analisis pengujian, ditemukan bahwa jika cover citra dan jumlah karakter teks yang digunakan sama, maka nilai MSE

dan PSNR yang dihasilkan juga akan sama. Namun, jika cover citra tetap sama tetapi jumlah karakter teks berbeda, maka nilai MSE dan PSNR yang dihasilkan juga akan berbeda. Dari hasil ini, dapat disimpulkan bahwa jika salah satu dari bahan uji coba ada yang berbeda maka akan menghasilkan MSE dan PSNR yang berbeda, sedangkan hasil MSE dan PSNR yang sama hanya diperoleh ketika semua bahan uji coba sama. Diketahui bahwa semakin kecil nilai MSE, maka kualitas *stego-image* semakin baik, dan semakin tinggi nilai PSNR, semakin baik pula kualitas *stego-image* yang diencoding.

Pengukuran kualitas *stego-image* sangat penting untuk menilai seberapa baik metode penyisipan data yang digunakan. Pengukuran kualitas gambar bertujuan untuk membandingkan antara gambar asli dan gambar yang telah disisipkan. Kualitas *stego-image* diukur secara kuantitatif menggunakan dua parameter utama, yaitu *Mean Square Error* (MSE) dan *Peak Signal-to-Noise Ratio* (PSNR) (Solichin, 2015).

1. *Mean Square Error* (MSE)

Mean Square Error (MSE) adalah rata-rata kesalahan kuadrat antara *cover image* dan *stego-image*. Semakin kecil nilai MSE, semakin kecil perubahan yang terjadi pada *stego-image*, sehingga *Fidelity* semakin tinggi. MSE dapat dirumuskan pada persamaan (2.1) :

$$MSE = \frac{1}{M.N}\sum_{x=1}^{M}\sum_{y=1}^{N}(S_{xy} - C_{xy})^2 \qquad \text{.....} \quad (2.1)$$

Dimana :

x dan y adalah koordinat dari suatu titik pada citra.

M dan N adalah ukuran atau dimensi dari citra.

S adalah citra hasil penyisipan pesan (*Stego-image*)

C adalah citra asli sebelum penyisipan pesan (Cover Image)

2. *Peak Signal-to-Noise Ratio* (PSNR)

PSNR adalah pengukuran rasio antara nilai maksimum sinyal yang diukur dengan tingkat gangguan atau derau yang memengaruhi sinyal tersebut. PSNR biasanya diukur dalam satuan desibel (dB). Semakin tinggi nilai PSNR, semakin baik kualitas *stego-image*. PSNR dirumuskan pada persamaan (2.2) :

$$\text{PSNR} = 10 \log_{10}(\frac{Max^2}{MSE}) \qquad \text{.....(2.2)}$$

Dimana :

Max adalah nilai bit piksel maksimum dalam citra.

MSE adalah *Mean Square Error* yang telah dihitung sebelumnya.

PSNR sering digunakan sebagai standar evaluasi kualitas citra hasil steganografi. Umumnya, nilai PSNR yang lebih besar dari 40 dB dianggap sangat baik, sedangkan nilai di bawah 30 dB menunjukkan adanya distorsi visual yang signifikan.

2.2. *State-of-the-Art*

State-of-the-art dalam penelitian ini merujuk pada berbagai metode steganografi berbasis *Least significant bit* (LSB) yang telah dikembangkan untuk meningkatkan keamanan terhadap deteksi oleh steganalisis. Beberapa penelitian menggunakan teknik seperti algoritma acak untuk menentukan lokasi penyisipan data secara lebih aman. Tabel 2.4 merangkum berbagai penelitian terkait, mencakup metode yang digunakan serta kelebihan dan kekurangannya.

Tabel 2.4 *State-of-the-Art*

No	**Nama Peneliti/Journal**		**Hasil Penelitian**
1	**Peneliti (tahun)**	(Njoum et al., 2024)	Penelitian ini menggunakan struktur data *AVL Tree* untuk melakukan seleksi piksel secara acak dalam menentukan lokasi penyisipan data pada steganografi berbasis *Least significant bit* (LSB), dengan tujuan mendistribusikan data secara lebih merata dan mengurangi pola yang dapat dikenali oleh steganalisis. Penyisipan data nya menggunakan *Pixel Indicator Technique* (PIT), dimana saluran hijau digunakan sebagai indikator, sementara 2 bit data disisipkan ke dalam saluran merah atau biru. Penelitian ini memiliki kelebihan yaitu mampu meningkatkan keamanan, serta tidak memerlukan kunci pembangkit,
	Judul	*High-SecuRed Image LSB Steganography Using AVL-Tree with Random RGB Channel Substitution*	

No	Nama Peneliti/Journal		Hasil Penelitian
	Algoritma/Metode	AVL Tree dan *Pixel Indicator Technique*	yang membuat ektraksi data lebih efisien. Meskipun pendekatan menggunakan AVL Tree untuk seleksi piksel acak serta *Pixel Indicator Technique* (PIT) dapat meningkatkan keamanan dan efisiensi ekstraksi data, metode ini masih memiliki kelemahan dalam hal kualitas *stego-image*. Hal ini disebabkan oleh perubahan nilai bit pada setiap piksel yang berada dalam rentang -3 hingga +3 bit akibat penyisipan 2 bit dalam satu saluran warna, yang dapat menimbulkan distorsi visual yang cukup signifikan dan berdampak pada penurunan nilai PSNR. Oleh karena itu, masih terdapat kebutuhan untuk mengembangkan metode penyisipan data yang mampu menjaga kualitas *stego-image* dengan cara meminimalkan perubahan nilai bit pada piksel, tanpa mengorbankan aspek keamanan dan efisiensi ekstraksi data.
2	**Peneliti (tahun)**	(Tiwari & Gangurde, 2021)	Penelitian ini menggunakan *Piksel Locator Sequence* (PLS) untuk menentukan lokasi penyisipan data dalam steganografi berbasis

No	Nama Peneliti/Journal		Hasil Penelitian
	Judul	*LSB Steganography Using Piksel Locator Sequence with AES*	*Least significant bit* (LSB), dengan tujuan menyebarkan data secara acak dan mengurangi kemungkinan deteksi oleh steganalisis. PLS dihasilkan secara acak menggunakan algoritma *Modern Fisher-Yates Shuffle* dan berfungsi sebagai kunci dalam proses ekstraksi, sehingga tanpa mengetahui urutan lokasi piksel, data tidak dapat diambil kembali. Selain itu, pesan rahasia yang disisipkan maupun urutan PLS dienkripsi menggunakan algoritma *Advanced Encryption Standard* (AES). Penyisipan data dilakukan dengan menggantikan bit paling tidak signifikan dari nilai RGB dalam urutan piksel yang ditentukan oleh PLS. Namun, dalam penelitian ini, proses ekstraksi memerlukan kunci yang harus dibagikan oleh pengirim kepada penerima, sehingga berisiko jika kunci tersebut jatuh ke tangan pihak yang tidak berwenang serta memengaruhi efisiensi proses ekstraksi data.
	Algoritma/Metode	*Piksel Locator Sequence* (PLS), *Modern Fisher-Yates Shuffle,* AES dan *Least significant bit* (LSB)	
3	**Peneliti (tahun)**	(Gahan & Devanagavi, 2020)	

No	Nama Peneliti/Journal		Hasil Penelitian
	Judul	*A Secure Steganography Model Using Random-Bit Select Algorithm*	Penelitian ini menggunakan *uniform random variable* untuk menentukan lokasi penyisipan data dalam steganografi berbasis *Least significant bit* (LSB), dengan tujuan untuk meningkatkan keamanan. Lokasi penyisipan data ditentukan secara acak dengan menggunakan kunci rahasia sebagai *seed* untuk menentukan posisi piksel yang akan dimodifikasi. Penyisipan data dilakukan menggunakan algoritma *Random-Bit Select* dengan menggantikan 1 hingga 4 bit dari setiap saluran warna (RGB) pada piksel, sesuai dengan jumlah bit yang ditentukan dalam parameter input. Namun, penelitian ini telah menerapkan lokasi penyisipan dan proses penyisipan data secara acak dengan tetap menggunakan kunci untuk ekstraksi. Jika kunci jatuh ke pihak ketiga, hal ini dapat menimbulkan risiko keamanan yang berpotensi mengungkap data tersembunyi serta memengaruhi efisiensi proses ekstraksi.
	Algoritma/Metode	*uniform random variable* dan *Random-Bit Select*	
4	**Peneliti (tahun)**	(Kordov & Zhelezov, 2021)	Penelitian ini menggunakan pseudo-*random* generator berbasis *Duffing Map* dan *Circle Map* untuk menentukan lokasi penyisipan data dalam steganografi berbasis *Least significant bit* (LSB).
	Judul	*Steganography in color images with random order of piksel*	

No	Nama Peneliti/Journal		Hasil Penelitian
		selection and encrypted text message embedding	Pendekatan ini, pemilihan piksel dan bit pesan rahasia dilakukan secara acak dan tidak berurutan, yang bertujuan untuk meningkatkan keamanan dengan mengurangi kemungkinan deteksi oleh teknik steganalisis. Penyisipan data dilakukan menggunakan metode LSB substitution, di mana 3 bit pesan disisipkan dalam setiap piksel, masing-masing satu bit dalam saluran merah (R), hijau (G), dan biru (B). Namun, dalam penelitian ini, proses ekstraksi bergantung pada kunci yang digunakan saat penyisipan, sehingga kunci tersebut harus dibagikan kepada penerima. Hal ini menimbulkan risiko keamanan jika kunci jatuh ke tangan pihak yang tidak berwenang.
	Algoritma/Metode	*pseudo-random generator* dan *Least significant bit (LSB).*	
5	**Peneliti (tahun)**	(Kurniasih et al., 2023)	

No	Nama Peneliti/Journal		Hasil Penelitian
	Judul	Penggabungan Affine Cipher dan *Least significant bit*-2 untuk Penyisipan Pesan Rahasia pada Gambar	Penelitian ini menggunakan *Linear Congruential Generator* (LCG) sebagai *Pseudo-Random Number Generator* (PRNG) untuk menentukan lokasi penyisipan data pada steganografi berbasis *Least significant bit*-2 (LSB-2). LCG digunakan untuk menghasilkan bilangan acak yang menentukan posisi piksel tempat data akan disisipkan, sehingga distribusi penyisipan menjadi acak dan sulit ditebak oleh steganalisis. Penyisipan data dilakukan menggunakan *Least significant bit*-2 (LSB-2), yaitu menggantikan 2 bit terakhir dari setiap saluran warna (R, G, B) dengan bit-bit pesan rahasia. Sebelum penyisipan, pesan dienkripsi menggunakan *Affine Cipher*. Metode ini membuat pesan rahasia lebih sulit untuk dideteksi oleh teknik steganalisis, sehingga meningkatkan keamanan. Namun, penggunaan kunci menimbulkan potensi risiko, karena kunci harus dibagikan kepada penerima yang berwenang, dan jika jatuh ke tangan yang tidak sah, pesan rahasia dapat diakses oleh pihak yang tidak berhak.
	Algoritma/Metode	*Linear Congruential Generator* (LCG), *Affine Cipher* dan *Least significant bit* (LSB)	
6	**Peneliti (tahun)**	(Deaz et al., 2025)	

No	Nama Peneliti/Journal		Hasil Penelitian
	Judul	Implementasi Keamanan Pada Citra Digital Menggunakan Algoritma Least Congruential Generator Dan *Least significant bit*	Penelitian ini menggunakan *Linear Congruential Generator* (LCG) untuk menentukan lokasi penyisipan data dalam steganografi berbasis *Least significant bit* (LSB), dengan tujuan untuk meningkatkan keamanan. Lokasi penyisipan ditentukan secara acak menggunakan LCG yang menghasilkan bilangan acak berdasarkan parameter tertentu. Penyisipan data dilakukan dengan menggantikan bit LSB dari piksel yang telah ditentukan, di mana pesan rahasia dikonversi ke dalam bentuk biner sebelum disisipkan. Dengan pendekatan ini, penyisipan tidak mengikuti pola tetap, yang dapat meningkatkan ketahanan terhadap analisis steganalisis berbasis statistik. Namun, penggunaan LCG dalam pemilihan lokasi penyisipan tetap memerlukan kunci rahasia yang harus dijaga dengan baik, karena kehilangan kunci dapat menghambat proses ekstraksi pesan dan meningkatkan risiko keamanan.
	Algoritma/Metode	*Linear Congruential Generator* (LCG) dan *Least significant bit* (LSB)	
7	**Peneliti (tahun)**	(Abbas et al., 2022)	

No	Nama Peneliti/Journal		Hasil Penelitian
	Judul	*Security and imperceptibility improving of image steganography using piksel allocation and random function techniques*	Penelitian ini menggunakan fungsi acak (*random function*) dan alokasi piksel ganjil/genap untuk menentukan lokasi penyisipan data dalam steganografi berbasis *Least significant bit* (LSB), dengan tujuan meningkatkan keamanan dan ketidakterlihatan data tersembunyi. Lokasi penyisipan data ditentukan secara acak dengan membagi gambar menjadi 64 blok kecil dan memilih piksel di dalam blok berdasarkan algoritma fungsi acak. Penyisipan data dilakukan dengan metode LSB, di mana bit pesan disisipkan pada layer LSB pertama atau kedua tergantung pada karakteristik piksel ganjil/genap. Namun, metode ini masih mengandalkan kunci dari fungsi acak untuk menentukan lokasi penyisipan, sehingga jika kunci diketahui atau ditebak oleh pihak ketiga, pesan rahasia dapat diekstraksi tanpa izin.
	Algoritma/Metode	*random function* dan *Least significant bit* (LSB)	
8	**Peneliti (tahun)**	(Ehsan Ali et al., 2021)	

No	Nama Peneliti/Journal		Hasil Penelitian
	Judul	*A LSB Based Image Steganography Using Random Piksel and Bit selection for High Payload*	Penelitian ini menggunakan *Pseudo Random Number Generator* (PRNG) untuk menentukan lokasi penyisipan data dalam steganografi berbasis *Least significant bit* (LSB), dengan tujuan meningkatkan keamanan dan kapasitas penyisipan data. Lokasi penyisipan data ditentukan dengan memilih piksel secara acak menggunakan PRNG pertama, yang memastikan bahwa setiap piksel hanya digunakan satu kali untuk penyisipan data. Penyisipan data dilakukan dengan metode 3-3-2 LSB substitution, di mana setiap byte pesan dibagi menjadi tiga kelompok, yaitu 3 bit disisipkan pada saluran merah (R), 3 bit pada hijau (G), dan 2 bit pada biru (B). Selain itu, PRNG kedua digunakan untuk memilih secara acak posisi bit dalam 5 *Most Significant Bits* (MSB) dari setiap saluran warna sebelum bit pesan disisipkan ke dalam LSB menggunakan operasi XOR. Namun, metode ini masih bergantung pada kunci PRNG untuk proses ekstraksi, sehingga jika kunci bocor atau diketahui pihak ketiga, pesan tersembunyi dapat diungkap dengan mudah.
	Algoritma/Metode	*Pseudo Random Number Generator* (PRNG) dan *Least significant bit* (LSB)	

No	Nama Peneliti/Journal		Rencana Penelitian
9	**Peneliti (tahun)**	Penelitian ini	Penelitian ini bertujuan untuk mengembangkan metode steganografi dengan mengombinasikan AVL Tree dan modifikasi Pixel Indicator Technique (PIT) guna meningkatkan kualitas *stego-image*. AVL Tree digunakan untuk menentukan lokasi penyisipan data secara acak tanpa memerlukan kunci pembangkit, sedangkan modifikasi Pixel Indicator Technique (PIT) diterapkan untuk proses penyisipan data dengan tujuan meminimalkan perubahan nilai bit pada setiap piksel. Penelitian ini memodifikasi PIT dengan menyebarkan penyisipan bit secara merata ke dua saluran warna, yaitu menyisipkan 1 bit pada saluran merah dan 1 bit pada saluran biru ketika nilai bit LSB dari saluran hijau (sebagai indikator) adalah 00 atau 11, serta menyisipkan 2 bit pada saluran biru ketika nilai bit LSB dari saluran hijau adalah 01 atau 10. Tujuan modifikasi ini adalah untuk mengurangi perubahan nilai bit piksel, sehingga mengurangi distorsi visual yang mampu meningkatkan nilai *Peak Signal-to-Noise Ratio* (PSNR) dan menurunkan nilai *Mean Square Error* (MSE). Penelitian ini berfokus
	Judul	Steganografi Dengan Kombinasi Avl Tree Dan Modifikasi Pixel Indicator Technique (Pit) Untuk Meningkatkan Kualitas *Stego-Image*	
	Algoritma/Metode	AVL Tree dan Modifikasi *Pixel Indicator Technique* (PIT)	

No	Nama Peneliti/Journal		Rencana Penelitian
			untuk meningkatkan kualitas *stego-image* pada steganografi yang menggunakan metode acak tanpa kunci pembangkit.

Tabel 2.4 merupakan penelitian terdahulu yang mencakup berbagai metode steganografi berbasis *Least significant bit* (LSB) yang bertujuan untuk meningkatkan keamanan dari deteksi oleh steganalisis. Penelitian (Ehsan Ali et al., 2021),(Abbas et al., 2022),(Deaz et al., 2025),(Kurniasih et al., 2023),(Kordov & Zhelezov, 2021),(Gahan & Devanagavi, 2020),(Tiwari & Gangurde, 2021) telah melakukan penelitian dengan menggunakan metode acak dalam menentukan lokasi penyisipan data guna meningkatkan keamanan steganografi. Salah satu pendekatan yang digunakan adalah *Linear Congruential Generator* (LCG), pseudo-*random* generator atau *Piksel Locator Sequence* (PLS), yang bertujuan untuk menyebarkan data secara tidak beraturan sehingga lebih sulit dideteksi oleh steganalisis. Namun, metode ini masih memiliki kelemahan karena bergantung pada kunci pembangkit yang harus dibagikan oleh pengirim kepada penerima, yang menimbulkan risiko keamanan dalam ekstraksi data jika jatuh kepada pihak yang tidak berwenang, serta dapat memengaruhi efisiensi ekstraksi data.

Solusi terhadap ketergantungan pada kunci pembangkit, penelitian (Njoum et al., 2024) mengusulkan metode steganografi dengan menggunakan struktur data *AVL Tree* untuk menentukan lokasi penyisipan secara acak tanpa memerlukan kunci pembangkit. Proses penyisipan data dilakukan dengan menggunakan teknik Pixel Indicator Technique (PIT), yaitu dengan menyisipkan 2 bit data ke dalam saluran merah atau biru berdasarkan nilai indikator pada saluran hijau. Pendekatan ini mampu meningkatkan keamanan dan memudahkan proses ekstraksi tanpa kunci pembangkit.

Penelitian (Njoum et al., 2024) masih menghadapi tantangan dalam hal perubahan piksel yang cukup besar, terutama karena penyisipan 2 bit dalam satu saluran menyebabkan perubahan nilai bit piksel dalam rentang -3 hingga +3 bit. Akibat perubahan nilai bit piksel tersebut, distorsi visual meningkat dan memengaruhi kualitas stego-image. Oleh karena itu, Penelitian ini bertujuan untuk mengimplementasikan steganografi dengan dengan mengoptimalkan metode yang ada melalui penyisipan data yang lebih halus, menggunakan kombinasi AVL Tree dan modifikasi *Pixel Indicator Technique* (PIT). Meskipun pada beberapa piksel terjadi perubahan dalam rentang -3 hingga +3 bit, tetapi pendekatan ini mampu menghasilkan perubahan nilai piksel dalam rentang -2 hingga +2 bit. Strategi ini dapat meningkatkan kualitas *stego-image.*

2.3. Matrix Penelitian

Tabel 2.5 Matrix Penelitian

No	Penelitian	Metode Lokasi Penyisipan	Metode Penyisipan Data	Kapasitas Penyisipan per Piksel	Perubahan nilai bit per Piksel	Kunci Pembangkit
1	(Njoum et al., 2024)	*AVL Tree*	*Pixel Indicator Technique* (PIT)	2-Bit	(-3 hingga +3)	Tidak Ada
2	(Tiwari & Gangurde, 2021)	*Piksel Locator Sequence* (PLS)	*Least significant bit* (LSB)	3-Bit	(-3 hingga +3)	Ada
3	(Gahan & Devanagavi, 2020)	*uniform random variable*	*Random-Bit Select*	3,6,9 atau 12-Bit	(-3 hingga +3), (-9 hingga +9), (-21 hingga +21) atau (-45 hingga +45)	Ada
4	(Kordov & Zhelezov, 2021)	*pseudo-random generator* (PRG)	*Least significant bit* (LSB)	3-Bit	(-3 hingga +3)	Ada
5	(Kurniasih et al., 2023)	*Pseudo Random Number Generator* (PRNG)	*Least significant bit-2* (LSB-2)	3-Bit	(-12, -8, -4, +4, +8 atau +12)	Ada

No	Penelitian	Metode Lokasi Penyisipan	Metode Penyisipan Data	Kapasitas Penyisipan per Piksel	Perubahan nilai bit per Piksel	Kunci Pembangkit
6	(Deaz et al., 2025)	*Linear Congruential Generator* (LCG)	*Least significant bit* (LSB)	3-Bit	(-3 hingga +3)	Ada
7	(Abbas et al., 2022)	*random function*	*Least significant bit* (LSB)	3-Bit	(-3 hingga +3)	Ada
8	(Ehsan Ali et al., 2021)	*Pseudo Random Number Generator* (PRNG)	3-3-2 *Least significant bit* (LSB)	8-Bit	(-17 hingga +17)	Ada
9	Penelitian ini	*AVL Tree*	*Pixel Indicator Technique* (PIT)	2-Bit	(-2 hingga +2) atau (-3 hingga +3)	Tidak Ada

Pada Tabel 2.5 menunjukkan perbedaan antara penelitian ini dengan penelitian terkait lainnya. Penelitian sebelumnya yaitu penelitian (Ehsan Ali et al., 2021),(Abbas et al., 2022),(Deaz et al., 2025),(Kurniasih et al., 2023),(Kordov & Zhelezov, 2021),(Gahan & Devanagavi, 2020),(Tiwari & Gangurde, 2021) menggunakan metode acak dengan kunci pembangkit dalam menentukan lokasi penyisipan data. Penggunaan kunci ini menimbulkan potensi risiko, karena kunci harus dibagikan kepada penerima yang berwenang. Apabila kunci tersebut jatuh ke tangan yang tidak berhak, maka pesan rahasia dapat dengan mudah diakses dan disalahgunakan oleh pihak yang tidak sah. Sementara itu, Penelitian (Njoum et al., 2024) menerapkan metode acak tanpa kunci memiliki kelemahan berupa

perubahan piksel yang cukup besar, yang dapat berdampak pada kualitas *stego-image*. Penelitian ini memiliki keterbaruan dalam hal kualitas *stego-image*, yaitu dengan menurunkan perubahan nilai bit piksel tanpa mengurangi kapasitas penyisipan data. Dengan teknik penyisipan yang lebih optimal, penelitian ini mempertahankan kapasitas penyisipan data yang sama dengan metode acak tanpa kunci sebelumnya, tetapi dengan distorsi yang lebih kecil, sehingga meningkatkan kualitas gambar hasil steganografi.