Robust Steganography Encoder(B15)

By Web Wizards :-

Anmol Chaudhary

Shivam Rana

Akshit

Aryan Govind Rao

Problem Statement

- Traditional steganography fails after JPEG compression
- Hidden data gets destroyed during social media transmission
- Pixel-based embedding is fragile and easily detectable
- No tolerance to resizing or recompression
- Lack of message confidentiality and integrity
- No intelligent region selection for secure embedding
- Most tools are not scalable or web-deployable
- Not suitable for real-world secure communication

What is Steganography ?

- Steganography means hiding a secret message.
- The message is hidden inside an image, audio, or video.
- The file looks normal to everyone.
- No one can see that a message is hidden.
- Only the receiver can extract the message.
- It is used for secure and private communication.

Clear Domain Positioning

- Steganography is not just “hiding text in image”.

It belongs to:

- Cybersecurity
- Digital Forensics
- Secure Communication
- Data Protection
- Military / Intelligence Communication

Solution’s Offered:

- It keeps messages safe and private.
- No one can see the hidden message.
- The image looks completely normal.
- It is simple and easy to use.
- It works without internet after loading.
- Useful for students, journalists, and professionals.

How It Works

- User uploads an image.
- User enters a secret message.
- The message is converted into binary (0s and 1s).
- The binary data is hidden inside the image pixels.
- The encoded image is downloaded and shared.
- The receiver uploads the image to decode the message.

Real-World Industrial Applications

- Cybersecurity

National Security Agency and defense agencies use covert communication methods.

- Banking & Finance

Hidden transaction verification signatures inside documents.

- Journalism

Whistleblowers securely sending hidden data under surveillance.

- Military Communication

Covert data transfer inside normal images.

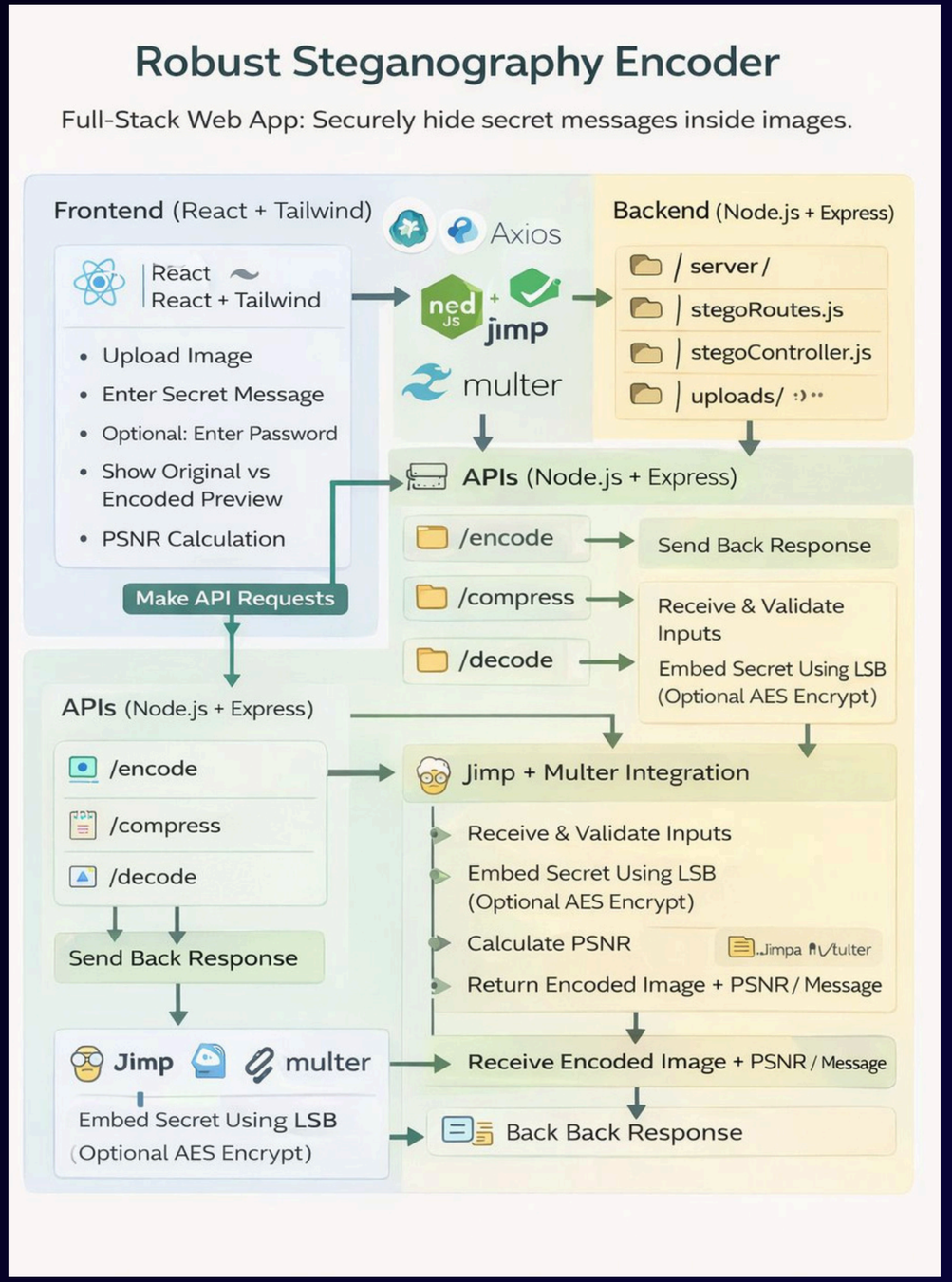
- DRM & Copyright

Embedding hidden ownership marks in media.

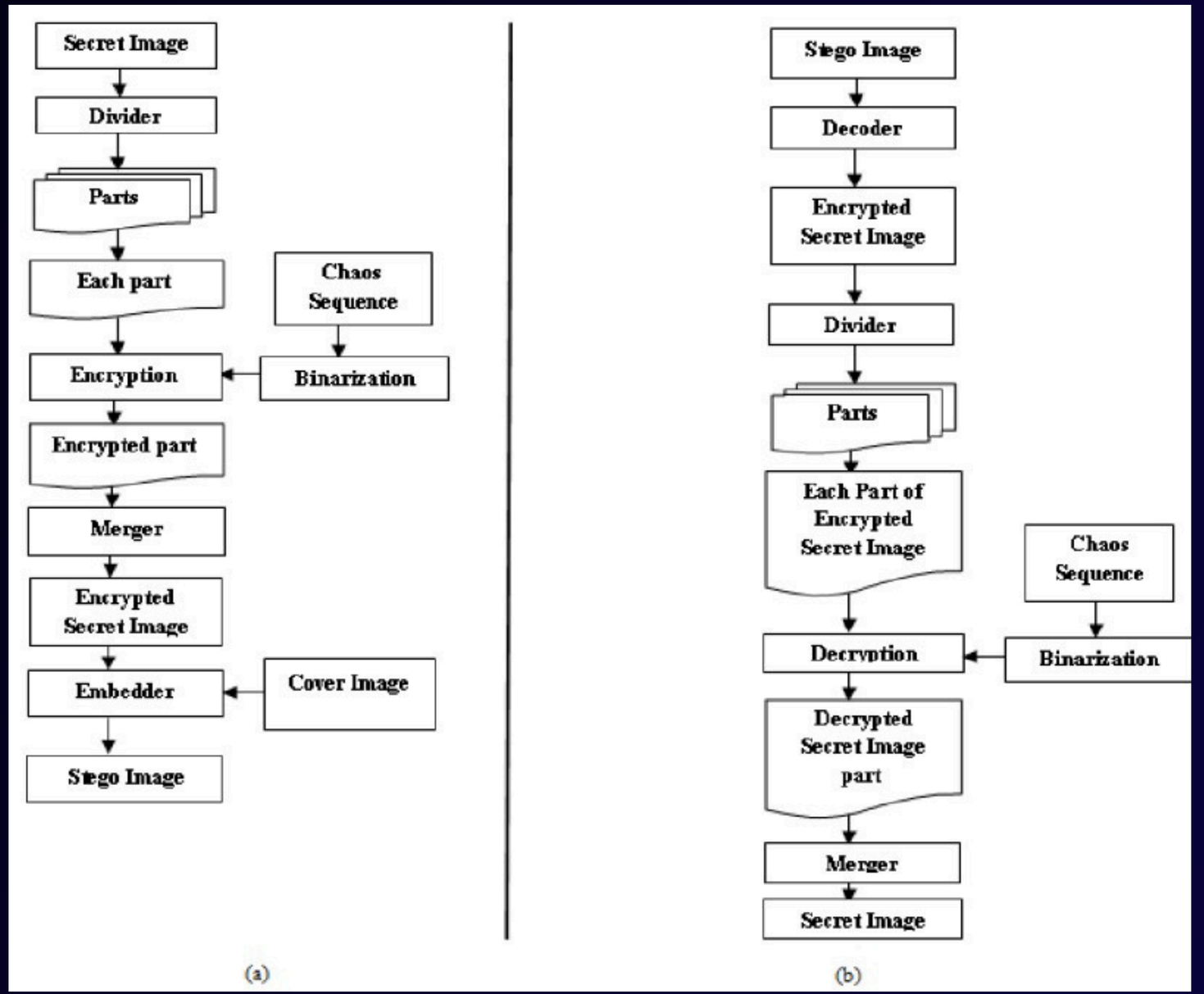
Tech Stack Used :

- React – To build the frontend user interface.
- Node.js – To run the backend server.
- Express.js – To handle API routes.
- Jimp – To process and modify images.
- Multer – To upload image files.
- AES Encryption – To secure the secret message with a password.
- ML Enhanced secure embedding

Flowchart


Robust Steganography Encoder
Full-Stack Web App: Securely hide secret messages inside images.
Frontend (React + Tailwind)
Axios
React
React + Tailwind
Upload Image
Enter Secret Message
Optional: Enter Password
Show Original vs Encoded Preview
PSNR Calculation
Make API Requests
jimp
multer
Backend (Node.js + Express)
/ server /
stegoRoutes.js
stegoController.js
uploads/
APIs (Node.js + Express)
/encode
Send Back Response
/compress
/decode
Receive & Validate Inputs
Embed Secret Using LSB (Optional AES Encrypt)
APIs (Node.js + Express)
/encode
/compress
/decode
Send Back Response
Jimp + Multer Integration
Receive & Validate Inputs
Embed Secret Using LSB (Optional AES Encrypt)
Calculate PSNR
Return Encoded Image + PSNR / Message
Jimp
multer
Embed Secret Using LSB (Optional AES Encrypt)
Receive Encoded Image + PSNR / Message
Back Back Response

Architecture



Novelty

1) Compression-Resilient Encoding

- Moving Beyond Basic LSB
- Traditional Approach
- Pixel-based LSB hiding
- Works mainly on PNG
- Fails after JPEG compression
- No resilience to recompression

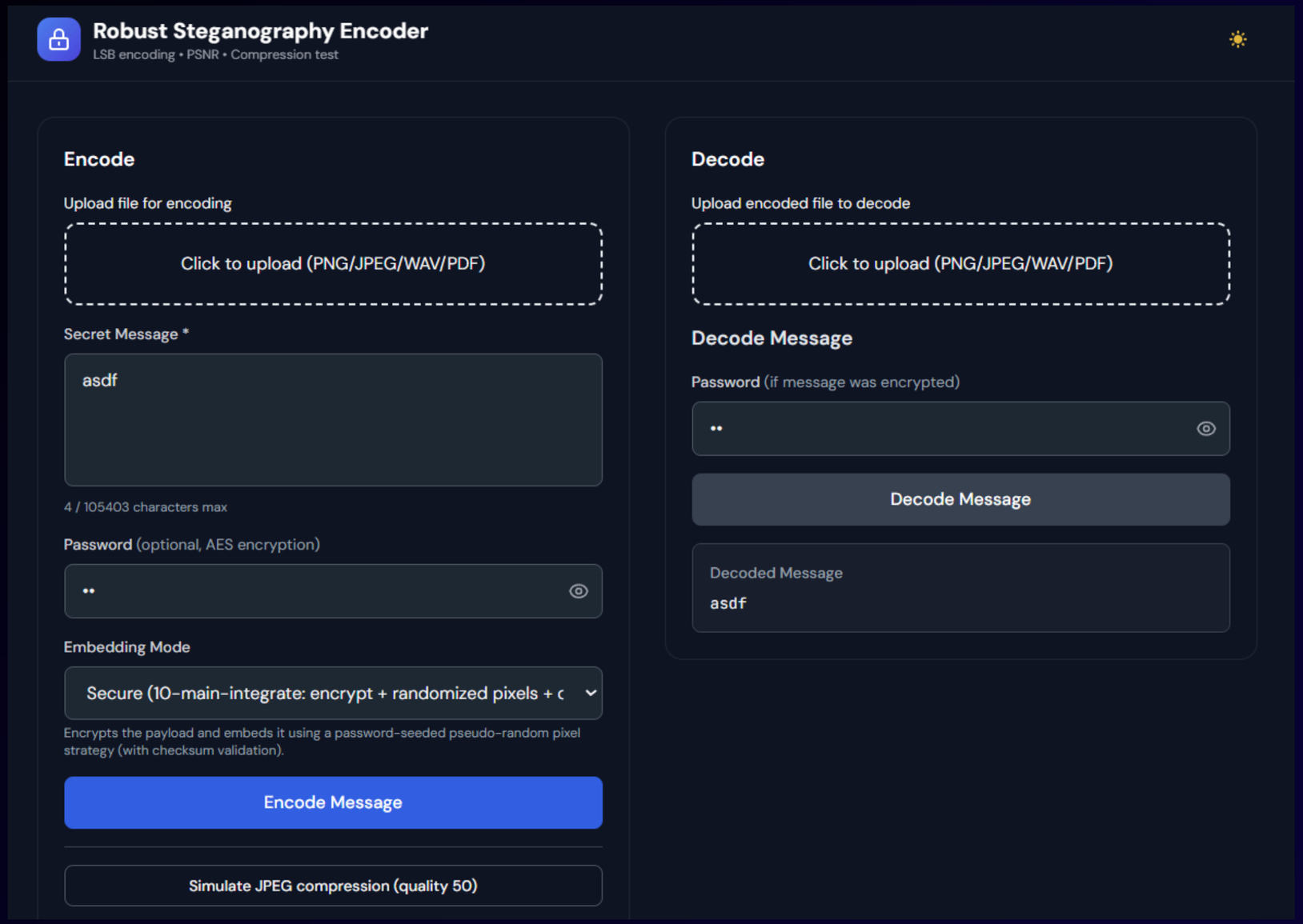
Novelty

2) Intelligent & Secure Embedding

Adaptive Region-Based Embedding, Instead of random hiding, we:

- Analyze edge density
- Detect textured/noisy regions
- Avoid flat regions (sky, walls)
- Embed where compression impact is minimal

User Interface



Robust Steganography Encoder

LSB encoding • PSNR • Compression test

Secure (10-main-integrate: encrypt + randomized pixels + c

Encrypts the payload and embeds it using a password-seeded pseudo-random pixel strategy (with checksum validation).

Encode Message

Simulate JPEG compression (quality 50)

PSNR (Peak Signal-to-Noise Ratio)

79.98 dB (Excellent quality)

Higher PSNR = less visible distortion. Typically 40+ dB is imperceptible.

Download Encoded

Embedding analysis

Mode: standard

Preview

Thank You