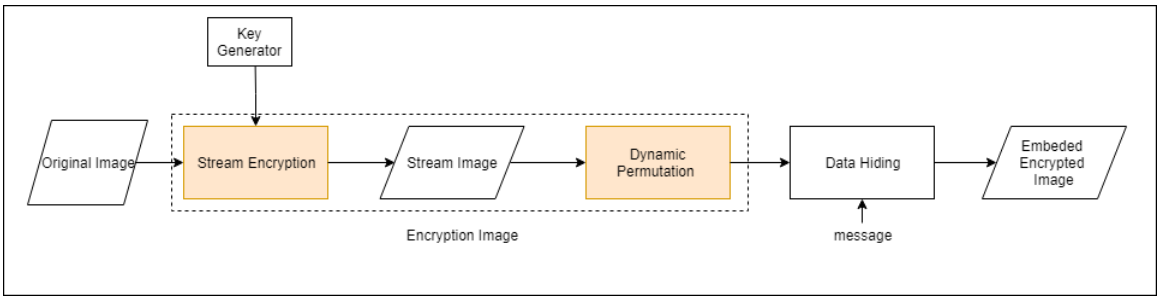

Gambar 2. Proses *Embedding*

3.1 Penyisipan (Embedding)

Proses embedding terdiri dari dua proses yaitu enkripsi gambar dan penyembunyian data, seperti yang ditunjukkan pada gambar 2. Karena korelasi antara piksel tetangga harus dipertahankan setelah enkripsi gambar maka kita dapat menerapkan RDH secara mandiri, sedemikian rupa sehingga penerima dapat mendekripsi gambar terenkripsi tanpa mengekstraksi pesan atau mendekripsi gambar terenkripsi setelah mengekstraksi pesan yang disisipkan.

3.1.1 Algoritma Enkripsi Gambar

Algoritma enkripsi terdiri dari dua proses, yaitu *stream encryption* dan permutasi. Untuk menjaga korelasi antara piksel tetangga setelah enkripsi gambar maka metode *Huang, Shi* digunakan.

Proses *stream* enkripsi mirip dengan yang diusulkan oleh *Huang, Shi* dkk dimana kami menerapkan kerangka kerja yang diusulkan oleh *Huang, Shi* . Dalam proses enkripsi, gambar sampul I dibagi menjadi N sub-blok yang tidak saling tumpang tindih (*non-overlapping*) (B_1, B_2,..., B_N). Ukuran sub-blok yang digunakan dalam penelitian ini adalah *3 x 3* dan penomoran blok dimulai dari blok paling kiri ke kanan. Kemudian, kami memiliki sub-blok N dan 9 piksel di setiap sub-blok. Didefinisikan $P_{i,j}$ ($1 \leq i \leq N$, $1 \leq j \leq 9$) adalah nilai piksel dari j^{th} pixel dalam sub-blok B_i, di mana i adalah indeks sub-blok, dan j adalah indeks piksel dalam sub-blok B_i.

Untuk menghasilkan kunci enkripsi K_i (kunci untuk mengenkripsi blok B_i) maka algoritma penjadwalan kunci RC4 digunakan. Untuk mempertahankan korelasi antara piksel tetangga setelah enkripsi gambar, setiap sub-blok dienkripsi dengan kunci yang sama. Pada proses mengenkripsi sub-blok B_i, setiap Pixel $P_{i,j}$ dari gambar sampul dilakukan operasi *ex-or* terhadap kunci K_i yang dihasilkan melalui proses sebelumnya, seperti yang ditunjukkan dalam persamaan 1.

$$E_{i,j} = P_{i,j} \oplus K_i (1 \leq i \leq N, 1 \leq j \leq 9) \tag{1}$$

Pada proses permutasi blok, kami menggunakan *polynomial congruence* (seperti yang ditunjukkan dalam persamaan (2), dimana g adalah angka antara 1 dan *P-1* dan harus relatif prima dengan P, dan x adalah indeks dari subblok. Untuk permutasi subblok, sejumlah N buah Y_i harus dihasilkan. Lalu kami menukar semua piksel disub-blok x dengan sub-blok Y_i yang telah dihasilkan. Pada langkah ini, kami hanya melakukan perubahan terhadap urutan sub-blok. Sementara itu, urutan piksel dalam setiap sub-blok tetap sama.

$$Y_i = g^x \ mod \ P \tag{2}$$

3.1.2 Penyembunyian Data (*Data Hiding*)

Pada fase ini, kami menggunakan algoritma RDH untuk menyisipkan pesan ke dalam gambar terenkripsi. Setelah proses enkripsi gambar, titik-titik perbedaan histogram pada gambar terenkripsi berkisar pada 0 dan $\pm$ 1 sehingga dapat diartikan bahwa korelasi antara piksel tetangga masih dipertahankan [2]. Dengan demikian, kita dapat dengan mudah menerapkan algoritma RDH ke dalam gambar terenkripsi. Dalam penelitian ini, kami menggunakan metode DHS. Kami menggunakan DHS karena kapasitas embeddingnya yang besar dan ketepatan yang tinggi. Gagasan utama dari metode ini adalah mengeksplorasi korelasi antara piksel tetangga dalam gambar sampul. Pada awalnya, Kami membagi gambar terenkripsi ke dalam sub-blok dengan ukuran 3 x 3 dan pixel dalam gambar terenkripsi diwakili oleh $C_{i,j}$ ($1 \leq \mathrm{i} \leq N_E$, $1 \leq \mathrm{j} \leq 9$), di mana N_E adalah jumlah sub-blok.

Untuk menghindari *overflow* atau *underflow* selama proses penyisipan, kami memodifikasi piksel yang memiliki nilai piksel 0 atau 255 dan untuk membedakan piksel yang dimodifikasi dengan yang tidak di modifikasi maka kami mencatatnya kedalam peta lokasi L (diinisialisasi menjadi kosong) sebagaimana dibahas dalam [7].