ENKRIPSI CITRA MENGGUNAKAN ALGORITMA KUBUS RUBIK DENGAN PEMBANGKIT KUNCI MD5

Yoga Andri Primadhana[1], Rosa Andrie Asmara[2], Ariadi Retno Tri Hayati Ririd[3]

[1,2,3] Teknik Informatika, Teknologi Informasi, Polteknik Negeri Malang
[1] yogandri@gmail.com, [2] rosaandrie@gmail.com, [3] faniri4education@gmail.com

Abstrak

Perkembangan teknologi memberikan pilihan untuk pegiriman dan penyimpanan informasi. Informasi tak hanya disimpan di komputer, namun dapat dikirimkan dan disimpan di internet. Informasi juga tak hanya disimpan pada tulisan, namun bisa dilakukan dalam bentuk citra digital, meskipun keamanan yang diberikan masih kurang. Hal ini berakibat rentannya pencurian citra oleh pihak ketiga untuk kepentingan pribadi sehingga bisa merugikan pemilik citra. Penelitian ini menawarkan untuk memberikan pengembangan keamanan pada citra digital. Keamanan tersebut berasal dari enkripsi citra dengan menggunakan algoritma kubus rubik. Keamanan diawali dengan pembangkitan deret kunci baris dan kunci kolom dengan fungsi MD5 berdasarkan kata kunci, kemudian dilakukan pengacakan pada tiap piksel secara horizontal dan vertikal sesuai dengan kunci kolom. Citra yang telah teracak akan mengimplementasikan gerbang XOR pada tiap pikselnya. Sehingga algoritma kubus rubik akan menghasilkan citra terenkripsi. Analisa yang dilakukan adalah tingkat keberhasilan proses enkripsi dan proses dekripsi, senstifitas kunci, serangan citra, kecepatan proses, tipe file penyimpanan, dan media pengiriman yang digunakan. Algoritma kubus rubik disimpulkan aman terhadap serangan kunci dengan parameter perbedaan 1 bit pada kunci baris dan kunci kolom yang digunakan. Penyimpanan tipe PNG memiliki keuntungan yang terbanyak, karena file PNG memiliki ukuran paling kecil dan dapat memberikan hasil dekripsi dengan kecocokan 100% dengan citra asli.

Kata kunci : **kriptografi, keamanan, pengolahan citra digital, kubus rubik, XOR**

1. Pendahuluan

Berbagi dan penyimpanan informasi semakin mudah dengan kemajuan teknologi. Informasi tak hanya pada teks, namun bisa berbentuk citra digital. Penyimpanan dan pengiriman informasi tersebut juga bisa dilakukan melalui media internet. Namun, dengan kemudahan yang didapat, terdapat pula kelemahan yang mengikutinya. Media pengiriman dan penyimpanan melewati internet termasuk media yang tidak aman. Dengan demikian, keamanan citra digital akan menjadi krusial jika terdapat informasi penting di dalamnya. Keamanan lebih diberikan untuk melindungi informasi yang terkandung di dalamnya agar tidak bisa dimanfaatkan oleh pihak yang tidak bertanggung jawab. Keamanan yang diberikan adalah dengan cara pengacakan posisi piksel pada citra. Cara tersebut berdasarkan permainan kubus rubik yang ditemukan oleh Erno Rubik pada tahun 1974. Pada permainan tersebut dimungkinkan adanya jutaan pergeseran dan perputaran yang bisa dilakukan. Dengan kemungkinan tersebut, maka akan sangat mungkin diimplementasikan pada citra digital untuk memberikan keamanan pada citra digital. Pergeseran bisa dilakukan di setiap baris dan kolom citra, kea rah kiri maupun kanan, atau ke atas serta ke bawah. Algoritma tersebut diimplementasikan sebelum citra tersimpan dan dikirimkan. Dengan demikian, meskipun citra dicuri, diambil, disalin pada saat melewati jaringan internet, pelaku tidak akan mendapatkan informasi yang terkandung di dalamnya. Proses enkripsi dan proses dekripsi hanya dilakukan oleh pihak yang berkepentingan menggunakan kunci simetris yang diketahui oleh kedua belah pihak.

2. Landasan Teori

2.1 Enkripsi

Enkripsi adalah sebuah proses yang melakukan perubahan sebuahkode yang bisa dimengerti menjadi sebuah kode yang tidak bisadimengerti (tidak terbaca). Enkripsi dapat diartikan sebagai kodeatau *chiper*. Sebuah chiper mengunakan suatu algoritma yang dapatmengkodekan semua aliran data (*stream*) bit dari sebuah pesanmenjadi *cryptogram* yang tidak dimengerti (*unintelligible*). Enkripsi dimaksudkan untuk melindungi informasi agar tidak terlihat oleh orang atau pihak yang tidak berhak.

2.2 Enkripsi Algoritma Kubus Rubik

Algoritma *Rubic'c cube* berkerja dengan menggunakan prisip dari kubus rubic, Dimisalkan I adalah citra digital dengan α-bit dan ukuran panjang x lebar adalah M x N. Nilai matrik piksel dari I diwakili dengan notasi Io, K untuk kunci. Proses Enkripsi merupakan proses untuk melakukan pengacakan Matrik Io dengan menggunakan metode Rubic's cube, berikut tahapan dari proses Enkripsi pada agoritma *Rubic's cube:*

1. Inisialisasi bilangan kunci secara acak pada dua vektor K_R dan K_C dengan panjang masing-masing M dan N. Pada masing elemen $K_R(i)$ dan $K_C(i)$ diisi dengan nilai A = {0, 1, 2, ..., 2α − 1}, dimana $K_R(i)$ dan $K_C(i)$ tidak memiliki nilai yang tetap.
2. Inisialisasi jumlah dari iterasi, ITERmax dan memberikan nilai ITER = 0.
3. Menambahkan nilai ITER = ITER +1.
4. Melakukan proses pada tiap-tiap baris *i* pada matrik Io, proses tersebut antara lain:
   a. Menghitung jumlah semua elemen pada baris *i*, dimana jumlah tersebut diwakili dengan notasi $\alpha(i)$, dengan persamaan:
   $$a(i) = \sum_{j=1}^{N} lo(i,j), i = 1,2,\dots,M \quad (1)$$
   b. Menghitung nilai $M_\alpha(i)$ dimana $M_\alpha(i)$ adalah modulo 2 dari $\alpha(i)$,
   $$Ma(i) = a(i) mod\ 2 \quad (2)$$
   c. Pada baris i dirotasi dengan posisi $K_R(i)$ dimana piksel pada citra diganti dengan posisi $K_R(i)$ pada arah kanan atau kiri, dan piksel pertama dipindah pada akhir piksel, dapat diperlihatkan dengan persamaan:
   $$if\ Ma(i) = 0 => Rotasi\ Kekanan \quad (3)$$
   $$else => Rotasi\ Ke\ Kiri$$
5. Melakukan proses pada tiap-tiap kolom *j* pada matrik Io, proses tersebut antara lain:
   a. Menghitung semua elemen pada kolom *j*, yang dinotasikan dengan $\beta(j)$, dengan menggunakan persamaan:
   $$\beta(j) = \sum_{i=1}^{M} lo(i,j), j = 1,2,\dots,N \quad (4)$$
   b. Menghitung $M_\beta(j)$ dimana $M_\beta(j)$ adalah modulo 2 dari $\beta(j)$.
   $$M\beta(i) = \beta(i)\ mod\ 2 \quad (5)$$
   c. Menggeser naik atau turun pada kolom j dengan posisi Kc(i), dapat diperlihatkan dengan persamaan:
   $$if\ M\beta(i) = 0 => Rotasi\ Keatas \quad (6)$$
   $$else => Rotasi\ Kebawah$$
6. Pada langkah 4, 5 dihasilkan image acak (I_{SCR}). Kemudian pada masingmasing baris I_{SCR}, dilakukan operasi biner XOR dengan vector K_C, dengan menggunakan persamaan:
   $$I1(2i-i,j) = Iscr(2i-i,j) \oplus Kc(j) \quad (7)$$
   $$I1(2i,j) = Iscr(2i,j) \oplus rot180(Kc(j))$$
   Dimana $\oplus$ dan rot180(K_C) adalah proses operasi bit XOR dan pergeseran vector K_C dari atas kebawah atau sebaliknya.
7. Pada masing-masing kolong dari I1 dilakukan operasi biner XOR dengan KR dengan menggunakan persamaan:
   $$Ienc(i,2j-1) = I1(i,2j-1) \oplus Kr(j) \quad (8)$$
   $$Ienc(i,2j) = I1(i,2j) \oplus rot180(Kr(j))$$
   Dimana rot180(*KR*) adalah proses pergeseran dari kiri ke kanan pada vector KR.
8. Jika ITER = ITERmax maka proses selesai dan gambar terenkripsi disimpan pada I_{enc}, dan jika tidak maka menuju langkah ke 3.

2.3 Dekripsi Algoritma Kubus Rubik

Proses Dekripsi adalah proses mengembalikan citra yang telah terenkripsi ke dalam bentuk citra asli, dimana vector KR, KC ITERmax dan merupakan kunci dari algoiritma ini. Gambar yang telah terenkirpsi dinotasikan dengan Ienc dan Io merupakan citra hasil Dekripsi dari parameter KR, Kc dan ITERmax. Langkah-langkat dari proses Dekripsi yakni:

1. Menambahkan dengan 1 nilai ITER, ITER = ITER +1.
2. Melakukan proses operasi bit XOR pada vector K_R dan tiap-tiap kolom dari citra yang terenkripsi I_{enc} dengan persamaan:
   $$I1(i,2j-1) = Ienc(i,2j-1) \oplus Kr(j) \quad (9)$$
   $$I1(i,2j) = Ienc(i,2j) \oplus rot180(Kr(j))$$
3. Kemudian dengan menggunakan vector K_C, dilakukan operasi bit XOR dengan tiap-tiap baris pada citra I_1 dengan persamaan
   $$Iscr(2i-i,j) = I1(2i-i,j) \oplus Kc(j) \quad (10)$$
   $$Iscr(2i,j) = I1(2i,j) \oplus rot180(Kc(j))$$
4. Pada tiap-tiap kolom I_{enc} yang merupakan citra teracak, dilakukan proses:
   a. Menghitung jumlah semua elemen pada colom *j*, yang disimbolkan dengan βSCR(*j*) dengan persamaan:
   $$\beta scr(j) = \sum_{i=1}^{M} lscr(i,j), j = 1,2,..,N \quad (11)$$
   b. $M_{\beta SCR}((i)$ adalah modulo 2 dari βSCR((*i*),
   $$\beta scr(i) = \beta scr(i)\ mod\ 2 \quad (12)$$
   c. Menggeser naik atau turun pada kolom j dengan posisi Kc(i), dapat diperlihatkan dengan:
   $$if\ M\beta(i) = 0 => Rotasi\ Kebawah \quad (13)$$
   $$else => Rotasi\ Keatas$$
5. Pada tiap kolom pada citra Iscr, dilakukan proses:

a. Menghitung jumlah semua elemen pada baris i yang dinotasi dengan $\alpha_{SCR}(i)$, dengan menggunkan persamaan:

$$ascr(i) = \sum_{j=1}^{N} lscr(i,j), i = 1,2,\dots,M \quad (14)$$

b. $M\alpha_{SCR}((i)$ adalah modulo 2 dari $\alpha_{SCR}(j)$,

$$Mascr(i) = ascr(i) mod\ 2 \quad (15)$$

6. Pada baris i digeser dengan posisi K_R(i), dapat diperlihatkan dengan:

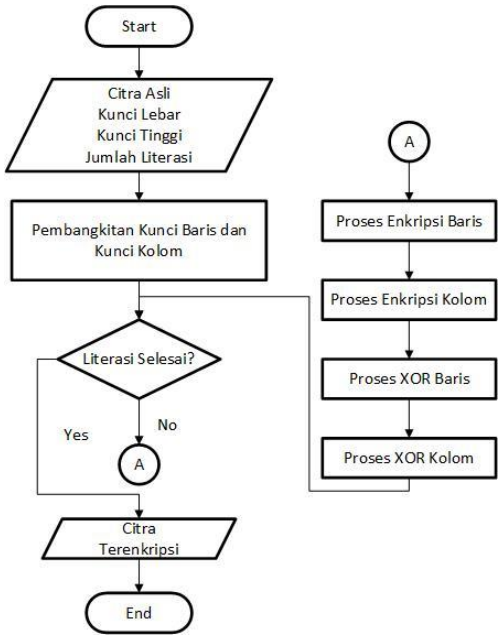
$$if\ Maacr(i) = 0 => Rotasi\ Kekiri \quad (16)$$
$$else => Rotasi\ Ke\ Kanan$$

Jika ITER = ITERmax maka proses selesai dan gambar hasil disimpan pada Ienc, dan jika tidak maka menuju langkah ke 2.

3. Analisis dan Perancangan

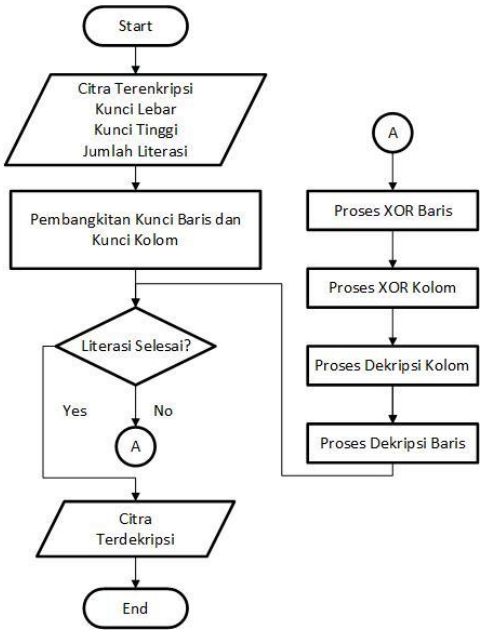
3.1 Sistem Aplikasi Umum

Pada perancangan yang dilakukan, didapatkan 2 proses utama yaitu proses enkripsi citra dan proses dekripsi citra. Proses Enkripsi adalah menyamarkan citra asli ke dalam bentuk terenkripsi, dan berikut flowchart proses enkripsi:



Gambar 1. Flowchart Proses Enkripsi

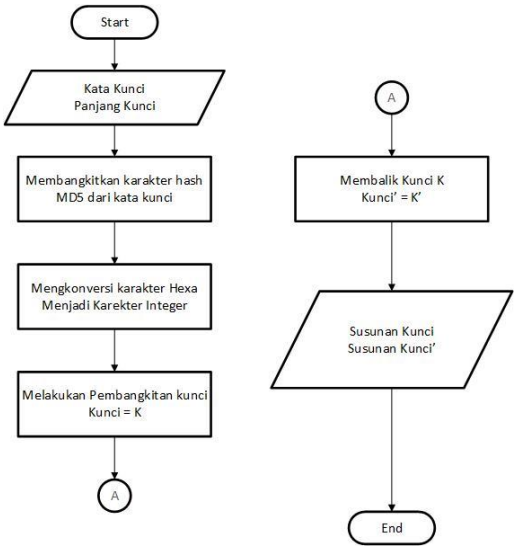
Proses Dekripsi adalah mengembalikan citra terenkripsi ke dalam bentuk terdekripsi/ asli, dan berikut flowchart proses dekripsi:



Gambar 2. Flowchart Proses Dekripsi

3.2. Sistem Pembangkit Kunci

Sistem pembangkit kunci digunakan untuk mempermudah pengguna dalam memasukkan kunci baris dan kunci kolom. Pada penelitian sebelumnya, pengguna memasukkan kunci satu per satu di tiap baris dan kolom dari awal sampai akhir. Sistem pembangkit kunci merupakan salah satu nilai tambah yang diberikan oleh pembuat pada sistem enkripsi citra menggunakan algoritma kubus rubik.



Gambar 3. Flowchart proses pembangkit kunci

Pada flowchat yang ada, terlihat pengguna hanya memasukkan sebuah kata kunci. Kata kunci tersebut bisa digunakan untuk kunci baris, maupun

knci kolom. Setelah kunci diberikan, maka sistem akan melakukan *generate* kunci sebanyak panjang kunci yang dibutuhkan. Kunci yang muncul juga sesuai dengan ketentuan, yaitu antara 0 – 255 dalam decimal / 0 – 11111111 dalam biner. Hal ini sesuai karena citra input memiliki 8 bit citra pada R, G, dan B. Selain itu, sistem juga membangkitkan kunci yang telah terotasi 180 derajat. Rotasi tak hanya pada susunan kunci, namun juga terjadi pada susunan bit penyusun kunci. Proses ini menghasilkan keluaran berupa array sepanjang lebar dan tinggi citra yang akan melakukan proses enkripsi dan dekripsi.

4. Pengujian dan Pembahasan

Pengujian dilakukan untuk menjamin dan memastikan bahwa sistem yang dirancang dapat berjalan seperti yang diharapkan. Pengujian dilakukan menggunakan 2 metode, yaitu menggunakan indra pengelihatan manusia berdasarkan hasil dan melakukan perhitungan pada citra asli dan citra hasil. Perhitungan untuk perbedaan hasil menggunakan rumus *Number of Pixel Change Rate (NPCR), y*ang mengindikasikan perbedaan pixel diantara dua citra. Dan rumus selanjutnya adalah *Unified Average Changing Intensity(UACI*), yang digunakan untuk rata-rata intensitas perbedaan pixel dari dua citra. Pengujian visual yang bisa dilakukan selanjutnya adalah menggunakan perhitungan MSE (*Mean Square Error*), yaitunilai error kuadrat rata-rata antara citra asli dengan citra pembanding dan perhitungan PSNR (*Peak Signal to Noise Ratio*), yaitu perbandingan antara nilai maksimum dari sinyal yang diukur dengan besarnya derau yang berpengaruh pada sinyal tersebut. Dan berikut adalah rumus matematika dari NPCR, UACI, MSE, dan PSNR:

$$NPCR = \frac{\sum_{i=1}^{M} \sum_{j=1}^{N} D(i,j)}{M \ x \ N} x \ 100\% \qquad (17)$$

$$with: D(i,j) = 0 \ if \ Io(i,j) = Ienc(i,j), 1$$

$$UACI = \left[\sum_{i=1}^{M} \sum_{j=1}^{N} \frac{|Io(i,j) - Ienc(i,j)|}{255} \right] \qquad (18)$$

$$x \frac{100\%}{M \ x \ N}$$

$$MSE = \frac{1}{m\,n} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I\,(i,j) - K(i,j)]^2 \qquad (19)$$

$$PSNR = 20.\log 10\,(MAX_I) - 10\,.\log 10(MSE) \qquad (20)$$

Hasil yang diinginkan pada pengujian NPCR, MSE adalah serendah mungkin. Pada pengurjian UACI adalah mendekati 0 %. Sedangkan nilai PSNR diharapkan memiliki nilai setinggi mungkin.

Dengan pengujian secara visual dan secara perhitungan matematika maka dapat dilakukan analisa hasil dengan cara sebagai berikut:

4.1 Pengujian Pembangkit Kunci

Pembangkit kunci digunakan untuk memberikan kunci di masing-masing baris atau kolom sesuai dengan posisi jan panjang. Pengujian menggunakan kata kunci, dan jumlah karakter yang diinginkan.

Tabel 1. Pengujian Kunci

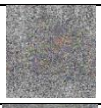
Kunci	Panjang	MD5	Angka
1	10	c4ca4238a0	12 4 12 10 4 2 3 8 10 0
2	11	c81e728d9d4	12 8 1 14 7 2 8 13 9 13 4
3	12	eccbc87e4b5c	14 12 12 11 12 8 7 14 4 11 5 12
4	13	a87ff679a2f3e	10 8 7 15 15 6 7 9 10 2 15 3 14

Pada pengujian, didapatkan nilai yang sangat berbeda meskipun kunci memiliki kedekatan susunan bit maupun pada deretan angka. Angka yang muncul pada tabel selanjutnya digunakan sebagai kunci pada baris dan kolom pada saat melakukan proses enkripsi maupun dekripsi.

4.2 Pengujian Visual dan Sensitifitas Kunci

Pengujian Visualdigunakan untuk melihat kecocokan citra antara citra asli, dan citra terdekripsi.Pada proses enkripsi, aplikasi diberi kunci K1 adalah 255 dalam decimal atau 11111111 dalam bilangan bit. Proses dekripsi menggunakan kunci K2 adalah 254 dalam decimal atau 11111110 dalam bilangan bit. Perbedaan terdapat pada bit akhir sebanyak 1 bit. Pengujian dilakukan berdasarkan parameter tersebut.

Tabel 2. Pengujian Visual dan Sensitifitas Kunci

No	Asli	Enkripsi K1	Dekripsi K1	Dekripsi K2
1				
2				

3				
4				

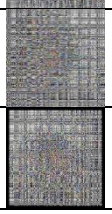
Analisa Visual berdasarkan pencocokan dilakukan citra asli dan citra hasil. Hal ini menunjukkan bahwa dengan kunci yang benar atau sama, maka proses enkripsi dan dekripsi dinyatakan berhasil dimana Citra asli dan citra dekripsi (K1) memiliki hasil yang sama. Sedangkan untuk citra asli dan citra enkripsi (K1) memiliki hasil yang sangat berbeda.

Pada pengujian sensitifitas kunci, dapat terlihat perbedaan yang sangat besar pada Tabel 1. Jika proses dekripsi menggunakan K1 maka citra akan kembali seperti citra asli. Sedangkan jika citra didekripsi menggunakan K2, citra tetap tidak kembali seperti semula.

4.3 Pengujian Serangan Terhadap Citra

Serangan pada citra terenkripsi bisa dilakukan untuk mengubah hasil citra terdekripsi. Terdapat berbagai serangan yang bisa dilakukan pada citra terenkripsi seperti pemberian *noise*, melakukan *rotation*, proses *filtering*, dan melakukan *cropping* pada citra. Pengujian dilakukan dengan melakukan serangan pada citra terenkripsi, kemudian dilakukan dekripsi dengan kunci yang sesuai. Dilakukan pengujian visual untuk mengetahui perbedaan citra yang terkena serangan dengan citra asli.

Tabel 3. Serangan Citra Terenkripsi

No	Serangan	Hasil	Dekripsi
1	Hapus 5 Pixel		
2	Filter Median 3 x 3 pixel		
3	Rotasi Kanan		
4	Blok 5px (ukuran sama)		

Pada pengujian serangan yang dilakukan, didapatkan hasil dekripsi yang tidak sempurna. Serangan apapun akan memberikan dampak kepada hasil dari dekripsi. Sehingga dari hasil yang didapat, dapat diketahui bahwa algoritma tidak tahan terhadap serangan.

4.4 Pengujian Kecepatan Proses

Parameter yang digunakan utnuk pengujian adalah ukuran citra serta banyaknya jumlah literasi. Pengujian ini dimaksudkan untuk melihat perbedaan kecepatan pada ukuran citra yang berbeda, serta kecepatan proses banyaknya jumlah literasi yang dilakukan.

Tabel 4. Kecepatan berdasarkan ukuran

No	Ukuran	Waktu Proses
1	128x128	0.359375 s
2	256x256	0.65625 s
3	512x512	1.265625 s
4	1024x1024	2.625 s
5	2048x2048	5.765625 s

Proses uji coba kecepatan berdasarkan jumlah literasi yang digunakan. Citra uji coba berukuran 512 x 512 piksel.

Tabel 5. Kecepatan berdasarkan ukuran

Iterasi	Waktu Proses
1	1.265625 s
2	2.1875 s
3	3.1875 s
4	4.25 s
5	5.3281 s

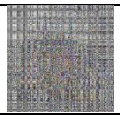
Dari tabel 4 dan tabel 5 dapat diketahui bahwa kecepatan proses berbanding lurus dengan ukuran citra dan jumlah literasi.

4.5 Pengujian Penyimpanan Citra

Pengujian dilakukan menggunakan citra asli dan dibandingkan dengan citra terdekripsi yang telah disimpan ke tipe file tertentu. Kemudian proses pengujian dilakukan dengan perhitungan untuk mengetahui perbedaan citra asli dan citra hasil dekripsi.

Tabel 6. Tipe data penyimpanan

No	Tipe Data	Hasil	Dekripsi
1	Bitmap 1025kb		
2	Jpeg 145kb		
3	Png 419kb		
4	Gif 211kb		

5	Tiff 966kb		

Pada tipe Jpeg dan Gif, pengujian kasat mata terlihat perubahan kualitas dari citra terdekripsi. Untuk pencocokan lebih lanjut, dilakukan perhitungan untuk kualitas citra asli dibantingkan citra terdekripsi.

Tabel 7. Pengujian tipe penyimpanan

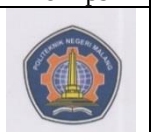
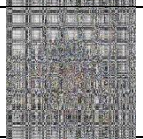
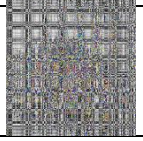
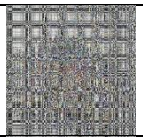
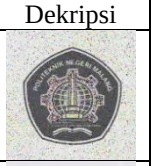
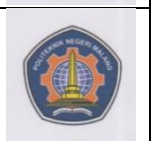
No	NPCR	UACI	MSE	PSNR
1	0	0	0	∞
2	97.965494	21.1643	6427.01	11.2605
3	0	0	0	∞
4	72.578684	15.2140	4332.83	11.9871
5	0	0	0	∞

Pengujian berdasarkan perhitungan dapat memperlihatkan bahwa tipe penyimpanan Jpeg dan Gif tidak cocok digunakan. Dan diantara tipe penyimpanan bitmap, png, dan tiff diketahui bahwa png memiliki ukuran yang paling kecil.

4.6 Pengujian Pengiriman Citra

Pengujian dilakukan untuk mencari media pengiriman yang terbaik untuk pengiriman citra terenkripsi, baik menggunakan media seperti email, aplikasi pengiriman pesan, dan penyedia jasa layanan cloud.

Tabel 8. Pengujian pengiriman citra

No	Skenario	Hasil	Dekripsi
1	Kirim Email (280kb .png)		
2	Kirim BBM (48kb .jpg)		
3	Kirim BBM HD (280kb .jpg)		
No	Skenario	Hasil	Dekripsi
4	Kirim WA (62kb .jpg)		
5	Kirim Bluetooth (419kb .png)		
6	Simpan Cloud (419kb .png)		

Analisa lebih lanjut menggunakan perhitungan berdasarkan perbandingan citra asli dengan citra terdekripsi.

Tabel 9. Analisa pengiriman citra

No	NPCR	UACI	MSE	PSNR
1	0	0	0	∞
2	97.6044	10.7425	2235.975	14.71227
3	0	0	0	∞
4	96.72871	8.84421	1677.571	16.11291
5	0	0	0	∞
6	0	0	0	∞

Pada tabel 9, dapat dilihat bahwa media yang berhasil melakukan dekripsi merupakan media yang tidak melakukan pengubahan pada kualitas, ukuran, serta kompresi.

5. Kesimpulan dan Saran

5.1 Kesimpulan

Berdasarkan penelitian yang dilakukan, kesimpulan yang didapat adalah sebagai berikut:

1. Proses Enkripsi dan proses dekripsi dinyatakan berhasil dengan Pengujian kecocokan citra asli dan citra terdekripsi sebesar 100%.
2. Algoritma yang digunakan aman terhadap serangan kunci dengan parameter perbedaan 1 bit kunci baris dan kolom yang digunakan.
3. Hasil dekripsi tidak tahan terhadap serangan. Hal ini dibuktikan dengan rusaknya citra hasil dekripsi.
4. Semakin besar citra, maka menimbulkan waktu yang lebih lama dalam prosesnya. Semakin banyak literasi juga membuat proses semakin lama, namun banyaknya literasi berpengaruh pada pengacakan yang lebih baik
5. Penyimpanan tipe PNG memiliki keuntungan yang terbanyak. File PNG memiliki ukuran paling kecil. Dan dapat memberikan hasil dekripsi dengan kecocokan 100% pada analisa yang dilakukan.
6. Media pengiriman yang terbaik adalah media yang tidak melakukan konversi, pengubahan ukuran, dan kompresi terhadap citra digital. Hal ini didukung dengan analisa yang dilakukan.

5.2 Saran

Dalam penelitian ini masih banyak hal yang dapat dikembangkan,seperti :

1. Pengembangan untuk aplikasi yang lebih kompleks yang digunakan untuk instansi yang membutuhkan keamanan pada citra digital.
2. Pengembangan algoritma yang tahan terhadap serangan citra. Sehingga citr

Sehingga citra terdekripsi bisa sama dengan citra setelah termanipulasi.

3. Menambahkan algoritma keamanan yang lain selain proses pengacakan dan proses XOR.
4. Penggunaan kunci public untuk proses enkripsi dan dekripsi citra digital.
5. Pengembangan aplikasi untuk media mobile dan media web.

Demikian saran yang dapat penulis berikan, semoga saran tersebut bisa dijadikan sebagai bahan masukan yang dapat bermanfaat bagi penulis khususnya dan umumnya bagi akademisi di kemudian hari.

Daftar Pustaka:

Abdul Kadir, dan Adhi Susanto. 2013. "*Teori dan Aplikasi Pengolahan Citra*". Yogyakarta: Andi.

Christof Paar, Jan Pelzl. 2010. "*Understanding Cryptography*". Berlin: Springer.

Enkripsi MD5 dan Pengujian Integritas Data. 2011 [Online] Tersedia: http://punyaku.web.id/enkripsi-md5-dan-pengujian-integritas-data.html [1 Mei 2016].

Khaled Loukhaoukha, Jean-Yves Chouinard, dan Abdellah Berdai. 2011. "*A Secure Image Encryption Algorithm Based on Rubik's Cube Principle*". Department of Electrical and Computer Engineering, Laval University, QC, Canada G1K 7P4.

Makalah Image Geometri. 2015 [Online] Tersedia:http://dokumen.tips/documents/makalah-image-grometri.html [3 Mei 2016].

Pengertian citra digital. 2013 [Online] Teredia: http://www.temukanpengertian.com/2013/08/pengertian-citra-digital.html [24 Nopember 2015].

PSNR and UACI Randomness Tests for Image Encryption. 2016 [Online] Tersedia: https://www.researchgate.net/profile/Yue_Wu14/publication/259190481_NPCR_and_UACI_Randomness_Tests_for_Image_Encryption/links/0c96052a80913343e2000000.pdf/download?version=vrp [1 Mei 2016].

Wijayadi Saputra, Ahmad Affif S, Dian Eka R. 2013. "*Enkripsi Citra Digital Dengan Mengunakan Algoritma Rubic's Cube*". Program Studi Ilmu Komputer, Fakultas Program Teknologi Informatika dan Ilmu Komputer Universitas Brawijaya Malang.

Yue Wuu, Joseph P. Noonan, dan Sos Agaian. 2011. "*NPCR and UACI Randomness Test for Image Encryption*". Multidisclipinary Journals in Science and Technology, JSAT.