

## PENDETEKSIAN *COPY-MOVE FORGERY* PADA CITRA DIGITAL

Era Chalis Kurniangesti<sup>1</sup>, Rosa Andrie Asmara<sup>2</sup>, Dwi Puspitasari<sup>3</sup>

<sup>1,2,3</sup> Teknik Informatika, Teknologi Informasi, Politeknik Negeri Malang

<sup>1</sup>erachalisk@gmail.com, <sup>2</sup>rosa.andrie@polinema.ac.id, <sup>3</sup>dwi.puspitasari@polinema.ac.id

---

### Abstrak

Pemalsuan citra memberikan dampak yang tidak bisa dianggap sepele bagi masyarakat dikarenakan citra merupakan salah satu sumber informasi yang sering digunakan. Kasus yang paling sering kita ketahui adalah munculnya berita-berita hoax yang tak jarang didukung dengan bukti berupa gambar atau citra yang telah dimanipulasi. Beberapa jenis pemalsuan citra, diantaranya adalah *Retouching*, *Copy-Move* dan *Splicing*. Penelitian kali ini membahas mengenai pendeteksian pemalsuan *Copy-Move*. *Copy-Move* adalah pemalsuan citra yang dilakukan dengan maksud untuk menghilangkan suatu objek dengan cara menutupinya dengan blok kecil yang disalin dari bagian lain dari gambar yang sama. Teknik ini juga digunakan untuk memperbanyak suatu objek dengan menempelkan objek pada posisi lain.

Tujuan utama dari penelitian ini adalah mendeteksi adanya pemalsuan yang dilakukan pada citra digital khususnya *Copy-Move Forgery*. Metode yang digunakan pada penelitian ini adalah metode DCT dan DCT Terkuantisasi yang mengubah citra kedalam domain frekuensi sebelum dilakukan pendeteksian. Proses pendeteksian dilakukan dengan langkah-langkah yaitu citra yang dicurigai mengandung pemalsuan akan diinputkan kedalam sistem dan dilakukan pre prosesing menjadi citra *grayscale*. Kemudian citra dibagi kedalam blok-blok pixel berukuran 8x8 dan mengubah blok-blok pixel tersebut kedalam domain frekuensi dengan DCT atau DCT Terkuantisasi. Setelah itu hasilnya akan dipisah sesuai dengan frekuensinya, yaitu frekuensi rendah, menengah dan tinggi untuk selanjutnya dilakukan pencocokkan. Setelah dilakukan pencocokkan, selanjutnya pixel-pixel yang dianggap sebagai *Copy-Move* akan ditandai.

Pengujian menggunakan 3 kategori citra dengan jumlah masing-masing kategori 3 citra. Selain itu pada masing-masing kategori citra terdapat citra *Copy-Move* asli dan citra *Copy-Move* yang telah dilakukan *postprocessing*. Dari hasil pengujian diperoleh pendeteksian dengan akurasi terbaik dilakukan dengan metode DCT Terkuantisasi Matrik1 dengan presentase *Copy-Move* yang Pendeteksian dengan akurasi terbaik dilakukan dengan metode DCT Terkuantisasi Matrik 1 dengan presentase *Copy-Move* yang **Terdeteksi** pada frekuensi rendah adalah 22,22%, pada frekuensi menengah sebesar 25% dan pada frekuensi tinggi 11,11%. Juga dihasilkan nilai **Terdeteksi dengan False Match** sebesar 55,56% pada frekuensi rendah, 41,67% pada frekuensi menengah, dan 72,22% pada frekuensi tinggi.

**Kata kunci** : *Copy-Move Forgery*, DCT, DCT Terkuantisasi, Citra Digital, Pemalsuan Citra

---

### 1. Pendahuluan

Citra digital merupakan salah satu konten informasi dalam teknologi informasi multimedia yang terus berkembang saat ini sehingga banyak diaplikasikan dalam berbagai bidang, Madenda (2015). Beberapa bidang yang memanfaatkan citra digital antara lain bidang jurnalistik, kedokteran, penyelidikan forensik, dan masih banyak lagi. Akan tetapi hal paling penting yang harus dipastikan adalah keaslian atau keotentikan dari suatu citra digital. Dengan semakin maju dan berkembangnya teknologi, saat ini banyak bermunculan aplikasi perangkat lunak pengolahan citra yang mudah untuk digunakan, bahkan untuk orang awam. Salah satu dampak negatif dari perkembangan perangkat lunak ini adalah maraknya pemalsuan citra digital. Umumnya, tujuan dari pemalsuan citra terbagi menjadi 3 jenis, yaitu: menambahkan suatu objek baru dalam citra, menghapus atau menyembunyikan

suatu bagian dalam citra dan mengubah makna dari suatu citra, Sunil (2011).

Beberapa jenis pemalsuan citra, diantaranya adalah *Retouching*, *Copy-Move* dan *Splicing*. *Copy-Move* merupakan jenis pemalsuan citra yang paling umum dan mudah untuk dilakukan karena menggunakan bagian dari citra itu sendiri untuk melakukan pemalsuan. Karena menggunakan bagian dari citra itu sendiri sebagai sumber pemalsuannya, maka komponen-komponen seperti *noise*, kecerahan, warna dan properti lainnya akan terlihat hampir mirip dan sulit untuk dibedakan, G. K. S. Gaharwar (2015).

Terdapat dua teknik dalam mengidentifikasi keaslian suatu citra yaitu Aktif dan Pasif teknik. Aktif teknik sendiri merupakan teknik pendeteksian keaslian citra dengan cara menambahkan suatu informasi pada citra melalui *watermarking* citra. Sehingga apabila citra tersebut dimanipulasi, informasi yang telah disisipkan akan mengalami

perubahan. Perubahan inilah yang menjadi bukti bahwa citra sudah tidak otentik. Sementara pada Pasif teknik tidak terdapat informasi apapun yang disisipkan pada citra. Sehingga lebih sulit untuk mengetahui apakah citra asli atau telah dipalsukan, G. K. S. Gaharwar (2015). Padahal dalam praktiknya citra yang telah diberikan *watermarking* lebih sedikit dibandingkan dengan citra tanpa *watermarking*. Karena itulah penelitian dalam bidang Pasif teknik untuk mendeteksi keaslian suatu citra terus dilakukan dengan berbagai metode untuk menemukan suatu metode yang efektif.

Beberapa penelitian sebelumnya terkait dengan pendeteksian *Copy-Move* antara lain Sina dan Harjoko yang mengusulkan pendeteksian *Copy-Move* menggunakan metode pencocokan *Exact Match*, Derwin (2016). Metode ini mampu mendeteksi bagian citra yang dipalsukan akan tetapi tingkat kesalahan deteksi juga cukup besar dan membutuhkan waktu komputasi yang lama. Popescu dan Farid menggunakan metode PCA untuk mendeteksi *Copy-Move Forgery*, Farid (2004). Metode ini didasarkan pada fakta bahwa representasi PCA lebih kebal terhadap random noise dan faktor kompresi JPEG. Keakuratan metode ini baik kecuali untuk ukuran blok kecil dan kualitas JPEG yang rendah. Zhang, Feng dan Su menggunakan metode DWT dalam penelitiannya. Metode ini mampu mendeteksi pemalsuan *Copy-Move* dengan efisien, Zhang (2008). Akan tetapi metode ini mengalami masalah ketika bagian yang dipalsukan berada di tengah-tengah dari citra. Zuldán, Arnia dan dan Muharar melakukan penelitian dengan menggunakan koefisien AC dan menghilangkan koefisien DC dari hasil DCT, ZulfaN (2016). Penelitian ini mampu mendeteksi adanya pemalsuan *Copy-Move*. Dalam penelitian ini juga mampu mendeteksi *Copy-Move* yang telah mengalami modifikasi rotasi dan histogram ekualisasi. Akan tetapi hasil deteksi masih kurang akurat dalam menentukan bagian yang mengalami pemalsuan.

Dalam penelitian kali ini Penulis akan menggunakan koefisien kuantisasi DCT sebagai koefisien yang akan dicocokkan. Diharapkan dengan metode ini hasil yang didapatkan bisa lebih akurat dalam pendeteksian pemalsuan citra *Copy-Move*.

**2. Landasan Teori**

**2.1 Copy-Move Forgery**

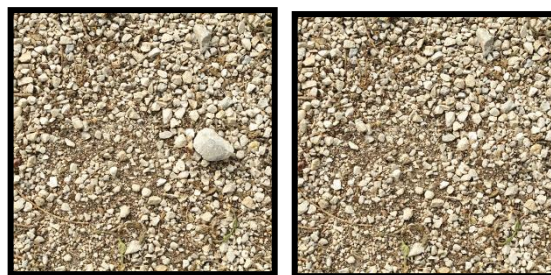
*Copy-Move* adalah pemalsuan citra yang dilakukan dengan maksud untuk menghilangkan suatu objek dengan cara menutupinya dengan blok kecil yang disalin dari bagian lain dari gambar yang sama. Teknik ini juga digunakan untuk memperbanyak suatu objek dengan menempelkan objek pada posisi lain. Untuk menutupi objek, umumnya citra yang disalin adalah tekstur seperti rumput, dedaunan, kerikil, ataupun objek lain yang memiliki pola yang tidak beraturan tujuannya agar

objek yang menutupi serupa dengan latar atau objek di sekitarnya sehingga sulit dibedakan dengan mata.

Contoh pemalsuan citra dengan tipe *copy-move* dapat diamati pada Gambar 2 dan 3, Arya (2017). Pada gambar 2 tujuan dari pemalsuan *copy-move* adalah untuk menggandakan suatu objek dalam gambar. Gambar 2 yang berada disebelah kiri merupakan gambar asli sementara yang berada disebelah kanan merupakan gambar hasil *copy-move*. Objek pohon pada gambar asli di *copy* dan ditempelkan pada bagian lain dari gambar tersebut sehingga mengakibatkan jumlah pohon bertambah.



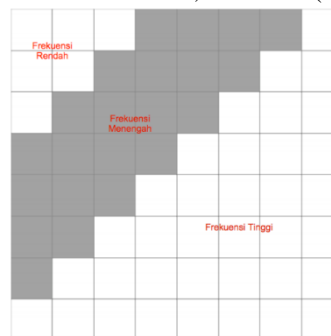
Gambar 2. Contoh Copy-Move untuk Menggandakan Objek



Gambar 3. Contoh Copy-Move untuk Menghilangkan Objek

**2.2 Transformasi DCT (Discrete Cosine Transform)**

Proses transformasi DCT adalah proses transformasi data dari domain spasial ke domain frekuensi, yang mampu memisahkan informasi pada frekuensi rendah hingga frekuensi tinggi. Dalam sebuah citra informasi utama yang dikandungnya lebih banyak terdapat pada frekuensi rendah dan menengah, sedang informasi frekuensi tinggi merupakan informasi detail., Madenda (2015).



Gambar 4. Gambaran Pembagian Frekuensi DCT pada Blok 8x8

Proses DCT-2D ini dilakukan secara terpisah dimulai dari proses DCT arah  $x$  dan kemudian hasilnya dilakukan proses DCT arah  $y$ .

$$F(u, y) = \sqrt{\frac{2}{N}} C_u \sum_{x=0}^{N-1} \cos \left[ \frac{(2x+1)u\pi}{2N} \right] \cdot f(x, y) \tag{1}$$

$$F(u, v) = \sqrt{\frac{2}{N}} C_v \sum_{y=0}^{N-1} F(u, y) \cdot \cos \left[ \frac{(2y+1)v\pi}{2N} \right] \tag{2}$$

$$\text{Dimana } C_u, C_v = \begin{cases} \sqrt{\frac{1}{2}} & \text{bila } u, v = 0 \\ 1 & \text{bila } u, v > 0 \end{cases} \tag{3}$$

Persamaan 2 merupakan hasil dari proses DCT-2D yang merepresentasikan domain frekuensi citra, dimana  $x, y$  adalah koordinat diskret dalam domain spasial citra dan  $u, v$  adalah koordinat diskret dalam domain frekuensi.  $N$  menunjukkan ukuran citra, dimana dalam proses ini merupakan persegi empat.

Selanjutnya persamaan 2 ini dapat direpresentasikan dalam bentuk perkalian matriks seperti yang ditunjukkan dalam persamaan 4:

$$[F]_{u,v} = ([DC]_{u,x} \cdot [f]_{x,y}) \cdot [DC]_{y,v} \tag{4}$$

Yang berlaku umum untuk matriks persegi empat dan merupakan perpangkatan dua.  $[DC]_{u,x}$  dan  $[DC]_{y,v}$  secara berurutan adalah matriks cosinus arah sumbu  $x$  dan arah sumbu  $y$ , yang dapat dihitung dengan persamaan 5 dan 6:

$$DC(u, x) = \sqrt{\frac{2}{N}} C_u \cos \left[ \frac{(2x+1)u\pi}{2N} \right] \tag{5}$$

$$DC(y, v) = \sqrt{\frac{2}{N}} C_v \cos \left[ \frac{(2y+1)v\pi}{2N} \right] \tag{6}$$

### 2.3 Kuantisasi DCT

Tujuan dari kuantisasi dalam pengolahan citra adalah menghilangkan sejumlah informasi yang secara visual tidak begitu signifikan. Dalam koefisien DCT, informasi yang kurang signifikan berada pada frekuensi tinggi dan terletak pada bagian kanan bawah dari matriks koefisien DCT tersebut, Madenda (2015).

Kuantisasi hasil DCT dinyatakan sebagai proses pembagian nilai setiap elemen pada koordinat  $(u, v)$  dari matriks  $[F]$  oleh nilai elemen kuantisasi pada koordinat yang sama dari matriks kuantisasi  $[Q]$ . Selanjutnya hasil pembagian ini dibulatkan ke nilai integer terdekat. Secara matematis proses ini dapat dilakukan menggunakan persamaan 7:

$$F^Q(u, v) = \text{round} \left( \frac{F(u,v)}{Q(u,v)} \right) \tag{7}$$

dimana matriks  $F^Q$  disebut matriks koefisien DCT terkuantisasi.

Terdapat beberapa matrik kuantisasi  $Q$  yang nilainya bisa didapatkan sendiri menggunakan suatu rumus (Overflow, 2015). Tetapi juga terdapat matrik-matrik kuantisasi yang sudah diketahui nilainya seperti matrik kuantisasi milik Photoshop. Dalam penelitian ini digunakan tiga jenis matrik kuantisasi dari Photoshop sebagai berikut (Madenda, 2015):

2 2 3 4 5 6 8 11	1 1 1 2 3 3 4 5	1 1 1 1 1 1 1 2
2 2 2 4 5 7 9 11	1 1 1 2 3 4 4 6	1 1 1 1 1 1 1 2
3 2 3 5 7 9 11 12	1 1 2 3 4 4 5 7	1 1 1 1 1 1 1 2
4 4 5 7 9 11 12 12	2 2 3 4 4 5 7 8	1 1 1 1 1 1 2 2
5 5 7 9 11 12 12 12	3 3 4 4 5 7 8 8	1 1 1 1 2 2 3 3
6 7 9 11 12 12 12 12	3 4 4 5 7 8 8 8	1 1 1 2 2 3 3 3
8 9 11 12 12 12 12 12	4 4 5 7 8 8 8 8	1 1 2 2 3 3 3 3
11 11 12 12 12 12 12 12	5 6 7 8 8 8 8 8	2 2 2 3 3 3 3 3

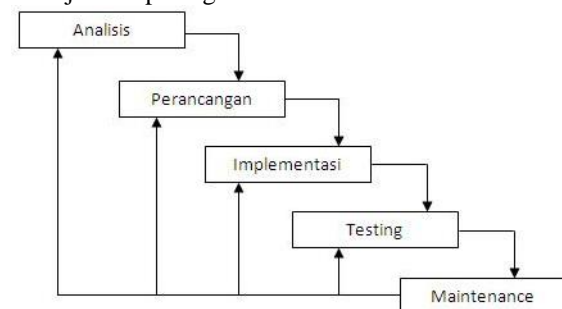
(a) (b) (c)

Gambar 5. Matrik Kuantisasi (a)Kualitas 10, (b) Kualitas 11, (c) Kualitas 12 Kompresi Citra JPEG Photoshop

### 3. Metode Penelitian

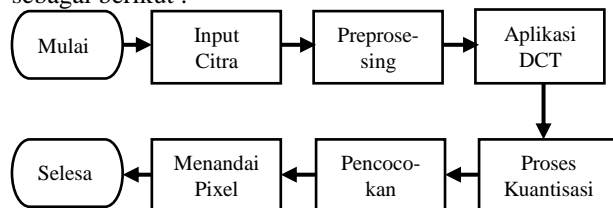
#### 3.1 Metode Pengembangan Sistem

Sistem deteksi pemalsuan citra digital dengan metode *Block Matching* berbasis DCT ini merupakan sebuah sistem yang bertujuan untuk mendeteksi adanya pemalsuan citra digital dengan tipe *Copy-Move*. Metode pengembangan sistem pada penelitian kali ini adalah dengan menggunakan metode *waterfall*. Tahapan dalam metode *waterfall* ditunjukkan pada gambar dibawah ini:



Gambar 6. Metode Pengembangan Sistem Waterfall

Pada tahap implementasi akan dilakukan *coding* tahapan-tahapan pendeteksian citra *Copy-Move*. Tahapan-tahapan tersebut antara lain adalah sebagai berikut :



Gambar 7. Alur Sistem

- Preprosesing  
Proses preprosesing dilakukan dengan mengubah citra input kedalam citra *grayscale*.
- Aplikasi DCT atau Kuantisasi DCT  
Setelah citra diubah menjadi citra *grayscale* selanjutnya citra *grayscale* tersebut dibagi kedalam blok-blok pixel dengan ukuran  $n \times n$  yang saling

overlap. Disini penulis menggunakan blok citra berukuran 8x8. Masing-masing blok berukuran 8x8 tersebut akan diubah kedalam domain frekuensi dengan menggunakan metode DCT dan dikuantisasi. Hasil akhir dari proses ini akan menghasilkan blok-blok pixel yang telah terbagi kedalam 3 frekuensi, yaitu frekuensi rendah, menengah dan tinggi. Hasil dari proses ini kemudian disimpan kedalam array 2 dimensi.

c. Pencocokan Nilai Frekuensi

Proses pencocokkan dilakukan dengan memilih terlebih dulu frekuensi mana yang akan dicocokkan. Pencocokkan bisa dilakukan pada frekuensi rendah, menengah dan tinggi. Setelah memilih frekuensi yang akan dicocokkan, akan dilakukan pencocokkan antar baris yang terdapat pada array dengan menggunakan frekuensi yang sesuai. Ketika ditemukan dua baris dengan nilai yang sama, langkah selanjutnya adalah mencari selisih koordinat posisi dari dua blok pixel yang disimpan pada baris tersebut yaitu  $(s_1, s_2) = (i_1 - i_2, j_1 - j_2)$ . Pada proses ini juga disediakan suatu counter yang nilainya akan bertambah setiap kali ditemukan dua baris yang sama pada jarak posisi yang sama yaitu  $counter(s_1, s_2) += 1$ .

d. Penandaan Pixel

Ketika proses pencocokkan selesai, langkah selanjutnya adalah menandai pixel-pixel yang dianggap sebagai pemalsuan. Pada proses ini akan diberikan suatu threshold bawah dan threshold atas yang akan dibandingkan dengan nilai  $counter(s_1, s_2) += 1$ . Apabila nilai  $counter(s_1, s_2) += 1$  berada diantara threshold bawah dan threshold atas, maka pixel-pixel yang bersangkutan akan ditandai sebagai pemalsuan Copy-Move. Sehingga output akhirnya adalah citra input yang telah diberikan tanda pada bagian yang dianggap mengandung pemalsuan Copy-Move.

3.2 Pengujian dan Pembahasan

3.2.1 Pengujian

Pengujian akurasi bertujuan untuk menguji kemampuan sistem dalam melakukan deteksi pemalsuan Copy-Move pada citra. Pada pengujian kali ini citra yang akan digunakan sebagai citra uji adalah citra RGB dengan ekstensi .png dan .bmp. citra dengan ekstensi .jpg / .jpeg tidak dapat digunakan karena terdapat kompresi pada proses penyimpanan dengan ekstensi .jpg / .jpeg sehingga tidak dapat dilakukan deteksi. Berikut merupakan rincian dari data citra yang akan diuji:



Gambar 8. (a) Citra hasil kamera dengan background sedikit (K1), (b) Citra hasil kamera dengan background banyak (K2), (c) Citra hasil buatan software (K3)

Tabel 1. Kategori dan Jumlah Citra sebagai Data Uji

Jenis Copy-Move	Kategori Citra	Jumlah
Copy-Move Asli	K1	3
	K2	3
	K3	3
Copy-Move Postprocessing Rotating	K1	3
	K2	3
	K3	3
Copy-Move Postprocessing Scalling	K1	3
	K2	3
	K3	3
Copy-Move Postprocessing Noise	K1	3
	K2	3
	K3	3

User dapat memilih akan mendeteksi citra dengan metode DCT atau Kuantisasi DCT. Seperti yang diketahui penelitian ini menggunakan metode DCT yang mengubah citra ke domain frekuensi dan membaginya kedalam tiga jenis frekuensi. Sehingga nantinya pengujian juga akan dilakukan dengan menggunakan ketiga frekuensi tersebut. Untuk metode Kuantisasi DCT atau DCT Terkuantisasi akan digunakan tiga matrik kuantisasi berukuran 8x8 seperti pada gambar 2.5:

Skenario pengujian yang akan dilakukan adalah sebagai berikut:

- a. **Copy-Move Terdeteksi** apabila bagian yang di Copy-Move berhasil ditandai dengan warna merah.
- b. **Copy-Move Terdeteksi dengan False Match** apabila bagian yang di Copy-Move berhasil ditandai dengan warna merah dan terdapat bagian lain yang bukan merupakan Copy-Move juga ikut ditandai dengan warna merah.
- c. **Copy-Move Tidak Terdeteksi** apabila bagian yang di Copy-Move tidak ditandai dengan warna merah.

Berikut merupakan hasil dari pengujian yang telah dilakukan dengan menggunakan metode DCT dan Kuantisasi DCT.

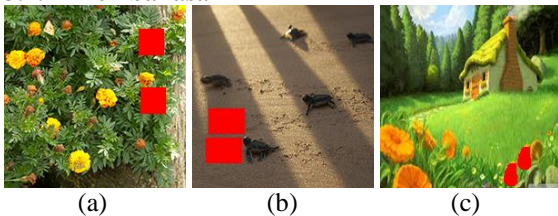
a. Hasil Pengujian Citra Secara Keseluruhan

Hasil pengujian secara keseluruhan ditunjukkan oleh tabel 2 dibawah ini.

Tabel 2. Hasil Pengujian Secara Keseluruhan

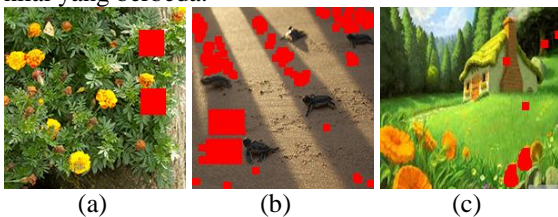
Jenis Pengujian	Presentase Hasil Pengujian								
	Terdeteksi			Terdeteksi dengan <i>False Match</i>			Tidak Terdeteksi		
	Frekuensi Rendah	Frekuensi Menengah	Frekuensi Tinggi	Frekuensi Rendah	Frekuensi Menengah	Frekuensi Tinggi	Frekuensi Rendah	Frekuensi Menengah	Frekuensi Tinggi
DCT	38.89%	38.89%	38.89%	16.67%	13.89%	16.67%	44.44%	47.22%	44.44%
DCT Kuantisasi 1	22.22%	25%	11.11%	55.56%	41.67%	72.22%	22.22%	33.33%	16.67%
DCT Kuantisasi 2	25%	19.44%	16.67%	33.33%	38.89%	55.56%	41.67%	41.67%	27.78%
DCT Kuantisasi 3	22.22%	38.89%	16.67%	36.11%	13.89%	38.89%	41.67%	47.22%	44.44%
Tanpa Metode	38.89%			13.18%			47.22%		

3.2.2 Pembahasan



Gambar 9. Hasil Pengujian dengan DCT pada Citra (a) K1, (b) K2, (c) K3

Pada pengujian masing-masing kategori citra dengan DCT, baik untuk frekuensi rendah, menengah atau tinggi, nilai terdeteksi dengan *false match* tidak terlalu besar walaupun citra sebagai data uji memiliki background dengan warna yang serupa khususnya citra pada kategori K2 dan K3 seperti ditunjukkan pada gambar 9. Hal ini dikarenakan nilai DCT merupakan nilai desimal yang detail. Sehingga perbedaan pixel sedikit akan memberikan nilai yang berbeda.

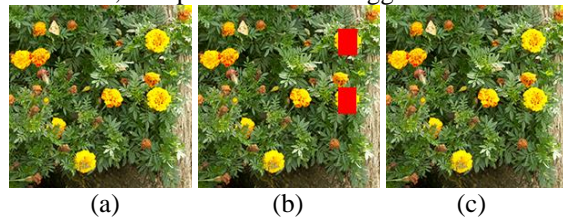


Gambar 10. Hasil Pengujian DCT Terkuantisasi Matrik 1 pada Citra (a) K1, (b) K2, (c) K3

Sementara pada DCT terkuantisasi baik dengan Matrik Kuantisasi 1, 2 ataupun 3, nilai terdeteksi dengan *false match* lebih besar dibandingkan dengan DCT, baik pada citra kategori K1, K2 dan K3. Hal ini dikarenakan proses kuantisasi membagi nilai hasil DCT dengan nilai matrik kuantisasi dan kemudian hasilnya dibulatkan sehingga pixel-pixel dengan nilai yang hampir sama akan menghasilkan nilai yang sama dan tertandai. Kuantisasi sendiri biasanya digunakan dalam kompresi untuk menghilangkan nilai pada frekuensi tinggi DCT yang dianggap kurang begitu penting. Hal tersebut menjelaskan kenapa terdapat banyak

*false match* terutama pencocokkan pada frekuensi tinggi. Pada kuantisasi yang digunakan, nilai *false match* terbanyak terdapat pada DCT Terkuantisasi Matrik 1 karena nilai Matrik 1 yang lebih besar dari dua matrik lainnya. Sementara pada kategori citra, nilai terdeteksi dengan *false match* terbanyak terdapat pada kategori K2 karena pada kategori K2 citra memiliki warna background yang hampir sama.

Pada pengujian secara keseluruhan bisa dikatakan pengujian menggunakan DCT Terkuantisasi Matrik 1 memiliki hasil lebih baik daripada yang lain. Hal ini dikarenakan pengujian dengan DCT Terkuantisasi Matrik 1 memiliki tingkat **Terdeteksi** 22,22% dan **Terdeteksi dengan *False Match*** 55,56% pada frekuensi rendah, **Terdeteksi** 25% dan **Terdeteksi dengan *False Match*** 41,67% pada frekuensi menengah, **Terdeteksi** 11,11% dan **Terdeteksi dengan *False Match*** 72,22% pada frekuensi tinggi.



Gambar 10. Hasil Pengujian dengan DCT pada Citra (a) *postprocessing rotation*, (b) *postprocessing scaling*, (c) *postprocessing noise*

Nilai **Tidak Terdeteksi** terbesar ada pada metode DCT yaitu 44,44% pada frekuensi rendah, 47,22 pada frekuensi menengah dan 44,44% pada frekuensi tinggi. Seperti yang dijelaskan diatas bahwa DCT memiliki nilai desimal yang detail, sehingga jika terdapat perubahan nilai pixel pada citra yang di *copy-move*, maka tidak akan terdeteksi. Perubahan nilai pixel ini disebabkan salah satunya oleh *postprocessing*. Karena itulah DCT sangat sulit melakukan pendeteksian pada citra *Copy-Move* yang telah dilakukan *postprocessing* seperti ditunjukkan pada gambar 10.



Gambar 11. Hasil Pengujian dengan DCT Terkuantisasi Matrik 1 pada Citra (a) *postprocessing rotation*, (b) *postprocessing scaling*, (c) *postprocessing noise*

Sementara pada DCT Terkuantisasi baik dengan Matrik 1, 2 ataupun 3, citra *Copy-Move* yang telah dilakukan *postprocessing* masih dapat terdeteksi. Ini dikarenakan perubahan kecil pada nilai pixel setelah dibagi dan dibulatkan akan menghasilkan nilai yang sama sehingga masih dapat dilakukan deteksi. Nilai matrik kuantisasi sangat mempengaruhi. Apabila nilainya terlalu kecil, maka kemungkinan deteksi pada adanya *postprocessing* akan lebih sulit. Tapi bila nilainya terlalu besar, maka akan semakin banyak daerah *false match*.



Gambar 12. Contoh *False Match* pada Frekuensi Tinggi

Daerah-daerah yang memiliki warna sejenis dan berukuran cukup luas seperti warna putih awan atau langit juga berpotensi mengalami *false match* walaupun itu dengan DCT yang nilainya detail. Pada penelitian ini penulis membagi citra kedalam blok pixel berukuran 8x8 yang kemudian dilakukan proses DCT. Sehingga apabila terdapat 2 buah blok pixel 8x8 dengan nilai pixel yang sama dan terletak pada koordinat yang sama pula, maka akan memberikan nilai transformasi yang sama. Sehingga daerah-daerah luas dengan nilai pixel sama akan berpotensi mengalami *false match*.

Pendeteksian tidak dapat dilakukan pada citra berekstensi JPG/JPEG. Hal ini dikarenakan adanya kompresi pada citra yang disimpan dalam format JPG/JPEG. Pada penelitian citra yang digunakan sebagai citra uji berekstensi PNG dan Bitmap.

Penelitian ini menggunakan nilai threshold pada proses pencocokkan. Tujuannya adalah untuk mengurangi *false match*. Untuk citra asli digunakan threshold bawah sebesar 30. Sehingga hanya jika terdapat blok 8x8 sejumlah 30 yang posisinya bersesuaian, maka akan ditandai sebagai *Copy-Move*. Akan tetapi pada citra *Copy-Move* hasil *postprocessing*, nilai threshold bawah lebih kecil yaitu 3. Hal ini karena setelah dilakukan proses *postprocessing*, blok-blok pixel yang nilainya sama menjadi lebih sedikit, sehingga bila penetapan threshold bawah terlalu tinggi maka tidak akan terdeteksi. Nilai threshold ini bukanlah nilai yang

paten dalam penelitian. Karena besarnya nilai threshold tergantung pada ukuran daerah *Copy-Move* nya. Hal ini sulit untuk menentukan nilai threshold terbaik, karena pada proses deteksi kita tidak tahu ukuran dari daerah *Copy-Move* yang dilakukan. Bisa saja ukuran citra luas akan tetapi daerah yang di *Copy-Move* hanya kecil atau sebaliknya. Sehingga jika kita menetapkan nilai threshold minimal terlalu tinggi, maka daerah *Copy-Move* tidak akan terdeteksi dan sebaliknya, jika menentukan nilai threshold minimal terlalu kecil maka akan terdapat banyak *false match*.

## 4. Kesimpulan dan Saran

### 4.1 Kesimpulan

Berdasarkan hasil penelitian yang dilakukan dapat ditarik beberapa kesimpulan diantaranya:

- Aplikasi ini mampu melakukan pendeteksian *Copy-Move Forgery* pada citra digital dengan tingkat akurasi yang cukup baik walaupun terdapat beberapa *false match*.
- Pendeteksian dengan akurasi terbaik dilakukan dengan metode DCT Terkuantisasi Matrik 1 dengan presentase *Copy-Move* yang **Terdeteksi** pada frekuensi rendah adalah 22,22%, pada frekuensi menengah sebesar 25% dan pada frekuensi tinggi 11,11%. Juga dihasilkan nilai **Terdeteksi dengan False Match** sebesar 55,56% pada frekuensi rendah, 41,67% pada frekuensi menengah, dan 72,22% pada frekuensi tinggi.
- Penggunaan jenis frekuensi juga berpengaruh dalam akurasi pendeteksian dan *false match* yang dihasilkan.
- Kesalahan deteksi atau *false match* masih sering terjadi pada citra yang memiliki daerah luas dengan warna pixel sama.
- Pendeteksian pada citra *Copy-Move postprocessing* belum cukup baik dan hanya mampu menangani sedikit perubahan pada pixel.
- Citra dengan format JPG/JPEG tidak dapat dideteksi karena mengalami kompresi.

### 4.2 Saran

Pada penelitian selanjutnya penulis dapat memberikan saran-saran sebagai berikut:

- Terdapat banyak jenis matrik kuantisasi, baik yang sudah ditentukan nilainya maupun menghitung sendiri dengan rumus yang telah disediakan. Nilai matrik kuantisasi memberikan pengaruh pada hasil pendeteksian. Pada penelitian selanjutnya bisa dikembangkan untuk mampu men *generate* matrik kuantisasi yang memberikan hasil deteksi terbaik pada proses pendeteksian secara otomatis.
- Penelitian selanjutnya bisa lebih mengembangkan lagi dalam hal peningkatan

akurasi pada pendeteksian citra *Copy-Move postprocessing*.

**Daftar Pustaka:**

- Arya Yudhi Wijaya, S. A. M. d. H. S., 2017. PENGEMBANGAN METODE BLOCK MATCHING. *JUTI: Jurnal Ilmiah Teknologi Informasi*, Volume XV No. 1, pp. 84-94.
- Derwin Rony Sina, A. h., 2016. Deteksi Copy Move Forgery Pada Citra Menggunakan. *IJEIS*, Volume VI No. 1, p. 26.
- Farid, A. C. P. a. H., 2004. Exposing Digital Forgeries by Detecting Duplicated Image Regions.
- G. K. S. Gaharwar, P. V. V. N. R. D. G., 2015. *COMPREHENSIVE STUDY OF DIFFERENT TYPES*. New Delhi, s.n.
- Madenda, S., 2015. *Pengolahan Citra dan Video Digital*. Jakarta: Erlangga.
- Madenda, S., 2015. Transformasi Kosinus Diskret Terkuantisasi. In: *Pengolahan Citra dan Video Digital*. Jakarta: Erlangga, p. 226.
- Putra, D., 2010. *Pengolahan Citra Digital*. Yogyakarta: Andi.
- Sunil Kumar, P. K. D. S., 2011. Copy-Move Forgery Detection in Digital. *International Journal on Computer Science and Engineering (IJCSE)*, Volume III No. 2, p. 652.
- Zhang, J. F. Z. d. S. Y., 2008. *A new approach for detecting copy-move forgery in digital images*. s.l., s.n.
- Zulfan, F. A. d. R. M., 2016. DETEKSI PEMALSUAN CITRA DENGAN TEKNIK COPY-MOVE. V No. 2(11).