

Deteksi Copy-Move pada Pemalsuan Citra Menggunakan Local Binary Pattern dan SVD-matching

Copy-move Image Forgery Detection using Local Binary Pattern and SVD-matching

Dian Masmawati¹, Tjokorda A. Budi Wirayuda, ST., MT.², Febryanti Stehvanie, ST., MT.³

Program Studi Teknik Informatika, Fakultas Informatika, Telkom University, Bandung

dianmasmawati@gmail.com¹, cokagung2001@gmail.com², febryantisthevanie@gmail.com³

Abstrak

Image Forgery merupakan tindakan pemalsuan citra yang dilakukan secara illegal. Pemalsuan Citra juga dapat didefinisikan sebagai proses manipulasi dari suatu citra digital untuk menyembunyikan beberapa informasi yang penting ada suatu citra. Salah satu metode manipulasi citra yang paling umum digunakan adalah *Copy-move*. Dengan adanya berbagai perangkat lunak pengolahan citra seperti Photoshop, pemalsuan citra sangat mudah dilakukan. Oleh karena itu, dibutuhkanlah suatu cara untuk mendeteksi pemalsuan citra.

Pada tugas akhir ini dilakukan deteksi *copy-move* pada pemalsuan citra digital dengan menggunakan metode *Local Binary Pattern* (LBP) dan *SVD-matching*. Dengan menggunakan kedua metode tersebut dilakukan pencocokan antar blok-blok piksel citra.

Berdasarkan hasil penelitian yang dilakukan, menunjukkan bahwa dengan menerapkan metode ini dihasilkan suatu metode yang dapat mendeteksi *copy-move* pada pemalsuan citra. Tingkat akurasi sistem terbaik yang diperoleh dari pengujian berbasis area mencapai 62.33% kemudian sistem mampu mendeteksi citra *copy-move* dan citra asli dengan baik karena memiliki akurasi mencapai 81.7% dan mempunyai nilai TPR dan TNR sebesar 0.800 ketika *threshold* stabil yang digunakan adalah 0.0051.

Kata Kunci : *Image forgery, copy-move, citra, deteksi, Local Binary Pattern (LBP), SVD-matching*

1. Pendahuluan

Citra digital secara luas digunakan sebagai media komunikasi untuk penyampaian informasi. Keaslian dari suatu citra memiliki peran penting dalam banyak bidang, termasuk penyelidikan forensik, investigasi kriminal, sistem *surveilans*, badan intelijen, pencitraan medis, dan jurnalisme. Dengan adanya berbagai perangkat lunak pengolahan citra membuat proses manipulasi citra menjadi lebih mudah dilakukan daripada sebelumnya [2] serta adanya berbagai media sosial sebagai sarana penyebaran membuat citra yang telah dimanipulasi sangat mudah tersebar kepada masyarakat. *Image Forgery* merupakan tindak pemalsuan citra yang dilakukan secara illegal. Pemalsuan Citra juga dapat didefinisikan sebagai proses manipulasi dari suatu citra digital untuk menyembunyikan beberapa informasi yang penting pada suatu citra. Pada umumnya ada tiga jenis pemalsuan citra, diantaranya *retouching*, *splicing*, dan *copy-move*. *Copy-move* merupakan salah satu metode manipulasi citra yang paling umum digunakan. *Copy-move* adalah proses menyalin salah satu bagian dari sebuah citra dan menyisipkannya ke bagian lain pada citra yang sama dengan posisi yang berbeda. [3] Dengan adanya tindak pemalsuan citra, penting untuk mengotentikasi keaslian citra digital, mengidentifikasi sumber asal citra, dan mendeteksi pemalsuan citra.

Banyak metode yang digunakan untuk mendeteksi pemalsuan citra *copy-move*. Metode pertama yang diusulkan yaitu dengan menggunakan koefisien *discrete cosine transforms* (DCT) dengan metode ini citra dibagi menjadi *overlapping blocks* kemudian blok tersebut diurutkan secara leksikografi, dan setiap bloknya dilakukan DCT untuk menemukan blok yang sama [4]. Kemudian metode selanjutnya mengusulkan metode *Local Binary Pattern* untuk menemukan area duplikasi dari citra yang mencurigakan menggunakan citra *grayscale* dan *Euclidean distance* untuk proses pencocokan [5].

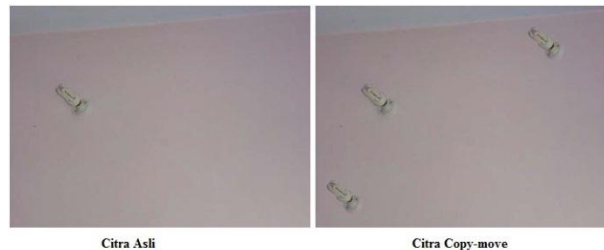
Dari beberapa metode yang telah disebutkan, dapat diketahui bahwa blok-blok piksel citra bisa menjadi acuan untuk mendeteksi pemalsuan citra. Pada penelitian tugas akhir ini digunakan *Local Binary Pattern* dan

SVD-matching untuk mendeteksi pemalsuan citra *copy-move* dengan menerapkan blok-blok piksel menjadi acuannya. *Local Binary Pattern* adalah salah satu metode ekstraksi ciri yang sensitif terhadap perbedaan dan digunakan *SVD-matching* pada proses pencocokan untuk menemukan atau mendeteksi area citra yang telah dimanipulasi *copy-move*.

2. Tinjauan Pustaka

2.1. Copy-Move

Copy-move merupakan salah satu metode manipulasi citra *clone* yang paling umum digunakan [6]. Berikut merupakan contoh citra *copy-move*.



Gambar 2.4 Gambar Copy-move [12]

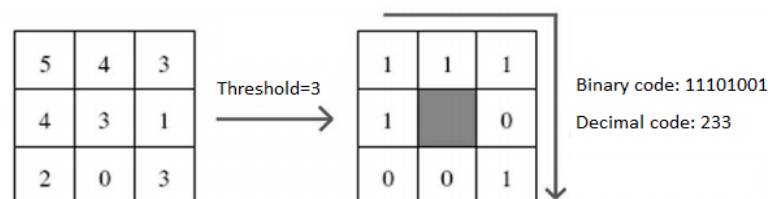
Copy-move adalah proses menyalin salah satu bagian dari citra dan menyisipkannya ke bagian lain pada citra yang sama dengan posisi yang berbeda [3]. Hal ini biasanya dilakukan dengan maksud untuk membuat sebuah objek "menghilang" dari gambar, dengan menutupinya dengan bagian lain dari gambar. Daerah yang disalin kemungkinan akan menyatu dengan latar belakang dengan kasat mata manusia kemungkinan tidak dapat dengan mudah membedakan objek apapun yang mencurigakan [4]. Ketika *copy-move* dilakukan dengan benar maka akan sangat sulit untuk mendeteksinya secara visual [6].

2.2. Local Binary Pattern (LBP)

Operator LBP adalah salah satu deskriptor tekstur terbaik dan telah banyak digunakan dalam berbagai aplikasi. LBP didefinisikan untuk memerintahkan set perbandingan dan menghitung perbedaan intensitas piksel antara piksel pusat dengan 8 piksel disekitarnya dan mempertimbangkan hasil tersebut sebagai angka biner. Tekstur dapat digambarkan oleh histogram dari nilai label ini [3]. Dapat dihitung pula dengan rumus berikut :

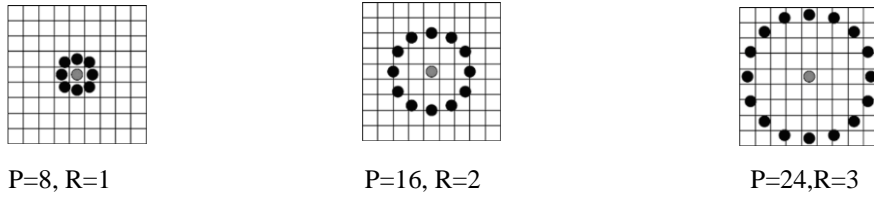
$$LBP = \sum_{i=0}^7 s(g_i - g_c) 2^i, \text{ dimana } s(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases} \quad (1)$$

Dari rumus tersebut digambarkan bahwa g_c adalah piksel pusat dan g_i mewakili piksel di sekitarnya [7]. Ketika piksel pusat nilainya lebih besar dari piksel sekelilingnya maka piksel sekelilingnya akan bernilai 0, jika sebaliknya piksel disekeliling akan bernilai 1. Contoh Komputasi LBP adalah sebagai berikut:



Gambar 2.1 Komputasi LBP [11]

LBP juga dapat diekstraksi pada lingkungan melingkar (P, R), contoh ekstraksi LBP *circular* dapat dilihat pada gambar berikut :



Gambar 2.2 LBP Circular [5]

P merupakan jumlah piksel tetangga dan R adalah radius atau jari-jari ketetanggaan.

2.3. SVD-matching

2.3.1. Singular Value Decomposition

Singular Value Decomposition (SVD) adalah teknik matematika yang berasal dari teorema berbasis aljabar linear. SVD menyatakan bahwa matrix A dapat dipecah menjadi *product of three matrices*, matriks ortogonal U, matriks diagonal S, dan transpose dari matrix ortogonal V [8]. Teoremanya seperti pada rumus berikut:

$$A_{mn} = U_{mm}S_{mn}V_{nn}^T \quad (2)$$

Kolom dari U adalah vektor eigen ortonormal dari AA^T , kolom dari V adalah vektor eigen dari $A^T A$, dan S adalah matriks diagonal berisi akar kuadrat dari nilai eigen dari U atau V di *descending order* [9].

2.3.2. Jarak Euclidean

Jarak *Euclidean* adalah jarak yang paling umum digunakan pada data numerik [10]. Jarak *Euclidian* merupakan suatu metode pencarian kedekatan nilai jarak dari 2 buah variabel, selain mudah metode ini juga tidak memakan waktu proses yang cepat. Mengukur jarak adalah bagian penting dari model *vector* berdasarkan deskriptor dua komponen, label jarak untuk setiap titik, terlihat bahwa peta jarak *Euclidean* dapat dihasilkan oleh algoritma sekuensial yang efektif. Dua buah titik $p_1 = (x_1, y_1)$ dan $p_2 = (x_2, y_2)$ jaraknya dengan menggunakan rumus *Euclidean* adalah :

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \quad (3)$$

2.3.3. Normalized Cross Correlation (NCC)

Korelasi silang ternormalisasi adalah metode perhitungan kecocokan citra yang umum digunakan. Metode ini mengukur nilai korelasi satu kanal warna berupa warna keabu-abuan. Koefisien korelasi yang dihasilkan memiliki nilai pada rentang -1 hingga 1. Nilai 1 menandakan jika objek atau gambar dengan pembandingnya itu identik. Nilai 0 berarti objek tersebut tidak berkorelasi. Nilai -1 menunjukkan bahwa objek pembandingnya merupakan kebalikan atau *inverse* dari objek asli.

NCC dapat digambarkan dengan formula berikut :

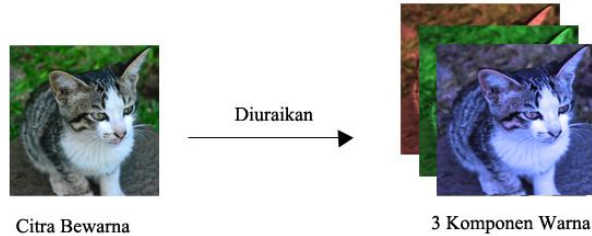
$$C_{ij} = \frac{1}{n} \sum_{x,y} \frac{(f(x,y) - \bar{f})(t(x,y) - \bar{t})}{\sigma_f \sigma_t} \quad (4)$$

Keterangan :

- n = n adalah jumlah piksel dalam t (x, y) dan f (x, y)
- \bar{f} = rata-rata dari f
- \bar{t} = rata-rata dari t
- σ_f = standar deviasi dari f
- σ_t = standar deviasi dari t

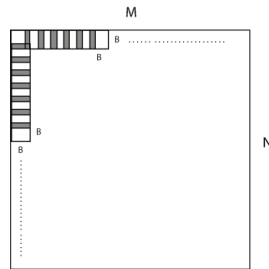
3. Rancangan Sistem

Sistem deteksi yang dibangun terdiri dari 3 proses utama yaitu *preprocessing* citra uji, ekstraksi ciri dengan menggunakan metode *Local Binary Pattern* (LBP) dan pencocokan blok dengan menggunakan *SVD-matching*. Proses *preprocessing* pada citra uji ini adalah dengan memisahkan setiap layer yang ada pada citra seperti gambar berikut :



Gambar 3.1 Penguraian Citra RGB

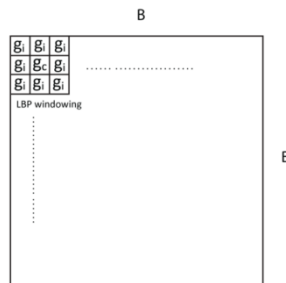
Dapat dilihat dari gambar 3.1 citra uji yang digunakan akan dipisahkan menjadi layer *Red*, *Green*, dan *Blue*. Setelah ke-tiga layer citra dipisahkan, kemudian layer-layer citra tersebut dipotong menjadi blok-blok yang *overlapping* atau saling tumpang tindih. Proses tersebut dapat digambarkan sebagai berikut :



Gambar 3.2 Gambaran Proses *Overlapping Block*

Citra dengan ukuran MXN dipotong menjadi blok-blok berukuran BXB dimulai dari pojok kanan atas sampai pojok kanan bawah citra. Ukuran dari blok-blok gambar yang digunakan adalah 16×16 piksel dengan ukuran atau *space overlapping* yang selanjutnya akan dicari konfigurasi terbaiknya. *Space overlapping* adalah jumlah piksel baris atau kolom yang sama pada setiap blok citra yang telah dipotong. Pada gambar tersebut *Space overlapping* digambarkan dengan bagian yang berwarna abu-abu.

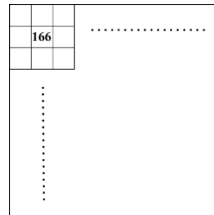
Selanjutnya dilakukan proses ekstraksi ciri dari blok-blok citra *overlapping*. Pada proses ini citra pada setiap komponen gambar RGB pada setiap blok yang telah dipotong *overlapping* di ekstraksi cirinya dengan menggunakan *Local Binary Pattern*. Tahap proses ekstraksi ciri adalah sebagai berikut :



Gambar 3.3 Ekstraksi Ciri dengan LBP

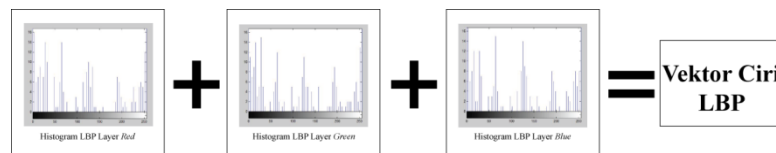
Kotak LBP yang digunakan untuk *default* dari sistem yang dibangun adalah 3×3 dengan $P=8$ dan $R=1$. Nilai *LBP* dicari dengan menggunakan rumus (1). Hasil dari perhitungan tersebut menghasilkan nilai biner 8 bit.

Kemudian nilai biner 8 bit yang dihasilkan akan dirubah kedalam desimal kembali kemudian nilai tersebut disimpan. Hasil dari konversi nilai biner ke nilai desimal tersebut selanjutnya dilakukan proses *mapping* dan akan menghasilkan citra baru hasil ekstraksi seperti gambar berikut :



Gambar 3.4 Replacement mapping hasil ekstraksi LBP

Setelah proses ekstraksi ciri dengan LBP itu selesai, blok-blok citra disetiap layer diambil nilai histogramnya dengan ukuran bin sebanyak 64 bin. Kemudian tiap histogram LBP dari layer-layer tersebut dijumlahkan nilainya. Proses ini dapat diilustrasikan dengan gambar berikut :



Gambar 3.4 Gambar vektor ciri LBP

Hasil akhir dari proses ini adalah vektor ciri dengan panjang 64 untuk setiap blok citra. Untuk satu keseluruhan citra dapat dikatakan jika hasil ciri yang dihasilkan adalah $N \times 64$. N menunjukkan jumlah blok *overlapping* yang dihasilkan. Selanjutnya vektor ciri tersebut akan masuk pada proses pencocokan blok dengan menggunakan *SVD-matching*. *SVD-matching* sendiri terdiri dari dua tahap diantaranya :

a. Menghitung Kedekatan Matriks

Untuk mencari dan menghitung kedekatan matriks pasangan blok menggunakan jarak *Euclidean*. Setiap nilai merepresentasikan jarak antar pasangan blok yang terdeteksi. Vektor ciri dengan panjang 64 untuk setiap blok citra yang telah didapat sebelumnya kemudian dihitung jarak antar bloknya menggunakan jarak *Euclidean*. Untuk menghitung jarak *Euclidean* dengan menggunakan rumus (3) setelah itu dicari nilai minimum yang berada pada baris dan kolom dari hasil matriks jarak *Euclidean*. Hasil akhir dari proses ini adalah matriks jarak *Euclidean* yang digunakan untuk proses perhitungan kemiripan antara matriks dan pasangan-pasangan blok yang diduga pasangan blok area termanipulasi *copy-move*.

b. Menghitung Kemiripan Matriks

Proses selanjutnya adalah menghitung kemiripan antar matriks menggunakan *normalized cross-correlation* dengan rumus (5). Untuk menghitung korelasi antar blok digunakan vektor ciri dengan panjang 64 untuk setiap blok citra dari proses ekstraksi ciri. Setiap nilai perhitungan korelasi antar blok yang dilakukan disimpan pada matriks korelasi C_{ij} . Hasil akhir dari perhitungan dengan *normalized cross-correlation* menghasilkan matriks korelasi C berukuran $n \times n$.

Kemudian matriks $n \times n$ yang dihasilkan dari perhitungan korelasi dan matriks jarak dari perhitungan menggunakan jarak *Euclidean* tersebut selanjutnya diproses menggunakan *Gaussian Kernel*. C_{ij} yang didapat digunakan pada perhitungan matriks G seperti berikut:

$$G_{ij} = \frac{(C_{ij}+1)}{2} e^{-\frac{d_{ij}^2}{2\sigma^2}} \quad (5)$$

Hasil dari matriks G kemudian difaktorisasi dengan SVD menggunakan rumus (3) sebagai berikut :

$$G = USV^T \quad (6)$$

Terakhir elemen sepanjang diagonal dari matriks S di-set nilainya menjadi 1. Kemudian menghasilkan matriks W yang dimodif menjadi :

$$W = UXV^T \quad (7)$$

dimana X sama dengan matriks S kecuali elemen diagonalnya digantikan dengan nilai 1. Matriks W memiliki bentuk yang sama dengan matriks korelasi. Dua fitur i dan j dipasangkan jika W_{ij} keduanya merupakan elemen tertinggi pada baris dan elemen tertinggi di kolomnya. Dengan memilih semua elemen ini pada W , pencocokan

fitur antar fitur dapat dilakukan. Hasil dari proses ini berupa pasangan blok yang diduga menjadi pasangan blok area terduplikasi *copy-move*.

4. Pengujian dan Hasil Analisis

Pengujian sistem dilakukan untuk mengetahui kelebihan dan kekurangan dari sistem deteksi *copy-move* pada pemalsuan citra. Implementasi dan pengujian sistem ini menggunakan perangkat lunak MATLAB 2012b.

4.1. Citra Uji

Citra uji yang digunakan adalah sebagai berikut :

Tabel 4-1 Citra Uji yang digunakan

Pengujian ke-	Jumlah Citra <i>Copy-move</i>	Jumlah Citra Asli
1	18	-
2	30	30

Citra *copy-move* dan citra yang asli atau otentik yang digunakan berukuran 160x160 piksel.

4.2. Analisis Hasil Pengujian

4.2.1. Pengujian 1

4.2.1.1. Skenario 1

Pengujian skenario pertama untuk mencari konfigurasi terbaik untuk presentase *space overlapping* blok yang digunakan sistem dalam mendeteksi adanya *copy-move* yaitu dengan 25% dan 50% dari ukuran blok *overlapping*. Berikut hasil yang didapatkan dari hasil pengujian ini :

Tabel 4-2 Hasil pengujian Skenario 1

<i>Space Overlapping</i>	<i>Running Time (s)</i>	Akurasi
4	136.462	61.7%
8	31.632	62.3%

Dari tabel 4-2 dapat dilihat bahwa ketika presentase ukuran *space overlapping* yang digunakan kecil maka waktu komputasi sistem untuk mendeteksi area *copy-move* pada pemalsuan citra akan lebih lama. Ketika *space overlapping* yang digunakan lebih besar proses pemotongan citra menjadi blok-blok yang *overlapping* menjadi lebih cepat karena jumlah blok yang dihasilkan semakin sedikit. Selain itu hasil akurasi deteksi yang dihasilkan lebih baik ketika presentase *space overlapping* adalah 50% dari ukuran blok *overlapping*nya yaitu sebesar 62.3%.

4.2.1.2. Skenario 2

Pengujian skenario kedua untuk mengetahui tingkat akurasi sistem terhadap pengaturan parameter-parameter yang ada pada *Local Binary Pattern*. Pengujian dilakukan dengan merubah parameter P yaitu jumlah *point* atau jumlah titik ketetanggaan dan R yaitu radius dari titik ketetanggaan yang digunakan. Berikut hasil yang didapatkan dari hasil pengujian kedua:

Tabel 4-3 Hasil pengujian Skenario 2

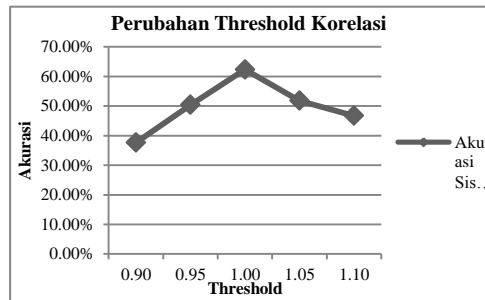
Jumlah P	Besar R	Akurasi
8	1	62.33%
	2	58.42%
	3	53.90%
16	1	53.32%
	2	43.98%

Dari hasil pengujian yang dilakukan, penurunan akurasi sistem terjadi seiring dengan perubahan parameter dari LBP. Ketika pengujian menggunakan konfigurasi parameter LBP dengan P = 8 dan R = 1

menunjukkan tingkat akurasi yang paling baik dibandingkan dengan konfigurasi dengan parameter lainnya. Hal ini terjadi proses ekstraksi ciri akan semakin mendetail dan lebih mengenali tekstur dari citra dengan baik. Sehingga pada proses pencocokan akan menghasilkan hasil deteksi yang lebih akurat.

4.2.1.3. Skenario 3

Pengujian skenario ketiga untuk untuk mengetahui pengaruh perubahan penggunaan tabel dan threshold SVD pada proses pencocokan citra yang telah dimanipulasi *copy-move*. Yang digunakan untuk pengujian ini adalah 100% penggunaan tabel X pada SVD, kemudian parameter yang diujikan adalah perubahan nilai perkalian dari *threshold* rata-rata korelasi dengan nilai 0.90, 0.95, 1.00, 1.05 dan 1.1. Berikut hasil yang didapatkan dari hasil pengujian :

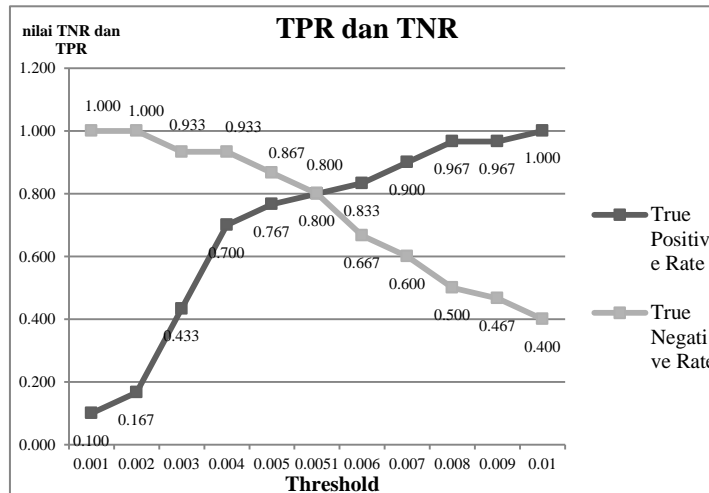


Gambar 4.1 Grafik Perubahan *Threshold* Korelasi terhadap Akurasi

Dari Gambar 4.1 Grafik perubahan *threshold* korelasi terhadap Akurasi sistem yang dihasilkan, dapat dilihat jika nilai rata-rata korelasi dikalikan dengan kurang dari 1 akurasi yang dihasilkan tidak mencapai 40% kemudian mengalami kenaikan akurasi ketika nilai rata-rata korelasi dikalikan dengan 1. Pada saat itu akurasi sistem mencapai pada puncaknya.

4.2.1.4. Skenario 4

Pengujian skenario keempat untuk untuk menguji performansi sistem dalam deteksi *copy-move* pada pemalsuan citra dan menentukan citra uji yang digunakan merupakan citra yang dimanipulasi *copy-move* atau merupakan citra asli. Berikut hasil yang didapatkan dari hasil pengujian :



Gambar 4.2 Grafik Perubahan *Threshold* terhadap TPR dan TNR

Dari Gambar 4.2 Grafik Perubahan *Threshold* terhadap TPR dan TNR, dapat dilihat jika perpotongan antara nilai TPR dan TNR terjadi ketika *threshold* yang digunakan pada rentang nilai 0.0051 hal ini menyatakan kestabilan ditentukannya citra *copy-move* yang dinyatakan *copy-move* dan citra asli yang dinyatakan asli ada pada rentang nilai tersebut. Hal tersebut terjadi karena nilai rata-rata pasangan blok untuk citra uji *copy-move* sebagian besar kurang dari 0.0051 dan nilai rata-rata pasangan blok untuk citra uji asli sebagian besar lebih dari 0.0051. Pada saat *threshold* yang digunakan adalah 0.0051 sistem

lebih stabil untuk mendeteksi citra uji *copy-move* maupun asli. Nilai *TPR* menyatakan akurasi citra *copy-move* yang dinyatakan *copy-move* sedangkan nilai *TNR* menyatakan akurasi citra asli yang dinyatakan asli..

Kesimpulan

Berdasarkan hasil pengujian dan analisis yang telah dilakukan dapat diambil kesimpulan bahwa Sistem Deteksi *Copy-Move* pada Pemalsuan Citra Menggunakan *Local Binary Pattern* dan *SVD-matching* memberikan hasil dengan tingkat akurasi mencapai 62.33% untuk pengujian berbasis area. Sistem baik dalam mendeteksi citra *copy-move* dan citra asli karena memiliki akurasi mencapai 81.7%. Selain itu mempunyai nilai *TPR* dan *TNR* sebesar 0.800 ketika *threshold* stabil yang digunakan adalah 0.0051.

Daftar Pustaka

- [1] M. Sridevi, Siddhant Sanyam, and C. Mala, "Comparative Study of Image Forgery and Copy-Move Techniques," 2012.
- [2] Shao-Jie Sun, Qiong Wu, and Guo-Hui Li, "Detection of Image Compositing Based on a Statistical Model for Natural Images," 2009.
- [3] Motasem AlSawadi, Ghulam Muhammad, Muhammad Hussain, and George Bebis, "Copy-Move Image Forgery Detection Using Local Binary Pattern and Neighborhood Clustering," *IEEE Computer Society*, 2013.
- [4] J Fridrich, D Soukal , and J Lukas, "Detection of copy–move forgery in digital images," 2003.
- [5] Ramandeep Kaur, "Copy-Move Forgery Detection Utilizing Local Binary Patterns," *International Journal of Emerging Technologies in Computational and Applied Sciences*, p. 292, 2014.
- [6] Hany Farid, "Image Forgery Detection," *IEEE SIGNAL PROCESSING MAGAZINE*, vol. 26, no. 2, pp. 16-25, 2009.
- [7] Ning Zheng, Yixing Wang, and Ming Xu, "A LBP-Based Method for Detecting Copy-Move Forgery with Rotation," *Springer Science+Business Media Dordrecht*, pp. 261-267, 2013.
- [8] Kirk Baker, "Singular Value Decomposition Tutorial," 2013.
- [9] Takwa Chihaoui, Sami Bourouis, and Kamel Hamrouni, "Copy-move Image Forgery Detection Based On SIFT Descriptors and SVD-matching ," *Conference on Advanced Technologies for Signal and Image Processing IEEE*, 2014.
- [10] G Goujoun , Chaoqun, and W Jianhong, "Data Clusterin :Theory, Algorithms, and Applications," *ASA*, 2007.
- [11] Abdenour Hadid, Timo Ahonen, Guoying Zhao, and Matti Pietikainen, *Computer Vision Using Local Binary Patterns*. Utrecht, Netherlands: Springer, 2011.
- [12] Sunil Kumar, Jagannath Desai, and Shaktidev Mukherjee, "A Fast DCT Based Method for Copy Move Forgery Detection," *IEEE Second International Conference*, p. 651, 2013.