

# Segmentasi Gambar Warna Menggunakan Sauvola Modifikasi Fuzzy C-Means (SMFCM)

Gilang Bayu Adhi<sup>1</sup>, Irawan Dwi Wahyono<sup>2</sup>

Institut Teknologi Sepuluh Nopember

Jalan ITS Raya 60111, Surabaya

e-mail: gilangbayu.adhi@gmail.com<sup>1</sup>, irawan2712@gmail.com<sup>2</sup>

## Abstrak

*Dalam proses segmentasi citra berwarna, beberapa metode memiliki kelebihan dan kekurangan. Ada satu metode segmentasi citra berwarna yang dapat mensegmentasi warna dengan baik, akan tetapi memiliki kekurangan yaitu memiliki peak dan valley kecil pada histogramnya yang menyebabkan hasil segmentasi kurang homogen. Untuk mengatasi permasalahan peak dan valley kecil ini, maka penulis ingin mencoba suatu metode baru dengan menggunakan metode Sauvola Modifikasi Fuzzy C-Means hybrid (SMFCM). Metode ini menggabungkan algoritma Modifikasi Sauvola yang telah dimodifikasi dengan algoritma Fuzzy C-means. Hasil penelitian menunjukkan bahwa metode ini dapat mengurangi peak dan valley kecil sampai 25%, sehingga warna yang serupa pada citra berwarna lebih homogen. Jumlah region warna juga berkurang sebanyak 54%. Hasil penelitian ini menunjukkan persentase kegagalan atau error rate sebesar 21%.*

**Kata kunci:** Segmentasi, Sauvola modifikasi, Fuzzy C-Means, Histogram

## Abstract

*In the image color segmentation process, several methods have its own advantages and disadvantages. There is one method of color image segmentation that segments the image very well, but has the disadvantage that it has a small peak and valley in its histogram and causing in less homogeneous segmentation results. To overcome the problem of this small peak and valley, we would like to try a new method using modified Sauvola Fuzzy C-Means hybrid. This method combines the algorithm of modified Sauvola with Fuzzy C-Means algorithm. Results showed that this method can reduce small peak and valley up to 25%, so that the similar color is more homogenous. Number of color regions also reduced by 54%. Result show that this study has a error rate of 21%.*

**Keywords:** Segmentation, Modified Sauvola, Fuzzy C-Means, Histogram

## 1. Pendahuluan

Pada gambar warna 24-bit, jumlah warna yang unik biasanya melebihi setengah dari ukuran gambar dan dapat mencapai 16 juta warna. Sebagian besar dari warna ini tidak dapat dibedakan oleh mata manusia yang hanya dapat mengenali 30 warna. Untuk semua warna unik ini, mereka dapat digabungkan untuk membentuk daerah yang homogen yang mewakili objek pada gambar sehingga gambar akan menjadi lebih bermakna dan mudah untuk dianalisa. Pada proses citra dan visi komputer, segmentasi gambar berwarna bertujuan untuk menganalisa gambar dan pengenalan pola [1]. Segmentasi gambar berwarna merupakan proses mempartisi sebuah gambar menjadi beberapa daerah yang homogen atas dasar persamaan karakteristik tertentu [2].

Gambar dapat dirubah menjadi binerisasi dalam bentuk histogram. Banyak metode dalam membuat warna menjadi binerisasi diantaranya metode *Otsu* yang mana merubah gambar berwarna menjadi keabuan yang lebih dikenal dengan global *thresholding*. Metode lainnya adalah berupa *thresholding* lokal yang bersifat adaptif atau disebut jendela lokal dengan

memperhatikan *pixel* tetangga. Metode yang menggunakan *local thresholding* diantaranya adalah *Sauvola* [3,4]. Dalam hal komputasi untuk menghasilkan *output*, metode *Otsu* lebih cepat dibandingkan metode *Sauvola*, akan tetapi dalam akurasi dan hasil, metode *Sauvola* lebih baik dibandingkan dengan metode *Otsu*. Metode *Sauvola* yang telah dimodifikasi dalam konsep *integral image* dapat menyamai kecepatan komputasi pada metode *Otsu*.

Gambar warna dapat dibagi dalam 3 histogram, yaitu warna merah, hijau dan biru. Pembuatan histogram dapat dilakukan secara global maupun lokal *thresholding*, karena terdapat 3 warna jadi menjadi 3 dimensi binerisasi yang mana memiliki *cluster* sendiri jika dijadikan satu kembali [5].

Ada 2 metode dalam melakukan pengelompokan *cluster* yaitu *K-Means* dan *Fuzzy C-Means* [6,7]. Keduanya mencari jarak optimal antara *centroid number*, *cluster* dan *pixel* dari 3 warna yaitu *Red*, *Green* dan *Blue*. Dalam metode segmentasi muncul beberapa gabungan algoritma diantaranya *Histogram Thresholding Fuzzy C-Means hybrid* (HTFCM) [2].

*Histogram Thresholding Fuzzy C-Means hybrid* (HTFCM) merupakan metode pendekatan baru pada pengenalan pola. Metode ini membagi sebuah gambar berwarna menjadi 3 *layer*, yaitu *layer red*, *green* dan *blue*. Setelah gambar berwarna dibagi menjadi 3 *layer*, kemudian dibuat histogramnya menggunakan global *thresholding*.

Akan tetapi, metode *histogram thresholding* menghasilkan banyak puncak dan lembah kecil pada berbagai daerah datar *histogram*-nya. Masalah puncak dan lembah ini dapat membuat warna suatu citra menjadi kurang homogen. Kurang homogennya warna ini dapat mempengaruhi proses segmentasi citra.

Paper ini, mengajukan suatu pendekatan baru dengan menggunakan metode hibrida *Sauvola* Modifikasi dan *Fuzzy C-Mean* (SMFCM). Metode SMFCM ini dapat mengatasi permasalahan segmentasi pada HTFCM yang menghasilkan puncak dan lembah kecil pada 3 *layer* daerah datar *histogram* pada metode HTFCM. Dengan pengurangan puncak dan lembah ini, *thresholding* suatu citra menjadi lebih homogen.

## 2. *Sauvola* Modifikasi *Fuzzy C-Means*

Dalam melakukan segmentasi, pada *paper* ini dilakukan dengan 2 tahap yaitu modul modifikasi *Sauvola* atau *local* adaptif *integral image* dan modul *Fuzzy C-Mean*. Dalam modul modifikasi *Sauvola* dilakukan 3 tahap yaitu: Langkah pertama adalah pembuatan *histrogram* dengan *Sauvola* modifikasi pada 3 warna yaitu merah, hijau dan biru. Langkah kedua adalah insialisasi Regional dalam 3 warna dan langkah berikutnya adalah penggabungan 3 warna atau *merging* berupa *cluster*.

### 2.1. Histogram dengan *Sauvola* Modifikasi

Gambar dokumen dalam *grayscale* yang mana  $g(x,y) \in [0,255]$  menjadi intensitas *pixel* pada  $(x,y)$ . Pada teknik *local adaptive thresholding* [4], tujuan utama dalam mencari *threshold*  $t(x,y)$  untuk masing – masing *pixel* dalam persamaan (1).

$$o(x,y) = \begin{cases} 0, & \text{jika } g(x,y) \leq t(x,y) \\ 255, & \text{yang lainnya} \end{cases}, \quad (1)$$

dimana  $o(x,y)$  adalah intensitas *pixel* pada koordinat  $x$  dan  $y$ .

Pada metode binerisasi *Sauvola*, *threshold*  $t(x,y)$  dihitung menggunakan *mean*  $m(x,y)$  dan standar deviasi  $s(x,y)$  pada intensitas *pixel* dalam  $w \times w$  pusat *window* sekeliling *pixel*  $(x,y)$  dalam persamaan (2).

$$t(x,y) = m(x,y) \left[ 1 + k \left( \frac{s(x,y)}{R} - 1 \right) \right] \quad (2)$$

dimana  $R$  adalah nilai maksimum dari standar deviasi ( $R = 128$  untuk dokumen *grayscale*) dan  $k$  adalah *parameter* nilai positif pada *range*  $[0.2, 0.5]$  dalam [2].

*Local mean*  $m(x,y)$  dan standar deviasi  $s(x,y)$  nilai *threshold* menurut kontras pada *pixel local* tetangganya.

Pada konsep *integral image*  $I$  pada input  $g$  gambar yang didefinisikan gambar dengan posisi intensitas *pixel* adalah sama dengan jumlah semua intensitas *pixel* diatas dan disamping posisi pada gambar aslinya. *Formula* intensitas posisi  $(x,y)$  dalam persamaan (3).

$$I_t(x, y) = \sum_{i=0}^x \sum_{j=0}^y g(i, j), \quad (3)$$

dimana  $g$  adalah *input* gambar,  $x$  dan  $y$  adalah posisi dan  $I_t$  adalah intensitas.

*Integral image* pada *grayscale* sangat efektif dihitung *single pass*, setelah *integral image*, *local image* pada beberapa ukuran *window* bisa dihitung secara sederhana dengan 2 kondisi dan 1 operasi *subtration* menghasilkan jumlah semua *pixel* pada *windows* menggunakan persamaan (4).

$$m_t(x, y) = \frac{I(x+\frac{w}{2}, y+\frac{w}{2}) + I(x-\frac{w}{2}, y-\frac{w}{2}) - I(x+\frac{w}{2}, y-\frac{w}{2}) - I(x-\frac{w}{2}, y+\frac{w}{2})}{w^2}, \quad (4)$$

dimana  $m_t$  adalah *local mean*,  $I$  adalah intensitas,  $w$  adalah ukuran *local window*.

Dan *local* variannya dalam persamaan (5).

$$s_t^2(x, y) = \frac{1}{w^2} \sum_{i=x-\frac{w}{2}}^{x+\frac{w}{2}} \sum_{j=y-\frac{w}{2}}^{y+\frac{w}{2}} g^2(i, j) - m_t^2(x, y), \quad (5)$$

dimana  $S_t$  adalah *local* varian,  $w$  adalah ukuran *local windows* dan  $m_t$  adalah *local mean* pada posisi  $x$  dan  $y$ .

Pada *histogram* 3 warna nilai  $t$  disubstitusi dengan *red*( $r$ ), *green*( $g$ ) dan *blue*( $b$ ) pada persamaan 3, 4 dan 5.

## 2.2. Insialisasi Region

Setelah mendapatkan *histogram* dari komponen merah, hijau dan biru pada algoritma modifikasi *sauvola*, insialisasi dominasi puncak pada setiap komponen *histogram* yaitu  $x$ ,  $y$  dan  $z$ .  $P_r = (i_1, i_2, \dots, i_x)$ ,  $P_g = (i_1, i_2, \dots, i_y)$  dan  $P_b = (i_1, i_2, \dots, i_z)$  adalah dominasi puncak pada setiap komponen yang mana nantinya ditandai sebagai keragaman *Region*. Untuk melakukan itu dibutuhkan algoritma *region* sebagai berikut: langkah pertama yaitu bentuk semua kemungkinan *cluster centroid*. Yang kedua, tandai setiap *pixel* yang terdekat dengan *cluster centroid* dan bentuk *set pixel* pada setiap *cluster* dengan menandai *pixel* yang berhubungan dengan *cluster centroid*. Berikutnya eliminasi semua *cluster centroid* yang mempunyai jumlah *pixel* yang ditandai kurang dari *threshold*. Untuk mengurangi jumlah inisial *cluster centroid* nilai dari *threshold* diset  $0.006N - 0.008N$  didapat dari [2], dimana  $N$  adalah jumlah *pixel* dalam gambar. Kemudian langkah keempat adalah menandai lagi setiap *pixel* gambar yang berdekatan dengan *cluster centroid*. Kemudian langkah terakhir yaitu meng-*update* setiap *cluster centroid*  $c_i$  dengan mode *pixel set*  $X_i$  masing-masing.

## 2.3. Merging

Algoritma *merging* dibutuhkan untuk menggabungkan *region* pada warna yang sama. *Tools* yang digunakan untuk mengukur kesamaan warna digunakan *Euclidean distance* yang mana mengukur perbedaan warna antara 2 *region uniform*. Bila  $c = (c_1, c_2, \dots, c_m)$  adalah *cluster centroid* dan  $m$  adalah jumlah *cluster centroid*.

Algoritma *Merging* yaitu: Langkah pertama, pilih *threshold* maksimum pada *Euclidean distance*,  $dc$  pada nilai *integer* positif. Langkah kedua hitung *distance*,  $D$  untuk 2 keluaran pada  $M$  *cluster centroid*.

$$D(c_j, c_k) = \sqrt{(R_j - R_k)^2 + (G_j - G_k)^2 + (B_j - B_k)^2}, \quad (6)$$

dimana  $1 \leq j \leq M$  dan  $1 \leq k \leq M$ ,  $R_j$ ,  $G_j$  dan  $B_j$  adalah nilai komponen *Red*, *Green* dan *Blue* pada  $j$  *cluster centroid* dan juga  $R_k$ ,  $G_k$  dan  $B_k$  adalah nilai komponen dari  $k$  *cluster centroid*.

Langkah ketiga yaitu mencari jarak minimum antar 2 *cluster centroid* berdekatan. Gabungkan *cluster* berdekatan dalam bentuk *cluster centroid* yang baru jika jarak minimum antara *cluster centroid* kurang dari  $dc$ . Jika tidak berhenti proses *merging*. Langkah keempat adalah memperbaharui *pixel set* dengan menandai pada *cluster centroid* yang baru. Kemudian langkah kelima adalah me-*refresh cluster centroid* yang baru. Setelah itu, kurangi jumlah *cluster centroids*  $M$  menjadi  $M-1$  dan ulangi langkah 2 sampai 6 sampai tidak ada jarak minimum antara 2 *cluster centroid* yang berdekatan yang kurang dari  $dc$ .

#### 2.4. Fuzzy C Means

Algoritma FCM adalah sama dengan teknik *hill-climbing*, ini digunakan untuk teknik *clustering* untuk segmentasi gambar. Pada FCM setiap *pixel* mempunyai derajat keanggotaan pada masing-masing *cluster centroid*. Derajat keanggotaan mempunyai *range* nilai  $[0,1]$  dan indikasi kuat pada asosiasi antar *pixel* dan bagian dari *cluster centroid*.

Algoritma FCM bertujuan membagi setiap *pixel* menjadi koleksi dari  $M$  *fuzzy cluster centroid* dengan memberikan beberapa kriteria.  $N$  adalah jumlah *pixel* pada gambar dan  $m$  adalah eksponensial derajat keanggotaan. Fungsi objektif dari FCM dalam persamaan (7).

$$W_m(U, C) = \sum_{i=1}^N \sum_{j=1}^M u_{ji}^m d_{ji}^2, \quad (7)$$

dimana  $U_{ji}$  adalah derajat keanggotaan  $i$  *pixel* ke  $j$  *cluster centroid*,  $d_{ji}$  adalah jarak antara  $i$  *pixel* dengan  $j$  *cluster centroid*.  $U_i = (U_{1i}, U_{2i}, \dots, U_{mi})$  adalah derajat keanggotaan  $i$  *pixel* diasosiasi dengan setiap *cluster centroid*,  $x_i$  adalah  $i$  *pixel* pada gambar dan  $c_j$  adalah  $j$  *cluster centroid*.  $U = (U_1, U_2, \dots, U_N)$  adalah matrik derajat keanggotaan dan  $C = (c_1, c_2, \dots, c_M)$  adalah *cluster centroid*.

Derajat kekompakan dan keseragaman *cluster centroid* sangat tergantung pada fungsi objektif FCM. Umumnya semakin kecil fungsi FCM mengindikasikan kekompakan dan keseragaman *cluster centroid*.

FCM digunakan untuk meningkatkan kekompakan pada *cluster* yang diperoleh dari modul *Sauvola* modifikasi. Algoritmanya sebagai berikut: langkah pertama adalah memilih iterasi akhir *thresholding*.  $\epsilon$  adalah jumlah positif terkecil pada *range*  $[0,1]$  dan jumlah iterasi  $q$  ke 0. Langkah kedua yaitu menghitung  $U^{(q)}$  menurut  $C^{(q)}$  dengan formula persamaan (8).

$$u_{ji} = \frac{1}{\sum_{k=1}^M \left( \frac{d_{ji}}{d_{ki}} \right)^{2/(m-1)}}, \quad (8)$$

dimana  $1 \leq j \leq M$  dan  $1 \leq i \leq N$ , jika  $d_{ji} = 0$  kemudian  $u_{ji} = 1$  dan pilih derajat keanggotaan lain pada *pixel* ke 0.

Langkah ketiga, hitung  $C^{(q+1)}$  berdasarkan  $U^{(q)}$  pada persamaan (9).

$$c_j = \frac{\sum_{i=1}^N u_{ji}^m x_i}{\sum_{i=1}^N u_{ji}^m}, \quad (9)$$

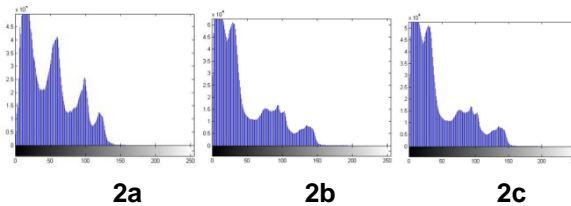
dimana  $1 \leq j \leq m$ .

Langkah keempat, perbaharui  $U^{(q+1)}$  berdasarkan  $C^{(q+1)}$  berdasarkan persamaan (8). Kemudian bandingkan  $U^{(q+1)}$  dengan  $U^{(q)}$ , jika  $\|U^{(q+1)} - U^{(q)}\| \leq \epsilon$  maka berhenti iterasi. Lainnya jika  $q = q + 1$  dan ulangi langkah 2 sampai langkah 4 sampai  $\|U^{(q+1)} - U^{(q)}\| > \epsilon$ .

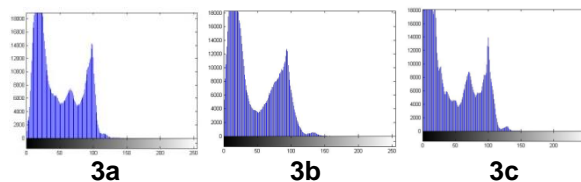
### 3. Hasil dan Pembahasan

Menggunakan algoritma SMFCM. Gambar *sample house* diperlihatkan dalam Gambar 1 ini kemudian dicari komponen histogramnya yaitu *red*, *green* dan *blue*. Gambar 2 memperlihatkan histogram komponen *red*, *green* dan *blue* yang didapat dari gambar asli *sample*. Setelah didapat *histogram*-nya, gambar *sampel* ini dilakukan komputasi menggunakan metode *local window* modifikasi *Sauvola* dalam persamaan (1, 2, 3, 4 dan 5) untuk mengurangi jumlah *peak* dan *valley* dalam histogram *red*, *green* dan *blue* dari gambar *sample house*. Gambar 3 memperlihatkan hasil *histogram* komponen *red*, *green* dan *blue* menggunakan *local window* pada algoritma modifikasi *Sauvola* yang mana jumlah *peak* dan *valley* telah berkurang dibandingkan dengan Gambar 2. Setelah di dapatkan masing-masing histogram pada warna *red*, *green* dan *blue* yang memiliki *local mean* dan *local variance* kemudian dilakukan inisialisasi *cluster centroid* menggunakan persamaan (6). Pada implementasi algoritma *Fuzzy C-Mean* menggunakan persamaan (7, 8 dan 9) didapat jumlah *cluster centroid* sebanyak 4.

Algoritma ini diuji pada 200 gambar warna yang didapat dari gambar umum segmentasi. Pada *paper* ini diambil 20 gambar untuk menampilkan kemampuan dari algoritma SMFCM, 5 buah gambar umum ditampilkan dalam ukuran 256x256 dan 15 gambar lainnya sebagai data pendukung berupa gambar sintetis. Pada *study literatur*, nilai *dc* adalah 28 didapat dalam [2].



Gambar 1. Gambar Sampel House      Gambar 2. Histogram 3 komponen (a) red (b) green (c) blue



Gambar 3. Histogram 3 Komponen RGB pada Gambar Sampel setelah dilakukan *local window* pada algoritma modifikasi *Sauvola* (a) red (b) green (c) blue

#### 3.1 Perbandingan Jumlah *Peak* dan *Valley* Algoritma SMFCM dengan HTFCM

Pada bagian ini membahas jumlah *peak* dan *valley* pada algoritma SMFCM dibandingkan dengan algoritma HTFCM dalam proses segmentasi. Tabel 1 memperlihatkan jumlah *peak* dan *valley* pada proses segmentasi pada beberapa gambar menggunakan algoritma SMFCM dan HTFCM, yang mana jumlah *peak* dan *valley* metode SMFCM lebih sedikit dibandingkan HTFCM.

#### 3.2 Evaluasi Hasil Segmentasi

Pada bagian ini, membahas hasil dari segmentasi SMFCM yang dievaluasi adalah jumlah *region* dan *error rate* pada masing-masing gambar dengan memisahkan antara *foreground* dan *background*. Gambar 4 memperlihatkan perbandingan gambar *sampel* asli dengan gambar hasil segmentasi dengan SMFCM. Hasil gambar segmentasi dengan SMFCM menghasilkan

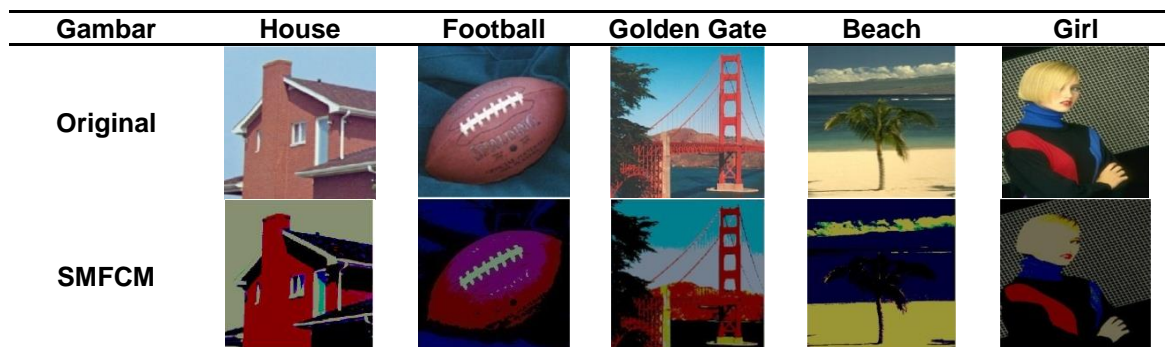
jumlah *region* lebih sedikit dibandingkan dengan hasil segmentasi dengan menggunakan HTFCM. Jumlah *region* dihitung didapat dalam persamaan (9), hasil perhitungan diperlihatkan dalam Tabel 2. Nilai *region* yang lebih sedikit menunjukkan bahwa kelompok warna lebih homogen. Dalam pengujian terhadap gambar sintetis sebanyak 15 warna gambar, pengujian mengevaluasi jumlah *region* dan *error rate* seperti diperlihatkan dalam Tabel 3.

**Tabel 1. Perbandingan Jumlah *Peak* dan *Valley* pada HTFCM dan SMFCM**

Gambar	Algoritma			
	HTFCM		SMFCM	
	<i>Peak</i>	<i>Valley</i>	<i>Peak</i>	<i>Valley</i>
House	9	9	2	2
Football	12	12	2	2
Golden Gate	11	11	3	3
Gate				
Beach	8	8	2	2
Girl	9	9	2	2

**Tabel 2. Jumlah *Region* yang Diproduksi pada Algoritma HTFCM dan SMFCM**

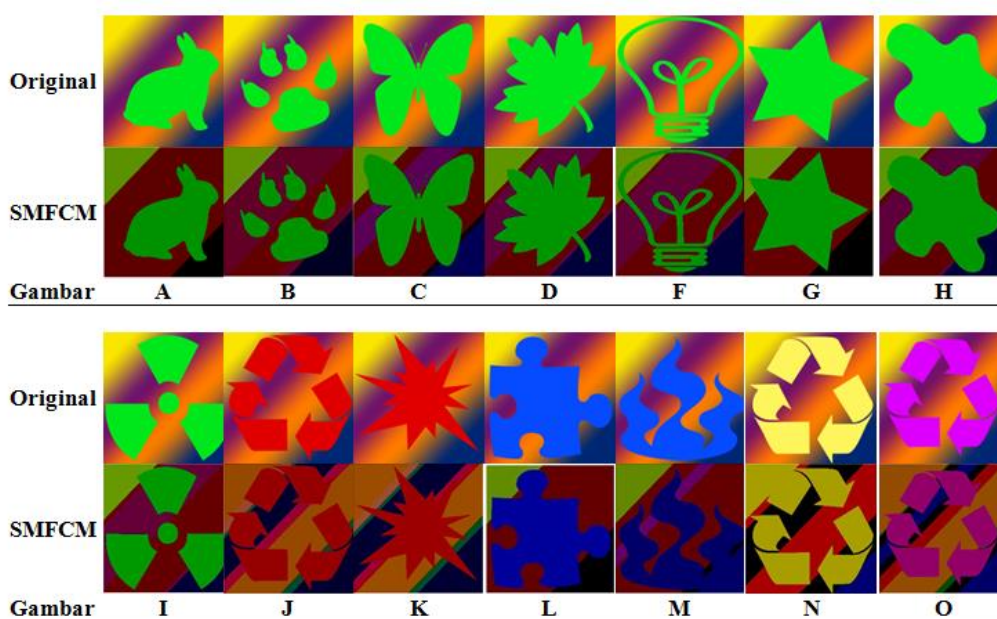
Gambar	Jumlah Region	
	HTFCM	SMFCM
House	7	4
Football	7	4
Golden Gate	11	5
Gate		
Beach	8	7
Girl	9	5



**Gambar 4. Perbandingan gambar original dengan gambar hasil dari metode SMFCM**

**Tabel 3. Jumlah *Cluster* Gambar Sintetis dan *Error Rate***

Gambar	Jumlah Region (M) Original	Jumlah Region (M) Segmentasi	Error Rate
A	7	3	0
B	6	2	0
C	6	3	0
D	7	5	0
E	6	5	0
F	6	4	0
G	6	5	0,1
H	6	4	0
I	6	6	0,5
J	5	6	0,8
K	5	4	0,1
L	5	4	0,1
M	5	7	0,8
N	6	8	0,8



Gambar 5. Hasil Segmentasi pada Gambar Sintetis

#### 4. Pembahasan

Berdasarkan hasil implementasi yang diperlihatkan dalam Gambar 2. dan Gambar 3. bahwa terjadi pengurangan *peak* dan *valley* pada masing-masing histogram *red*, *green* dan *blue*. Pengurangan *peak* dan *valley* pada gambar sampel *House* sebesar 25% jika dihitung berdasarkan perbandingan *peak* dan *valley* pada masing-masing histogram.

Penyebab berkurangnya *peak* dan *valley* ini diakibatkan segmentasi menggunakan algoritma Modifikasi *Sauvola* dalam persamaan (4 dan 5). Gambar sampel yang sudah disegmentasi dirubah dalam bentuk *grayscale* untuk didapatkan histogramnya, kemudian dibandingkan antara gambar sampel asli dan gambar sampel segmentasi dalam bentuk histogram, maka gambar sampel original yang telah di *grayscale* menghasilkan 8 *peak* dan 8 *valley* pada histogramnya, sedangkan gambar sampel hasil segmentasi yang dirubah ke dalam *grayscale* menghasilkan 2 *peak* dan 2 *valley* pada histogramnya. Pengurangan *peak* dan *valley* antara gambar sampel original dan gambar segmentasinya sebesar 75%. Perbandingan *histogram* 3 komponen *red*, *green* dan *blue* antara gambar sampel dan gambar segmentasi mendekati dari bentuk multi modal ke uni modal.

Begitu juga jika dirubah dalam *grayscale* bentuk histogram gambar sampel original adalah multi modal, sedangkan gambar segmentasi adalah uni modal. Jadi *histogram* antar warna gambar dalam 3 komponen *red*, *green* dan *blue* mempunyai bentuk yang sama dalam bentuk *grayscale* baik gambar sampel original maupun gambar hasil segmentasi. Penyebab *histogram* hasil segmentasi berbentuk uni modal karena persamaan (2) *local window* dalam algoritma Modifikasi *Sauvola*.

Perbandingan antara *peak* dan *valley* pada gambar original sampel dan hasil segmentasi dalam Tabel 1 didapat pengurangan *peak* dan *valley* sebesar 0,25 atau 25%. Jadi berdasarkan hasil ini, metode SMFCM mampu mengurangi jumlah *peak* dan *valley* gambar *House* dalam histogram 3 komponen yaitu *red*, *green* dan *blue* sebesar 25% sehingga gambar lebih homogen dalam segmentasi. Pegujian SMFCM pada gambar *House*, *Football*, *Golden*, *Gate*, *Beach* dan *Girl* dengan mengevaluasi *peak* dan *valley* didapat pengurangan jumlah *peak* dan *valley* antara gambar original dan gambar hasil segmentasi sebesar 25% diperlihatkan dalam Tabel 2.

Data sintetis dibuat secara manual dengan menggunakan aplikasi *adobe photoshop CS2* dengan pewarnaan antara *background* dan *foreground* mendekati sama warna degradasinya. Hasil pengujian terhadap 15 gambar sintetis dihasilkan hampir sama dalam pengurangan jumlah *peak* dan *valley* sebesar 25% pada masing-masing histogramnya. Jadi Metode SMFCM mampu mengurangi jumlah *peak* dan *valley* yang menjadi permasalahan pada metode HTFCM.

Pada pengujian SMFCM pada gambar *House, Football, Golden, Gate, Beach* dan *Girl* yang diperlihatkan dalam Gambar 4 didapat jumlah region yang berkurang dibandingkan dengan algoritma HTFCM. Jumlah pengurangan region diperlihatkan dalam Tabel 2. Pengurangan region atau *cluster* sebanyak 54% sehingga gambar segmentasi lebih homogen. Pengurangan region ini lebih banyak disebabkan dari algoritma *Fuzzy C-Mean* dalam mengurangi jumlah *cluster centroid* atau region sesuai persamaan (9).

Untuk mengetahui *error rate* pada SMFCM dilakukan pengujian menggunakan data sintetis. Pada hasil pengujian dengan SMFCM menggunakan data sintetis sebanyak 15 gambar warna didapat hasil jumlah region dan *error rate* seperti dalam Tabel 3. Dalam Tabel 3 hanya gambar tertentu yang tidak terjadi pengurangan, akan tetapi terjadi penambahan region, hal ini disebabkan karena degradasi warna yang hampir sama antara *foreground* dan *background*. Sedangkan untuk *error rate* lebih dari 0,1 dalam Tabel 3 terjadi pada gambar sintetis yang hasil segmentasinya terjadi penambahan jumlah region pada hasil segmentasinya. *Error rate* didapat dari pemisahan antara *background* dan *foreground* menggunakan persamaan (1 dan 2) menggunakan algoritma *Savola Thresholding*. Dalam Tabel 3 didapat rata-rata *error rate* untuk 15 gambar sintetis adalah 21%. Hal ini disebabkan memiliki derajat warna yang sama (derajat kemerahan, derajat kehijauan, derajat kebiruan) antara *foreground* dan *background*. Untuk mengurangi jumlah *error rate* dalam segmentasi warna antara *foreground* dan *background* dapat digunakan pengelompokan metode *cluster* lain.

## 5. Kesimpulan

Hasil penelitian menunjukkan bahwa metode SMFCM berhasil mengurangi jumlah *peak* dan *valley* yang terdapat pada metode HTFCM dengan pengurangan sebesar 25%. Pengurangan *peak* dan *valley* menyebabkan gambar warna menjadi lebih homogen sehingga kurang baik dalam membedakan *background* dan *foreground* yang memiliki warna sama. Metode SMFCM memiliki *error rate* sebesar 21%.

## Daftar Pustaka

- [1] M. Mirmehdi, M. Petrou, Segmentation of Color Textures, IEEE Trans. Pattern Anal. Mach. Intell. 2000; 22(2): 142-159.
- [2] Khang Siang Tan, Nor Ashidi Mat Isa, Color Image Segmentation Using Histogram Thresholding Fuzzy C-Means Hybrid approach, Pattern Recognition. 2011; 44: 1-15.
- [3] Faisal Shafait, dkk, Efficient Implementation of Local Adaptive Thresholding Techniques Using Integral Images, project IpeT (01 IW D03), German Federal Ministry of Education and Research.
- [4] J.Sauvola, dkk, Adaptive Document Image Binarization, Pattern Recognition. 2000; 33(2): 255-236.
- [5] Enno Litmann, dkk, Adaptive Color Segmentation – A Comparison of Neural and Statistical Methods, IEEE Trans. On Neural Network. 1997; 8(1).
- [6] X.L. Xie, G.A. Beni, Validity Measure for Fuzzy Clustering, IEEE Trans. Pattern Anal. Mach. Intell. 1991; 13(4): 841-847.
- [7] J.C. Bezdek, Cluster Validity with Fuzzy Set, Cybernet syst. 1974; 3(3): 58-73