

Peningkatan Ekstraksi Fitur Berbasis *Scale Invariant Feature Transform* Menggunakan Metode *Multiscale Retinex* Untuk Meningkatkan Jumlah *Keypoint*

Dwi Prasetyo¹, Sunaryono², Sulistiyasni³

^{1,2,3}STMIK Widya Utama

¹dwiprasetyo919@gmail.com, ²aryo.indonesia@gmail.com, ³sulistiyasnisulis123@gmail.com

Abstract--The purpose of this research is to perform the improvement of feature extraction based on SIFT algorithm with MSR preprocessing step to increase the number of keypoint. Often in recognizing the SIFT algorithm pattern is affected by image quality. The better the recognizable image, the more number of keypoints it can get. For that need to do preprocessing stages so that the accuracy produced SIFT higher. In this study the researchers tried to combine SIFT algorithm with Multi-Scale Retinex (MSR) method. The MSR method is used because it has advantages in improving image quality in images that have less illumination (darkness). From the test results using MSE and PSNR the proposed method (MSR) is better than the previous methods. The average test result of number of keypoint without preprocessing is 1083.523, Contrast Stretching is 1093.797, CLAHE is 1105.891 and MSR is 1399.162. From this test it can be concluded that the SIFT combination with the preprocessing stages of Multi-Scale Retinex produces a good number of keypoints compared with other methods and without preprocessing stages.

Keywords: SIFT, Preprocessing, Number of Keypoint, MSR

1. PENDAHULUAN

SIFT merupakan sebuah algoritma dalam *computer vision* untuk mendeteksi dan mendeskripsikan fitur-fitur lokal dari citra. Beberapa penelitian yang menggunakan algoritma SIFT diantaranya adalah penelitian yang dilakukan oleh Yu-Yao Wang, Zheng-Ming Li, Long Wang dan Min Wang [1] menggunakan algoritma SIFT untuk pengenalan wajah. Dataset yang digunakan ada data image wajah dari FERET database. Dari hasil penelitian didapatkan kesimpulan bahwa algoritma SIFT mampu mengenali wajah dengan akurasi diantara 64,3 % - 76,7 %.

Penelitian kedua penelitian tentang SIFT yang dilakukan oleh Mr. Amit Kr.Gautam dan Ms. Twisha [2] dengan melakukan pengenalan wajah / *face recognition*. Penelitian ini menggunakan berbagai metode diantaranya metode PCA, ICA, FISHER, 2D PCA, SIFT dan metode yang diusulkan yaitu SIFT dan SVM. Dari hasil penelitian algoritma SIFT menghasilkan hasil *recognition rate* sebesar 96,3 % jauh lebih baik dibanding beberapa metode yang lain.

Dari penelitian 1 dan 2 algoritma SIFT cukup baik dalam melakukan pengenalan pola namun dalam

mengenali pola metode SIFT dipengaruhi oleh beberapa hal. Salah satunya adalah masalah kualitas yang digunakan. Semakin buruk citra yang diolah semakin sulit citra itu untuk dikenali dikarenakan sedikitnya jumlah keypoint yang dihasilkan disebabkan oleh pencahayaan yang kurang sehingga menyebabkan rendahnya akurasi. Untuk mengatasi masalah tersebut perlu dilakukan proses preprocessing terlebih dahulu pada citra yang mengalami penurunan mutu khususnya pada citra yang kurang akan pencahayaan dengan tujuan memperbaiki kualitas citra.

Penelitian-penelitian sebelumnya dalam penggunaan tahapan preprocessing sebelum melakukan proses pengenalan pola pada algoritma SIFT, diantaranya penelitian yang dilakukan oleh Palma Olvera R. D., Martinez Zeron E., Pedraza Ortega J.C., Ramos Arreguin J.M. dan Gorrostieta Hurtado E [3] dengan menggunakan metode *Contrast Limited Adaptive Histogram Equalization* (CLAHE) sebagai proses perbaikan kualitas citra. Dari hasil penelitian metode SIFT dengan tahapan preprocessing menunjukkan keluaran yang lebih baik daripada tidak dilakukan tahapan preprocessing menggunakan CLAHE dengan pengujian menunjukkan bahwa pattern 1 tanpa preprocessing melakukan fitur ekstraksi sebanyak 115 dengan waktu pemrosesan sebanyak 0.867 seg dan dengan preprocessing (CLAHE) dapat melakukan ekstraksi fitur sebanyak 2099 dengan waktu pemrosesan lebih baik yaitu sebesar 3.409 seg serta pada kasus pattern 2 tanpa preprocessing dengan fitur ekstraksi sebanyak 142 dengan lama waktu pemrosesan 0.823 dan dengan preprocessing (CLAHE) dapat melakukan ekstraksi fitur sebesar 3786 dengan lama waktu pemrosesan sebesar 3.439 seg. Penelitian selanjutnya dilakukan oleh Pulung Nurtantio Andono, I Ketut Eddy Purnama dan Mochamad Hariadi [4] mengusulkan menggunakan metode CS, CLAHE dengan *Rayleigh Scattering*. Dari hasil penelitian menunjukkan bahwa setelah menerapkan metode peningkatan citra dengan CLAHE mendapatkan hasil yang lebih baik sampai 41% dibandingkan dengan metode Contrast Stretching (CS). Dari penelitian [3] dan [4] menunjukkan adanya peningkatan hasil dari SIFT setelah dilakukan proses preprocessing.

Penggunaan algoritma SIFT dengan tahapan preprocessing menggunakan metode CLAHE menghasilkan peningkatan dibanding tanpa melakukan tahapan preprocessing, namun hasil peningkatan tidak begitu signifikan untuk kasus citra yang gelap atau kurang pencahayaan. Muncul beberapa metode yang berfokus pada peningkatan citra pada citra gelap atau kurang pencahayaan. Salah satu metode tersebut adalah Multi-Scale Retinex (MSR).

Menurut penelitian oleh In-Su Jang, Tae-Hyoung Lee, Ho-Gun Ha dan Yeong-Ho Ha [5] penggunaan metode Multi-Scale Retinex (MSR) dapat meningkatkan kualitas citra pada citra yang kurang akan pencahayaan. Hasil dari penelitian ini menunjukkan hasil yang cukup signifikan dimana MSR memiliki z-skor yang tinggi dibanding dengan metode sebelumnya yaitu jobson's method dan Rahman's method. Penelitian berikutnya oleh Zahid Mahmood, Tauseef Ali, Shahid Khattak, Mudassar Aslam dan Huzaifa Mehmood [6] melakukan penelitian dengan menggunakan MSR, HE, Histogram Specification dan Gamma Correction. Pada penelitian ini waktu rata-rata untuk melakukan proses perbaikan citra MSR lebih singkat atau efisien dibanding dengan HE yaitu 2,12 detik dibanding 2,53 detik. Penelitian oleh Kozue Kawasaki dan Akira Taguchi [7] melakukan penelitian untuk perbaikan kualitas citra dengan menggunakan Single-Scale Retinex (SSR) dan MSR. Dari hasil penelitian menunjukkan MSR jauh lebih baik dibanding dengan SSR dengan biaya komputasi rendah dan perbaikan kualitas citra yang dihasilkan dengan hasil index Q sebesar 7.270 untuk SSR dan 7.264 Untuk MSR serta nilai Entropy 7.515 berbanding 7.506.

Prinsip tujuan peningkatan kualitas citra adalah menghasilkan gambar yang lebih sesuai dari gambar aslinya untuk diterapkan pada aplikasi yang lebih khusus [8]. Setelah dilakukan proses perbaikan citra atau preprocessing dengan meminimalkan penurunan mutu kualitas citra sehingga menghasilkan citra baru, proses berikutnya dilakukan proses pengenalan pola atau pencocokan gambar. Dalam proses pencocokan gambar, fitur yang diekstraksi digunakan untuk mendeteksi apakah gambar serupa atau tidak dan teknik yang digunakan harus efisien dalam mencari atau mengenali kesamaan dalam jangka waktu yang lebih rendah.

Berdasarkan penelitian-penelitian yang disebutkan diatas, dapat disimpulkan bahwa algoritma SIFT memiliki kelebihan dalam mendeteksi dan mendeskripsikan fitur-fitur lokal itu dapat dipergunakan dalam pengenalan pola citra, namun demikian jumlah keypoint SIFT cenderung menurun untuk citra kurang akan pencahayaan. Untuk itu peneliti akan melakukan penelitian pengolahan pola citra menggunakan SIFT dengan melakukan teknik preprocessing terlebih dahulu menggunakan MSR. Pengolahan citra SIFT dan Teknik preprocessing CS dan CLAHE sebagai penelitian

sebelumnya atau *current method* dan SIFT dengan MSR sebagai metode yang diusulkan atau *proposed method*.

2. METODE PENELITIAN

2.1 Tahapan dan metode yang diusulkan

Pada penelitian ini, citra dilakukan proses preprocessing untuk menghasilkan citra yang memiliki kualitas baik dibanding citra sebelumnya. Preprocessing dilakukan dengan menggunakan metode CS dan CLAHE sebagai *current method* atau penelitian terdahulu dan metode MSR sebagai *proposed method* atau metode yang diusulkan. Hasil dari preprocessing adalah citra baru dan citra orisinal tetap digunakan pada tahapan berikutnya.

Setelah melalui tahapan preprocessing dilakukan tahapan berikutnya yaitu pencocokan citra atau pengenalan pola dengan menggunakan algoritma SIFT. Algoritma SIFT pertama dilakukan pada citra yang tanpa mengalami preprocessing atau citra orisinal, berikutnya SIFT akan dikombinasikan dengan masing-masing metode atau teknik preprocessing yang sudah berhasil dilakukan.

Kombinasi itu adalah algoritma SIFT dengan teknik preprocessing CS, SIFT dengan CLAHE dan SIFT dengan *proposed method* yaitu SIFT kombinasi dengan MSR. SIFT berhasil dilakukan dengan melakukan uji coba dengan kombinasi metode yang ada, tahap terakhir adalah melakukan analisa dan evaluasi sejauh mana metode yang diusulkan (SIFT kombinasi MSR) dibanding dengan metode terdahulu. Tahapan Analisa dan evaluasi dilakukan dengan tujuan melakukan perbandingan antar metode dan untuk menarik kesimpulan dari penelitian ini. Untuk evaluasi pada penelitian ini digunakan MSE dan PNSR. Semakin kecil nilai MSE dan semakin besar nilai PNSR maka akan menghasilkan nilai yang baik pada sebuah penelitian.

3.2.1. Proses Preprocessing Menggunakan MSR

Tahapan proses preprocessing menggunakan metode MSR, bertujuan untuk mengolah gambar yang pencahayaanya kurang terang.

3.2.2. Proses Pengenalan Pola Menggunakan SIFT

Pengenalan Pola dengan SIFT dilakukan untuk mencari dan menentukan *keypoint*, kemudian menentukan descriptor dari masing masing *keypoint* tersebut. Descriptor inilah yang akan menjadi dasar dalam pencocokan citra dalam penelitian ini. Dalam penelitian ini menggunakan 4 tahapan yaitu sebagai berikut:

1. Mencari Nilai Ekstrim Pada Skala Ruang (*Scale-Space Extrema Detection*)
2. Menentukan Keypoint (*Keypoint Localization*)
3. Penentuan Orientasi (*Orientation Assignment*)
4. Deskriptor Keypoint (*Keypoint Descriptor*)

3. HASIL PENELITIAN DAN PEMBAHASAN

3.1 Tahapan Preprocessing Implementasi Metode Contrast Stretching

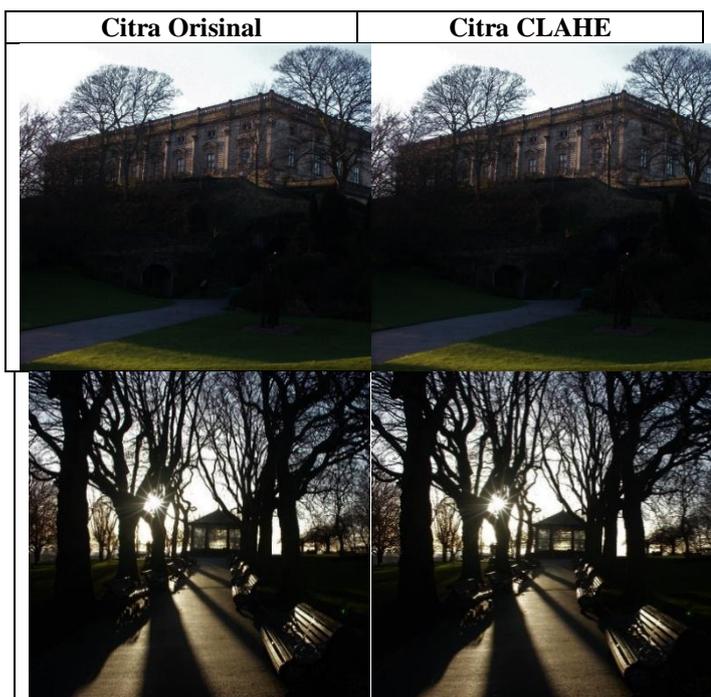
Berikut adalah beberapa contoh hasil dari Contrast Stretching yang ditunjukkan pada Gambar 4.1.



Gambar 4.1 Hasil Citra Contrast Stretching

Implementasi Metode CLAHE

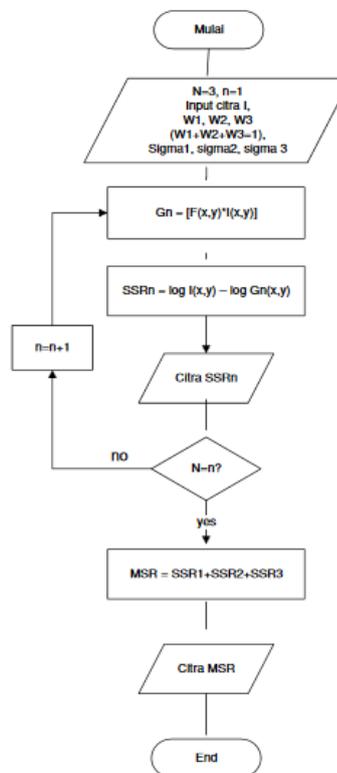
Berikut adalah beberapa contoh hasil dari CLAHE yang ditunjukkan pada Gambar 4.2.



Gambar 4.2 Hasil Citra CLAHE

Implementasi Metode Multiscale Retinex

Tahap selanjutnya yang dilakukan untuk perbaikan citra (*Image Enhancement*) adalah metode Multi-Scale Retinex (MSR). Pada Gambar 4.3 merupakan langkah-langkah proses Multi-Scale Retinex.



Gambar 4.3 Flowchart untuk proses Multiscale Retinex

Tahap awal yang akan dilakukan untuk perbaikan citra menggunakan metode Multi-Scale Retinex (MSR), harus terlebih dahulu melewati tahap SSR, dengan melewati tiga tahap SSR. Seperti telah dibahas pada bab sebelumnya bahwa SSR adalah Retinex versi dinamis untuk meniru sistem kerja neuron pada Sistem persepsi warna manusia. Berbeda dengan Retinex statis, Retinex versi dinamis tidak menggunakan jalur, tetapi menggunakan fungsi sekitar (surround function) untuk meradiasikan intensitas ke piksel sekitarnya, dengan rumus: Metode MSR merupakan pengembangan dari SSR, dengan rumus:

$$SSR_n(x, y) = \log I_n(x, y) - \log G_n(x, y)$$

Berikut contoh pengujian perhitungan manual:

Langkah pertama yang dilakukan adalah mencari nilai Gaussian, yaitu:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Dimana:

σ = sebuah sigma

π = 22/7

e = 2,71828182846

Misalnya $\sigma = 5,5$

$$\begin{bmatrix} \frac{1}{2\pi(5,5)^2} e^{-\frac{-1^2+1^2}{2(5,5)^2}} & \frac{1}{2\pi(5,5)^2} e^{-\frac{0^2+1^2}{2(5,5)^2}} & \frac{1}{2\pi(5,5)^2} e^{-\frac{-1^2+0^2}{2(5,5)^2}} \\ \frac{1}{2\pi(5,5)^2} e^{-\frac{-1^2+0^2}{2(5,5)^2}} & \frac{1}{2\pi(5,5)^2} e^{-\frac{0^2+0^2}{2(5,5)^2}} & \frac{1}{2\pi(5,5)^2} e^{-\frac{1^2+0^2}{2(5,5)^2}} \\ \frac{1}{2\pi(5,5)^2} e^{-\frac{-1^2+-1^2}{2(5,5)^2}} & \frac{1}{2\pi(5,5)^2} e^{-\frac{0^2+-1^2}{2(5,5)^2}} & \frac{1}{2\pi(5,5)^2} e^{-\frac{1^2+-1^2}{2(5,5)^2}} \end{bmatrix}$$

$$\begin{bmatrix} 0.0050902359 & 0.0051750712 & 0.0050902359 \\ 0.0051750712 & 0.0052613204 & 0.0051750712 \\ 0.0050902359 & 0.0051750712 & 0.0050902359 \end{bmatrix}$$

Total: 0.046322549

Total dari seluruh elemen matriks yang dihasilkan harus sama dengan 1, jika tidak memenuhi maka percobaan akan diulang kembali dengan rumus baru yaitu:

$$new\ a = a \times \frac{1}{total}$$

Contoh:

$$New\ a_{(1,1)} = 0.0050902359 \times \frac{1}{0.046322549} = 0.10988678$$

$$New\ a_{(1,2)} = 0.0051750712 \times \frac{1}{0.046322549} = 0.11171819$$

Perhitungan dilakukan pada setiap elemen matriks, sehingga didapatkan hasil sebagai berikut:

$$\begin{bmatrix} 0.10988678 & 0.11171819 & 0.10988678 \\ 0.11171819 & 0.11358011 & 0.11171819 \\ 0.10988678 & 0.11171819 & 0.10988678 \end{bmatrix}$$

Hasil konvolusi yang didapatkan dari perkalian antara pixel citra awal dan citra kernel Gaussian dapat dilihat pada Tabel 4.1, 4.2, dan 4.3.

Tabel 4.1 Perkalian Pixel Citra awal dan Kernel Gaussian

	8	8	1	3	2	
F(x,y)	4	3	5	2	1	G(x,y)
	1	1	5	4	3	

6	7	6	5	8
2	4	6	8	7

0.109887	0.111718	0.109887
0.111718	0.11358	0.111718
0.109887	0.111718	0.109887

Tabel 4.2 Lanjutan Perkalian Pixel Citra awal dan Kernel Gaussian

0	0	0	0	0	0	0
0	8	8	1	3	2	0
0	4	3	5	2	1	0
0	1	1	5	4	3	0
0	6	7	6	5	8	0
0	2	4	6	8	7	0
0	0	0	0	0	0	0

2.578915				
----------	--	--	--	--

Tabel 4.3 Lanjutan Perkalian Pixel Citra awal dan Kernel Gaussian 2

0	0	0	0	0	0	0
0	8	8	1	3	2	0
0	4	3	5	2	1	0
0	1	1	5	4	3	0
0	6	7	6	5	8	0
0	2	4	6	8	7	0
0	0	0	0	0	0	0

2.578915	3.23824			
----------	---------	--	--	--

Hasil konvolusi, didapatkan dengan perhitungan sebagai berikut:

$$(0 \times 0.10988678) + (0 \times 0.11171819) + (0 \times 0.10988678) + (0 \times 0.11171819) + (8 \times 0.11358011) + (8 \times 0.11171819) + (0 \times 0.10988678) + (4 \times 0.11171819) + (3 \times 0.10988678) = 2.5789195$$

$$(0 \times 0.10988678) + (0 \times 0.11171819) + (0 \times 0.10988678) + (8 \times 0.11171819) + (8 \times 0.11358011) + (1 \times 0.11171819) + (4 \times 0.10988678) + (3 \times 0.11171819) + (5 \times 0.10988678) = 3.23824018$$

Seluruh hasil perhitungan konvolusi dapat dilihat pada Tabel 4.4.

Tabel 4.4 Hasil Perhitungan Konvolusi

2.57892	3.23824	2.450505	1.558652	0.893807
2.448765	3.999969	3.554998	2.888251	1.664815
2.441315	4.208694	4.222347	4.327829	2.562284
2.349198	4.232685	5.111907	5.780196	3.903055
2.11355	3.448735	4.011055	4.457983	3.131997

Hasil SSR1

$SSR_n(x, y) = \log l_n(x, y) - \log G_n(x, y)$
Dimana $\log l_n(x, y)$ merupakan pixel citra awal.

Contoh:

- $8 - 2.5789195 = 5.4210805$
- $8 - 3.23824018 = 4.76175982$

Seluruh perhitungan nilai SSR1 dapat dilihat pada Tabel 4.5.

Tabel 4.5 Perhitungan Nilai SSR₁

5.4210805	4.76175982	-1.45050505	1.44134804	1.10619346
1.5512348	-0.9999694	1.44500193	-0.88825127	-0.6648149
-1.441348	-3.2086935	0.77653315	-0.31782889	0.43771574
3.6508017	2.76731508	0.88092571	-0.78019587	4.09694491
-0.113549	0.55126535	1.98894488	3.54201678	3.86800293

Misal $\sigma = 6.5$

$$\begin{bmatrix} \frac{1}{2\pi(6,5)^2} e^{-\frac{-1^2+1^2}{2(6,5)^2}} & \frac{1}{2\pi(6,5)^2} e^{-\frac{0^2+1^2}{2(6,5)^2}} & \frac{1}{2\pi(6,5)^2} e^{-\frac{-1^2+1^2}{2(6,5)^2}} \\ \frac{1}{2\pi(6,5)^2} e^{-\frac{-1^2+0^2}{2(6,5)^2}} & \frac{1}{2\pi(6,5)^2} e^{-\frac{0^2+0^2}{2(6,5)^2}} & \frac{1}{2\pi(6,5)^2} e^{-\frac{1^2+0^2}{2(6,5)^2}} \\ \frac{1}{2\pi(6,5)^2} e^{-\frac{-1^2+-1^2}{2(6,5)^2}} & \frac{1}{2\pi(6,5)^2} e^{-\frac{0^2+1^2}{2(6,5)^2}} & \frac{1}{2\pi(6,5)^2} e^{-\frac{1^2+1^2}{2(6,5)^2}} \end{bmatrix}$$

$$\begin{bmatrix} 0.00368074 & 0.00372456 & 0.00368074 \\ 0.00372456 & 0.0037689 & 0.00372456 \\ 0.00368074 & 0.00372456 & 0.00368074 \end{bmatrix}$$

Total seluruh elemen matriks = 0.0333901, karena tidak sama dengan 1 maka dihitung kembali dengan rumus:

$$new a = a x \frac{1}{total}$$

$$\begin{bmatrix} 0.11023447 & 0.11154683 & 0.11023447 \\ 0.11154683 & 0.11287477 & 0.11154683 \\ 0.11023447 & 0.11154683 & 0.11023447 \end{bmatrix}$$

Hasil konvolusi yang didapatkan dari perkalian antara piksel citra awal dan kernel Gaussian dapat dilihat pada Tabel 4.6

Tabel 4.6 Hasil Perhitungan Konvolusi SSR₁

2.57226353	3.23367035	2.44879629	1.55776528	0.89109344
2.78217127	3.99998429	3.55514049	1.99996878	1.66534387
2.4422346	4.21254792	4.22310912	4.32939093	2.56037439
2.34388927	4.22970208	5.1168117	5.77951525	3.89901425
2.11285913	3.44751516	4.0079052	4.45412368	3.12604502

Hasil SSR₂

Perhitungan nilai SSR2 dapat dilihat pada Tabel 4.7 berikut ini.

Tabel 4.7 Hasil SSR₂

5.42773647	4.76632965	-1.44879629	1.44223472	1.1890656
1.21782873	-0.9999843	1.44485951	0.00003122	-0.665343
-1.4422346	-3.2125479	0.77689088	-0.32939093	0.43962561
3.65611073	2.77029792	0.8831883	-0.77951525	4.10098575
-0.1128591	0.55248484	1.9920948	3.54587632	3.87395498

Misal $\sigma = 7.5$

$$\begin{bmatrix} \frac{1}{2\pi(7,5)^2} e^{-\frac{-1^2+1^2}{2(7,5)^2}} & \frac{1}{2\pi(7,5)^2} e^{-\frac{0^2+1^2}{2(7,5)^2}} & \frac{1}{2\pi(7,5)^2} e^{-\frac{-1^2+1^2}{2(7,5)^2}} \\ \frac{1}{2\pi(7,5)^2} e^{-\frac{-1^2+0^2}{2(7,5)^2}} & \frac{1}{2\pi(7,5)^2} e^{-\frac{0^2+0^2}{2(7,5)^2}} & \frac{1}{2\pi(7,5)^2} e^{-\frac{1^2+0^2}{2(7,5)^2}} \\ \frac{1}{2\pi(7,5)^2} e^{-\frac{-1^2+-1^2}{2(7,5)^2}} & \frac{1}{2\pi(7,5)^2} e^{-\frac{0^2+1^2}{2(7,5)^2}} & \frac{1}{2\pi(7,5)^2} e^{-\frac{1^2+1^2}{2(7,5)^2}} \end{bmatrix}$$

$$\begin{bmatrix} 0.002881575 & 0.002805749 & 0.002881575 \\ 0.002805749 & 0.002830856 & 0.002805749 \\ 0.002881575 & 0.002805749 & 0.002881575 \end{bmatrix}$$

Total seluruh elemen matriks = 0.02558, karena tidak sama dengan satu, maka dihitung kembali dengan rumus:

$$new a = a x \frac{1}{total}$$

$$\begin{bmatrix} 0.11264953 & 0.10968526 & 0.11264953 \\ 0.10968526 & 0.11066677 & 0.10968526 \\ 0.11264953 & 0.10968526 & 0.11264953 \end{bmatrix}$$

Hasil konvolusi yang didapatkan dari perkalian antara pixel citra awal dan kernel Gaussian dapat dilihat pada Tabel 4.8 dibawah ini.

Tabel 4.8 Hasil Konvolusi SSR₂

2.53950587	3.2140305	2.42887858	1.55632379	0.88537364
2.77273597	3.99607793	3.56226419	2.88638675	1.66301123
2.44369993	4.23127106	4.22334002	4.34093658	2.5464554
2.32410087	4.21641023	5.10180361	5.77079074	3.88240742
2.10673285	3.43974034	3.99012966	4.43676226	3.0928792

Hasil SSR₃

Perhitungan nilai SSR3 dapat dilihat pada Tabel 4.9 berikut ini.

Tabel 4.9 Hasil SSR₃

5.46049413	4.78459695	-1.42887858	1.44367621	1.114626
1.22726403	-0.9960779	1.43773581	-0.88638675	-0.663011
-1.4436999	-3.2312711	0.77665998	-0.34093658	0.4535446
3.67589913	2.78358977	0.89819639	-0.77079074	4.1175926
-0.1067328	0.56025966	2.00987034	3.56323774	3.9071208

Tahap selanjutnya yang dilakukan untuk perbaikan citra (Image Enhancement) adalah metode Multiscale Retinex (MSR), dimana metode MSR merupakan pengembangan dari SSR, dengan rumus:

$$R_{msri} = \sum_{n=1}^N Wn Rni$$

R merupakan output dari Multi-Scale Retinex (MSR) yang merupakan jumlah dari output SSR yang masing-masing diberi bobot. N adalah jumlah skala yang digunakan. wn adalah bobot yang diasosiasikan dengan skala ke- n. i Rn merupakan output dari SSR yang diasosiasikan dengan skala ke- n. Simbol i menyatakan channel warna, misalnya Red(R), Green(G), dan Blue(B). Contoh:

Perhitungan manual pada Multiscale Retinex didapatkan dengan menjumlahkan pixel dari SSR1, SSR2 dan SSR3 dan dikalikan dengan nilai bobot yaitu 1. Hasil penjumlahan MSR tersebut dapat dilihat pada Tabel 4.10 berikut ini

Tabel 4.10 Hasil Perhitungan Multiscale Retinex

16.3093111	14.31268642	-4.32817992	4.32725897	3.32972638
3.99632759	-2.99603167	3.66652142	-1.7746068	-1.99317
-4.32728253	-9.65251251	2.33008401	-0.9981564	1.33088592
10.98281156	8.32120277	2.6623104	-2.33050186	12.31552324
-0.33314156	1.66400985	5.99091002	10.65113084	11.64907871

Setelah mendapat hasil Multiscale Retinex, pixel Multiscale Retinex dinormalisasikan, dengan rumus:

$$\text{New Value} = \text{abs} \frac{(\text{value} - \text{min})}{(\text{max} - \text{min})} * 255$$

Dari piksel hasil perhitungan Multiscale Retinex didapat, Nilai max = 16.3093111 Nilai min = -9.65251251 Sesuai dengan rentang warna dari yang minimal sampai dengan nilai maksimal yaitu 0 – 255 maka nilai piksel maksimal 16.3093111 diubah menjadi 255 dan nilai minimal - 9.65251251 diubah menjadi 0.

$$\text{Misalnya} = \text{abs} \frac{(16.3093111 - (-9.65251251))}{((16.3093111 - (-9.65251251)) * 255 = 255$$

Maka, hasil normalisasi dapat dilihat pada Tabel 4.11 berikut ini:

Tabel 4.11 Hasil Normalisasi

255	235	52	137	127
134	65	131	77	75
52	0	118	85	108
203	176	121	72	216
91	111	154	199	209

Gambar 4.4 Citra Orisinal dan Hasil Multiscale Retinex

Pengenalan Pola dengan SIFT

Tahapan berikutnya setelah dilakukan tahapan preprocessing adalah melakukan pengenalan pola atau pencocokan citra menggunakan algoritma SIFT. Pada tahapan ini dilakukan ujicoba SIFT dengan Citra Orisinal dan SIFT dengan citra setelah preprocessing. Ujicoba menggunakan Aplikasi Matlab dan nantinya akan dilakukan Analisa dan Evaluasi menggunakan Mean Square Error (MSE) dan Peak Signal to Noise Ratio (PSNR).

Dalam melakukan pengenalan pola menggunakan SIFT ada beberapa tahapan yaitu mencari nilai ekstrim pada skala ruang, menentukan keypoint, penentuan orientasi dan deskriptor keypoint.

Tabel 4.12 Rata rata Jumlah keypoint yang terdeteksi pada citra

Jumlah Keypoint Terdeteksi			
Orisinal	CS	CLAHE	MSR
1083.523	1093.797	1105.891	1399.162

1.1

3.2 Tahapan Analisa & Evaluasi

Dari hasil penelitian dengan menggunakan 50 data citra diuji dengan algoritma SIFT tanpa preprocessing dan dengan preprocessing didapatkan hasil yang diuraikan pada berikut ini.

Perhitungan dengan Mean Square Error (MSE)

Contoh penghitungan nilai MSE

4	1	2
2	6	4
1	2	6

Citra asli

2	2	4
2	4	3
1	3	4

Citra hasil pengolahan

$$MSE = \frac{1}{MN} \sum_{y=1}^M \sum_{x=1}^N [I(x,y) - I'(x,y)]^2$$

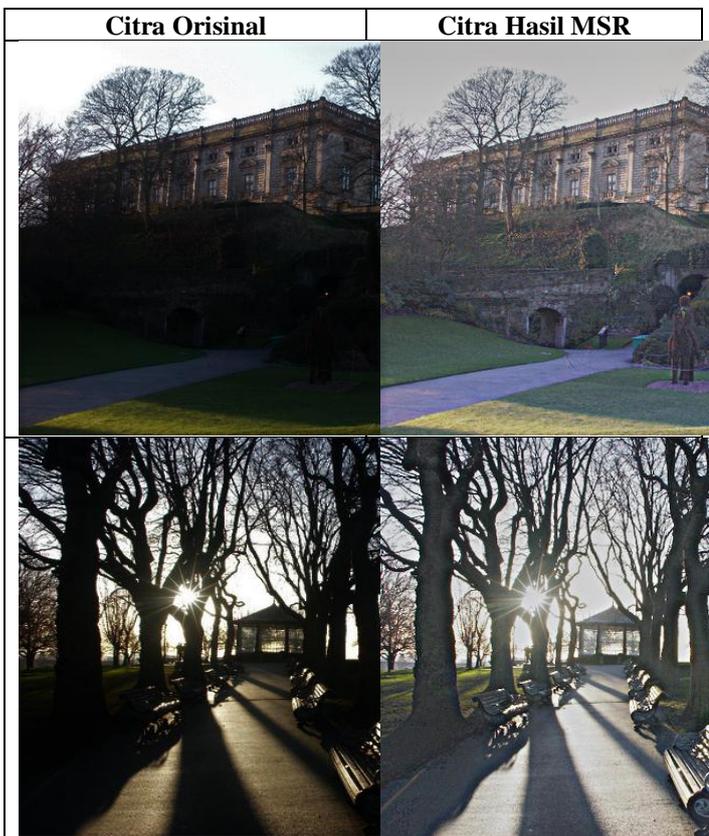
$$= \frac{[(4-2)^2 + (1-2)^2 + (2-4)^2 + (2-2)^2 + (6-4)^2 + (4-3)^2 + (1-1)^2 + (2-3)^2 + (6-4)^2]}{3 \times 3}$$

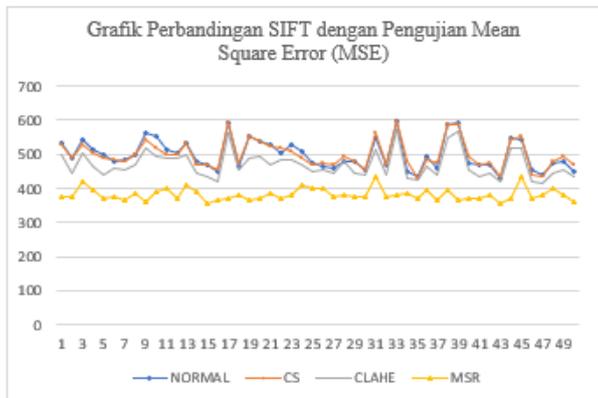
$$= \frac{4 + 1 + 4 + 0 + 1 + 4 + 0 + 1 + 4}{9}$$

$$= \frac{19}{9}$$

$$= 2,11 \text{ dB}$$

Dari contoh perhitungan diatas, didapatkanlah hasil perhitungan dengan menggunakan MSE yang dapat dilihat pada Gambar 4.5





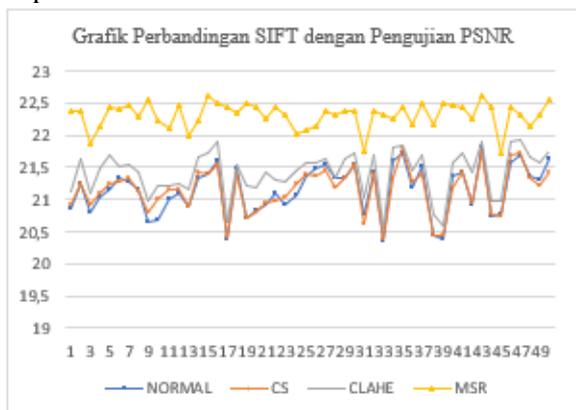
Gambar 4.5 Grafik Hasil Pencocokan citra dengan algoritma SIFT dengan menggunakan MSE

Perhitungan dengan Peak Signal to Noise Ratio (PSNR)

Perhitungan Peak Signal to Noise Ratio didapatkan setelah menghitung nilai dari MSE. Pada bagian sebelumnya nilai MSE adalah sebesar 2,11 dB.

$$\begin{aligned}
 \text{MSE} &= 2,11 \text{ sehingga PSNR adalah:} \\
 &= 10 \log \text{MAX}i^2 - 10 \log \text{MSE} \\
 &= 10 \log 6^2 - 10 \log 2,11 \\
 &= 15,56 - 3,24 \\
 &= 12,31 \text{ dB}
 \end{aligned}$$

Dari contoh perhitungan diatas, didapatkanlah hasil perhitungan dengan menggunakan PSNR yang dapat dilihat pada Gambar 4.6



Gambar 4.6 Hasil Pencocokan citra dengan algoritma SIFT dengan menggunakan PSNR

Perbandingan hasil Rata-Rata MSE dan PNSR

Tabel 4.13 Rata-rata Citra dengan MSE

Citra	Rata-Rata
SIFT + Normal	499.5301
SIFT + CS	499.3311
SIFT + CLAHE	470.0352
SIFT + MSR	381.8123

Tabel 4.14 Rata-rata Citra dengan PSNR

Citra	Rata-Rata
SIFT + Normal	21.1514
SIFT + CS	21.1527

SIFT + CLAHE	21.4189
SIFT + MSR	22.3148

Dari Tabel 4.13 dan Tabel 4.14 ditarik kesimpulan bahwa metode yang diusulkan yaitu kombinasi algoritma SIFT dengan MSR memberikan hasil yang signifikan dibanding dengan metode terdahulu. MSE pada SIFT dengan MSR memberikan hasil paling rendah. Semakin rendah nilai MSE semakin baik hasil yang didapatkan. Untuk PSNR algoritma SIFT dengan MSR memberikan hasil paling tinggi, sehingga semakin tinggi nilai PSNR maka akan semakin baik.

4. KESIMPULAN DAN SARAN

4.1 Kesimpulan

Dari pembahasan dan evaluasi pada bab-bab sebelumnya, dalam perbaikan kualitas pencocokan citra pada citra yang kurang akan pencahayaan diperoleh kesimpulan yaitu:

Metode yang diusulkan yaitu Multi-Scale Retinex (MSR) mampu meningkatkan jumlah keypoint pengenalan pola berbasis Scale Invariant Feature Transform (SIFT) dengan rata-rata keypoint yang terdeteksi MSR sebanyak 1399 Jauh lebih baik dari metode yang lainnya. Untuk melakukan pengujian menggunakan Mean Square Error (MSE) dan Peak Signal to Noise Ratio (PSNR). Adapun hasil pengujian MSE dengan SIFT tanpa Preprocessing sebesar 499,5301, kombinasi SIFT dengan CS sebesar 499,5301, SIFT dengan CLAHE sebesar 470,0352 dan MSR 381,8123. Semakin kecil nilai dari MSE performa yang dihasilkan sebagai bagus dengan kata lain kombinasi SIFT dengan tahapan preprocessing Multi-Scale Retinex menghasilkan keluaran yang baik dibanding tanpa preprocessing dan tahapan preprocessing lainnya. SIFT dengan pengujian PSNR untuk tanpa preprocessing sebesar 21,1514, dengan preprocessing CS sebesar 21.1527, preprocessing CLAHE 21,4189 dan dengan preprocessing MSR sebesar 22,314.

Pengujian PNSR semakin besar nilainya semakin baik performa SIFT yang dihasilkan. Dari pengujian ini dapat disimpulkan kombinasi SIFT dengan tahapan preprocessing Multi-Scale Retinex menghasilkan performa yang baik dibanding dengan metode yang lain dan tanpa tahapan preprocessing. Dari kedua pengujian tersebut metode yang diusulkan mampu menyelesaikan masalah dari penelitian ini.

4.2 Saran

Berdasarkan hasil analisa dan pembahasan pada penelitian ini, ada beberapa saran yang mungkin dapat dijadikan dasar untuk pengembangan penelitian ini yaitu:

- (1) Menggunakan data sampel yang lebih banyak sehingga akurasi yang dihasilkan lebih akurat.
- (2) Dalam melakukan preprocessing, citra hasil Multi-Scale Retinex mengalami kecerahan terlalu terang pada beberapa citra sehingga perlu dilakukan Color Restoration. Teknik ini merupakan pengembangan dari Multi-Scale Retinex yaitu Multi-Scale Retinex with Color Restoration (MSRCR).

DAFTAR PUSTAKA

- [1] Yu-Yao Wang, Zheng-Ming Li, Long Wang and Min Wang "A Scale Invariant Feature Transform Based Method," *Journal of Information Hiding and Multimedia Signal Processing*, Ubiquitous International, Vol.4 No. 2, 2013.
- [2] Mr. Amit Kr.Gautam and Ms. Twisha" Improved Face Recognition Technique using Sift," *IOSR Journal of Electrical and Electronics Engineering (IOSR-JEEE)*, pp. 72-76, 2014.
- [3] Palma Olvera R. D.,Martinez Zeron E.,Pedraza Ortega J.C.,Ramos Arreguin J.M. and Gorrostieta Hurtado E."A Feature Extraction Using SIFT with a Preprocessing by Adding CLAHE Algorithm to Enhance Image Histograms," *International Conference on Mechatronics, Electronics and Automotive Engineering*, pp. 20-25, 2014.
- [4] Pulung Nurtantio Andono, I Ketut Eddy Purnama and Mochamad Hariadi "Underwater Image Enhancement Using Adaptive Filtering For Enhanced SIFT-Based Image Matching," *Journal of Theoretical and Applied Information Technology*, vol. Vol. 51 No.3, pp. 393-399, 2013.
- [5] In-Su Jang, Tae-Hyoung Lee, Ho-Gun Ha and Yeong-Ho Ha "Adaptive Color Enhancement Based on Multi-Scaled Retinex Using Local Contrast of the Input Image," *IEEE*, 2010 K. Struss, "A Chaotic Image Encryption," pp. 1–19, 2009.
- [6] Zahid Mahmood, Tauseef Ali, Shahid Khattak, Mudassar Aslam and Huzaifa Mehmood"Color Image Enhancement Technique Using Multiscale Retinex," *International Conference on Frontiers of Information Technology*, pp. 119-124, 2013. [7] M. Mishra, "High Security Image Steganography with Modified Arnold 's Cat Map," vol. 37, no. 9, pp. 16–20, 2012.
- [7] Kozue Kawasaki and Akira Taguchi "A Multiscale Retinex with Low Computational Cost," *International Symposium on Communications and Information Technologies (ISCIT)*, pp. 787-790, 2013.
- [8] Ricardus Anggi Premunendar, Guruh Fajar Shidik, Catur Supriyanto, Pulung Nurtantio Andono and Mochamad Hariadi "Auto Level Color Correction for Underwater Image Matching Optimization," *IJCSNS International Journal of Computer Science and Network Security*, Vols. 13, No 1, 2013.
- [9] Deepa Raj and Ms Pushpa Mamoria "Comparative analysis of Contrast Enhancement," *International Conference on Green Computing and Internet of Things (ICGCIoT),IEEE*, pp. 27-31, 2015.Palma Olvera R. D., "A Feature Extraction Using SIFT with a Preprocessing by Adding CLAHE Algorithm to Enhance Image Histograms," *International Conference on Mechatronics, Electronics and Automotive Engineering*, pp. 20-25, 2014.
- [10] Balvant Singh, Ravi Shankar Mishra and Puran Gour "Analysis of Contrast Enhancement Techniques For Underwater Image," *International Journal of Computer Technology and Electronics Engineering (IJCTEE)*, vol. 1, no. 2, pp. 190-194, 2011.
- [11] Chang-Hsing Lee, Cheng-Chang Lien and Chin-Chuan Han"Color Image Enhancement Using Multiscale Retinex and Image Fusion Techniques," *World Academy of Science, Engineering and Technology International Journal of Computer, Electrical, Automation, Control and Information Engineering*, Vols. 8 ,No 10, pp. 1791-1797, 2014.
- [12] Dehesa Gonzales Mario,Rosales Silva Alberto J, Gallegos Funes Francisco J, Volodymyr Ponomaryov and Victor Kravchenko "Cromaticity improvement in images with poor lighting using the Multiscale-Retinex MSR algorithm," *International Kharkiv Symposium on Physics and Engineering of Microwaves, Millimeter and Submillimeter Waves (MSMW)*, pp. 1-4, 2016.
- [13] Raman Maini and Himanshu Aggarwal "A Comprehensive Review of Image Enhancement Techniques", *Journal of Computing* Vol. 2, 2010.
- [14] Kohichi Izumino, Yoshikatsu Hoshi and Akira Taguchi"Hue-preserving Multiscale Retinex for Color Images," *In International Symposium on Intelligent Signal Processing and Communication Systems*, p 440-444, 2013.
- [15] David G Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, 2004.
- [16] David G Lowe, "Object Recognition from local Scale-Invariant Features," *In International Conference on Computer Vision, Corfu* 1999.
- [17] Y.Ke and R.Sukthankar, "A More Distinctive Representation for Local Image Descriptors", *Res.Intel*, 2003.
- [18] J. Jogleker and S.S. Gedam, "Image Matching With Sift Features - A Probabilistic Approach," *IAPRS*, vol. XXXVIII, no. September, pp.339-347, 2010.
- [19] [Http://homepages.lboro.ac.uk/~cogs/datasets/ucid/ucid.html](http://homepages.lboro.ac.uk/~cogs/datasets/ucid/ucid.html). [Diakses pada : Senin 10 Oktober 2016].