

Pendeteksian Objek Pada Citra Menggunakan Pencocokan Titik-Titik Fitur Berbasis Algoritma SURF dan MSER

Homa. P. Harahap

Sistem Informasi, Universitas Trilogi
Jl. Kampus Trilogi, No.1, Kalibata, Jakarta Selatan 12760
posniroha@universitas-trilogi.ac.id

Abstrak

Penelitian ini mendeskripsikan proses pendeteksian sebuah objek tertentu dalam sebuah citra dengan menemukan kecocokan antara titik-titik fitur dari masing-masing citra dengan menggunakan algoritma Speeded Up Robust Features (SURF) dan algoritma Maximally Stable Extremal Regions (MSER). Penelitian ini menggunakan satu citra target dan dua citra referensi, kesemua citra akan di konversi dalam citra grayscale. Metode yang digunakan adalah menggabungkan algoritma feature descriptor dengan regions detector. Hasil penelitian menunjukkan bahwa: 1) waktu proses pencocokan fitur sebesar 1,347 detik, 2) waktu menampilkan hasil sebesar 1,125 detik, 3) waktu proses membandingkan dan plotting titik sebesar 0,859 detik, 4) waktu proses deteksi fitur sebesar 0,251 detik, dan 5) waktu ekstraksi fitur sebesar 0,093 detik. Waktu yang dibutuhkan untuk proses eksekusi kedua algoritma ini secara keseluruhan adalah 5,216 detik. Keakuratan penggabungan kedua algoritma ini sangat baik dalam menunjukkan kecocokan dan lokasi objek dalam sebuah citra. Hasil tingkat akurasi untuk setiap proses adalah sebagai berikut; 1) akurasi deteksi fitur sangat baik, 2) ekstraksi fitur sangat baik, 3) kesesuaian titik kandidat sangat baik, 4) penentuan batasan lokasi objek sangat baik, dan 5) waktu proses eksekusi algoritma cepat. Berdasarkan hasil penelitian ini maka penggabungan algoritma SURF dan MSER baik digunakan dalam menentukan sebuah objek dalam sebuah citra.

Keywords: Citra, Grayscale, Speeded Up Robust Features (SURF), Maximally Stable Extremal Regions (MSER), Titik Fitur

1. Pendahuluan

Pendeteksian objek tertentu dalam sebuah citra melalui pencocokan titik-titik fitur pada dasarnya adalah proses bagaimana menemukan kecocokan titik fitur yang saling berhubungan antara titik fitur pada sebuah citra referensi dengan titik fitur pada citra target, dengan pengertian lain bagaimana menemukan kecocokan fitur unik dari masing-masing citra. Titik fitur adalah informasi abstrak atau karakteristik unik dari sebuah objek dalam citra.

Permasalahan utama dalam proses pencocokan dan klasifikasi citra adalah bagaimana menemukan keseragaman informasi terkait dengan ukuran, intensitas, rotasi, skala, noise dan viewpoint. Proses pencocokan dan klasifikasi citra berbasis pada diskriptor fitur, salah satunya adalah algoritma SURF. Untuk mendapatkan hasil yang lebih akurat dipadukan dengan algoritma MSER yang merupakan detektor area. Dalam penelitian ini algoritma SURF bertindak sebagai *feature descriptor* dan algoritma MSER sebagai *regions detector*.

Informasi abstrak yang disebut fitur pada sebuah citra dinamakan titik fitur, ini dapat diaplikasikan untuk melakukan proses pendeteksian objek. Dalam penelitian ini akan dijelaskan tentang pencocokan titik-titik fitur citra dengan menggunakan algoritma Speeded Up Robust Features (SURF) (Bay H et al., 2006) bertujuan untuk mendeteksi fitur lokal sebuah citra dengan cepat dan lebih akurat. Algoritma SURF menggunakan penggabungan algoritma citra integral (*integral image*) dan blob detection (MSER) berdasarkan determinan dari matriks Hessian.

2. Metode Penelitian

Metodologi penelitian yang dilakukan meliputi studi literatur, rancangan aplikasi simulasi, implementasi, pengujian, dan dokumentasi. Berikut ini penjelasan masing-masing tahapan.

▪ Studi literatur

Studi literatur merupakan tahap awal penelitian yang dimulai dengan mengumpulkan berbagai referensi yang berhubungan dengan citra, pengolahan citra, penggunaan algoritma *Speeded Up Robust Feature* (SURF) dan algoritma *Maximally Stable Extremal Regions* (MSER).

▪ Rancangan Aplikasi Simulasi

Pada tahap ini dimulai dengan analisa kinerja dari masing-masing algoritma *Speeded Up Robust Feature* (SURF) dan algoritma *Maximally Stable Extremal Regions* (MSER) yang akan dijelaskan dalam *flowchart*, kemudian juga dilakukan perancangan terhadap tampilan dari aplikasi simulasi yang akan dibuat.

▪ Implementasi

Langkah selanjutnya adalah implementasi dari kedua algoritma *Speeded Up Robust Feature* (SURF) dan *Maximally Stable Extremal Regions* (MSER) untuk melakukan proses pendeteksian objek pada citra dengan menggunakan pencocokan titik-titik fitur

Langkah pertama adalah start untuk memulai program selanjutnya melakukan pembacaan citra referensi dan citra target (read image reference image and target image) sebagai inputan, kemudian melakukan pendeteksian titik-titik fitur (*detection of feature points*) yang kemudian diteruskan pada proses ekstrak fitur (*extract feature descriptor*), selanjutnya menemukan titik-titik yang cocok diantara citra referensi dan citra target (*find match points between reference and target image*) dan kita teruskan dengan proses penentuan lokasi objek dengan menggunakan pencocokan titik (*locate the object using matching points*) dan proses yang terakhir adalah menampilkan objek yang telah terdeteksi.

▪ Pengujian

Selanjutnya adalah melakukan proses pengujian, implementasi algoritma dengan tujuan untuk mendapatkan performa dan hasil yang diinginkan.

▪ Dokumentasi

Dokumentasi hasil penelitian adalah untuk membuat laporan penelitian sebagai bukti tertulis kegiatan penelitian telah dilaksanakan secara baik dan benar serta dapat bermanfaat bagi pengembangan dan penelitian selanjutnya.

2. 1. *Speeded Up Robust Features* (SURF)

Algoritma *Speeded Up Robust Features* (SURF) terdiri dari beberapa tahapan utama yang dapat dijelaskan sebagai berikut:

Interest Point Detection

Kestabilan sebuah citra digital terhadap gangguan lokal dan global menjadi fokus penting dalam bagian tahapan algoritma *Speeded Up Robust Features* (SURF) ini. Untuk itu tahap pertama dimulai dengan deteksi *interest point* yang bertujuan untuk memilih titik yang mengandung banyak informasi dan sekaligus stabil terhadap gangguan lokal atau global (perubahan; skala, rotasi, *blurring*, pencahayaan, dan perubahan bentuk) dalam citra digital. Algoritma *Speeded Up Robust Features* (SURF), dipilih detektor *interest point* yang mempunyai sifat invarian terhadap skala, yaitu *blob detection* yang menggunakan *Maximally Stable Extremal Regions* (MSER). Dimana *blob* merupakan area pada citra digital yang memiliki sifat yang konstan atau bervariasi dalam kisaran tertentu. Untuk melakukan komputasi *blob detection* ini,

digunakan determinan matriks Hessian (DoH) dari citra. Definisi Gaussian diartikan sebagai:

$$g(\sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \quad (4)$$

Dimana $S_{\alpha\beta}$ mewakili derivatif kedua dari penghalusan citra. Determinan dari matriks Hessian inilah yang akan digunakan sebagai dasar algoritma SURF karena sifat invarian terhadap skala, kestabilan dan berulang dengan mudah. Dimana pada algoritma *Speeded Up Robust Features* determinan matriks Hessian dihitung dari respon wavelet Haar dengan menggunakan *integral image* nya secara optimal (Crow, 1984). *Integral image* merupakan *image* $I(x)$ dimana setiap nilai $x = (x, y)^T$ yang menjadi penjumlahan semua piksel pada area *integral image* dan dihitung dengan persamaan berikut.

$$I(x) = \sum_{i=0}^{i \leq x} \sum_{j=0}^{j \leq y} I(x,y) \quad (5)$$

Scale Space Representation

Ukuran citra yang tidak seragam atau tidak sama menjadi faktor kesulitan dalam proses membandingkan fitur yang terdapat dalam sebuah citra. Terkait hal tersebut maka diperlukan suatu mekanisme tersendiri yang dapat menyelesaikan proses dan menangani perbedaan ukuran dengan menggunakan metode perbandingan skala. Dalam metode ini kita menggunakan *scale space*, di mana citra diimplementasikan dalam bentuk sebuah *image pyramid* (Lowe DG, 2004). Citra secara berulang akan diperhalus (*smoothing*) dengan fungsi Gaussian dan secara berurutan dengan cara *sub-sampling* untuk mencapai tingkat tertinggi pada piramida. Dengan menggunakan *integral image*, perhitungan ini tidak perlu dilakukan secara iteratif dengan menggunakan filter yang sama, tetapi dapat menggunakan filter dengan ukuran berbeda-beda ke dalam beberapa skala citra yang berbeda.

Features Description

Fitur didefinisikan sebagai bagian yang mengandung banyak informasi pada sebuah citra, dan fitur ini digunakan sebagai titik awal untuk algoritma deteksi objek. Tujuan dari proses deteksi fitur ini adalah untuk mendapatkan deskripsi dari fitur-fitur dalam citra yang diamati. Langkah pertama (Bay H *et al*, 2008) adalah melihat orientasi yang dominan pada *interest point* yang terdapat dalam citra, kemudian membangun suatu area yang akan diambil nilainya dan mencari fitur korespondensi pada citra perbandingan. Dalam penentuan orientasi suatu citra kita menggunakan filter wavelet Haar, disini dapat ditentukan tingkat kemiringan suatu fitur yang diamati. Selanjutnya untuk deskripsi fitur dalam algoritma *Speeded Up Robust Features* (SURF), digunakan hanya perhitungan *gradient histogram* dalam empat kelompok (*bins*) saja dengan tujuan mempercepat perhitungan, yaitu $v = \sum d_x, \sum d_y, \sum |d_x|, \sum |d_y|$ dengan d_x adalah respon *wavelet Haar* pada arah horizontal dan d_y dalam arah vertikal. *Scale space* terbagi kedalam bilangan yang disebut *octave*.

Setiap *octave* merepresentasikan respon filter yang diperoleh dengan melakukan proses konvolusi citra yang dimasukkan dengan ukuran filter yang menaik. *Lokalisasi keypoint* dilakukan dengan beberapa proses; proses pertama, menentukan *threshold* untuk *keypoint* ketika *threshold* dinaikan jumlah *keypoint* yang terdeteksi lebih kecil begitupun sebaliknya. Oleh karena itu, *threshold* biasanya disesuaikan pada setiap aplikasi. Proses kedua, *non-maximumsuppression*, proses ini dilakukan untuk mencari sekumpulan kandidat *keypoint* dengan membandingkan tiap-tiap piksel citra pada *scale space* dengan 26 tetangga. 26 tetangga piksel itu terdiri atas 8 titik di skala asli dan 9 titik di tiap-tiap skala diatas dan dibawahnya. Proses inilah yang menghasilkan *keypoint* dari sebuah citra. Sekali fitur-fitur sudah terdeteksi, maka setiap

area disekitar fitur akan diekstrak dan dideskripsikan dengan demikian proses penemuan kesamaan atau kemiripan dalam citra lain akan lebih mudah dilakukan.

2.1.4 *Feature Matching and Recognition*

Dalam tahap ini, kita akan membandingkan fitur hasil perhitungan pada proses sebelumnya tetapi hanya bila terdapat perbedaan kontras, yang dideteksi melalui tanda dari *trace* matriks Hessian. Objek diwakili oleh vektor fitur, kesamaan dan perbedaan antara dua citra dapat diketahui dengan cara membandingkan antara urutan kedua vektor fitur tersebut untuk dapat melakukan klasifikasi citra. Secara umum metode untuk membandingkan citra dapat dilakukan dengan; 1) melalui perbandingan jarak kedua vektor fitur, 2) melalui kesamaan kedua vektor fitur. Misalnya, dua citra dibandingkan dengan menghitung jarak antara dua vektor fitur, semakin pendek jarak maka lebih besar kesamaan dan lebih kecil perbedaan.

Dimana untuk setiap *sub-region*, penjumlahan dari dx , $|dx|$, dy dan $|dy|$ dihitung berdasarkan pada orientasi sampel grid, dimana dx dan $|dx|$ adalah x respon wavelet Haar pada arah horizontal dan dy dan $|dy|$ adalah y respon wavelet Haar pada arah vertikal dan kedua arah ini merujuk pada orientasi titik utama (*keypoint*). Sehingga vektor dimensi 4 dapat dihitung dari *sub-region* dengan formula sebagai berikut:

$$v = \sum d_x, \sum d_y, \sum |d_x|, \sum |d_y| \quad (7)$$

3.1.5. Keypoint

Penentuan titik utama (*keypoint*) merupakan salah satu proses penting dimana *keypoint* adalah titik-titik dari sebuah citra yang nilainya tidak akan berubah atau terpengaruh terhadap terjadinya perubahan skala, rotasi, *blurring*, pencahayaan, dan juga perubahan bentuk. Agar invarian terhadap perubahan skala maka proses yang dilakukan pertama kali adalah membuat ruang skala (*scalspace*). Perubahan bentuk pada citra dapat terjadi karena; perubahan yang disebabkan kurang sempurnanya data citra, terhalang oleh objek-objek tertentu, atau dikarenakan hasil awal dari pengambilan citra yang kurang sempurna. Deteminan Hessian dapat dihitung dengan menggunakan formula sebagai berikut:

$$\det(H^t) \approx D_{xx}^t D_{yy}^t - (0.9_{xy}^t)^2 \quad (8)$$

Maximally Stable Extremal Regions (MSER)

Algoritma *Maximally Stable Extremal Regions* (MSER) secara dasar adalah *regions detector*. Dasar perhitungan MSER dimulai dengan melakukan pemilihan atau sortir urutan piksel-piksel dari intensitas rendah ke intensitas tinggi atau sebaliknya (misal pada citra *grayscale* yang mempunyai intensitas $\{0, \dots, 255\}$). Intensitas ini yang akan dinamakan *threshold*. Iterasi dimulai dari *threshold* rendah (0) ke *threshold* tinggi (255) dan pada masing-masing *threshold* dilakukan perhitungan area. Diman area yang tidak mengalami perubahan ketika *threshold* diubah-ubah dinamakan MSER *regions*. MSER banyak digunakan pada aplikasi *text localization and recognition*. Dalam setiap *threshold* citra, *areaextremal* ditandai sebagai komponen yang berhubungan, sehingga terbentuk sebuah urutan dari masing-masing komponen. Di beberapa *threshold* dua atau lebih komponen akan bergabung menjadi satu. Pada *threshold* ini terjadi perubahan bentuk komponen secara signifikan, hal ini menjadikan komponen tidak stabil dan letak lokasi menjadi tidak presisi, terutama dalam hal perubahan intensitas dan *noise*. *Maximally Stable Extremal Regions* akan diekstrak dari urutan komponen pada *threshold* yang direpresentasikan oleh perbedaan minimum komponen lokal dalam area di dalam ruang *threshold*. MSER sebagai detektor dikembangkan untuk memecahkan perbedaan korespondensi dalam *baseline stereo system*. Sistem ini menciptakan cekungan dan kedalaman geometris yang kompleks, dan tergantung pada dasar antara kamera dan jarak subjek ke kamera,

berbagai efek geometris harus dikompensasi. Dalam sistem stereo secara luas, fitur lebih dekat dengan kamera akan lebih terdistorsi di bawah transformasi, sehingga sulit untuk menemukan ketepatan dan kesesuaian antara bagian kiri atau kanan dari pasangan citra. Dalam MSER termasuk didalamnya proses pemilihan piksel menjadi satu himpunan area (*set of regions*) berdasarkan *threshold* intensitas biner; area dengan nilai piksel yang sama pada nilai *threshold* dalam pola komponen yang terhubung merupakan nilai yang dianggap paling stabil (*maximally stable*).

Algoritma *Maximally Stable Extremal Regions* (MSER) terdiri dari beberapa tahapan utama yang dapat dijelaskan sebagai berikut:

- Pengurutan seluruh piksel berdasarkan intensitas.
- Meletakkan piksel satu persatu (sesuai urutan intensitas) di dalam citra, dan melakukan pembaharuan struktur keterkaitan komponen, yang berasal dari tingkatan area-area *extremal*.
- Menghitung variasi area dari setiap area-area *extremal*. Dengan menggunakan formula:

$$var_i = \frac{Area_{i+\Delta} - Area_i}{Area_i} \quad (9)$$

Dimana i mewakili area-area *extremal* dengan nilai intensitas tertinggi dan $i + \Delta$ merujuk pada perluasan area ke i , dengan intensitas maksimal sebesar $i + \Delta$, var_i adalah pebedaan relatif dari area dimana terdapat intensitas tertinggi dari i ke $i + \Delta$.

- Melalui urutan/tingkatan area-area *extremal*. Mencari satu *maximally stable extremal* yang memiliki "var" terkecil dari tingkat utama.

Pencocokan fitur

Fitur merupakan titik unik yang berada pada lokasi tertentu, sebuah objek di representasikan sebagai vektor, dimana dua buah citra dapat dibandingkan kesamaan atau perbedaannya dengan cara membandingkan urutan kedua vektor didalam sebuah citra. Pada dasarnya ada dua cara untuk melakukan perhitungan perbandingan ini; pertama dengan menghitung jarak kedua vektor dan kedua dengan menghitung kesamaan atau kemiripan vektor. Hubungan hubungan atau koresponden fitur seperti dua himpunan matriks fitur p sebagai masukan, yang dibagi menjadi matriks m dan matriks n . Masukan setiap urutan pasangan akan mengalami proses pencocokan antara fitur1 dan fitur2. Urutan elemen pertama fitur dalam fitur1 yang cocok dengan fitur dalam urutan elemen kedua pada fitur2. Nilai-nilai matriks yang sudah cocok menjadi nilai dasar batasan perhitungan matriks yang dipilih. Nilai matriks kecocokan berhubungan pada *output* urutan pasangan matriks fitur (*indexpairs*). Dimana nilai akan sangat tergantung pada setiap pemilihan proses pencocokan fitur.

3. Hasil dan Analisis

Penelitian ini menggunakan dua buah citra sebagai citra referensi dan satu buah citra sebagai citra target kesemua citra dikonversikan kedalam citra dengan format *grayscale*. Analisa hasil penelitian secara keseluruhan dapat dijelaskan sebagai berikut:

Analisa Citra Target

Menyiapkan citra sebagai masukan yang terdiri dari tiga buah citra yang terbagi menjadi; satu citra target dan dua sebagai citra referensi. Kemudian lakukan konversi citra RGB menjadi citra dengan format *grayscale* dengan tujuan agar proses perhitungan citra menjadi lebih mudah.

Analisa Citra Referensi

Perlakuan yang sama dengan langka pertama, menyiapkan dua buah citra sebagai citra referensi pertama dan kedua dengan format RGB dan mengkonversikannya kedalam citra dengan format *grayscale*. Proses konversi citra format RGB ke dalam citra *grayscale* pada dasarnya adalah proses penghilangan informasi hue dan saturasi dalam peta warna pada saat proses penguatan lumen. Secara matematis konversi ini adalah proses pengurangan bobot penjumlahan nilai R (*red*), G (*green*), dan B (*blue*).

Analisa Deteksi Fitur

Tahapan berikutnya adalah menentukan atau mendeteksi titik-titik fitur menggunakan algoritma SURF (*feature descriptor*) pada citra target, citra referensi 1 dan citra referensi 2, hal ini bertujuan untuk menentukan titik fitur pada citra referensi pertama dan citra referensi kedua serta citra target. Kemudian menampilkan seberapa banyak titik fitur yang ditemukan pada citra referensi pertama atau citra referensi kedua serta seberapa banyak titik fitur yang ditemukan pada citra target. Dimana titik fitur ini merupakan *interest point* dari masing-masing citra

Proses selanjutnya adalah mendeteksi area dari masing-masing citra dengan menggunakan algoritma MSER (*regions descriptor*) yang digunakan sebagai teknik dalam proses *blob detection* berdasarkan determinan dari matriks Hessian {persamaan (1), (2) dan (3)}, sementara perhitungan untuk *integral image* menggunakan persamaan (6). Hasil yang didapatkan dapat dilihat pada gambar 3.b yang ditandai dengan garis-garis berwarna hijau berbentuk elips (1b, 2b, dan 3b).

Proses pendeteksian area dengan menggunakan *Maximally Stable Extremal Regions* untuk masing-masing citra pada dasarnya adalah proses transformasi invarian yang harus stabil terhadap perubahan-perubahan seperti; perubahan iluminasi, translasi, rotasi, skala dan *viewpoint*. Karena hal tersebut maka deteksi area harus stabil dan dapat diulang. Dalam penelitian ini sebagai contoh untuk melihat kestabilan terhadap salah satu perubahan akan ditampilkan hasil perubahan rotasi, hasil yang didapatkan deteksi area cukup stabil terhadap perubahan rotasi dari citra

Hasil korelasi nilai n fitur pada area CR1 dan CR3 (*inliers*) sementara CR2 (*outliers*) pada masing-masing citra dari hasil proses transformasi antara citra referensi 1, citra referensi 2 dan citra target. Hasil ini nantinya digunakan sebagai *output* korelasi pasangan-pasangan vektor fitur 1 dan fitur 2.

Pada hasil pengujian deteksi area yang dilakukan pada ketiga citra menggunakan algoritma MSER terlihat proses deteksi area untuk masing-masing citra berbeda, hal ini akan tergantung pada fitur-fitur yang dianggap sebagai titik-titik kandidat pada area tertentu dalam sebuah citra.

Hasil penggabungan ini memperlihatkan dengan hasil yang lebih kaya akan penemuan fitur yang nantinya akan diekstraksi. Ini menandakan penggabungan kedua algoritma memiliki performa yang lebih baik dalam menemukan kandidat titik-titik fitur dalam sebuah citra. Hal ini juga mengindikasikan bahwa proses pendeteksian kecocokan titik-titik fitur akan lebih baik hasilnya jika dibanding kita hanya menggunakan algoritma SURF saja tanpa memadukannya dengan algoritma MSER.

Analisa Ekstraksi Fitur

Selanjutnya adalah melakukan ekstraksi fitur (*extract feature descriptors*) dari titik perhatian (*interest point*) dari masing-masing citra (citra target dan citra referensi). Kemudian dilanjutkan dengan menemukan pasangan titik yang diduga cocok dengan menggunakan deskriptor masing-masing, kemudian menampilkan hasil kecocokan fitur

sehingga kita dapat mengetahui titik-titik kecocokan secara keseluruhan(garis berupa lingkaran-lingkaran berwarna hijau termasuk *outliers*).

3.1.5 Analisa Kesesuaian Titik Kandidat

Langkah berikutnya adalah menentukan kandidat titik-titik kecocokan(*putative point matches*) termasuk semua titik - titik kandidat dan titik diluar area target. (*outliers*). Tahapan berikutnya adalah menentukan informasi terkait lokasi titik pada citra target, ini kita sebut dengan perkiraan transformasi geometris, dimana transformasi ini berkaitan dengan titik-titik kecocokan pada citra, sekaligus menghilangkan titik-titik yang diduga merupakan titik kecocokan pada bagian luar (*outliers*) objek. Transformasi ini memungkinkan kita untuk melakukan lokalisasi atau menentukan lokasi titik-titik kecocokan dengan lokasi tertentu di dalam citra target. Proses selanjutnya adalah menampilkan hasil pasangan-pasangan titik kecocokan bagian dalam (*outliers*) saja, setelah kandidat titik kecocokan bagian luar (*outliers*) pada citra dihapus

Analisa Penentuan Batasan Lokasi Objek

Penentuan batasan citra target merupakan tahapan berikutnya, kemudian menempatkan batasan kedalam sistem kordinat citra target, ini nantinya akan menandai letak objek dalam citra target (LC1 dan LC2).

Tahap akhir dari implemtasi ini adalah membuat batasan lokasi objek dalam citra target sedemikian rupa yang terdiri dari; sisi atas kanan, sisi atas kiri, sisi bawah kanan dan sisi bawah kiri sehingga membentuk kotak persegi yang nantinya secara otomatis akan menunjukkan dan membatasi lokasi citra referensi yang telah terdeteksi dan dibatasi dalam citra target.

Kotak-kotak pembatas menunjukkan lokasi objek dalam citra target. Kemudian menampilkan kotak berwarna yang ditandai oleh kotak warna kuning (LC1) dan hijau (LC2) dalam citra target.

Analisa Waktu Proses dan Tingkat Akurasi

Hasil dari proses eksekusi kedua algoritma antara SURF dan MSER secara detail dapat dilihat pada tabel 1 di bawah ini. Dari total waktu pemrosesan sebesar 5.216 detik, waktu terlama adalah pada proses menampilkan kecocokan fitur sebesar 1.374 detik atau setara dengan 26.3%, kemudian waktu pada proses menampilkan hasil sebesar 1.125 detik setara dengan 21.6% dan kemudian waktu proses perbandingan dan *ploting* sebesar 0.859 detik setara dengan 16.5%. Waktu proses yang lain sangat kecil dan secara keseluruhan tertera pada tabel 1.

Tabel 1. Hasil Waktu Proses

Fungsi	Waktu (s)	%Waktu (%)
Menampilkan Kecocokkan Fitur	1.374	26.3
Menampilkan Hasil Keseluruhan	1.125	21.6
Membandingkan dan plot Titik	0.859	16.5
Proses Deteksi Fitur	0.251	4.8
Proses Ekstrasi Fitur	0.093	1.8
Proses Penandaan	0.007	1.5
Perhitungan Transformasi Geometris	0.047	0.9
Proses pembacaan	0.047	0.9
Proses Pencocokan Fitur	0.032	0.6
Transformasi Geometris ke dalam Matrik	0.015	0.3
Ploting/pengaturan Hasil	0	0
Properti Plot dan Aksis pada Grafik	0	0

Fungsi lain-lain	1.296	24.9
Total	5.216	100

4. Kesimpulan

Berdasarkan dari kelemahan-kelemahan dasar pada algoritma Speeded Up Robust Features (SURF) terkait perubahan-perubahan terhadap rotasi, ukuran, blurring, dan bentuk maka tulisan ini bertujuan untuk mendeskripsikan penggabungan dua algoritma, Speeded Up Robust Features (SURF) sebagai deskriptor fitur dengan Maximally Stable Extremal Regions (MSER) sebagai detektor area. Pertama adalah penggunaan Hessian matriks SURF sebagai metoda untuk proses penghitungan titik fitur yang dilanjutkan dengan menghitung pendeteksian area dengan menggunakan algoritma regions detector MSER. Penggabungan kedua algoritma ini menggunakan citra berformat grayscale, ini dimaksudkan untuk memudahkan proses perhitungan untuk mendapatkan titik fitur dalam sebuah citra. Hasil dari percobaan dengan menggunakan penggabungan kedua algoritma SURF dan MSER memperlihatkan adanya peningkatan hasil jumlah pendeteksian titik-titik fitur. Dengan menggunakan gabungan deksriptor fitur SURF dan detektor area MSER kita dapat meningkatkan waktu proses deteksi titik-titik kecocokan. Deskriptor akan melakukan evaluasi terhadap ketidakstabilan yang diakibatkan oleh perubahan rotasi, ukuran, blurring, iluminasi dan bentuk. Hal ini dapat menambah akurasi pada proses pendeteksian titik-titik kecocokan. Kecepatan waktu eksekusi didapatkan sebagai berikut; total waktu pemrosesan sebesar 5,216 detik, waktu proses menampilkan kecocokan fitur sebesar 1,374 detik atau setara dengan 26,3%, kemudian waktu pada proses menampilkan hasil sebesar 1,125 detik setara dengan 21,6% dan kemudian waktu proses perbandingan dan plotting sebesar 0,859 detik setara dengan 16,5%. Hasil tingkat akurasi untuk setiap proses adalah sebagai berikut; akurasi deteksi fitur sangat baik, ekstraksi fitur sangat baik, kesesuaian titik kandidat sangat baik, penentuan batasan lokasi objek sangat baik dan waktu proses eksekusi algoritma cepat. Walaupun penelitian ini belum ideal kemungkinan penyempurnaan dan penggunaan variasi beberapa algoritma terkait pendeteksian objek dalam sebuah citra dimasa datang dapat terus dilakukan mengikuti perkembangan teknologi khususnya di bidang komputer visi dan pengolahan citra sesuai kebutuhan.

Referensi

- [1] Alwanin, Rawabi (2014). An Approach to Image Classification Based on SURF Descriptors and Colour Histograms. *A dissertation submitted to the University of Manchester for the degree of Master of Science in the Faculty of Engineering and Physical Sciences*, 39 – 40. University of Manchester.
- [2] Bay, H, Ess A, Tuytelaars, T, Van Gool, L (2008). Speeded-Up robust features (SURF). *Computer Vision and Image Understanding (CVIU)*, 110 (3), 346 – 359.
- [3] Bay, H, Ess A, Tuytelaars, T, Van Gool, L (2006). SURF: speeded up robust features. *Proceedings of the 9th European Conference on Computer Vision*, 3951 (1), 404 – 417. Springer LNCS.
- [4] Crow, Franklin (1984). Summed-area tables for texture mapping. *SIGGRAPH '84: Proceedings of the 11th Annual Conference on Computer Graphics and Interactive Techniques*, 207 – 212.
- [5] Forssén Per-Erik and David G Lowe. (2007). [Shape descriptors for maximally stable extremal regions](#). *Computer Vision, ICCV*. IEEE 11th International Conference on, 1-2,
- [6] Huang Yea-Shuan, Yu Hung-Hsiu, Ou Zhi-Hong, Hsieh Hsiang-Wen (2012). A Texture-Based Feature Point Detection and Matching Method, *Journal of Information Technology and Applications*, Vol. 6, No. 1. 3 – 4. Chung-Hua University, Hsinchu, Taiwan.

- [7] J. Matas, O. Chum, M. Urban and T. Padjla. (2002). Robust wide baseline stereo from maximally stable external regions, *Proc. British Machine Vision Conference*, Vol. 1, 384-393.
- [8] Pedersen Jacob Toft (2011). Group Study. SURF: Feature detection & description.
Computer Science, Aarhus.
- [9] Lowe, D.G. (2004). Distinctive image features from scale-invariant keypoints,
International Journal of Computer Vision.
- [10] Lili (2014). Image Matching Algorithm based on Feature-point and DAISY Descriptor, *Journal Of Multimedia*, Vol 9, No. 6. 830 – 832. School of Business, Sichuan Agricultural University, Sichuan Dujianyan 611830, China.
- [11] Lowe, D. G. (1999). Object recognition from local scale-invariant features.
Proceeding of the International Conference on Computer Vision, Corfu.
- [12] Singh Upendra, Singh Sidhant Shekhar, Srivastava Manish Kumar (2015). Object Detection and Localization Using SURF Supported By K-NN, *International Journal of Computer Science Trends and Technology (IJCST)*, Vol 3, Issue 2.89 – 90. Department of Computer Science and Engineering. Madan Mohan Malaviya University of Technology.