

## LOKALISASI *MOBILE ROBOT* BERDASARKAN CITRA KAMERA OMNI MENGUNAKAN FITUR SURF

Susijanto T. Rasmana<sup>\*1</sup>, Harianto<sup>2</sup>, Pauladie S.<sup>3</sup>, Anan P. Abseno<sup>4</sup>, Zendi ZR. Permana<sup>5</sup>

Program Studi Teknik Komputer, Universitas Dinamika

Email: <sup>1</sup>susyanto@dinamika.ac.id, <sup>2</sup>hari@dinamika.ac.id, <sup>3</sup>pauladie@dinamika.ac.id, <sup>4</sup>pepe@dinamika.ac.id,  
<sup>5</sup>zendiraga@gmail.com

\*Penulis Korespondensi

(Naskah masuk: 04 Oktober 2019, diterima untuk diterbitkan: 8 Oktober 2020)

### Abstrak

Deteksi lokasi diri atau lokalisasi diri adalah salah satu kemampuan yang harus dimiliki oleh *mobile robot*. Kemampuan lokalisasi diri digunakan untuk menentukan posisi robot di suatu daerah dan sebagai referensi untuk menentukan arah perjalanan selanjutnya. Dalam penelitian ini, lokalisasi robot didasarkan pada data citra yang ditangkap oleh kamera *omnidirectional* tipe *catadioptric*. Jumlah fitur terdekat antara citra 360° yang ditangkap oleh kamera Omni dan citra referensi menjadi dasar untuk menentukan prediksi lokasi. Ekstraksi fitur gambar menggunakan metode Speeded-Up Robust Features (SURF). Kontribusi pertama dari penelitian ini adalah optimasi akurasi deteksi dengan memilih nilai *Hessian Threshold* dan jarak maksimum fitur yang tepat. Kontribusi kedua optimasi waktu deteksi menggunakan metode yang diusulkan. Metode ini hanya menggunakan fitur 3 gambar referensi berdasarkan hasil deteksi sebelumnya. Optimasi waktu deteksi, untuk lintasan dengan 28 gambar referensi, dapat mempersingkat waktu deteksi sebesar 8,72 kali. Pengujian metode yang diusulkan dilakukan menggunakan *omnidirectional mobile robot* yang berjalan di suatu daerah. Pengujian dilakukan dengan menggunakan metode *recall*, presisi, akurasi, *F-measure*, *G-measure*, dan waktu deteksi. Pengujian deteksi lokasi juga dilakukan berdasarkan metode SIFT untuk dibandingkan dengan metode yang diusulkan. Berdasarkan pengujian, kinerja metode yang diusulkan lebih baik daripada SIFT untuk pengukuran dengan *recall* 89,67%, akurasi 99,59%, *F-measure* 93,58%, *G-measure* 93,87%, dan waktu deteksi 0,365 detik. Metode SIFT hanya lebih baik pada presisi 98,74%.

**Kata kunci:** *Mobile Robot*, SURF, Kamera Omni, Lokalisasi, Optimasi waktu.

## MOBILE ROBOT LOCALIZATION BASED ON OMNI CAMERA USING SURF FEATURES

### Abstract

*Self-location detection or self-localization is one of the capabilities that must be possessed by the mobile robot. The self-localization ability is used to determine the robot position in an area and as a reference to determine the next trip direction. In this research, robot localization was by vision-data based, which was captured by catadioptric-types omnidirectional cameras. The number of closest features between the 360° image captured by the Omni camera and the reference image was the basis for determining location predictions. Image feature extraction uses the Speeded-Up Robust Features (SURF) method. The first contribution of this research is the optimization of detection accuracy by selecting the Hessian Threshold value and the maximum distance of the right features. The second contribution is the optimization of detection time using the proposed method. This method uses only the features of 3 reference images based on the previous detection results. Optimization of detection time, for trajectories with 28 reference images, can shorten the detection time by 8.72 times. Testing the proposed method was done using an omnidirectional mobile robot that walks in an area. Tests carried out using the method of recall, precision, accuracy, F-measure, G-measure, and detection time. Location detection testing was also done based on the SIFT method to be compared with the proposed method. Based on testing, the proposed method performance is better than SIFT for measurements with recall 89.67%, accuracy 99.59%, F-measure 93.58%, G-measure 93.87%, and detection time 0.365 seconds. The SIFT method is only better at precision 98.74%.*

**Keywords:** *Mobile Robot*, SURF, Omni Camera, Localization, Time Optimization.

## 1. PENDAHULUAN

Mobile robot merupakan jenis robot yang memiliki kemampuan berpindah dari satu tempat ke tempat yang lain. Robot jenis ini dapat digunakan untuk memindahkan barang, untuk penunjuk jalan (robot *guide*) (Dayoub, Cielniak, & Duckett, 2011), penjaga keamanan (robot *security*) (Theodoridis & Hu, 2012), robot *line follower* (Wajiansyah, Supriadi, Nur, & Bramanto W.P., 2018), juga dapat diaplikasikan sebagai robot penyelamat korban bencana (Kuswadi, O, Tamara, & S, 2018).

Untuk dapat berjalan secara independen, *mobile robot* perlu dibekali kemampuan untuk mengetahui posisi keberadaannya (lokalisasi diri), menelusuri area yang telah ditentukan (*trajectory tracking*) serta menghindari halangan (*obstacle avoidance*) yang ditemuinya. Lokalisasi diri pada robot adalah kemampuan untuk mengetahui keberadaan robot pada satu waktu. Informasi keberadaan ini digunakan untuk menentukan gerakan robot selanjutnya dalam melanjutkan perjalanan hingga tujuan.

Permasalahan yang ada pada proses lokalisasi ini mendeteksi tanda-tanda lingkungan di sekitar robot dan menentukan posisi robot saat itu berada (Piasco, Sidibé, Demonceaux, & Gouet-Brunet, 2018). Beberapa metode dikenalkan dalam melakukan lokalisasi, diantaranya menggunakan jaringan IEEE 802.15.4a dan *laser range finder* yang dilakukan oleh Röhrig (Röhrig, Hess, Kirsch, & Künemund, 2010). Tak cukup menggunakan satu kamera, Yang untuk melakukan lokalisasi robotnya menggunakan dua kamera yang dipasang di depan dan samping robot (Yang, Scherer, Yi, & Zell, 2017). Namun kamera yang digunakan adalah kamera reguler, meskipun menggunakan dua kamera namun jangkauan pandangan hanya di bagian depan dan samping kiri atau kanan saja. Selain secara visual, lokalisasi pada robot dilakukan menggunakan *Light Distance And Ranging* (LIDAR) dan kamera (Su et al., 2017). Su menggunakan deskriptor global untuk melakukan lokalisasi. Lokalisasi diri berdasarkan data citra arena sepak bola robot dilakukan oleh Hibino (Hibino, Yuta, Takahashi, & Maeda, 2012). Fitur citra yang diambil dari kamera omni dibandingkan dengan citra acuan. Hanya satu citra acuan yang digunakan karena lapangan hanya digunakan pada area lapangan yang tidak luas.

Lokalisasi berdasarkan visual juga dilakukan oleh Qu (Qu, Soheilian, Habets, & Paparoditis, 2016). Qu membandingkan metode SURF dan SIFT untuk lokalisasi berdasarkan data dari kamera. Deteksi lokasi dilakukan di luar gedung (*outdoor*) dengan membandingkan fitur dari dua citra yang dihasilkan SIFT dan SURF. Data lokasi diambil menggunakan kamera beresolusi tinggi (HD) yang diletakkan di atas sebuah mobil van. Pada Penelitian ini Qu mengklaim bahwa SIFT lebih unggul dari SURF, namun waktu yang diperlukan oleh SIFT lebih lama. Waktu yang digunakan untuk proses deteksi sangatlah penting

terutama pada deteksi yang bersifat *real time* (Li et al., 2019).

Berbeda dengan Qu, Valgren mengklaim bahwa SURF lebih unggul dibanding SIFT (Valgren & Lilienthal, 2010). Valgren membandingkan kedua metode untuk melakukan deteksi lokasi berdasarkan fitur lokal yang terbukti handal. Namun baik Qu maupun Valgren tidak membahas akurasi dan waktu yang diperlukan untuk deteksi dengan data dari robot yang sedang berjalan, serta penentuan lokasi robot yang menggunakan banyak data penanda area (*landmark*).

Berbeda dengan Qu dan Valgren yang melakukan lokalisasi di area *outdoor*, peneliti ini dilakukan di dalam gedung *indoor*. Menggunakan kamera *omnidirectional* yang memiliki sudut pandang horisontal atau dikenal dengan istilah *Field of View* (FOV) sebesar 360° kamera ini mampu mengambil citra di hampir seluruh lingkungan sekitar. Penelitian ini bertujuan mencari metode terbaik untuk lokalisasi robot berdasarkan data dari kamera omnidireksional menggunakan metode SURF. Kontribusi penelitian adalah lokalisasi robot pada area *indoor* dengan kamera *omni*. Selain itu juga dilakukan optimasi waktu deteksi berdasarkan data *landmark* sehingga waktu yang dibutuhkan untuk menentukan lokasi keberadaan robot menjadi lebih singkat.

Penggunaan SURF untuk deteksi lokasi karena metode ini memiliki kelebihan dalam kecepatan dan akurasi ekstraksi fitur (Bay, Ess, Tuytelaars, & Van Gool, 2008). Metode ini juga telah digunakan di banyak penelitian. Untuk menguji tingkat keberhasilan metode yang digunakan, dilakukan berdasarkan ketepatan dan kecepatan dalam mendeteksi lokasi keberadaan robot. Dalam pengujian dilakukan perbandingan dengan metode Scale Invariant Feature Transform (SIFT). Mengapa SIFT, karena berdasarkan penelitian yang dilakukan Qu dan Valgren terdapat perbedaan hasil untuk lokalisasi menggunakan SIFT dan SURF. Dalam penelitian Qu menyatakan SIFT lebih unggul, sedangkan Valgren menyatakan SURF lebih unggul. Maka dengan tujuan mengetahui kinerja dari metode yang diusulkan dibandingkan metode SURF dan SIFT untuk ekstraksi ciri citra area *indoor*.

## 2. METODE PENELITIAN

Penelitian dilakukan menggunakan *mobile omnidirectional robot* yang dijalankan menelusuri lantai dalam sebuah gedung. Terlebih dahulu dilakukan pengambilan data area yang akan ditelusuri untuk digunakan sebagai penanda area (*landmark*). Pengambilan data sebagai landmark dan juga untuk deteksi lokasi menggunakan kamera *omnidirectional* dengan jenis *catadioptric*.



Gambar 1. Robot omni

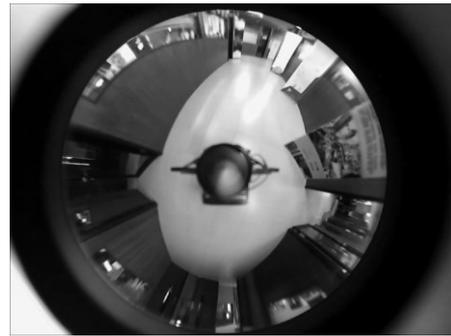
### 2.1 Omnidirectional Mobile Robot

Jenis *mobile robot* yang digunakan untuk pengambilan data dan pengujian pada penelitian ini adalah *omnidirectional mobile robot* (robot omni). Robot jenis ini dipilih karena memiliki sistem kontrol yang mampu menggerakkan robot ke segala arah secara langsung. Kemampuan robot omni ini didapatkan berkat penggunaan omni directional *poly roller wheel* (*Omniwheel*) (Harianto, Rivai, & Purwanto, 2013). *Omniwheel* atau roda *omni* merupakan roda yang memiliki silinder kecil mengelilingi roda dan dapat berputar tegak lurus terhadap arah bergulirnya roda. Dengan menggunakan roda omni maka robot dapat bergerak maju atau mundur dengan kekuatan penuh, dan dapat bergeser ke samping dengan mudah.

Robot yang digunakan berukuran  $p \times l \times t = 52\text{cm} \times 52\text{cm} \times 76\text{cm}$  seperti yang ditampilkan pada Gambar 1. Menggunakan penggerak tiga roda *omni* dengan desain *holonomic* membuat robot dapat menjangkau setiap lokasi dalam ruang 2 dimensi tanpa memutar badan.

### 2.2 Kamera Omnidirectional

Disebut juga kamera omni, adalah kamera yang dapat mengambil citra  $360^\circ$  dalam arah horisontal. Jenis kamera omni yang digunakan adalah *catadioptric camera*. Kamera jenis ini terdiri dari sensor *charge-coupled device* (CCD) dan cermin *hyperboloidal* yang terpasang di depan kamera (Feng, Chen, & Horng, 2010). Dengan desain ini kamera *catadioptric* secara horisontal memiliki FOV sebesar  $360^\circ$ , sedangkan secara vertikal sebesar kurang lebih  $100^\circ$ . Citra hasil yang ditangkap kamera jenis ini disajikan pada Gambar 2.

Gambar 2. Citra hasil tangkapan kamera omni jenis *catadioptric*

### 2.3 Speeded-Up Robust Features

SURF merupakan metode yang digunakan untuk mendeteksi fitur-fitur pada objek citra. Metode ini diperkenalkan oleh Bay (Bay et al., 2008). Dalam mendeteksi fitur SURF memiliki kelebihan dalam hal kecepatan dan konsisten walaupun objek mengalami perubahan karena translasi dan rotasi (Valgren & Lilienthal, 2010). Tahapan proses yang dilakukan dalam algoritma *SURF* adalah:

- Membuat citra integral dari citra yang akan dideteksi. Pengubahan bertujuan untuk menyederhanakan dan mempercepat perhitungan. Citra integral  $I_\Sigma$  dari citra  $I$  di lokasi  $l$  yang berukuran  $(x, y)$  persegi merupakan jumlah nilai piksel yang ada di dalamnya  $I_\Sigma(l)$ , yang didefinisikan sebagai persamaan (1).

$$I_\Sigma(l) = \sum_{i=0}^{i \leq x} \sum_{j=0}^{j \leq y} I(i, j) \quad (1)$$

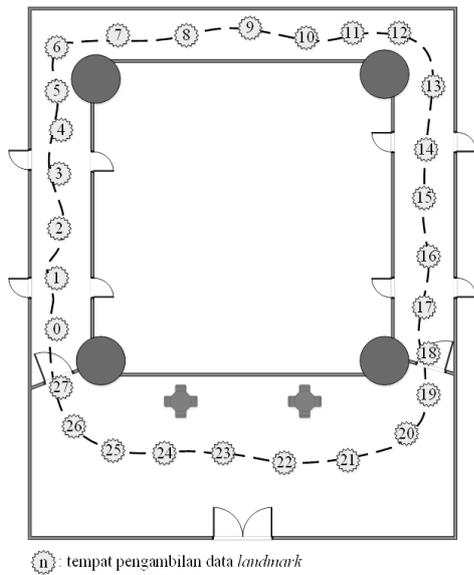
- Menentukan lokasi *keypoint* yang diawali dengan menentukan *threshold* untuk *keypoint* yang dilakukan pada citra integral, dilanjutkan dengan proses *non-maxima suppression* dalam rangka mencari calon-calon *keypoint*. Selanjutnya dilakukan proses interpolasi data untuk menentukan lokasi *keypoint*. Proses ini menggunakan menggunakan determinan *Hessian* dengan persamaan (2) seperti yang dilakukan oleh Brown (Brown & Lowe, 2002).

$$H(x) = H + \frac{\partial H^T}{\partial x} x + \frac{1}{2} x^T \frac{\partial^2 H}{\partial^2} x \quad (2)$$

Lokasi ekstrim yang diinterpolasi ditemukan menggunakan persamaan (3) yang merupakan turunan dari persamaan (2).

$$\hat{x} = -\frac{\partial^2 H}{\partial x^2} \frac{\partial H}{\partial x} \quad (3)$$

- Membuat deskriptor di daerah pixel sekitar *keypoint*. Proses diawali dengan mencari kesamaan orientasi berdasarkan informasi yang



Gambar 3. Peta lintasan dan tempat pengambilan data landmark

diperoleh dari daerah berbentuk lingkaran yang menjadi *keypoint* disekitar pixel. Selanjutnya dilakukan ekstraksi deskriptor di area *keypoint* menggunakan *Haar-wavelet* dalam arah horisontal  $d_x$  dan vertikal  $d_y$ . Kemudian respon *wavelet* yang didapatkan, untuk setiap sub region seluruhnya dijumlahkan. Masing masing sub region mempunyai 4 dimensi deskriptor vektor  $v$ , yaitu  $d_x, d_y, |d_x|$  dan  $|d_y|$  dengan persamaan (4).

$$y = \left( \sum dx, \sum dy, \sum |dx|, \sum |dy| \right) \quad (4)$$

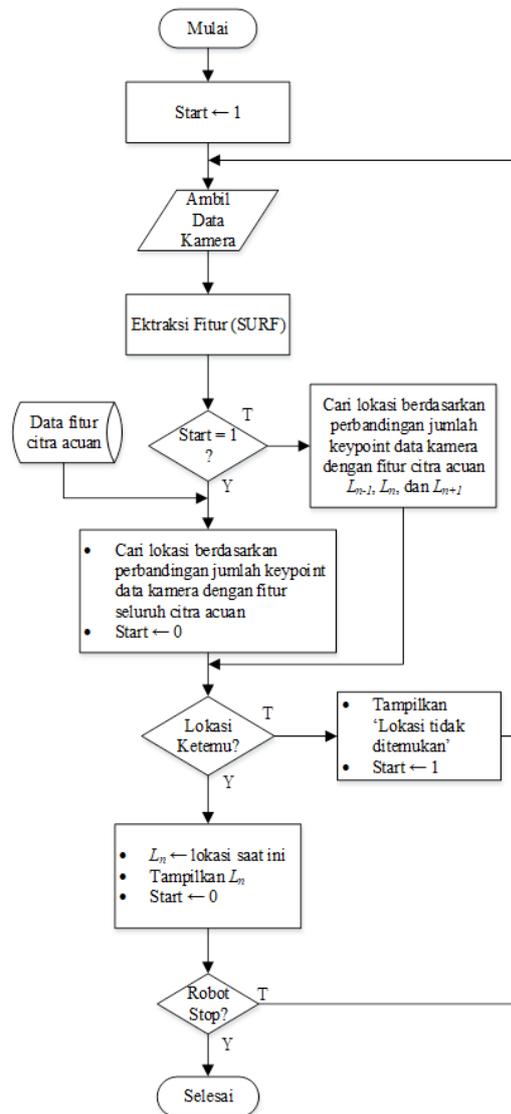
d. Fitur-fitur deskriptor yang telah dihasilkan menjadi acuan dalam mencocokkan kesamaan gambar. Kesamaan dua gambar dilakukan dengan mengukur kedekatan deskriptor yang merupakan jarak *euclidean* yang paling kecil dari kedua citra.

**2.4 Area Lintasan dan Citra Acuan**

Untuk keperluan pengujian lokalisasi robot, digunakan area lintasan di dalam gedung. Area yang digunakan adalah jalan di dalam gedung dengan luas  $\pm 600 \text{ m}^2$ , dengan lebar jalan 280 cm, dan panjang lintasan jalan  $\pm 6.400 \text{ cm}$ . Di sepanjang lintasan yang akan dilalui, sebelumnya dilakukan pengambilan beberapa data citra acuan (*landmark*). Total terdapat 28 *landmark* yang diambil di sepanjang lintasan. Peta lintasan dan area pengambilan *landmark* dapat dilihat pada Gambar 3.

**2.5 Deteksi Lokasi**

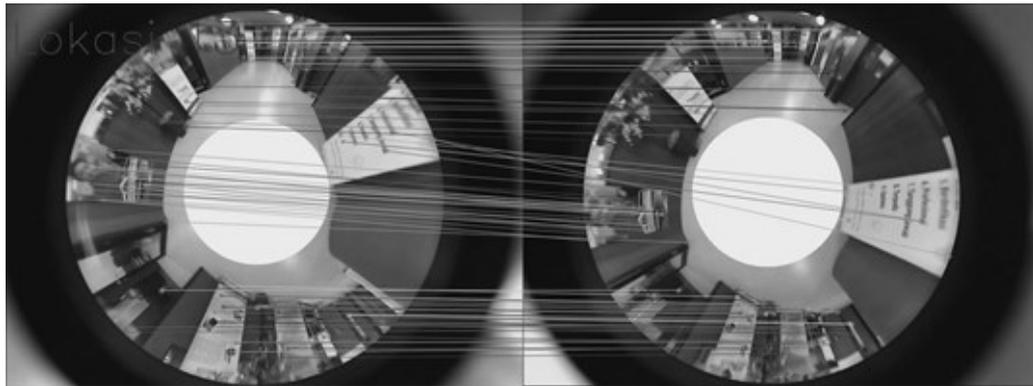
Deteksi lokasi keberadaan robot dilakukan berdasarkan kesamaan fitur citra yang ditangkap kamera robot ( $I_r$ ) pada saat itu dengan fitur citra acuan ( $I_p$ ). Baik  $I_r$  maupun  $I_p$  adalah fitur yang dihasilkan dari metode SURF. Tingkat kesamaan dari



Gambar 4. Flowchart optimasi deteksi lokasi keberadaan robot berdasarkan data kamera

citra didasarkan pada jumlah *keypoint* ( $n(I)$ ) yang sama dari  $I_r$  dan  $I_p$  kedua citra dengan kedekatan jarak *euclidean* yang telah ditentukan. Maka dari  $I_p$  yang berjumlah  $M$  dibandingkan satu persatu dengan  $I_r$  dan dipilih  $I_p$  ke- $m$  ( $I_{p_m}$ ) yang memiliki jumlah kesamaan fitur terbanyak.

Deteksi dengan membandingkan fitur suatu citra dengan fitur citra acuan akan memerlukan waktu yang relatif lama jika citra acuan berjumlah banyak. Dari studi yang telah dilakukan, untuk mendeteksi lokasi dari 1 frame menggunakan 28 citra acuan, rata-rata waktu yang diperlukan adalah 3,17 detik. Tentunya ini waktu yang relatif lama untuk deteksi lokasi, dan hal ini akan sangat menghambat jalannya robot. Dalam proses robot berjalan, tentunya lintasan yang dilalui merupakan suatu urutan. Misalkan robot melakukan perjalanan di area Gambar 3, jika perjalanan dimulai dari *landmark* 0 dan berjalan memutar searah jarum jam maka perjalanan berikutnya tentunya akan melewati *landmark* 1, 2, 3,



Gambar 5. Deteksi lokasi berdasarkan *keypoint* yang sama pada citra acuan (kiri) dan frame video (kanan)

dan seterusnya. Jika perjalanan dilakukan berlawanan arah jarum jam maka akan melewati *landmark* 27, 26, 25, dan seterusnya. Tidak mungkin robot akan melompat dari *landmark* 0 langsung menuju *landmark* 2. Oleh karena itu saat posisi awal dari robot sudah diketahui, misal berada di *landmark*  $L_n$ , maka deteksi hanya perlu dilakukan pada citra acuan untuk *landmark*  $L_{n+1}$  yang ada di depan dan  $L_{n-1}$  yang ada di belakang posisi robot saat itu. Dan hal ini berarti hanya tiga citra acuan yang akan dibandingkan fiturnya. Membandingkan fitur seluruh citra acuan hanya dilakukan saat pertama robot akan melakukan perjalanan atau saat terjadi kehilangan acuan di perjalanan. Dengan metode ini dapat mempercepat pencarian lokasi keberadaan robot sehingga waktu yang digunakan lebih optimal. Metode deteksi lokasi yang telah dioptimasi ini digambarkan dalam bentuk *flowchart* tersaji di Gambar 4.

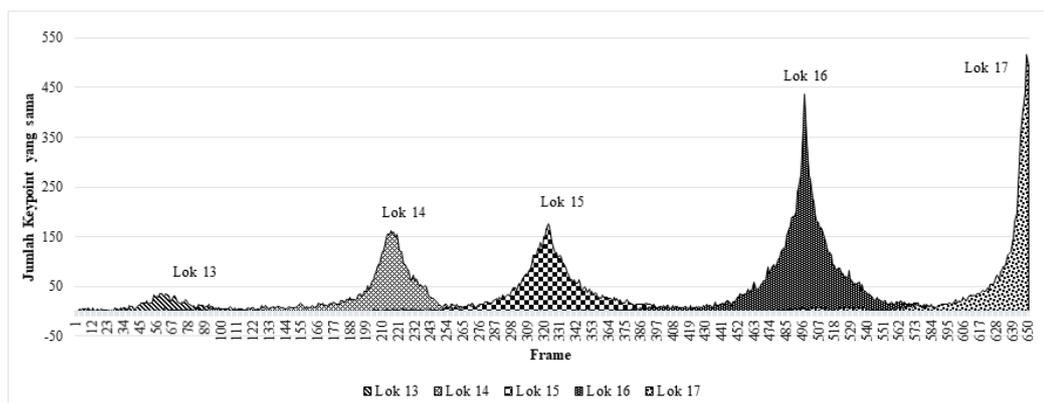
### 3. HASIL DAN PEMBAHASAN

Uji coba awal deteksi lokasi robot dilakukan dengan menjalankan robot dari lokasi 13 hingga 17 di lintasan yang ditampilkan pada Gambar 3. Saat berjalan robot merekam citra area yang dilalui dengan kamera *omni*. Hasil rekaman berupa video dengan durasi 54 detik dan jumlah frame 650. Dari rekaman tersebut setiap frame diekstraksi fitur dengan deskriptor SURF. Selanjutnya dilakukan deteksi dan penghitungan jumlah *keypoint* yang sama pada kedua

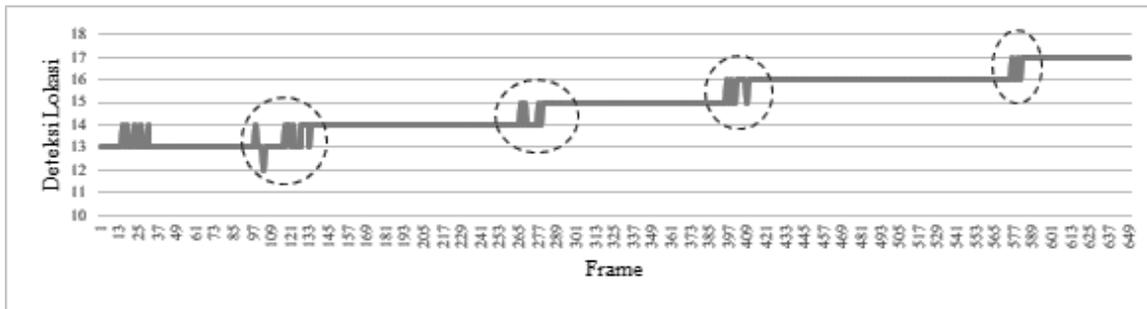
citra (frame video dan citra acuan). Pada uji coba ini setiap frame dibandingkan dengan seluruh citra acuan. Hasil deteksi dengan fitur SURF dapat dilihat pada Gambar 5. Terlihat *keypoint* yang sama dari kedua citra dihubungkan dengan garis. Pada bagian tengah citra hasil kamera omni sengaja dihilangkan. Hal ini dilakukan untuk menghilangkan bagian dari robot yang terlihat di tengah citra dan dapat mengganggu proses deteksi.

Untuk perhitungan jumlah *keypoint* yang sama dari frame video dan citra acuan disajikan dalam bentuk grafik di Gambar 6. Terlihat bahwa jumlah *keypoint* yang sama pada tiap *frame* video dan gambar acuan naik turun dan membentuk pola seperti segitiga. Saat robot mendekati salah satu *landmark* maka jumlah *keypoint* yang sama dengan *landmark* tersebut akan meningkat. Sebaliknya saat robot menjauhi *landmark* maka jumlah *keypoint* yang sama dengan *landmark* tersebut akan turun.

Saat robot berada diantara dua *landmark* jumlah *keypoint* keduanya bisa sama, maka yang dianggap lokasi robot adalah lokasi yang telah dideteksi sebelumnya hingga lokasi selanjutnya memiliki jumlah *keypoint* lebih banyak. Perubahan posisi robot ini dapat menyebabkan terjadinya beberapa kali perubahan deteksi lokasi yang berdekatan (osilasi) dalam proses deteksi posisi robot seperti terlihat pada Gambar 7. Osilasi ini dapat menjadi hambatan bahkan menyebabkan kesalahan dalam penentuan lokasi robot. Untuk meminimalkan hal ini dapat



Gambar 6. Grafik jumlah *keypoint* yang sama pada hasil ekstraksi fitur dengan SURF antara video dan citra acuan



Gambar 7. Grafik perubahan deteksi lokasi robot, terdapat osilasi (ditandai dengan garis putus-putus) deteksi saat posisi robot berada diantara dua landmark.

dilakukan dengan pemilihan nilai nilai *Hessian Threshold*. Untuk itu dilakukan uji coba guna mendapatkan nilai *Hessian Threshold* terbaik. Pencarian ini dilakukan menggunakan video yang sama dengan sebelumnya (di lokasi 13 hingga 17 dari lintasan). Nilai *Hessian Threshold* yang diuji coba mulai dari 370 hingga 410. Langkah yang sama juga dilakukan menggunakan metode SIFT untuk mengoptimalkan pemakaian metode ini. Hasil penghitungan perubahan dengan nilai *Hessian Threshold* yang berbeda ini disajikan dalam bentuk grafik di Gambar 8. Berdasarkan uji coba pada metode SURF terdapat dua nilai *Hessian Threshold* yang memiliki jumlah perubahan sama, yaitu 399 dan 400 dengan jumlah osilasi 26. Selanjutnya nilai *Hessian Threshold* yang dipilih adalah 400 karena memiliki rata-rata waktu deteksi yang lebih pendek yaitu 0,35 detik. Untuk *Hessian Threshold* 399 memiliki rata-rata waktu deteksi yang lebih pendek yaitu 0,36 detik. Sedang untuk metode SIFT, osilasi terkecil terdapat pada nilai *Hessian Threshold* 390, 395, dan 396 dengan jumlah osilasi 33. Dari ketiga nilai tersebut, rata-rata waktu terkecil ada di *Hessian Threshold* 390 yaitu 0,386 detik. *Hessian Threshold* 396 dan 396 memiliki rata-rata waktu terkecil 0,42 dan 0,391 detik. Maka untuk metode SIFT digunakan nilai *Hessian Threshold* 390.

Optimasi juga dilakukan dengan mencari nilai kedekatan fitur kedua citra untuk mendapatkan kesalahan deteksi terendah. Pengujian dilakukan dengan menelusuri seluruh area lintasan yang dimulai dari landmark 3 maju ke landmark 4, 5, 6 hingga berakhir di landmark 3 kembali dengan kondisi lintasan yang sepi tanpa gangguan. Jarak maksimal

citra yang diujikan mulai 0,4 hingga 0,5. Hasil setiap frame ( $\mathcal{F}$ ) dari video yang didapatkan dibandingkan dengan *ground truth* (GT). Pembuatan GT dilakukan secara manual dari setiap frame video berdasarkan posisi landmark yang telah ditetapkan. Untuk area yang berada diantara dua landmark maka label GT diberikan keduanya. Jika prediksi lokasi sesuai dengan GT maka prediksi benar, tetapi jika tidak sesuai maka dihitung sebagai kesalahan (*error*) yang didefinisikan sebagaimana persamaan (5). Selanjutnya dihitung nilai persentase *error* menggunakan persamaan (6).

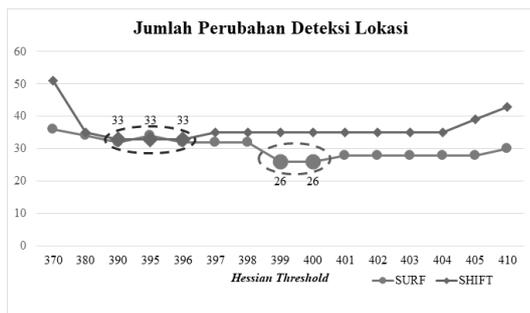
$$error = \begin{cases} 0, & \text{jika } \mathcal{F} = GT \\ 1, & \text{jika } \mathcal{F} \neq GT \end{cases} \quad (5)$$

$$E = \frac{\sum error}{F} \times 100\% \quad (6)$$

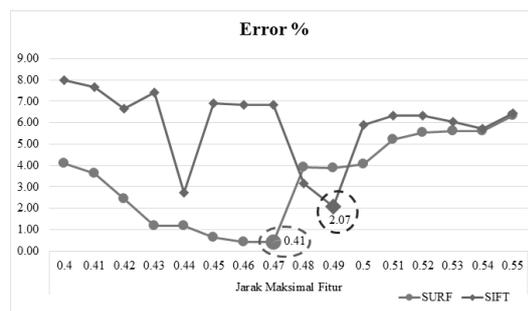
Di mana  $F$  adalah jumlah frame dari video. Hasil pengukuran error berdasarkan jarak fitur untuk metode SURF dan SIFT dapat dilihat dalam bentuk grafik di Gambar 9. Berdasarkan pengujian, untuk metode SURF nilai jarak maksimal terbaik untuk deteksi lokasi ini adalah 0,47 dengan nilai *error* terendah yaitu 0,41%. Sedangkan untuk metode SIFT jarak maksimal terbaik adalah 0,49 dengan nilai *error* 2,07.

Selanjutnya dengan menggunakan nilai *Hessian Threshold* dan jarak fitur maksimal yang telah didapatkan dilakukan beberapa kali pengujian dengan skenario sebagai berikut:

- a. Pengujian untuk pencarian nilai jarak maksimal fitur dijadikan sebagai skenario pertama. Dalam melakukan penelusuran, posisi robot dikondisikan seideal mungkin sehingga tidak banyak melakukan



Gambar 8. Jumlah perubahan deteksi lokasi dengan beberapa variasi nilai *Hessian Threshold*.



Gambar 9. Prosentase error berdasarkan jarak maksimal fitur.

- rotasi untuk mengubah posisi hadap robot. Rotasi hanya dilakukan saat arah lintasan berubah.
- b. Skenario kedua menjalankan robot mulai dari *landscape* 0 berjalan dan memutar ke *landscape* 1, 2, 3, dan seterusnya dan berakhir di *landscape* 3. Saat berjalan robot melakukan beberapa manuver berupa rotasi untuk mengubah arah hadap robot, dan jumlah manuver lebih banyak dari skenario 1.
  - c. Skenario ketiga, dengan rute yang sama dengan skenario kedua, namun pada skenario ini uji coba dilakukan dengan keadaan lintasan yang ramai dengan orang yang lalu lalang dan beberapa rotasi arah robot. Hal ini juga bertujuan untuk menguji apakah pengenalan lokasi masih dapat dilakukan dengan adanya gangguan.
  - d. Skenario keempat, robot berjalan berputar dengan arah sebaliknya. Perjalanannya mulai dari *landscape* 3 berjalan ke *landscape* 2, 1, 0, 27, 26, dan seterusnya berputar hingga berakhir di *landscape* 1. Dan sama seperti pada skenario ketiga, uji coba dilakukan pada lintasan yang ramai dengan orang yang lalu lalang.

Pengujian untuk kinerja lokalisasi diri diukur secara kuantitatif dengan lima jenis pengukuran: *recall*, presisi, akurasi, *F-measure*, dan *G-measure*. Keseluruhan pengukuran didasarkan pada matrik *confusion*. *Recall*, presisi, dan *F-measure* digunakan untuk pengukuran hasil deteksi, klasifikasi, maupun segmentasi berdasarkan GT (Gangwar, Joshi, Singh, Alonso-Fernandez, & Bigun, 2016). Akurasi, *F-measure*, dan *G-measure* digunakan untuk pengukuran keberhasilan deteksi dan klasifikasi citra oleh Kim (Kim & Lattimer, 2015). *Recall* merupakan ukuran keberhasilan sistem dalam memprediksi lokasi benar dari seluruh data lokasi yang benar, yang didefinisikan sebagai persamaan (7):

$$R = \frac{TP}{TP + FN} \tag{7}$$

Dimana *TP* (*True Positive*) adalah jumlah lokasi benar dan diprediksi benar, dan *FN* (*False Negative*) adalah jumlah lokasi benar dan diprediksi salah.

Presisi adalah ukuran keberhasilan sistem dalam memprediksi lokasi benar terhadap seluruh data lokasi yang diprediksi benar, yang didefinisikan sebagai persamaan (8):

$$P = \frac{TP}{TP + FP} \tag{8}$$

Dimana *FP* (*False Positive*) adalah jumlah lokasi salah dan diprediksi benar. Akurasi adalah tingkat keberhasilan prediksi lokasi terhadap seluruh data dan didefinisikan sebagai persamaan (9):

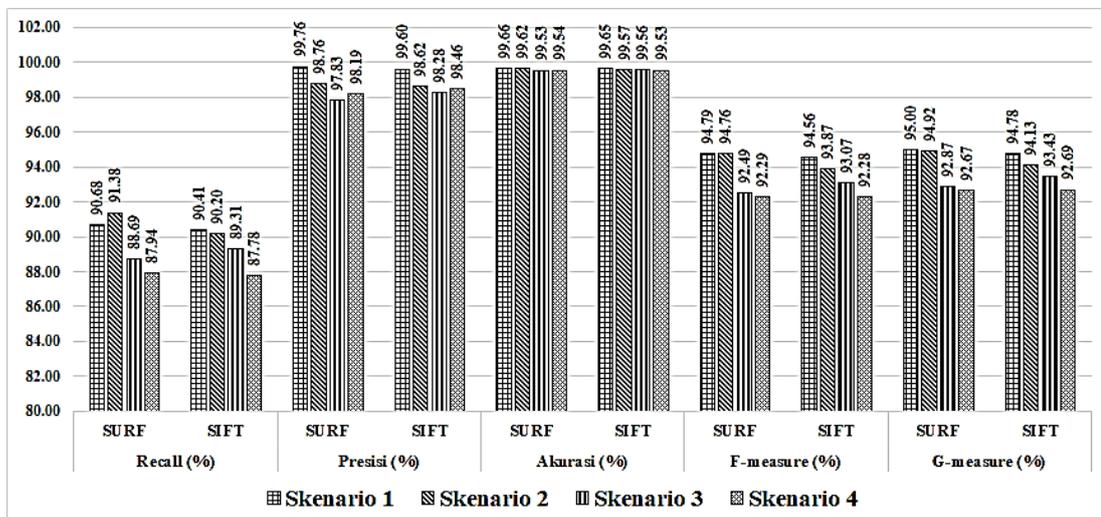
$$A = \frac{TP + TN}{TP + TN + FP + FN} \tag{9}$$

Dimana *TN* (*True Negative*) adalah jumlah lokasi salah dan diprediksi salah. *F-measure* dan *G-measure* adalah ukuran rata-rata harmonik dan rata-rata geometri dari *recall* dan presisi (Powers, 2011) yang didefinisikan sebagai (10) dan (11):

$$FM = \frac{2 \times R \times P}{R + P} \tag{10}$$

$$GM = \sqrt{R \times P} \tag{11}$$

Hasil pengujian dilakukan pada seluruh skenario pada fitur SURF dan SIFT disajikan dalam bentuk grafik pada Gambar 10. Pada pengukuran *recall*, metode SIFT memiliki nilai lebih tinggi pada skenario 3 dengan nilai 89,31%, sedangkan pada skenario 1, 2, dan 4 SURF lebih unggul dengan nilai 90,68%, 91,38%, dan 87,94%. Untuk pengukuran presisi, SURF unggul di skenario 1 dengan nilai 99,76% dan skenario 2 dengan nilai 98,76%, sedangkan SIFT unggul di skenario 3 dengan nilai 98,28% dan skenario 4 dengan nilai 98,46%. Persaingan yang ketat terjadi pada pengukuran akurasi, dimana SURF unggul tipis di skenario 1, 2,



Gambar 10. Hasil pengujian deteksi lokasi

Tabel 1. Nilai rata-rata hasil pengujian lokalisasi diri robot, nilai yang tercetak tebal adalah nilai terbaik.

Pengujian		Nilai Rata-rata
Recall (%)	SURF	89,67
	SIFT	89,42
Presisi (%)	SURF	98,63
	SIFT	<b>98,74</b>
Akurasi (%)	SURF	<b>99,59</b>
	SIFT	99,58
F-measure (%)	SURF	<b>93,58</b>
	SIFT	93,44
G-measure (%)	SURF	<b>93,87</b>
	SIFT	93,76
Waktu deteksi/frame (dt)	SURF	<b>0,365</b>
	SIFT	0,371

dan 4 dengan nilai 99,66%, 99,62%, dan 99,54%. Sedangkan di skenario 3 SIFT unggul tipis dengan nilai 99,56%. Selanjutnya pada pengukuran *F-measure* SURF unggul di skenario 1 dengan nilai 94,76%, skenario 2 dengan nilai 94,76%, dan di skenario 4 dengan nilai 92,29%. Pada pengukuran ini SIFT unggul di skenario 3 dengan nilai 93,07%. Pada pengujian terakhir yaitu dengan *G-measure*, SURF unggul di skenario 1 dengan nilai 95,00% dan skenario 2 dengan nilai 94,92%, sedangkan SIFT unggul di skenario 3 dengan nilai 93,43% dan skenario 4 dengan nilai 92,69%.

Untuk membandingkan performa dari metoda SURF dan SIFT, nilai rata-rata pengujian dari grafik Gambar 10 ditambah pengujian waktu deteksi disajikan pada Tabel 1. Meskipun dengan selisih yang kecil namun metode SURF unggul di 5 dari 6 pengujian. SURF unggul pada pengujian recall dengan nilai 89,67%, akurasi dengan nilai 99,59%, *F-measure* dengan nilai 93,58%, dan *G-measure* dengan nilai 93,87%. Metode SIFT hanya unggul di pengujian presisi dengan nilai 98,74%.

Berdasarkan grafik pengujian Gambar 10 dapat kita telaah bahwa SURF unggul di seluruh pengujian untuk skenario 1 dan 2. Untuk SIFT unggul di seluruh pengujian skenario 3. Sedangkan pada skenario 4, SURF unggul di pengujian *recall*, akurasi dan *F-measure*, dan SIFT unggul di pengujian presisi dan

*G-measure*. Analisa ini memberikan gambaran bahwa SURF secara umum lebih unggul namun untuk kondisi lintasan yang ramai seperti skenario 3 metode SIFT lebih unggul. Pada kondisi ramai dapat menyebabkan citra lingkungan yang ditangkap oleh kamera menjadi tidak lengkap. Selain itu robot juga harus lebih banyak ber-manuver yang mengakibatkan citra yang ditangkap menjadi kabur. Hal ini dapat mengakibatkan kesalahan deteksi lokasi seperti yang disajikan pada Gambar 11.

Pada skenario 4, kondisi lintasan tidak jauh berbeda dengan skenario 3 yaitu ramai dengan orang lalu-lalang. Namun pada skenario 4 robot dijalankan dengan arah sebaliknya sehingga citra yang ditangkap oleh kamera omni akan akan terbalik posisinya dengan citra acuan. Terbalik yang dimaksud adalah posisi kiri dan kanan serta posisi depan dan belakang. Pada skenario ini SURF unggul pada tiga pengukuran dan SIFT unggul di dua pengukuran. Jika kita bandingkan nilai-nilai pengujian pada skenario 3 dan 4 yang memiliki kondisi hampir sama tapi dengan arah yang berbeda, hasil pengujian pada metode SIFT lebih banyak mengalami penurunan. Terdapat empat pengujian yang nilainya lebih kecil yaitu pada *recall*, akurasi, *F-measure*, dan *G-measure*. Hanya satu pengujian yang mengalami peningkatan yaitu Presisi. Pada metode SURF penurunan terjadi pada *recall*, *F-measure*, dan *G-measure*, sedangkan pada pengujian presisi dan akurasi mengalami peningkatan.

Jika digabungkan analisa berdasarkan Gambar 10 dan Tabel 1 maka SIFT memiliki ketahanan dalam mendeteksi lokasi dengan gangguan. Meskipun masih kalah unggul dengan SURF pada skenario 4 namun nilai Presisi SIFT secara umum lebih tinggi. Meskipun demikian secara keseluruhan SURF lebih unggul dibandingkan SIFT. Dan yang tak kalah penting adalah pengujian waktu deteksi. Nilai rata-rata waktu deteksi satu frame menggunakan SURF lebih cepat 6 mili detik dibandingkan SIFT. Dimana waktu deteksi setiap frame untuk SURF adalah 0,365 detik dan SIFT 0,371 detik.

Gambar 11. Citra *blurr* menyebabkan (a) lokasi tidak terdeteksi, atau (b) prediksi salah

Tabel 2. Hasil pengujian waktu deteksi.

Skenario	Waktu Deteksi (detik)			
	SIFT		SURF	
	Reguler	Optimasi	Reguler	Optimasi
1	3,233	0,350	3,257	0,339
2	2,951	0,364	3,056	0,381
3	3,422	0,376	3,148	0,373
4	3,186	0,394	3,251	0,365
<b>Rata-rata</b>	<b>3,198</b>	<b>0,371</b>	<b>3,178</b>	<b>0,365</b>

Pengujian juga dilakukan terhadap peningkatan waktu deteksi hasil dari implementasi optimasi deteksi lokasi dapat. Pengujian dilakukan dengan membandingkan rata-rata waktu deteksi tanpa optimasi dan dengan optimasi. Hasil pengujian disajikan pada Tabel 2. Berdasarkan pengujian terdapat peningkatan yaitu waktu yang diperlukan untuk deteksi menjadi lebih singkat. Pada metode SIFT terjadi peningkatan 8,62 kali, dimana untuk deteksi tanpa optimasi (reguler) rata-rata waktu deteksi yang diperlukan adalah 3,198 detik/frame dan dengan optimasi menjadi 0,371 detik/frame. Pada metode SURF, waktu deteksi yang diperlukan, untuk reguler 3,178 detik/frame dan dengan optimasi 0,365 detik/frame. Hal ini berarti pada metode SURF dengan optimasi deteksi terjadi peningkatan kecepatan deteksi 8,72 kali.

Rata-rata waktu deteksi 0,365 detik/frame adalah perhitungan waktu rata-rata untuk deteksi berdasarkan minimal 3 citra acuan. Jumlah citra acuan dapat bertambah saat terjadi kesalahan deteksi. Dimana saat terjadi kesalahan seluruh citra acuan yang berjumlah 28 digunakan. Jika diperhitungkan waktu yang diperlukan untuk deteksi dengan satu citra acuan maka waktu deteksi rata-rata yang didapatkan tersebut dibagi dengan 3 dan didapatkan hasil 0,122 detik/citra. Waktu deteksi ini sudah termasuk waktu untuk ekstraksi fitur dan menghitung kesamaan fitur. Jika dibandingkan dengan penelitian yang dilakukan Valgren (Valgren & Lilienthal, 2010), menyatakan bahwa waktu yang diperlukan untuk lokalisasi dengan SURF adalah 0,25 detik, dan dengan SIFT bervariasi antara 0,5 hingga 2 detik. Sedangkan Qu(Qu et al., 2016), untuk catatan waktu terkecil yang disajikan dalam bentuk grafik adalah antara 0,2 hingga 0,25 detik

#### 4. KESIMPULAN

Berdasarkan hasil pengujian lokalisasi diri robot di area *indoor* menggunakan serta dilakukan optimasi deteksi dan waktu deteksi berhasil dengan beberapa capaian. Capaian pertama adalah optimasi akurasi deteksi dengan pemilihan nilai *Hessian Threshold* dan jarak maksimal fitur yang digunakan. Dari optimasi pada SURF dan SIFT ini secara umum SURF memberikan hasil lebih baik. Dengan nilai *Hessian Threshold* 400 dan jarak fitur maksimal 0,47 didapatkan nilai rata-rata hasil pengujian *recall*

89,67%, presisi 98,63%, akurasi 99,59%, *F-measure* 93,58%, dan *G-measure* 93,87%. Hal ini menjadi alasan digunakan SURF untuk ekstraksi fitur deteksi lokasi berdasarkan data citra. Capaian kedua yang juga menjadi kontribusi dari penelitian ini adalah optimasi waktu deteksi. Optimasi deteksi berdasarkan citra acuan terbukti dapat memperpendek waktu deteksi dari 3,178 detik/frame tanpa optimasi menjadi 0,365 detik/frame dengan optimasi. Hal ini berarti terdapat peningkatan kecepatan deteksi 8,72 kali. Sedangkan untuk perhitungan deteksi lokasi berdasarkan citra acuan didapatkan waktu deteksi 0,122 detik/citra. Waktu ini lebih cepat jika dibandingkan dengan penelitian yang dilakukan Qu dan Valgren.

Hasil yang telah didapatkan ini masih mungkin ditingkatkan. Rencana penelitian selanjutnya adalah mengembangkan metode deteksi menggunakan *Deep Learning*. Dan apa yang telah didapatkan dalam penelitian ini menjadi dasar untuk penelitian lanjutan tersebut.

#### UCAPAN TERIMA KASIH

Ucapan terima kasih penulis sampaikan kepada Universitas Dinamika, atas dukungan dana pada penelitian ini melalui program Hibah Penelitian Internal tahun anggaran 2019 dari bagian Penelitian dan Pengabdian kepada Masyarakat.

#### DAFTAR PUSTAKA

- BAY, H., ESS, A., TUYTELAARS, T., & VAN GOOL, L. 2008. Speeded-Up Robust Features (SURF). *Computer Vision and Image Understanding*, 110(3), 346–359. <<https://doi.org/10.1016/J.CVIU.2007.09.014>>
- BROWN, M., & LOWE, D. 2002. Invariant Features from Interest Point Groups. *Proceedings of the British Machine Vision Conference 2002*, 23.1-23.10. <<https://doi.org/10.5244/C.16.23>>
- DAYOUB, F., CIELNIAK, G., & DUCKETT, T. 2011. A sparse hybrid map for vision-guided mobile robots. In *5th European Conference on Mobile Robots (ECMR 2011)*. Retrieved from <<http://eprints.qut.edu.au/72973/>>
- FENG, H. M., CHEN, C. Y., & HORNG, J. H. 2010. Intelligent omni-directional vision-based mobile robot fuzzy systems design and implementation. *Expert Systems with Applications*, 37(5), 4009–4019. <<https://doi.org/10.1016/j.eswa.2009.11.030>>
- GANGWAR, A., JOSHI, A., SINGH, A., ALONSO-FERNANDEZ, F., & BIGUN, J. 2016. IrisSeg: A fast and robust iris segmentation framework for non-ideal iris images. *2016 International Conference on Biometrics, ICB 2016*. <https://doi.org/10.1109/ICB.2016.7550096>
- HARIANTO, H., RIVAI, M., & PURWANTO, D. 2013. Implementation of Electronic Nose in

- Omni-directional Robot. *International Journal of Electrical and Computer Engineering (IJECE)*, 3(3). <<https://doi.org/10.11591/ijece.v3i3.2531>>
- HIBINO, F., YUTA, TAKAHASHI, Y., & MAEDA, Y. 2012. Self-localization based on image features of omni-directional image. 6th International Conference on Soft Computing and Intelligent Systems, and 13th International Symposium on Advanced Intelligence Systems, SCIS/ISIS 2012, 2040–2043. <<https://doi.org/10.1109/SCIS-ISIS.2012.6505325>>
- KIM, J.-H., & LATTIMER, B. Y. 2015. Real-time probabilistic classification of fire and smoke using thermal imagery for intelligent firefighting robot. *Fire Safety Journal*, 72, 40–49. <<https://doi.org/10.1016/J.FIRESAF.2015.02.007>>
- KUSWADI, S., O, A. N. G., TAMARA, M. N., & S, I. A. 2018. Optimasi Sistem Navigasi Robot Bencana Dengan Algoritma Bug Dan Jaringan Syaraf Tiruan Optimisation of Disaster Robot Navigation Systems Using Bug. *Jurnal Teknologi Informasi Dan Ilmu Komputer (JTIK)*, 5(5), 635–642. <<https://doi.org/10.25126/jtiik2018551153>>
- LI, MENG, XIE, ZHANG, HUANG, JIANG, & LIU. 2019. Reliable and Fast Localization in Ambiguous Environments Using Ambiguity Grid Map. *Sensors*, 19(15), 3331. <<https://doi.org/10.3390/s19153331>>
- PIASCO, N., SIDIBÉ, D., DEMONCEAUX, C., & GOUET-BRUNET, V. 2018. A survey on Visual-Based Localization: On the benefit of heterogeneous data. *Pattern Recognition*, 74, 90–109. <<https://doi.org/10.1016/j.patcog.2017.09.013>>
- POWERS, D. M. W. 2011. Technical Report SIE-07-001: Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness & Correlation. *Journal of Machine Learning Technologies*, 2(1), 37–63.
- QU, X., SOHEILIAN, B., HABETS, E., & PAPANODITIS, N. 2016. Evaluation Of Sift And Surf For Vision Based Localization. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 685–692. <<https://doi.org/10.5194/isprsarchives-XLI-B3-685-2016>>
- RÖHRIG, C., HESS, D., KIRSCH, C., & KÜNEMUND, F. 2010. Localization of an omnidirectional transport robot using IEEE 802.15.4a ranging and laser range finder. 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, 3798–3803. <<https://doi.org/10.1109/IROS.2010.5651981>>
- SU, Z., ZHOU, X., CHENG, T., ZHANG, H., XU, B., & CHEN, W. 2017. Global localization of a mobile robot using lidar and visual features. 2017 IEEE International Conference on Robotics and Biomimetics (ROBIO), 2377–2383. <<https://doi.org/10.1109/ROBIO.2017.8324775>>
- THEODORIDIS, T., & HU, H. 2012. Toward intelligent security robots: A survey. *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, 42(6), 1219–1230. <<https://doi.org/10.1109/TSMCC.2012.2198055>>
- VALGREN, C., & LILIENTHAL, A. J. 2010. SIFT, SURF & seasons: Appearance-based long-term localization in outdoor environments. *Robotics and Autonomous Systems*, 58(2), 149–156. <<https://doi.org/10.1016/J.ROBOT.2009.09.010>>
- WAJANSYAH, A., SUPRIADI, S., NUR, S., & BRAMANTO W.P., A. 2018. Implementasi Fuzzy Logic Pada Robot Line Follower. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 5(4), 395. <<https://doi.org/10.25126/jtiik.201854747>>
- YANG, S., SCHERER, S. A., YI, X., & ZELL, A. 2017. Multi-camera visual SLAM for autonomous navigation of micro aerial vehicles. *Robotics and Autonomous Systems*, 93, 116–134. <<https://doi.org/10.1016/j.robot.2017.03.018>>