

Realisasi Prediksi Jalur Visual Berdasarkan Citra Tunggal dengan *Spatial Matching Network*

Mohamad Hafiz Gema Takbir Akbar¹, Riko Arlando Saragih²
Fakultas Teknik, Program Studi Teknik Elektro, Universitas Kristen Maranatha
Jl. Surya Sumantri No.65, Bandung 40164, Indonesia
hafizgta@icloud.com¹, riko.as@eng.maranatha.edu²

Diterima : 22 Agustus 2022

Disetujui : 01 Oktober 2022

Abstract—Jalur visual dapat didefinisikan sebagai cara manusia menafsir secara visual terhadap sebuah lintasan yang dapat dilalui. Prediksi jalur visual adalah suatu bidang riset kecerdasan buatan untuk memprediksi dan menafsirkan situasi dalam suatu *scene* yang meniru kemampuan berfikir manusia. Pada tahun 2016, sebuah jaringan syaraf tiruan berbasis *Convolutional Neural Network* (CNN) bernama *Spatial Matching Network* (SMN) dibangun untuk memprediksi jalur visual dari satu citra berdasarkan analisis spasial. Fungsi dari SMN adalah untuk mencari kesesuaian konteks spasial dari objek observasi dalam sebuah citra dengan lingkungannya. Dalam tulisan ini, prediksi jalur visual menggunakan SMN dikembangkan berdasarkan 4 jenis percobaan yang melibatkan orientasi dari objek dan perubahan jumlah node dari *directed graph* yang merepresentasikan hubungan pada tiap posisi pada citra. Kontribusi penelitian ini adalah kenaikan akurasi dari rata-rata hasil prediksi jalur visual menggunakan evaluasi metrik *Modified Hausdorff Distance*, dengan rincian kenaikan akurasi hasil 0,04% dari percobaan dengan penambahan jumlah node pada *directed graph*, kenaikan sebesar 14,9% dari percobaan dengan penambahan orientasi obyek, dan kenaikan akurasi hasil sebesar 20,5 % pada percobaan dengan penambahan kedua parameter tersebut.

Keywords— *Spatial Matching Network*, *Convolutional Neural Network*, *Fully Connected Layer*, jalur visual, analisis spasial, orientasi obyek, node, *directed graph*

I. PENDAHULUAN

Secara hurufiah, *pathway* atau jalur artinya adalah lintasan yang dapat dilalui oleh suatu objek [1]. Sedangkan visual berarti hal yang berkaitan dengan penglihatan manusia [2]. Jadi, arti dari jalur visual adalah tafsir penglihatan manusia terhadap sebuah lintasan yang dapat dilalui. Kemampuan untuk memprediksi dan memastikan suatu *scene* dalam memprediksi jalur visual merupakan salah satu tujuan utama dalam membangun sistem yang meniru kemampuan berfikir manusia [3], khususnya dalam perkembangan ilmu mengenai kecerdasan buatan.

Teknologi prediksi jalur visual membutuhkan banyak data untuk citra latih, sehingga sistem tersebut akan semakin rumit

apabila data yang akan diobservasi jumlahnya terbatas. Misalkan sebuah *scene* harus diprediksi berdasarkan satu buah citra saja. Untuk kondisi ini, maka prediksi dan pemastian jalur visual akan lebih sulit [4].

Berbeda dengan mesin, manusia dapat memprediksi suatu situasi dengan mudah. Manusia dibekali kemampuan intuitif untuk mengerti situasi secara simultan. Mesin memerlukan *set instruction* yang jelas sebelum menafsirkan suatu situasi [4]. Dari fakta tersebut, kesulitan lain dari merancang sistem prediksi jalur visual adalah bagaimana merancang sistem yang dapat menghadapi situasi yang tak menentu [5].

Dalam beberapa penelitian yang sudah ada, secara umum terdapat 2 metode dalam sistem

prediksi dan pemastian jalur, yaitu *matching-based methods* [6]–[8] dan *learning-based methods* [5], [9]. *Matching-based methods* hanya mengambil informasi dari *database* ke *queries* tanpa membangun model pemastian jalur. Sementara *learning-based methods*, selain dapat membangun model pemastian jalur, metode ini lebih baik dalam membedakan dan menggeneralisasi situasi, sehingga memiliki kemampuan yang lebih baik dalam beradaptasi untuk situasi yang rumit [5]. Di dalam tulisan ini digunakan *learning based-methods* untuk memprediksi jalur visual.

Spatial Matching Network (SMN) merupakan salah satu jaringan syaraf tiruan berbasis *Convolutional Neural Network* (CNN) yang digunakan pada penelitian dari Siyu Huang dan kawan-kawan untuk memprediksi jalur visual dari satu citra. Fungsi dari SMN adalah untuk mencari kesesuaian konteks spasial dari objek observasi dalam sebuah citra. Dalam tulisan ini akan dijelaskan realisasi sebuah perangkat lunak untuk memprediksi dan memastikan jalur visual dari citra tunggal berdasarkan penelitian tersebut. Hasil utama yang diperoleh dalam penelitian ini adalah *unified deep learning framework* yang dibangun dengan menggunakan *deep feature learning* untuk menghasilkan *representasi visual* dan model *spatio-temporal* dalam bentuk SMN.

II. TEORI PENDUKUNG

Spatio-temporal analysis merupakan *spatial analysis* dalam konteks temporal [10]. *Spatial analysis* adalah sebuah jenis dari analisis geografi yang bertujuan untuk menjelaskan pola dari perilaku dan ekspresi manusia terhadap kondisi spasial, konteks temporal pada spatial analysis adalah analisis spatial yang menyatakan relativitas waktu ketika konteks spasial tersebut diobservasi.

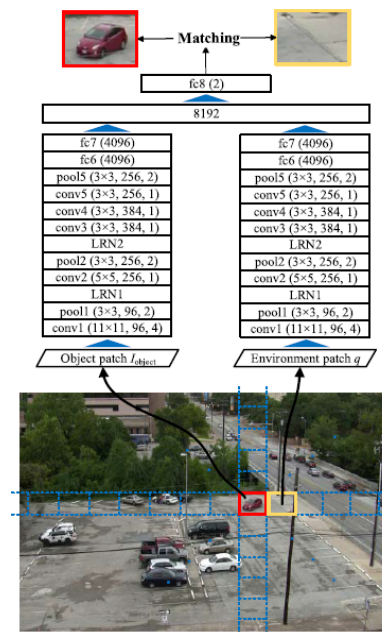
A. *Spatial Matching Network* (SMN)

SMN merupakan sebuah jaringan syaraf tiruan berbasis *Convolutional Neural Network* yang dikembangkan oleh Siyu Huang dan kawan-

kawan [4] yang bertujuan untuk menganalisis kesesuaian spasial dari objek observasi dengan lingkungannya dalam sebuah citra tunggal. Visualisasi dari cara kerja jaringan syaraf tiruan ini dapat dilihat pada Gambar 1.

Cara kerja dari SMN adalah sebagai berikut: sebuah *bounding box* dibangun untuk memisahkan objek observasi dari lingkungannya, lalu potongan citra I_{obyek} tersebut bersama potongan citra lingkungan q yang didapat dari *overlapped sliding window bounding box* diinputkan ke dalam masing-masing dari 2 alur dari jaringan syaraf tiruan SMN, hingga akhirnya menghasilkan 2 buah klasifikasi yang menyatakan kesesuaian dari konteks spasial kedua potongan citra tersebut.

Arsitektur dari SMN memanfaatkan arsitektur dari AlexNet [11] yang disesuaikan dengan kebutuhan analisis spasial. 2 buah input diteruskan kedalam 2 alur jaringan yang merupakan layer 1 hingga 7 dari AlexNet, sebelum akhirnya digabungkan menggunakan layer penggabungan (*concatenation layer*) dan dilanjutkan ke sebuah *fully connected layer* yang akan mengklasifikasikan hasil dari kedua citra tersebut menjadi 2 label klasifikasi, $L_s \in \{0, 1\}$. Bila kedua input dari citra berkesesuaian secara spasial maka Label L_s adalah 1.



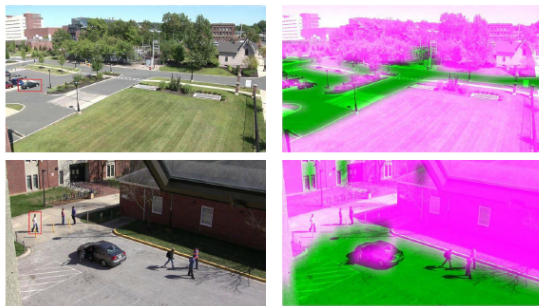
Gambar 1. Visualisasi dari jaringan SMN

Bila tidak, L_s akan di-set menjadi 0. Label-label tersebut merupakan parameter kemungkinan (*likelihood*) dari kesesuaian spasial dari I_{obyek} dan potongan citra q . Dari nilai r yang telah didapatkan dari SMN, yaitu sebuah *reward map*, dibentuk untuk memetakan kesesuaian konteks spasial dari keseluruhan bagian dari lingkungan dari I_{obyek} , yang diformulasikan seperti berikut:

$$R_{reward}(s_i) = FS(I_{object}, q_{s_i}) \psi(s) \quad (1)$$

F_s adalah variabel *forward propagation* dari *spatial matching network* dan ψ_s adalah parameter yang telah dipelajari (*learned parameters*) dari jaringan tersebut. $R_{reward}(s) \in [0, 1]$ adalah nilai *reward* r untuk semua posisi s_i .

Bila $R_{reward} = 1$, maka kemungkinan objek akan mencapai posisi tersebut di masa depan amat besar. Dan sebaliknya, bila $R_{reward} = 0$, kemungkinan objek tersebut akan berada di posisi tersebut amat kecil. Dapat dilihat pada Gambar 2, bagian dari citra yang memiliki $R_{reward} = 1$ ditandai oleh warna hijau, sementara bagian yang menggunakan warna merah muda merupakan bagian dari citra yang memiliki $R_{reward} = 0$.



Gambar 2. Visualisasi dari *reward map* [4]

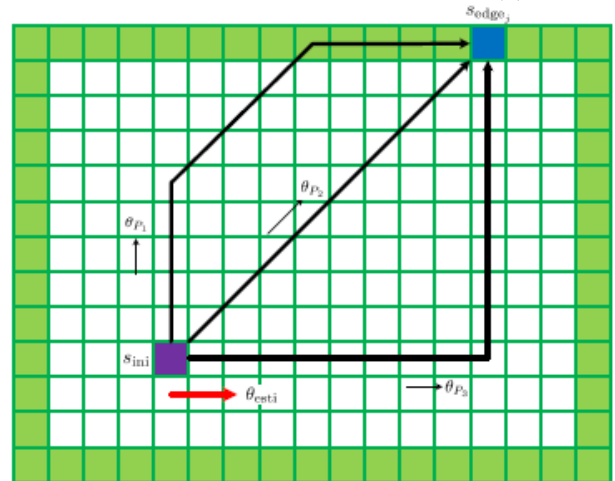
B. Proses Pencarian Jalur Visual

Pada percobaan yang dilakukan oleh Siyu Huang dan kawan-kawan [4], proses pencarian jalur visual dari citra tunggal menggunakan pendekatan optimasi, Tujuan dari persoalan optimasi ini adalah untuk memastikan jalur (P) yang paling memungkinkan $P=(s_1, s_2, s_3, \dots, s_n)$ dari objek pada *bounding box* $B = (b_1, b_2, w, h)$ di

waktu yang akan datang, dengan b_1 dan b_2 merupakan koordinat pojok kiri atas dari citra masukan, w dan h adalah lebar dan tinggi dari B , $s=(x,y)$ merupakan koordinat dari sebuah posisi dari gambar, dan P adalah kumpulan koordinat posisi yang berurut dan berdekatan.

Sebelum menghitung persoalan optimasi dari jalur, citra masukan terlebih dahulu diilustrasikan sebagai grafik *grid*, dimana tiap *grid* mewakili posisi dari *scene* tertentu pada citra tersebut (s_i), sesuai dengan yang ditampilkan dalam Gambar 3. Untuk mencari jalur yang paling sesuai dari jalur-jalur yang memungkinkan, persoalan optimasi digunakan untuk mencari jalur P dengan nilai *Cost* (C) paling kecil:

$$\begin{aligned} & \min_P C(P) \\ & \text{s. t. } P \in P \end{aligned} \quad (2)$$



Gambar 3. Ilustrasi dari cara kerja prediksi jalur visual [4]

Untuk memformulasikan nilai *cost* dari jalur, secara intuitif didapatkan: semakin sedikit rintangan yang menghalangi jalur, maka nilai *cost*-nya akan semakin kecil. Hal ini diformulasikan seperti berikut:

$$C_s(P) = \sum_{s_i \in P} R_{cost}(s_i) \quad (3)$$

Dengan R_{cost} adalah nilai *cost map* yang merupakan nilai yang mewakili *cost* dari sebuah posisi dari citra secara konteks spasial. C_s adalah *spatial matching cost*, Nilai ini merepresentasikan nilai *cost* dari rintangan berdasarkan model spasial dari citra. R_{cost} diformulasikan seperti berikut:

$$R_{cost}(s) = \frac{1}{1 + e^{-\alpha(R_{reward}(s)-\gamma)}} \quad (4)$$

Dimana α merupakan toleransi dari rintangan, dan γ adalah setengah dari nilai maksimum dari R_{reward} , yaitu 0.5. Dari nilai R_{cost} tersebut, *spatial matching cost* (C_s) dapat dihitung berdasarkan jumlah R_{cost} yang dilalui oleh obyek dalam lingkup P , untuk kemudian disubstitusikan ke dalam Persamaan (4).

Untuk mendapatkan nilai *cost* yang memiliki kesesuaian konteks spatio-temporal dari I_{obyek} , nilai *cost* dari jalur diformulasikan sebagai berikut:

$$\mathcal{C}(P) = \min_p \sum_{s_i \in P} R_{cost}(s_i) + \varepsilon D(\theta_p, \theta_{GT}) \quad (5)$$

s.t. s_i dan s_{i+1} adalah tetangga secara spasial, dengan $s_i = s_{ini}$ dan $s_{i+1} = s_{edge}$ ($j = 1, 2, 3, \dots, m$)

Definisi dari kedekatan spasial adalah sebuah posisi pada citra yang paling dekat dari I_{obyek} dalam jarak tertentu d , yang mematuhi batasan posisi, yaitu atas, bawah, kiri, kanan, kanan atas, kiri atas, kanan atas, dan kiri atas dari objek observasi. Sementara, $D(\theta_p, \theta_{GT})$ adalah nilai *Euclidean Distance* dari orientasi objek θ_{GT} dan orientasi dari jalur θ_p , dengan nilai $\theta \in (-\pi, +\pi]$. Nilai orientasi objek θ_{GT} dalam penelitian ini di-inputkan secara manual pada masing-masing objek observasi, dan nilai ε adalah *trade-off coefficient* yang berfungsi untuk mengatur signifikansi dari orientasi pada kalkulasi jalur.

C. Algoritma Dijkstra

Directed graph merupakan matriks yang menyatakan hubungan (*edges*) antar satu node (*vertices*) dengan node lainnya beserta nilai beban (*weights*) pada masing-masing hubungan tersebut [16]. Hasil dari *directed graph* yang telah dibangun kemudian dapat digunakan untuk mencari suatu permasalahan jalur terpendek. salah satunya menggunakan algoritma Dijkstra [17].

Algoritma Dijkstra adalah sebuah algoritma yang digunakan untuk mencari jalur terpendek di antara node dalam sebuah *graph* [13]. Cara kerja dari algoritma Dijkstra adalah, berdasarkan sekumpulan node di dalam sebuah *graph*, seluruh jalur yang memungkinkan dari sekumpulan node tersebut ditelusuri hingga jalur

tersebut mencapai node destinasi yang telah ditentukan. Algoritma Dijkstra menggunakan label yang merupakan angka riil yang berurutan. Urutan-urutan tersebut bisa digeneralisasikan untuk menggunakan label tertentu, misal, diurutkan secara *ascending*, *descending* atau urutan lain yang telah ditentukan.

D. AlexNet

AlexNet adalah Jaringan syaraf tiruan berbasis *Convolutional Neural Network* yang dikembangkan oleh Alex Krizhevsky dan kawan-kawan dalam partisipasi mereka pada kompetisi *ImageNet Large Scale Visual recognition Challenge* (ILSVRC) pada tahun 2012. Tujuan dari kompetisi tersebut adalah mencari arsitektur terbaik untuk mengklasifikasikan 1000 citra dari masing-masing 1000 kategori dari dataset ImageNet. Pada kompetisi tersebut, tim SuperVision yang dipunggawai oleh Alex Krizhevsky dan kawan-kawan berhasil membangun arsitektur jaringan syaraf tiruan yang memiliki *error rate* yang paling kecil dari tim lainnya [11–15].

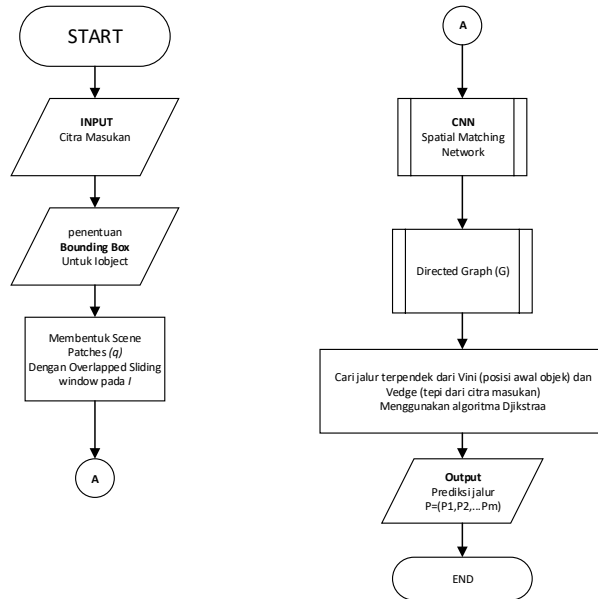
III. METODOLOGI PENELITIAN

Dalam bagian ini akan dijelaskan perancangan sistem yang direalisasikan dalam penelitian ini, yaitu proses pembentukan *bounding box* (meliputi pembentukan I_{obyek} dan pemisahan potongan citra lingkungan di sekitar I_{obyek}), realisasi perancangan *spatial matching network* yang dilakukan, serta proses pencarian jalur visual terpendek. Selain itu, terdapat 4 jenis percobaan dalam prediksi jalur visual yang dilakukan :

Prediksi jalur visual tanpa memperhitungkan orientasi objek observasi dengan jarak antar node sebesar jarak diagonal dari *bounding box* objek
 Prediksi jalur visual tanpa memperhitungkan orientasi objek observasi dengan jarak antar node sebesar setengah dari jarak diagonal *bounding box* dari I_{obyek} .
 Prediksi jalur visual dengan memperhitungkan orientasi objek observasi dengan jarak antar node sebesar jarak diagonal dari *bounding box* objek

1. Prediksi jalur visual dengan memperhitungkan orientasi objek observasi dengan jarak antar node sebesar setengah dari panjang diagonal *bounding box* dari I_{objek}

Diagram alir dari percobaan-percobaan tersebut ditampilkan pada Gambar 4. Rincian percobaan dalam penelitian ini adalah:



Gambar 4. Diagram alir rancangan percobaan yang direalisasikan dalam penelitian ini

1. Percobaan menggunakan 4 scene *Virat Video Dataset Release 2.0* yang telah sesuai dengan batasan masalah dari penelitian ini.
2. Atas keterbatasan data yang dapat diperoleh dari dataset tersebut, jumlah citra yang digunakan dalam 1 jenis scene memiliki jumlah yang berbeda-beda, dengan total 15 citra yang memiliki masing-masing memiliki 1 objek observasi dengan rincian:
 - a) Scene 1: menggunakan 5 citra
 - b) Scene 2: menggunakan 2 citra
 - c) Scene 3: menggunakan 4 citra
 - d) Scene 4: menggunakan 4 citra

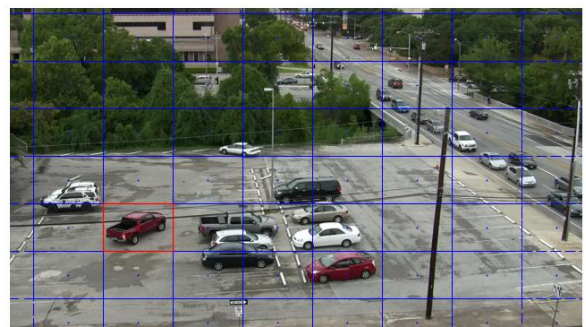
A. Membangun *bounding box*

Bounding box yang digunakan dalam penelitian ini didapat menggunakan sintaks *ginput* pada program matlab. Sintaks *ginput*

tersebut digunakan untuk mendapatkan koordinat piksel dari dari titik paling kiri atas objek observasi, dan titik paling kanan bawah dari objek.

Titik-titik yang telah didapatkan diolah menjadi *bounding box* ($B=[b_1, b_2, w, h]$) yang memisahkan dan menandai objek dari lingkungan sekitarnya, b_1 dan b_2 merupakan koordinat pixel dari titik paling kiri atas dari *bounding box*, w adalah lebar dari *bounding box*, dan h adalah tinggi dari *bounding box*. Bagian citra input yang berada di dalam *Bounding box* tersebut merupakan potongan citra I_{objek} yang akan digunakan pada SMN dengan posisi tengah dari *bounding box* tersebut merupakan nilai $s_{ini}=(x,y)$, yang menyatakan sebuah posisi objek pada citra dalam koordinat kartesian.

Untuk mendapatkan potongan citra lingkungan ($q(i)$) pada keseluruhan posisi dari citra input, *bounding box* B digunakan sebagai sliding window untuk mendapatkan $q(i)$ dari keseluruhan posisi dari citra input dengan ukuran yang sama dengan ukuran *bounding box* (B). i merupakan indeks dari potongan-potongan citra tersebut dengan posisi tengah dari potongan tersebut merupakan nilai $s_i=(y_i, x_i)$ yang menyatakan posisi potongan citra tersebut dalam koordinat kartesian. Ilustrasi visualisasi I_{objek} dan q dapat dilihat pada Gambar 5.



Gambar 5. Visualisasi dari Potongan Citra Iobject dan Potongan Citra Lingkungan q

B. *Spatial Matching Network* (SMN)

Pada tahap perancangan SMN, telah ditemukan kendala dalam melaksanakan arsitektur jaringan syaraf tiruan yang dirancang oleh Siyu Huang dan kawan-kawan tersebut. Perangkat Lunak MATLAB yang digunakan

dalam penelitian ini tidak memperbolehkan penggunaan 2 input pada jaringan syaraf tiruan yang digunakan pada perangkat lunak tersebut. Oleh karena itu, dalam penelitian ini bagian input dari jaringan syaraf tiruan yang digunakan dimodifikasi menggunakan teknik pemisahan matriks citra 6 *channel* yang digunakan oleh Alfifi Mahmoud dalam salah satu percobaannya [17].

Input dari modifikasi ini menggunakan matriks citra 6 *channel*. Tiga *channel* pertama dari matriks tersebut adalah matriks citra dari I_{obyek} , sementara 3 *channel* selanjutnya adalah matriks citra dari $q(i)$. Kemudian matriks citra tersebut diteruskan pada 6 buah *splitter layer* yang digunakan untuk mengekstraksi nilai-nilai piksel pada masing-masing *channel* dari citra input, lalu *channel-channel* tersebut digabungkan kembali ke masing-masing alur untuk dilanjutkan menuju SMN.

Dalam penelitian ini, proses pelatihan dari SMN menggunakan dataset yang terdiri dari 4 citra dari 4 scene yang berbeda yang telah dilabeli secara manual kecocokan spasial antara I_{obyek} dengan potongan citra lingkungan q pada masing-masing citra masukan dengan rincian sebagai berikut:

- a) Total 814 citra dari semua objek
- b) Terdapat 614 citra sebagai citra latih
- c) Ada 200 citra sebagai citra validasi
- d) Citra-citra tersebut di-*resize* ke ukuran 227×227 piksel (sesuai dengan spesifikasi input dari AlexNet)

Perbandingan spesifikasi proses latih dari SMN dalam penelitian ini dan SMN pada percobaan Siyu Huang dan kawan-kawan ditampilkan pada Tabel 1.

Tabel 1. Spesifikasi dari Proses Latih SMN

Spesifikasi	Penelitian ini	Siyu Huang [4]
Citra Latih	614 potongan Citra	256 Potongan Citra
Proses Re-training	Menggunakan Proses Retraining	Tidak Menggunakan Proses Retraining

Learning Rate	10^{-4}	10^{-3}
Jumlah Iterasi	558 iterasi	2000 iterasi

Dapat dilihat dari Tabel 1, jumlah iterasi dari proses latih *Spatial Matching Network* dalam penelitian ini adalah 558 iterasi. Proses iterasi tersebut menggunakan mini-batch sebesar 20 citra dan jumlah epoch 18 kali. Proses pelatihan *Spatial Matching Network* dalam penelitian ini menggunakan komputer dari Laboratorium Komputer Jurusan Teknik Elektro Universitas Kristen Maranatha dengan spesifikasi *CPU Intel Core i7, GPU Nvidia GeForce GT640, dan RAM 8 GB DDR3*. Hasil dari proses latih ini menghasilkan validasi kesuksesan sebesar 91,8% dengan waktu pelatihan selama 4 jam 38 menit.

C. Proses Pencarian Jalur Visual Terpendek

Untuk mencari jalur visual terpendek dari objek observasi menuju tepi dari gambar, *directed graph* dibangun untuk memetakan posisi-posisi tertentu yang menyatakan node sebagai *vertices* dari *directed graph*. Jarak antar node adalah nilai d dengan formulasi berbeda untuk tiap jenis percobaan. kemudian node tersebut dihubungkan dengan node lain yang berdekatan secara spasial dengan node tersebut, dengan bobot dari *directed graph* merupakan nilai *cost* diantara kedua jalur tersebut berdasarkan $C(P)$. Sesuai dengan rancangan percobaan yang telah dipaparkan pada bagian sebelumnya, penelitian ini menggunakan 4 percobaan prediksi jalur visual berdasarkan perubahan jumlah node dan penggunaan orientasi objek pada perhitungan *cost* dari jalur. Perbedaan formulasi dari tiap percobaan adalah sebagai berikut:

- a) Formulasi d pada percobaan 1 dan 3 adalah:

$$d = \lfloor \sqrt{w^2 + h^2} \rfloor \tag{6}$$

- b) Formulasi d pada percobaan 2 dan 4 adalah:

$$d = \frac{|\sqrt{w^2+h^2}|}{2} \quad (7)$$

c) Formulasi $C(P)$ pada percobaan 1 dan 2 adalah:

$$C(P) = \min_P \sum_{s_i \in P} R_{cost}(s_i)$$

$s.t \quad s_i \text{ dan } s_j +$
 $1 \text{ berdekatan secara spasial}$

$$s_1 = s_{ini}$$

$$s_n = s_{edge_j}, j = 1, \dots, m, \quad (8)$$

d) Formulasi $C(P)$ pada percobaan 3 dan 4 adalah:

$$C(P) = \min_P \sum_{s_i \in P} R_{cost}(s_i) + \varepsilon D(\theta_p, \theta_{GT}),$$

$s.t \quad s_i \text{ dan } s_j +$
 $1 \text{ berdekatan secara spasial}$

$$s_1 = s_{ini}$$

$$s_n = s_{edge_j}, j = 1, \dots, m, \quad (9)$$

Directed graph dari seluruh node, *edges* dan *weights* yang telah dibentuk kemudian digunakan untuk mencari jalur-jalur terpendek dari sini menuju seluruh node tepi dari citra masukan menggunakan algoritma Dijkstra, kemudian jalur-jalur tersebut diurutkan secara ascending untuk keperluan perumusan hasil akhir dari penelitian ini.

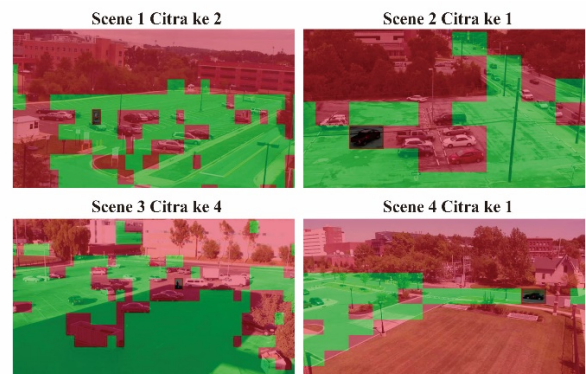
Pada tahap evaluasi dari penelitian ini, 10 jalur $P(i)$ terbaik pada tiap percobaan dari tiap citra dikalkulasi menggunakan *Modified Hausdorff Distance* (MHD). Caranya adalah seluruh koordinat piksel dari tiap node yang dilalui $P(i)$ dihimpun ke dalam variabel A , kemudian variabel A tersebut dikalkulasi kemiripan jalurnya dengan koordinat-koordinat piksel dari jalur *ground truth* yang telah diinputkan secara manual pada persiapan percobaan. Koordinat piksel dari jalur *ground truth* tersebut merupakan variable B . kedua variabel tersebut kemudian dikalkulasi menggunakan *MHD* yang memiliki persamaan matematis sebagai berikut [18]:

$$(A, B) = \max(h(A, B), h(B, A)) \quad (10)$$

Dengan notasi $h(A, B)$ dan $h(B, A)$ merupakan nilai jarak berarah dari 2 kumpulan titik A dan B .

IV. HASIL EKSPERIMEN DAN PEMBAHASAN

Pada bagian ini akan dipaparkan hasil yang diperoleh dari percobaan yang telah dilakukan beserta pembahasan dari hasil yang telah diperoleh. beberapa contoh Reward Map yang dihasilkan dari SMN dapat dilihat pada Gambar 6.

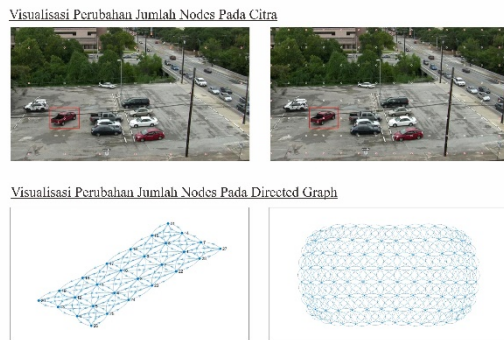


Gambar 6. Contoh Reward Map hasil dari SMN

Pada Gambar 6, bagian citra yang berwarna hijau, menunjukkan bagian-bagian dari citra yang memiliki kesesuaian konteks spasial dengan objek observasi. sementara bagian citra yang berwarna merah, menunjukkan bagian dari citra yang tidak memiliki kesesuaian konteks spasial dengan objek observasi.

Dalam Gambar 6 dapat dilihat juga bahwa keterperincian dari *reward map* yang dihasilkan oleh SMN tergantung ukuran *bounding box* yang digunakan untuk menandai dan memisahkan obyek yang diobservasi. Ketika *bounding box* pada citra berukuran cukup besar, seperti pada *Scene 2 Citra ke 1*, dan *Scene 2 Citra ke 1* yang memiliki objek observasi berupa sebuah mobil, hasil dari pencocokan konteks spasial pada *reward map* tidak terlalu terperinci dibanding gambar-gambar lain yang ukuran *bounding box* dan objek observasinya lebih kecil. Hal ini disebabkan oleh proses pencocokan konteks spasial oleh SMN mencari kecocokan spasial berdasarkan ukuran *bounding box* dari objek

dalam proses perancangannya. Akan tetapi, secara umum berdasarkan hasil *reward map* yang dihasilkan, SMN yang telah dilatih dapat membedakan kecocokan konteks spasial dari obyek dan lingkungannya dengan cukup baik bila ditinjau secara kualitatif. Visualisasi dari perubahan jumlah node ditampilkan pada Gambar 7.

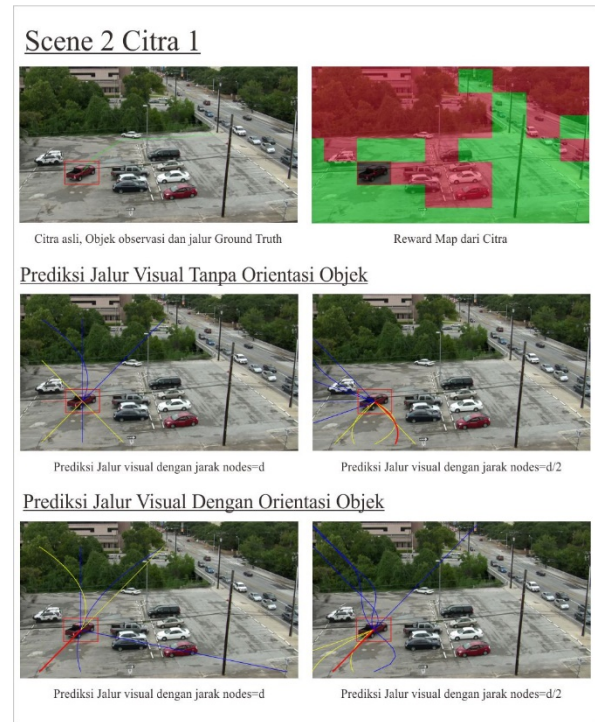


Gambar 7. Visualisasi perubahan jumlah node pada scene 2 citra 1 percobaan

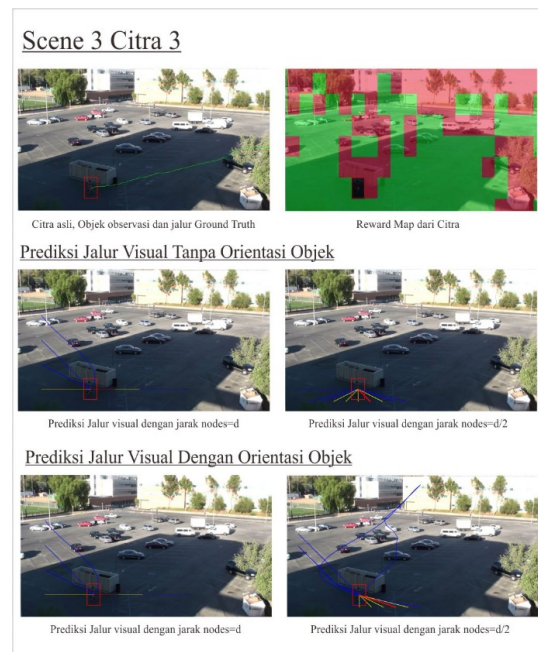
Gambar-gambar pada bagian kiri adalah visualisasi node pada percobaan 1 dan 3, sementara gambar-gambar pada bagian kanan merupakan visualisasi dari node dari percobaan 2 dan 4. Pada Gambar 8 dan Gambar 9 telah diperlihatkan contoh dari semua percobaan yang telah dilaksanakan pada sebuah citra masukan untuk mencari prediksi jalur visual dari objek tertentu. Pada gambar-gambar tersebut telah ditampilkan hasil dari 10 prediksi jalur visual terpendek dari masing-masing percobaan dengan kode warna; merah adalah jalur terpendek, kuning adalah jalur-jalur terpendek peringkat 2 sampai 5, sedangkan warna biru adalah jalur-jalur terpendek peringkat 6 sampai 10.

Dapat dilihat pada gambar yang telah disajikan, hasil prediksi jalur visual yang memperhitungkan orientasi obyek pada perhitungannya menghasilkan jalur yang lebih sesuai dengan jalur asli dari objek observasi, terutama pada jalur selain 5 terpendek. Arah dari jalur-jalur yang diprediksi secara umum sudah sesuai dengan arah gerak dari obyek. Sementara, jika ditinjau dari jumlah node dari *graph G*, beberapa hasil observasi obyek yang menggunakan jumlah node lebih banyak akan menghasilkan jalur yang secara umum lebih baik

dibandingkan dengan jalur yang diperhitungkan menggunakan jumlah node lebih sedikit.



Gambar 8. Contoh Visualisasi Dari Percobaan Pencarian Jalur Terpendek dari Scene 2 Citra 1



Gambar 9. Contoh visualisasi dari percobaan pencarian jalur terpendek dari scene ke-2 citra ke-1

Hasil evaluasi hasil dari seluruh percobaan dalam penelitian ini menggunakan metrik MHD dapat dilihat pada Tabel 2 sampai Tabel 5.

Tabel 2. Evaluasi MHD pada percobaan ke-1

EVALUASI MHD PADA PERCOBAAN 1				
All scene				
Shortest Path	Scene 1	Scene 2	Scene 3	Scene 4
Top	223,2793153	162,9062485	231,9883939	233,3606191
Top-5 average	211,6325185	159,1800299	227,2165278	225,0854077
Top-10 Average	227,0382042	156,6874667	221,7792048	214,0866323

Tabel 3. Evaluasi MHD pada percobaan ke-2

EVALUASI MHD PADA PERCOBAAN 2				
All scene				
Shortest Path	Scene 1	Scene 2	Scene 3	Scene 4
Top	223,2793153	162,9062485	231,9883939	233,3606191
Top-5 average	211,8594613	150,084526	227,0833925	232,0488884
Top-10 Average	205,5842339	157,3302624	228,7415493	228,8765225

Tabel 4. Evaluasi MHD pada percobaan ke-3

EVALUASI MHD PADA PERCOBAAN 3				
All scene				
Shortest Path	Scene 1	Scene 2	Scene 3	Scene 4
Top	215,6221205	157,623532	182,9774697	133,6799352
Top-5 average	225,0964171	150,6989835	200,712535	132,4237514
Top-10 Average	237,2431308	154,2729788	189,9588078	141,9855452

Tabel 5. Evaluasi MHD pada Percobaan ke-4

EVALUASI MHD PADA PERCOBAAN 4				
All scene				
Shortest Path	Scene 1	Scene 2	Scene 3	Scene 4
Top	149,7743504	149,7743504	192,3572437	185,3918776
Top-5 average	152,4831302	152,4831302	200,9655962	147,5429936
Top-10 Average	154,4684676	154,4684676	204,663621	138,7432582

Pada Tabel 2 sampai Tabel 5, indeks *shortest path* adalah jalur-jalur terpendek yang dihasilkan dalam penelitian ini. *top path* adalah rata-rata dari jalur terpendek dari semua percobaan pada masing-masing scene. Sementara, masing-masing dari indeks *top-5 average*, dan *top-10 average* adalah nilai rata-rata pada seluruh jalur yang diprediksi dari citra pada sebuah *scene* lalu dirata-ratakan kembali sesuai dengan peringkat panjang jalur dari nilai tersebut dalam dalam klasifikasi 5 terbaik dan 10 terbaik. Indeks *total average*, merupakan rata-rata total dari *top path*, *top-5 average*, dan *top-10 average*.

Berdasarkan hasil yang disajikan dalam Tabel 2 hingga Tabel 5, dapat dilihat terdapat peningkatan yang signifikan dari evaluasi metrik menggunakan *MHD* dalam percobaan yang memperhitungkan orientasi objek dalam perumusan prediksi jalur visualnya dibandingkan percobaan yang tidak memperhitungkan orientasi objek dalam perumusannya. Sementara itu perubahan dari jumlah node tidak menghasilkan perbaikan yang signifikan dalam hasil evaluasi metriknya. Dapat dilihat juga dalam tabel-tabel tersebut, nilai evaluasi yang paling baik paling banyak berada di jalur diluar *top-1*, peningkatan kualitas dari jalur *top-1* terjadi pada evaluasi percobaan yang memperhitungkan orientasi objek pada perumusannya.

V. SIMPULAN

Berdasarkan hasil percobaan yang diperoleh ada beberapa kesimpulan yang diperoleh, yaitu: Berdasarkan analisis kuantitatif dari penelitian ini, hasil pencarian Jalur visual yang memperhitungkan orientasi dari objek menghasilkan prediksi yang lebih baik daripada hasil pencarian jalur yang tidak memperhitungkan orientasi dari objek.

Secara kualitatif, SMN dapat menilai kesesuaian konteks spasial dari objek dengan lingkungannya (dapat ditinjau dari hasil *reward map* dari penelitian ini), dimana hasil keterperincian dari *reward map* bergantung dari seberapa besar *bounding box* yang digunakan untuk mengidentifikasi objek.

Pengaruh dari jumlah node dalam prediksi jalur visual tidak terlalu signifikan bila ditinjau secara kuantitatif, terlebih pada jalur paling pendek dari hasil prediksi. Tetapi terdapat kenaikan tingkat akurasi pada jalur-jalur prediksi yang berada di peringkat *top-3* sampai *top-10*. Walaupun demikian, secara kualitatif, pengaruh dari jumlah node yang lebih banyak membentuk jalur yang lebih masuk akal dibanding dengan hasil prediksi jalur dengan jumlah node lebih sedikit, terlebih pada obyek-obyek yang diobservasi dengan ukuran besar.

DAFTAR PUSTAKA

- [1] Cambridge Dictionary, “Meaning of Pathway in English.” [Online]. Available: <https://dictionary.cambridge.org/dictionary/english/pathway>.
- [2] Cambridge Dictionary, “Meaning of Visual in English.” [Online]. Available: <https://dictionary.cambridge.org/dictionary/english/visual>.
- [3] J. Hawkins and S. Blakeslee, *On intelligence: How a New Understanding of the Brain will lead to Truly Intelligent Machines*. 2004.
- [4] S. Huang *et al.*, “Deep Learning Driven Visual Path Prediction from a Single Image,” *IEEE Trans. Image Process.*, vol. 25, no. 12, pp. 5892–5904, Dec. 2016.
- [5] J. Walker, A. Gupta, and M. Hebert, “Patch to the future: Unsupervised visual prediction,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2014.
- [6] C. Liu, J. Yuen, A. Torralba, J. Sivic, and W. T. Freeman, “SIFT flow: Dense correspondence across different scenes,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2008.
- [7] J. Yuen and A. Torralba, “A data-driven approach for event prediction,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2010.
- [8] C. G. Keller and D. M. Gavrilu, “Will the pedestrian cross? A study on pedestrian path prediction,” *IEEE Trans. Intell. Transp. Syst.*, 2014.
- [9] K. M. Kitani, B. D. Ziebart, J. A. Bagnell, and M. Hebert, “Activity forecasting,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2012.
- [10] F. Berzal and N. Matin, “Data mining,” *ACM SIGMOD Rec.*, vol. 31, no. 2, p. 66, Jun. 2002
- [11] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” in *Advances in Neural Information Processing Systems*, 2012.
- [12] M. Cavaioni, “DeepLearning series: Convolutional Neural Networks,” 2018. [Online]. Available: <https://medium.com/machine-learning-bites/deeplearning-series-convolutional-neural-networks-a9c2f2ee1524>
- [13] S. Yantis and R. A. Abrams, *Sensation and Perception*, 2nd ed. New York, NY: Worth Publishers, 2017.
- [14] S.-H. Tsang, “Review: AlexNet, CaffeNet — Winner of ILSVRC 2012 (Image Classification),” 2018. [Online]. Available: <https://medium.com/coinmonks/paper-review-of-alexnet-caffenet-winner-in-ilsvrc-2012-image-classification-b93598314160>
- [15] M. Afifi, “11K Hands: Gender recognition and biometric identification using a large dataset of hand images,” *Multimed. Tools Appl.*, vol. 78, no. 15, pp. 20835–20854, Aug. 2019
- [16] J. B. J. and G. Gutin, “Digraphs: theory, algorithms and applications,” *Softw. Testing, Verif. Reliab.*, vol. 12, no. 1, pp. 59–60, 2002.
- [17] J. Daintith, “A dictionary of computing,” *Oxford dictionary of computing*. 2008.
- [18] M.-P. Dubuisson and A. K. Jain, “A modified Hausdorff distance for object matching,” 2002, pp. 566–568