

Segmentasi Informasi pada *Raid Entry Page* Gim *Granblue Fantasy* Menggunakan Deteksi Tepi

Naufal Yahya Kurnianto - 13519141

Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
E-mail (gmail): 13519141@std.stei.itb.ac.id

Abstract—Pada masa-masa maraknya terjadi globalisasi saat ini, persebaran informasi menjadi lebih mudah untuk dilakukan. Menggunakan internet, banyak orang dapat dengan mudah mencari informasi dan tentunya melakukan rekreasi. Salah satu metode rekreasi adalah dengan bermain gim. *Granblue fantasy* merupakan gim daring berbasis *web browser* yang telah dimainkan oleh jutaan orang. Untuk membantu pemain dapat bekerja sama dengan pemain lainnya, diperlukan otomatisasi yang mungkin dapat dicapai dengan memanfaatkan informasi yang ada pada halaman *entry raid*. Karena halaman tersebut juga dapat dibidang sebagai citra, tercetus ide untuk mengambil informasi dari halaman tersebut menggunakan segmentasi citra. Makalah ini akan membandingkan berbagai metode deteksi tepi untuk citra halaman *entry raid* serta kemudian menyegmentasikan halaman citra tersebut untuk mengklasifikasi informasi-informasi yang terdapat pada halaman tersebut.

Keywords—Citra; Deteksi Tepi; Segmentasi Citra; Informasi; Halaman *Entry Raid*

I. PENDAHULUAN

Antarmuka merupakan suatu bagian yang penting antara interaksi manusia dengan mesin. Sejak zaman dahulu, untuk mengoperasikan sesuatu yang memiliki cara kerja yang relatif kompleks dibutuhkan suatu medium antara manusia dengan yang dioperasikan agar yang mengoperasikan tidak perlu mengeluarkan usaha lebih banyak dari seharusnya. Medium tersebut terus berkembang hingga zaman sekarang yang mana penggunaan berbagai alat yang relatif canggih semakin berkembang.

Salah satu hal umum yang pada zaman sekarang digunakan oleh manusia adalah internet. Menggunakan internet, manusia dapat melakukan hampir segala aktivitas. Berbagai aktivitas yang dapat dilakukan adalah membaca berita, menonton video, berkomunikasi dengan orang lain, maupun bermain gim. Internet ini tentu yang terjadi di balik layar pengguna bersifat lumayan kompleks dan tentu saja tidak bisa dilakukan secara manual. Maka dari itu, antarmuka kembali menjadi medium penting bagi manusia dengan mesin untuk berkomunikasi. Secara simpel, manusia dapat beraktivitas memanfaatkan internet untuk berselancar di dunia maya.

Seperti yang telah disebutkan sebelumnya, salah satu aktivitas yang dapat dilakukan adalah bermain gim. Terdapat kategori gim yang hanya dapat dimainkan jika terhubung

dengan internet. Gim-gim tersebut memanfaatkan internet di antaranya untuk mengakses *server* dari gim tersebut, berkomunikasi antara pemain gim secara daring, maupun bekerja sama antara pemain untuk mengalahkan atau menyelesaikan beberapa misi di dalam gim tersebut. Salah satu gim daring tersebut adalah *Granblue Fantasy*.

Granblue fantasy merupakan gim daring yang berbasis giliran atau bersifat *turn-based* serta merupakan gim *gacha* yang dibuat oleh salah satu pembuat gim di Jepang yaitu *cygames*. Gim ini tersedia secara bebas pada peramban situs web atau *web browser* dan dapat diakses pada <https://game.granbluefantasy.jp/>. Salah satu halaman antarmuka di gim ini adalah *raid entry page* yang akan dijelaskan pada bagian berikutnya.

Beranjak dari topik antarmuka, belakangan ini konsep otomatisasi mulai populer. Memanfaatkan otomatisasi, berbagai tugas manusia dapat dikerjakan dengan lebih mudah dan tentunya lebih cepat. Salah satu aplikasi otomatisasi yang dapat digunakan untuk memudahkan pemain *granblue fantasy* adalah dengan membuat kompilasi dari berbagai informasi *raid entry page*. Karena halaman ini dapat dibidang juga berupa gambar, dapat dimanfaatkan teknik pengolahan atau pemrosesan citra untuk mengambil informasi yang ada pada halaman tersebut.

Pengolahan citra sendiri merupakan bidang keilmuan komputer yang secara umum berkuat dalam pengolahan berbagai jenis citra dan mendapatkan informasi dari citra tersebut. Salah satu aplikasi pengolahan citra adalah untuk menyegmentasi citra agar mesin dapat melihat citra dan mengambil informasi dari citra tersebut dengan lebih efisien. Maka dari itu, akan dilakukan berbagai percobaan segmentasi citra pada halaman *raid entry* pada gim *granblue fantasy* yang akan memanfaatkan deteksi tepi.

Secara umum percobaan akan membandingkan berbagai metode deteksi tepi yang ada serta digunakan berbagai citra cuplikan layar yang memiliki ukuran beragam. Hasilnya kemudian akan dibandingkan untuk menentukan metode deteksi tepi yang paling cocok.

II. LANDASAN TEORI

A. Granblue Fantasy

Granblue fantasy merupakan gim daring sifatnya berbasis giliran dan permainannya sendiri berbasis *web*. Selain itu, *granblue fantasy* juga merupakan gim *gacha* yang akhir-akhir ini semakin banyak dan semakin populer. Untuk gim ini sendiri, sejatinya sudah ada dari delapan tahun yang lalu dan dikembangkan oleh perusahaan jepang bernama *cygames*.

Sebagai gim yang interaktif, tentu terdapat berbagai halaman antarmuka yang ditampilkan pada pemain. Berbagai halaman yang umumnya digunakan oleh pemain adalah halaman *home* atau beranda yang secara umum menampilkan berbagai informasi halaman lain dan notifikasi untuk pemain. Selain itu, ada pula halaman misi, toko, inventori, maupun profil. *Granblue fantasy* yang sejatinya merupakan gim daring tentu memiliki fitur yang membuat pemain dapat bekerja sama untuk menuntaskan misi.

Fitur bekerja sama pada *granblue fantasy* memiliki nama yang memang umum pada gim-gim permainan peran yaitu *raid*. Pada *raid* ini pemain dapat meminta bantuan pemain lain dengan membuka misi *raid* yang sedang dilakukan pemain tersebut. Bagi pemain lain yang ingin bergabung untuk membantu pemain yang membuka misi *raid* tersebut, terdapat halaman awal yang ditampilkan sebelum bisa memasuki ke dalam misi *raid*. Halaman awal tersebut adalah halaman *raid entry page* yang contohnya dapat dilihat pada *fig. 1*.



Fig. 1. Contoh Halaman *raid entry page*

Pada halaman awal tersebut, terdapat berbagai bagian berisi informasi yang dapat diolah oleh pemain di antaranya sebagai berikut:

1. Navigation Bar

Pada bagian atas halaman, terdapat tiga subbagian yang berisikan tombol *home* atau beranda untuk kembali ke halaman beranda, tombol *menu* untuk membuka *menu* yang berisi berbagai fitur lain *granblue fantasy*, serta nama halamannya yang merupakan halaman pemilihan *summon* atau entitas yang dapat membantu pemain.

2. Bagian Informasi Raid

Pada bagian ini, terdapat berbagai informasi yang dapat diolah oleh pemain yaitu nama *raid* atau musuhnya, banyak nyawa dari musuh yang tersisa, jumlah *EP* atau energi yang harus digunakan, jumlah pemain yang telah bergabung pada misi *raid* tersebut, banyak waktu yang tersisa untuk menyelesaikan misi *raid* tersebut, serta pemain yang membuka misi *raid* tersebut. Tentu tidak semua informasi tersebut penting untuk diolah atau disimpan. Informasi yang paling penting adalah banyak nyawa tersisa, jumlah pemain yang telah bergabung, dan banyak waktu yang tersisa untuk misi *raid* tersebut.

3. Bagian Pemilihan Summon

Summon merupakan entitas yang harus dipilih pemain untuk membantu pemain dalam menyelesaikan misi apapun pada gim *granblue fantasy*. *Summon* sendiri dipisah dalam berbagai elemen. Pemilihan *summon* umumnya akan memperhatikan elemen, efek *summon* dan tingkat kekuatan *summon*.

Seperti yang telah disebutkan sebelumnya, terdapat berbagai informasi penting yang dapat diolah dan disimpan dengan tujuan salah satunya untuk membentuk sistem otomatisasi maupun untuk memperoleh wawasan dari berbagai data berisi informasi halaman awal tersebut.

B. Citra

Citra sering disebut juga sebagai gambar yang ada pada bidang dua dimensi atau dwimatra. Citra sendiri merupakan sinyal dwimatra yang bersifat menerus atau kontinu yang dapat diamati oleh sistem visual manusia. Secara matematis, sejatinya citra merupakan fungsi dwimatra yang menyatakan intensitas cahaya pada bidang dwimatra tersebut.

Citra ini dapat menjadi atau dikategorikan sebagai citra digital melalui proses digitalisasi yang memanfaatkan *sampling* dan kuantisasi. Citra digital kemudian direpresentasikan sebagai matriks $M \times N$ yang merupakan resolusi citra. Tiap elemen pada matriks $M \times N$ tersebut menyatakan sebuah *pixel* (*picture element*). Contoh representasi matriks $M \times N$ dapat dilihat pada *fig.2*.

$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & \dots & f(1,N-1) \\ \dots & \dots & \dots & \dots \\ f(M-1,0) & f(M-1,1) & \dots & f(M-1,N-1) \end{bmatrix}$$

Fig. 2. Representasi Citra Digital dalam Bentuk Matriks M x N

Pada matriks M x N tersebut, nilai dari $f(x,y)$ menyatakan nilai keabuan atau *gray level*. Secara umum nilai keabuan yang digunakan adalah 256 level atau setara dengan delapan bit.

C. Deteksi Tepi

Tepi adalah bagian dari citra yang mengalami perubahan intensitas nilai keabuan yang mendadak dalam jarak yang singkat. Perubahan nilai keabuan yang mendadak sendiri artinya selisih nilai *pixel* antara satu dengan sebelumnya bernilai besar seperti dari warna terang yang memiliki nilai keabuan 200an menjadi warna yang sangat gelap yang mana nilai keabuannya mungkin bernilai satu atau dua. Tepi juga memiliki arah dan arahnya berbeda-beda tergantung pada perubahan intensitas.

Umumnya, tepi terdapat pada batas antara dua daerah yang perbedaan intensitasnya sangat cepat dalam citra. Tepi memiliki beberapa macam seperti tepi curam (*step edge*), tepi landai (*ramp edge*), tepi garis (*line edge*), dan tepi atap (*roof edge*). Contoh tepi dalam citra dan ilustrasi berbagai macam tepi dapat dilihat pada *fig.3*.

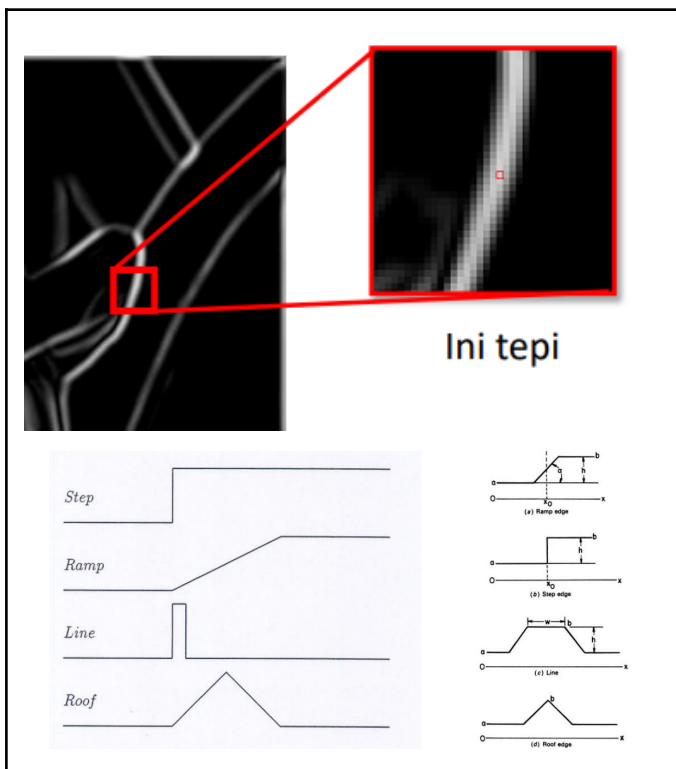


Fig. 3. Contoh Tepi dan Ilustrasi Macam-Macam Tepi

Seperti yang dapat dilihat pada contoh tepi tersebut, tentu akan sangat sulit untuk mendeteksi tepi yang benar-benar mewakili tepi pada citra jika citra banyak mengandung derau. Maka dari itu, umumnya dilakukan *preprocessing* citra terlebih dahulu untuk menghilangkan derau seperti menggunakan *image smoothing*.

Deteksi tepi kemudian dilakukan dengan tujuan untuk meningkatkan penampakan dari tepi selagi tidak menghiraukan bagian citra sisanya yang bukan merupakan tepi. Maka dari itu, citra hasil pendeteksian tepi idealnya hanya berisikan garis-garis yang mewakili tepi yang terdapat dalam citra. Metode untuk mendeteksi tepi ada beragam, dapat dimulai dari yang pertama yaitu operator gradien yaitu turunan pertama. Metode gradien ini menggunakan pendekatan kalkulus diferensial dengan menganggap perubahan intensitas yang besar sebagai fungsi kemiringan yang besar. Ilustrasinya dapat dilihat pada *fig.4*.

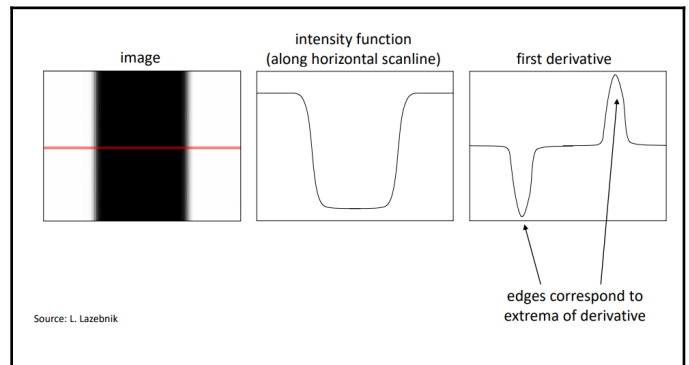


Fig. 4. Ilustrasi Turunan Pertama

Metode kedua adalah dengan memanfaatkan turunan kedua yang dinamakan operator laplace. Menggunakan operator laplace, dapat diidentifikasi tepi yang curam karena pada turunan keduanya terdapat *zero-crossing*. Selain itu, operator laplace juga dapat dimodifikasi menjadi operator LoG yang mana aplikasinya adalah dengan melakukan *image smoothing* terlebih dahulu menggunakan fungsi gaussian yang kemudian dideteksi tepinya menggunakan operator laplace. Untuk deteksi tepi citra sendiri, ketiga operator menggunakan pendekatan diferensial ini dapat diaplikasikan menggunakan teknik konvolusi matriks. Menurut berbagai sumber yang ada, terdapat beberapa matriks yang *eligible* dan sesuai untuk dikonvolusikan dengan citra sebagai pengaplikasian ketiga operator tersebut. Matriks dapat dilihat pada bagian eksperimen.

Selain ketiga metode tersebut, ada empat metode lagi yaitu operator sobel, roberts, prewitt, dan canny. Sama seperti metode-metode sebelumnya, pengaplikasian deteksi tepi pada citra adalah dengan melakukan konvolusi matriks antara matriks operator dengan matriks citra. Untuk operator canny sendiri, digunakan dua nilai ambang sehingga dapat dihasilkan tepi yang relatif akurat dan berukuran 1 *pixel*. Secara umum, operator canny akan menghasilkan hasil yang sangat memuaskan seperti pada contoh yang dapat dilihat pada *fig.5*.

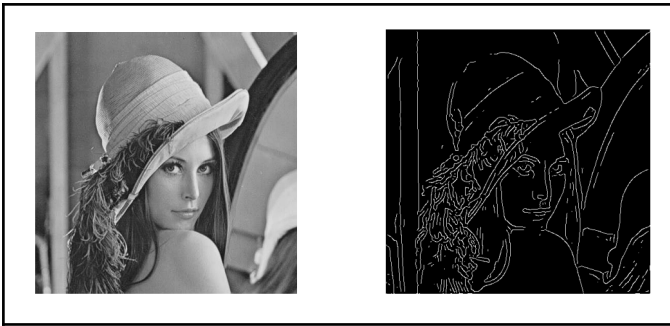


Fig. 5. Contoh Aplikasi Deteksi Tepi Operator Canny

D. Segmentasi Citra

Segmentasi citra adalah operasi mempartisi citra menjadi sebuah koleksi yang terdiri dari sekumpulan *pixel* yang terhubung satu sama lain. Tujuan dari segmentasi citra adalah menemukan bagian citra yang koheren atau objek spesifik. Berbagai aplikasi dari segmentasi citra adalah untuk *autonomous vehicle*, *medical imaging*, *object recognition*, dan *scene understanding*.

Metode segmentasi citra dapat dibagi menjadi dua. Pertama adalah melalui diskontinuitas yang melakukan partisi citra dengan memanfaatkan deteksi tepi. Kedua adalah melalui pendekatan *similarity* yang mana kemiripan area dalam citra berdasarkan properti yang sebelumnya telah ditentukan. berbagai metode pendekatan *similarity* adalah pengambangan, *region growing*, *split and merge*, dan *clustering*.

III. IMPLEMENTASI

Solusi akan diimplementasikan menggunakan program Matlab R2022a. Secara umum, percobaan atau eksperimen terhadap pendekatan solusi adalah dengan membandingkan berbagai metode deteksi tepi untuk berbagai ukuran dari halaman *raid entry* pada gim *granblue fantasy*. Setelah itu, juga perlu diaplikasikan berbagai batasan agar percobaan akan lebih jelas. Berbagai batasan tersebut diantaranya adalah:

1. Operator yang akan digunakan akan menggunakan *library* yang telah disediakan matlab untuk operator yang tersedia, di antaranya adalah log, sobel, robert, prewitt, dan canny. Untuk operator laplace akan menggunakan buatan sendiri.
2. Ukuran citra yang akan digunakan secara umum dapat dikelompokkan menjadi tiga yaitu ukuran kecil, sedang, dan besar. Tujuan modifikasi ukuran ini adalah untuk mendapatkan ukuran terkecil bagi operator yang paling cocok hingga hasilnya relatif memuaskan.
3. Halaman *raid entry* yang akan disegmentasikan hanya pada bagian informasi *raid*.

Untuk batasan nomor dua dan tiga, contoh ukuran citra dan bentuk citra dapat dilihat pada *fig.6*.

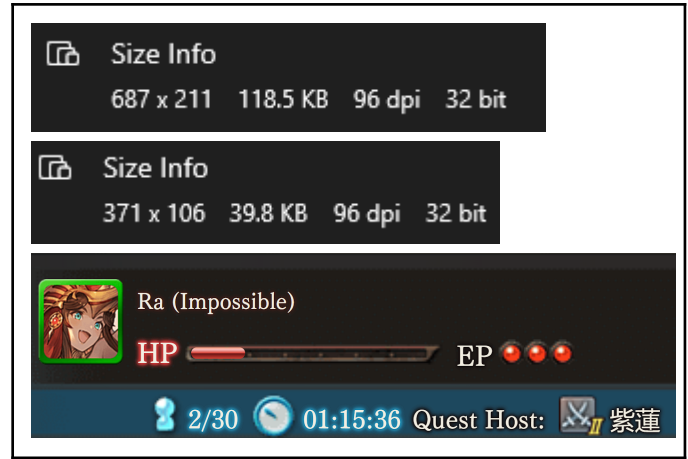
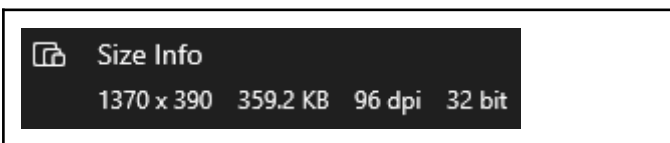


Fig. 6. Contoh Ukuran Citra Besar, Sedang, dan Kecil, Serta Contoh Tampilan Citra Halaman *Entry Raid*

Sebelum proses segmentasi dan perancangan solusi segmentasi dilakukan, akan dilakukan komparasi terlebih dahulu antara tiap-tiap metode deteksi tepi pada tiap-tiap ukuran gambar. Proses ini simpel saja karena hanya mengaplikasikan tiap metode pada tiga gambar berukuran berbeda dan proses evaluasi untuk ini hanya dilakukan secara kasat mata yaitu melalui evaluasi pribadi penguji. Untuk kode proses ini dapat dilihat pada *fig.7* dan untuk hasilnya dapat dilihat pada bagian hasil dan evaluasi.

```
img = imread('img/sample1.png');
img = imadjustn(img);
figure, imshow(img);
imgin = im2gray(img);

laplaceimg = laplace(imgin,'normal');
logimg = edge(imgin, 'log');
sobelimg = edge(imgin, 'Sobel');
prewittimg = edge(imgin, 'Prewitt');
robertsimg = edge(imgin, 'Roberts');
cannyimg = edge(imgin, 'Canny');

figure, imshow(img);
figure, imshow(laplaceimg);
figure, imshow(logimg);
figure, imshow(sobelimg);
figure, imshow(prewittimg);
figure, imshow(robertsimg);
figure, imshow(cannyimg);
```

Fig. 7. Kode Proses Komparasi

Setelah proses komparasi dilakukan, hasil terbaik di antara kombinasi metode deteksi tepi dan ukuran gambar akan

menjadi acuan untuk proses segmentasi. Segmentasi akan dilakukan kepada hasil citra deteksi tepi dengan tujuan akhir untuk mengklasifikasikan bagian informasi. Kode proses segmentasi dapat dilihat pada *fig.8*.

```
function imgout = raidsegmentation(imgin,
imgoriginal)
    linelength = 1;
    se0 = strel('line', linelength, 0);
    se45 = strel('line', linelength, 45);
    se90 = strel('line', linelength, 90);
    se135 = strel('line', linelength, 135);
    dilated = imdilate(imgin, [se135 se90 se45
se0]);
    dilated = imclearborder(dilated,4);
    imgout = imfill(dilated, 'holes');
    imgout = bsxfun(@times, imgoriginal,
cast(imgout, 'like', imgoriginal));
end
```

Fig. 8. Contoh Aplikasi Deteksi Tepi Operator Canny

IV. HASIL DAN ANALISIS

Hasil dari proses komparasi dapat dilihat pada *fig.9*. Menurut hasil proses tersebut, secara umum citra dengan ukuran sedang menggunakan metode canny memberikan hasil yang relatif bagus. Tentunya penguji juga melihat tiap-tiap citra yang mana masing-masing hanya menggunakan satu *figure* agar hasil dapat diamati dengan lebih baik.

Menggunakan acuan ukuran citra sedang dan metode deteksi tepi canny, dapat dilihat berbagai contoh hasil segmentasi yang dilakukan pada *fig.10*. Hasil segmentasi hanya dari metode deteksi tepi belum sempurna. Namun demikian, masih dapat diberikan berbagai modifikasi tambahan menggunakan metode tambahan selain deteksi tepi.

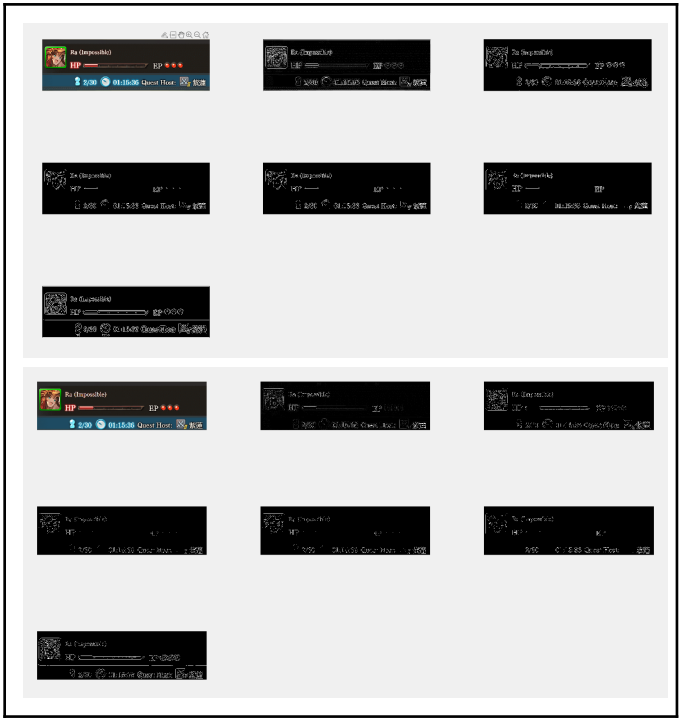


Fig. 9. Contoh Aplikasi Deteksi Tepi Operator Canny

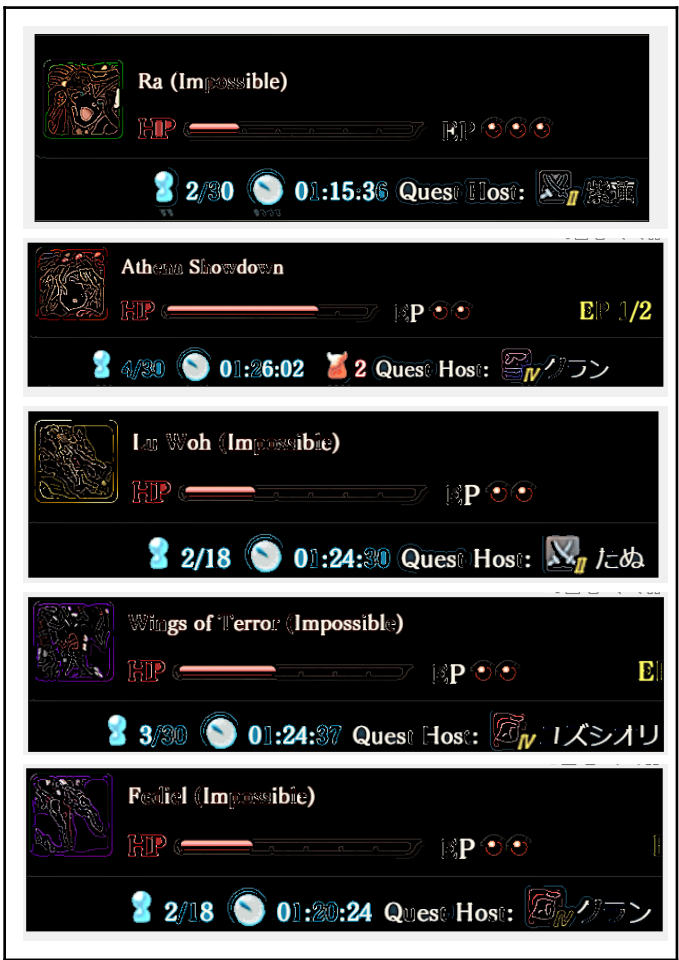


Fig. 10. Hasil Segmentasi

V. KESIMPULAN

Secara umum, hasil segmentasi dari pengujian yang sudah dilakukan dapat membagi-bagi bagian dari citra halaman *raid entry* yang bersifat informasi. Sayangnya, masih dapat banyak kekurangan yang dapat dilihat seperti berbagai angka 1 atau 4 maupun 30 yang agak kabur pada bagian banyaknya pemain yang sudah berada di dalam misi. Selain itu, umumnya juga yang berkaitan dengan angka tidak tersegmentasi dengan terlalu baik. Terkait informasi banyaknya nyawa musuh yang tersisa, sudah tersegmentasi relatif lebih baik daripada segmentasi angka.

Berbagai hal yang dapat menyebabkan kurang baiknya hasil segmentasi adalah kemungkinan adanya noise periodikal yang tidak di-*preprocess* terlebih dahulu karena tidak terlalu terlihat secara kasat mata. Selain itu, ada pula kemungkinan spek dari metode deteksi lain atau ukuran citra yang lain dapat memberikan hasil segmentasi yang lebih baik. Kemudian, proses segmentasi juga masih dapat diperbaiki lagi dengan menambahkan berbagai metode pengolahan citra di luar dari metode deteksi tepi seperti *masking* menggunakan koordinat.

Kesimpulan yang dapat diambil adalah metode deteksi tepi dapat membentuk citra segmentasi yang lumayan baik dalam kasus halaman *raid entry*. Namun demikian, hasil citra segmentasi masih dapat dimodifikasi kembali agar citra dapat tersegmentasi lebih baik lagi dan lebih mudah terbaca oleh mesin agar untuk tujuan otomatisasi atau pembentukan dataset dapat berjalan dengan lebih mudah.

VI. UCAPAN TERIMA KASIH

Pertama-tama, penulis mengucapkan puji syukur kepada Allah SWT karena berkat kehendak-Nya penulis dapat mendapatkan dorongan secara spiritual untuk terus mengerjakan tugas ini agar penulis dapat menyelesaikan tugas makalah untuk mata kuliah Interpretasi dan Pengolahan Citra ini dengan baik. Selain itu, penulis juga ingin mengucapkan terima kasih kepada keluarga yang telah senantiasa memberikan *support* di rumah; juga kepada teman-teman yang selalu mengingatkan untuk terus memberikan proses pada makalah khususnya jika telah mulai mendekati *deadline*; serta berbagai pihak lain yang mungkin telah membantu dan tidak dapat disebut. Terakhir, penulis juga senantiasa ingin memberikan terima kasih sebanyak-banyaknya kepada Bapak Dr. Ir. Rinaldi Munir, MT. selaku dosen IF4073 Interpretasi dan Pengolahan Citra yang telah memberikan ilmu pengetahuan selama perjalanan kuliah pada semester gasal

Tahun 2022/2023 ini. Hal tersebut tentunya sangat berkontribusi pada proses pembentukan makalah ini.

VII. REFERENSI

- [1] Munir, Rinaldi. "Pengantar Pengolahan Citra (Bagian 1)". <https://informatika.stei.itb.ac.id/~rinaldi.munir/Citra/2022-2023/01-Pengantar-Pengolahan-Citra-Bag1-2022.pdf>. Diakses pada 20 Desember 2022.
- [2] Munir, Rinaldi. "Pengantar Pengolahan Citra (Bagian 2)". <https://informatika.stei.itb.ac.id/~rinaldi.munir/Citra/2022-2023/01-Pengantar-Pengolahan-Citra-Bag2-2022.pdf>. Diakses pada 20 Desember 2022.
- [3] Munir, Rinaldi. "Pendeteksian Tepi (Bagian 1)". <https://informatika.stei.itb.ac.id/~rinaldi.munir/Citra/2021-2022/18-Pendeteksian-Tepi-Bagian1-2022.pdf>. Diakses pada 20 Desember 2022.
- [4] Munir, Rinaldi. "Pendeteksian Tepi (Bagian 2)". <https://informatika.stei.itb.ac.id/~rinaldi.munir/Citra/2021-2022/18-Pendeteksian-Tepi-Bagian2-2022.pdf>. Diakses pada 20 Desember 2022.
- [5] Munir, Rinaldi. "Segmentasi Citra (Bagian 1)". <https://informatika.stei.itb.ac.id/~rinaldi.munir/Citra/2021-2022/22-Segmentasi-Citra-Bagian1-2022.pdf>. Diakses pada 20 Desember 2022.
- [6] Munir, Rinaldi. "Segmentasi Citra (Bagian 2)". <https://informatika.stei.itb.ac.id/~rinaldi.munir/Citra/2021-2022/22-Segmentasi-Citra-Bagian2-2022.pdf>. Diakses pada 20 Desember 2022.
- [7] "Granblue Fantasy Wiki". https://gbf.wiki/Main_Page. Diakses pada 20 Desember 2022.
- [8] Matlab. "Detect Cell Using Edge Detection and Morphology". <https://www.mathworks.com/help/images/detecting-a-cell-using-image-segmentation.html>. Diakses pada 20 Desember 2022.

VIII. PRANALA PROGRAM DAN VIDEO

<https://github.com/ayahyaaa/pengcit/tree/main/makalah>
<https://youtu.be/vrw4FZcl8go>

IX. PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Jakarta, 19 Desember 2022



Naufal Yahya Kurnianto 13519141