

Deteksi Plat Nomor Kendaraan Bermotor dengan Segmentasi Gambar

Yasinta Oktaviana Legu Rema^a

^aProgram Fakultas Pertanian, Universitas Timor, Kefamenanu, TTU – NTT, Indonesia, rema.avana@gmail.com

Article Info

Article history:

Received 20 Mei 2019

Received in revised form 17 Juni 2019

Accepted 27 Juni 2019

DOI

<https://doi.org/10.32938/slk.v2i1.794>

Keywords:

Kendaraan Bermotor

Segmentasi Gambar

RGB

Deteksi Tepi

Segmentasi

Abstrak

Banyak sistem dengan banyak variasi untuk pengenalan plat di negara lain dan metode. Metode ini digunakan untuk mendeteksi plat kendaraan berdasarkan pada bentuk data gambar. Ini dapat membantu sistem yang lebih besar yang digunakan untuk identifikasi pelat. Input yang digunakan adalah file gambar. Aplikasi akan menunjukkan kepada pengguna input dalam bentuk gambar dan memprosesnya menggunakan segmentasi gambar. Segmentasi gambar akan mengekstraksi objek dari gambar yang tidak akan menggunakan dan membagi gambar menjadi beberapa bagian. Sebelum menemukan pelat, aplikasi akan memproses gambar untuk mendapatkan gambar berkualitas lebih baik dengan banyak cara termasuk RGB dan pemrosesan skala abu-abu, ukuran gambar, perhitungan area, deteksi tepi dan segmentasi. Dan hasilnya akan ditampilkan dalam file gambar juga.

1. Pendahuluan

Ketika menuju suatu tempat untuk memenuhi kebutuhan hidup maka seseorang memerlukan transportasi sebagai alat mobilitas. Alat mobilitas yang biasa digunakan adalah kendaraan. Alat mobilitas biasanya dipilih berdasarkan keamanan, kenyamanan, kecepatan dan kebutuhannya (Imelda, 2012). Menurut Undang – Undang Republik Indonesia Nomor 22 Tahun 2009 tentang Lalu Lintas dan Angkutan Jalan pada Bab VII pasal 47, terdiri dari kendaraan bermotor dan kendaraan tidak bermotor. Kendaraan bermotor dikelompokkan berdasarkan jenisnya yaitu: sepeda motor, mobil penumpang, mobil bus, mobil barang dan kendaraan khusus. Perkembangan mode transportasi seperti ini memang sangat memudahkan manusia untuk akses perjalanan. Di masa modern ini, Sistem transportasi merupakan elemen dasar infrastruktur yang berpengaruh pada pola pengembangan perkotaan.

Plat nomor yang cukup akrab dengan kita adalah Plat Nomor Indonesia. Negara Indonesia menganut format tiga nomor seri. Bagian depan mewakili tempat nomor tersebut terdaftar. Kode huruf belakang sendiri biasanya mewakili tempat kendaraan tersebut terdaftar. Nomor polisi sendiri dialokasikan untuk jenis kendaraan-kendaraan tertentu. Kendaraan penumpang menggunakan nomor seri 1 - 2999, 8000 – 8999. Nomor diantara seri kendaraan penumpang dibagi untuk motor, kendaraan beban, dan bus. Nomor polisi diberikan sesuai dengan urutan pendaftaran kendaraan bermotor. Terdiri dari 1 -4 angka, ditempatkan setelah kode wilayah pendaftaran. Apabila nomor urut pendaftaran yang telah dialokasikan habis digunakan, maka nomor urut pendaftaran berikutnya kembali ke nomor awal yang telah dialokasikan dengan diberi tanda pengenalan huruf seri A – Z di belakang angka pendaftaran. Kode wilayah pendaftaran kendaraan bermotor ditetapkan berdasarkan Peraturan Kapolri Nomor 4 Tahun 2006 (Mellolo, 2012).

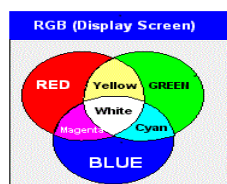
2. Metode

Alur proses deteksi plat nomor ini menggunakan beberapa tahap, yaitu:

2.1 RGB

Citra digital adalah teknik untuk mentransformasikan suatu citra menjadi citra lain dengan menggunakan komputer, dengan tujuan agar citra lebih mudah diinterpretasikan, salah satu contoh adalah citra RGB. Representasi citra RGB triplet akan terdiri dari 3 angka yang mengatur intensitas dari Red (R), Green (G), dan Blue (B) dari suatu triplet. Setiap triplet akan mempresentasikan 1 piksel (picture element). Citra setiap piksel pada citra direpresentasikan dengan 24 bit, 8 bit untuk Red, 8 bit untuk Green, dan 8 bit untuk Blue. Penelitian memperlihatkan bahwa kombinasi warna yang memberikan rentang warna yang paling lebar adalah Red (R), Green (G), dan Blue (B). ketiga warna tersebut diberi nama warna pokok (primaries), dan sering disingkat sebagai warna dasar RGB (Nelson Diazb, 2019). Warna-warna lain dapat diperoleh dengan mencampurkan ketiga warna pokok tersebut dengan perbandingan tertentu meskipun sepenuhnya benar, karena tidak semua kemungkinan warna dapat dihasilkan dengan kombinasi RGB saja sesuai dengan teori Young (1802) yang menyatakan bahwa sembarang warna dapat dihasilkan dari pencampuran warna-warna pokok C1, C2, C3 dengan presentasi tertentu :

$$C = a C1 + b C2 + c C3$$



Bila citra warna digitalisasi, maka tiga buah filter digunakan untuk mengekstraksi intensitas warna merah, hijau, dan biru. Bila ketiganya dikombinasikan akan memperoleh persepsi warna. Setiap citra dalam RGB, masing-masing komponen warna RGB mempunyai histogram. Histogram tingkat keabuan adalah suatu fungsi yang menunjukkan jumlah titik yang ada didalam suatu citra untuk setiap tingkat keabuan. Absis (sumbu x) nya adalah tingkat keabuan, dan ordinat (sumbu y) adalah frekuensi kemunculan atau banyaknya titik dengan nilai keabuan tertentu (S. Mas, 2019). Histogram dapat pula direpresentasikan sebagai vector fitur (fitur vector) dalam ruang vector berdimensi n. ukuran histogram ditentukan oleh jumlah bit perkanal warna (bin). Jadi histogram citrawarna 24 bit memiliki ukuran 256 pada masing-masing kanal warna RGB, sehingga secara keseluruhan terdiri dari 256x3=768 dimana masing-masing bin dipetakan menuju warna RGB tertentu.

Citra RGB mempunyai array berukuran m x n x 3 yang mendefinisikan warna merah, warna hijau, dan warna biru untuk setiap pixelnya. Warna pada setiap pixel ditentukan dari kombinasi merah, hijau, dan biru. RGB merupakan citra 24 bit dengan komponen merah, hijau, dan biru disimpan masing-masing berukuran 8 bit yang berarti memiliki tingkat kecerahan warna sampai 256 level. Dengan kata lain tiap layer warna dapat menyumbang tingkat kecerahan warnanya dari level 0 sampai level 255. Jika tiap layer memberikan nilai intensitas warna yang sama maka warna yang dihasilkan menjadi derajat keabuan dari hitam (tiap layer memberi intensitas warna level 0) sampai putih (tiap layer memberi intensitas warna level 255). Begitu juga bila tiap layer memberikan nilai intensitas yang dominan dari layer lain maka warna kombinasi yang dihasilkan juga akan dominan pada layer dengan intensitas yang terbesar tersebut.

2.2 Grayscale

Citra grayscale atau bisa juga disebut citra beraras keabuan adalah citra yang merepresentasikan warna kedalam tingkat keabuan. Atau lebih khususnya citra digital grayscale adalah citra dimana nilai tiap pixel didalamnya adalah berupa sampel tunggal. Hasil citra yang ditunjukkan jenis ini secara khusus terdiri atas warna kelabu yang terbagi dalam rentang warnadari hitam di intensitas yang paling lemah sampai putih dengan intensitas yang paling kuat. Citra jenis ini biasa juga disebut citra black and white atau disebut sebagai monokromatik (S. Mas, 2019). Citra grayscale dapat dihasilkan dari citra RGB dengan mengalikan komponen warna citra RGB dengan suatu koefisien yang jumlahnya satu. Dalam komputasi, walaupun grayscale dapat dihitung dengan bilangan rasional, piksel citra akan disimpan dalam bentuk biner dan sudah dikuantisasi. Hal ini disebabkan karena grayscale citra memiliki struktur yang lebih sederhana daripada citra warna RGB sehingga untuk komputasi dan pengolahan citra dapat lebih cepat dan efisien (Nelson Diazb, 2019). Intensitas keabuan biasanya disimpan sebagai data citra 8 bit per sampel piksel atau 256 intensitas warna gray dari nilai 0 (hitam) dan 255 (putih).

Untuk melakukan perubahan suatu gambar full color (RGB) menjadi suatu citra grayscale (gambar keabuan), metode yang umum digunakan, yaitu:

$$\frac{R + G + B}{3}$$

Dimana :

R : Unsur warna Merah

G : Unsur warna Hijau

B : Unsur warna Biru

Nilai yang dihasilkan dari persamaan diatas akan diinputkan ke masing-masing unsur warna dasar citra grayscale [14].

Proses pengembangan (thresholding) citra grayscale secara umum menggunakan rumus :

$$g(x,y) = \begin{cases} 1 & \text{jika } f(x,y) \geq T \\ 0 & \text{jika } f(x,y) < T \end{cases}$$

Nilai $g(x,y)$ adalah citra biner yang dihasilkan dari citra grayscale $f(x,y)$, nilai T adalah nilai ambang. Nilai T sangat menentukan kualitas citra biner yang dihasilkan.

2.3 Deteksi Tepi (Edge Detection)

Deteksi tepi (*Edge Detection*) adalah teknik yang digunakan untuk mendeteksi tepi dari objek dalam citra. Proses ini dilakukan dengan cara menelusuri citra secara vertikal dan horizontal untuk menemukan perbedaan nilai yang signifikan antara suatu piksel dengan piksel tetangganya. Suatu titik dikatakan tepi (*edge*) dari suatu citra bila titik tersebut mempunyai perbedaan yang tinggi dengan tetangganya. Deteksi tepi digunakan untuk menyaring citra dengan mengurangi banyaknya data tapi tidak menghilangkan informasi penting yang terdapat didalamnya (McIlhagga, 2018). Deteksi tepi akan memisahkan objek dengan latarnya. Faktor kunci dalam mengekstraksi ciri adalah kemampuan mendeteksi keberadaan tepi (*edge*) dari objek di dalam citra. Pendeteksian tepi merupakan langkah pertama untuk melingkupi informasi di dalam citra. Tepi mencirikan batas-batas objek dan karena itu tepi berguna untuk proses segmentasi dan identifikasi di dalam citra. Tujuan pendeteksian tepi adalah untuk menandai bagian yang menjadi detail citra dan untuk memperbaiki detail dari citra yang kabur, yang terjadi karena error atau adanya efek dari proses akuisisi citra (Yao Qin, 2019). Keuntungan tambahan dari deteksi tepi adalah sudut visualisasi data keluaran. Deteksi tepi menyediakan jelas gambar dengan warna yang menunjukkan sudut serat, yang dapat mudah ditafsirkan untuk melihat dominan sudut serat keselarasan (B. Wang, 2018). Macam – macam metode untuk proses deteksi tepi ini antara lain: metode *Sobel*, *Prewitt*, *Robert*, *Laplacian of a Gaussian*, *Canny*, dan lain-lain (Luyang Wang, 2019).

2.4 Segmentasi Plat

Segmentasi citra adalah proses membagi citra digital menjadi beberapa daerah atau kelompok, dimana masing-masing daerah terdiri dari sekumpulan piksel. Segmentasi citra menyederhanakan dan merubah representasi citra ke sesuatu yang lebih bermakna dan lebih mudah untuk dianalisis. Segmentasi citra di-gunakan untuk mencari obyek yang ingin di cari dan batas-batas bentuk objek seperti garis, kurva dalam citra. Tujuan segmentasi citra adalah untuk mempartisi gambar menjadi beberapa wilayah yang tidak tumpang tindih dengan karakteristik yang homogen, seperti intensitas, warna, dan tekstur. Seiring dengan berkembangnya teknologi pada aplikasi yang memproses sebuah obyek seperti rekonstruksi tiga dimensi, pengenalan benda, pengenalan tulisan, deteksi wajah, pengkodean obyek dan lain-lain maka proses segmentasi menjadi semakin diperlukan. Hasil segmentasi juga harus semakin akurat karena ketidak akuratan hasil segmentasi akan mempengaruhi pula hasil proses selanjutnya (Feng-Ping Ana, 2019).

Pola merupakan suatu entitas yang terdefinisi dan dapat diidentifikasi dan diberi nama. Pola merupakan kumpulan hasil pengukuran yang bisa dinyatakan dengan notasi vektor atau matriks. Pemisahan ciri pada citra dilakukan berdasarkan blok. Pemisahan ciri berdasarkan blok biasanya digunakan setelah pemrosesan sebelumnya seperti proses deteksi tepi atau deteksi garis. Citra yang telah mengalami proses deteksi tepi dibagi menjadi beberapa blok. Pembagian blok dapat dilakukan dengan dua model yaitu pembagian blok saling tumpang tindih dan pembagian blok tidak saling tumpang tindih. Vektor dihitung dengan cara membagi citra kedalam blok berukuran tertentu sehingga akan menghasilkan blok dengan ukuran kecil. Setiap blok dilakukan penghitungan piksel. Jika dalam blok terdapat piksel dengan nilai 1 dengan jumlah minimal yang ditentukan, maka blok itu akan diberi nilai 1. Dan jika jumlah piksel bernilai 1 sebanyak kurang dari jumlah yang ditentukan, maka blok itu akan diberi nilai 0. Vektor yang dihasilkan memiliki elemen dengan nilai 0 atau 1. Proses ini bertujuan untuk menurunkan keragaman ukuran karakter angka pada nomor polisi yang disebabkan karena jarak pengambilan foto atau bentuk *style* karakter yang berbeda (Alberto Silvio Chiappa, 2019). Setelah menemukan nomor plat, karakter diperiksa untuk proses lebih lanjut. Tahapan segmentasi plat adalah sebagai berikut:

- 2.4.1 Mengindex semua objek yang terdapat pada image
- 2.4.2 Mencari luas masing-masing objek
- 2.4.3 Mencari objek yang mempunyai luas sebagai plat nomor

Tujuan dari proses ini adalah untuk mengekstrak plat nomor dari gambar yang diambil. Output dari proses ini adalah gambar plat nomor yang tepat pada plat nomornya. Terdapat dua pendekatan utama dalam segmentasi citra yaitu didasarkan pada tepi (*edge based*) dan didasarkan pada wilayah (*region based*) (Falang Shen, 2018). Segmentasi didasarkan pada tepi membagi citra berdasarkan diskontinuitas di antara subwilayah (*sub-region*), sedangkan segmentasi yang didasarkan pada wilayah bekerjanya berdasarkan keseragaman yang ada pada sub-wilayah tersebut. Hasil dari segmentasi citra adalah sekumpulan wilayah yang melingkupi citra tersebut, atau sekumpulan kontur yang diekstrak dari citra (pada deteksi tepi). Tiap piksel dalam suatu wilayah mempunyai kesamaan karakteristik atau properti yang dapat dihitung (*computed property*) seperti : warna (*color*), intensitas (*intensity*), dan tekstur (Breve, 2019) (Jonathan D. Victor, 2019).

2.5 Morphology

Proses *morfologi* merupakan suatu alat untuk mengekstraksi komponen citra yang dapat digunakan dalam representasi dan deskripsi dari suatu obyek dalam citra. Dalam kesempatan ini proses akan diawali dengan transisi dari beberapa proses terhadap citra keluaran untuk melakukan proses terhadap citra tersebut

seperti intensitas maupun tingkat skala keabuan terhadap suatu titik (*pixel*). Proses tingkat terendah (*low level process*) seperti pra pemrosesan citra untuk mengurangi noise (tampilan yang mengganggu citra), peningkatan kontras serta penajaman citra, sementara itu untuk tingkat menengah (*mid level process*) diantaranya adalah segmentasi (pembagian citra menjadi beberapa area pengamatan maupun obyek) dan klasifikasi (pengenalan) obyek dalam suatu citra. Untuk proses tingkat menengah ini menghasilkan suatu karakter berdasarkan kenyataan citra yang diberikan, namun keluarannya adalah beberapa atribut ekstraksi citra. Sedangkan untuk proses tingkat tinggi (*higher level process*) terhadap suatu citra adalah kemampuan untuk "mengenali" obyek obyek yang dikenalnya dari suatu citra, analisis citra hingga integrasi beberapa fungsi kognitif secara normal maupun dengan visi (*vision*).

2.6 Thresholding

Thresholding adalah suatu proses yang digunakan untuk menghasilkan citra biner yaitu citra yang memiliki dua nilai tingkat keabuan yaitu hitam dan putih bergantung apakah piksel asli tersebut lebih besar atau lebih kecil dari nilai ambang batas (T). Piksel akan diubah menjadi putih jika nilai tingkat keabuannya lebih besar dari nilai T dan akan diubah menjadi hitam jika nilai tingkat keabuannya lebih kecil atau sama dengan T (Seyed Jaleddin Mousavirad, 2019).

Thresholding digunakan untuk mengatur jumlah derajat keabuan yang ada pada citra. Dengan menggunakan *thresholding* maka derajat keabuan bisa diubah sesuai keinginan, misalkan diinginkan menggunakan derajat keabuan 16, maka tinggal membagi nilai derajat keabuan dengan 16. Proses *thresholding* ini pada dasarnya adalah proses pengubahan kuantisasi pada citra, sehingga untuk melakukan *thresholding* dengan derajat keabuan dapat digunakan rumus:

$$x = b.\text{int}\left(\frac{w}{b}\right)$$

dimana :

w = nilai derajat keabuan sebelum *thresholding*

x = nilai derajat keabuan setelah *thresholding*

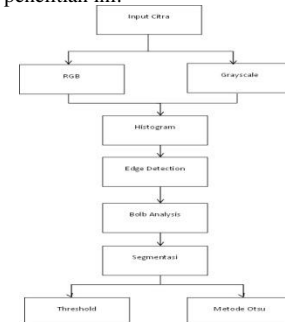
Dari citra *Grayscale*, *thresholding* dapat digunakan untuk membentuk citra biner. Sebuah citra biner adalah sebuah citra digital yang hanya memiliki dua kemungkinan nilai untuk tiap pixel. Kedua warna tersebut adalah hitam dan putih. Warna yang digunakan untuk object dalam citra adalah warna *foreground* sedangkan sisa dari citra adalah warna *background* (Mohamed Abd Elaziz, 2019). Intensitas *threshold* dalam membedakan mana objek yang lebih baik dari *background* gambar , dan pada kebanyakan kasus , pilihan *threshold* membutuhkan penilaian subjektif dari manusia sebagai operatornya.

2.7 Metode Otsu

Metode *Otsu* merupakan metode dalam segmentasi citra yang menghitung nilai ambang T secara otomatis berdasarkan citra masukan. Pendekatan yang digunakan oleh metode *Otsu* adalah dengan analisis diskriminan yaitu menentukan suatu variabel yang dapat membedakan antara dua atau lebih kelompok yang muncul secara alami. Analisis diskriminan akan memaksimalkan variabel tersebut agar dapat membagi objek latar depan dan latar belakang (Ta Yang Goh, 2018). Metode *Otsu* digunakan secara otomatis untuk membentuk *thresholding* citra berdasarkan bentuknya. Dalam metode *Otsu* kita mencari nilai *threshold* yang meminimalkan variasi intra-class, didefinisikan sebagai penjumlahan bobot dari variasi dari kedua kelas (M. Malarvel, 2017).

3. Hasil dan Pembahasan

Pada penelitian ini, dilakukan pengolahan citra dengan beberapa metode yang telah disebutkan pada bab sebelumnya. Berikut adalah diagram alir proses pengolahan citra pada penelitian ini:



Gambar 1. Diagram Alir Pengolahan Citra

3.1 Input Citra

Penelitian ini menggunakan data masukan berupa file citra. Pada Matlab, file ini dipanggil dengan fungsi *imread* dan diikuti dengan nama filenya.

3.2 Konversi RGB

Selanjutnya citra di konversi berdasarkan level warna pada *Red*(merah), *Green*(Hijau), *Blue*(Biru). Berikut citra asli dan hasil pengolahan menurut RGB:



Gambar 2. Citra Asli



Gambar 3. Citra RGB

Fungsi yang digunakan pada proses ini adalah:

```
gambar=imread('Image001.jpg');
red=gambar(:,:,1);
green=gambar(:,:,2);
blue=gambar(:,:,3);
imshow(gambar)
imshow(red)
imshow(green)
imshow(blue)
```

3.3 Konversi RGB ke Grayscale

Setelah didapatkan hasil RGB, citra bisa dikonversi menjadi Grayscale. Pada perubahan sebuah gambar menjadi grayscale dapat dilakukan dengan cara mengambil semua pixel pada gambar kemudian warna tiap pixel akan diambil informasi mengenai 3 warna dasar yaitu merah, biru dan hijau (melalui fungsi *gray2*), ketiga warna dasar ini akan dijumlahkan. Berikut citra asli dan hasil pengolahannya:



Gambar 4. Citra Asli



Gambar 5. Citra Grayscale

Fungsi yang digunakan pada proses ini adalah:

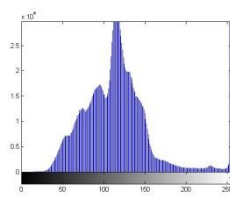
```
gambar=imread('Image002.jpg');
red=gambar(:,:,1);
green=gambar(:,:,2);
blue=gambar(:,:,3);
gray2=0.3*red+0.5*green+0.2*blue ;
imshow(gambar)
imshow(gray2)
```

3.4 Histogram

Pada proses ini, citra diubah menjadi bentuk histogram. Berikut citra asli dan histogramnya:



Gambar 6. Citra Asli



Gambar 7. Histogram

Histogram asli dari suatu citra seringkali tidak menempati seluruh tingkat gradasi yang tersedia. Histogram diperoleh dengan menghitung jumlah kemunculan dari setiap nilai piksel, yang kemudian dipetakan terhadap nilai pikselnya (intensitas). Berikut adalah fungsi pada matlab yang digunakan untuk menampilkan histogram pada citra:

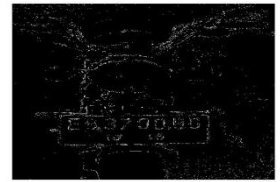
```
gambar=imread('Image002.jpg');
red=gambar(:,:,1);
green=gambar(:,:,2);
blue=gambar(:,:,3);
merahgray2=0.3*red+0.5*green+0.2*blue ;
imhist(red)
imhist(green)
imhist(blue)
imhist(gray)
```

3.5 Edge Detection

Tepi atau sisi dari sebuah obyek adalah daerah di mana terdapat perubahan intensitas warna yang cukup tinggi. Proses deteksi tepi (*edge detection*) akan melakukan konversi terhadap daerah ini menjadi dua macam nilai yaitu intensitas warna rendah atau tinggi, contoh bernilai nol atau satu. Deteksi tepi akan menghasilkan nilai tinggi apabila ditemukan tepi dan nilai rendah jika sebaliknya. Berikut adalah hasil pengolahan citra dengan metode *Edge Detection*:



Gambar 8. Edge Detection dengan Metode Prewitt



Gambar 9. Edge Detection dengan Metode Sobel

Berdasarkan hasil penelitian yang dilakukan, ditemukan 2 metode yang hampir mendekati pendeteksian tepi objek secara keseluruhan. Kedua metode ini adalah metode *Prewitt* dan metode *Sobel*. Fungsi untuk menggunakan metode *Prewitt* pada matlab adalah:

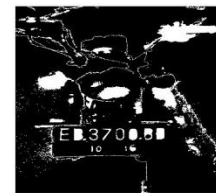
```
g = imread('Image002.jpg');
gray=rgb2gray(g);
x = edge(gray,'prewitt');
imshow(x);
```

Dan fungsi untuk menggunakan metode *Sobel* pada matlab adalah:

```
g = imread('Image002.jpg');
gray=rgb2gray(g);
x = edge(gray,'sobel');
imshow(x);
```

3.6 Segmentasi

Pada proses ini harus melalui berbagai tahapan, seperti konversi RGB ke *Grayscale*, mengubah citra ke dalam *binary image*, *Edge Detection*, *Morfology*, dan *Blob Analysis*. Warna pada *background* akan dibuat semakin gelap sehingga objek dengan letak di bagian depan akan makin jelas terlihat. Berikut hasilnya:



Gambar 10. Hasil Segmentasi Citra

3.7 Thresholding dan Metode Otsu

Metode ini dilakukan untuk mengubah derajat keabuan citra pada proses segmentasi. Setelah derajat keabuan dipertajam, maka proses berlanjut dengan menggunakan metode Otsu. Warna pada *background* citra akan dibuat semakin cerah dan objek utamanya akan berubah menjadi semakin gelap. Berikut hasilnya:



Gambar 11. Hasil Segmentasi citra dengan Thresholding dan metode Otsu

4 Simpulan

Berdasarkan hasil penelitian ini dapat disimpulkan bahwa:

1. *Thresholding* digunakan untuk mengatur jumlah derajat keabuan yang ada pada citra.
 2. Kombinasi metode pada segmentasi yaitu konversi RGB ke *Grayscale*, mengubah citra ke dalam *binary image*, *Edge Detection*, *Morfology*, dan *Blob Analysis* akan mengubah ketajaman *background* menjadi semakin gelap dan mempertajam derajat keabuan pada objek utama.
 3. *Thresholding* dengan menggunakan metode *Otsu* mempertajam derajat keabuan pada *background* citra dan mempertajam warna gelap pada objek utama.
 4. *Edge Detection* dengan metode *Prewitt* dan metode *Sobel* lebih baik daripada metode *Edge Detection* yang digunakan pada penelitian ini.
- Pada penelitian ini masih banyak kekurangan dan belum banyak membuktikan metode-metode pendeteksian objek plat kendaraan. Masih banyak faktor-faktor penentu yang belum dilakukan seperti citra blur, jarak pengambilan citra, ukuran citra, konversi tipe data citra, dan lain-lain. Diharapkan penelitian ini setidaknya menjadi referensi tambahan untuk penelitian-penelitian lainnya supaya bisa meningkatkan kemampuan dan kehandalan aplikasi pengolahan citra sejenis ini.

Pustaka

- Alberto Silvio Chiappa, S. M. (2019). Mesh adaptation-aided image segmentation. *Commun Nonlinear Sci Numer Simulat*, 147-166.
- B. Wang, L. C. (2018). A novel method on the edge detection of infrared image. *Optik*.

- Breve, F. (2019). Interactive image segmentation using label propagation through complex networks. *Expert Systems With Applications*, 18-33.
- Falong Shen, G. Z. (2018). Semantic Image Segmentation via Guidance of Image Classification. *Neurocomputing*.
- Feng-Ping Ana, Z.-W. L. (2019). Medical image segmentation algorithm based on feedback mechanism convolutional neural network. *Biomedical Signal Processing and Control*.
- Imelda, A. H. (2012). Klasifikasi Kendaraan Menggunakan LEARNING. *IJEIS*, 57-66.
- Jonathan D. Victor, S. M. (2019). Image segmentation driven by elements of form. *Vision Research*, 21-34.
- Luyang Wang, Y. S. (2019). An Accurate and Efficient Multi-Category Edge Detection Method. *Cognitive Systems Research*.
- M. Malarvel, G. S. (2017). An improved version of Otsu's method for segmentation of weld defects on X-radiography images. *Optik - International Journal for Light and Electron Optics*.
- McIlhagga, W. (2018). Estimates of edge detection filters in human vision. *Vision Research*, 30-36.
- Mellolo, O. (2012). Pengenalan plat nomor polisi kendaraan. *Jurnal ilmiah sains*.
- Mohamed Abd Elaziz, S. L. (2019). Many-objectives Multilevel Thresholding Image Segmentation using Knee Evolutionary Algorithm. *Expert Systems With Applications*.
- Nelson Diazb, C. H. (2019). Adaptive grayscale compressive spectral imaging using optimal blue noise coding patterns. *Optics and Laser Technology*, 147-157.
- S. Mas, A. T. (2019). Use of physiological information based on grayscale images to improve mass spectrometry imaging data analysis from biological tissues. *Analytica Chimica Acta*.
- Seyed Jalaleddin Mousavirad, H. E.-K. (2019). Human mental search-based multilevel thresholding for image segmentation. *Applied Soft Computing Journal*.
- Ta Yang Goh, S. N. (2018). Performance analysis of image thresholding: Otsu technique. *Measurement*, 298-307.
- Yao Qin, Y. W. (2019). Method for pests detecting in stored grain based on spectral residual saliency edge detection. *Grain & Oil Science and Technology*.