

Aplikasi Pengenal Citra Nomor Kendaraan Bermotor Menggunakan Metode *Template Matching*

Bayu Sy. Kurniawan, Steven Ray Sentinuwo, Oktavian A. Lantang
Teknik Informatika Universitas Sam Ratulangi Manado, Indonesia.
bayulasahido@gmail.com, steven@unsrat.ac.id, oktavianlantang@gmail.com

Abstrak - Untuk pengelolaan jasa parkir suatu kawasan pertokoan dan pusat perbelanjaan biasanya dikelola oleh suatu perusahaan pengelola parkir yang profesional, namun membutuhkan sejumlah karyawan pada setiap pos untuk memasukan data plat nomor kendaraan secara manual. Proses secara manual ini mempengaruhi lama waktu dan akurasi yang diakibatkan oleh faktor manusia. Untuk mempermudah kinerja, dibutuhkan sebuah aplikasi berbasis *Computer Vision* menggunakan pengolahan *Image Processing* yang dapat membaca nomor kendaraan secara otomatis. Skripsi ini dibuat untuk merancang dan mengimplementasikan aplikasi yang dapat mengenali karakter angka dan huruf plat nomor kendaraan bermotor pada citra hasil pemotretan kamera *digital* menggunakan metode *template matching*. Yang nantinya dapat mempermudah kinerja pengelola jasa parkir. Berdasarkan hasil implementasi terhadap 30 sampel citra mobil dengan jumlah 238 karakter didapatkan akurasi 80,25%. Dengan pengujian pada citra ideal yang didapatkan berdasarkan jarak pengambilan gambar dan kecerahan, diperoleh 22 citra dengan 176 karakter menghasilkan akurasi 97,77%.

Kata kunci : *Computer Vision, Image Processing, Nomor Kendaraan Bermotor*

I. PENDAHULUAN

Di kota Manado, untuk pengelolaan jasa parkir suatu kawasan pertokoan dan pusat perbelanjaan biasanya dikelola oleh suatu perusahaan pengelola parkir yang profesional. Peralatan yang digunakan dan prosedur yang dijalankan sudah memadai, serta sistem yang digunakan sudah cukup kompleks dan terstruktur dikarenakan sudah terintegrasi dengan *database* yang terpusat pada *server* yang dimiliki namun tetap membutuhkan cukup banyak tenaga kerja (karyawan).

Contohnya sistem parkir yang ada pada kawasan Megamas Manado. Kendaraan akan mendapat tiket parkir saat masuk, dan nantinya akan diberikan lagi saat keluar. Untuk mengkonfirmasi kebenaran pemilik karcis dengan kendaraan, dibutuhkan pemasukan data nomor kendaraan. Saat kendaraan keluar, operator akan mencocokkan antara kendaraan, karcis, dan foto yang diambil saat kendaraan masuk. Untuk itu pada setiap pos terjadi penginputan tersebut dengan pengambilan gambar, namun penginputan dilakukan secara manual oleh operator/karyawan pada komputer di pos yang jaringannya sudah terhubung dengan server.

Hal ini yang akan diteliti penulis, yaitu bagaimana caranya agar proses penginputan dilakukan otomatis oleh

aplikasi yang dibantu dengan kamera yang sudah tersedia. Tentu mengurangi kinerja dan dapat menekan biaya pada penggunaan karyawan, serta memaksimalkan penggunaan alat yang tersedia. Dimana dalam sistem dibutuhkan suatu program yang dapat membaca nomor kendaraan melalui citra kendaraan yang diambil oleh kamera. Data citra tersebut nantinya akan diproses sedemikian rupa, sehingga informasi nomor kendaraan dapat terdeteksi dengan sendirinya oleh sistem. Dalam perancangan dan pembuatan aplikasi ini, terdapat beberapa langkah, yakni :

- Penentuan lokasi plat nomor dari citra kendaraan bermotor
- Persiapan pengolahan citra area plat nomor
- Pengenalan masing-masing karakter

II. LANDASAN TEORI

A. Peraturan Penomoratan Plat Nomor Mobil Pribadi

Di Indonesia sebuah plat nomor pribadi memiliki warna dasar hitam dengan karakter berwarna putih [4], sedangkan pola karakter di dalamnya memiliki kesamaan dengan pola kendaraan umum, dengan pola sebagai berikut di mana :

- KA : Kode Area mobil tersebut berupa huruf dengan maksimum jumlah digit 2 dan minimum 1, seperti B untuk Jakarta dan sekitarnya, AB untuk Yogyakarta, DB untuk Minahasa, dan lain-lain.
- NP : Nomor Plat mobil tersebut berupa huruf dengan maksimum jumlah digit 4 dan minimum 1, seperti 1, 5685, 9373, dan lain-lain.
- KT : Karakter Tambahan yang bias ada atau tidak (opsional), berupa huruf dengan batas maksimum 3 digit dan minimum 0, seperti A, BS, SGD, dan lain – lain.

B. Citra

Citra atau *image* adalah angka, dari segi estetika, citra atau gambar adalah kumpulan warna yang bisa terlihat indah, memiliki pola, berbentuk abstrak dan lain sebagainya [5]. Citra dapat berupa foto udara, penampang lintang (*cross section*) dari suatu benda, gambar wajah, hasil tomografi otak dan lain sebagainya. Dari segi ilmiah, citra adalah gambar 3-dimensi (3D) dari suatu fungsi, biasanya intensitas warna sebagai fungsi spatial x dan y . Di komputer, warna dapat dinyatakan, misalnya sebagai angka dalam bentuk skala RGB. Karena citra adalah angka, maka citra dapat diproses secara digital.

C. Citra Digital

Sebuah citra digital dapat diwakili oleh sebuah matriks dua dimensi $f(x,y)$ yang terdiri dari M kolom dan N baris, dimana perpotongan antara kolom dan baris disebut piksel atau elemen terkecil dari sebuah citra [2].

D. Citra RGB

Untuk citra berwarna maka digunakan model RGB (*Red-Green-Blue*), satu citra berwarna dinyatakan sebagai 3 buah matrik *grayscale* yang berupa matrik untuk *Red (R-layer)*, matrik *Green (G-layer)* dan matrik untuk *Blue(B-layer)*. *R-layer* adalah matrik yang menyatakan derajat kecerahan untuk warna merah (misalkan untuk skala keabuan 0-255, nilai 0 menyatakan gelap (hitam) dan 255 menyatakan merah. *G-layer* adalah matrik yang menyatakan derajat kecerahan untuk warna hijau, dan *B-layer* adalah matrik yang menyatakan derajat kecerahan untuk warna biru. Dari definisi tersebut, untuk menyajikan warna tertentu dapat dengan mudah dilakukan, yaitu dengan mencampurkan ketiga warna dasar RGB[5].

E. Citra Grayscale

Citra *grayscale* atau citra beraras keabuan adalah citra yang hanya menggunakan warna yang merupakan tingkatan warna abu-abu. Warna abu-abu adalah satu-satunya warna pada ruang RGB dengan komponen merah, hijau, dan biru mempunyai intensitas yang sama. Pada citra beraras keabuan hanya perlu menyatakan nilai intensitas untuk tiap piksel sebagai nilai tunggal, sedangkan pada citra berwarna perlu tiga nilai intensitas untuk tiap pikselnya[6]. Intensitas citra beraras keabuan disimpan sebagai integer 8 bit sehingga memberikan $2^8 = 256$ tingkat keabuan dari warna hitam sampai warna putih. Dengan menggunakan pola 8-bit ini citra beraras keabuan membutuhkan ruang memori dan waktu pengolahan yang lebih sedikit daripada citra berwarna (RGB).

F. Citra Biner

Citra Biner (*binary image*) adalah citra yang hanya mempunyai dua nilai derajat keabuan yaitu hitam dan putih. Oleh karena itu, setiap piksel pada citra biner cukup direpresentasikan dengan 1 bit[1].

G. Computer Vision

Computer vision adalah proses otomatis yang mengintegrasikan sejumlah besar proses untuk persepsi visual, seperti akuisisi citra, pengolahan citra, pengenalan dan pembuatan keputusan. *Computer vision* mencoba meniru cara kerja sistem visual manusia (*human vision*) yang sesungguhnya sangat kompleks, bagaimana manusia melihat objek diteruskan ke otak untuk diinterpretasi sehingga manusia mengerti objek apa yang tampak dalam pandangan mata. Selanjutnya hasil interpretasi ini digunakan untuk pengambilan keputusan[7]. Visi computer ini terdiri dari 2 bidang ilmu yaitu:

- Pengolahan Citra, ilmu yang mempelajari bagaimana memperbaiki kualitas citra untuk mendapatkan hasil yang mudah untuk dikenali manusia atau mesin
- Pengenalan Pola, merupakan proses pengelompokan data numeric dan simbolik (termasuk citra) secara otomatis, yang bertujuan untuk dapat mengidentifikasi objek pada citra.

H. Pengolahan Citra

Pengolahan citra atau *Image Processing* adalah suatu sistem dimana proses dilakukan dengan masukan (*input*) berupa citra (*image*) dan hasilnya (*output*) juga berupa citra (*image*). Pada awalnya pengolahan citra ini dilakukan untuk memperbaiki kualitas citra, namun dengan

berkembangnya dunia komputasi yang ditandai dengan semakin meningkatnya kapasitas dan kecepatan proses komputer, serta munculnya ilmu - ilmu komputer yang memungkinkan manusia dapat mengambil informasi dari suatu citra maka *image processing* tidak dapat dilepaskan dengan bidang *computer vision*[5].

I. Konversi Citra

Pada umumnya setiap pemrosesan citra mempunyai pola awal yang hampir sama, yakni pemrosesan diawali oleh konversi terhadap citra berwarna atau citra RGB ke bentuk akhir citra biner namun melalui konversi ke bentuk *grayscale*. Dikarenakan proses pengolahan citra dengan citra biner lebih mudah untuk diproses ke tahap selanjutnya.

Bermula dari citra RGB, masing-masing warna memiliki range 0 - 255, maka totalnya adalah $255^3 = 16.581.375$. Konversi diawali dari citra RGB ke bentuk citra *grayscale*, citra yang setiap pikselnya mempunyai warna gradasi mulai dari putih sampai hitam. Rentang tersebut berarti bahwa setiap piksel dapat diwakili oleh 8 bit, atau 1 byte. Citra *grayscale* sebenarnya merupakan hasil rata-rata dari citra RGB.

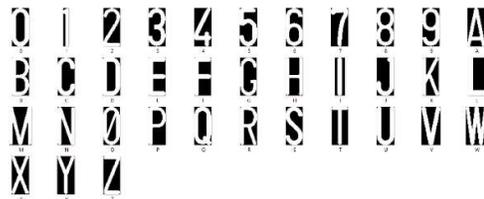
J. Template Matching

Template matching adalah sebuah teknik dalam pengolahan citra digital untuk menemukan bagian-bagian kecil dari gambar yang cocok dengan template gambar. Metode *Template matching* merupakan salah satu metode yang digunakan untuk menjelaskan bagaimana otak kita mengenali kembali bentuk-bentuk atau pola-pola[3].

III. METODOLOGI PENELITIAN

A. Metode Pengumpulan Data

Data *font* yang di dapatkan berupa citra seperti terlihat pada Gambar 1. Untuk mendapatkan data set plat nomor, setiap karakter di potong dan dikonversikan ke bentuk citra biner. Menghasilkan dataset *template* karakter. Seperti terlihat pada gambar berikut :



Gambar 1. Dataset Karakter

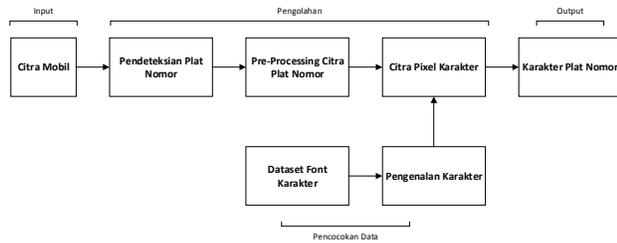
Data selanjutnya yang dikumpulkan adalah data citra mobil, digunakan nantinya pada pengujian aplikasi. Jenis mobil yang diambil gambarnya ialah mobil yang mempunyai plat nomor sesuai standar peraturan penomoran plat nomor di Indonesia. Waktu pengambilan data dilakukan pada siang hari, sehingga tidak dibutuhkan cahaya bantuan. Sesuai pengamatan, demi mendapatkan hasil yang tetap penulis juga mempunyai standar dalam pengambilan data. Standar dalam pengambilan data memiliki rentang ukuran untuk menyesuaikan kondisi pencahayaan dan jarak. Untuk jarak penulis memperkirakan antara 2 hingga 4 meter. Posisi kamera

tidak menentu agar dapat mencari posisi dan jarak ideal dalam mengambil citra foto.

Kualitas citra juga mempengaruhi proses pengolahan citra. Untuk mendapatkan hasil yang sama dengan kualitas terbaik dilakukan beberapa pengaturan manual pada kamera. Pengambilan data dilakukan pada siang hari, atau di kondisi sedang terang. Di kondisi siang hari pengambil gambar menggunakan ISO 100-200. ISO di jaga tetap rendah agar citra foto yang dihasilkan tidak memiliki banyak *noise*, yang nantinya mengurangi kualitas citra foto. Ada juga yang disebut *Aperture* yaitu pengaturan bukaan lensa diatur sekitar f4.5, dan *Shutter Speed* yang merupakan kecepatan pengambilan gambar diatur sekitar 100 hingga 400 fps (*frame per second*).

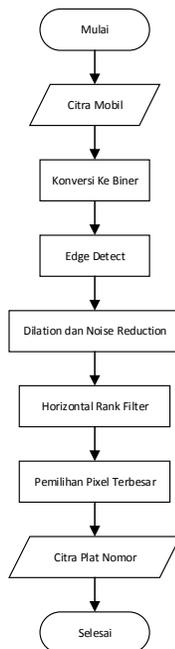
B. Perancangan Sistem

Secara garis besar cara kerja dari sistem yang akan dibuat terlihat pada Gambar 2. Dimana bentuk data masukan berupa citra kendaraan, dan informasi yang dihasilkan berupa teks nomor kendaraan.



Gambar.2 Blok Diagram cara kerja sistem

Dari keseluruhan proses, sistem yang dikembangkan dibagi atas 3 tahap utama, proses pendeteksian plat nomor, pre-processing citra plat nomor, dan pengenalan karakter.

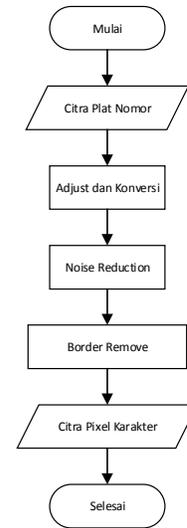


Gambar.3 Diagram Alur Pendeteksian Plat Nomor

Diawali dengan pendeteksian plat nomor, terlihat pada Gambar.3 dimana program diawali dengan proses memasukan citra kendaraan. Keseluruhan data sampel

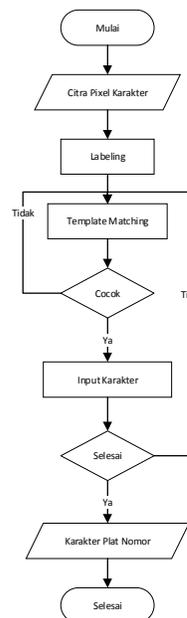
merupakan citra kendaraan utuh dengan citra plat nomor didalamnya. Tujuan tahap ini untuk mendeteksi dan memisahkan citra plat dari kendaraannya. Berikut merupakan alur proses dalam tahapan ini.

Citra plat nomor sudah didapatkan pada tahap ini, selanjutnya dilakukan tahap *pre-processing* bertujuan menjadi tahap awal sebelum menjutahapan pendeteksian karakter. Dibutuhkan citra biner yang tanpa gangguan agar nantinya proses template matcing tidak terganggu. Berikut pada Gambar.4 merupakan alur proses dalam tahap *pre-processing*.



Gambar.4 Diagram Alur Pre-Processing Citra Plat Nomor

Setelah citra telah melalui tahap diatas, seharusnya citra hanya berisi kumpulan piksel karakter yang akan di proses untuk pencocokan. Disediakan dataset karakter berbentuk citra biner, citra karakter yang dibaca menjadi pola dan dicocokkan dengan *template*. Seperti yang terlihat pada Gambar.5 berikut.



Gambar.5 Diagram Alur Pengenalan Karakter

IV. HASIL DAN PEMBAHASAN

A. Pengujian Sistem

Sistem akan dijalankan dengan data sampel yang telah di kumpulkan berjumlah 30 citra (*image*), yang diambil dengan standar yang telah disebutkan dalam metode pengumpulan data. Citra kendaraan terkadang diambil dari sisi depan, dan sebagian dari sisi belakang. Sebagian besar data memiliki jarak antara 2 meter seperti pada contoh Gambar.6, yakni kendaraan yang pada citra berada *full frame* pada citra. Ada juga yang tergolong pada jarak 3 meter dan 4 meter Posisi kendaraan biasanya berada di tengah, namun terdapat objek lain di sekelilingnya pada citra.



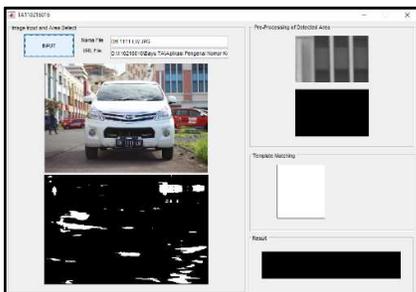
Gambar.6 Data sampel dengan jarak 2 meter

Tujuan pengujian ini adalah untuk mengetahui keberhasilan dan kegagalan dari sistem. Diambil beberapa sampel dengan rentang jarak dimaksudkan untuk menentukan posisi ideal pengambilan data. Serta yang terpenting untuk mengukur keakuratan, dan nantinya dapat ditarik kesimpulan sesuai pengamatan dari pengujian ini.

Contoh pada gambar terlihat pendeteaksian nomor kendaraan berhasil keseluruhan. Untuk citra dengan jarak 2 hingga 3 meter contohnya pada Gambar.7 kebanyakan dari hasilnya cukup baik. Namun tetap terdapat beberapa kesalahan.



Gambar.7 Uji coba citra jarak 2 meter



Gambar.8 Uji coba citra jarak 4 meter

Untuk jarak 4 meter seperti pada Gambar.8, lebih terlihat kegagalan dalam mendeteksi. Pendeteksian plat nomor pada umumnya gagal karena tidak dianggap sebagai piksel terbesar dalam sistem. Hal ini dipengaruhi oleh objek sekeliling yang mencolok pada citra, berdasarkan pengamatan hal ini dapat mengganggu proses pendeteksian. Pada contoh Gambar 4.11 saja yang terdeteksi sebagai piksel terbesar, namun tidak semua area plat nomor terambil dan mengakibatkan karakter yang dibaca hanya sebagian saja.

B. Hasil Implementasi Data

Keseluruhan data telah diuji untuk ditemukan keakuratan proses pendeteksian nomor kendaraan. Untuk penghitungan, dihitung berdasarkan tiap – tiap karakter pada citra sampel. Untuk perhitungan data sebagai berikut.

$$\text{Persentase Keberhasilan} = \frac{\sum \text{Karakter Benar}}{\sum \text{Jumlah Karakter}} \times 100\% \quad (1)$$

$$\text{Persentase Kegagalan} = \frac{\sum \text{Karakter Salah}}{\sum \text{Jumlah Karakter}} \times 100 \quad (2)$$

Hasil dari perhitungan pada Tabel.1 akan dicari nilai akurasi menggunakan persamaan (1) dan (2). Hasil dari perhitungan persamaan (3) menunjukkan akurasi pendeteksian karakter citra mobil adalah 80,25%. Sebagian besar data yang terbaca dengan baik adalah citra yang berjarak 2-3m. Dan juga terlihat aspek kegagalan pada perhitungan (4) yang berjumlah 19,75% didominasi oleh citra yang tergolong pada jarak 4m. Jadi bisa disimpulkan jarak citra ideal dalam pengambilan data adalah 2 hingga 3 meter.

Tabel.1 Hasil Implementasi Seluruh Data

No	Plat	Jumlah Karakter	Jarak	Terdeteksi Piksel Terbesar	Terdeteksi Seluruh Karakter	Karakter		
						Terbaca	Benar	Salah
1	DB1056QE	8	2m	Ya	Ya	DB1056QE	8	0
2	DB1071AX	8	2m	Ya	Ya	DB1071AX	8	0
3	DB1074LC	8	2m	Ya	Ya	DB1074LC	8	0
4	DB1111LW	8	4m	Tidak	Tidak	-	0	8
5	DB1179AO	8	2m	Ya	Ya	DB1179AO	8	0
6	DB1208AS	8	3m	Ya	Ya	DB1208AS	8	0
7	DB1277AV	8	2m	Ya	Ya	DB1277AM	7	1
8	DB1308LC	8	2m	Ya	Ya	DB1308LC	8	0
9	DB1324QA	8	3m	Ya	Ya	DB1324QA	8	0
10	DB1327AQ	8	4m	Tidak	Tidak	-	0	8
11	DB1399AE	8	2m	Ya	Tidak	130QAE	4	4
12	DB1402AZ	8	2m	Ya	Ya	DB1402AZ	7	1
13	DB1514AL	8	2m	Ya	Ya	DB1514AL	8	0
14	DB1534AS	8	2m	Ya	Ya	DB1534AS	8	0
15	DB1591CL	8	4m	Ya	Tidak	B1591	5	3
16	DB1593MB	8	2m	Ya	Ya	DB1593MB	8	0
17	DB1641AF	8	2m	Ya	Ya	DB1641AF	8	0
18	DB1921JA	8	3m	Ya	Ya	DB1921QA	7	1
19	DB1969MC	8	2m	Ya	Ya	DB1969MC	8	0
20	DB1986AL	8	2m	Ya	Ya	DB1986AL	8	0
21	DB2692AC	8	4m	Tidak	Tidak	-	0	8
22	DB2997AD	8	2m	Ya	Ya	DB2997AD	8	0
23	DB412FA	7	2m	Ya	Tidak	M412FA	5	2
24	DB4284AV	8	3m	Ya	Ya	DB4284AV	8	0
25	DB4557AU	8	3m	Ya	Ya	DB4557AU	8	0
26	DB4602FB	8	2m	Ya	Ya	DB456Q2FB	7	1
27	DB4733AT	8	2m	Ya	Ya	DB4733AT	8	0
28	DB8076FA	8	2m	Ya	Ya	DB8076FA	8	0
29	DB8701LC	8	2m	Tidak	Tidak	-	0	8
30	DB999RW	7	2m	Ya	Tidak	B999R	5	2
TOTAL		238					191	47

Namun ada beberapa citra yang berjarak ideal namun tidak terbaca keseluruhan disebabkan kecerahan yang tidak terlalu baik untuk proses pengolahan. Masalah ini dapat di atasi dengan mengubah kembali metode pengumpulan data dengan jarak yang ideal. Untuk mencari keakuratan pengujian yang berfokus pada *template matching* saja, penulis memisahkan citra ideal untuk diuji.

$$\begin{aligned}
 \text{Persentase Keberhasilan} &= \frac{191}{238} \times 100\% \quad (3) \\
 &= 0.802521 \times 100\% \\
 &= 80.25\%
 \end{aligned}$$

$$\begin{aligned}
 \text{Persentase Kegagalan} &= \frac{47}{238} \times 100\% \quad (4) \\
 &= 0.197479 \times 100\% \\
 &= 19.75\%
 \end{aligned}$$

Dilakukan percobaan ini untuk mengukur sejauh mana keberhasilan pendeteksian apabila keseluruhan data merupakan data ideal. Hasilnya pada Tabel.2 berikut, keberhasilan mencapai 97,77%, dengan kegagalan 2,23% dengan perhitungan (5) dan (6).

Tabel.2 Hasil Implementasi Data Ideal

No	Plat	Jumlah Karakter	Jarak	Terdeteksi Piksel Terbesar	Terdeteksi Seluruh Karakter	Karakter		
						Terbaca	Benar	Salah
1	DB1056QE	8	2m	Ya	Ya	DB1056QE	8	0
2	DB1071AX	8	2m	Ya	Ya	DB1071AX	8	0
3	DB1074LC	8	2m	Ya	Ya	DB1074LC	8	0
4	DB1179AO	8	2m	Ya	Ya	DB1179AO	8	0
5	DB1208AS	8	3m	Ya	Ya	DB1208AS	8	0
6	DB1277AV	8	2m	Ya	Ya	DB1277AM	7	1
7	DB1308LC	8	2m	Ya	Ya	DB1308LC	8	0
8	DB1324QA	8	3m	Ya	Ya	DB1324QA	8	0
9	DB1402AZ	8	2m	Ya	Ya	DB14Q2AZ	7	1
10	DB1514AL	8	2m	Ya	Ya	DB1514AL	8	0
11	DB1534AS	8	2m	Ya	Ya	DB1534AS	8	0
12	DB1593MB	8	2m	Ya	Ya	DB1593MB	8	0
13	DB1641AF	8	2m	Ya	Ya	DB1641AF	8	0
14	DB1921JA	8	3m	Ya	Ya	DB1921QA	7	1
15	DB1969MC	8	2m	Ya	Ya	DB1969MC	8	0
16	DB1986AL	8	2m	Ya	Ya	DB1986AL	8	0
17	DB2997AD	8	2m	Ya	Ya	DB2997AD	8	0
18	DB4284AV	8	3m	Ya	Ya	DB4284AV	8	0
19	DB4557AU	8	3m	Ya	Ya	DB4557AU	8	0
20	DB4602FB	8	2m	Ya	Ya	DB456Q2FB	7	1
21	DB4733AT	8	2m	Ya	Ya	DB4733AT	8	0
22	DB8076FA	8	2m	Ya	Ya	DB8076FA	8	0
TOTAL		176					172	4

$$\begin{aligned}
 \text{Persentase Keberhasilan} &= \frac{172}{176} \times 100\% \quad (5) \\
 &= 0.977273 \times 100\% \\
 &= 97.77\%
 \end{aligned}$$

$$\begin{aligned}
 \text{Persentase Kegagalan} &= \frac{4}{176} \times 100\% \quad (6) \\
 &= 0.022727 \times 100\% \\
 &= 2.23\%
 \end{aligned}$$

C. Analisa Kesalahan

Terdapat beberapa faktor yang dapat mempengaruhi, diantaranya :

1. Kesalahan Pada Tahap Pendeteksian Karakter

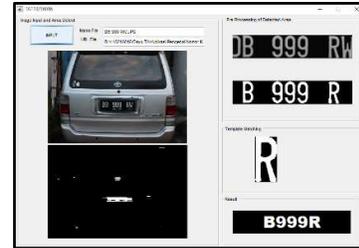
4 dari 7 citra yang tidak ideal didapati kesalahan pada tahap ini. Dari citra tersebut beberapa kesalahan sebagai berikut :

- Jarak pengambilan data

Sesuai dengan pengamatan citra ideal dalam sistem adalah citra yang diambil dengan jarak 2-3m. Untuk mendapatkan area plat nomor dengan tepat dan menyeluruh. Berdasarkan data, citra dengan jarak 4 meter sudah tidak layak digunakan karena lebih rentan gagal.

- Tingkat kecerahan yang tidak sesuai

Pencahayaan sangat penting dalam pengambilan data, sehingga dibutuhkan pengaturan manual pada kamera bertujuan untuk menyesuaikan dengan kondisi cahaya yang ada. Contohnya seperti Gambar.9 ini.



Gambar.9 Uji coba citra DB 999 RW.JPG

Terkadang data citra sampel tidak memenuhi standar itu yang dapat menyebabkan citra plat tidak terdeteksi dengan baik walaupun dengan jarak ideal.

2. Kesalahan Pada Tahap Pre-processing

- Citra karakter tidak terdeteksi keseluruhan

Proses pada aplikasi memiliki tahapan secara berurutan, jadi wajar saja bila kesalahan pada tahapan sebelumnya berimbas pada tahapan setelahnya. Contoh Gambar 9 di atas, dikarenakan kecerahan yang tidak baik menghasilkan posisi citra yang terdeteksi terlalu kecil dan memotong sebagian karakter pada tahap pendeteksian.

Pada tahapan ke 2 (pre-processing) terdapat proses *border remove*, yang fungsinya menghilangkan objek piksel yang menempel pada tepi. Gunanya menghilangkan objek piksel yang tidak dibutuhkan, namun pada contoh ini piksel karakter yang dibutuhkan menempel di tepi sehingga piksel di hapus/di hilangkan.

3. Kesalahan Pada Tahap Pengenalan Karakter

- Bentuk karakter yang tidak sesuai standar

Dalam beberapa data terkadang kondisi plat nomor bengkok atau posisi kendaraan tidak sejajar atau miring seperti Gambar.10 di bawah, sehingga karakter berada pada posisi berbeda.



Gambar.10 Citra karakter v pada citra DB 1277 AV.JPG



Gambar.11 Citra proses menggunakan template v dan m

Dengan begitu pemrosesan dengan font V dan M yang terlihat pada Gambar.11 di atas, terlihat piksel lebih besar pada font M. Sehingga Karakter yang seharusnya terbaca V, dianggap M.

- Bentuk karakter yang mirip

Beberapa karakter mirip antara satu dengan yang lain menyebabkan terjadi kesalahan dalam identifikasi. Contoh yang sering ditemukan adalah karakter 0 (nol) yang sering dibaca huruf Q dapat dilihat pada Gambar.13.



Gambar.13 Citra *template* 0 dan Q

Setelah dilakukan proses *resize*, bentuk lingkaran karakter 0(nol) hampir sama dengan karakter Q, dan jika volume citra karakter lebih tebal daripada *template*, maka akan lebih sering terbaca karakter Q.

V. PENUTUP

A. Kesimpulan

1. Aplikasi pengenalan nomor kendaraan bermotor telah berhasil dibuat dan mengenali karakter menggunakan keseluruhan 30 data sampel dengan total 238 karakter dengan tingkat keberhasilan 80,25 %
2. Data sampel ideal untuk sistem ini adalah data sampel yang memiliki nilai kecerahan yang baik dan dengan jarak pengambilan data 2 hingga 3 meter.
3. Dengan pengujian yang hanya menggunakan citra ideal berjumlah 22 data sampel dengan total 176 karakter, didapatkan hasil tingkat akurasi hingga 97,77%

B. Saran

1. Diharapkan untuk penelitian selanjutnya lebih diperhatikan dalam metode pengambilan data untuk jarak, sudut pengambilan gambar, kondisi mobil yang bergetar, dan tingkat kecerahan citra.
2. Pada program lebih baik lagi bila ada tahap awal sebelum pengolahan utama terhadap citra, agar citra yang tidak ideal dapat terbaca juga.

DAFTAR PUSTAKA

- [1] Ginting Elias D. Jurusan Teknik Informatika Universitas Gunadarma. "Deteksi Tepi Menggunakan Metode Canny dengan Matlab Untuk Membedakan Uang Asli dan Uang Palsu".
- [2] Kusumanto RD. (2011). Jurusan Teknik Komputer Politeknik Negri Sriwijaya Palembang. "Pengolahan Citra Digital Untuk Mendeteksi Obyek Menggunakan Pengolahan Warna Model Normalisasi RGB".
- [3] Leksono B., Hidayatno A., Isnanto R. Rizal (2011). Fakultas Teknik Universitas Diponegoro Semarang."Aplikasi Metode Template Matching untuk Klasifikasi Sidik Jari."

- [4] Manurung Patardo Marasi. (2007). Fakultas Teknik, Universitas Indonesia. "Perangkat Lunak Pengenalan Plat Nomor Mobil Menggunakan Jaringan Kompetitif dan Jaringan Kohonen."
- [5] Mulyawan H. , Samsono M. Z. Hadi dan Setiawardhana (2012) Jurusan Telekomunikasi, Institut Teknologi Sepuluh Nopember (ITS) Surabaya. "Identifikasi Dan Tracking Objek Berbasis Image Processing Secara Real Time".
- [6] Praharasty Anggit (2008). Jurusan Teknik Elektro, Universitas Diponegoro Semarang. "Simulasi Pendeteksian Garis Lurus (Stright Line) Pada Citra Digital Menggunakan Matlab 7.1"
- [7] Wijaya Tria Adhi (2010). Jurusan Teknik Informatika, Fakultas Teknologi Industri Universitas Islam Indonesia Yogyakarta. "Implementasi Visi Komputer Dan Sementasi Citra Untuk Klasifikasi Bobot Telur Ayam Ras".



Sekilas dari penulis dengan nama lengkap Bayu Sy. Kurniawan, lahir di kota Manado, Provinsi Sulawesi Utara. Anak ke-2 dari 2 bersaudara. Dengan pendidikan Sekolah Dasar Negeri 11Manado. Kemudian Melanjutkan ke Sekolah Menengah Pertama Yayasan Kemala Bhayangkari 1 Surabaya. Kemudian melanjutkan ke Sekolah Menengah Atas SMAN 9Manado. Setelah lulus tahun 2011 melanjutkan ke Perguruan Tinggi di Universitas Sam Ratulangi Manado dengan mengambil Jurusan Teknik Informatika. Pada tahun 2015 bulan april, penulis membuat Skripsi demi memenuhi syarat Sarjana (S1) dengan penelitian berjudul Aplikasi Pengenal Citra Nomor Kendaraan Bermotor yang dibimbing oleh dua dosen pembimbing yaitu Dr. Eng Steven R. Sentinuwo, ST., MTI dan Oktavian Abraham Lantang, ST., MTI sehingga pada tanggal 6 April 2016 penulis resmi lulus di Teknik Informatika Universitas Sam Ratulangi Manado dan menyandang gelar Sarjana Komputer dengan predikat Sangat Memuaskan.