

Sistem Pendeteksian Lahan Parkir Kosong pada Static Image dengan Image Processing

Albert Liem, Iwan Njoto Sandjaja, Rudy Adipranata

Program Studi Teknik Informatika,

Fakultas Teknologi Industri, Universitas Kristen Petra

Jln. Siwalankerto 121-131 Surabaya 60236

Telp. (031)-2983455, Fax. (031)-8417658

albertliem7@yahoo.com, iwanns@petra.ac.id, rudya@petra.ac.id

ABSTRAK

Dalam paper ini akan diterangkan tentang pembuatan sistem pendeteksian lahan parkir dengan *Image Processing*. Program ini dibuat dalam Visual Studio dengan bahasa pemrograman C++ dan *library* OpenCV. *Image processing* memiliki cakupan yang luas dan pada pembuatan sistem ini akan digunakan beberapa metode untuk mendeteksi lapangan parkir. Pada program pendeteksian ini, *input* berupa *feed* video dari lapangan parkir, video tersebut diolah dan dianalisa dengan metode klasifikasi *thresholding* dan SVM. Hasil berupa suatu indikator yang memberitahu pengguna apakah *slot* parkir kosong atau tidak.

Kata Kunci: Pengolahan citra digital, substraksi *background*, pendeteksian lahan parkir.

ABSTRACT

In this paper we will explain about vacant parking space detection system through image processing. This program was made with Visual Studio using C++ programming language and OpenCV library. Image processing itself has a vast research area and some of its methods will be used to create a parking space detection system. In this program, the input consists of the video feed of the parking lot, the said video is processed and analyzed with thresholding classification and SVM classification, resulting in an indicator to show if certain parking slot is vacant or not.

Keywords: *Image Processing, Background Substraction, Parking space detection.*

1. PENDAHULUAN

1.1 Latar Belakang Masalah

Salah satu kendala dalam mencari tempat parkir adalah [6] terbuangnya waktu dan usaha yang dilakukan seseorang. Memang ada beberapa gedung yang meberikan sedikit kemudahan para pengunjung dengan menginstalasi sensor parkir dengan lampu, akan tetapi tetap saja para pengunjung harus mencari-cari, akan lebih mudah jika hal ini ditampilkan secara visual agar para pengunjung dapat diarahkan. Hal ini bisa dilakukan dengan bantuan *image processing*. [2]

Dengan banyaknya metode *image processing*, beberapa metode akan digunakan, diantaranya merupakan *corner detection*, dan *background subtraction*. Akan tetapi dalam menentukan dan klasifikasi tempat parkir, akan digunakan SVM dan metode *thresholding*.

1.2 Perumusan Masalah

Bagaimana dengan metode-metode yang akan digunakan dapat memberikan hasil maksimal dalam membedakan tempat parkir yang kosong atau tempat parkir yang sudah terisi mobil.

1.3 Tujuan Skripsi

Tujuan tugas akhir ini adalah untuk membuat suatu sistem pendeteksi lapangan parkir dengan menggunakan metode yang sudah disebutkan.

2. LANDASAN TEORI

2.1 Parking Space Extraction

Ekstraksi [6] warna pada lapangan parkir perlu dilakukan sebelum mengolah input lebih jauh, sehingga membantu meningkatkan jumlah kendaraan yang dapat dilihat. Penempatan ROI juga dilakukan pada setiap slot lapangan parkir, karena kamera tidak berubah-ubah atau tidak ada gerakan pada kamera, setiap ROI dilakukan secara manual, hal ini akan dilakukan sekali saat instalasi program.

2.1.1 Thresholding

Thresholding [1] merupakan teknik dalam *image segmentation* yang dihasilkan dari input berupa *greyscale*, mengubah gambar dalam pixel hitam dan putih atau *binary image*. Hal ini memungkinkan *thresholding* sebagai bentuk klasifikasi dalam membedakan lapangan parkir yang kosong atau tidak.

2.1.2 Region of Interest

Region of Interest atau ROI [3] adalah sebuah subset yang dari sebuah sample data. Pada program ini ROI berfungsi sebagai suatu 'slot parkir' yang dapat dibaca oleh program. Maka setiap slot parkir pada input akan diberi ROI untuk memudahkan proses penggolongan. Image yang ada didalam ROI juga digunakan sebagai training set dalam SVM.

2.2 Corner Detection

Corner Detection seperti namanya mendeteksi [4] intensitas dari perbedaan posisi kedua sudut, sehingga program dapat menemukan *corner* pada gambar inputan.



Gambar 1. *Corner detection* pada suatu image.

Pada gambar 1, *corner* pada image ditandai dengan 5 lingkaran hitam pada mobil. Fungsi ini membantu menemukan fitur mobil yang memiliki *corner* lebih dari 1. Hal ini juga menjadi salah satu alat klasifikasi obyek yang berada dalam slot parkir.

2.3 Support Vector Machine

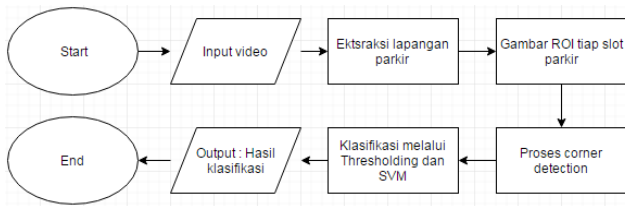
Support vector machine menjadi alternatif dalam mengklasifikasi lapangan parkir. SVM sendiri adalah [7] algoritma klasifikasi yang menghitung margin hyperplane, memisahkan training data dengan class masing-masing, lalu menguji data hasil dengan data inputan. SVM yang digunakan pada program ini merupakan binary SVM untuk mengklasifikasi 2 kemungkinan, slot parkir kosong atau sudah terisi.

3. DESAIN SISTEM

3.1 Desain Sistem Keseluruhan

Program akan melakukan extraction pada lapangan parkir dengan mengeleminasi setiap warna pada input image atau video dengan memberikan ROI berukuran kecil pada warna lapangan parkir (aspal parkir), dari ROI yang diberikan setiap warna didalam ROI tersebut akan digunakan sebagai warna threshold pada input image atau video.

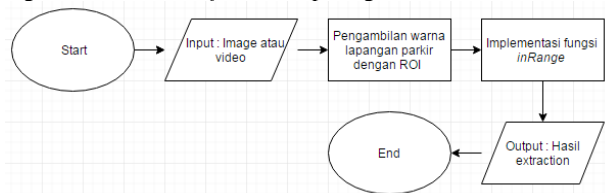
Gambar 2 menjelaskan untuk setiap slot parkir yang berada dalam input image atau video, perlu digambarkan ROI. Hal ini memungkinkan program untuk mengklasifikasi setiap slot parkir. Setelah pemberian ROI pada input, dilakukan implementasi *corner detection* hal ini bertujuan untuk menemukan fitur mobil. Setelah proses ini, ada 2 proses klasifikasi, pendeteksian mobil dengan *thresholding* dan pendeteksian mobil melalui SVM. Pendeteksian mobil dengan *thresholding* menggunakan fungsi penghitungan pixel hitam pada ROI dalam input image atau video yang sudah di *thresholding*, sementara pendeteksian melalui SVM dicari dengan hasil training dari image ROI *corner detection*.



Gambar 2. Flowchart Sistem Keseluruhan

3.2 Desain Sistem Background Substraction

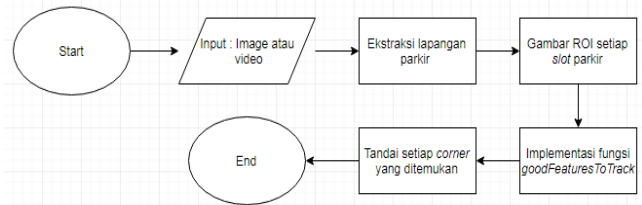
Pada bagian ini extraction menggunakan fungsi [2] *inRange* yang merupakan salah satu fitur library OpenCv. Fungsi *inRange* merupakan salah satu metode *thresholding* dimana inputan merupakan suatu kisaran scalar yang didapat dari ROI. Hal ini digambarkan melalui *flowchart* pada gambar 3.



Gambar 3. Flowchart sistem Background Substraction

3.3 Desain Proses Corner Detection

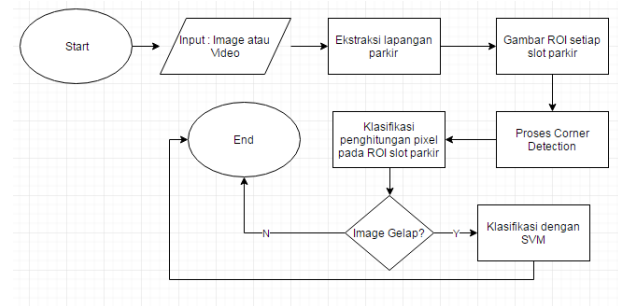
Gambar 4 berisi tentang proses *corner detection* yang membantu program dalam memeriksa fitur dari mobil pada ROI. Hal ini dilakukan agar program dapat membedakan apakah benda dalam ROI adalah mobil atau tidak.



Gambar 4. Flowchart proses Corner Detection

3.4 Desain Klasifikasi Parkir Kosong

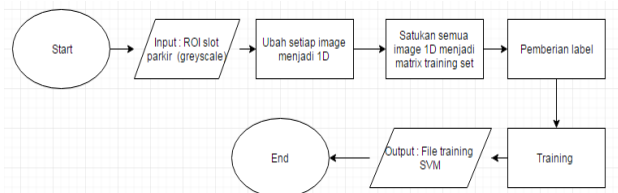
Pada gambar 5 terdapat 2 klasifikasi pada slot parkir, yang pertama merupakan penghitungan pixel pada ROI yang sudah di *threshold* dengan parking space extraction dan yang kedua adalah pengecekan melalui SVM setelah ROI diberikan *corner detection*.



Gambar 5. Flowchart Proses Klasifikasi

3.5 Flowchart SVM Training

Pada gambar 6 *flowchart SVM training* dilakukan dengan bantuan *image ROI* yang hasilnya adalah file *training SVM* yang digunakan sebagai klasifikasi.



Gambar 6. Flowchart Proses SVM Training

4. PENGUJIAN SISTEM

Bab ini berisikan hasil pengujian sistem pendeteksian lapangan parkir.

4.1 Tampilan

Tampilan pada Gambar 7 merupakan output jadi dimana klasifikasi dilakukan dengan penghitungan pixel pada ROI, pada ROI berwarna merah terdapat angka yang didapatkan dalam klasifikasi perbandingan pixel berdasarkan *thresholding*. Angka ini menjadi acuan dalam program untuk mendeteksi lapangan parkir yang kosong atau terisi. Serta angka juga menunjukkan

pengguna output hasil klasifikasi. Tampilan threshold dan corner detection berada pada 2 windows lain.



Gambar 7. Tampilan Program

4.2 Uji Coba

Uji coba menggunakan video yang diambil dari lapangan parkir gedung T dari Universitas Kristen Petra. Video sample memiliki 3 timelapse berbeda siang, sore dan malam. Klasifikasi dengan fungsi penghitungan pixel diuji coba dengan 9 slot parkir dengan pencahayaan berbeda-beda. Setiap ROI memiliki resolusi yang sama yaitu 40x54 pixels.

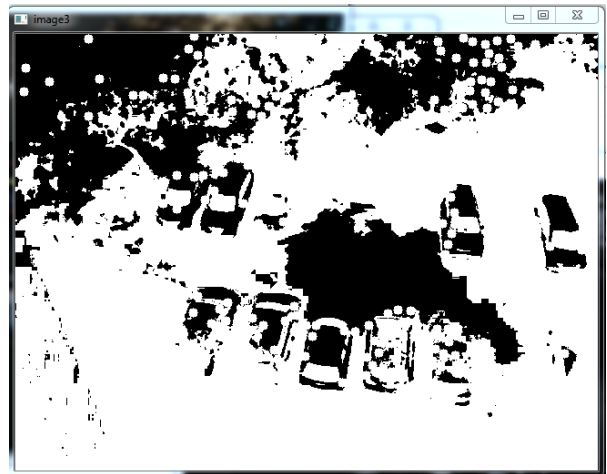
Pada pencahayaan malam hari di gambar 10, klasifikasi pixel menjadi sulit untuk diterapkan karena gambar input menjadi terlalu gelap, maka perlu dilakukan klasifikasi dengan SVM. Klasifikasi SVM diuji coba dengan 30 gambar ROI dengan pencahayaan yang kurang dan memiliki keakuratan 86%.

Uji coba yang dilakukan meliputi uji coba klasifikasi *pixel* dan uji coba klasifikasi SVM

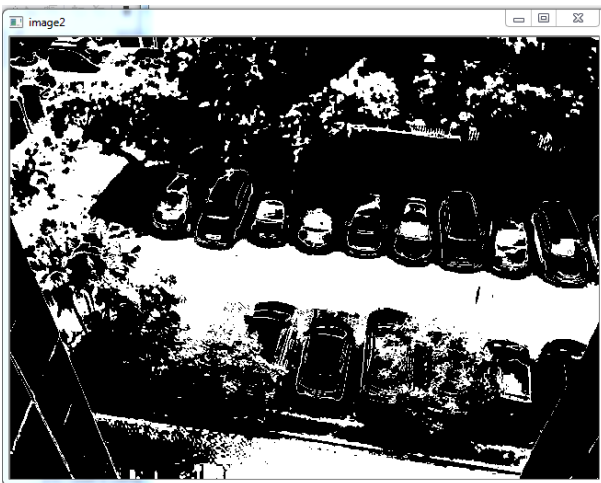
Pada gambar 8 dan 9, pencahayaan siang dan sore klasifikasi dengan penghitungan pixel memiliki keakuratan 88% , dimana program juga bisa salah membaca jika adanya warna mobil yang terlalu ambigu dengan warna lapangan parkir. Tapi hal ini jarang ditemukan.



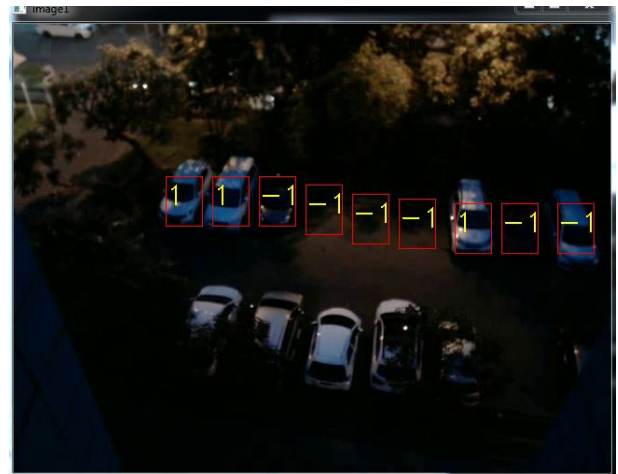
Gambar 9. Uji Coba Pencahayaan (Sore)



Gambar 10. Uji Coba Pencahayaan (Malam)



Gambar 8. Uji coba pencahayaan (pagi/siang)



Gambar 11. Uji Coba Klasifikasi SVM

5. KESIMPULAN

Jurnal ilmiah ini mempresentasikan sistem pendeteksian lapangan parkir. Menggunakan beberapa algoritma untuk menentukan klasifikasi yang sudah baik dalam mendeteksi juga membedakan lapangan parkir terisi dan kosong. Kelemahan pada sistem ini berada pada input yang memberikan gambar terlalu gelap sehingga sistem tidak dapat bekerja dengan baik. Untuk pengembangan kedepannya diharapkan sistem dapat diimplementasikan pada perangkat android agar mempermudah pencarian slot parkir bagi lebih banyak orang.

- Program dapat ekstraksi lapangan parkir yang menjadi salah satu fondasi dasar agar sistem bisa berjalan.
- Program dapat mengimplementasi algoritma *corner detection* dengan baik.
- Program dapat bekerja dengan baik dalam input dengan pencahayaan yang terang
- Program tidak bekerja dengan baik jika input tidak jelas atau gelap.
- Program dapat memberikan hasil klasifikasi yang akurat.
- Program masih memerlukan banyak perkembangan agar menjadi sistem yang benar-benar matang.

6. DAFTAR PUSTAKA

- [1] Baggio, D.L. 2012. *Mastering OpenCV with practical computer vision projects*. Birmingham: Packt Publishing Ltd.
- [2] Dalal, J. dan Patel, S. 2013. *Instant OpenCV Starter*. Birmingham: Packt Publishing Ltd.
- [3] Gaikwad, A., Shah, B., dan Shelke, C. 2016. *Parking Lot Classification*. URI = <http://vision.soic.indiana.edu/b657/sp2016/projects/bbshah/paper.pdf>
- [4] Kaehler, A. dan Bradski, G. 2016. *Learning OpenCV 3: Computer Vision in C++ with the OpenCV Library*. Canada: O'Reilly Media, Inc.
- [5] Patel, T.P. 2014. *Corner Detection Techniques: An Introductory Survey*. URI = <https://www.ijedr.org/papers/IJEDR1404047.pdf>
- [6] True, N. 2007. *Vacant Parking Space Detection in Static Image*. URI = <http://cseweb.ucsd.edu/classes/wi07/cse190/a/reports/ntrue.pdf>
- [7] Wu, Q. dan Zhang, Y. 2006. *Parking Lots Space Detection*. Technical Reports., Carnegie Mellon University, Pittsburgh, Pennsylvania, USA.