**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI**

**INSTITUT TEKNOLOGI SEPULUH NOPEMBER**

**USULAN TUGAS AKHIR**

# **IDENTITAS PENGUSUL**

Nama : **Mujiono**

NRP : **5108 100 090**

Dosen Wali : **Umi Laili Yuhana, S.Kom, M.Sc.**

1. **JUDUL TUGAS AKHIR *“Implementasi Sistem Visi Komputer untuk Pengenalan Kode Peti Kemas Secara Otomatis”***
2. **LATAR BELAKANG**

Dalam beberapa tahun terakhir, transportasi kapal perindustrian menjadi semakin berkembang di dunia internasional. Hal ini berbanding lurus dengan jumlah peti yang diangkut juga mengalami peningkatan dengan tren ini. Peti Kemas atau yang dalam bahasa Inggris disebut sebagai *Container* adalah salah satu alat untuk mengemas (kargo) yang dapat diangkut dengan mudah menggunakan truk, kapal, atau pesawat terbang, terutama untuk lintas pulau atau lintas negara.

Setiap peti kemas memiliki kode yang berisi informasi. Untuk mengetahui informasi yang dimiliki oleh peti kemas tertentu harus dilakukan pengenalan terhadap kode tersebut. Pengenalan kode peti kemas ini merupakan salah satu langkah penting dalam manajemen aliran peti kemas. Dewasa ini, pengenalan semua kode dilakukan oleh manusia, yang dapat saja terjadi kesalahan dalam membaca kode peti kemas, disamping itu akan menghabiskan banyak waktu dalam penerapannya.

Masalah ini menimbulkan ketertarikan untuk melakukan sebuah riset yang menghasilkan sistem otomatis berbasis visi komputer yang memungkinkan kemudahan untuk mengenali kode peti kemas secara otomatis. Dengan citra peti kemas yang diproses menggunakan metode – metode yang digunakan untuk pengenalan karakter pada visi komputer, akan dilakukan pengekstrakan kode peti kemas yang nantinya akan dibaca dan dikenali. Dengan sistem ini, *human-error* atau kesalahan yang bisa dilakukan oleh manusia dalam mengenali kode peti kemas dan waktu yang banyak dibutuhkan untuk melakukannya dapat dihindari.

1. **RUMUSAN MASALAH**

Rumusan masalah yang diangkat dalam Tugas Akhir ini adalah sebagai berikut:

1. Bagaimana melakukan Konstruksi Model (*Model Construction*) Citra Peti Kemas, yang meliputi
   1. Proses Segmentasi Citra menggunakan *Locally Thresholding Segmentation*.
   2. Ekstraksi Karakter Berbasis Fitur-fitur Geometri (*Character Extraction Based on Geometry Features*)
   3. *Eigen-feature Models*
2. Bagaimana melakukan pengenalan kode (*Code Recognition*) Peti Kemas
3. **BATASAN MASALAH**

Permasalahan yang dibahas dalam Tugas Akhir ini memiliki beberapa batasan, yaitu sebagai berikut :

1. Sistem perangkat lunak diagnosa melanoma ini bangun dengan menggunakan perangkat lunak MATLAB R2008a
2. Data citra harus memiliki resolusi yang bagus dan tidak terlalu kecil. Sebisa mungkin menggunakan citra dengan format 24 bits BMP atau JPG dengan tipe RGB. Ukuran resolusi citra 480 x 640 x 3 pixel, ukuran memori sekitar 67Kbytes.
3. **TUJUAN TUGAS AKHIR**

Tugas akhir ini bertujuan mendesain dan merancang sistem perangkat lunak berbasis visi komputer yang dapat melakukan pengenalan terhadap kode kontainer secara otomatis.

1. **MANFAAT TUGAS AKHIR**

Tugas akhir ini dikerjakan dengan harapan dapat memberikan manfaat yang besar pada bidang Industri dan Perdagangan yang menggunakan peti kemas dalam usahanya. Pengenalan kode peti kemas ini dapat mempermudah dan mempercepat pengkategorian peti kemas itu sendiri.

1. **RINGKASAN TUGAS AKHIR**

Metode manual untuk mengenali kode peti kemas memiliki banyak kekurangan, diantaranya lambat, tingkat kesalahan tinggi, dan sebagainya. Sehingga dibutuhkam sistem otomatis yang dapat digunakan untuk mengetahui informasi dari suatu peti kemas. Karena dapat meningkatkan efisiensi yang bernilai ekonomis di terminal peti kemas.

Pada tugas akhir ini penulis akan mendesain dan merancang sistem berbasis visi komputer yang dapat membaca dan mengenali kode peti kemas. Sistem dikembangkan kedalam dua bagian, yaitu konstruksi model dan pengenalan kode peti kemas.

* Tahap Konstruksi Model (*Model Construction*)

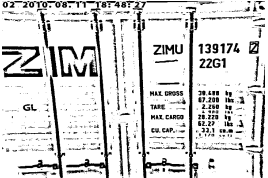
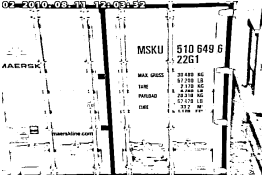
Tahap ini memiliki tiga metode yang dilakukan, diantaranya :

1. *Locally Thresholding Segmentation*

Segmentasi karakter – karakter kode dilakukan dengan memperhatikan tiga property intensitas pada citra peti kemas. Pertama, karakter – karakter dengan intensitas yang mirip satu sama lain. Kedua, besarnya intensitas kontras yang mengelilingi karakter – karakter. Ketiga, kemungkinan pencahayaan oleh sinar matahari yang tidak merata pada citra peti kemas. Berdasarkan tiga observasi ini, metode *locally thresholding segmentation* [4] dipilih untuk mensegmentasi kode peti kemas.

Metode *locally thresholding* ini dirancang berdasarkan distribusi intensitas ketetanggaan di setiap piksel pada citra. Yang pertama dilakukan adalah mentransformasi citra input RGB menjadi citra grayscale. Filter yang telah ditentukan ukurannya dijalankan pada citra, setiap piksel yang dilalui oleh pusat filter dihitung intensitas rata – rata dari piksel yang bertetangga yang masuk pada filter. Piksel tersebut diklasifikasi sebagai foreground jika intensitasnya lebih kecil dari *c* persen dari rata – rata intensitas, dan diklasifikasi sebagai background jika sebaliknya.





|  |  |
| --- | --- |
| a | b |
| c | d |

Gambar 1. *Locally Thresholding Segmentation*. (a) dan (b) merupakan citra input peti kemas. (c) dan (d) merupakan hasil segmentasi dari citra input (a) dan (b).

Piksel yang termasuk foreground disebut sebagai **FG**. Panjang filter yang dijalankan pada citra ditentukan sebasar 1/30 panjang citra input, dan *c* sebagai konstanta ditentukan sebesar 80. Kriteria thresholding ini hanya cocok untuk citra peti kemas, dimana daerah kode peti kemas lebih gelap dari daerah sekitarnya.

Kesalahan segmentasi dapat terjadi karena kesalahan klasifikasi ketika kode peti kemas dibaca sebagai background. Oleh sebab itu, perlu dibuat sebuah strategi pengecekan unutk mencegah hal ini. Pertama kali yang dilakuan yaitu melakukan segmentasi dengan metode Otsu [1] pada citra grayscale peti kemas dan membaginya kedalam dua cluster piksel **C1** dan **C2**. Karena peti kemas melingkupi sebagian besar citra, maka dimasukkan kedalam cluster yang lebih besar (diasumsikan **C1**). Disamping itu, karakter – karakter kode dimasukkan kedalam cluster **C2** karena memiliki perbedaan intensitas kontras dengan sekitarnya. Secara berurutan, dihitung intensitas rata – rata dari citra grayscale untuk **C1** dan **C2**. Error terdeteksi pada segmentasi local jika intensitas rata – rata **FG** dan **C1** lebih kecil dari **FG** atau **C2**. Sehingga mengubah criteria dan memproses citra lagi. Kriteria dirubah menjadi piksel dengan rata – rata intensitasnya lebih besar dari tetangganya diklasifikasikan sebagai foreground, dan sebaliknya diklasifikasikan menjadi background. Dengan proses pengecekan ini, akan didapatkan hasil segmentasi karakter – karakter kode peti kemas baik hitam maupun putih.

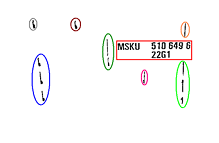
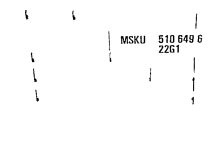
1. *Character Extraction Based on Geometry Features*

Metode thresholding bisa saja menghasilkan sejumlah daerah yang bukan merupakan karakter – karakter kode peti kemas. Karakter – karakter ini pastinya memiliki properti geometri, dengan ukuran pada range tertentu yang bisa diperkirakan untuk setiap properti geometri, termasuk luas, keliling, panjang, dan lebar. Jika nilai dari parameter suatu daerah diluar rentang yang diperkirakan secara empiris, maka bukan termasuk daerah kode, sebaliknya termasuk daerah kode. Untuk mengambil daerah karakter kode, maka dirancang algoritma *distance-based grouping*. Berikut step – step algoritma :

* + - Step 1. Beri label FALSE semua region dan set status groupingnya UNGROUP.
    - Step 2. Set TRUE suatu region yang berlabel FALSE.
    - Step 3. Jika ada region berlabel FALSE memenuhi dua aturan sebagai berikut:
      * (1) Jarak antara pusat dan pusat dari setiap region TRUE & UNGROUP lebih kecil dari seratus piksel,
      * (2) Perbedaan x atau y sumbu koordinat antara pusat dan pusat berbagai daerah TRUE & UNGROUPRED lebih kecil dari delapan piksel,

maka daerah diperiksa diberi label sebagai TRUE.

* + - Step 4. Ulangi step 3 sampai tidak ada lagi daerah TRUE ditemukan.
    - Step 5. Kelompokkan region TRUE & UNGROUP, dan set status groupingnya GROUP.
    - Step 6. Ulangi step 2 sampai tidak ada region yang dapat dikelompokkan lagi.





|  |  |
| --- | --- |
| a | b |
| c |  |

Gambar 2. *Character Extraction* dari Gambar 1(c). (a) hasil filtering menggunakan *geometry feature* . (b) hasil *grouping* dengan *character distance* (c) hasil *character extraction*.

1. *Eigen-feature Models*

Menggunakan metode segmentasi yang diusulkan, daerah karakter secara otomatis dapat ekstrak dari citra training. Untuk setiap karakter alphabet dan numeric dikonstruk Eigen-feature model berdasarkan PCS untuk mengkarakterisasi variasi kemunculannya pada citra training yang berbeda. Pada proses kontruksi model,citra biner hasil segmentasi dipisah satu sama lain untuk menghilangkan perbedaan pose. Hasil pemisahan antar karakter berfungsi sebagai sampel training, dinotasikan sebagai X = (x1, x2, ... , xi, xm-1, xm) dimana m merupakan jumlah sampel training. Setiap sampel training xi direpresentasikan sebagai vector satu dimensi (x1, x2, ..., xj, ..., xn-1, xn)T, dimana xj merupakan nilai intensitas piksel ke-j pada sampel training dan n meruapakan jumlah piksel. Kemudian dirata – rata m sampel untuk memperoleh nilai mean vector x, selanjutnya menghitung matrik kovarian C yang mengimplikasikan variansi setiap sampel training dengan mean vector. Dengan C diketahui dari persamaan :

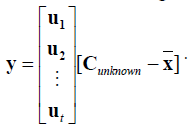
Sehingga akan ditemukan eigenvalues (v1, v2, ... , vi, ... , vn-1, vn ) dan eigenvector (u1, u2, ... , ui, ... , un-1, un) dari matrik kovarian. Mengingat eigenvalue diatur secara descending, dimensi dari eigen feature direduksi dengan menjaga t komponen yang pertama, dengan kondisi :

Sehingga mengahasilkan eigen-feature, yang ditandai dengan mean vector dan eigenvector. Dengan menerapkan proses kontruksi untuk semua karakter termasuk 0-9 dan A-Z, akan diperoleh sebuah set model eigen-feature yang dapat digunakan untuk tahap selanjutnya.

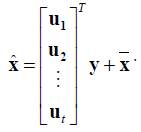
* Tahap Pengenalan Kode (*Code Recognition*)

Pada tahap pengenalan kode, pertama memisahkan kode karakter dari citra yang diberikan. Setiap karakter yang tidak diketahui dibandingkan dengan semua model eigen-feature, dan dikenali dengan menemukan model terbaik yang cocok dengan error rekonstruksi PCA terkecil. Detail dari algoritma pengenalan kode karakter sebagai berikut :

* Step – step
  + Step 1. Input unknown karakter Cunknown  dan semua model Eigen-feature.
  + Step 2. Pilih sebuah Eigen-feature dan proyeksikan vektor karakter Cunknown



* + Step 3. Reconstruct karakter setelah melalui proses proyeksi



* + Step 4. Hitung reconstruction error
  + Step 5. Simpan nilai error dan kembali ke Step 2 sampai semua model Eigen-feature dibandingkan.
  + Step 6. Pilih karakter dengan reconstruction error minimal sebagai hasil pengenalan Cunknown

Sebuah contoh dari pengenalan karakter ditunjukkan dalam Gambar. 3. Karakter yang tidak diketahui ditunjukkan pada Gambar. 3 (a), dan penampilan direkonstruksi menggunakan Eigen-fitur model '2 ', '3', dan '7 'ditampilkan dalam Gambar. 3 (b) - (d), masing-masing. Hal ini diamati bahwa menggunakan model Eigen-fitur yang sesuai untuk merekonstruksi penampilan karakter dapat menyebabkan distorsi terkecil, karakter dengan distorsi terkecil inilah yang nantinya digunakan sebagai karakter yang diakui. Setelah semua karakter yang diakui, kita dapat memperoleh hasil pengakuan dari kode seluruh peti kemas.



|  |  |  |  |
| --- | --- | --- | --- |
| a | b | c | d |

Gambar 3. Contoh pengenalan karakter. (a) karakter yang tidak diketahui. (b)-(d) hasil rekontruksi penampilan karakter menggunakan tiga model *eigen-feature* yang berbeda.

Di sisi lain, untuk meningkatkan tingkat pengenalan dari sistem yang diusulkan, penulis menggabungkan informasi lebih lanjut dari aturan pengkodean peti kemas ke dalam proses pengenalan karakter didasarkan pada standar ISO 6346 [5]. Aturan pengenalan menyediakan informasi untuk karakter yang seharusnya disesuaikan dengan karakter numerik maupun abjad, seperti yang ditunjukkan pada Tabel I. Misalnya, tiga karakter pertama dari kode peti mewakili id pemilik dalam karakter abjad. Oleh karena itu, hanya karakter abjad yang diambil sebagai solusi yang layak untuk mengenalinya. Sehingga, kesalahan dalam mengenali antara karakter numerik dan karakter abjad dengan tampilan yang mirip, misalnya, '1 'dan' aku ', dapat dengan mudah dihindari.

|  |  |  |
| --- | --- | --- |
| Tabel 1 | | |
| Aturan Pengenalan Kode Peti Kemas untuk Algoritma yang diusulkan | | |
|  |  |  |
| Baris Pertama |  |  |
| Kode Karakter | Arti | Tipe Karakter |
| Tiga karakter pertama | Id Pemilik | Karakter Abjad |
| Karakter keempat | Kategori | Karakter Abjad |
| Karakter terakhir | Cek Digit | Karakter Numerik |
| Karakter yang lain | Nomer Seri | Karakter Numerik |
|  |  |  |
| Baris Kedua |  |  |
| Kode Karakter | Arti | Tipe Karakter |
| Dua karakter pertama | Kode Ukuran | Karakter Numerik |
| Karakter ketiga | Kode Tipe | Karakter Abjad |
| Karakter terakhir | Kode Tipe | Karakter Numerik |

1. **METODOLOGI**

Metodologi yang akan dilakukan dalam Tugas Akhir ini memiliki beberapa tahapan, diantaranya sebagai berikut:

1. Penyusunan Proposal Tugas Akhir

Tahap awal untuk memulai pengerjaan Tugas Akhir adalah penyusunan Proposal Tugas Akhir. Pada proposal ini, penulis mengajukan gagasan pembuatan system otomatis untuk membaca dan mengenali kode peti kemas.

1. Studi Literatur

Pada tahap ini dilakukan pencarian, pengumpulan, penyaringan, pembelajaran dan pemahaman literatur yang berhubungan dengan proses pengolahan citra dokumen terdegradasi, khususnya yang meliputi permasalahan mengenai *preprocessing*, *locally* *segmentation*, *character extraction, eigen-feature model, principal component analysis,* dan *code recognition* . Literatur yang digunakan dalam pengerjaan Tugas Akhir ini sebagian besar berasal dari internet berupa makalah ilmiah, tesis, artikel, materi kuliah, serta beberapa buku referensi.

1. Implementasi

Implementasi merupakan tahap untuk membangun system tersebut.

1. Pengujian dan Evaluasi

Pada tahap ini dilakukan uji coba terhadap sistem yang telah dibuat, mengamati kinerja sistem yang baru dibuat, serta mengidentifikasi kendala yang mungkin timbul.

1. Penyusunan Buku Tugas Akhir

Tahap terakhir merupakan penyusunan laporan yang memuat dokumentasi mengenai pembuatan serta hasil dari implementasi perangkat lunak yang telah dibuat.

1. **JADWAL KEGIATAN TUGAS AKHIR**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No. | Kegiatan | Bulan | | | | | | | | | |
| September | | Oktober | | November | | Desember | | Januari | |
| 1. | Penyusunan Proposal Tugas Akhir |  |  |  |  |  |  |  |  |  |  |
| 2. | Studi Literatur |  |  |  |  |  |  |  |  |  |  |
| 3. | Implementasi |  |  |  |  |  |  |  |  |  |  |
| 4. | Pengujian dan Evaluasi |  |  |  |  |  |  |  |  |  |  |
| 5. | Penyusunan Buku Tugas Akhir |  |  |  |  |  |  |  |  |  |  |

1. **DAFTAR PUSTAKA**
2. Otsu, N., “A threshold selection method from gray-level histograms,” IEEE Trans. Systems, Man and Cybernetics, vol. 9, no. 1, pp. 62–66, Jan. 1979.
3. Pradhan, S.S., and Nanda, P.K., “Adaptive thresholding based image segmentation with uneven 11 lighting condition,” IEEE Region 10 Colloquium and the third ICIIS 2008, pp. 1–6.
4. Peng, J.Y., and Hsu, C.N., “Adaptive local thresholding for fluorescence cell micrographs,” Technical Rep. No. TR-IIS-09-008, Nov. 11, 2009.
5. Bradley, D., and Roth, G., “Adaptive thresholding using the integral image” *Journal of Graphics, GPU, & Game Tools*, vol. 12, no. 2, pp. 13–21, 2007.
6. *Container Handbook: Cargo loss prevention information from German marine insurers*, ch. 3.4. [Online]. Available:

http://www.containerhandbuch.de/chb\_e/stra/index.html?/chb\_e/stra/stra\_03\_04\_00.html

**LEMBAR PENGESAHAN**

###### **Surabaya, September 2010**

Menyetujui,

Dosen Pembimbing I

# **(Dr. Nanik Suciati, S.Kom, M.Kom)**

# **( NIP. ----------- 200112 2 002)**

Dosen Pembimbing II

# **(Arya Yudhi Wijaya, S.Kom, M.Kom )**

**(NIP. --------- 200112 2 002)**