



Pemeriksa Lembar Jawaban dengan Image Recognition

Checking the Answer Sheet with Image Recognition

Dicky Adityanta Sinuraya¹⁾*, Muhammad Siddik Hasibuan¹⁾

¹⁾Politeknik LP3I Medan, Indonesia

*Corresponding Email: dicky.sinuraya112@gmail.com

Abstrak

Pemeriksaan lembar jawaban siswa rutin dilakukan oleh para guru setiap akhir semester. Pemeriksaan lembar jawaban komputer yang dilakukan oleh para guru pada saat ini masih secara manual. Dalam kuesioner penulis, disimpulkan bahwa para guru menyadari bahwa pengisian lembar jawaban komputer secara manual memakan waktu dan tenaga yang lebih agar mereka dapat menyelesaikan pemeriksaan tersebut dengan tepat waktu. Oleh karena itu dibutuhkan sebuah aplikasi yang mempermudah para guru untuk memeriksa lembar jawaban tersebut. Adapun pembuatan aplikasi ini dibuat dengan bahasa pemrograman C# dan EmguCV sebagai library. Aplikasi ini memanfaatkan teknologi Image Recognition dalam pemeriksaan lembar jawaban. Adanya aplikasi penghitung nilai lembar jawaban komputer dengan Image Recognition ini, dapat disimpulkan bahwa aplikasi tersebut dapat membantu para pengguna dalam pemeriksaan lembar jawaban secara efisien baik dari segi waktu maupun tenaga.

Kata Kunci: Image Recognition; Lembar Jawaban Komputer; Emgu CV.

Abstract

Checks of student answers sheets routine done by the teachers at the end of each semester. Examination of computer answer sheets done by teachers at this time still manually. In the authors' questionnaire, it was concluded that teachers realized that completing computer answer sheets manually took more time and effort so that they could complete the examination in a timely manner. Therefore an application is needed that makes it easier for teachers to check the answer sheets. The creation of this application is made with C# programming language and EmguCV as a library. This application utilizes Image Recognition technology in checking answer sheets. The application of computer value sheet calculation counter with Image Recognition, it can be concluded that the application can help users in checking the answer sheets efficiently both in terms of time and energy.

Keywords : Image Recognition, Computer Answer Sheet, Emgu CV.

How to Cite: Sinuraya, D.A. Hasibuan, M.A. (2018). Pemeriksa Lembar Jawaban dengan Image Recognition. *JITE (Journal Of Informatics And Telecommunication Engineering)*. Vol (2). No (1): 1-8

PENDAHULUAN

Pengetahuan tentang *Image Processing* banyak dimanfaatkan untuk bidang keamanan, pertanian, dan industri. *Image Processing* dapat meningkatkan kualitas citra, mengidentifikasi objek, penggabungan dengan bagian citra yang lain. Kemampuan dari teknologi tersebut memungkinkan dapat digunakan lebih efektif dan efisien untuk mengidentifikasi jumlah lingkaran yang benar pada pengisian lembar jawaban komputer.

Lembar jawaban computer (LJK) sudah menjadi lembaran yang dipakai untuk metode ujian pilihan berganda baik itu untuk ujian sekolah hingga ujian nasional. Lembar jawaban tersebut akan diisi oleh siswa yang kemudian akan diperiksa oleh guru. Pemeriksaan hasil ujian yang dilakukan oleh seorang guru secara manual membuat pemeriksaan tersebut kurang efektif. Selain karena waktu yang dibutuhkan jauh lebih lama, hal ini juga dapat membuat pemeriksaan hasil tidak akurat karena faktor kurangnya konsentrasi dan kelelahan saat memeriksa. Oleh karena itu pemeriksaan lembar jawaban komputer dengan image processing dapat menjadi alat alternatif untuk membantu para guru dalam memeriksa lembar jawaban siswa.

Sebelumnya telah ada alat berupa *scanner* yang dapat memeriksa lembar

ujian nasional (UN) dengan *metode scanning*. Pada tugas akhir ini dirancang sebuah software *image processing* yang dihubungkan dengan alat berupa webcam. Setelah itu lembar jawaban yang telah diletakkan tepat dibawah webcam akan diproses dan dihitung jumlah lingkaran yang benar.

METODE PENELITIAN

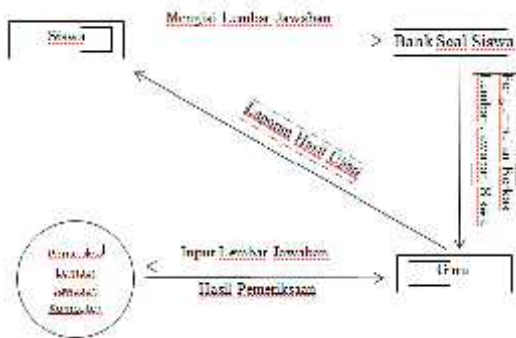
Sistem Sedang Berjalan

Proses pemeriksaan masih manual dikarenakan belum memiliki aplikasi yang dapat mempermudah proses pengolahan data. Proses awal dalam pemeriksaan secara manual ialah data lembar jawaban dikumpulkan lalu dibuatlah kunci jawaban yang benar. Data kemudian diperiksa satu per satu nomor di tiap lembarnya lalu guru akan mencocokkannya dengan kunci jawaban yang telah ditetapkan. Hasil dari pencocokan tersebut lalu dihitung nilai kebenarannya. Pelaporan hasil jawaban tersebut ditulis di atas lembar jawaban. Terakhir semua data hasil dikumpulkan untuk dimasukkan ke entry data siswa.

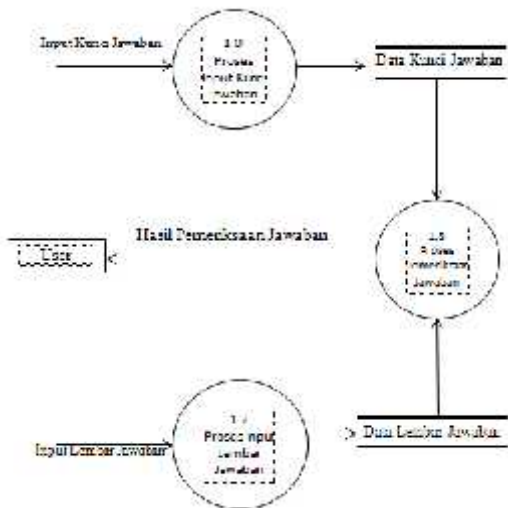
Perancangan Aplikasi

Penulis melakukan perancangan awal sistem aplikasi dengan menggunakan DFD (*Data Flow Diagram*) sesuai dengan data penelitian. Alur DFD ini dapat dipahami tentang sistem yang dirancang tersebut.

Berikut ini adalah gambaran Pemeriksa Lembar Jawaban Komputer dengan Image Recognition.



Gambar 1. Diagram Konteks



Gambar 2. Diagram Level Detail

Dari diagram diatas diperoleh 3(tiga) proses dalam pemeriksaan lembar jawaban tersebut yaitu :

Capture Lembar Jawaban Siswa

Aplikasi ini menangkap gambar dari webcam komputer maupun webcam eksternal. Hasil dari tangkapan webcam kemudian akan diroses.

Capture Kunci Jawaban

Gambar kunci jawaban ditangkap dengan cara yang sama dengan lembar jawaban. Setelah kunci jawaban ditangkap,

kemudian akan di tentukan posisi atau koordinat lingkaran hitam sebagai kunci jawabannya.

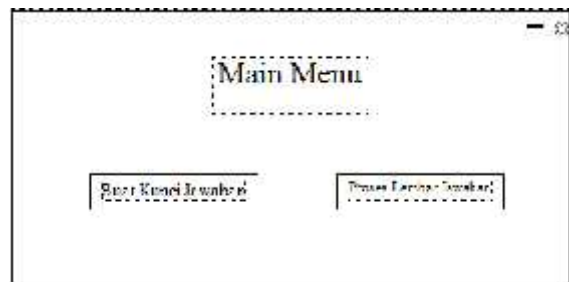
Hasil

Lembar jawaban akan dicocokkan jumlah benarnya dengan kunci jawaban. Setelah mencocokkan maka akan ditampilkan pada form lembar jawaban berapa jumlah jawaban benar yang diperoleh.

Perancangan Tampilan

Halaman Utama

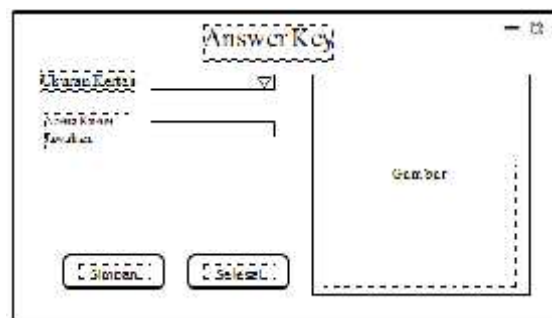
Berikut adalah tampilan halaman utama.



Gambar 3. Rancangan Halaman Utama

Halaman Input Kunci Jawaban

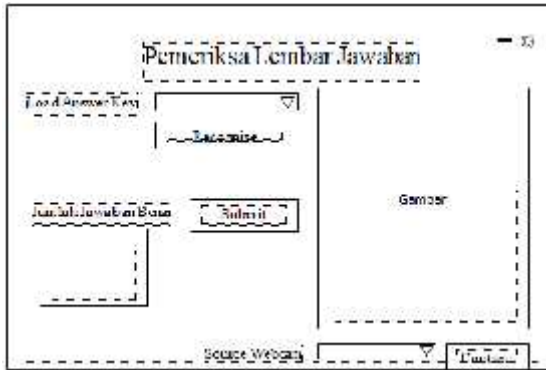
Berikut adalah halaman untuk menginput kunci jawaban



Gambar 4. Rancangan Halaman Input Kunci Jawaban

Halaman Lembar Jawaban

Berikut adalah halaman untuk menginput lembar jawaban dan memprosesnya untuk mendapatkan jumlah nilai yang benar.



Gambar 5. Rancangan Halaman Lembar Jawaban

Algoritma dan Penerapan

Algoritma dari aplikasi ini terdiri dari 2 langkah yaitu:

- a). Memasukkan template dari image lembar jawaban tersebut, lalu menghitungnya dengan transformation matrix.
- b). Menentukan letak pixel dari tiap lingkaran dan menganalisis lingkaran yang diisi dan yang tidak diisi.

Pengimplementasian dari aplikasi ini ialah sebagai berikut :

Langkah 1

Pada langkah ini, akan dilakukan pendeteksian titik kunci (key point) dari lembar jawaban. Pendeteksian key point ini dilakukan dengan pendeteksi SURF, yaitu teknik pendeteksian yang

membandingkan nilai kesamaan dari lembar jawaban.

Langkah 2

Pada langkah ini, deskripsi dari penghitungan key point yang telah ditentukan pada langkah sebelumnya.

Langkah 3

Pada langkah ini, key point dari lembar jawaban dicocokkan dengan key point lembar kunci jawaban. Kita menggunakan FLANN (Fast Library for Approximate Nearest Neighbour) untuk menghilangkan field yang tidak perlu dicocokkan.

Langkah 4

Setelah field yang akan dicocokkan telah ditemukan, kesamaan dari lembar jawaban akan ditemukan dengan transformasi matrix.

Langkah 5

Setelah koordinat lokasi lingkaran jawaban ditemukan, diproyeksikan mana lingkaran yang benar dan tidak.

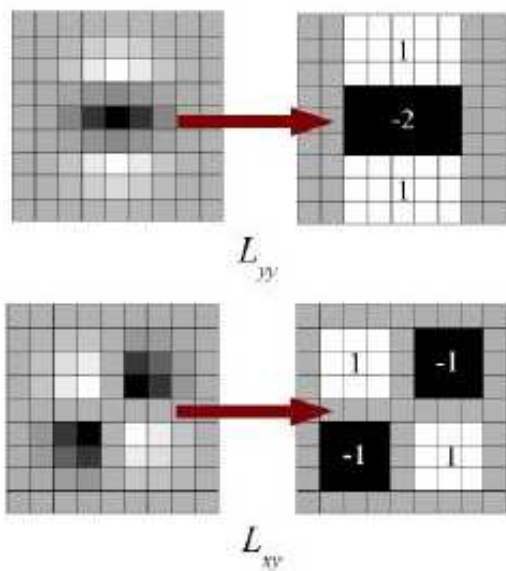
Penjelasan lebih lanjut dari rangkaian langkah pada arsitektur umum metode yang diajukan adalah sebagai berikut:

SURF (Speed-Up Robust Features)

Pada tahun 2006, tiga orang, Bay, H., Tuytelaars, T. dan Van Gool, L, menerbitkan makalah lain, "SURF: *Speeded Up Robust Features*" yang

memperkenalkan algoritma baru yang disebut SURF.

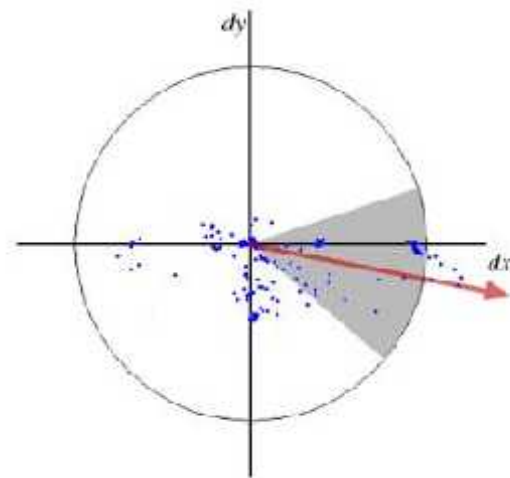
SURF berjalan sedikit lebih jauh dan mendekati LoG dengan Box Filter. Gambar di bawah menunjukkan demonstrasi semacam perkiraan. Satu keuntungan besar dari pendekatan ini adalah, konvolusi dengan filter kotak dapat dengan mudah dihitung dengan bantuan gambar integral. Dan itu bisa dilakukan secara paralel untuk skala yang berbeda. Juga SURF bergantung pada determinan matriks Hessian untuk skala dan lokasi.



Gambar 6. Box Filter

Untuk penugasan orientasi, SURF menggunakan respons wavelet dalam arah horizontal dan vertikal untuk lingkungan dari ukuran 6s. Bobot gaussian yang memadai juga diterapkan untuk itu. Kemudian mereka diplot dalam ruang seperti yang diberikan di bawah gambar. Orientasi dominan diperkirakan dengan

menghitung jumlah semua respons dalam jendela orientasi geser sudut 60 derajat. Hal yang menarik adalah, respon wavelet dapat ditemukan dengan mudah menggunakan gambar integral dengan skala apa pun. Untuk banyak aplikasi, invarian rotasi tidak diperlukan, jadi tidak perlu mencari orientasi ini, yang mempercepat proses. SURF menyediakan fungsi yang disebut Upright-SURF atau U-SURF. Ini meningkatkan kecepatan dan kuat hingga $\pm 15^\circ$. OpenCV mendukung keduanya, tergantung pada bendera, tegak. Jika 0, orientasi dihitung. Jika 1, orientasi tidak dihitung dan lebih cepat.



Gambar 7. Integral SURF

Untuk deskripsi fitur, SURF menggunakan respons Wavelet dalam arah horizontal dan vertikal (sekali lagi, penggunaan gambar integral membuat semuanya lebih mudah). Lingkungan ukuran 20X20 diambil di sekitar keypoint di mana ukurannya. Ini dibagi menjadi sub

wilayah 4×4 . Untuk setiap subregion, respon wavelet horisontal dan vertikal diambil dan vektor dibentuk seperti ini, $v = (\sum \{d_x\}, \sum \{d_y\}, \sum \{|d_x|\}, \sum \{|d_y|\})$. Ini ketika direpresentasikan sebagai vektor memberikan deskripsi fitur SURF dengan total 64 dimensi. Turunkan dimensi, semakin tinggi kecepatan komputasi dan pencocokan, tetapi berikan ciri khas yang lebih baik.

Untuk kekhasan lebih, deskriptor fitur SURF memiliki versi 128 dimensi yang diperluas. Jumlah d_x dan $|d_x|$ dihitung secara terpisah untuk $d_y < 0$ dan $d_y \geq 0$. Demikian pula, jumlah d_y dan $|d_y|$ dipisah berdasarkan tanda d_x , sehingga menggandakan jumlah fitur. Ini tidak menambah banyak kerumitan komputasi. OpenCV mendukung keduanya dengan mengatur nilai flag yang diperpanjang dengan 0 dan 1 untuk 64-dim dan 128-dim masing-masing (default adalah 128-redup).

Peningkatan penting lainnya adalah penggunaan tanda Laplacian (trace of Hessian Matrix) untuk titik bunga yang mendasarinya. Ini tidak menambah biaya komputasi karena sudah dihitung selama deteksi. Tanda Laplacian membedakan gumpalan terang pada latar belakang gelap dari situasi sebaliknya. Pada tahap pencocokan, kami hanya membandingkan fitur jika mereka memiliki jenis kontras

yang sama (seperti yang ditunjukkan pada gambar di bawah). Informasi minimal ini memungkinkan pencocokan lebih cepat, tanpa mengurangi kinerja pendeskripsi.



Gambar 8. SURF Matching

Singkatnya, SURF menambahkan banyak fitur untuk meningkatkan kecepatan di setiap langkah. Analisis menunjukkan itu 3 kali lebih cepat daripada SIFT sementara kinerja sebanding dengan SIFT. SURF baik dalam menangani gambar dengan kabur dan rotasi, tetapi tidak pandai menangani perubahan sudut pandang dan perubahan iluminasi.

FLANN(Fast Library for Approximate Nearest Neighbour)

FLANN adalah *library* untuk melakukan pencarian benda terdekat dengan cepat di ruang dimensi tinggi. Ini berisi kumpulan algoritma yang ditemukan berfungsi paling baik untuk pencarian benda terdekat dan sistem untuk secara otomatis memilih algoritma terbaik dan parameter optimal tergantung pada dataset. FLANN ditulis dalam bahasa C++ dan berisi binding untuk bahasa-

bahasa pemrograman berikut: C, MATLAB dan Python.

dilihat desain menu utama pada aplikasi yang dibuat.

HASIL DAN PEMBAHASAN

Hasil Penelitian

Dari hasil penelitian, penulis mengamati penggunaan *image recognition* dalam aplikasi ini dapat membuat pemeriksaan lembar jawaban lebih efisien.

Dalam percobaan penulis dalam penerapan *image recognition* ini, tingkat keakuratan aplikasi sangat bagus. Berikut merupakan data yang didapat penulis dalam percobaan tingkat keakuratan aplikasi.

Tabel 1. Tabel Hasil Keakuratan

Percobaan	Jumlah Soal	Terbaca	Error
1	40	40	0
2	40	39	1
3	40	39	1
4	40	40	0
5	40	38	2

Pembahasan

Pembahasan aplikasi yang dibuat terdiri atas beberapa proses, yaitu :

Halaman Utama

Halaman Utama dibuat sebagai langkah jikalau pengguna belum memasukkan data kunci jawaban maka tombol kunci jawaban dapat dipilih dan apabila kunci jawaban telah dibuat, maka pengguna dapat langsung memilih tombol lembar jawaban. Pada gambar 9 dapat



Gambar 9. Halaman Utama

Halaman Kunci Jawaban

Halaman ini dibuat untuk menginput kunci jawaban agar di *recognize* dengan lembar jawaban nantinya. Pertama, pengguna harus menentukan jenis kertas yang akan dibuat lalu diberi nama kunci jawabannya. Pengisian kunci jawaban dilakukan dengan memilih nomor soal lalu memilih jawaban yang benar pada nomor. Setelah itu, tekan tombol set untuk mengisi jawaban yang benar pada soal tersebut. Lakukan hal tersebut pada tiap nomor. Setelah semua jawaban telah diisi, kemudian kunci jawaban disimpan. Pada gambar 10 dapat dilihat desain kunci jawaban yang dibuat.



Gambar 10. Halaman Kunci Jawaban

Halaman Periksa Lembar Jawaban

Halaman ini dibuat untuk mengoperasikan *image recognition* yang mencocokkan kunci jawaban dengan lembar jawaban. Lembar jawaban terlebih dahulu ditangkap lewat webcam lalu diproses. Setelah di-*recognize*, kemudian hitung jumlah lingkaran yang sesuai dengan kunci jawaban. Pada gambar 11 dapat dilihat desain form periksa lembar jawaban.



Gambar 11. Halaman Periksa Lembar Jawaban

SIMPULAN

Berdasarkan pembahasan pada bab – bab sebelumnya maka pada bab ini penulis dapat mengambil kesimpulan bahwa penerapan *image recognition* membuat tingkat keakuratan aplikasi menjadi tinggi, sehingga meminimalkan kesalahan dalam membaca lembar jawaban. Aplikasi ini dapat memeriksa secara real-time sehingga mempercepat pemeriksaan bagi para guru. Adapun lembar jawaban ujian

yang dapat diperiksa ialah lembar jawaban komputer dalam berbagai ukuran.

DAFTAR PUSTAKA

- Darmayuda, Ketut, (2014), Aplikasi Basis Data Dengan Visual Basic.Net, Cetakan Pertama, Bandung : Informatika Bandung.
- Sukanto, Rosa A dan M.Shalahuddin, (2014), Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek. Bandung : Informatika.
- Salahudin, (2013), <http://www.ijns.org/journal/index.php/speed/article/view/1124>, diakses pada 20 Februari 2018 pukul 17.10 WIB, Medan.
- Sukarto, (2015), <http://www.radarplanologi.com/2015/11/pengertian-transportasi-manfaat-fungsi-jenisnya.html>, diakses pada 25 Maret 2017 pukul 13.00 WIB, Medan.