

RANCANG BANGUN APLIKASI KOREKSI LEMBAR JAWABAN KOMPUTER MENGUNAKAN METODE DETEKSI TEPI CANNY

Bayu Putra

Program Studi S1 Teknik Informatika, Fakultas Teknologi Informasi, Universitas Hasyim Asy'ari
Email : bayup.unhasy@gmail.com

I Kadek Dwi Nuryana

Fakultas Teknologi Informasi, Universitas Hasyim Asy'ari
Email : dwinuryana@unesa.ac.id

Reza Augusta Jannatul Firdaus

Program Studi S1 Sistem Informasi, Fakultas Teknologi Informasi, Universitas Hasyim Asy'ari
Email : rezafirdaus@unhasy.ac.id

ABSTRAK

Penilaian Akhir Semester adalah salah satu bagian dari proses evaluasi hasil belajar yang menggunakan Lembar Jawaban Komputer (LJK) sebagai tempat siswa menuliskan jawaban. Umumnya LJK dikoreksi menggunakan *scanner* tertentu yang harganya relatif mahal, sehingga beberapa sekolah memilih untuk mengoreksinya secara manual termasuk di MA Ar-Rahman Sumoyono. Untuk mengatasi masalah yang ada, maka penelitian ini adalah salah satu solusi yang memanfaatkan *smartphone* dan teknologi pengolahan citra digital khususnya menggunakan metode deteksi tepi *canny*. Rancang bangun aplikasi ini nantinya akan mengkombinasikan metode deteksi tepi *canny* untuk mencari tepi dari citra dan fungsi *hough circle transformation* untuk mendeteksi lingkaran yang telah dihitamkan. Rancang bangun aplikasi ini juga menggunakan nilai parameter *canny* yang dapat dirubah oleh *user*, mengingat tingkat kecerahan pada setiap pengambilan gambar berbeda-beda. Hasil dari citra yang telah terdeteksi jawabannya akan dilakukan proses *tracking* untuk menentukan jawaban yang dipilih. Jawaban dari siswa nantinya akan dicocokkan dengan kunci jawaban yang telah disimpan dalam *database SQLite*. Hasil dari pengujian terhadap aplikasi yang telah dibuat menggunakan 30 sampel LJK, didapatkan rata-rata presentase akurasi kesesuaian hasil deteksi jawaban dengan jawaban sesungguhnya di LJK adalah 91,41 %.

Kata Kunci: android, LJK, *canny*.

ABSTRACT

Last Semester Evaluation is part of study result evaluation which use Computer Answer Sheet as the place for student to write the answers. Computer Answer Sheet generally corrected by specific scanner with the expensive price, so some school choose to correct the answer sheet manually like at Ar-Rahman Sumoyono Islamic Senior High School. To solve that problem, this research is one of some solution which use the benefit of smartphone and digital image processing technology specially canny edge detection method. This prototype will combine canny edge detection method to find the edge of image and hough circle transformation to detect marked answers. This prototype use the convertible canny parameter value by user, consider the brightness value of every captured image is different. Result of detected answer with hough circle transformation will be tracked to determine the chosen answer. The chosen answer by students will be saved in array, and will be matched with the answer keys which saved in SQLite database. The result of comparison between the result of detected answer with the real answer based on 30 answer sheet samples is 91,41 %.

Keywords: android, answer sheet, *canny*.

PENDAHULUAN

Pendidikan adalah serangkaian proses yang berisikan teknik dan metode belajar mengajar sesuai dengan standar yang telah ditetapkan yang bertujuan untuk menyebarkan pengetahuan (Siagian, 2006). Pendidikan memiliki salah satu hal memiliki peran yang urgen, yakni proses evaluasi hasil belajar. Proses ini dapat berupa Penilaian Harian, Penilaian Tengah Semester, dan Penilaian Akhir Semester. Pelaksanaan Penilaian Akhir Semester tidak dapat dilepaskan dari Lembar Jawaban Komputer (LJK) yang merupakan lembar kertas yang digunakan siswa untuk menuliskan jawaban.

Adapun proses koreksi LJK Penilaian Akhir Semester di MA Ar-Rahman masih manual, yakni dikoreksi sendiri oleh Bapak / Ibu Guru dengan mencocokkan manual hasil jawaban setiap siswa dengan kunci jawaban yang ada. Hal ini

Rancang Bangun Aplikasi Koreksi Lembar Jawaban Komputer Menggunakan Metode Deteksi Tepi Canny

terjadi karena madrasah belum mampu membeli *scanner* khusus yang digunakan untuk memeriksa dan mengoreksi LJK Penilaian Akhir Semester.

Melihat kondisi yang ada di madrasah tersebut, maka muncul inisiatif untuk membantu mempermudah Bapak/Ibu Guru di MA Ar-Rahman Sumoyono dalam mengoreksi LJK Penilaian Akhir Semester dengan membuat rancang bangun / membuat sebuah *prototype* aplikasi koreksi LJK berbasis android dengan menggunakan metode deteksi tepi *canny*.

Berdasarkan latar belakang yang telah dijelaskan maka dapat diperoleh rumusan masalah, yakni bagaimana membuat rancang bangun aplikasi koreksi lembar jawaban penilaian akhir semester di MA Ar-Rahman Sumoyono menggunakan metode deteksi tepi *canny*.

METODE

Android

Android adalah sistem operasi berbasis *linux* bagi telepon seluler seperti *smartphone* dan *tablet*. *Android* juga menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri yang bisa digunakan untuk berbagai macam piranti gerak (Safaat, 2012). Sampai saat ini, *android* mengalami perkembangan yang cukup pesat, mulai dari versi pertama yakni *android* versi 1.1 hingga yang terbaru saat ini yakni *android* versi *Pie*.

Citra Digital

Citra adalah gambar yang terdapat dalam bidang 2 dimensi. Apabila ditinjau dari sudut pandang matematis, citra merupakan fungsi yang berkesinambungan dari intensitas cahaya pada bidang dua dimensi (Sitorus, 2006).

Satuan terkecil dari data citra digital adalah *bit*, yaitu angka biner, 0 atau 1. Kumpulan dari data sejumlah 8 *bit* adalah sebuah unit data yang disebut *byte*, dengan nilai dari 0 – 255. Kumpulan *byte* ini dengan struktur tertentu bisa dibaca oleh *software* dan disebut citra digital 8-bit.

Pixel (*picture element*) adalah sebuah titik yang merupakan elemen paling kecil pada sebuah citra. Angka numerik (*1 byte*) dari pixel disebut *digital number* (DN). DN bisa ditampilkan dalam warna kelabu, berkisar antara putih dan hitam (*grayscale*), tergantung level energi yang terdeteksi. Pikel yang disusun dalam susunan yang benar akan membentuk sebuah citra.

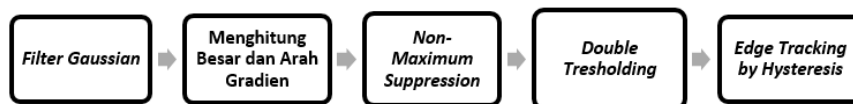
Deteksi Tepi

Tepi adalah daerah di mana terdapat perubahan intensitas warna yang cukup tinggi, atau bisa juga disebut batas dari suatu objek. Proses deteksi tepi (*edge detection*) akan melakukan konversi terhadap daerah ini menjadi dua macam nilai yaitu intensitas warna rendah atau tinggi, contoh bernilai 0 atau 1 (Lusiana, 2013).

Deteksi tepi merupakan salah satu operasi dasar dalam pemrosesan citra digital. Deteksi tepi sangat diperlukan pada tahap sebelum pemrosesan segmentasi citra dalam hal klasifikasi citra. Tepi dari objek pada suatu citra dapat dideteksi dari perbedaan tingkat keabuannya (Purnomo, 2010).

Metode Canny

Canny adalah metode deteksi tepi yang dikembangkan menggunakan algoritma multi-tahap untuk mendeteksi berbagai tepi dari sebuah citra oleh *John F. Canny*. Langkah-langkah dari deteksi tepi *canny* adalah sebagai berikut (Zhou, 2011) :



Gambar 1. Tahapan metode *canny*

Filter Gaussian

Filter Gaussian adalah *filter 2D convolution operator* yang digunakan untuk membuat suatu gambar menjadi lebih halus dengan menghilangkan atau mengurangi *noise* pada gambar tersebut. *Filter Gaussian* memiliki penapis sebagai berikut :

$$\frac{1}{159} \begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix} \quad (1)$$

Menghitung besar dan arah gradien

Penghitungan besar dan arah gradien menggunakan *operator sobel* dengan melakukan pencarian secara horizontal (G_x) dan secara vertikal (G_y).

$$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} \quad (2)$$

$$G_y = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \quad (3)$$

Keterangan :

G_x = Matriks *filter* horizontal

G_y = Matriks *filter* vertikal

Besar dan arah gradien dicari dengan rumus :

$$G = \text{sqrt}(G_x^2 + G_y^2) \quad (4)$$

$$\theta = \arctan\left(\frac{G_y}{G_x}\right) \quad (5)$$

Keterangan :

G = Besar gradien

G_x = Hasil konvolusi dengan matriks *filter* horizontal

G_y = Hasil konvolusi dengan matriks *filter* vertikal

Non-Maximum Suppression (Perampingan garis tepi).

Langkah ini bertujuan membuang potensi gradien yang bukan merupakan maksimal lokal pada arah tepi di posisi piksel tersebut. Hanya nilai maksimum yang ditandai sebagai tepi sehingga didapatkan garis tepi yang lebih ramping.

Double Thresholding.

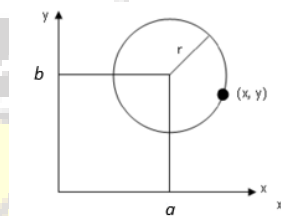
Untuk membuat gambar biner, diterapkan dua buah *thresholding* yaitu *low thresholding (Threshold1)* dan *high thresholding (Threshold2)*. Nilai yang kurang dari *Threshold1* akan diubah menjadi hitam (bernilai 0) dan nilai yang lebih dari *Threshold2* diubah menjadi putih (bernilai 255).

Edge Tracking by Hysteresis.

Tepi akhir ditentukan dengan menekan semua sisi yang tidak terhubung dengan tepian yang kuat. Piksel yang besar gradiennya berada di antara *Threshold1* dan *Threshold2*, namun terhubung dengan piksel tetangga yang besar gradiennya berada di atas *Threshold2*, maka piksel tersebut merupakan sebuah tepi. Apabila terdapat piksel yang besar gradiennya berada di antara *Threshold1* dan *Threshold2*, namun tidak bertetangga / terhubung dengan piksel yang besar gradiennya berada di atas *Threshold2*, maka piksel tersebut bukan sebuah tepi.

Hough Circle Transformation

Hough Circle Transformation digunakan untuk mendeteksi keberadaan lingkaran pada sebuah citra. Pada lingkaran dilakukan pemberian nilai *threshold* tertentu yang dapat digunakan dalam tahap pengenalan citra yang diuji. Fungsi *hough circle* ini dapat digambarkan dalam ruang parameter sehingga sumbu x adalah nilai a dan sumbu y adalah nilai b . sedangkan sumbu z adalah jari-jari (Bradski, 2008).



Gambar 2. Ilustrasi koordinat

Dalam penerapan *hough circle transformation*, ditentukan jari-jari minimum ($r - \text{min}$) dan jari-jari maksimum ($r - \text{max}$) yang nantinya hanya akan mengenali karakteristik lingkaran sesuai dengan $r - \text{min}$ dan $r - \text{max}$ yang telah ditentukan.

OpenCV

OpenCV (Open Source Computer Vision) adalah sebuah pustaka / *library open source*, yang telah dioptimalkan untuk analisa gambar dan video dimana di dalamnya berisi lebih dari 500 algoritma. *OpenCV* telah menjadi *development tool* yang paling banyak diadopsi oleh para pengembang pada bidang *computer vision* sejak diperkenalkan pada tahun 1999 (Laganier, 2011).

Rancang Bangun Aplikasi Koreksi Lembar Jawaban Komputer Menggunakan Metode Deteksi Tepi Canny

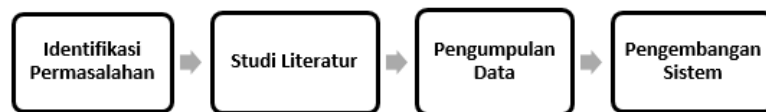
Library ini ditulis dalam bahasa *C* dan *C++*, dan dapat dijalankan dalam beberapa *platform*, diantaranya *Linux*, *Windows*, *Mac OS X*, serta *Android*. *OpenCV* memiliki fokus kuat pada aplikasi *realtime* dan telah didesain untuk komputasi yang efisien. *OpenCV* juga dapat bekerja dengan *multicore processor*. (Bradski. 2008)

SQLite

SQLite adalah mesin *database SQL* yang tertanam. *SQLite* digunakan oleh produk-produk terkemuka seperti *Adobe Integrated Runtime (AIR)*, *Airbus* pada software penerbangan mereka, *Python ship* dengan *SQLite*, *PHP*, dan masih banyak lagi. Dalam *mobile domain*, *SQLite* adalah pilihan yang sangat populer karena sifatnya yang ringan. *Apple* menggunakannya dalam *iPhone* dan *Google* dalam sistem operasi *Android*. *SQLite* digunakan sebagai format berkas aplikasi, *database* untuk *gadget* elektronik, *database* untuk situs web, dan sebagai suatu perusahaan *Relational Data Base Management System (RDBMS)* (Aditya, 2014).

Metodologi Penelitian

Aplikasi yang akan dibuat dalam penelitian ini adalah aplikasi berbasis *android*. Format lembar jawaban yang dijadikan objek penelitian ini hanyalah lembar jawaban yang dikeluarkan oleh Badan Koordinasi Madrasah Aliyah (BKMA) Kabupaten Jombang. Aplikasi ini hanya mendeteksi bulatan pada jawaban pilihan ganda saja, tidak mendeteksi bulatan nama siswa, nomor peserta siswa, mata pelajaran, serta jawaban isian. Langkah-langkah yang dilakukan untuk mencapai tujuan dari penelitian ini adalah seperti gambar berikut :



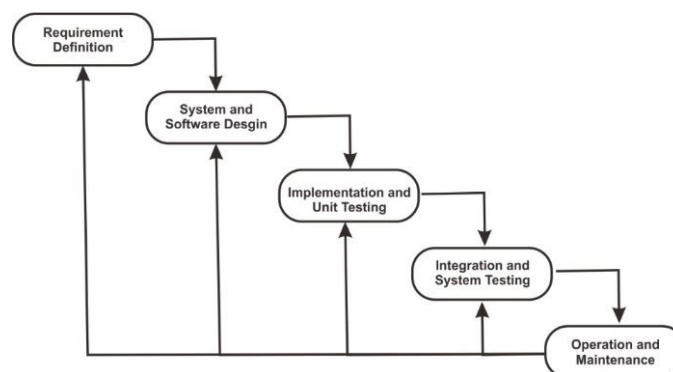
Gambar 4. Diagram alur prosedur penelitian

Metode pengumpulan data yang digunakan adalah studi dokumen, yakni dengan mengumpulkan data LJK yang nantinya digunakan sebagai bahan pengujian kinerja aplikasi. Data yang diambil untuk penelitian ini adalah data *primer*, yakni LJK hasil Penilaian Akhir Semester Ganjil Tahun Pelajaran 2018/2019. Berikut adalah tabel yang berisi data LJK yang digunakan dalam penelitian ini :

Tabel 1. Jumlah Data LJK

No	Mata Pelajaran	Kelas	Jumlah LJK
1	B. Indonesia	XII	21
2	Sosiologi	X	15
3	Sosiologi	XI	15
4	Sosiologi	XII	22
Jumlah			73

Adapun model pengembangan sistem yang digunakan dalam penelitian ini adalah metode *waterfall*. Metode ini sering dinamakan siklus hidup klasik (*classic life cycle*).



Gambar 5. Model pengembangan sistem *waterfall*

Kebutuhan Sistem (Requirement Definition)

Kebutuhan fungsional dari aplikasi ini adalah sebagai berikut :

- 1) Pengguna dapat mengatur kunci jawaban
- 2) Pengguna dapat melihat kunci jawaban yang telah diatur
- 3) Pengguna dapat mengambil citra LJK
- 4) Pengguna dapat mengetahui jawaban yang terdeteksi dari LJK siswa
- 5) Pengguna dapat melihat jumlah jawaban benar yang diperoleh siswa.

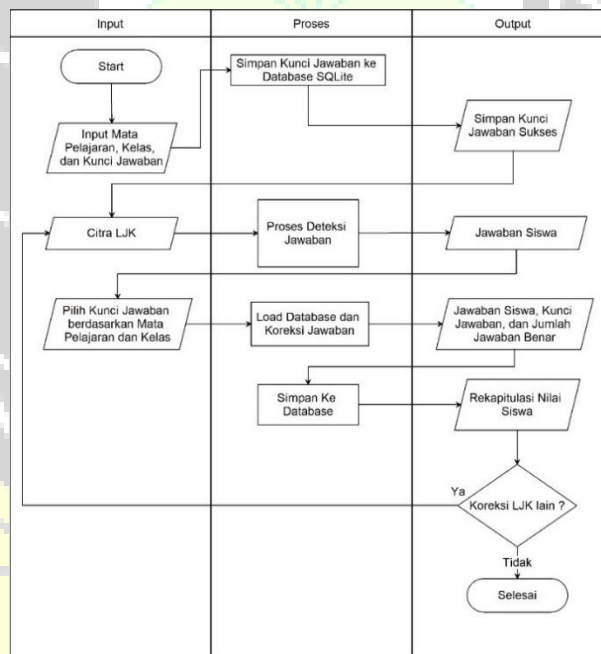
Sedangkan kebutuhan *non-fungsional* dari aplikasi ini adalah sebagai berikut :

- 1) *Smartphone android* dengan versi 5.0 (*Lollipop*) ke atas. Alasan dari dipilihnya versi *Lollipop* sebagai versi terendah untuk aplikasi ini adalah sebagai berikut :
 - a. Sudah mendukung adanya kamera autofocus, yang membantu memfokuskan proses pengambilan citra LJK.
 - b. *API Android* versi *Lollipop* sudah mendukung adanya *Camera2*, yang dalam hal ini *Camera2* merupakan salah satu komponen inti dari aplikasi ini.
 - c. *Smartphone* yang dimiliki oleh hampir semua Bapak/ Ibu Guru sudah versi *Lollipop* dan *Marshmallow*.
- 2) *RAM smartphone* minimal 1 GB
- 3) *Smartphone android* yang memiliki kamera autofocus dengan kondisi kamera masih normal (kamera tidak cacat).
- 4) Penyimpanan *internal smartphone* minimal 8 GB

Desain Sistem (System and Software Design)

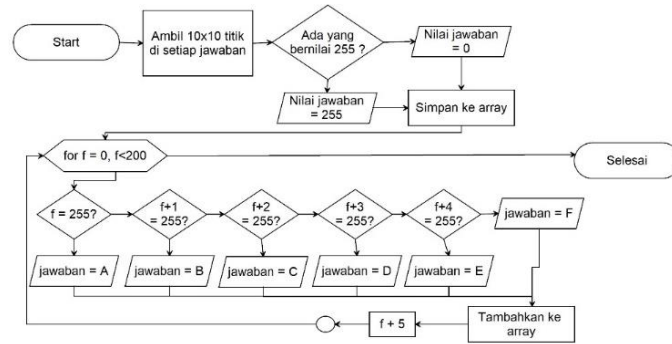
Desain Flowchart Sistem

Berikut adalah desain alur dari aplikasi yang akan dibuat dalam penelitian ini :



Gambar 6. Desain Flowchart Sistem

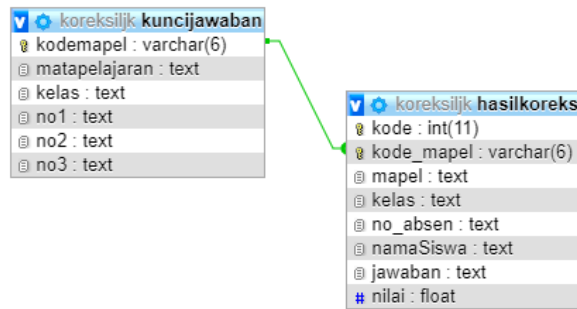
Rancang Bangun Aplikasi Koreksi Lembar Jawaban Komputer Menggunakan Metode Deteksi Tepi Canny



Gambar 9. flowchart proses penentuan jawaban

Desain Database

Berikut adalah rancangan dari database SQLite :



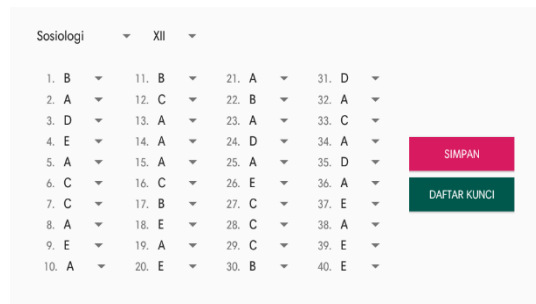
Gambar 10. Desain relasi tabel

Konstruksi Aplikasi (Implementation and Unit Testing)

Proses konstruksi aplikasi ini menggunakan bahasa pemrograman Java yang dibangun menggunakan Android Studio IDE dengan bantuan library OpenCV. Bentuk dari aplikasi ini nantinya adalah file berekstensi .apk yang nantinya bisa dipasang pada smartphone pengguna. Sedangkan alasan dipilihnya library OpenCV dikarenakan OpenCV merupakan library yang sangat mendukung untuk proses pengolahan citra digital, termasuk deteksi tepi canny dan fungsi hough circle transformation yang digunakan dalam penelitian ini.

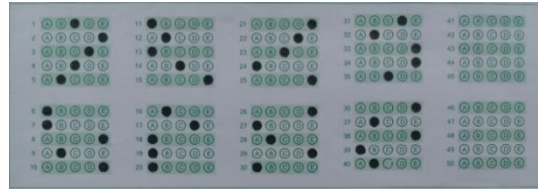
HASIL DAN PEMBAHASAN

Berikut akan dijelaskan hasil dan pembahasan dari aplikasi yang telah dibuat dalam penelitian ini :



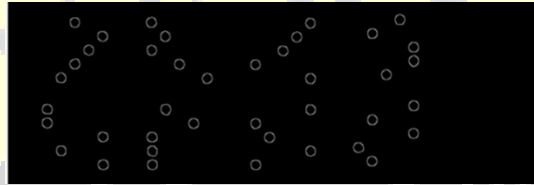
Gambar 11. Atur kunci jawaban

Gambar di atas adalah tampilan dari proses atur kunci jawaban yang dilakukan user sebelum melakukan pengambilan citra LJK. Data kunci jawaban nantinya akan disimpan pada database SQLite.



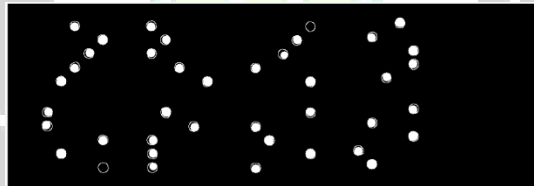
Gambar 12. Contoh citra LJK yang telah diambil

Gambar di atas adalah contoh citra LJK yang telah diambil dan telah dilakukan proses *resize* menggunakan aplikasi yang telah dibuat. Langkah selanjutnya adalah merubah citra LJK di atas menjadi *grayscale*. Konversi ini bertujuan agar citra yang telah diambil bisa diproses ke proses selanjutnya.



Gambar 14. Hasil metode *canny* pada citra LJK

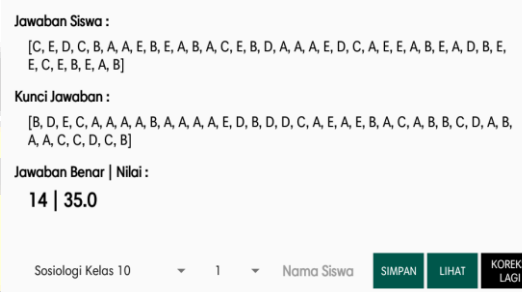
Gambar di atas adalah hasil pemrosesan citra *grayscale* menggunakan metode deteksi tepi *canny*. Langkah selanjutnya ialah proses deteksi lingkaran yang telah dihitamkan menggunakan fungsi *hough circle transformation*.



Gambar 15. Hasil *hough circle transformation*

Berdasarkan gambar di atas, citra LJK hasil metode *canny* pada gambar 13 dapat dideteksi sebagai lingkaran oleh fungsi *hough circle transformation* dengan mewarnai lingkaran jawaban dengan warna putih. Nilai jari-jari lingkaran minimum yang dapat terdeteksi adalah 5, dan nilai maksimumnya adalah 12. Jika terdapat lingkaran yang memiliki jari-jari kurang dari nilai minimum atau lebih dari nilai maksimum, maka tidak akan terdeteksi sebagai lingkaran.

Langkah selanjutnya ialah proses *tracking* jawaban dari lingkaran yang telah terdeteksi. Berikut adalah salah satu contoh hasil proses *tracking* jawaban dari hasil deteksi lingkaran menggunakan *hough circle transformation* :



Gambar 16. Hasil *tracking* jawaban

Selanjutnya dilakukan proses penghitungan nilai keakuratan hasil deteksi jawaban dengan jawaban yang dijawab oleh siswa pada LJK menggunakan 30 sample LJK.

Rancang Bangun Aplikasi Koreksi Lembar Jawaban Komputer Menggunakan Metode Deteksi Tepi Canny

Tabel 2. Hasil pengujian aplikasi dengan sampel

No	Nama	Mata Pelajaran	Hasil Akurasi
1	Siti Nur Indah S	Sosiologi X	90 %
2	M. Indra S	Sosiologi X	97.5 %
3	Nur Laily I	Sosiologi X	95 %
4	Yatim Masruroh	Sosiologi X	97.5 %
5	Kurnia Anggraeni	Sosiologi X	95 %
6	Diva Nur O	Sosiologi X	97.5 %
7	Riki Andika R	Sosiologi X	87.5 %
8	Tri Afrizal	Sosiologi X	95 %
9	Ari Wahyudi	Sosiologi X	97.5 %
10	Siti Fitri Ning A	Sosiologi X	95 %
11	Akhmad	Sosiologi XI	92.5 %
12	Heny Lukita S	Sosiologi XI	97.5 %
13	M. Andre	Sosiologi XI	97.5 %
14	M. Amirul Fahmi	Sosiologi XI	95 %
15	M. Rizal Adityaa	Sosiologi XI	92.5 %
16	Nurma Dwi L	Sosiologi XI	80 %
17	Siti Aisyah	Sosiologi XI	90 %
18	Aqmariana S	Sosiologi XI	82.5 %
19	Sayyidatun M	Sosiologi XI	97.5 %
20	M. Riski Joan H	Sosiologi XI	82.5 %
21	Agus Ja'far S	B Indo XII	97.5 %
22	Wulandari	B Indo XII	90 %
23	Rizal Widiensyah	B Indo XII	85 %
24	Auliya Putri A	B Indo XII	92.5 %
25	M. Dwi Yanti	B Indo XII	82.5 %
26	Lailatul Fitriyah	B Indo XII	90 %
27	M. Ilham Ari P	B Indo XII	82.5 %
28	Bayu Aji	B Indo XII	85 %
29	Rifqi Arifianto	B Indo XII	92.5 %
30	Heeba Alya R	B Indo XII	90 %
Rata-rata presentase akurasi			91.41 %

Berdasarkan hasil pengujian di atas, diperoleh rata-rata presentase kesesuaian deteksi jawaban dengan jawaban yang dijawab siswa adalah 91.41%.

PENUTUP

Simpulan

Berdasarkan hasil pengujian sistem yang telah dibangun terhadap LJK siswa, dapat ditarik kesimpulan bahwa terdapat beberapa jawaban yang tidak terdeteksi sebagai jawaban. Beberapa penyebab dari hal ini adalah perbedaan tingkat kecerahan dari proses pengambilan gambar dan kurang sesuainya pengambilan gambar dengan *ROI* yang telah ditentukan. Dari keseluruhan hasil percobaan dengan 30 sampel LJK, diperoleh rata-rata kesesuaian antara jawaban yang terdeteksi dengan jawaban yang dijawab oleh siswa ialah 91.41 %.

Saran

Untuk selanjutnya, dapat dikembangkan aplikasi koreksi LJK berbasis *android* yang dapat secara otomatis meluruskan citra LJK yang posisinya melenceng, sehingga *user* tidak perlu menyesuaikan batas LJK dengan batas persegi yang telah ditetapkan dalam aplikasi, dan dapat diperoleh jawaban yang lebih akurat. Selain itu juga dapat dikembangkan sistem berbasis web untuk menyimpan nilai siswa hasil koreksi dari aplikasi ini.

DAFTAR PUSTAKA

- Aditya, Sunny Kumar dan Vikash Kumar Khan. 2014. *Android SQLite Essentials*. Birmingham : Packt Publishing Ltd.
- Bradski, G., dan Kaebler, A. 2008. *Learning OpenCV, Computer Vision with the OpenCV Library*. Sebastopol : O'Reilly Media.
- Laganier, R., 2011. *OpenCV 2 Computer Vision Application Programming Cookbook. 1st ed.* Birmingham: Packt Publishing Ltd.
- Lusiana, Veronica. 2013. Deteksi Tepi pada Citra Digital Menggunakan Metode *Kirsch* dan *Robinson*. Jurnal Teknologi Informasi DINAMIK. Vol. 18 (2) hal 182-189.
- Purnomo, M. H., Muntasa, A. 2010. *Konsep Pengolahan Citra Digital dan Ekstraksi Fitur*. Yogyakarta : Graha Ilmu.
- Safaat, Nazrudin. 2012 (*Edisi Revisi*). *Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android*. Bandung : Informatika.
- Siagian, P. Sondang. 2006. *Manajemen Sumber Daya Manusia*. Jakarta : Bumi Aksara.
- Sitorus, Syahriol, dkk. 2006. *Pengolahan Citra Digital*. Medan : USU Press.
- Zhou, Ping, dkk. 2011. "An Improved Canny Algorithm for Edge Detection". *Journal of Computational Information System*. Vol. 75 (5). Hal 1516-1523.