

## **Exam Answer Detection using Proposed Nested Loop Pixel Checking Based on Image Processing**

### *Sistem Deteksi Lembar Jawaban Ujian dengan Menggunakan Metode Nested Loop Pixel Checking yang Diusulkan Berbasis Pengolahan Citra Digital*

Reza Augusta Jannatul Firdaus  
Afaizah

Indana Lazulfa

Bayu Putra

Program Studi Sistem Informasi, Fakultas Teknologi  
Informasi, Universitas Hasyim Asy'ari

Program Studi Teknik Informatika, Fakultas Teknologi  
Informasi, Universitas Hasyim Asy'ari

Program Studi Teknik Informatika, Fakultas Teknologi  
Informasi, Universitas Hasyim Asy'ari

*Tracking object is one of the most discussed problems in digital images processing. The objectives of tracking objects include the counting, correction, classification and so on. This study discusses how tracking objects, in this case the multiple choice chosen by students, is carried out on the exam answer sheet using the proposed nested loop pixel checking method. The answer sheet used here is the answer sheet for the exam from SMK Khoiriyah Hasyim Tebuireng Jombang because each school has a different exam answer sheet format so that the coordinate parameters of each answer sheet between schools must be different. Before the nested loop pixel checking method, preprocessing was carried out first after taking the image, namely cropping the Region of Interest, conversion to grayscale, filtering, and edge detection using the adaptive Canny edge detection method. The process of nested loop pixel checking is done by iterating successively with steps per answer option, per number of answers, and per column. In addition, a special case is given to find out whether this proposed method works as desired. The results show that this method can track answers for both normal and special cases.*

## **Pendahuluan**

Pengolahan citra digital merupakan salah satu cabang ilmu yang mempelajari tentang bagaimana komputer mengolah sebuah input dalam bentuk citra (gambar) untuk memperoleh informasi yang dibutuhkan manusia. Informasi yang dibutuhkan dapat berupa pengenalan objek seperti wajah, sidik jari, kendaraan, dan sebagainya. Permasalahan tersebut dapat diselesaikan dengan cara pengenalan pola.

Pada dunia pendidikan, salah satu masalah tersebut adalah proses koreksi lembar jawaban ujian. Di beberapa sekolah, proses koreksi lembar jawaban masih manual yaitu dengan mencocokkan satu per satu jawaban pada lembar jawaban dengan kunci jawaban, hal ini dikarenakan pengadaan mesin scanner lembar jawaban yang terbilang mahal. Selain itu terdapat kelemahan pada scanner lembar jawaban komputer yaitu diperlukan pensil 2B dengan ketebalan tertentu yang terkadang dapat menimbulkan kesalahan apabila tidak terhapus secara sempurna ketika hendak mengganti jawaban.

Oleh karena itu, pada penelitian ini dibahas "Deteksi Lembar Jawaban menggunakan Metode Nested Loop Pixel Checking". Metode ini digunakan setelah metode Deteksi Tepi Canny Adaptif diterapkan pada citra lembar jawaban, dengan kata lain dalam bentuk citra biner (hitam dan putih). Sehingga dapat dikatakan bahwa hasil dari metode nested loop pixel checking ini bergantung pada

hasil daripada metode deteksi tepinya, apabila terdapat banyak noise pada hasil deteksi tepi maka hasil dari pixel checking akan tidak sempurna. Oleh karena itu metode yang digunakan pada deteksi tepi ini adalah adaptif Canny yang bersifat adaptif terhadap kondisi gambar<sup>1</sup>. Untuk memfokuskan penelitian, lembar jawaban ujian yang digunakan pada penelitian ini adalah lembar jawaban dari SMK Khoiriyah Hasyim Tebuireng Jombang, hal ini dikarenakan setiap sekolah memiliki format lembar jawaban ujian yang berbeda. Perbedaan format ini berpengaruh pada penentuan parameter koordinat opsi jawaban yang ada.

Proses awal yang dilakukan pada penelitian ini adalah mengonversi gambar (lembar jawaban) ke dalam bentuk grayscale (derajat keabuan), hal ini dilakukan untuk mempercepat proses komputasi. Selanjutnya dilakukan filtering menggunakan median filter untuk mengurangi noise pada gambar dengan tujuan mempermudah proses deteksi. Selanjutnya untuk mendeteksi tepian dari object (dalam hal ini berfokus pada objek lingkaran) dilakukan deteksi tepi Canny. Deteksi tepi Canny yang dilakukan adalah deteksi tepi adaptif Canny sehingga parameter yang digunakan didalamnya menyesuaikan input dari gambar yang akan diolah. Setelah pola lingkaran muncul, proses pelacakan jawaban menggunakan metode nested loop pixel checking ini dilakukan. Hal ini dilakukan untuk mengetahui koordinat dari lingkaran pada masing-masing nomor, sehingga komputer dapat mengenali opsi jawaban yang terpilih. Perlu digaris bawahi bahwa baik buruknya hasil dari proses deteksi jawaban ini bergantung pada hasil dari proses deteksi tepinya.

Penelitian ini lebih berfokus pada bagaimana melakukan tracking dan mendeteksi jawaban yang dipilih oleh siswa pada lembar jawaban ujian. Harapannya, hasil penelitian ini merupakan aplikasi Android yang mampu melakukan koreksi lembar jawaban dengan mudah, dan cepat, serta dapat dioperasikan oleh siapa saja yang membutuhkannya, sehingga dapat mengatasi masalah efektivitas waktu, uang dan energi dalam lembar jawaban proses pencocokan dan menjadi alternatif pengganti mesin pemindai lembar jawaban realistis untuk beberapa sekolah.

## Metodologi Penelitian

Penelitian ini dilakukan dengan melakukan uji coba parameter untuk mengenali opsi jawaban yang terpilih berdasarkan koordinat dari tiap-tiap opsi jawaban yang ada pada lembar jawaban ujian SMK Khoiriyah Hasyim Tebuireng Jombang. Selain itu berikut dijelaskan beberapa literature yang berkaitan dengan penelitian ini diantaranya adalah.

### Citra Digital

Secara harfiah, citra adalah gambar yang terdapat dalam bidang 2 dimensi<sup>2</sup>. Sedangkan jika ditinjau dari sudut pandang matematis, citra merupakan fungsi yang berkesinambungan dari intensitas cahaya pada bidang dua dimensi. Objek dalam bentuk citra dapat terekam karena adanya sumber cahaya yang menerangi objek tersebut, kemudian objek memantulkan kembali sebagian dari berkas cahaya. Pantulan cahaya tersebut kemudian ditangkap oleh alat-alat optik, seperti mata, kamera, alat pemindai (scanner), dan lain-lain. Citra dapat bersifat optik yakni berupa foto, analog berupa sinyal video seperti gambar pada monitor televisi, dan digital yang dapat langsung disimpan pada media penyimpanan magnetik<sup>3</sup>.

Secara matematis, citra digital dapat direpresentasikan dalam fungsi 2 variabel, yakni fungsi (x,y), di mana x dan y adalah titik koordinat horizontal dan vertikal dari piksel gambar<sup>4</sup>. Gambar digital tersusun dari beberapa elemen yang disebut piksel. Masing-masing piksel memiliki nilai yang bergantung pada ukuran bit gambar tersebut. Sebagai contoh apabila terdapat gambar berukuran 8 bit maka tiap piksel berisikan angka bervariasi antara 0-255 atau sebanyak 256 karakter ( $2^8=256$ ) yang merepresentasikan gambar tersebut<sup>5</sup>.

### Filtering

Filtering adalah tahapan untuk mengurangi noise pada gambar<sup>6</sup>, dengan tujuan mempermudah proses deteksi atau proses yang akan dilakukan setelah proses filtering. Filtering termasuk dalam tahapan preprocessing di mana tahapan ini dilakukan untuk proses setelahnya (processing). Pada penelitian ini filter yang digunakan adalah median filter. Proses filtering dengan Median filter adalah dengan cara mensubstitusi nilai intensitas setiap piksel dengan nilai rata-rata piksel tetangganya<sup>7</sup>. Nilai median dihitung dengan mengurutkan jendela (tetangga piksel pusat) dan mengganti bagian tengah jendela dengan nilai median yang diperoleh. Proses ini dapat digambarkan pada [Figure 1](#).<sup>8</sup>

**Figure 1.** Contoh filtering dengan menggunakan median filter

## Metode Deteksi Tepi Canny

Metode deteksi tepi Canny dikembangkan oleh John F. Canny menggunakan algoritma multi-tahap untuk menentukan tepi dari objek pada suatu gambar. [Figure 2](#) merupakan langkah-langkah dari metode deteksi tepi Canny <sup>9, 10</sup>

**Figure 2.** Langkah-langkah pada metode deteksi tepi Canny

Seperti yang terlihat pada [Figure 2](#), proses deteksi tepi Canny terdiri atas 5 langkah. Untuk masing-masing langkah pada [Figure 2](#) akan dijelaskan sebagai berikut.

### 1. Filter Gaussian

Filter Gaussian adalah filter 2D yang digunakan pada proses konvolusi, konvolusi merupakan operator yang digunakan untuk memperhalus gambar dan menghilangkan atau mengurangi noise pada gambar tersebut<sup>11</sup>. Filter Gaussian memiliki penapis sebagai berikut:

$$1159245424912945121512549129424542$$

### 2. Menghitung besarnya arah dan gradient

Penghitungan besar dan arah gradien dilakukan dengan menggunakan operator sobel, berikut merupakan filter yang digunakan secara horizontal (Gx) dan secara vertikal (Gy).

$$G_x = -10+1-20+2-10+1$$

$$G_y = +1+2+1000-1-2-1$$

Keterangan :

$$G = \sqrt{G_x^2 + G_y^2}$$

$$\theta = \arctan(G_y/G_x)$$

Keterangan :

### 3. Non-Maximum Suppression (Perampingan garis tepi).

Langkah ini bertujuan membuang potensi gradien di suatu piksel dari kandidat tepi jika piksel tersebut bukan merupakan maksimal lokal pada arah tepi di posisi piksel tersebut. Tahap ini dilakukan di sepanjang tepi dengan cara menghilangkan piksel – piksel yang tidak dianggap

sebagai tepi. Hanya nilai maksimum yang ditandai sebagai tepi, sedangkan piksel yang tidak memiliki nilai maksimum tidak akan dianggap sebagai tepi (nilainya diubah menjadi 0), sehingga didapatkan garis tepi yang lebih ramping.

#### 4. Double Thresholding

Untuk mengkonversi gambar menjadi citra biner, didefinisikan terlebih dua buah thresholding (parameter) yaitu low thresholding (Threshold1) atau threshold bawah, dan high thresholding (Threshold2) atau threshold atas. **Figure 3** adalah penggambaran dari proses double thresholding :

**Figure 3.** Penerapan Double Threshold

Kedua nilai parameter ini mempengaruhi hasil dari deteksi tepi Canny. Nilai yang kurang dari low threshold (Threshold1) atau threshold bawah akan diubah menjadi hitam (nilai 0) dan nilai yang lebih dari high threshold (Threshold2) atau threshold atas akan diubah menjadi putih nilai 255). Adapun penentuan nilai dari (Threshold1) dan (Threshold2) menyesuaikan kebutuhan pengguna, sehingga masing masing gambar membutuhkan nilai parameter yang tidak selalu sama.

#### 5. Edge Tracking by Hysteresis.

Penentuan piksel yang terdeteksi sebagai tepi pada tahap ini mempertimbangkan hubungan dengan piksel tetangga. Dalam tahap ini digunakan konsep keterhubungan / ketetanggaan piksel. Adapun keterhubungan antara satu piksel dengan piksel lainnya menggunakan jenis 8-ketetanggaan ditunjukkan pada **Figure 4** berikut :

**Figure 4.** Ketetanggaan/ Keterhubungan piksel

Semua piksel yang besar gradiennya berada di antara T1 dan T2, namun terhubung / bertetangga dengan piksel yang besar gradiennya berada di atas T2, maka piksel tersebut dianggap sebagai tepi. Apabila ada piksel yang besar gradiennya berada di antara T1 dan T2, namun tidak terhubung / tidak bertetangga dengan piksel yang besar gradiennya berada di atas T2, maka piksel tersebut tidak dianggap sebagai tepi. Pada penelitian ini digunakan metode deteksi tepi adaptif Canny, dimana parameter bawah dan atas ditentukan menyesuaikan citra input yang akan digunakan, yang dinyatakan dengan persamaan:

$$\text{lower}=\max[0,((1-\sigma)*v)]\text{upper}=\min[0,((1+\sigma)*v)]$$

lower dan upper merupakan batas bawah dan atas dari citra tersebut. Sedangkan  $\sigma$  bernilai 0.33,  $v$  merupakan rata-rata intensitas pada gambar sehingga nilai  $v$  bergantung pada keadaan dan kualitas citra input, hal inilah yang membuat nilai 2 threshold yang digunakan menjadi adaptif.

### Metode yang diusulkan

Secara umum proses pendeteksian jawaban pada lembar jawaban menggunakan metode yang sudah ada, seperti hough circle, atau deteksi objek. Namun dalam studi ini, metode yang digunakan adalah Nested Loop Pixel Checking. Alur dari metode ini adalah memeriksa nilai piksel dari setiap area jawaban, dan mengulang sesuai dengan format are lembar jawaban. Jika area jawaban yang dicentang berisi warna putih atau salah satu pikselnya berisi "255", area tersebut dianggap sebagai jawaban. Namun jika area jawaban yang dicentang tidak mengandung warna putih atau tidak ada satupun piksel yang mengandung "255", maka area tersebut bukan jawaban yang dipilih dan proses perulangan dilanjutkan untuk memeriksa area jawaban lainnya. Perlu digaris bawahi setiap format lembar jawaban menggunakan parameter yang berbeda pada penerapan algoritma ini.

Sebelum proses deteksi jawaban dimulai, tahapan yang harus dilalui adalah proses deteksi tepi adaptif Canny. Hasil dari proses deteksi tepi adaptif canny adalah gambar hitam, dengan beberapa lingkaran putih di tepinya yang direpresentasikan pada [Figure 5](#). Beberapa lingkaran dengan tepi putih diberi nomor sesuai dengan koordinat dari lembar jawaban tersebut. Berikut contoh hasil deteksi tepi dengan adaptive canny pada salah satu lembar jawaban:

**Figure 5.** Hasil deteksi tepi adaptif Canny

Hasil dari proses deteksi lembar jawaban menggunakan nested loop pixel bergantung pada hasil dari deteksi canny. Untuk meminimalisir kesalahan penggunaan parameter pada proses deteksi tepi tersebut maka penelitian ini menggunakan deteksi tepi adaptif Canny.

Proses deteksi jawaban dengan metode pengecekan nested loop pixel ini terdiri dari 3 loop bersarang, yaitu loop per jawaban (A, B, C, D, E), looping per angka (1-10, 11-20, 21-30, 31 -40), dan perulangan per bidang jawaban. Ilustrasi dari proses tersebut direpresentasikan oleh [Figure 6](#)

**Figure 6.** Ilustrasi proses pengecekan dengan menggunakan nested loop pixel

Garis hitam adalah garis yang menunjukkan opsi perulangan per jawaban. Sedangkan garis hijau mewakili perulangan per angka, serta garis merah adalah garis yang menunjukkan perulangan per bidang jawaban. Kotak biru adalah area yang diambil dalam setiap proses perulangan per opsi jawaban yang akan diperiksa nilai pikselnya. Lebih jelasnya proses pengecekan lembar jawaban dengan metode pengecekan nested loop pixel diilustrasikan oleh flowchart yang terdapat pada [Figure 7](#), sedangkan proses penyimpanan jawaban yang terpilih ke dalam array dapat dilihat pada [Figure 8](#).

**Figure 7.** Flowchart nested loop pixel

[Figure 7](#) merupakan alur dari proses pengecekan jawaban yang dipilih ke dalam array. Seperti yang dijelaskan sebelumnya proses dimulai dari sebanyak opsi jawaban, sebanyak nomor pada kolom, dan sebanyak ruas kolom. Sedangkan [Figure 8](#) menjelaskan proses penyimpanan jawaban ke dalam array untuk menyimpan hasil jawaban tersebut apabila terdapat beberapa lembar jawaban siswa yang akan dilakukan pengecekan. Untuk lebih jelaskan langkah tersebut diilustrasikan oleh [Figure 9](#).

**Figure 8.** Flowchart penyimpanan opsi jawaban yang dipilih.

**Figure 9.** Ilustrasi lembar jawaban pada saat diproses

Proses tracking dimulai dengan cara melakukan perulangan bertingkat, mulai dari perulangan per jawaban, per nomor, dan per bagian jawaban. Untuk mengidentifikasi jawaban yang dipilih, program melakukan perulangan untuk menemukan piksel putih yang ada pada lembar jawaban (piksel bernilai 255). Apabila sebuah titik perulangan bernilai 255, maka titik perulangan saat ini adalah jawabannya. Jika nilainya tidak (atau 0), itu akan terus mengulang ke poin berikutnya.

Setelah jawaban berhasil dideteksi, hasilnya dikumpulkan dalam bentuk array. Array yang

terbentuk kemudian dibandingkan dengan kunci jawaban dari database untuk dilakukan proses koreksi. Hasil koreksi akan muncul dan disimpan di database.

## Hasil dan Pembahasan

Fitur aplikasi ini antara lain pengambilan gambar, penyimpanan kunci jawaban, proses koreksi, dan penyimpanan jawaban. Penelitian ini lebih difokuskan pada tahapan pengecekan lembar jawaban sehingga hanya sebagian fitur yang dijelaskan. Proses dimulai dengan mengambil foto lembar jawaban seperti di [Figure 10](#).

**Figure 10.** Foto lembar jawaban

[Figure 10](#) menunjukkan area yang difokuskan dari lembar jawaban, seperti yang terlihat hanya sebagian dari jawaban setiap angka. Hal tersebut dilakukan karena dalam penelitian inilah ROI (Region of Interest) yang menjadi masalah utama penelitian. Setelah itu gambar tersebut dikonversi ke dalam bentuk grayscale. Proses selanjutnya adalah mean filtering yang dapat dilihat pada [Figure 11](#).

**Figure 11.** Hasil dari proses mean filtering.

Mean filtering merupakan salah satu pre-processing yang dilakukan dengan tujuan untuk mereduksi noise pada suatu citra agar proses selanjutnya (deteksi) menjadi mudah untuk dilakukan. Setelah noise diminimalkan, proses selanjutnya adalah deteksi tepi menggunakan adaptif canny, yang hasilnya dapat dilihat pada gambar di bawah ini.

**Figure 12.** Hasil dari proses adaptif canny

Pada [Figure 12](#) telah diketahui koordinat jawaban dari setiap soal, dan koordinat tersebut dapat menjadi petunjuk untuk mencari jawaban mana yang sudah dipilih oleh siswa. Namun terlebih dahulu kita harus mengetahui koordinat setiap opsi jawaban (A B C D E), proses penentuan koordinat opsi jawaban ditentukan dengan cara eksperimen dan identifikasi dari format lembar jawaban di awal penelitian sehingga diperoleh parameter koordinat dari masing masing opsi jawaban, dan nomor soal. Setelah itu proses pendeteksian dapat dilakukan dengan nested loop pixel, dimulai dari perulangan untuk setiap opsi, setiap nomor, dan setiap nomor kolom. Setelah dilakukan pengecekan dengan algoritma yang diusulkan diperoleh hasil sebagai berikut.

**Figure 13.** Hasil pengecekan pada sample lembar jawaban ujian.

[Figure 13](#) memperlihatkan bahwa sistem yang dibuat mampu melakukan pengecekan pada sample lembar jawaban yang diuji. Selain itu dilakukan pengujian dengan kondisi khusus pada beberapa nomor soal seperti [Figure 14](#).

**Figure 14.** Foto lembar jawaban kedua dengan kondisi khusus



Dapat dilihat pada [Figure 14](#) terdapat beberapa kondisi khusus yaitu pada 22, 26, 27, 28, 29 dan 30. Kondisi khusus tersebut diantaranya, tidak dijawab, terdapat 2 jawaban, dijawab namun dengan bentuk lain seperti disilang, dicoret, lingkaran tidak penuh, serta bekas penghapus yang dapat dilihat pada gambar [Figure 14](#). Berikut ini hasil uji coba dengan sample dari [Figure 14](#)

**Figure 15.** Hasil dari proses filtering lembar jawaban kedua

**Figure 16.** Hasil dari proses adaptif canny lembar jawaban kedua

[Figure 15](#) dan [Figure 16](#) secara berturut turut merupakan hasil dari proses filtering dan adaptif canny. Dapat dilihat pada nomor yang terdapat kasus khusus yaitu

1. Jawaban nomor 22 tidak dijawab, maka pada gambar 13 nomor 22 tidak terdapat lingkaran yang terdeteksi.
2. Jawaban nomor 26 terdapat dua jawaban maka pada gambar 13 terdapat dua lingkaran yang terdeteksi, pada kasus ini jawaban yang tersimpan adalah jawaban yang terletak paling kiri.
3. Jawaban nomor 27, 28 dan 30 terlihat bahwa soal tersebut tidak dijawab dengan lingkaran utuh, maka pada Gambar 13 tidak ada lingkaran yang terdeteksi.
4. Jawaban nomor 29 terdapat sisa jawaban yang dihapus dan mengganti jawaban, maka pada gambar 13 terdapat satu lingkaran yang terdeteksi.

Selain itu [Figure 17](#) di bawah ini merupakan hasil tracking jawaban dengan uji coba lembar jawaban pada [Figure 14](#).

**Figure 17.** Hasil dari proses tracking lembar jawaban kedua

Dapat dilihat pada [Figure 14](#) hasil tracking untuk lembar jawaban yang terdapat pada [Figure 11](#) sesuai, dengan untuk semua nomor, baik jawaban dengan kondisi umum maupun kasus khusus yang telah disebutkan di atas.

## Kesimpulan

Deteksi dengan metode nested loop pixel pada penelitian ini dilakukan pada citra biner dengan cara melakukan deteksi tepi canny adaptif terlebih dahulu. Selain itu, Metode nested loop pixel yang digunakan pada penelitian ini dengan objek lembar jawaban, dapat berjalan dengan baik. Hal ini dapat dilihat pada uji coba 1 dan 2 yang dilakukan di atas, berturut-turut melambangkan kasus normal dan kasus khusus. Pada beberapa kasus khusus yang disebutkan sebelumnya, metode ini tetap mampu untuk mendeteksi jawabannya yang dipilih oleh siswa tersebut.

## References

1. Akbar, Ronny Makhfuddin dan Endang Setyati. 2015. "Penilaian Otomatis Lembar Jawaban Komputer (LJK) Secara Real Time Dengan Memanfaatkan Webcam". Seminar Nasional "Inovasi dalam Desain dan Teknologi" - IdeaTech. 334-341
2. Derisma. 2016. "Perbandingan Kinerja Metode Deteksi Tepi pada Pengenalan Objek Menggunakan OpenCV". Jurnal Informatika Mulawarman. Vol. 11, 17-21
3. Kirana, Mira Chandra, dkk. 2017. "Penerapan Metode Canny dalam Koreksi Lembar Jawaban Komputer Untuk Try Out". Prosiding SENTIA, Vol. 9 Malang, 9-14.
4. Rafael, C., Gonzales, Ricard, E., Woods. 2002. "Digital Image Processing". Tom Robbins Publisher. United States of America.
5. Sitorus, Syahriol, dkk. 2006. Pengolahan Citra Digital. Medan : USU Press.
6. Solomon, C., Breckon, T. 2013. "Fundamental of Digital Image Processing". John Wiley and Sons, Ltd. Chichester, UK.
7. Zhou, Ping, dkk. 2011. "An Improved Canny Algorithm for Edge Detection". Journal of Computational Information System. Vol. 75 (5). Hal 1516-1523.