

## *Simple Handwriting Calculator Application Using Optical Character Recognition (OCR)*

### **Aplikasi Kalkulator Tulisan Tangan Sederhana Menggunakan Optical Character Recognition (OCR)**

**Supriadi**

Dosen Program Studi Rekayasa Perangkat Lunak,  
Universitas Nurul Jadid

---

#### **INFORMASI ARTIKEL**

##### **Riwayat Artikel:**

Dikirimkan 01 Jun 2019,  
Direvisi 01 Juli 2019,  
Diterima 01 Agustus 2019.

---

##### **Kata Kunci:**

Image Processing,  
Optical Character Recognition,  
Handwritten Recognition.

---

##### **Penulis Korespondensi:**

Supriadi,  
Universitas Nurul Jadid.  
Probolinggo Indonesia  
[supriadiadefa85@gmail.com](mailto:supriadiadefa85@gmail.com)

---

#### **ABSTRACT / ABSTRAK**

*The calculator is a calculation tool that is widely used in various specialized fields of business and commerce. The use of a calculator makes it easier for humans to perform calculation operations, but there are obstacles in the process of inputting numbers if you want to calculate the value of numbers on written media such as paper, whiteboards and so on. The user must first see the text on written media, then read it and remember it then type the writing on a calculator tool or application. The drawback of this method is that when the user forgets the writing on the written media, the user will see the written text and remember it again so that it takes longer to perform calculations using a calculator. The method used in this study is Optical Character Recognition, this method can recognize text contained in images or handwritten images of mathematical number operations. The results of the introduction of the text will then be carried out by arithmetic calculations to get the calculation results. From the trials on 20 handwritten images of mathematical number operations, the results obtained were 85% accuracy of extraction and accuracy of handwritten images that can be calculated and correct by 85%.*

Kalkulator adalah alat hitung yang banyak digunakan berbagai bidang khususnya bisnis dan perdagangan. Penggunaan kalkulator memudahkan manusia dalam melakukan operasi hitung, tetapi terdapat kendala dalam proses penginputan angka jika ingin menghitung nilai dari angka-angka pada media tulis seperti kertas, papan tulis dan sebagainya. Pengguna harus terlebih dahulu melihat teks pada media tulis, kemudian membacanya dan mengingatkannya lalu mengetik tulisan tersebut pada alat atau aplikasi kalkulator. Kekurangan dari cara ini adalah ketika pengguna lupa dengan tulisan pada media tulis, maka pengguna akan melihat teks tulisan dan mengingatkannya lagi sehingga membutuhkan waktu yang lebih lama dalam melakukan perhitungan menggunakan kalkulator. Metode yang digunakan pada penelitian ini adalah *Optical Character Recognition*, metode ini dapat mengenali teks yang terdapat pada gambar atau citra tulisan tangan operasi bilangan matematika. Hasil dari pengenalan teks tersebut kemudian akan dilakukan perhitungan aritmetika untuk mendapatkan hasil hitungnya. Dari uji coba terhadap 20 citra tulisan tangan operasi bilangan matematika, diperoleh hasil akurasi ekstraksi sebesar 85% dan akurasi citra tulisan tangan yang dapat dihitung dan benar sebesar 85%.

*This work is licensed under a [Creative Commons Attribution-Share Alike 4.0](https://creativecommons.org/licenses/by-sa/4.0/)*



---

#### **Sitasi Dokumen ini: (Biarkan bagian ini)**

S. Supriadi, “Simple Handwriting Calculator Application Using Optical Character Recognition (OCR),” *Buletin Ilmiah Sarjana Teknik Elektro*, vol. 2, no. 3, pp. 163–170, 2020. DOI: [10.12928/biste.v2i3.2695](https://doi.org/10.12928/biste.v2i3.2695)

## 1. PENDAHULUAN

Kalkulator merupakan alat hitung yang ditemukan oleh Blaise Pascal seorang matematikawan bangsa Prancis pada tahun 1642. Perkembangan kalkulator telah berevolusi mulai dari segi ukuran hingga fitur yang semakin canggih [1]. Penggunaan kalkulator banyak digunakan dalam berbagai bidang salah satunya bisnis atau transaksi penjualan. Kalkulator dapat memberikan hasil perhitungan yang tepat dan efisien daripada menghitung secara manual yang memerlukan waktu yang lebih lama dan masih dapat berpotensi terjadi kesalahan dalam perhitungan.

Penggunaan kalkulator memudahkan manusia dalam melakukan operasi hitung, tetapi terdapat kendala dalam proses penginputan angka jika ingin menghitung nilai dari angka-angka pada media tulis seperti kertas, papan tulis dan sebagainya. Pengguna harus terlebih dahulu melihat teks pada media tulis, kemudian membacanya dan mengingatnya lalu mengetik tulisan tersebut pada alat atau aplikasi kalkulator. Kekurangan dari cara ini adalah ketika pengguna lupa dengan tulisan pada media tulis, maka pengguna akan melihat teks tulisannya lagi dan mengingatnya lagi sehingga membutuhkan waktu yang lebih lama dalam melakukan perhitungan menggunakan kalkulator [2]. Oleh karena itu dibutuhkan suatu metode agar komputer dapat membaca tulisan tangan khususnya angka dan operator matematika serta dapat melakukan proses perhitungan secara otomatis tanpa harus melakukan pengetikan atau penginputan angka lagi. Untuk mengatasi permasalahan tersebut, penelitian ini akan membuat Aplikasi Kalkulator Tulisan Tangan Sederhana Dengan Menggunakan Optical Character Recognition (OCR). Beberapa metode lain juga akan diterapkan dalam penelitian ini, salah satunya dengan menggunakan metode image processing atau pengolahan citra.

Metode ini berfungsi untuk mengambil gambar atau citra tulisan tangan angka dan operator matematika yang kemudian mengubahnya menjadi informasi yang bisa diproses. Informasi yang didapat dari citra tulisan tangan tersebut kemudian akan diterjemahkan oleh OCR untuk mendapatkan teks angka dan operator matematika yang kemudian dilakukan operasi aritmatika agar mendapatkan hasil perhitungan [3]

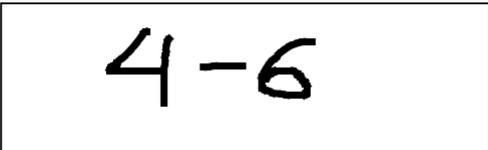
Tujuan penelitian ini adalah untuk membuat aplikasi kalkulator tulisan tangan sederhana menggunakan Optical Character Recognition (OCR) dengan menambahkan metode segmentasi proyeksi histogram. Dengan adanya penelitian ini, diharapkan dapat menerapkan metode Image Processing dan Optical Character Recognition untuk menerjemahkan dan menghitung citra atau gambar tulisan tangan yang mengandung operasi bilangan matematika dengan akurat [4].

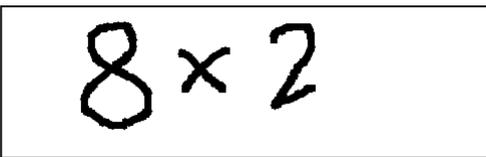
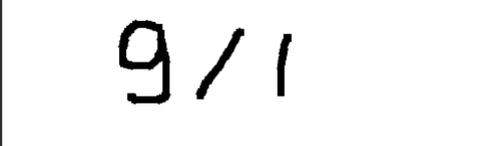
## 2. METODE PENELITIAN

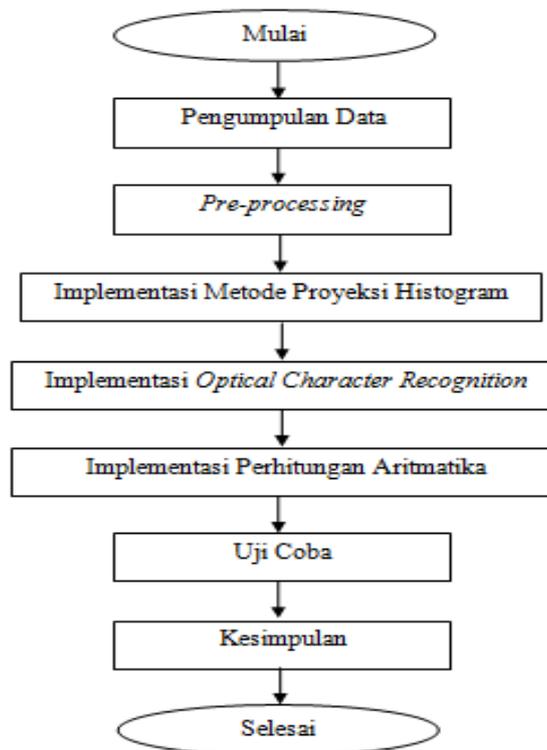
Metodologi yang digunakan dalam penelitian ini meliputi pengumpulan data citra tulisan tangan operasi bilangan matematika, pre-processing [5] (Gambar yang telah diambil dari canvas atau media tulis kemudian akan dilakukan tahap *pre-processing* dengan mengubah citra ke *grayscale*), segmentasi citra menggunakan metode proyeksi histogram, implementasi optical character recognition, implementasi perhitungan aritmetika, uji coba, serta kesimpulan dari uji coba yang telah dilakukan [6]. Adapun kerangka penelitian dapat dilihat pada Gambar 1.

Pada penelitian ini *dataset* yang digunakan adalah citra tulisan tangan operasi bilangan matematika 20 citra yang terdiri dari 5 citra tulisan tangan operasi bilangan penambahan, 5 citra tulisan tangan operasi bilangan pengurangan, 5 citra tulisan tangan operasi bilangan perkalian dan 5 citra tulisan tangan operasi bilangan pembagian. Contoh *dataset* yang digunakan dapat dilihat pada Tabel 1.

Tabel 1. Contoh *Dataset*

No	Nama Citra	Keterangan	Citra
1	penambahan1.png	Operasi bilangan penjumlahan	
2	pengurangan1.png	Operasi bilangan pengurangan	

3	perkalian1.png	Operasi bilangan pengurangan	
4	pembagian1.png	Operasi bilangan pembagian	

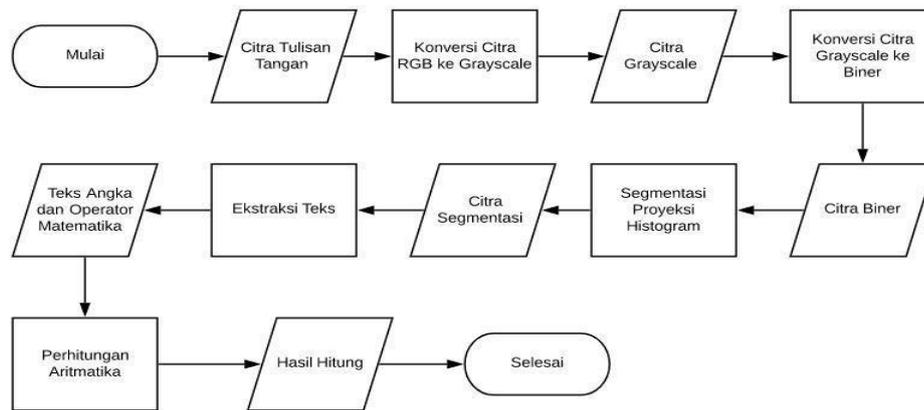


Gambar 1. Motor DC

Pengujian dilakukan pada citra secara individu, dengan warna background canvas dan warna tulisan yang sama baik untuk citra tulisan tangan penambahan, pengurangan, perkalian dan pembagian.

### 3. HASIL DAN PEMBAHASAN

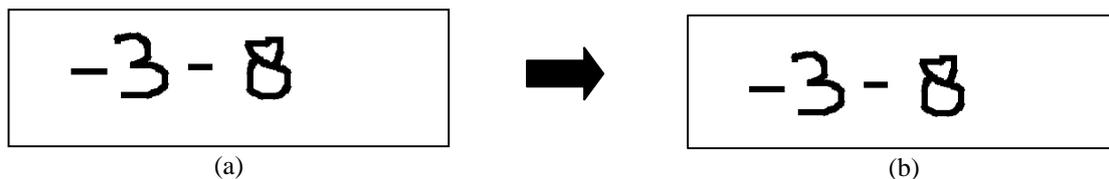
Gambar 2 adalah diagram alir sistem kalkulator tulisan tangan. Pada bagian ini ditampilkan hasil uji coba alir sistem yang telah dilakukan mulai dari tahap *pre-processing*, segmentasi citra dengan menggunakan proyeksi histogram, implementasi OCR, dan perhitungan aritmatika hingga tahap uji coba. Cara kerja dari sistem ini adalah dengan menulis angka dan operator matematika pada media tulis, kemudian hasil dari tulisan tangan tersebut akan disimpan ke dalam bentuk gambar dan kemudian menjadi masukkan untuk sistem yang selanjutnya diproses sesuai tahapan- tahapan pada Gambar 2.



Gambar 2. Diagram Alir Sistem Kalkulator Tulisan Tangan

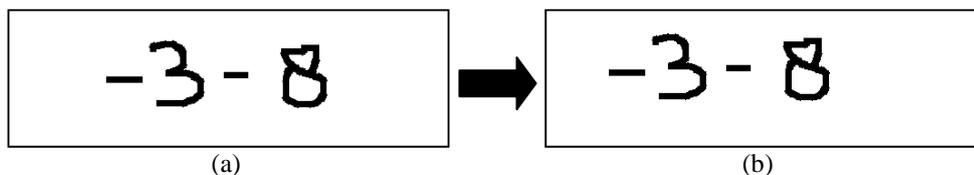
#### A. *Pre-processing* citra

Gambar 3 adalah proses konversi citra asli Menjadi *Grayscale*. Pada tahap ini, citra tulisan tangan akan diproses menjadi citra *grayscale* yang bertujuan untuk memudahkan proses pengolahan citra, pada tahap *pre-processing* ini, dataset citra tulisan tangan yang berjumlah 20 gambar diubah menjadi gambar berskala keabuan. Adapun proses konversi citra tulisan tangan menjadi citra *grayscale* dapat dilihat pada Gambar 3 [7].



Gambar 3. Konversi Citra Asli (a) menjadi citra *Grayscale* (b)

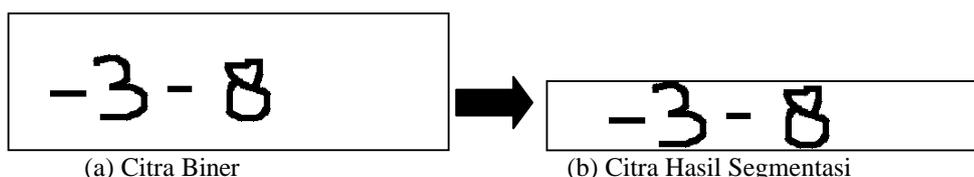
Gambar 4 adalah konversi Citra *Grayscale* Menjadi Citra Biner Setelah citra *grayscale* didapatkan. Langkah selanjutnya yaitu dengan mengubah citra *grayscale* menjadi citra biner yang hanya mempunyai nilai intensitas 0 dan 1. Cara mengubah citra *grayscale* menjadi citra biner dapat dilakukan dengan menggunakan metode *thresholding*. Adapun proses *thresholding* dapat dilihat pada Gambar 4.



Gambar 4. Konversi Citra *Grayscale* (a) menjadi Citra Biner (b)

#### B. Segmentasi Citra Menggunakan Proyeksi Histogram

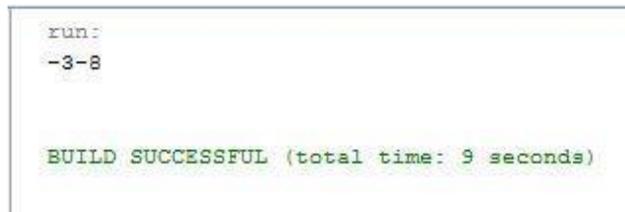
Proses segmentasi ditunjukkan pada Gambar 5. Hasil Proyeksi Histogram adalah menjelaskan Langkah yaitu melakukan segmentasi yang bertujuan untuk mengambil wilayah yang merupakan objek teks pada citra. Pada penelitian ini metode yang digunakan yaitu Proyeksi Histogram Horizontal untuk mendapatkan baris teks. Hasil dari segmentasi proyeksi histogram ditunjukkan pada Gambar 5 [8].



Gambar 5. Hasil Proyeksi Histogram

### C. Implementasi *Optical Character Recognition (OCR)*

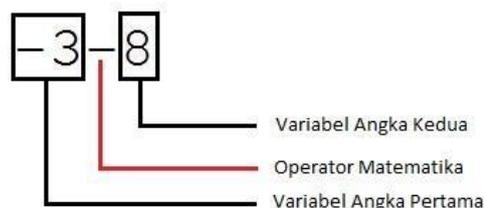
Pada tahap ini dilakukan ekstraksi teks dari citra hasil segmentasi yang telah dilakukan. Sebelum melakukan ekstraksi teks, langkah yang harus dilakukan yaitu dengan mengimport *library* Tess4j yang berfungsi untuk mengenali teks pada citra. Kemudian mengatur lokasi tessdata yang merupakan tempat *database* pengenalan karakter pada Tesseract [9]. Pada penelitian ini output yang dikeluarkan adalah teks angka dan operator matematika, sehingga dilakukan pengaturan teks yang diinginkan dengan menggunakan fungsi *whitelist* dan pencocokkan karakter agar hasil *output* seperti yang diharapkan. Hasil *output* dari ekstraksi teks citra tulisan tangan operasi bilangan matematika dapat dilihat pada Gambar 6.



Gambar 6. Hasil Ekstraksi Teks

### D. Perhitungan Aritmetika

Pada Gambar 7 akan menjelaskan hasil ekstraksi teks didapatkan, terakhir dilakukan perhitungan aritmetika dengan melakukan operasi hitung antara variabel angka pertama dan variabel angka kedua dengan operator matematika yang terdapat pada citra tulisan tangan.



Gambar 7. Proses Perhitungan Aritmatika

Pada contoh hasil *output* ekstraksi teks dari sebuah data set yang terdapat pada Gambar 8. Algoritme perhitungan aritmetika dimulai dengan cara mencari variabel angka pertama dan variabel angka kedua dari teks bilangan operasi matematika. Kemudian mencari operator matematika yang terdapat pada teks, Untuk bilangan negatif, jika terdapat tanda - (minus) diawal angka maka angka tersebut akan diubah menjadi bilangan negatif dan digabungkan menjadi 1 variabel. Jika operator ditemukan, maka akan dilakukan perhitungan aritmetika antara variabel angka pertama dan variabel angka kedua. Adapun implementasi dari perhitungan aritmetika pada Gambar 8 dapat dilihat pada persamaan (1).

Diketahui : Hasil ekstraksi teks yaitu -8-3

Variabel angka pertama= -8

Variabel angka kedua= 3

Operator matematika= -

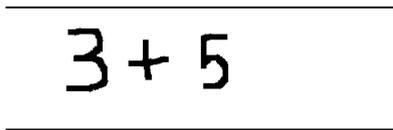
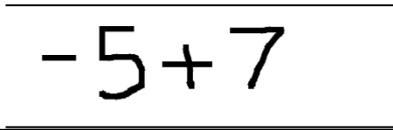
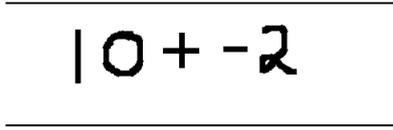
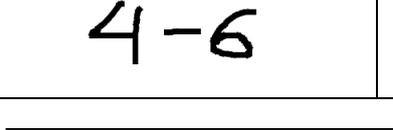
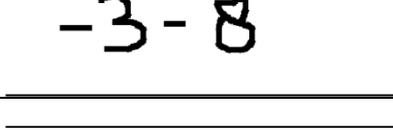
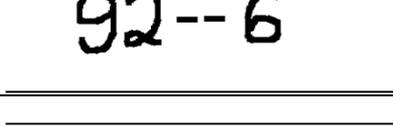
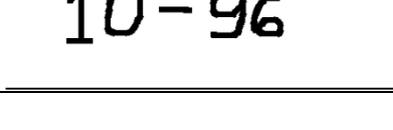
Sehingga perhitungan aritmetikanya adalah

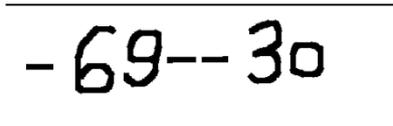
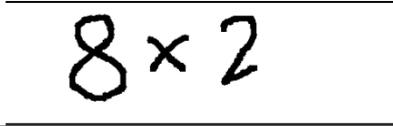
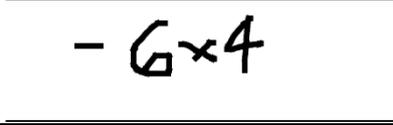
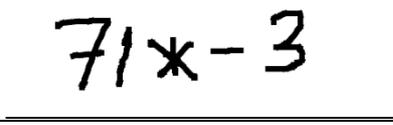
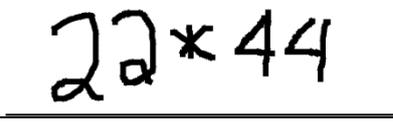
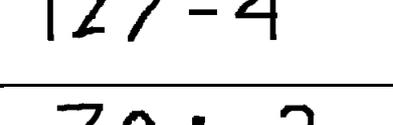
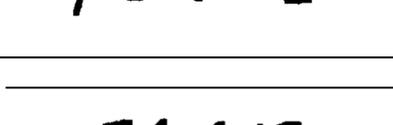
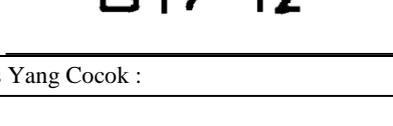
$$\begin{aligned} \text{Hasil Hitung} &= \text{Variabel angka pertama} - \text{Variabel angka kedua} & (1) \\ &= -8 - 3 \\ &= -11 \end{aligned}$$

Uji coba hasil ekstraksi teks dilakukan terhadap citra tulisan tangan operasi bilangan matematika. Jumlah dataset yang digunakan yaitu sebanyak 20 citra yang terdiri dari 5 citra tulisan tangan operasi bilangan penambahan, 5 citra tulisan tangan operasi bilangan pengurangan, 5 citra tulisan tangan perkalian dan 5 citra tulisan tangan operasi bilangan pembagian [10]. Tabel 2 merupakan hasil ekstraksi teks dari citra tulisan tangan operasi bilangan matematika.

Setelah pengujian ekstraksi teks terhadap citra tulisan tangan selesai dilakukan, pengujian berikutnya yaitu untuk mengetahui hasil perhitungan aritmetika dari teks angka dan huruf dari hasil ekstraksi teks [12]. Indikator penilaian dari uji coba perhitungan aritmetika yaitu hasil perhitungan sesuai dengan tulisan tangan yang dibuat. Adapun hasil uji coba perhitungan aritmetika dari hasil ekstraksi teks yang dilakukan dapat dilihat pada Tabel 2.

Tabel 2. Hasil Ekstraksi Teks Citra Tulisan Tangan Operasi Bilangan Matematika

No	Nama Citra	Citra Asli	Output Teks	Pencocokan
1	penambahan1.png		3+5	Cocok
2	penambahan2.png		-5+7	Cocok
3	penambahan3.png		10+-2	Cocok
4	penambahan4.png		30+64	Cocok
5	penambahan5.png		-80+55	Cocok
6	pengurangan1.png		4-6	Cocok
7	pengurangan2.png		-3-8	Cocok
8	pengurangan3.png		92--6	Cocok
9	pengurangan4.png		10-96	Cocok

10	pengurangan5.png		-69—30	Cocok
11	perkalian1.png		8*2	Cocok
12	perkalian2.png		-6*4	Cocok
13	perkalian3.png		71*-3	Cocok
14	perkalian4.png		29*44	Tidak Cocok
15	perkalian5.png		-10*-20	Tidak Cocok
16	pembagian1.png		9/1	Cocok
17	pembagian2.png		-6/3	Cocok
18	pembagian3.png		12/-4	Cocok
19	pembagian4.png		70-2	Tidak Cocok
20	pembagian5.png		-84/-12	Cocok
Total Ekstraksi Teks Yang Cocok :				17

Dari tabel hasil ekstraksi teks pada Tabel 2, terdapat beberapa hasil ekstraksi teks yang tidak cocok dengan citra tulisan tangan. Hal itu disebabkan karena Tesseract OCR mengklasifikasikan karakter tersebut menjadi karakter lain yang tidak sesuai dengan harapan seperti angka 2 yang terklasifikasi menjadi angka 9 pada gambar

perkalian4 [11].png dan juga dapat terklasifikasi menjadi angka 1 pada gambar perkalian5.png. Penyebab lainnya yaitu terdapat karakter yang tidak terdeteksi seperti yang ditunjukkan pada gambar pembagian4.png, Dimana Tesseract OCR tidak mampu mengenali karakter : (titik dua) yang merupakan simbol pembagian akan tetapi dapat membaca simbol pembagian yang lain nya yaitu / (garis miring). Untuk mengetahui akurasi dari ekstraksi teks dapat dilihat pada persamaan berikut.

$$\begin{aligned} \text{akuterasi} &= \frac{\text{jumlah hasil ekstraksi teks cocok}}{\text{jumlah total gambar}} 100\% \\ &= \frac{17}{20} 100\% \\ \text{akutasi} &= 85\% \end{aligned}$$

#### 4. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan, dapat disimpulkan bahwa Aplikasi kalkulator tulisan tangan sederhana mampu melakukan perhitungan aritmetika dari citra tulisan tangan operasi bilangan matematika. Tesseract OCR mampu melakukan ekstraksi teks pada citra tulisan tangan. Untuk menghasilkan output berupa karakter angka dan operator matematika dapat dilakukan dengan cara mendaftarkan karakter angka dan operator pada daftar whitelist. Dari uji coba terhadap 20 citra tulisan tangan operasi bilangan matematika diperoleh hasil akurasi ekstraksi teks sebesar 85% dan akurasi citra tulisan tangan yang dapat dihitung dan benar sebesar 85%

#### REFERENSI

- [1] L. Ratnasari dan A. G. Abbasi, "Perancangan Aplikasi Kalkulator Penilaian Kategorisasi Data Berbasis Android," *JURNAL ILMIAH INFORMATIKA KOMPUTER*, vol. 2, no. 23, 2018.
- [2] M. Maskuri, "Implementasi Metode Template Matching Untuk Media Pembelajaran Pengenalan Karakter Huruf Alfabeta," *Simki-Techsain*, p. 11, 2017.
- [3] A. Mahmudi, "Aplikasi MATLAB untuk Mengenali Karakter Tulisan Tangan," *MATICS*, vol. 1, no. 9, p. 18, 2017.
- [4] F. Liantoni, "Pengenalan karakter angka menggunakan metode Integral Proyeksi," *Jurnal Ilmiah Teknologi Sistem*, p. 57, 2017.
- [5] M. Rohmah, "Perancangan Aplikasi Game Aritmatika Dasar Berbasis Android," *J. Saintek*, vol. 13, no. 1, pp. 36-43, 2016.
- [6] F. Maria, "Segmentasi Citra Digital Bentuk Daun Pada Tanaman Di Politan Samarinda Menggunakan Metode Thresholding," *Jurti*, vol. 2, no. 1, p. 37-46, 2018.
- [7] H. Masrani, "Aplikasi Pengenalan Pola Pada Huruf Tulisan Tangan Menggunakan Jaringan Saraf Tiruan Dengan Metode Ekstraksi Fitur Geometri," *Sistem Komputer Untan*, vol. 02, no. 06, p. 69-78, 2018.
- [8] A. Septiarini, "Segmentasi Karakter Menggunakan Profil Proyeksi," *Jurnal Informatika Mulawarman*, vol. 2, no. 7, pp. 66-69, 2012.
- [9] A. Setiawan, "Implementasi Optical Character Recognition (OCR) pada Mesin Penerjemah Bahasa Indonesia ke Bahasa Inggris," *urnal Sistem Dan Teknologi Informasi (JUSTIN)*, vol. 2, no. 5, pp. 135-141, 2017.
- [10] M. Fauzi, "Mengubah Tulisan Tangan Menjadi Text Digital Ocr (Optical Character Recognition) Dengan Menggunakan Metode Segmentasi Dan Korelasi," *Transient,Sistem*, pp. 1-5, 2014.
- [11] P. Chakraborty, "An Open Source Tesseract based Tool for Extracting Text from Images with Application in Braille Translation for the Visually Impaired," *International Journal of Computer Applications*, vol. 16, no. 68, pp. 26-32, 2013.
- [12] F. N. Fajri dan W. J. Shudiq, "Aplikasi Fire Bus" Sebagai Media Penyampaian Informasi Keberangkatan Bis Secara Realtime Menggunakan Notifikasi Berbasis Android," *Nusantara Journal of Computers and Its Applications*, vol. 4, no. 2, pp. 85-91, 2019.