



Implementasi Keras Library dan Convolutional Neural Network Pada Konversi Formulir Pendaftaran Siswa

Wahyu Andi Saputra¹, Muhammad Zidny Naf'an², Asyhar Nurrochman³

^{1 2 3}Program Studi Informatika, Institut Teknologi Telkom Purwokerto

¹andi@ittelkom-pwt.ac.id, ²zidny@ittelkom-pwt.ac.id, ³azhar_rockman@yahoo.com

Abstract

Form sheet is an instrument to collect someone's information and in most cases it is used in a registration or submission process. The challenge being faced by physical form sheet (e.g. paper) is how to convert its content into digital form. As a part of study of computer vision, Optical Character Recognition (OCR) recently utilized to identify hand-written character by learning pattern characteristics of an object. In this research, OCR is implemented to facilitate the conversion of paper-based form sheet's content to be stored properly into digital storage. In order to recognize the character's pattern, this research develops training and testing method in a Convolutional Neural Network (CNN) environment. There are 262.924 images of hand-written character sample and 29 paper-based form sheets from SDN 01 Gumilir Cilacap that implemented in this research. The form sheets also contain various sample of human-based hand-written character. From the early experiment, this research results 92% of accuracy and 23% of loss. However, as the model is implemented to the real form sheets, it obtains average accuracy value of 63%. It is caused by several factors that related to character's morphological feature. From the conducted research, it is expected that conversion of hand-written form sheets become effortless.

Keywords: *optical character recognition, form-sheet conversion, keras library, convolutional neural network*

Abstrak

Lembar formulir merupakan salah satu media dalam mengumpulkan informasi seseorang dan umum digunakan pada proses registrasi/pendaftaran. Penggunaan media ini cukup mudah, namun hambatan yang kerap dihadapi dari penggunaan lembar formulir berbentuk fisik (misal: kertas) adalah pemindahan konten ke dalam bentuk digital. *Optical Character Recognition (OCR)* merupakan salah satu segmen dalam disiplin ilmu pengolahan citra yang dapat melakukan pengenalan karakter tulisan dengan mempelajari pola karakteristik suatu objek. Pada penelitian ini, OCR diimplementasikan pada *keras library* untuk memfasilitasi konversi konten tulisan tangan yang ada di sebuah kertas formulir sehingga dapat disimpan ke dalam media penyimpanan digital. Dalam pola pengenalan karakter, dikembangkan model pelatihan dan pengujian dengan menggunakan *Convolutional Neural Network (CNN)*. Data yang digunakan pada penelitian adalah 262.924 karakter sampel tulisan tangan dalam bentuk citra dan 29 sampel kertas formulir dari SDN 01 Gumilir Cilacap. Model dengan akurasi tertinggi memperoleh nilai 92% dan *loss* sebesar 23%. Sedangkan, pada hasil pengujian menggunakan 29 formulir didapatkan nilai akurasi rata-rata sebesar 63%. Hal ini disebabkan oleh beberapa faktor yang berkaitan dengan faktor morfologi dari citra tiap karakter huruf. Dari penelitian yang dilakukan diharapkan dapat mempermudah proses dokumentasi informasi dari lembar formulir tulisan tangan ke dalam media penyimpanan digital.

Kata kunci: *pengenalan karakter, konversi lembar formulir, keras library, convolutional neural network*

© 2019 Jurnal RESTI

1. Pendahuluan

Sekolah adalah bangunan atau lembaga untuk belajar dan mengajar serta tempat menerima dan memberi pelajaran [1]. Di Indonesia, penjurangan calon peserta didik dilakukan melalui penerimaan siswa baru. Pada masa ini, sekolah berhak menyaring siapa saja siswa yang pantas untuk masuk di sekolah tersebut atau tidak. Dalam pelaksanaannya, salah satu tahapan pada proses pendaftaran siswa baru adalah pengisian biodata atau informasi diri menggunakan kertas formulir yang telah disediakan oleh pihak sekolah. Setelah itu, calon peserta didik mengisi formulir tersebut.

Data yang diperoleh dari situs Kementerian Pendidikan dan Budaya menyatakan bahwa terdapat 1.030 Sekolah Dasar (SD) yang ada di Kabupaten Cilacap; terdiri dari 976 berstatus negeri dan 54 berstatus swasta. Setelah ditelaah, informasi dari Dinas Pendidikan dan Kebudayaan Kabupaten Cilacap menyebutkan bahwa sekolah-sekolah tersebut masih menyelenggarakan sistem pendaftaran secara *offline* menggunakan lembar kertas formulir[2].

Pendaftaran menggunakan sistem *offline* memiliki tahap yang panjang dimana calon siswa akan diberikan lembar formulir, diisi dengan data diri, lalu mereka akan mengembalikan formulir ke pihak sekolah. Pihak sekolah kemudian akan merekap informasi yang ada di dalam formulir calon siswa ke dalam media penyimpanan digital (misal: *database*) secara manual satu-persatu sehingga membutuhkan waktu yang lama dan tidak efisien. Sayangnya, skema penginputan informasi secara konvensional ini masih diterapkan di sebagian besar sekolah yang ada di Indonesia. Secara tidak langsung, hal ini akan mempengaruhi kualitas layanan dari sekolah terkait seperti keterlambatan proses pendaftaran siswa baru [3].

Pada sekolah-sekolah yang terletak di kota dan memiliki akses internet, sistem pendaftaran dapat dilakukan dengan bantuan komputer. Ini disebabkan para orangtua di zaman modern telah fasih dalam menggunakan gawai dan piranti komputer terkini. Berbeda halnya dengan kondisi sekolah yang ada di daerah pinggiran. Pendaftaran calon siswa baru sulit dilakukan pendataan dengan menggunakan komputer karena para orang tua disana belum lancar dalam menggunakan media ini. Akibatnya, sekolah-sekolah di pinggiran masih menggunakan lembar kertas formulir dalam mendata identitas diri siswa meskipun hal ini tidak efektif dan tidak efisien [4].

Lambat dan tidak efisiennya metode penginputan informasi dari media kertas formulir menjadi latar belakang permasalahan. Tujuan dari penelitian ini adalah membangun sistem yang dapat mengkonversi informasi pada lembar kertas pendaftaran siswa baru ke dalam bentuk digital. Dengan mengimplementasikan *Optical Character Recognition* (OCR) untuk mengenali tulisan tangan pada lembar formulir, sistem akan dilatih

untuk dapat merubah isi dari informasi ke dalam *digital text*. Setelah itu, informasi disimpan ke dalam *database* dengan format CSV guna memudahkan proses selanjutnya, baik itu dilakukan secara *online* maupun *offline* [5]. OCR mampu mengenali dan mengidentifikasi citra tulisan berupa huruf dan angka untuk dikonversikan ke dalam bentuk *digital text*[6].

Pada sistem yang dikembangkan, pendekatan *image processing* yang dilakukan (dengan masukan-keluaran berupa citra) adalah untuk melakukan manipulasi citra agar lebih mudah dikenali pada proses pelatihan dan pengujian. Adapun proses pada pengolahan citra yang memiliki peran paling besar pada penelitian ini adalah penentuan *Region of Interest (RoI)* dari citra hasil pemindaian. RoI akan melakukan pemilihan area yang kiranya memiliki pengaruh paling signifikan dalam mengenali karakter atau rangkaian teks dari hasil tulisan tangan pada formulir [7]–[9]. Lokasi dari tiap karakter terpecah-pecah sesuai dengan *field* yang ada pada formulir. Oleh sebab itu, menjadi tantangan tersendiri pada proses RoI untuk menentukan *area of interest* pada formulir yang akan dilakukan konversi.

Dalam penelitian ini, metode *deep learning Convolutional Neural Network (CNN)* digunakan untuk melakukan klasifikasi dari model yang dilatih. CNN merupakan variasi dari *Multilayer Perceptron* yang memiliki cara kerja seperti jaringan syaraf manusia, dapat beroperasi pada data dua dimensi, dan bagian dari *Deep Neural Network* karena kedalaman jaringan yang tinggi dan banyak diaplikasikan pada data citra khususnya pada kasus klasifikasi citra [10]–[12]. Algoritme CNN dapat mengenali karakter baru yang sebelumnya tidak ada dalam *dataset* dan dapat melakukannya dengan lebih ringkas karena di dalam CNN itu sendiri sudah terkandung *Feature Extraction Layer* [13]. Dalam penelitian ini, juga digunakan *keras library* pada *python environment* agar CNN dapat dijalankan dengan tepat dan lebih ringkas. Secara umum, dari penelitian ini, diharapkan dapat memberikan pengetahuan baru mengenai penerapan OCR dan CNN dalam melakukan konversi informasi dari lembar kertas formulir ke dalam bentuk digital.

2. Metode Penelitian

2.1. Analisis dan Observasi

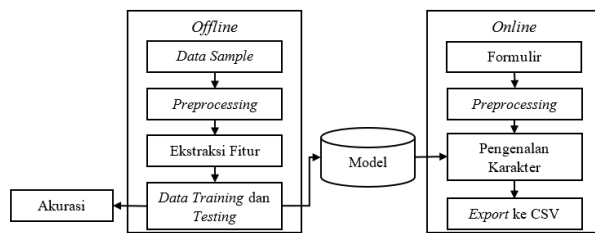
Observasi dilakukan melalui wawancara kepada para guru di SDN 01 Gumilir Cilacap untuk menanyakan alur pendaftaran dan permasalahan yang dihadapi dalam proses pendaftaran calon siswa baru. Permasalahan yang didapat adalah sistem pendaftaran calon siswa yang dilakukan secara manual atau tulis tangan; tanpa menggunakan media komputer, sehingga pihak sekolah memasukan data calon siswa dengan mengetik ulang isi formulir satu demi satu. Hal ini dirasa tidak efisien dan memakan waktu.

2.2. Pengumpulan Data

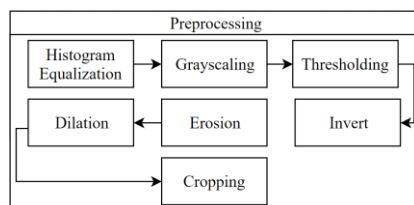
Sampel tulisan tangan didapat dari dataset pada *kaggle.com* [14] sebanyak 262.924 karakter sebagai dataset pelatihan. *Dataset* yang digunakan pada pengujian merupakan formulir pendaftaran siswa baru di SDN 01 Gumilir Cilacap sebanyak 29 sampel kertas formulir.

2.3. Pembuatan Sistem dan Penerapan Metode CNN

Sistem difasilitasi dengan Python 3. Dalam pengoperasiannya, sistem terdiri dari 2 alur operasi, yaitu operasi yang dilakukan secara *offline* dan operasi yang dilakukan secara *online*. Pada *offline*, operasi yang dilakukan adalah pembuatan model dan pelatihan *dataset* menggunakan CNN. Sedangkan, pada aktivitas *online* yaitu pembuatan konversi formulir yang akan digunakan untuk pengenalan karakter tulisan tangan pada formulir. Sistem konversi akan menjalankan beberapa tahapan proses untuk mengambil informasi yang dibutuhkan dari formulir, kemudian diterjemahkan dalam sebuah teks digital serta akan dimasukkan ke dalam format CSV. Gambar 1 menunjukkan gambaran sistem secara keseluruhan. Gambar 2 menunjukkan tahapan prapengolahan.



Gambar 1. Diagram dari sistem yang dikembangkan



Gambar 2. Tahapan *Preprocessing*

Dalam pembuatan model, dilakukan 3 kali percobaan dengan perbandingan pelatihan dan pengujian data yang beragam. Tabel 1 menunjukkan komposisi perbandingan pada penelitian ini.

Untuk menguji sistem, pada penelitian ini dibangun sebuah website berbasis Flask-Python agar lebih mudah dalam pengoperasiannya. Dalam website yang dibangun, kumpulan citra diunggah untuk dilakukan proses komputasi agar diketahui nilai akurasi pada pengenalan karakternya. Pengujian dilakukan sebanyak 1 kali yaitu pada 29 citra formulir yang telah berisi tulisan tangan.

Tabel 1. Perbandingan Data Training dan Testing

Percobaan	Pelatihan	Pengujian
1	60%	40%
2	70%	30%
3	80%	20%

2.4. Perhitungan Nilai Akurasi dengan *Sequence Matcher*

Sequence Matcher merupakan salah satu metode untuk mengetahui tingkat kemiripan antar teks yang didasarkan pada algoritma Ratcliff/Obershelp [15]. Secara sederhana, algoritma ini akan membandingkan antara banyaknya kata yang terdeteksi secara benar dengan jumlah kata yang dibandingkan secara keseluruhan.

$$D_{r,o} = (2 \times k_m) / (|S_1| + |S_2|) \quad (1)$$

dengan k_m adalah banyaknya huruf yang terdeteksi dengan benar dari kedua kata, S_1 adalah huruf asli yang seharusnya muncul dari kata, dan S_2 adalah huruf yang terdeteksi oleh sistem dari kata.

Contoh perhitungan *Sequence Matcher* dapat dilihat pada ilustrasi berikut:

	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]
S_1	J	A	K	E	T	M	U		
S_2	T	A	K	3	T	N	I	I	

Dari bagan ilustrasi di atas, dapat diketahui:

1. Nilai k_m adalah 3, yaitu 3 huruf yang terdeteksi dengan benar dari kedua kata tersebut, yaitu A, K, dan T
2. Nilai S_1 adalah 7, yaitu banyaknya huruf yang muncul pada kata yang seharusnya.
3. Nilai S_2 adalah 8, yaitu banyaknya huruf yang muncul dari hasil pendeteksian sistem.

sehingga, dari ilustrasi tersebut didapatkan nilai akurasi sebesar:

$$D_{r,o} = (2 \times 3) / (7 + 8) = 6/15 = 0,4$$

3. Hasil dan Pembahasan

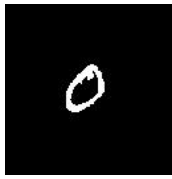
3.1. Pembuatan Model secara *offline*

Dalam pembuatan model, *dataset* diambil dari *kaggle.com* dan beberapa sampel tulisan tangan dari mahasiswa. *Dataset* ini berbentuk kumpulan citra dengan format .png yang berjumlah kurang lebih 5000 citra berdimensi 128x128 piksel pada tiap karakter. *Dataset* tersebut akan dilakukan beberapa tahap untuk dijadikan model pada sistem konversi seperti prapengolahan dan pelatihan menggunakan CNN.

3.1.1 *Preprocessing* pada dataset

Pada tahapan prapengolahan, *dataset* akan dilakukan operasi keabuan untuk dapat dilanjutkan operasi *thresholding*. Dalam operasi *thresholding* dan *invert*, keabuan yang awalnya putih dan objek yang awalnya

hitam dibalik menjadi *background* hitam dan objek putih seperti pada Gambar 3.



Gambar 3. Citra hasil *thresholding*

Operasi berikutnya adalah melakukan dilasi, dimana pada operasi ini objek akan terlihat lebih jelas dan tegas seperti ditunjukkan pada Gambar 4. Operasi dilasi dilakukan dengan menggunakan nilai kernel (2,2) dimana objek akan semakin tebal sebesar 2 piksel.



Gambar 4. Citra hasil dilasi

Setelah dilakukan dilasi, proses akan menjalankan pencarian *Region of Interest* (ROI) untuk mencari area utama pada citra. Hal ini ditunjukkan dengan Gambar 5. Setelah mendapatkan ROI pada citra, selanjutnya area tersebut dilakukan *cropping* untuk mendapatkan objek yang dibutuhkan (Gambar 5 dan Gambar 6).



Gambar 5. ROI pada citra



Gambar 6. Hasil *cropping* dari ROI

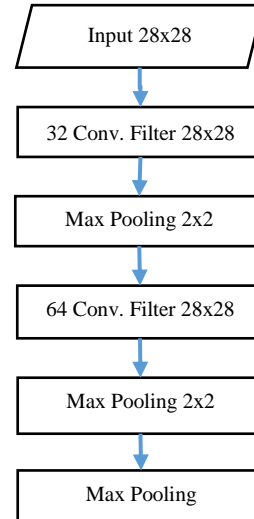
Tahap terakhir yaitu melakukan operasi *resize* untuk mengubah ukuran ROI dari 128x128 menuju 28x28 piksel. Pemilihan 28x28 piksel ini untuk mempercepat waktu pelatihan karena dimensi yang lebih kecil namun masih memiliki detail yang baik pada citra tersebut. Gambar 7 menunjukkan hasil *resizing* dari ROI.



Gambar 7. Hasil *resizing* dari ROI

3.1.2 Training dan Testing Data dengan CNN

Dataset yang telah dilakukan prapengolahan berjumlah 262.924 citra, selanjutnya *dataset* tersebut dilakukan pelatihan dan pengujian untuk dijadikan model prediksi dan mendapatkan nilai akurasi awal. Pada proses *training* dan *testing* dilakukan 3 kali percobaan dengan komposisi berbeda. Secara umum, proses pelatihan dan pengujian dilakukan dengan skema Gambar 8.



Gambar 8. Arsitektur CNN pada penelitian

Untuk melakukan pelatihan dengan metode CNN, penelitian ini difasilitasi dengan Google Colab yang dapat mengeksekusi program dengan Bahasa pemrograman *python* seperti *jupyter notebook*. Selain itu, Google Colab juga dapat mengeksekusi program dengan lebih cepat. Adapun spesifikasi CNN dari proses komputasi yang dilakukan adalah 1 *layer input*, 2 *hidden layer*, dan 1 *output layer*. Hasil pembuatan model disajikan pada Tabel 2.

Tabel 2. Hasil Pembuatan Model pada Data Training dan Testing

Percobaan	Pelatihan	Pengujian	Akurasi	Loss
1	60%	40%	91%	24%
2	70%	30%	92%	23%
3	80%	20%	91%	24%

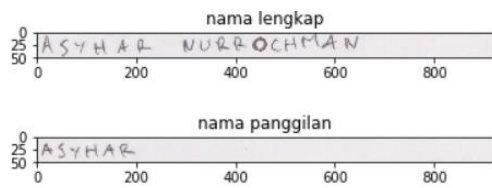
Dari ketiga percobaan pada Tabel 2, percobaan yang memiliki akurasi tertinggi dan *loss* terendah adalah percobaan kedua, yaitu dengan jumlah *data training* 70% dan *data testing* 30%. Oleh karena itu, pada penelitian ini digunakanlah komposisi model kedua.

3.2 OCR untuk mengkonversi lembar formulir ke dalam digital teks secara online

3.2.1 Prapengolahan – Pemisahan Antar Kata

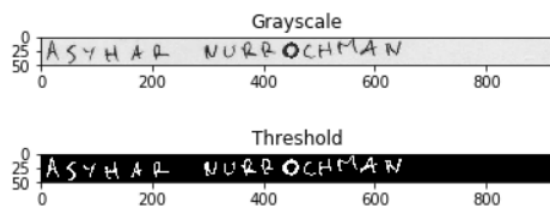
Library OpenCV digunakan dalam melakukan tahapan prapengolahan. Setelah itu, dilakukan *bulk processing* agar operasinya dapat dijalankan secara simultan. Untuk mendapatkan objek atau karakter yang diperlukan, dilakukan *cropping* secara manual dengan

menentukan koordinat pada masing-masing objek seperti Gambar 9.



Gambar 9. Hasil *cropping* pada formulir

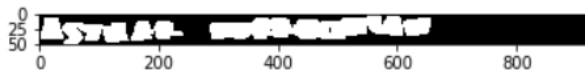
Objek tersebut kemudian dilakukan operasi *grayscale* dan *thresholding*. Operasi ini akan menghasilkan citra berskala keabuan dan biner yang akan membagi nilai derajat keabuan pada setiap piksel citra dalam 2 kelas, yaitu 0 (hitam) dan 1 (putih). Hasil dari proses ini ditunjukkan pada Gambar 10.



Gambar 10. Hasil dari *Grayscale* dan *Thresholding*

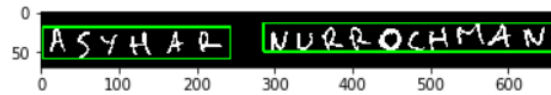
Untuk mengatasi derau, dilakukan operasi morfologi erosi di mana proses ini akan menipiskan atau mengecilkan objek derau. Nilai erosi yang digunakan adalah kernel (2, 2) dimana nilai kernel ini akan mengeliminasi 2 piksel pada sisi objek terhadap sumbu x dan y. Imbasnya, selain menghilangkan derau, operasi ini juga dapat memperkecil citra aslinya.

Untuk mengatasi masalah ukuran citra yang berkurang, dilakukan proses dilasi yaitu menebalkan objek pada citra dengan menambahkan piksel pada objek sesuai nilai piksel yang diberikan. Untuk proses dilasi dilakukan 2 tahap, dimana untuk tahap pertama menggunakan nilai kernel (5,25), yakni 5 piksel pada sumbu y dan 25 piksel pada sumbu x. Hal ini berfungsi untuk menggabungkan tiap karakter pada setiap kata (Gambar 11) sehingga proses ROI yang dilakukan dapat memisahkan antarkata dengan lebih sempurna.

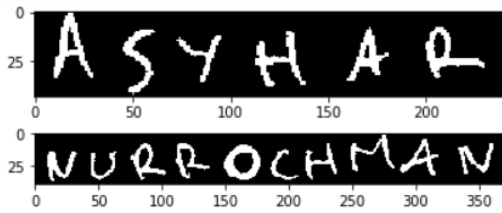


Gambar 11. Hasil dari dilasi tahap pertama

Proses selanjutnya yaitu pendeteksian ROI untuk memisahkan objek yang akan diolah dengan latar citra. Proses ini dilakukan sesuai dengan skema pada 3.1.1. Ketika ROI telah terdeteksi, proses akan dilanjutkan dengan *cropping* pada citra hasil *thresholding*. Untuk menghindari derau ikut terdeteksi maka dilakukan proses erosi agar derau bisa hilang atau tidak terdeteksi. Proses *bounding box* dan *cropping* tiap objek ditunjukkan pada Gambar 12 dan Gambar 13.



Gambar 12. Hasil *Bounding Box* dari ROI



Gambar 13. Hasil *cropping* dari *Bounding Box*

Citra pada Gambar 16 nantinya akan dilakukan dilasi kembali namun dengan nilai kernel (2, 2); yakni 2 piksel pada sumbu y dan 2 piksel pada sumbu x. Hal ini bertujuan untuk menyambungkan karakter yang tidak terhubung seperti huruf "i" dan "j" agar terdeteksi sebagai sebuah karakter yang utuh.

3.2.2 Preprocessing – Pemisahan Antar Karakter

Dalam memisahkan antar karakter pada kata, dilakukan proses pendeteksian ROI dan *cropping* yang sama dari tahap sebelumnya, namun digunakan fungsi *if w<40 and h>15* dimana objek yang memiliki lebar kurang dari 40 piksel dan tinggi lebih dari 15 piksel akan diseleksi (Gambar 14). Fungsi ini digunakan untuk membuang objek yang tidak dibutuhkan seperti pilihan pada jenis kelamin, warga negara, tempat tinggal dan lainnya pada formulir.



Gambar 14. Hasil *cropping* tanpa pelabelan

3.2.3 Pengenalan Karakter

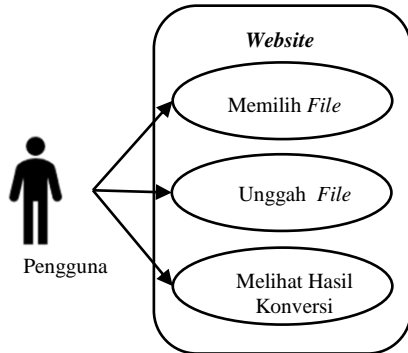
Pada tahap pengenalan karakter, citra yang telah dilakukan *cropping* akan dilakukan proses deteksi dan prediksi karakter sesuai dengan dataset pada proses pelatihan. Untuk mendeteksi karakter ini digunakan model *Hierarchical Data Format 5* (.hdf5) dimana model ini akan mendeteksi citra karakter satu persatu berdasarkan probabilitas yang muncul dari klasifikasi pada pembuatan model sebelumnya [16], [17]. Contoh hasil dari deteksi dapat dilihat pada Gambar 15 yang kemudian hasil dari pengenalan karakter disimpan dalam format CSV untuk memudahkan dalam pengolahan pada proses selanjutnya.



Gambar 15. Hasil *Cropping* dengan pelabelan

3.3 Implementasi Sistem

Pada tahap ini, model yang telah ditentukan dan sistem yang telah dirancang akan diimplementasikan ke dalam bentuk situs aplikasi. Dalam pembuatan aplikasi berbasis *web*, digunakan bahasa pemrograman python dengan bantuan Microframework Flask. Fitur yang dibuat dalam sistem ini adalah mengkonversikan formulir yang telah dipindai selanjutnya informasi yang dibutuhkan pada formulir akan disimpan dalam bentuk CSV dan akan ditampilkan pada halaman hasil. Gambar 16 menunjukkan *usecase* pada pengembangan *website*.

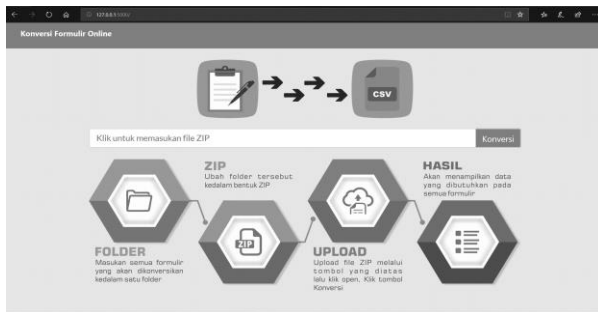


Gambar 16. Usecase Website

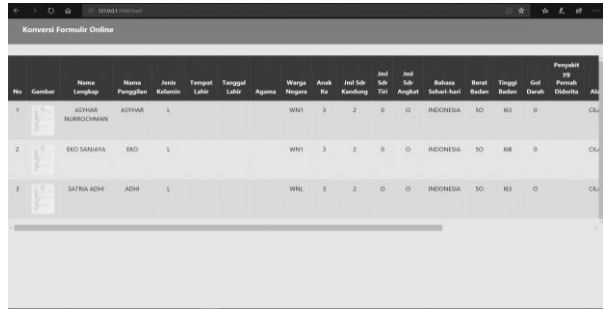
Gambar 16 menunjukkan sistem ini melibatkan satu aktor yaitu *user*, dimana *user* dapat memilih *file zip* yang akan dikonversi, kemudian *user* dapat meng-*upload* file, dan selanjutnya *user* dapat melihat hasil dari sistem konversi formulir yang isinya berupa informasi dari seluruh formulir yang sudah diunggah.

Untuk menggunakan *website* ini, file formulir dimasukkan ke dalam satu *folder* yang kemudian dikompres ke dalam bentuk ZIP. Untuk mengunggah file, user dapat menekan tombol “Pilih File” untuk memilih *file ZIP* yang telah dikompres tadi. Lalu, *file* akan diunggah dan diproses. Halaman awal dari *website* dapat dilihat pada Gambar 17.

Setelah melakukan pengunggahan file ZIP, sistem akan menampilkan tombol “Konversi” untuk menjalankan proses konversi. Hasil konversi akan ditampilkan di layar seperti Gambar 18. *Output* akan tersimpan dalam bentuk CSV.



Gambar 17. Halaman awal pada *website*



Gambar 18. Halaman hasil konversi pada *website*

3.4 Pengujian dan Evaluasi

Pada pengujian pertama, digunakan 10 lembar formulir percobaan yang dibuat sendiri dengan isi data fiktif sebagai *dummy content*. Kemudian, dilakukan proses konversi ke dalam digital guna mengetahui akurasi dari hasil percobaan; apakah hasil dari konversi sesuai dengan isi dari data pada formulir. Hasil dari percobaan dapat dilihat pada Tabel 2.

Nilai akurasi didapat dari perbandingan antara jumlah karakter (huruf) yang berhasil dideteksi benar dengan banyaknya karakter yang ada pada satu kata. Misalkan, pada kata “SISWA” banyaknya huruf yang terkandung di dalamnya adalah 5 huruf yang terdiri dari S, I, S, W, A. Tingkat akurasi akan bernilai 100% jika kelima huruf tersebut dapat dideteksi dengan benar oleh sistem. Namun, tingkat akurasi akan berkurang jika setidaknya salah satu huruf tidak terdeteksi dengan benar. Misalkan, dari kata “SISWA” ternyata terdeteksi sebagai “SISW4”, maka nilai akurasi adalah 80%. Nilai ini didapat dari 4 huruf yang terdeteksi dengan benar dibandingkan dengan 5 huruf yang ada pada kata tersebut.

Tabel 2. Contoh hasil konversi pada salah satu formulir *dummy content*

Baris ke-	Output Sistem	Kata yang Seharusnya	Akurasi
1	SYJFA REHITQ	SYIFA RENITA	78%
	AYU	AYU	
2	RAN1I2DHANI	RAMADHANI	80%
3	SY1FA	SYIFA	53%
4	PEPENIDUAH	PEREMPUAN	52%
5	C1LOCQP 20	CILACAP 30	52%
6	AUUSEUC ZOO9	AGUSTUS 2009	60%
7	J5LAM	ISLAM	86%
8	WNII	WNI	100%
9	2	2	100%
10	2	2	100%
11	IMUCNE575Z	INDONESIA	32%
12	3SE42	35 Kg	20%
13			100%
14			100%
15			100%
16	SL DARUMA	JL BARUNA	69%
	TIMUA SATU	TIMUR SATU	
	OeI286497SZ	08128644782	
	OT2RgTU2IMCr	Orang tua/	
17	UMP2AgIUCM24	menumpang/	45%
	D	asrama	

Baris ke-	Output Sistem	Kata yang Seharusnya	Akurasi
18	EDV PURMOMO O135	EDI PURNOMO /35	77%
19	EKAPUFWAMTI MII35 EKA	EKA PURMANTINI / 35	48%
20	PUFWAMTJMII 35		
21	SLTA	SLTA	100%
21	SLTA TA	SLTA	73%
22	PBg2W2INCgCI2	Pegawai Negeri	
22	SIPIIIABNIPB8S	Sipil/ ABRI/ Peg	
22	WNW2BWN	Swasta/	29%
22	PU2RWNCZ2Y2	Wiraswasta/ Petani/	
22	DHJ	Nelayan	
23			100%
24			100%
25			100%
26			100%
27			100%
28	SLSW26UUKCI2	Siswa baru kelas I/	29%
28	SIPLZGM2	Pindahan	
29	TUM2I2C2AKg2	rumah tangga/ TK	39%
29	1TK		
30			100%
31			100%
32			100%
33	S0MPO7ISPLA	SDN PORIS	60%
33	PLAWAP IU	PLAWAD 10	
34			100%
35	3	3	100%
36			100%
37			100%
38	4ZTCTMVGITOI	diterima/ ditolak	32%
38	WM		
39			100%
Rata-Rata			78.5%

Form ke	Jumlah Karakter yg Dikenali	Jumlah Karakter Sebenarnya	Akurasi
Form 9	164	242	69.01%
Form 10	232	267	75.44%
Form 11	183	328	57.55%
Form 12	209	263	72.95%
Form 13	231	295	64.49%
Form 14	206	231	64.49%
Form 15	173	267	39.13%
Form 16	215	254	59.26%
Form 17	224	311	71.31%
Form 18	318	299	42.54%
Form 19	218	270	68.47%
Form 20	356	284	78.91%
Form 21	248	246	64.62%
Form 22	229	294	56.30%
Form 23	150	284	63.07%
Form 24	172	214	63.42%
Form 25	217	296	57.82%
Form 26	202	232	61.78%
Form 27	212	244	58.34%
Form 28	214	246	74.08%
Form 29	289	355	51.58%
Rata-rata			63%

Dari Tabel 3, didapatkan nilai akurasi rata-rata sebesar 63%. Akurasi tertinggi dapat dilihat pada Formulir 20 sebesar 78,91%. Dari pengamatan dan analisis yang dilakukan, akurasi dengan nilai tinggi tersebut dihasilkan dari karakter dalam formulir yang menggunakan huruf kapital dan jarak antar karakter tidak saling berhimpitan. Hal ini mengakibatkan sistem dapat membaca karakter dengan lebih tepat oleh karena proses pemisahan antar karakter dalam formulir dapat dilakukan dengan lebih mudah oleh sistem.

Terdapat perbedaan satu karakter yang mirip yaitu 1 (satu) dan I (huruf kapital i) sehingga sistem kesulitan untuk menerjemahkannya dengan tepat. Dari sini, didapatkan nilai akurasi sebesar 80%, yakni 4 dari total 5 karakter yang benar. Meskipun sama-sama terdapat 5 karakter yang terbaca, namun nilai akurasi didasarkan pada karakter yang terbaca dengan benar, bukan hanya sebatas terbaca. Contoh lain, baris 9 dan 10 merupakan contoh yang menunjukkan baris kosong yang tidak mengandung karakter.

Untuk pengujian data utama digunakan 29 sampel formulir lembar pendaftaran siswa yang didapat dari SDN 01 Gumilir Cilacap yang telah diisi oleh orang tua calon siswa. Sampel tersebut kemudian dilakukan konversi dan didapatkan hasil akurasi yang disajikan pada Tabel 3.

Tabel 3. Hasil pengujian pada seluruh formulir

Form ke	Jumlah Karakter yg Dikenali	Jumlah Karakter Sebenarnya	Akurasi
Form 1	272	357	59.26%
Form 2	240	318	56.03%
Form 3	347	338	78.50%
Form 4	396	384	59.43%
Form 5	346	432	58.28%
Form 6	198	261	71.90%
Form 7	167	210	61.04%
Form 8	226	265	68.08%

Lebih lanjut, akurasi terendah didapatkan pada formulir 15 dengan tingkat akurasi 39,13%. Hal ini disebabkan oleh karakter dari tulisan tangan yang saling berhimpitan sehingga sistem membacanya sebagai *noise* atau derau. Sebagaimana didukung oleh penelitian dari [18], kelemahan CNN hingga kini adalah sulitnya mengenali karakter tulisan tangan yang saling berhimpitan. Penggunaan operasi pada pengolahan citra, yakni erosi dan dilasi sangat berpengaruh dalam memisahkan maupun menggabungkan antarkarakter agar terbaca oleh sistem. Operasi ini memiliki dampak yang signifikan karena secara langsung mempengaruhi morfologi dari citra. Selain alasan tersebut, formulir 15 masih mengandung huruf non-kapital dimana pada model CNN yang dilatih nilai akurasi pada huruf kecil memiliki akurasi yang minim.

4. Kesimpulan

4.1 Kesimpulan

Kombinasi antara Keras Library dan CNN dapat digunakan untuk mengkonversi lembar formulir pendaftaran siswa yang berisi tulisan tangan ke dalam bentuk teks digital. Beberapa operasi pengolahan citra yang dilakukan agar karakter dapat terbaca oleh sistem adalah konversi *grayscale*, *thresholding*, erosi, dilasi,

ROI *detection, cropping*, dan *resize*. Dari penelitian ini, digunakan model pengklasifikasian CNN dengan nilai akurasi sebesar 92%. Namun, setelah diterapkan secara riil, didapatkan nilai akurasi sebesar 63%. Beberapa faktor yang mempengaruhi keakuratan sistem dalam mengenali teks adalah karakter yang saling berhimpitan, tingkat kemiringan karakter, jenis kapital karakter, dan operasi pada *praprocessing* yang belum sempurna.

4.2 Saran

Diharapkan pada penelitian selanjutnya dapat meningkatkan akurasi sistem dalam mengkonversi lembar formulir dari tulisan tangan ke dalam bentuk teks digital. Saran untuk pengembangan sistem kedepannya adalah:

1. Memperbanyak jumlah *dataset* serta varian karakter sehingga dapat memperkaya mesin komputasi dalam mempelajari ciri khas tiap karakter.
2. Menambah variasi *praprocessing* yang dilakukan sehingga dapat memisahkan antar karakter dengan lebih sempurna seperti penggunaan ekstraksi fitur yang khusus ditujukan pada kasus pengenalan huruf.
3. Membuat formulir khusus dengan garis bantuan berbentuk kotak-kotak agar lebih mudah dalam memisahkan antarkarakter.

Daftar Rujukan

- [1] Tim Pusat Bahasa Departemen Pendidikan Nasional, *Kamus Bahasa Indonesia*, Edisi XVI. Jakarta: Pusat Bahasa, 2008.
- [2] Direktorat Jenderal Pendidikan Dasar dan Menengah, 2018. *Jumlah Data Satuan Pendidikan (Sekolah) per Kabupaten/Kota: Kabupaten Cilacap*. [Daring]. Tersedia di: http://referensi.data.kemdikbud.go.id/index21_tkra.php?kode=026600&level=2.
- [3] B. Warsita, 2015. *Evaluasi Sistem Penerimaan Peserta Didik Baru (PPDB) Online untuk Peningkatan Kualitas Pembelajaran*. Jurnal Kwangsan. 3 (1). hal. 27.
- [4] S. Maulina, *Respon Orang Tua Peserta Didik SMP atas Layanan Inormasi Penerimaan Peserta Didik Baru (PPDB) Melalui Media Online oleh Dinas Pendidikan Kota Malang*, 2013.
- [5] J. Sutresna, 2017. *Perancangan Sistem Formulir Pelayanan Kedukaan Online Menggunakan Metode Web Base Engineering Pada Pt . Abadi Cahaya Universal (Rumah Duka Abadi) Jakarta*. Jurnal Informatika Universitas Pamulang. 2 (2). hal. 108–113.
- [6] S. Hartanto, A. Sugiharto, dan S. N. Endah, 2015. *Optical Character Recognition Menggunakan Algoritma Template Matching Correlation*. Jurnal Masyarakat Informatika. 5 (9). hal. 1–12.
- [7] R. F. Falah, O. D. Nurhayati, dan K. T. Martono, 2016. *Aplikasi Pendeteksi Kualitas Daging Menggunakan Segmentasi Region of Interest Berbasis Mobile*. Jurnal Teknologi dan Sistem Komputer. 4 (2). hal. 333–343.
- [8] B. Al-Mahadeen, M. S. Altarawneh, dan I. H. Altarawneh, 2010. *Signature Region of Interest using Auto cropping*. IJCSI International Journal of Computer Science Issues. 7 (2). hal. 1–5.
- [9] H. A. Nugroho, W. A. Saputra, A. E. Permasari, dan E. E. H. Murhandarwati, 2017. *Automated determination of plasmodium region of interest on thin blood smear images*. 2017 International Seminar on Intelligent Technology and Its Application: Strengthening the Link Between University Research and Industry to Support ASEAN Energy Sector, ISITIA 2017 - Proceeding. 2017-Janua . hal. 352–355.
- [10] M. Zufar dan B. Setiyono, 2016. *Convolutional Neural Networks untuk Pengenalan Wajah Secara Real-Time*. Jurnal Sains dan Seni. 5 (3). hal. 1–6.
- [11] Sukardi, Z. Arifin, dan M. Risaldi, *Klasifikasi Penentuan Gambar Berbasis Tensorform Dan Framework Dengan Algoritma CNN*, in *Seminar Nasional APTIKOM (SEMNASTIKOM)*, 2017, hal. 1–4.
- [12] Y. Pramudana, 2015. *Pengenalan Tulisan Tangan Dengan Menggunakan Metode Diagonal Feature Extraction dan K-Nearest Neighbour*. Bandung.
- [13] J. Pujoseno, *Implementasi Deep Learning Menggunakan Convolutional Neural Network Untuk Klasifikasi Alat Tulis*, Universitas Islam Indonesia, Yogyakarta, 2018.
- [14] <https://www.kaggle.com/tejasreddy/iam-handwriting-top50>.
- [15] S. A. J. Zaidi, A. Buriro, M. Riaz, A. Mahboob, dan M. N. Riaz, 2019. *Implementation and comparison of text-based image retrieval schemes*. International Journal of Advanced Computer Science and Applications. 10 (1). hal. 611–618.
- [16] M. Jackson, J. P. Simmons, dan M. De Graef, 2010. *MXA: A customizable HDF5-based data format for multi-dimensional data sets*. Modelling and Simulation in Materials Science and Engineering. 18 (6).
- [17] R. Bhowmik, J. Hartog, dan M. Govindaraju, 2013. *Processing HDF5 datasets on multi-core architectures*. Proceedings - International Conference on Advanced Information Networking and Applications, AINA. hal. 666–673.
- [18] R. Wiles, 2019. *Have we solved the problem of handwriting recognition?* [Daring]. Tersedia di: <https://towardsdatascience.com/https-medium-com-rachelwiles-have-we-solved-the-problem-of-handwriting-recognition-712e279f373b>. [Diakses: 02-Okt-2019].