

Pemanfaatan *Robotic Process Automation* Dan *Optical Character Recognition* Dalam Otomatisasi Proses Rekrutmen Karyawan

Utilization Of Robotic Process Automation And Optical Character Recognition In Automation Of Employee Recruitment Process

1st Muhammad Raihan
Fakultas Teknik Elektro
Universitas Telkom

Bandung, Indonesia
mraihanpolos@student.telkomuniversity.ac.id

2nd Kris Sujatmoko
Fakultas Teknik Elektro
Universitas Telkom

Bandung, Indonesia
krissujatmoko@telkomuniversity.ac.id

3rd Iwan Iwut Tritoasmoro
Fakultas Teknik Elektro
Universitas Telkom

Bandung, Indonesia
iwaniwut@telkomuniversity.ac.id

Abstrak—Digitalisasi proses merupakan sebuah trend yang sedang marak dilakukan oleh sebagian besar industri saat ini. Banyak solusi instan untuk dalam implementasi digitalisasi proses, salah satunya adalah dengan memanfaatkan program *Robotic Process Automation* (RPA) dan *Optical Character Recognition* (OCR). Kedua program tersebut dapat digunakan untuk proses digitalisasi pada tahap penerimaan karyawan oleh Divisi HR. Tahapan yang akan dilalui pada sistem ini, yaitu : seorang karyawan yang telah lolos seleksi mengirim data KTP melalui email, selanjutnya diunduh menggunakan RPA. Data KTP akan diekstraksi dan dilakukan validasi. Setelah itu, data KTP disimpan ke dalam file Excel dan robot akan mensubstitusi data yang telah didapatkan kedalam template dokumen PKWT / PKWTT. Tahapan terakhir adalah mengunggah file ke dalam Google Drive. Terdapat dua parameter utama yang diuji, yaitu durasi pembuatan dokumen serta akurasi mesin OCR. Dari hasil yang

diperoleh, dapat diketahui bahwa implementasi dari RPA dan OCR dalam proses rekrutmen karyawan ini dapat memangkas durasi sebesar 48.69% dibandingkan dengan proses manual. Selain itu, tingkat akurasi yang dihasilkan oleh sebuah mesin OCR tanpa dilakukan validasi dapat mencapai 95.95%. Dengan demikian, dapat disimpulkan bahwa pemanfaatan RPA dan OCR cukup efektif untuk diimplementasikan pada proses rekrutmen karyawan.

Kata kunci—*optical character recognition* (ocr), *robotic process automation* (rpa), digitalisasi, otomatisasi, penerimaan karyawan.

Abstract—Most sectors nowadays are following the trend of process digitalization. The use of robotic process automation (RPA) and optical character recognition (OCR) software is one of the numerous quick fixes for implementing process digitalization. The HR Division can utilize both applications to digitize the hiring process at this level. The steps that will be

completed in this system are as follows: a successful applicant transmits their ID card information by email, after which it is retrieved using RPA. Data from ID cards will be retrieved and verified. The information from the ID card is then stored into an Excel file, where it is substituted by the robot into the PKWT/PKWTT document template. The file must then be uploaded to Google Drive as the last step. The length of document creation and the OCR engine's precision are the two key factors that are examined. The results show that using RPA and OCR to automate parts of the staff recruiting process may cut down on time by 48.69% when compared to a manual procedure. Additionally, an OCR machine's accuracy level without validation might reach 95.95 percent. In light of this, it can be said that implementing RPA and OCR in the hiring process is highly successful.

Keywords—*optical character recognition (ocr), robotic process automation (rpa), digitization, automation, employee recruitment.*

I. PENDAHULUAN

Perkembangan yang pesat dalam hal teknologi informasi serta terjadinya pandemi Covid-19 pada awal tahun 2020 telah menjadi katalisator setiap perusahaan untuk melakukan transformasi digital secara masif. Meskipun sarana dan prasarana yang belum memadai, namun setidaknya pandemi telah mempercepat akselerasi transformasi digital tersebut. Salah satu divisi yang tidak luput dari transformasi digital merupakan Divisi *Human Resource* (HR). Dengan diterapkannya Pemberlakuan Pembatasan Kegiatan Masyarakat (PPKM) yang membatasi mobilitas masyarakat, divisi HR terpaksa untuk melakukan tahap rekrutmen jarak jauh menggunakan *conference video call* dan pengiriman dokumen secara daring [1].

Salah satu dokumen yang diperlukan saat proses rekrutmen adalah dokumen PKWT (Perjanjian Kerja Waktu Tertentu) ataupun PKWTT (Perjanjian Kerja Waktu Tak Tertentu). Dokumen PKWT dan PKWTT merupakan sebuah dokumen perjanjian kontrak kerja yang dibuat untuk menandai dimulainya kesepakatan kerja antara pekerja/buruh dengan industri. Dokumen PKWT merupakan dokumen yang umum dibuat bagi karyawan yang dikontrak dengan durasi kerja yang telah ditentukan sedangkan

PKWTT merupakan dokumen yang dibuat bagi karyawan tanpa durasi kontrak tertentu [2].

Biasanya, seorang HR melakukan pemindahan biodata pada dokumen KTP yang diterima melalui *email* ke dokumen PKWT/PKWTT dengan cara manual. Namun, pada tahap ini setelah seorang HR membuat cukup banyak dokumen PKWT/PKWTT akan terjadi penurunan konsentrasi yang menyebabkan turunnya akurasi dalam pemindahan biodata seorang karyawan. Selain itu, untuk pemindahan biodata dalam jumlah banyak juga membutuhkan durasi yang cukup lama. Seluruh permasalahan tersebut dapat diminimalisir apabila menggunakan sistem *Optical Character Recognition* (OCR) dan *Robotic Process Automation* (RPA).

Optical Character Recognition (OCR) merupakan sebuah program yang dapat membaca sebuah dokumen dalam bentuk tulisan tangan maupun sebuah mesin pencetak seperti *printer* ataupun mesin ketik. Pada saat seperti ini, OCR merupakan solusi yang paling efektif untuk melakukan konversi dokumen berbentuk gambar menjadi dokumen berbentuk teks [3]. Empat mesin OCR yang digunakan pada penelitian ini adalah Microsoft OCR, Tesseract OCR, OmniPage OCR, dan UiPath OCR. Akan dilakukan perbandingan hasil akurasi dari keempat mesin OCR diatas. Dengan penggunaan mesin OCR yang terbaik, diharapkan tingkat akurasi dalam membaca banyak file akan meningkat tanpa terjadinya penurunan kualitas.

Pembuatan dokumen PKWT/PKWTT juga dapat dipermudah dengan menggunakan *Robotic Process Automation* (RPA). RPA merupakan sebuah robot dalam bentuk perangkat lunak yang dapat mengotomatisasi proses bisnis. RPA dapat mengurangi proses interaksi antara manusia dengan komputer dengan cara meniru interaksi yang dilakukan secara berulang-ulang. Implementasi RPA dapat mengurangi durasi kerja serta meningkatkan akurasi hingga 100% [4].

II. KAJIAN TEORI DAN METODE

A. Robotic Process Automation

RPA merupakan sebuah robot berbentuk software yang dapat mengerjakan sebuah task komputer yang dilakukan secara berulang, terstruktur dan rutin. RPA dapat memberikan manfaat yang sangat besar apabila diimplementasikan pada sebuah task yang

memiliki skala cukup besar. Implementasi RPA dapat dilakukan dengan dua jenis mode yaitu, mode *Attended* maupun *Unattended*. Perbedaan dari kedua mode tersebut adalah :

1. Attended

Merupakan sebuah mode RPA yang masih membutuhkan kehadiran manusia dalam penerapannya. Pada mode *attended* robot hanya dapat mulai berjalan sesuai dengan instruksi manusia.

2. Unattended

Mode *unattended* adalah mode RPA yang dapat bekerja tanpa membutuhkan campur tangan manusia. Pada mode ini, robot diberikan sebuah pemicu agar dapat melakukan tugasnya secara otomatis. Dalam mode ini, seluruh aktivitas dapat diselesaikan dengan sendiri oleh robot tanpa adanya batasan waktu 24/7/356. Manfaat dari adanya mode ini sangat signifikan karena dapat menyelesaikan tugas dengan cepat dan efisien tanpa campur tangan manusia [5]

B. Optical Character Recognition (OCR)

OCR merupakan program yang dapat mengkonversi sebuah gambar menjadi teks yang dapat dikenali oleh komputer. Dalam hal ini, OCR dapat mengidentifikasi sebuah karakter yang berasal tulisan tangan maupun hasil mesin pencetak. OCR bekerja dengan cara mencocokkan pola-pola dari sebuah karakter. Dalam Tugas Akhir ini digunakan empat jenis mesin OCR, yaitu :

1. UiPath OCR

UiPath OCR Merupakan sebuah mesin OCR yang dibuat oleh UiPath. Penggunaan Mesin UiPath OCR hanya terbatas pada lingkup aplikasi UiPath. Untuk dapat menjalankan UiPath OCR dibutuhkan sebuah API Key yang dapat diambil lisensinya dari akun UiPath pada UiPath Orchestrator. Terdapat dua jenis UiPath OCR yang tersedia pada aplikasi UiPath. Jenis yang pertama merupakan UiPath Document OCR yang berfungsi untuk mengidentifikasi karakter yang bersumber dari sebuah dokumen. Sedangkan, untuk jenis kedua merupakan UiPath Screen OCR yang berfungsi untuk mengidentifikasi karakter yang bersumber dari layar monitor secara *real time*.

2. Microsoft OCR

Mesin OCR ini dibuat oleh perusahaan Microsoft. Implementasi Microsoft OCR pada aplikasi terbagi menjadi tiga tipe profil,

yakni tipe profil tipe profil *screen* untuk memindai sebuah aplikasi secara langsung, tipe profil *scan* untuk memindai dokumen serta tipe profil *legacy* yang merupakan profil *default* yang telah diatur oleh UiPath. Seluruh tipe profil tersebut dapat disesuaikan dengan kebutuhan pada penggunaan OCR

3. OmniPage OCR

Salah satu mesin OCR yang dapat dijalankan pertama kali pada perangkat komputer adalah OmniPage OCR. OmniPage OCR mulai dikembangkan pada dekade 1980-an yang setelah itu dijual kepada perusahaan Caere. Pada tahun 2019 OmniPage OCR dibeli oleh perusahaan Kofax inc yang memiliki kantor pusat di California, Amerika Serikat. Metode yang digunakan pada mesin OmniPage OCR terbagi menjadi tiga tahap yakni *Image Pre-processing*, *Layout analysis and Zoning* dan metode terakhir merupakan tahap *Text Recognition*[6].

4. PyTesseract OCR

Tesseract OCR merupakan sebuah mesin OCR bersifat open source yang mulai dikembangkan antara tahun 1985 sampai 1995 oleh perusahaan Hewlett Packard. Sebagian besar pengembangan Tesseract OCR diambil alih oleh Google pada tahun 2006 [7]. Saat ini, Tesseract OCR merupakan salah satu OCR yang paling banyak digunakan. Hal tersebut disebabkan oleh akurasi yang dihasilkan dari mesin OCR ini sudah mendekati sempurna.

C. Regular Expression (Regex)

Regex merupakan sebuah struktur bahasa yang mencocokkan teks berdasarkan pola tertentu, terutama untuk situasi yang kompleks. Penggunaan Regex sangat efektif dalam proses penguraian kata (text parsing). Regex sangat mungkin untuk dimanfaatkan dalam hal pencarian, substitusi, maupun pemisahan *string* pada kasus yang sangat rumit [8]. Dengan adanya Regex, proses *string manipulation* yang cukup rumit dapat dengan mudah diselesaikan menggunakan formula-formula tertentu.

D. UiPath

UiPath merupakan sebuah aplikasi yang dikembangkan khusus untuk melakukan otomatisasi sebuah pekerjaan agar pekerjaan tersebut dapat lebih efektif dan efisien. Selain itu, penggunaan aplikasi UiPath juga sangat bermanfaat dalam proses bisnis, karena UiPath dapat terintegrasi dengan berbagai

macam aplikasi pihak ketiga. UiPath memiliki tiga jenis aplikasi yang berbeda menyesuaikan dengan fungsinya. Berikut merupakan ketiga jenis aplikasi serta yang dimilikinya :

1. UiPath Studio

UiPath Studio memiliki fungsi untuk menyusun serta menjalankan algoritma yang ingin dibuat. Tahap penyusunan program yang meliputi instalasi paket, perancangan aktivitas, perancangan algoritma, ujicoba program hingga tahap stabilisasi program yang telah jadi, semuanya dilakukan pada aplikasi ini.

2. UiPath Assistant

Perangkat lunak UiPath Assistant dapat berfungsi untuk menjalankan sebuah pekerjaan yang telah selesai disusun. Pada perangkat lunak ini, kita juga dapat menambahkan pengingat untuk menjalankan pekerjaan tertentu.

3. UiPath Orchestrator

Fungsi utama dari perangkat lunak UiPath Orchestrator adalah sebagai pengelolaan manajemen aset. Didalam UiPath Orchestrator terdapat banyak sekali panel-panel untuk mengelola seluruh aktivitas robot pekerja. Pemasangan sebuah aktivitas pemicu terhadap robot yang bersifat *attended* maupun *unattended* dilakukan pada perangkat lunak ini.

E. Levenshtein Distance

Algoritma Levenshtein Distance. Algoritma ini ditemukan oleh seorang penemu asal Rusia yakni Vladimir Losifovich Levenshtein pada tahun 1965. Penggunaan algoritma ini memiliki tujuan untuk mencari dan menghitung jumlah minimum mutasi titik yang dibutuhkan dalam mengoreksi sebuah *string* terhadap *string* yang lainnya. Mutasi titik yang dapat dihitung adalah berupa penambahan *string* (*insertion*), penukaran *string* (*substitution*) dan juga pengurangan *string* (*Deletion*) [9]. Jarak Levenshtein dapat didefinisikan pada persamaan dibawah [10].

$$D(s, t) = d(s_1, t_1) + d(s_2, t_2) \dots + d(s_i, t_i) \quad (1)$$

$$D(s, t) = \sum_{i=1}^i d(s_i, t_i) \quad (2)$$

Dimana : $s_i, t_i \in V$ untuk $i = 1, 2, \dots, l$

$$d(s_i, t_i) = 0 \text{ jika } s_i = t_i \text{ dan} \\ d(s_i, t_i) = 1 \text{ jika } s_i \neq t_i$$

S : *String* Sumber

T : *String* Target

$D(s, t)$ merupakan banyaknya operasi minimum yang dibutuhkan untuk mengoreksi dua buah *string*. Dalam implementasi dari Algoritma Levenshtein, setiap *insertion*, *deletion* maupun *substitution* dapat dilakukan secara bersamaan.

Contoh operasi dalam implementasi Levenshtein Distance adalah jika terdapat *string* sumber (S) = "Mobil" dan *string* target (T) = "Motor". Maka jumlah tersebut dapat dituliskan dengan $|s| = 5, |t| = 5$. Apabila kedua *string* tersebut dicocokkan, maka perbandingan yang didapat adalah sebagai berikut:

12345

S= Mobil

T= Motor

Dari contoh diatas, terdapat tiga buah proses penukaran *string* (*substitution*) pada karakter "t", "o" dan "r" pada indeks 3, 4 dan 5. Oleh karena itu, proses perhitungan Jarak Levenshtein yang dapat dilakukan adalah sebagai berikut :

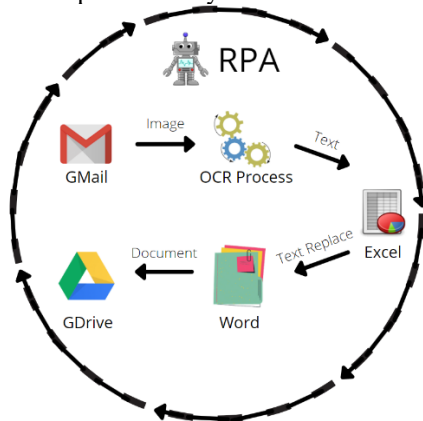
$$\begin{aligned} D(s, t) &= \sum_{i=1}^5 d(s_i, t_i) \\ &= d(M, M) + d(o, o) + d(b, t) + d(i, o) \\ &\quad + d(l, r) \\ &= 0 + 0 + 1 + 1 + 1 \\ &= 3 \end{aligned}$$

Nilai Jarak Levenshtein yang didapatkan dari perbandingan *string* diatas adalah $D(s, t) = 3$.

F. Desain Sistem

Seluruh sistem pada tugas akhir ini dilakukan dalam ruang lingkup RPA yang mana sistem tersebut terbagi menjadi enam proses utama. Keenam proses itu diawali dengan pengambilan data menggunakan *email*, ekstraksi gambar dengan OCR, penyimpanan data ke Excel, substitusi data, penyimpanan data di lokal, hingga

penyimpanan data di *Gdrive*. Proses ekstraksi OCR menggunakan empat jenis mesin OCR yakni UiPath OCR, Microsoft OCR, OmniPage OCR dan PyTesseract OCR. Untuk UiPath OCR, Microsoft OCR dan Omnipage OCR, ketiganya merupakan mesin OCR yang telah disediakan oleh UiPath dengan hanya deployment dalam penggunaannya. Sedangkan untuk mesin OCR PyTesseract merupakan mesin OCR yang perlu dikembangkan dengan menggunakan Bahasa Pemrograman Python dalam implementasinya.



GAMBAR 1
GAMBARAN BLOK DIAGRAM SISTEM.

G. Preprocessing

Pada tahap ekstraksi OCR yang dilakukan dalam Tugas Akhir ini perlu dilakukan tahap *preprocessing* terlebih dahulu. Tahap *preprocessing* merupakan pengubahan warna RGB menjadi warna *grayscale* pada gambar yang digunakan. Untuk melakukan tahap *preprocessing* dapat menggunakan *activity convert color* pada aplikasi UiPath.

H. Proses Utama

Proses utama pada Tugas Akhir ini merupakan proses ekstraksi data KTP yang dilakukan oleh mesin OCR serta substitusi data pada *template* dokumen PKWT / PKWT. Mesin OCR UiPath, Microsoft dan OmniPage merupakan mesin OCR yang telah disediakan oleh UiPath. Untuk dapat melakukan ekstraksi sebuah dokumen KTP diperlukan sebuah rancangan struktur data yang akan diambil. Struktur data yang dapat disesuaikan pada *Taxonomy Manager* dapat mencakup jenis dokumen, struktur dari dokumen, serta kolom-kolom yang terdapat pada dokumen tersebut. *Activity Taxonomy Manager* hanya dapat bekerja dengan baik jika digunakan dengan mesin OCR yang

disediakan UiPath dan menggunakan metode validasi UiPath. Sedangkan, mesin OCR PyTesseract dikembangkan menggunakan Bahasa Pemrograman Python.

Untuk proses ekstraksi data pada Mesin OCR PyTesseract dilakukan dengan menggunakan Regular Expression karena *Taxonomy Manager* tidak dapat bekerja pada mesin OCR ini. Dengan alasan tersebut, proses validasi reguler membutuhkan perhatian yang lebih serta durasi yang lebih lama dibanding mesin OCR lainnya. Proses validasi ini perlu disesuaikan dengan format yang ada pada dokumen KTP secara manual oleh pengguna.

I. Parameter Performansi

Tahap terakhir pada Tugas Akhir ini adalah mendapatkan parameter performansi. Terdapat dua buah parameter yang diambil pada Tugas Akhir ini. Kedua parameter tersebut adalah :

1. Performansi RPA

Performansi RPA merupakan sebuah parameter yang digunakan untuk menghitung seberapa efektif RPA dalam menghemat durasi waktu yang diperlukan. Parameter Performansi RPA dapat didefinisikan dengan persamaan sebagai berikut :

$$P = \frac{TM - TA}{TM} \times 100\% \quad (1)$$

Keterangan :

P = Performansi RPA.

TM = Durasi pembuatan dokumen secara manual.

TA = Durasi pembuatan dokumen dengan robot.

2. Performansi OCR

Performansi OCR merupakan parameter yang diperlukan untuk menghitung tingkat keberhasilan atau akurasi dari sebuah mesin OCR dalam mengekstraksi suatu kata. Dalam implementasinya, pengukuran parameter ini juga membutuhkan Algoritma *Levenshtein Distance* agar hasil yang didapat lebih valid dan akurat. Parameter ini dapat didefinisikan oleh persamaan sebagai berikut :

$$A = \frac{T - LD}{T} \times 100\% \quad (2)$$

Keterangan :

Akurasi = Akurasi OCR.

T = Jumlah karakter

sebenarnya.

LD = Nilai Levenshtein

Distance.

III. HASIL DAN PEMBAHASAN

Skenario yang dilakukan untuk membuat analisa pada Tugas Akhir adalah durasi pembuatan Dokumen PKWT / PKWTT dengan RPA, manual, durasi validasi, serta performansi dari empat jenis mesin OCR yang digunakan. Parameter pada uji performansi masing-masing OCR akan

dapat menentukan jenis mesin OCR yang dapat memberikan performansi paling baik.

A. Pengujian Pembuatan Dokumen PKWT / PKWTT

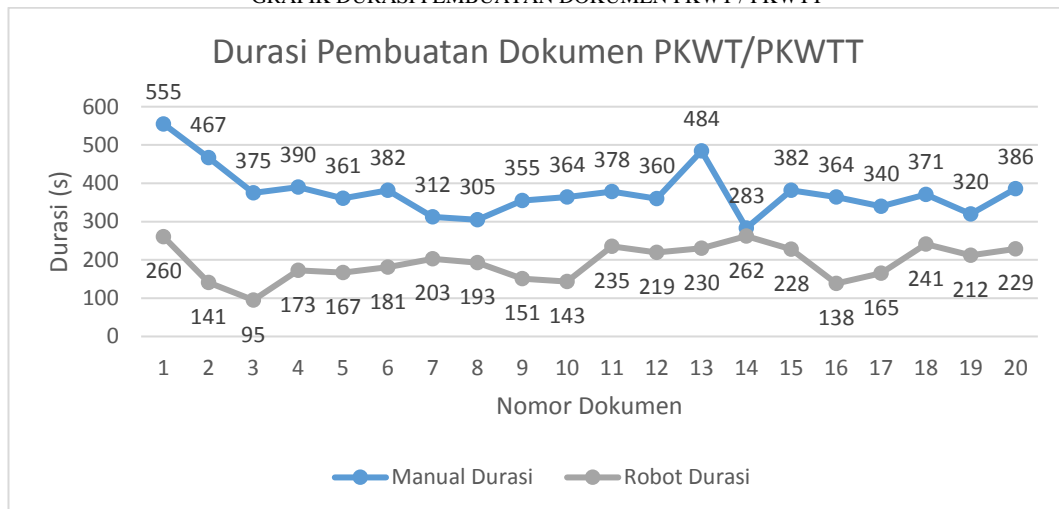
Pada percobaan pertama dilakukan pembuatan Dokumen PKWT / PKWTT sebanyak 20 buah sampel. Tujuan dari percobaan ini adalah mendapatkan parameter durasi waktu yang dibutuhkan dalam pembuatan tersebut. Parameter waktu akan dibuat dalam satuan detik (second). Hasil dari percobaan ini dapat dilihat pada tabel 1 dan gambar 2.

TABEL 1
DURASI PEMBUATAN 20 DOKUMEN PKWT / PKWTT

Nomor Dokumen	Durasi Manual (s)	Durasi Robot (s)	Jenis Dokumen
1	555	260	PKWT
2	467	141	PKWT
3	375	95	PKWTT
4	390	173	PKWTT
5	361	167	PKWTT
6	382	181	PKWT
7	312	203	PKWTT
8	305	193	PKWTT
9	355	151	PKWTT
10	364	143	PKWTT
11	378	235	PKWT
12	360	219	PKWTT
13	484	230	PKWT
14	283	262	PKWTT
15	382	228	PKWTT
16	364	138	PKWT
17	340	165	PKWTT
18	371	241	PKWT

19	320	212	PKWTT
20	386	229	PKWT
Total (s)	7534 / 02:05:34	3866 / 01:04:26	
Rerata (s)	376,7	193,3	

GAMBAR 2
GRAFIK DURASI PEMBUATAN DOKUMEN PKWT / PKWTT



Dari grafik berwarna abu yang merupakan grafik durasi pembuatan dokumen dengan robot, dapat dilihat bahwa durasi pembuatan Dokumen PKWT / PKWTT paling lama membutuhkan waktu 262 detik pada dokumen pertama. Durasi pembuatan dokumen PKWT / PKWTT paling sebentar membutuhkan waktu 95 detik pada dokumen ketiga. Total durasi yang

dibutuhkan untuk membuat 20 dokumen adalah selama 1 jam 4 menit 26 detik.

Pada grafik berwarna biru, atau grafik durasi pembuatan 20 dokumen secara manual menunjukkan bahwa durasi tercepat dalam membuat dokumen secara manual adalah selama 283 detik pada dokumen nomor 14. Sedangkan, durasi terlama dalam membuat dokumen secara manual adalah selama 555 detik pada percobaan pertama.

TABEL 2
PERBANDINGAN DURASI PEMBUATAN DOKUMEN PKWT / PKWTT

Durasi	Manual	Robot
Durasi Total	2:05:34	1:04:26
Durasi / dokumen	376.7	193.3
Pengurangan Durasi	$P = \frac{7534-3866}{7534} \times 100 = 48.69\%$	

Dari data yang diperoleh dapat disimpulkan bahwa RPA dapat memangkas

waktu pembuatan Dokumen PKWT / PKWTT sebanyak 48.69%. Dalam proses

bisnis, selisih waktu yang didapatkan dari implementasi RPA sangat bermanfaat untuk meningkatkan produktivitas seseorang dalam bekerja.

B. Pengujian Pembuatan Dokumen PKWT

Selanjutnya merupakan perhitungan durasi pembuatan Dokumen PKWT berdasarkan tabel 1. Skenario ini dilakukan untuk mendapatkan perbandingan durasi pembuatan Dokumen PKWT antara proses robot dan proses manual.

TABEL 3
DURASI PEMBUATAN DOKUMEN PKWT

Durasi Dokumen PKWT	Manual	Robot
Durasi Total	56:27	27:35
Durasi / dokumen	423,38	206,88
Pengurangan Durasi	$P = \frac{3387-1655}{3387} \times 100 = 51.13\%$	

Berdasarkan grafik yang ditampilkan pada tabel 3, durasi yang dibutuhkan dalam membuat delapan Dokumen PKWT dengan robot membutuhkan waktu selama 27 menit 35. Apabila dilakukan secara manual, maka durasi yang dibutuhkan untuk membuat delapan Dokumen PKWT adalah selama 56 menit 27 detik. Hasil dari perhitungan performansi yang diperoleh pada tabel 3. menunjukan bahwa penggunaan RPA dapat memangkas waktu sebesar 51,13% dalam pembuatan dokumen PKWT.

C. Pengujian Pembuatan Dokumen PKWTT

Skenario pengujian ketiga adalah perhitungann pembuatan Dokumen PKWTT secara manual dan secara robot yang diambil dari tabel 1. Tujuan dari skenario ini adalah didapatnya perbedaan durasi serta efektivitas RPA dalam membuat Dokumen PKWT dengan parameter yang digunakan adalah parameter durasi (s). Hasil data akan ditampilkan oleh tabel 4.

TABEL 4
PERBANDINGAN DURASI PEMBUATAN DOKUMEN PKWTT

Durasi Dokumen PKWT	Manual	Robot
Durasi Total	01:09:07	36:51
Durasi / dokumen	345,58	184,25
Pengurangan Durasi	$P = \frac{4147-2211}{4147} \times 100 = 46,68\%$	

Dari data yang diperoleh pada tabel 4, waktu pengerjaan yang dibutuhkan untuk membuat 12 Dokumen PKWTT dengan menggunakan RPA adalah selama 36 menit 51. Sedangkan, waktu yang dibutuhkan untuk membuat 12 Dokumen PKWTT secara manual adalah selama 1 jam 9 menit 7 detik. Sementara itu, berdasarkan pengukuran uji

performansi RPA dalam pembuatan Dokumen PKWTT, RPA dapat memangkas waktu durasi sebesar 46,68% jika dibandingkan dengan durasi pembuatan Dokumen PKWTT secara manual.

D. Pengujian Durasi Validasi

Skenario pengujian ini merupakan pengujian dari dua buah jenis validasi yang digunakan pada pelaksanaan Tugas Akhir ini. Seluruh data yang didapat pada pengujian ini

akan dibuatkan perbandingan durasi yang dibutuhkan antara Validasi UiPath dengan Validasi Reguler. Data yang didapatkan terlampir pada tabel 7.

TABEL 7
DURASI VALIDASI

No. Dokumen	Durasi (s)	
	Validasi UiPath	Validasi Reguler
1	108	170
2	41	105
3	22	130
4	65	97
5	62	93
6	79	100
7	65	84
8	54	153
9	47	129
10	29	147
11	44	110
12	92	152
13	87	106
14	135	85
15	147	101
16	21	65
17	22	76
18	80	74
19	63	71
20	122	101
Durasi Total	1385/23:05	2149/35:49
Rerata	69,25	107,45

Dari data yang diperoleh, total durasi yang dibutuhkan untuk melakukan validasi dengan metode Validasi UiPath adalah

selama 23 menit 5 detik. Sedangkan, total durasi yang dibutuhkan pada proses Validasi reguler adalah selama 35 menit 49.

TABEL 8
PERBANDINGAN DURASI VALIDASI

Durasi Validasi	Validasi UiPath	Validasi Reguler
Durasi Total	23:05	35:49
Durasi / dokumen	69,25	107,45
Pemangkasan Durasi	$\frac{2149 - 1385}{2149} \times 100\% = 34,71\%$	

Dari data yang ditampilkan oleh tabel 8, dapat diketahui bahwa durasi validasi dengan menggunakan metode Validasi UiPath akan lebih cepat sebesar 34,71% apabila dibandingkan dengan durasi validasi dengan metode Validasi Reguler. Oleh karena itu, Validasi UiPath sangat direkomendasikan dalam penggunaannya.

Pengujian selanjutnya merupakan uji performansi mesin UiPath OCR. Pengujian ini dilakukan dengan menghitung akurasi yang dihasilkan oleh UiPath OCR. Tujuan dari dilakukannya pengujian ini adalah untuk menentukan akurasi yang dihasilkan dari hasil ekstraksi mesin UiPath OCR dengan mendapatkan nilai persentasenya.

E. Pengujian Performansi UiPath OCR

TABEL 9
HASIL PENGUJIAN UIPATH OCR.

Akurasi UiPath OCR	
No Dokumen	Nilai Akurasi (%)
1	92.2
2	97.22
3	100
4	94.95
5	100
6	99
7	97.93
8	100
9	99.38
10	100
11	97.66
12	100
13	94.70
14	92.78
15	78.96
16	100
17	98.61
18	97.84
19	96.44
20	81.40

Rerata Total	95.95
--------------	-------

Hasil yang ditampilkan pada tabel 3 menunjukkan bahwa performa yang dihasilkan dalam ekstraksi dengan menggunakan UiPath OCR cukup baik. Dua akurasi terendah dimiliki oleh dokumen sampel dengan nomor 15 dan 20. Kedua dokumen tersebut memiliki nilai akurasi rerata sebesar 78.96 % dan 81.40 %. Akurasi rerata yang didapat dari pembuatan 20 dokumen dengan menggunakan UiPath OCR adalah sebesar 95.95 %.

F. Pengujian Performansi Microsoft OCR

Skenario berikutnya adalah pengujian ekstraksi 20 sampel foto KTP dengan menggunakan Microsoft OCR. Sama seperti pengujian Performansi UiPath OCR, uji performansi ini memiliki tujuan untuk mendapatkan nilai akurasi dari Microsoft OCR.

TABEL 10
HASIL PENGUJIAN MICROSOFT OCR.

Akurasi Microsoft OCR	
No Dokumen	Nilai Akurasi (%)
1	86.34
2	92.74
3	85.81
4	80.91
5	72.26
6	41.75
7	68.91
8	87.5

TABEL 11
HASIL PENGUJIAN OMNIPAGE OCR.

Akurasi OmniPage OCR	
No.	Rerata (%)
1	94.95
2	95.28
3	93.21
4	88.04
5	93.18
6	50.28
7	71.28
8	81.51
9	93.92
10	97.05
11	94.76
12	95.37
13	19.92
14	24.29

9	90.36
10	87.5
11	62.95
12	78.91
13	54.71
14	35.69
15	88.28
16	86.61
17	66.15
18	96.88
19	20.83
20	66.86
Rerata Total	72,60

Dari data yang didapat, terdapat tiga dokumen yang memiliki nilai akurasi dibawah 50 %. Ketiga data tersebut merupakan data sampel nomor 6, 14 dan 19 dengan nilai persentase berturut-turut adalah sebesar 41.75%, 35.69% dan 20,83%. Nilai akurasi rerata dari pembuatan 20 sampel ktp dengan menggunakan Microsoft OCR adalah sebesar 72.60 %.

G. Pengujian OmniPage OCR

Pada skenario ke-lima ini dilakukan ekstraksi sampel dokumen KTP dengan menggunakan OmniPage OCR. Skenario ini dilakukan untuk mengetahui nilai akurasi OmniPage OCR secara keseluruhan.

15	49.72
16	98.16
17	95.26
18	97.19
19	71.24
20	50.17
Rerata Total	78.02

Tabel 11 menampilkan data nilai akurasi yang dihasilkan oleh OmniPage OCR. Dari data tersebut, didapatkan hasil rerata akurasi dalam proses ekstraksi 20 sampel dokumen dengan menggunakan OmniPage OCR mendapatkan nilai akurasi sebesar 78.02%.

H. Pengujian PyTesseract OCR

Skenario pengujian ini merupakan ekstraksi data 20 sampel KTP dengan menggunakan PyTesseract OCR. Dalam implementasinya, penggunaan PyTesseract OCR menggunakan metode validasi reguler yang mana harus menggunakan format yang sama persis dengan format data pada KTP.

TABEL 12

HASIL PENGUJIAN PYTESSERACT OCR.

Akurasi PyTesseract OCR	
No.	Rerata (%)
1	86.23
2	95.46
3	91.81
4	97.34
5	98.81
6	97.72
7	97.66
8	93.51
9	95.96
10	96.36
11	98.44
12	95.64


13	94.67
14	91.49
15	97.92
16	99.22
17	93.59
18	98.98
19	100
20	94.14
Rerata Total	95.75






Dari data yang ditampilkan oleh tabel 12, tidak ada sampel data yang memiliki nilai akurasi dibawah 85%. Nilai terkecil yang dihasilkan oleh PyTesseract OCR adalah sebesar 86.23% pada sampel dokumen 1. Selain itu, nilai akurasi rerata yang dihasilkan oleh PyTesseract OCR dalam membuat 20 dokumen adalah sebesar 95.75%. Akurasi yang dihasilkan dapat bersaing dengan akurasi pada UiPath OCR.





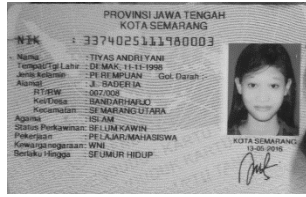
I. Perbandingan Akurasi Mesin OCR




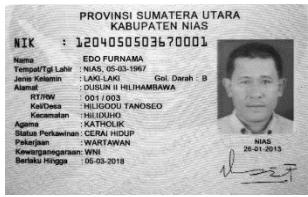
Skenario terakhir merupakan perbandingan nilai akurasi yang dihasilkan oleh masing-masing OCR pada setiap dokumen. Pada bagian ini akan dianalisa faktor apa saja yang dapat mempengaruhi hasil data ekstraksi dari sisi karakteristik gambar yang digunakan. Terdapat empat variabel karakteristik gambar yang dapat mempengaruhi hasil ekstraksi, yaitu : format gambar, dimensi gambar, kecerahan dan kejelasan gambar dan sudut pengambilan gambar.



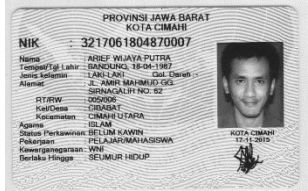

TABEL 13
PERBANDINGAN AKURASI MESIN OCR.

No.	UiPath OCR	Microsoft OCR	OmniPage OCR	PyTesseract OCR	Karakteristik Gambar
1	92,20	86,34	94,95	86,23	 <p>- Format : JPG</p> <p>- Dimensi : 736 x 452</p>

					- Gambar cukup cerah dan jelas - Sudut foto tegak lurus
2	97,22	92,74	95,28	95,46	 - Format : JPG - Dimensi : 563 x 332 - Gambar cukup cerah dan jelas - Sudut foto tegak lurus
3	100,00	85,81	93,21	91,81	 - Format : JPG - Dimensi : 564 x 365 - Gambar cukup cerah dan jelas - Sudut foto tegak lurus
4	94,95	80,91	89,95	97,35	 - Format : JPG - Dimensi : 564 x 358 - Gambar sedikit gelap dan jelas - Sudut foto tegak lurus
5	100,00	72,26	93,18	98,81	 - Format : JPG - Dimensi : 563 x 344 - Gambar cukup cerah dan jelas - Sudut foto tegak lurus
6	99,00	41,75	50,28	97,72	 - Format : JPG - Dimensi : 894 x 562 - Gambar sefikit gelap dan jelas

					- Sudut foto tegak lurus
7	97,93	68,91	71,28	97,66	 <p>- Format : JPG</p> <p>- Dimensi : 563 x 359</p> <p>- Gambar sedikit gelap dan buram</p> <p>- Sudut foto sedikit miring</p>
8	100,00	87,50	81,57	93,51	 <p>- Format : JPG</p> <p>- Dimensi : 564 x 364</p> <p>- Gambar cukup cerah dan jelas</p> <p>- Sudut foto tegak lurus</p>
9	99,38	90,36	93,92	95,96	 <p>- Format : JPG</p> <p>- Dimensi : 563 x 351</p> <p>- Gambar cukup cerah dan jelas</p> <p>- Sudut foto sedikit miring</p>
10	100,00	87,50	98,61	96,36	 <p>- Format : JPG</p> <p>- Dimensi : 563 x 376</p> <p>- Gambar cukup cerah dan jelas</p> <p>- Sudut foto tegak lurus</p>
11	97,66	62,95	94,76	98,44	 <p>- Format : JPG</p> <p>- Dimensi : 1024 x 645</p> <p>- Gambar sedikit gelap dan buram</p> <p>- Sudut foto tegak lurus</p>

12	99,68	78,91	95,37	95,64	 <p>- Format : JPG</p> <p>- Dimensi : 563 x 365</p> <p>- Gambar cukup cerah dan jelas</p> <p>- Sudut foto tegak lurus</p>
13	94,70	54,71	19,92	94,67	 <p>- Format : JPG</p> <p>- Dimensi : 717 x 459</p> <p>- Gambar sedikit gelap dan jelas</p> <p>- Sudut foto tegak lurus</p>
14	92,78	35,69	24,29	91,49	 <p>- Format : JPG</p> <p>- Dimensi : 752 x 564</p> <p>- Gambar sedikit gelap dan buram</p> <p>- Sudut foto sedikit miring</p>
15	78,96	88,28	51,81	97,92	 <p>- Format : JPG (dari pdf)</p> <p>- Dimensi : 564 x 357</p> <p>- Gambar sedikit gelap dan buram</p> <p>- Sudut foto tegak lurus</p>
16	100,00	86,61	98,16	99,22	 <p>- Format : PNG</p> <p>- Dimensi : 608 x 381</p> <p>- Gambar cukup cerah dan jelas</p> <p>- Sudut foto tegak lurus</p>

17	98,61	66,15	95,26	93,59	 <p>- Format : PNG - Dimensi : 493 x 300 - Gambar cukup cerah dan jelas - Sudut foto tegak lurus</p>
18	97,84	96,88	97,19	98,98	 <p>- Format : JPG - Dimensi : 563 x 355 - Gambar cukup cerah dan jelas - Sudut foto tegak lurus</p>
19	96,44	20,83	71,24	100,00	 <p>- Format : PNG - Dimensi : 1070 x 707 - Gambar cukup cerah dan jelas - Sudut foto tegak lurus</p>
20	81,40	66,86	50,17	94,14	 <p>- Format : JPG - Dimensi : 564 x 363 - Gambar sedikit gelap dan cukup jelas - Sudut foto tegak lurus</p>
Akurasi Rerata	95,95	72,60	78,02	95,75	

Dari hasil yang didapat pada perbandingan hasil akurasi mesin OCR dapat dianalisa bahwa Mesin OCR UiPath bisa bekerja dengan baik tidak tergantung dengan skala dimensi, kondisi kecerahan gambar ataupun sudut pandang foto. Sementara itu, data ekstraksi yang dihasilkan oleh Microsoft

OCR sangat berpengaruh terhadap dimensi gambar. Oleh karena itu, penggunaan Microsoft OCR diperlukan pencocokan skala dengan dimensi gambar yang akan digunakan. Untuk hasil data ekstraksi pada OmniPage OCR, sudut foto dan kecerahan gambar cukup berpengaruh terhadap hasil

ekstraksi. Sedangkan pada mesin OCR PyTesseract cukup stabil dalam mengekstraksi seluruh jenis gambar yang ada. Mesin OCR PyTesseract dapat mengekstraksi dengan baik, tidak terpengaruh oleh dimensi, kecerahan gambar ataupun sudut gambar.

IV. KESIMPULAN

Sistem robot yang dirancang untuk melakukan otomatisasi proses rekrutmen karyawan dapat bekerja dengan baik menggunakan metode *attended* robot pada *environment* UiPath. Penggunaan metode *attended* disebabkan oleh masih terdapatnya peran manusia dalam memvalidasi data KTP. Implementasi RPA dapat mempersingkat durasi waktu yang diperlukan dalam membuat dokumen sebanyak 48.69%. Selain mempersingkat durasi, RPA juga dapat meningkatkan akurasi pembuatan dokumen hingga 100% apabila proses validasi dilakukan dengan benar. Disamping itu, produktivitas bisnis juga dapat meningkat karena pengguna sistem ini dapat mengerjakan pekerjaan lain dari durasi waktu yang berhasil dikurangi. Mesin OCR yang direkomendasikan untuk digunakan dalam proses ekstraksi adalah UiPath OCR dengan akurasi performansi sebesar 95.95%. Salah satu alasan dari rekomendasi UiPath OCR adalah karena metode validasi yang digunakan dapat lebih cepat sebesar 34.71% dibandingkan metode validasi pada PyTesseract OCR. Implementasi PyTesseract OCR dapat digunakan dengan baik apabila terdapat alternatif cara untuk mempersingkat durasi waktu pada proses validasi dokumen.

REFERENSI

- [1] Senen, F. Riyandi, Nurhayati, V. Komalasari, M. Monika, and R. Yadi, "PEREKUTAN SDM DI ERA PANDEMI COVID 19," vol. 2, p. 332, 2021, Accessed: Nov. 10, 2021. [Online]. Available: <http://openjournal.unpam.ac.id/index.php/JAL/article/view/10506/6513#>.
- [2] A. Asuan, "Perlindungan Hukum Terhadap Pekerja Berstatus Perjanjian Kerja Waktu Tertentu (PKWT) Menurut Undang-Undang Nomor 13 Tahun 2003 Tentang Ketenagakerjaan," *Solusi*, vol. 17, no. 1, pp. 23–31, 2019.
- [3] N. Achyar, K. Sujatmoko, and I. I. Tritoasmoro, "Penggunaan Template Matching Pada Optical Character Recognition Dengan Threshold," *eProceedings Eng.*, vol. 8, no. 5, 2021.
- [4] D. Fernando and H. Harsiti, "Studi Literatur: Robotic Process Automation," *JSiI (Jurnal Sist. Informasi)*, vol. 6, no. 1, pp. 6–11, 2019.
- [5] A. Hutomo, A. S. Raharjo, and M. H. Syahbani, "Perancangan Aplikasi Rute Tercepat Perjalanan Paket Berbasis Mobile Menggunakan Metode Robotic Process Automation (rpa)," *eProceedings Eng.*, vol. 8, no. 5, 2021.
- [6] K. Inc, "The Art and Science of Kofax OmniPage OCR," 2021. <https://www.kofax.com/-/media/files/white-papers/en/the-art-and-science-of-omnipage-ocr.pdf> (accessed Nov. 23, 2021).
- [7] R. W. Smith, "History of the Tesseract OCR engine: what worked and what didn't," in *Document Recognition and Retrieval XX*, 2013, vol. 8658, p. 865802.
- [8] N. R. D. P. Astuti, F. Noviyanto, and D. Soyusiawati, "Forensik Digital Metode RegEx (Regular Expression) dari Grab Google Search Api dalam Proses Pelacakan Terhadap Kejahatan Online," *InfoTekJar J. Nas. Inform. dan Teknol. Jar.*, vol. 3, no. 1, pp. 90–94, 2018.
- [9] D. Kastowo, A. Saputra, W. D. Suryono, and E. Setyowati, "Analisis Perbandingan Algoritma Nazief Adriani dan Levenshtein Distance untuk mengukur Tingkat Similaritas Berita Menggunakan Rabin Krap: Studi Kasus Berita Berbahasa Jawa," *JNANALOKA*, pp. 1–10, 2022.
- [10] A. Setiawan, O. N. Pratiwi, and R. Y. Fa'rifah, "Question Answering System Dalam Bentuk Chatbot Pada Platform Line Untuk Mata Pelajaran Sejarah Sma/ma Dengan Menggunakan Algoritma Levenshtein Distance," *eProceedings Eng.*, vol. 8, no. 5, 2021.