PENERJEMAHAN HURUF NON-LATIN CYRILLIC RUSIA KE HURUF LATIN MENGGUNAKAN ALGORITMA RANDOM FOREST

(TRANSLATION RUSSIAN CYRILLIC TO LATIN ALPHABET USING RANDOM FOREST ALGORITHM)

Muhammad Mufti Ramadhan[1], Budhi Irawan S.Si., M.T.[2], Casi Setianingsih S.T., M.T.[3]

[1,2,3]Prodi S1 Sistem Komputer, Fakultas Teknik Elektro, Telkom University
Bandung, Indonesia

[1]m.muftiramadhan@gmail.com, [2]budhiirawan@telkomuniversity.ac.id, [3]setiacasie@telkomuniversity.ac.id

Abstrak

Bahasa Rusia adalah salah satu bahasa dengan penutur terbanyak yang menempati peringkat ke-6 di dunia. Hal ini menunjukan tingginya minat penduduk dunia untuk menggunakan dan mempelajari bahasa tersebut. Bahasa Rusia memiliki aksara yang berbeda dengan aksara Indonesia yang disebut *cyrillic*, di mana bentuk hurufnya berbeda dari huruf pada umumnya sehingga terdapat beberapa kendala untuk mempelajari, memahami dan melafalkannya. Pesatnya perkembangan teknologi saat ini merubah gaya hidup masyarakat yang lebih bersifat praktis dan *mobile* sehingga smartphone menjadi salah satu kebutuhan bagi masyarakarat. Android merupakan salah satu sistem operasi yang banyak digunakan pada smartphone saat ini, sehingga aplikasi Capture to Translate dibangun dengan basis Android.

Capture to Translate adalah media yang dirancang untuk menjadi sebuah solusi dari permasalahan tersebut. Sistem dibangun berdasarkan *image processing*, proses ekstraksi ciri dan kecerdasan buatan *(artificial intelligence)* menggunakan algoritma klasifikasi *Random Forest* dengan antar muka aplikasi *mobile* Android yang memanfaatkan kamera perangkat sebagai inputnya.

Pengujian yang dilakukan pada penelitian ini memberikan nilai akurasi terbaik sebesar 93,33% untuk 1 silabel, 90% untuk 2 silabel dan 84,29% untuk 3 silabel dan dicapai pada resolusi 6MP serta dengan jarak 30cm. Pengujian juga menunjukkan bahwa tingkat kompleksitas huruf, resolusi, jarak, sudut pengambilan serta tingkat pencahayaan suatu citra berpengaruh dalam proses transliterasi.

Kata kunci : Rusia, *Cyrillic*, *Capture to Translate*, *Image Processing*, Kecerdasan Buatan, *Random Forest*, *Android*.

Abstract

Russian language ranked as the sixth most spoken language in the world. It shows a high interest of the world population to use and learn the language. The Russian language has different alphabets from Indonesian language, which is called Cyrillic. It has a different shape from the alphabets in general which makes some obstacles in order to learn, understand and pronounce it. Currently, technology is growing really fast, and it changed people's lifestyle to be more practical and mobile so that smartphone becomes one of the society's needs. Android is one of the most used operating system in smartphone nowadays, so the Capture to Translate was built in android-based.

Capture to Translate is a media which is designed as a solution for those problems. The system was built based on image processing, feature extraction process and artificial intelligence using Random Forest algorithm classification with interface Android mobile application that utilizes the camera as an input device.

The test conducted in this study gives the best accuracy value at 93,33% for one syllable, 90% for two syllables, and 84,29% for three syllables and it achieved using 6MP resolution with a distance of 30cm. The test also shows that the level of alphabet complexity, resolution, distance, angle and lighting of an image have an influence on transliteration process.

Keywords : Russian, Cyrillic, Capture to Translate, Image Processing, Artificial Intelligence, Random Forest, Android.

1. Pendahuluan

Bahasa Rusia adalah salah satu bahasa dengan penutur terbanyak yang menempati peringkat ke-6 didunia. Bahasa Rusia merupakan satu dari enam bahasa resmi yang digunakan oleh *United Nation (UN)* atau yang biasa

dikenal di Indonesia sebagai Perserikatan Bangsa-Bangsa (PBB). Hal ini menunjukan bahwa tingginya minat penduduk dunia untuk menggunakan dan mempelajari bahasa tersebut akan tetapi terdapat beberapa kendala untuk mempelajari, memahami dan melafalkan Bahasa Rusia dikarenakan bahasa ini menggunakan aksara non-latin *cyrillic*.

Beberapa contoh permasalahan yang sering terjadi adalah seperti saat masyarakat asing hendak membaca tulisan bahasa Rusia namun terdapat kesulitan dalam membacanya/melafalkannya, kesulitan dalam mengetikkan sebuah nama tempat atau nama daerah dalam bahasa tersebut, dan kesulitan dalam mencari arti dari bahasa tersebut.

Capture to translate adalah salah satu media yang dibangun untuk menjadi sebuah solusi dari permasalahan tersebut, sistem ini dibangun berdasarkan pengolahan *image processing* dan kecerdasan buatan *(Artificial Intelligence).* Sistem menggunakan antar muka aplikasi *mobile* android yang memanfaatkan kamera dan galeri perangkat sebagai media inputan sistem. Setelah melalui proses akuisisi, citra yang mengandung teks dalam bahasa Rusia akan diproses terlebih dahulu ke dalam tahapan *image processing*. Setelah itu sistem akan berjalan secara otomatis menjalankan proses transliterasi ke aksara latin. Proses transliterasi ini berjalan diatas *machine learning* yang sebelumnya telah diberikan proses pelatihan kemudian di sinkronisasi menggunakan algoritma klasifikasi *Random Forest.* Hasil akhir yang didapatkan dari aplikasi adalah hasil transliterasi ke huruf latin.

2. Dasar Teori

2.1 Bahasa Rusia

Bahasa Rusia adalah salah satu bahasa dengan penutur terbanyak yang menempati peringkat ke-6 di dunia. Huruf *cyrillic* memiliki huruf yang berbeda dengan huruf latin dan setiap masing-masing huruf dapat memiliki lebih dari satu huruf transliterasi.

А Б В Г Д Е
Ё Ж З И Й К
Л М Н О П Р
С Т У Ф Х Ц
Ч Ш Щ Ъ Ы Ь
Э Ю Я

Gambar 2.1 Rusia

2.2 Digital Image Processing

Digital Image Processing adalah kumpulan metode-metode yang mempelajari tentang teknik-teknik mengolah sebuah citra. Sebuah citra digital dapat diwakili oleh sebuah matriks dua dimensi *f*(*x*,*y*) yang terdiri dari *M* kolom dan *N* baris, di mana perpotongan antara kolom dan baris disebut piksel (*pixel = picture element*) atau elemen terkecil dari sebuah citra.[1]

$$f(x,y) \approx \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,M-1) \\ f(1,0) & f(1,1) & \dots & f(1,M-1) \\ \vdots & \vdots & \vdots & \vdots \\ f(N-1,0) & f(N-1,1) & \dots & f(N-1,M-1) \end{bmatrix}$$

Gambar 2.2 Elemen Citra

Suatu citra *f*(x,y) dalam fungsi matematis dapat dituliskan sebagai berikut:

$0 \le x \le M-1$ .....(2.1)

$0 \le y \le N-1$ .....(2.2)

$0 \le f(x,y) \le G-1$ .....(2.3)

dimana : M = jumlah piksel baris (row) pada array citra

N = jumlah piksel kolom (column) pada array citra

G = nilai skala keabuan (graylevel)

Besarnya nilai M, N dan G pada umumnya merupakan perpangkatan dari dua.

$$M = 2^m ; N = 2^n; G = 2^k$$

di mana nilai m, n dan k adalah bilangan bulat positif. Interval (0,G) disebut skala keabuan (*grayscale*). Besar G tergantung pada proses digitalisasinya. Biasanya keabuan 0 (nol) menyatakan intensitas hitam dan 1 (satu) menyatakan intensitas putih. Untuk citra 8 bit, nilai G sama dengan 28 = 256 warna (derajat keabuan) [2].

2.3 Optical Character Recognition (OCR)

OCR (*Optical Character Recognition*) adalah sebuah aplikasi komputer yang digunakan untuk mengidentifikasi citra huruf maupun angka untuk dikonversi ke dalam bentuk file tulisan [3]. Sistem pengenal huruf ini dapat meningkatkan fleksibilitas atau kemampuan dan kecerdasan sistem komputer [4].

2.4 Algoritma Klasifikasi Random Forest

Random Forest yang dihasilkan memiliki banyak *tree*, dan setiap *tree* ditanam dengan cara yang sama[5]. *Tree* dengan variabel x akan ditanam sejauh mungkin dengan *tree* dengan variabel y. *Ensemble* adalah pendekatan *divide* and *conquer* yang digunakan untuk meningkatkan kinerja. Prinsip utama di balik metode ensemble adalah bahwa kelompok *"weak-learner"* dapat dikumpulkan dan membentuk sebuah *"strong-learner"*[6]. Runtimes *random forest* cukup cepat, dan mampu menangani data yang tidak seimbang dan tidak lengkap. Kelemahan *random forest* pada regresi tidak dapat memprediksi nilai yang diluar jangkauan pada data training dan memiliki kemungkinan melakukan *over-fit* pada data yang memiliki sangat banyak *noise*. Algoritma *training* untuk *random forest* adalah dengan menggunakan *bootstrap aggregating* (*bagging*). Proses latih dilakukan dengan mengambil satu set data latih yang kemudian akan dimasukkan ke dalam suatu *tree*. Pemilihan atribut dalam setiap kali sebuah node akan dipecah diambil secara acak. Bagging melakukan pemilihan sample berulang kali, dengan penggantian[7].

2.5 Deteksi Tepi

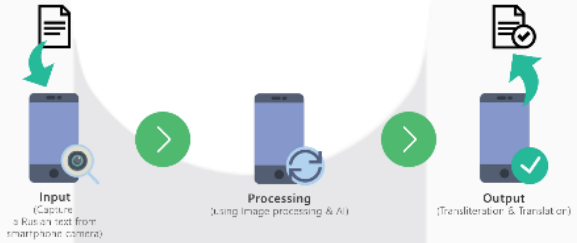
Deteksi tepi (*Edge detection*) adalah operasi yang dijalankan untuk mendeteksi garis tepi (*edges*) yang membatasi dua wilayah citra homogen yang memiliki tingkat kecerahan yang berbeda. Tepi (*edge*) adalah perubahan nilai intensitas derajat keabuan yang cepat/tiba-tiba (besar) dalam jarak yang singkat. Sedangkan deteksi tepi (*Edge Detection*) pada suatu citra adalah suatu proses yang menghasilkan tepi-tepi dari obyek-obyek citra, tujuannya adalah untuk (a) menandai bagian yang menjadi detail citra; dan (b) memperbaiki detail dari citra yang kabur, yang terjadi karena error atau adanya efek dari proses akuisisi citra. Suatu titik (x,y) dikatakan sebagai tepi (*edge*) dari suatu citra bila titik tersebut mempunyai perbedaan yang tinggi dengan tetangganya.[8]

3. Pembahasan

Capture to Translate merupakan sebuah aplikasi dimana sistem yang dirancang dan dibangun pada perangkat smartphone berbasis Android. Sistem dirancang untuk dapat mengenali setiap huruf non-latin *cyrillic* Rusia dan melakukan proses transliterasi ke huruf latin. Sistem menggunakan prinsip dasar OCR (*Optical Character Recognition*) yaitu proses pengenalan dimulai dari pengambilan citra atau proses akuisisi citra melalui kamera perangkat atau melalui gambar yang tersimpan di galeri perangkat. Setelah itu gambar diteruskan sebagai *input* oleh sistem, dan sistem akan secara otomatis melakukan *image processing*, *feature extraction* dan klasifikasi oleh A.I yang akan menghasilkan kata transliterasi ke dalam huruf latin.

3.1 Gambaran Umum Sistem

Untuk sistem yang dirancang memiliki gambaran umum sistem seperti gambar berikut.



Gambar 3.1 Gambaran Umum Sistem

3.2 Gambaran Khusus Sistem

Berikut gambaran khusus sistem pada penelitian ini :

1. *Input*

Citra merupakan hasil dari proses akuisisi dari kamera perangkat smartphone maupun melalui pengambilan citra dari galeri *smartphone* berformat JPEG. Citra yang dimaksud merupakan sebuah citra teks Cyrillic Rusia.

2. *Processing*

Terdiri dari *image processing*, *feature extraction* menggunakan metode *find contour*, algoritma klasifikasi *random forest* dan sebuah *database* terjemahan ke bahasa Indonesia.

3. *Output*

Output yang dihasilkan dari sistem ini berupa sebuah transliterasi dari huruf non-latin *cyrillic* Rusia ke huruf latin.

3.3 Fungsionalitas Sistem

Sistem yang dirancang memiliki fungsionalitas sebagai berikut:

a) Dapat mendeteksi kata non-latin *cyrillic* Rusia dari hasil proses akuisisi citra yang didapatkan baik dari kamera perangkat maupun gambar dari galeri perangkat.
b) Dapat menghasilkan hasil transliterasi kata dari huruf non-latin *cyrillic* Rusia ke dalam huruf latin Alphabet.

Pada perancangan sistem Capture to Translate terbagi menjadi empat proses dimana terdiri dari akuisisi citra, *pre-processing*, ekstraksi ciri dan klasifikasi.

3.4 Akuisisi Citra

Akuisisi citra yang didapatkan melalui dua cara yaitu pengambilan langsung oleh pengguna menggunakan media kamera pada *smartphone* atau memilih citra yang sebelumnya telah tersedia pada media penyimpanan (*gallery*). Terdapat beberapa faktor yang harus diperhatikan dalam proses akuisisi citra menggunakan kamera *smartphone* seperti sudut pengambilan, jarak, pencahayaan dan resolusi kamera.

3.5 Preprocessing

Preprocessing dilakukan untuk proses pengenalan teks pada sebuah citra dengan tujuan untuk menentukan bagian data yang dapat diolah. Di dalam penelitian ini terdapat beberapa proses yang dilakukan meliputi *grayscalling*, *thresholding* dan segmentasi. *Grayscalling* adalah proses untuk mengubah citra yang sebelumnya memiliki format warna 3 layer (RGB) ke dalam 1 layer (grayscale) agar memudahkan dalam melakukan proses pengolahan citra. *Rescalling* adalah sebuah operasi untuk memperbesar atau memperkecil ukuran citra digital. *Thresholding* adalah proses yang berfungsi untuk membedakan teks yang terdapat pada sebuah citra dengan backgroundnya. Morfologi citra adalah sebuah operasi sederhana yang mengacu pada bentuk sebuah citra dimana digunakan untuk mengidentifikasi bentuk. Segmentasi adalah sebuah tahapan dimana bagian citra yang hendak diolah dan dibagi (dipotong) bedasarkan huruf secara horizontal maupun vertikal.

3.6 Ekstraksi Ciri

Proses ekstraksi ciri adalah tahapan penting yang digunakan oleh algoritma klasifikasi dalam mengenal dan mempelajari sebuah citra. Teknik ini digunakan untuk mengambil informasi ciri dari sebuah objek yang dapat menggambarkan karakterisitik objek tersebut. Citra yang akan diproses adalah citra yang sebelumnya telah melalui tahap *preprocessing*. Teknik ekstraksi ciri yang digunakan pada penelitian ini menggunakan deteksi tepi dimana mengenali sebuah obyek dari bentuknya. Bagian dari citra yang telah dikenali sebagai obyek akan diolah untuk diambil informasi cirinya. Setelah didapatkan informasi ciri tersebut maka akan dilanjutkan ke dalam proses klasifikasi *machine learning*.

3.7 Klasifikasi Menggunakan Algoritma Random Forest

Algoritma klasifikasi *random forest* digunakan untuk mengklasifikasikan setiap sampel data yang masuk sebagai proses training. Tree akan ditanam berdekatan sesuai dengan variabel yang dikenali. Semakin berkembangnya dataset akan menghasilkan tree baru dan pada proses klasifikasinya akan didasrkan pada *vote* suara terbanyak pada kumpulan populasi *tree*. Terdapat beberapa perubahan nilai pada beberapa parameter. Parameter yang diatur adalah *max_depth*, *min_sample_count* dan *regression accuracy*.

Berikut gambaran secara garis besar cara kerja algoritma klasifikasi pada penelitian ini:

1. Proses *training*

Merupakan sekumpulan citra dari seluruh huruf *cyrillic* yang berjumlah 33 huruf. Masing-masing huruf memiliki 5 sampel citra, sehingga total citra training pada penelitian ini berjumlah 165 citra. Setiap 33 huruf Rusia telah diberikan *label* yang dimana pada algoritma klasifikasi disebut *class*. Sampel citra *training* memiliki resolusi 32x32 pixel.

2. Proses testing

Citra uji merupakan citra dengan 1, 2 dan 3 silabel lalu memasuki tahap *preprocessing* hingga segmentasi. Citra uji yang diolah adalah hasil dari proses segmentasi dan telah memiliki resolusi yang sama sebesar 32x32 pixel. Citra ini akan diproses ke tahap ekstraksi ciri oleh *find contour* sehingga menghasilkan keluaran berupa vektor. Vektor tersebut digunakan sebagai input ke tahapan klasifikasi untuk menguji dan memprediksi sebuah transliterasi. citra uji yang diolah adalah hasil dari proses segmentasi dan memiliki resolusi yang sama sebesar 32x32 pixel.

4. Pengujian

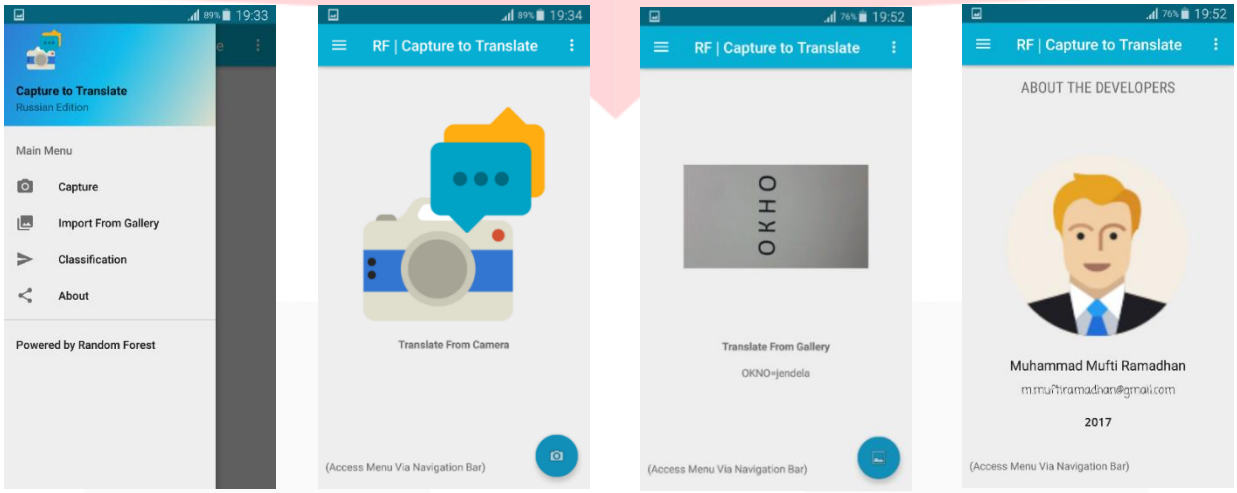
4.1 Pengujian Perangkat Lunak

Pengujian ini bertujuan untuk mengetahui hasil akhir dari sistem dan hasil yang diharapakan jika sistem diberikan suatu keadaan masukkan tertentu. Pengujian terdiri dari pengujian antarmuka aplikasi dan *black box*.

4.1.1 Pengujian Antarmuka Aplikasi

Seperti yang terlihat dari gambar 4.1 (a), aplikasi memiliki *navbar* yang berisi menu yang terdapat pada aplikasi. Pengguna dapat melakukan *swipe from left to right* atau klik pada ikon *option* yang berada pada pojok kiri atas aplikasi. Berikut penjelasan setiap menu yang terdapat pada aplikasi:

1. *Capture*, untuk melakukan akuisisi gambar melalui kamera perangkat.
2. *Import From Gallery*, untuk melakukan akuisisi gambar melalui galeri perangkat.
3. *Classification*, untuk melakukan proses klasifikasi dimana sistem akan mengenali objek huruf Rusia pada citra yang sebelumnya telah terakuisisi dan tersegmentasi oleh sistem.
4. *About* , untuk melihat lebih lanjut mengenai aplikasi.



(a) (b) (c) (d)

Gambar 4.1 (a) *Layout Navigation Panel*, (b) *Layout Capture from Camera*, (c) *Layout Translate*, (d) *Layout About*

4.1.2 Pengujian *Black Box*

Pengujian ini meliputi pengujian fungsi kamera dan fungsi *open gallery* sebagai media input / akuisisi citra, fungsi menu dan navigasi dan fungsi – fungsi lainnya yang berkaitan dengan *user interface*.

Tabel 4.1 Pengujian dan Analisis *Black Box*

No.	Data Masukkan	Hasil yang diharapkan	Hasil Sistem dan Pengamatan	Kesimpulan
1	Klik tombol floating button “capture”	Dapat menampilkan activity camera untuk akuisisi citra	Menampilkan activity camera dan dapat mengakuisisi citra	Diterima
	Hasil akuisisi citra dari kamera	Proses akuisisi dapat berjalan dan dapat menampilkan hasil akuisisi	Menampilkan hasil akuisisi citra	Diterima
2	Klik tombol “load picture”	Dapat menampilkan activity gallery untuk akuisisi citra	Menampilkan activity gallery dan dapat mengakuisisi citra	Diterima
	Hasil akuisisi citra dari gallery	Proses akuisisi dapat berjalan dan dapat menampilkan hasil akuisisi	Menampilkan hasil akuisisi citra	Diterima
3	Melakukan gesture swipe from left to right pada homescreen	Dapat menampilkan navigation drawer	Menampilkan navigation drawer dan seluruh pilihan menu	Diterima
4	Klik tombol “about” pada navigation drawer	Dapat menampilkan layout “about”	Menampilkan layout “about” dan penjelasan informasi aplikasi	Diterima

4.2 Pengujian Performansi Sistem

Salah satu acuan yang dapat digunakan sebagai dasar untuk mengukur tingkat keberhasilan sistem adalah menggunakan akurasi dengan metode validasi *Correct Acceptance* (CA).

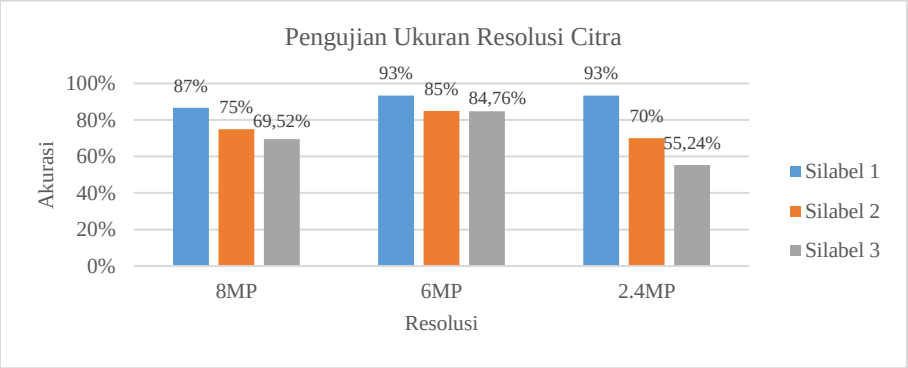
$$Akurasi = \frac{Jumlah\ huruf\ tertransliterasi\ benar}{jumlah\ total\ seluruh\ huruf} \text{ x } 100\% \qquad (4.1)$$

4.2.1 Pengujian Huruf dan Silabel

Tujuan dari pengujian ini adalah untuk mengetahui bagaimana performansi dan keberhasilan sistem saat diuji mendeteksi huruf *cyrillic* Rusia. Hasil pengujian nilai akurasi terbesar berturut-turtut dicapai oleh pengujian pada 1 silabel sebesar 93.33%, silabel 2 sebesar 90% dan silabel 3 sebesar 84,29%. Hal yang memengaruhi akurasi tinggi pada sampel uji 1 silabel adalah selain jumlah huruf yang sedikit tetapi juga tingkat kompleksitas huruf itu sendiri.

4.2.2 Pengujian Ukuran Resolusi Citra

Tujuan dari pengujian ini adalah untuk mengetahui bagaimana hasil dari sistem ketika diberikan masukan yang memiliki perbedaan ukuran resolusi kamera.

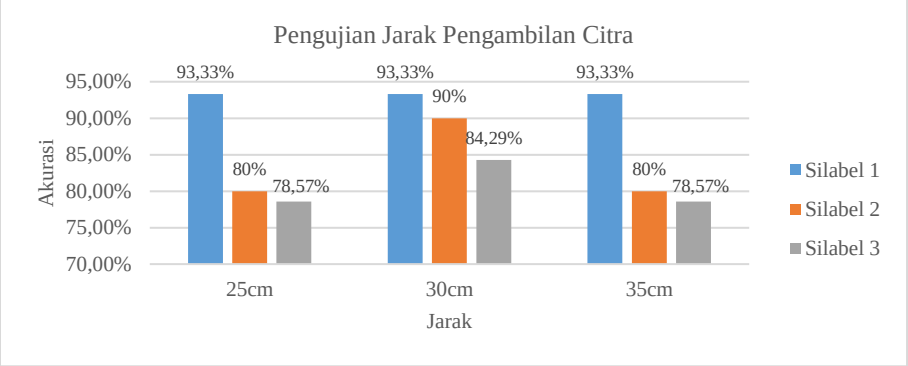


Gambar 4.2 Grafik Pengujian Ukuran Resolusi Citra

Berdasarkan gambar 4.2 dapat dilihat bahwa nilai akurasi tertinggi dicapai pada resolusi 3264x1836 (6MP). Faktor yang menyebabkan nilai akurasi tinggi pada 6MP adalah saat proses *preprocessing* gambar hasil akuisisi melalui tahap *rescaling* dengan resolusi 1024x576 yang memiliki rasio 16:9. Pada penelitian ini resolusi kamera yang memiliki rasio 16:9 adalah 2.4MP dan 6MP sehingga nilai akurasi tertinggi dicapai oleh 6MP.

4.2.3 Pengujian Jarak Pengambilan Citra

Pengujian ini dilakukan untuk mengetahui tingkat akurasi dari sistem dengan adanya pengaruh perbedaan jarak pengambilan citra teks yang terakuisisi oleh kamera smartphone.

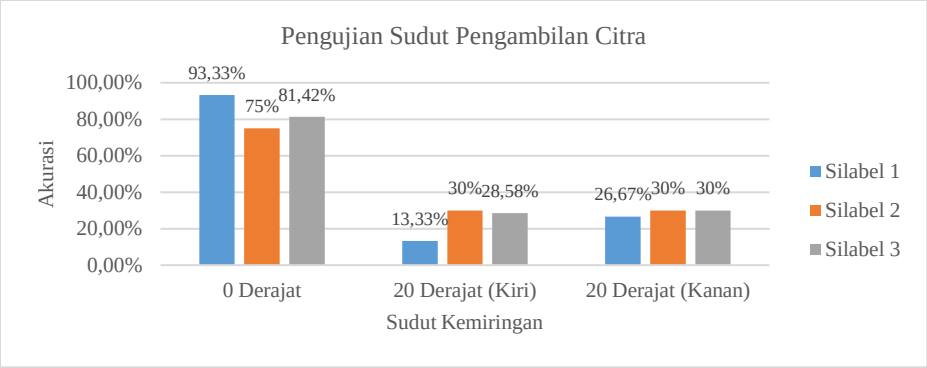


Gambar 4.3 Grafik Pengujian Jarak Pengambilan Citra

Dari hasil pengujian yang dapat dilihat pada gambar 4.3, menunjukkan bahwa nilai akurasi tertinggi dicapai oleh jarak 30cm. Pada pengujian ini dapat disimpulkan bahwa jarak pengambilan objek citra yang ideal pada penelitian ini adalah 30cm, sedangkan pada jarak 25cm dan 35cm terdapat beberapa huruf yang tidak terdeteksi sehingga tidak dapat memasuki proses transliterasi.

4.2.4 Pengujian Sudut Pengambilan Citra

Pengujian ini dilakukan untuk mengetahui seberapa besar pengaruh sudut pengambilan citra dalam keberhasilan sistem mendeteksi huruf Rusia.

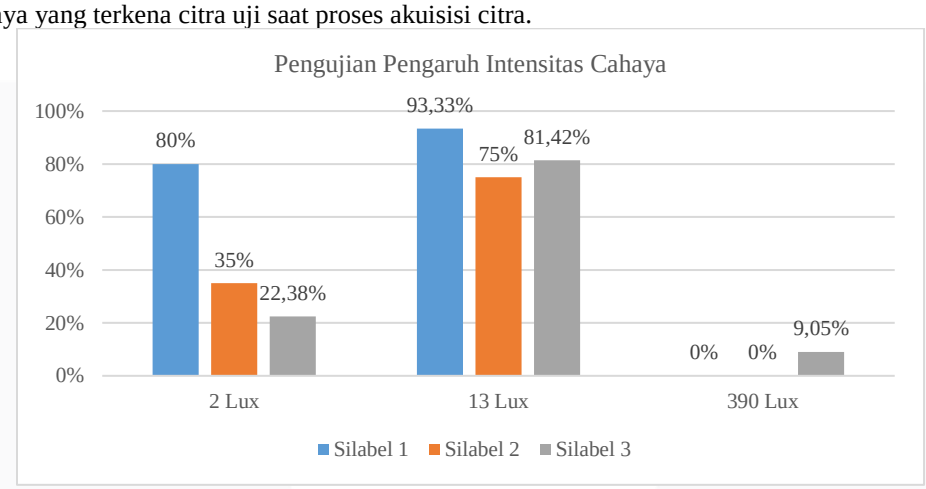


Gambar 4.4 Grafik Pengujian Sudut Pengambilan Citra

Gambar 4.4 menunjukkan bahwa nilai akurasi tertinggi dicapai saat sudut pengambilan citra sebesar 0 derajat dimana posisi lurus dengan citra uji. Pada sudut 20 derajat ke kiri dan sudut 20 derajat ke kanan menghasilkan kesimpulan bahwa sistem tidak dapat mendeteksi huruf secara sempurna sehingga berakibat pada proses pengenalan huruf. Beberapa huruf tidak dapat ditransliterasi dengan tepat karena terdapat kesulitan saat pencocokan ciri.

4.2.5 Pengujian Pengaruh Intensitas Cahaya Citra

Pengujian ini dilakukan untuk menguji seberapa besar pengaruh yang dihasilkan saat diberikan beragam intensitas cahaya yang terkena citra uji saat proses akuisisi citra.



Gambar 4.5 Grafik Pengaruh Intensitas Cahaya

Dari hasil pengujian, dapat disimpulkan bahwa tingkat akurasi tertinggi dicapai oleh tingkat pencahayaan sebesar 13 lux. Hal ini disebabkan citra latih yang digunakan pada penelitian ini memiliki tingkat pencahayaan sebesar 13 lux sehingga pada tingkat pencahayaan 2 lux memiliki penurunan nilai akurasi yang signifikan dan nilai akurasi yang kurang baik saat diberi tingkat pencahayaan 390 lux dimana terjadi karena perbedaan jarak cahaya yang sangat jauh. Cahaya berpengaruh pada hasil citra yang terakusisi dan saat melalui proses ekstraksi ciri, sistem akan mengalami kesulitan dalam mencocokan ciri dengan citra latih.

4.2.6 Pengujian Algoritma Klasifikasi

Pengujian ini dilakukan untuk menguji parameter algoritma *default* dan parameter yang telah dikustomisasi untuk penelitian ini. Parameter algoritma yang digunakan pada penelitian ini adalah algoritma *random forest*.

Tabel 4.2 Pengujian Algoritma

Silabel	Waktu		Akurasi	
	Default parameter	Custom parameter	Default parameter	Custom parameter
Silabel 1	37.2s	45s	93.33%	93.33%
Silabel 2	104.2s	119.6s	70%	85%
Silabel 3	164.4s	199.6s	62.38%	84.76%

Berdasarkan tabel 4.2 dapat disimpulkan bahwa dalam pengujian waktu kedua algoritma, hasil waktu tercepat adalah algoritma dengan parameter default dengan perbedaan waktu 7,8 detik pada 1 silabel, 15.4 detik pada 2 silabel dan 35.2 detik pada 3 silabel. Hal ini disebabkan karena pada algoritma penelitian ini menggunakan *regression accuracy*, dimana mengakibatkan sistem membutuhkan waktu yang sedikit lebih lama. Sedangkan hasil akurasi terbaik dicapai oleh algoritma dengan parameter kustom dengan perbedaan nilai akurasi sebesar 0% pada 1 silabel, 15% pada 2 silabel dan 22,38% pada 3 silabel. Hal ini disebabkan karena pada algoritma penelitian ini menggunakan *regression accuracy*, dimana memberikan hasil prediksi transliterasi yang lebih akurat.

5. Kesimpulan dan Saran

5.1 Kesimpulan

Berdasarkan hasil pengujian dan analisis dari tugas akhir ini dapat diambil beberapa kesimpulan yaitu:

1. Capture to Translate adalah sebuah aplikasi translate dari bahasa cyrillic Rusia ke huruf latin Indonesia dengan basis android. Sistem dibangun menggunakan algoritma *find contour* dan dilanjutkan proses klasifikasi menggunakan metode random forest dengan penambahan nilai untuk parameter *regression accuracy* sehingga menghasilkan nilai akurasi lebih tinggi dari parameter default. Aplikasi ini menerima citra hasil dari akuisisi melalui media kamera ataupun akuisisi citra yang diambil dari galeri smartphone sebagai sebuah input sistem.
2. Dari hasil seluruh pengujian pada penelitian ini dapat disimpulkan bahwa nilai akurasi tertinggi dicapai pada citra dengan resolusi 6MP, jarak 30cm, sudut 0 derajat dan cahaya normal sebesar 13 lux. Pengujian dengan hasil akurasi tertinggi dicapai dengan nilai akurasi sebesar 93,33% pada 1 silabel, 85% pada 2 silabel dan 84,76% pada 3 silabel.

5.2 Saran

Berdasarkan dari hasil yang telah diperoleh dari penelitian ini, penulis memberikan beberapa saran untuk pengembangan riset selanjutnya.

1. Membangun aplikasi Capture to Translate untuk dapat mendeteksi dan menerjemahkan dari citra hasil tulisan tangan.
2. Menggunakan algoritma ekstraksi ciri lainnya, sehingga dapat menghasilkan hasil yang lebih akurat.
3. Menganalisis parameter-parameter pada algoritma random forest sehingga dapat memberikan hasil klasifikasi yang lebih akurat.
4. Mengembangkan aplikasi Capture to Translate ke aplikasi mobile lainnya seperti iOS dan Windows Phone.

DAFTAR PUSTAKA

[1] Gonzales, Rafael C. and Woods, Richard. 1993.Digital Image Processing.USA: Addison-Wesley Publishing Company.

[2] JIawei Han, Micheline Kamber.2002.Data Mining Concept and Techniques.Academic Press.

[3] Cheriet M., Kharma N., Liy C., Suen C.Y., 2006, “Character Recognition System A Guide for Student and Practioners’, John Willey & Sons. Inc.

[4] Suryo Hartanto, Aris Sugiharto (dkk), 2014, “Optical Character Recognition Menggunakan Algoritma Template Matching Correlation”, Jurnal Masyarakat Informatika, Volume 5, Nomor 9, Semarang.

[5] Rong Jia, Li Gang, Chen Yi-Ping P., 2009, Accoustic Feature Selection For Automatic Emotion Recognition From Speech, Information Processing and Management 45 (2009) 315–328.

[6] Imaduddin, Abdullah, 2014, “Pengenalan Karakter Huruf Hangul Korea Menggunakan Random Forest”, Telkom University, Bandung.

[7] Han, Jiawei., Kamber, Micheline., dan Pei, Jian. 2012. Data Mining Concepts and Techniques. Elsevier Inc

[8] Mahmud Yunus, 2010, “PERBANDINGAN METODE-METODE EDGE DETECTION UNTUK PROSES SEGMENTASI CITRA DIGITAL”, STMIK PPKIA Pradnya Paramita, Malang.

[9] Kenneth R. Castleman.1996.Digital Image Processing.Prentice Hall.

[10] Sofani, Rachmah, 2009, “Sistem OCR”, Institut Teknologi Telkom, Bandung.

[11] JIawei Han, Micheline Kamber.2002.Data Mining Concept and Techniques.Academic Press..

[12] Breiman, L., 2001, Random Forests, Machine Learning 45(1), 5-32.

[13] Ali, Jehad, 2012, “Random Forest and Decision Trees”, International Journal of Computer Science Issues, Vol. 9, Issue 5, No 3.

[14] Saputra, Wahyu S. J. (dkk), 2011, “Seleksi Fitur Menggunakan Random Forest Dan Neural Network”, Politeknik Elektronika Negeri Surabaya, Surabaya.

[15] Pitas, I., 1993. Digital Image Processing Algorithms, Prentice Hall . Singapore.