

**NORMALISASI UKURAN DAN BINERISASI INPUT IMAGE AKSARA
LAMPUNG PADA KANVAS ANDROID**

Skripsi

Oleh

**HAGAI ARITONANG
NPM 1617051024**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2021**

**NORMALISASI UKURAN DAN BINERISASI INPUT IMAGE AKSARA
LAMPUNG PADA KANVAS ANDROID**

Oleh

HAGAI ARITONANG

Skripsi

**Sebagai Salah Satu Syarat untuk Mencapai Gelar
SARJANA KOMPUTER**

Pada

**Jurusan Ilmu Komputer
Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2021**

ABSTRAK

NORMALISASI UKURAN DAN BINERISASI INPUT IMAGE AKSARA LAMPUNG PADA KANVAS ANDROID

Oleh

HAGAI ARITONANG

Aksara Lampung merupakan salah satu kekayaan budaya Lampung, tidak semua daerah memiliki aksaranya sendiri. Aksara Lampung menjadi salah satu aksara daerah yang ada di Indonesia yang harus dijaga kelestariannya, aksara Lampung saat ini mulai jarang digunakan dalam kehidupan masyarakat. Penelitian ini bertujuan membuat kanvas pada Android yang dapat menampung aksara Lampung yang dituliskan melalui sentuhan tangan pada layar *smartphone*. Dengan cara ini aksara Lampung akan terdigitalisasi dan dapat digunakan dalam bentuk digital. Aplikasi kanvas ini dibangun untuk melaksanakan *preprocessing* data gambar aksara Lampung. *Preprocessing* yang ditangani oleh aplikasi ini meliputi akuisisi data, normalisasi dengan interpolasi Bilinear dan binerisasi dengan metode Otsu. *Source code* aplikasi dikembangkan menggunakan Android Studio dengan bantuan library OpenCV dan aplikasi akhir dapat dijalankan pada perangkat Android. *Output* yang dapat dihasilkan dari aplikasi kanvas ini adalah citra RGB, grayscale, dan biner. Aplikasi ini merupakan solusi alternatif untuk pengumpulan data tulisan tangan aksara Lampung melalui perangkat Android. Aplikasi ini telah dapat digunakan untuk melakukan *preprocessing* gambar aksara Lampung dan digunakan untuk pengumpulan data aksara Lampung dengan skala terbatas.

Kata kunci : Aksara Lampung, *preprocessing*, library OpenCV, Android

ABSTRACT

NORMALISASI UKURAN DAN BINERISASI INPUT IMAGE AKSARA LAMPUNG PADA KANVAS ANDROID

By

HAGAI ARITONANG

Lampung script is one of Lampung cultures, not all regions have their script. The Lampung script is one of the regional scripts in Indonesia that must be preserved, the Lampung script is now rarely used in people's lives. This study aims to create a canvas on Android that can accommodate Lampung's script written by hand on a smartphone screen. In this way, the Lampung script will be digitized and can be used in digital form. This canvas application was built for preprocessing of Lampung script image data. Preprocessing handled by this application includes data acquisition, normalization with Bilinear interpolation, and binarization with the Otsu method. The source code of the application is developed using Android Studio with OpenCV library and the final application can be run on Android devices. The outputs that can be generated from this canvas application are RGB, grayscale, and binary images. This application is an alternative solution for collecting Lampung script handwriting data through Android devices. This application can be used to preprocessing Lampung script images and is used to collect Lampung script data with a limited scale.

Key words: Lampung script, preprocessing, OpenCV library, Android

Judul Skripsi : **NORMALISASI UKURAN DAN BINERISASI
INPUT IMAGE AKSARA LAMPUNG PADA
KANVAS ANDROID**

Nama Mahasiswa : **Hagai Aritonang**

Nomor Pokok Mahasiswa : **1617051024**


Program Studi : **Ilmu Komputer**

Fakultas : **Matematika dan Ilmu Pengetahuan Alam**




1. **Komisi Pembimbing**


Dr. rer. nat. Akmal Junaidi, M.Sc.
NIP 19710129 199702 1 001


Ardiansyah, S.Kom., M.Kom.
NIP 19870128 201803 1 001

2. **Ketua Jurusan Ilmu Komputer**


Didik Kurniawan, S.Si., M.T.
NIP 19800419 200501 1 004

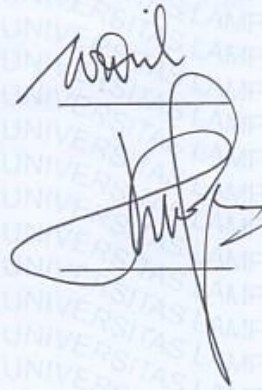
MENGESAHKAN

1. Tim Penguji

Ketua : **Dr. rer. nat. Akmal Junaidi, M.Sc.**



Sekretaris : **Ardiansyah, S.Kom., M.Kom.**



Anggota : **Dwi Sakethi, S.Si., M.Kom.**

2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam



Dr. Eng. Suripito Dwi Yuwono, S.Si., M.T.

NIP 19740705 200003 1 001

Tanggal Lulus Ujian Skripsi : **28 Juni 2021**

PERNYATAAN

Saya yang bertanda tangan dibawah ini, menyatakan bahwa skripsi saya yang berjudul “**Normalisasi Ukuran dan Binerisasi *Input Image* Aksara Lampung pada Kanvas Android**” merupakan hasil pekerjaan saya sendiri dan bukan pekerjaan orang lain. Semua tulisan yang tertuang dalam skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Apabila di kemudian hari terbukti bahwa skripsi ini merupakan hasil salinan atau dikerjakan oleh orang lain, maka saya bersedia menerima sanksi sesuai dengan ketentuan akademik yang berlaku.

Bandar Lampung, 29 Juni 2021



Hagai Artonang
NPM 1617051024

RIWAYAT HIDUP

Penulis dilahirkan di Jakarta pada tanggal 16 Juli 1998, sebagai anak ketiga dari empat bersaudara dari bapak T.M. Ompusunggu dan ibu Rismauli Napitupulu.

Penulis menyelesaikan pendidikan tingkat Sekolah Dasar (SD) di SDN 09 Mekarsari di tahun 2010, Sekolah Menengah Pertama (SMP) di SMP PGRI Tambun Selatan pada tahun 2013, dan Sekolah Menengah Atas (SMA) di SMAS MAHANAIM mengambil jurusan Ilmu Pengetahuan Alam (IPA) pada tahun 2016.

Pada tahun 2016 penulis terdaftar sebagai mahasiswa Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung melalui jalur Seleksi Nasional Masuk Perguruan Tinggi Negeri (SNMPTN). Selama menjadi mahasiswa beberapa kegiatan telah dilakukan penulis antara lain.

- 1 Menjadi Asisten Dosen untuk mata kuliah “Keamanan Sistem Informasi” pada periode 2017/2018.
- 2 Menjadi Sekertaris Divisi Hubungan Masyarakat Unit Kegiatan Mahasiswa Kristen (UKM-Kristen) Universitas Lampung pada periode 2018.
- 3 Pada bulan Desember 2018 sampai dengan bulan Februari 2019 penulis melaksanakan Kerja Praktik di Balai Karantina Pertanian Kelas I Bandar Lampung.
- 4 Menjadi Ketua Divisi Hubungan Masyarakat Unit Kegiatan Mahasiswa Kristen (UKM-Kristen) Universitas Lampung pada periode 2019.

PERSEMBAHAN

Skripsi ini saya persembahkan untuk Bapak dan Mamak yang telah memberikan setiap perhatian dan pelajaran kehidupan yang telah diajarkan baik secara verbal maupun non verbal. Serta untuk abang, kaka, dan adik kandung penulis sebagai penyokong dana kuliah.

Terima kasih teman-teman Ilmu Komputer yang selalu mendukung dan berbagi cerita, serta Almamater hijau.

UNIVERSITAS LAMPUNG

SANWACANA

Puji syukur penulis tuliskan kepada Tuhan Yesus Kristus untuk segala berkat, karunia, kesempatan, dan anugerah-Nya sehingga penulis dapat menyelesaikan skripsi dengan judul “Normalisasi Ukuran dan Binerisasi *Input Image* Aksara Lampung pada Kanvas Android”.

Penulis mengucapkan terima kasih kepada semua pihak yang telah membantu penulis dalam penyusunan skripsi ini, yaitu:

1. Bapak Dr. rer. nat. Akmal Junaidi, M.Sc. selaku pembimbing utama yang telah memberikan tema penelitian, memberikan bimbingan, pengertian, kritik, saran, dan solusi serta bimbingan sehingga dapat menyelesaikan skripsi ini.
2. Bapak Ardiansyah, S.Kom., M.Kom. selaku pembimbing kedua yang telah memberikan *key word* pencarian mengenai skripsi, pengertian, kritik, saran, dan solusi serta bimbingan bapak sehingga skripsi dapat diselesaikan dengan baik.
3. Bapak Dwi Sakethi, S.Si., M.Kom. selaku pembahas yang telah memberikan cara berpikir dengan logika yang baik, kritik, saran, dan revisi serta bimbingan bapak sehingga skripsi dapat diselesaikan.
4. Bapak Dr.Eng.Admi Syarif selaku pembimbing akademik yang telah memberikan saran dan masukan untuk perkuliahan yang dijalani.
5. Taruna Pemuda Meksa Bersatu selaku teman yang kurang memiliki rasa malu dan nalar yang berlebih, sebagai teman yang dapat bercanda dalam hal apapun, dan tempat untuk menampilkan kebodohan.

6. Jemaat GKKL yang memberikan fasilitas berupa tempat untuk berkumpul dan berpikir mengenai hal liar dan menarik serta mendorong untuk terus melanjutkan kehidupan.
7. Anggota Kriskat 2016 selaku perkumpulan manusia yang tersesat yang dapat memberikan cerita yang berbeda.
8. Kawan POMMIPA yang telah menjadi tempat untuk bertumbuh dan berbagi kebaikan.
9. Bala pengurus UKM-Kristen yang telah memberikan pelajaran bertahan hidup, mencari dana, tempat, teman, dan pengalaman yang bisa ditertawakan bersama.
10. Personel BIC yang telah memberikan panggung untuk menampilkan rasa percaya diri yang terlalu tinggi.
11. Stackoverflow yang telah menjadi penunjuk jalan bagi penulis yang tersesat dan tidak tahu arah ke depan serta menemukan jawaban untuk permasalahan skripsi.
12. Bapak dan Ibu Dosen Jurusan Ilmu Komputer yang telah memberikan ilmu serta pengalaman hidup yang berharga selama penulis menjadi mahasiswa.
13. Teman-teman Angkatan 2016 yang selalu membantu penulis selama perkuliahan serta kakak tingkat dan adik tingkat yang penulis tidak dapat disebutkan satu per satu.
14. Almamater hijau, Universitas Lampung yang sudah memberikan kenyamanan saat penulis memakainya di kampus dan pengalaman berharga.

Bandar Lampung, Juni 2021

Hagai Aritonang

DAFTAR ISI

	Halaman
DAFTAR TABEL	iv
DAFTAR GAMBAR.....	v
I. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian.....	3
1.5 Manfaat penelitian	3
II. TINJAUAN PUSTAKA	4
2.1 Aksara Lampung	4
2.2 Android.....	7
2.3 SDK (<i>Software Development Kit</i>)	8
2.4 Android Studio	8
2.5 Desain	9
2.6 Desain <i>User Interface</i> (UI).....	9
2.7 Citra	9
2.8 Citra Berwarna.....	10

2.9	Citra <i>Grayscale</i>	11
2.10	Citra Biner	11
2.11	<i>Character Recognition</i>	12
2.12	Akuisisi Gambar	12
2.13	<i>Preprocessing</i>	12
2.14	Binerisasi	13
2.15	Metode Niblack	13
2.16	Metode Sauvola	13
2.17	Metode Otsu	14
2.18	Interpolasi Citra	14
2.19	Interpolasi Nearest Neighbor.....	14
2.20	Interpolasi Bilinear	15
2.21	Interpolasi Bicubic	15
2.22	<i>Library OpenCV</i>	16
III.	METODOLOGI PENELITIAN	17
3.1	Waktu dan Tempat Pelaksanaan.....	17
3.2	Alat Pendukung	17
3.3	Proses Pembuatan Aplikasi	18
3.4	Membuat Area Tulisan	18
3.5	Menampilkan Aksara	19
3.6	Menangkap Gambar	20
3.7	Normalisasi Gambar	20
3.8	Binerisasi Gambar	20
3.9	Perancangan Aplikasi	21
IV.	HASIL DAN PEMBAHASAN	24
4.1	Kanvas	24

4.2	Menangkap Gambar	26
4.3	Layar Aksara	28
4.4	Normalisasi Gambar	29
4.5	<i>Resize</i> Gambar	32
4.6	Binerisasi Gambar	34
V.	SIMPULAN DAN SARAN	38
5.1	Simpulan.....	38
5.2	Saran	38
	DAFTAR PUSTAKA	40

DAFTAR TABEL

Tabel	Halaman
1. Perbandingan Ukuran <i>File</i> Ekstensi Gambar	27
2. Perbandingan Ukuran <i>File</i> Interpolasi	32
3. Perbandingan Ukuran <i>File Thresholding</i>	34

DAFTAR GAMBAR

Gambar	Halaman
1. Karakter dasar aksara Lampung (Junaidi, 2016).....	5
2. Diakritik atas (Junaidi, 2016).....	5
3. Diakritik bawah (Junaidi, 2016).....	6
4. Diakritik kanan (Junaidi, 2016).....	6
5. Tanda baca aksara Lampung (Junaidi, 2016).....	7
6. Tahap pembuatan aplikasi.....	18
7. Area tulisan dan layar aksara.	19
8. <i>Splash screen</i>	21
9. Halaman utama.....	22
10. Potongan <i>source code</i> pada objek Paint.....	24
11. Potongan <i>source code</i> pada MotionEvent.....	25
12. Perbandingan kualitas ekstensi gambar.	27
13. Potongan <i>source code</i> pada normalisasi gambar.	29
14. Metode pengambilan kontur.	30
15. Gambar aksara RGB sebelum normalisasi (a) gambar <i>grayscale</i> setelah normalisasi (b).....	31
16. Perbandingan kualitas gambar interpolasi.	32
17. Potongan <i>source code</i> pada <i>resize</i> gambar.....	33
18. Perbandingan kualitas gambar <i>thresholding</i>	35
19. Gambar aksara sebelum binerisasi dalam citra <i>grayscale</i> (a) gambar hasil binerisasi (b).....	36

20. Potongan <i>source code</i> pada binerisasi gambar.....	36
21. Gambar RGB awal 656 x 766 piksel (a) menjadi citra <i>grayscale</i> 430 x 393 piksel (b) hasil citra biner 200 x 200 piksel (c).	37

I. PENDAHULUAN

1.1 Latar Belakang

Aksara Lampung atau yang disebut juga sebagai aksara Kaganga merupakan salah satu kekayaan budaya Lampung yang digunakan sebagai media untuk memberikan informasi. Aksara Lampung berupa tulisan tangan yang sampai saat ini masih dipraktikkan oleh sebagian masyarakat Lampung. Aksara Lampung terdiri dari 20 jenis aksara dasar dan 12 anak aksara.

Perkembangan teknologi di dunia telah membuat hidup manusia menjadi semakin mudah. Teknologi perangkat *mobile* saat ini sangat populer dikarenakan perangkat tersebut mudah untuk dioperasikan serta mudah dalam mobilisasi. *Smartphone* merupakan teknologi perangkat *mobile* yang dimiliki oleh banyak pengguna dengan segala manfaatnya seperti untuk membantu pekerjaan harian, sebagai media hiburan bermain *game* atau menonton video. Indonesia merupakan salah satu negara pengguna *smartphone* terbesar di dunia, Indonesia memiliki 92 juta pengguna *smartphone* pada tahun 2019 (eMarketer, 2019).

Android merupakan sistem operasi *mobile phone open source* yang berbasis Linux menggunakan bahasa pemrograman Java. Android merupakan sistem operasi pada *smartphone* yang mudah untuk dioperasikan dan digemari oleh banyak pengguna, terdapat banyak aplikasi pada Android yang banyak membantu untuk kebutuhan maupun pekerjaan. Android juga telah banyak digunakan pada berbagai bidang seperti bidang pendidikan, seni, hiburan, dan bisnis.

Sebagai salah satu warisan budaya Masyarakat Lampung, Aksara Lampung pada saat ini sudah mulai ditinggalkan oleh masyarakat Lampung yang kemudian akan menyebabkan punahnya aksara Lampung. Aksara Lampung mulai tergerus oleh perkembangan zaman dan semakin jarang digunakan oleh masyarakat sehingga diperlukan adanya upaya-upaya untuk melestarikan aksara Lampung agar dapat terus lestari serta dapat dipelajari oleh masyarakat. Adapun beberapa cara yang telah dilakukan seperti mendokumentasikan aksara Lampung, digitalisasi aksara Lampung oleh Junaidi, (2016) dalam penelitiannya yang berjudul *Lampung Handwritten Character Recognition* dan penelitian lain yang dilakukan oleh (Hara dkk., 2016) dalam pembuatan sistem pengenalan aksara Lampung berbasis desktop, untuk memaksimalkan cara melestarikan aksara Lampung tersebut maka dilakukan penelitian *preprocessing* pengenalan aksara Lampung berbasis Android.

Penelitian ini akan membangun aplikasi berbasis Android untuk tahap *preprocessing* pengenalan aksara Lampung. Dalam sistem ini, pengguna diharuskan memberi masukan berupa tulisan aksara Lampung dengan sentuhan tangan pada layar *smartphone* pada kanvas yang disediakan. Proses *preprocessing* yaitu normalisasi ukuran dan binerisasi pada *input* gambar saat pengguna selesai menuliskan aksara pada aplikasi menggunakan *library* OpenCV *open source*.

1.2 Rumusan Masalah

Rumusan masalah pada penelitian ini yaitu bagaimana mendesain dan mengimplementasikan aplikasi yang dapat melakukan tahap *preprocessing input* tulisan aksara sehingga aksara Lampung yang dituliskan pengguna melalui *smartphone* dapat digunakan oleh aplikasi pada proses selanjutnya.

1.3 Batasan Masalah

Pada penelitian ini terdapat batasan masalah sebagai berikut :

1. Aplikasi mampu melakukan tahap *preprocessing* agar keluarannya dapat diproses untuk tahap selanjutnya.
2. Aplikasi dapat menampung 5 aksara Lampung untuk sekali proses.
3. Keluaran aplikasi ini adalah representasi gambar yang disimpan dalam bentuk *file* gambar dengan citra RGB, *grayscale*, dan biner menggunakan ekstensi PNG.

1.4 Tujuan Penelitian

Tujuan dari dilakukannya penelitian ini adalah untuk melakukan proses *preprocessing* pada gambar berupa normalisasi ukuran dan binerisasi gambar aksara Lampung yang telah dituliskan dengan sentuhan jari di layar *smartphone* sebanyak 5 karakter.

1.5 Manfaat penelitian

Manfaat dari penelitian ini diharapkan dapat :

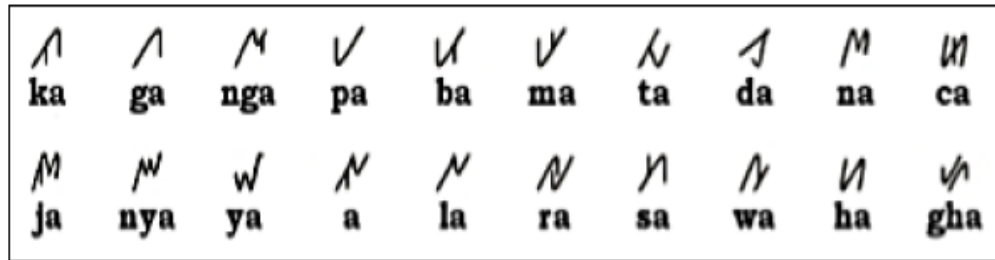
1. Melestarikan kebudayaan Lampung dalam format digital.
2. Mengedukasi masyarakat agar dapat mempelajari aksara Lampung.
3. Memperkenalkan aksara Lampung pada masyarakat umum.

II. TINJAUAN PUSTAKA

2.1 Aksara Lampung

Serupa dengan suku lainnya di Indonesia, Lampung memiliki abjad atau aksaranya sendiri yang dikenal dengan aksara Lampung. Aksara Lampung merupakan tulisan aksara perkembangan dari aksara Devanagari yang berasal dari India Selatan. Aksara Lampung yang disebut juga aksara Ka ga nga memiliki 20 jenis aksara dasar dan 12 anak aksara (diakritik). Diakritik melekat pada induk aksara akan menghasilkan jenis cara membaca yang berbeda-beda. Terdapat jenis kombinasi cara membaca aksara Lampung sebanyak 560 suku kata (Aryantio dan Munir, 2015).

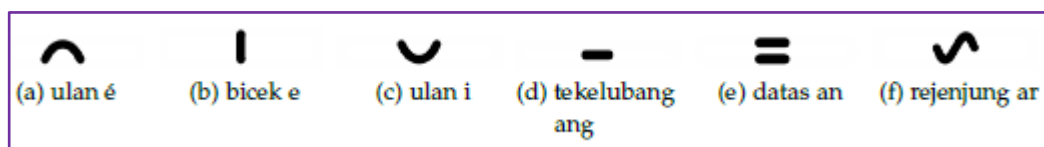
Aksara Lampung pada dasarnya memiliki kurva untuk setiap aksara atau dengan lengkungan yang menghadap ke atas maupun ke bawah. Aksara Lampung terdiri dari satu simbol yang membentuk aksara, kecuali aksara “ha” dan “ga” yang terdiri dari dua simbol yang berpasangan sehingga membentuk satu aksara yang terdapat pada Gambar 1. Aksara Lampung cenderung mengandung unsur zig-zag pada setiap aksara. Posisi penulisan pada aksara Lampung juga terlihat tidak tegak lurus melainkan miring mengarah ke kanan atas atau dapat dikatakan aksara Lampung memiliki orientasi diagonal. Pada aksara Lampung struktur tulisan bahasa Arab juga turut mempengaruhi sistem penulisan pada aksara Lampung, pengaruh tulisan Arab ini terdapat pada sistem aksara Lampung yang menggunakan anak huruf atau disebut juga diakritik.



Gambar 1. Karakter dasar aksara Lampung (Junaidi, 2016).

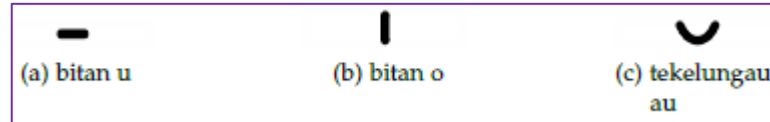
Setiap satu induk huruf, aksara Lampung memiliki pasangan berupa sebuah huruf konsonan dengan sebuah huruf vokal a dalam satu kesatuan, sehingga semua huruf akan diakhiri oleh huruf vokal a. Pada aksara Lampung diakritik berfungsi untuk mengubah bunyi pelafalan huruf induk. Diakritik diletakkan di sekitar induk huruf, diakritik aksara Lampung terbagi menjadi 3 diakritik berdasarkan posisi diakritik terhadap induk aksara tersebut yaitu, diakritik atas, diakritik bawah, dan diakritik kanan.

Diakritik atas merupakan diakritik yang berada di atas induk aksara yang terdiri dari enam karakter yang memiliki nama atau istilah masing-masing seperti pada Gambar 2. Terdapat tiga dari enam diakritik yang memiliki pelafalan yang sama dengan huruf vokal yang digunakan dalam alfabet. Diakritik ulan i yang menggantikan huruf vokal pada karakter dasar menjadi huruf vokal i. Diakritik ulan (e) yang dilafalkan [ə] seperti pada kata “pertandingan itu berakhir seri (sêri)”. Diakritik bicek (é) yang dilafalkan [e] seperti pada kata “Anak-anak bermain di teras (téras)”. Diakritik tekelubang yang memiliki arti karakter -ang. Datas yang memiliki arti karakter -an. Diakritik rejunjung yang memiliki arti karakter -ar yang berfungsi untuk menempa huruf vokal yang melekat pada aksara induk berupa ekspansi huruf konsonan “ng”, “n”, dan “r” di akhir vokal.



Gambar 2. Diakritik atas (Junaidi, 2016).

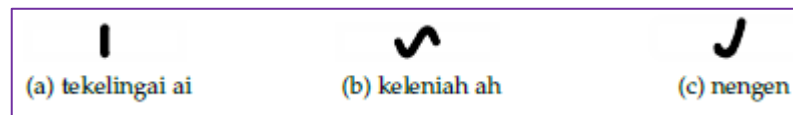
Diakritik bawah merupakan diakritik yang berada di bawah induk aksara. Diakritik bawah terdiri dari tiga diakritik dengan dengan simbol masing-masing. Bentuk dari diakritik bawah dapat dilihat pada Gambar 3.



Gambar 3. Diakritik bawah (Junaidi, 2016).

Diakritik bitan u dan bitan o memiliki nama yang sama yaitu bitan, namun terdapat perbedaan simbol dan fungsi. Bitan u berupa satu garis lurus secara horizontal yang berada di bawah induk aksara sedangkan bitan o berupa satu garis lurus secara vertikal yang berada di bawah induk aksara. Bitan u berfungsi mengubah akhiran karakter vokal a menjadi karakter vokal u seperti pada aksara “la” (L) bila ditambahkan diakritik bitan u akan menjadi “lu” (L). Bitan o yang berfungsi mengubah akhiran karakter vokal a menjadi karakter vokal o seperti pada aksara “ta” (T) bila ditambahkan diakritik bitan o akan menjadi “to” (T). Diakritik tekelungau dengan simbol kurva terbuka yang berfungsi mengubah akhiran karakter vokal a menjadi huruf diftong -au seperti pada aksara “ja” (J) bila ditambahkan diakritik tekelungau au menjadi “jau” (J).

Diakritik kanan merupakan diakritik yang berada pada kanan aksara induk. Diakritik kanan terdiri dari tiga diakritik yaitu tekelingai -ai, keleniah -ah, dan nengen. Diakritik kanan terdapat pada Gambar 4.

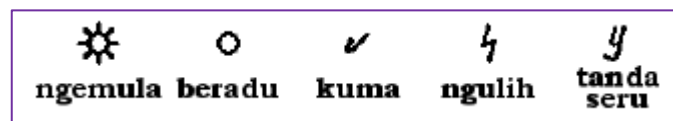


Gambar 4. Diakritik kanan (Junaidi, 2016).

Diakritik tekelingai -ai berfungsi untuk merubah akhiran karakter a menjadi diftong -ai seperti pada kata “ramai” (RV) yang terjadi perubahan pelafalan dari kata “rama” setelah diberikan diakritik tekelingai -ai. Diakritik keleniah -ah yang berfungsi merubah akhiran karakter vokal a menjadi karakter -ah

seperti pada kata “rama” (\mathcal{NV}) yang berubah pelafalan menjadi kata “ramah” ($\mathcal{NV}h$) setelah penambahan diakritik keleniah -ah pada aksara serta. Diakritik nengen yang berfungsi untuk membungkam huruf vokal (*vocal silencer*) yang melekat pada karakter dasar sehingga bagian yang tersisa hanyalah konsonan dari karakter dasar (Junaidi, 2016).

Aksara Lampung juga memiliki tanda baca seperti sistem penulisan pada umumnya. Terdapat 5 tanda baca seperti pada Gambar 5.



Gambar 5. Tanda baca aksara Lampung (Junaidi, 2016).

Tanda baca unik ngemula yang berbentuk seperti matahari berfungsi untuk memulai sebuah kalimat. Tanda baca beradu yang berbeda dengan tanda baca ngemula, tanda baca beradu berfungsi untuk menyelesaikan kalimat dan berada di akhir kalimat. Tanda baca kuma sama seperti koma pada aksara latin yang berfungsi sebagai penjeda kalimat. Tanda baca ngulih yang berfungsi sebagai tanda baca tanya pada kalimat serta tanda baca seru yang sama seperti aksara latin berupa tanda baca seru yang berfungsi untuk memberikan perintah ataupun untuk menegaskan suatu kalimat (Junaidi, 2016).

2.2 Android

Android adalah sebuah sistem operasi untuk perangkat *mobile* berbasis linux yang mencakup sistem operasi, *middleware*, dan aplikasi. Android menyediakan *platform* yang terbuka bagi para pengembang untuk menciptakan aplikasi mereka. Android merupakan generasi baru *platform mobile*, *platform* yang memberikan pengembang untuk melakukan pengembangan sesuai dengan yang diharapkannya. Sistem operasi yang mendasari Android dilisensikan di bawah GNU, *General Public Lisensi Versi 2 (GPLv2)*, yang sering dikenal dengan istilah "*copyleft*" lisensi di mana setiap perbaikan pihak ketiga harus

terus jatuh di bawah *terms*. Android didistribusikan di bawah Lisensi Apache *Software* (ASL/Apache2), yang memungkinkan untuk distribusi kedua dan seterusnya (Safaat, 2011).

2.3 SDK (*Software Development Kit*)

Android SDK adalah sekumpulan alat bantu dan program perangkat lunak berupa *debugger*, *software libraries*, emulator, dokumentasi, sampel kode, *tutorial*, dan API. SDK diperlukan untuk mengembangkan aplikasi pada *platform* Android menggunakan bahasa pemrograman Java. Sebagai *platform* aplikasi netral, Android memberi kesempatan bagi semua orang untuk membuat aplikasi yang dibutuhkan, yang bukan merupakan aplikasi bawaan *Smartphone* (Safaat, 2011).

2.4 Android Studio

Android Studio merupakan *Integrated Development Environment* (IDE) yang digunakan dalam pengembangan aplikasi berbasis Android. Android Studio merupakan pengembangan yang terintegrasi secara resmi khusus untuk pengembangan Android dan pertama kali diumumkan pada tahun 2013 di konferensi Google I/O. Android Studio tersedia dibawah lisensi 2.0, sebelumnya Android Studio dipratinjau dari versi 0.1 pada Mei 2013 dan kemudian memasuki tahap beta versi 0.8 yang keluar pada tahun 2014. Android Studio yang pertama kali stabil keluar pada bulan Desember 2014 yang dimulai dari versi 1.0 berdasarkan JetBrains IntelliJ IDEA. Android Studio tersedia di berbagai Sistem Operasi seperti windows, Mac OS X, dan Linux. Android Tool (ADT) merupakan IDE utama Google untuk pengembangan aplikasi Android (Rojatkar dkk., 2016).

2.5 Desain

Desain bisa digunakan sebagai kata benda maupun kata kerja. Sebagai kata kerja, desain memiliki arti proses untuk membuat dan menciptakan objek baru. Sebagai kata benda, desain digunakan untuk menyebut hasil akhir dari sebuah proses kreatif, baik itu berwujud sebuah rencana, proposal, atau berbentuk objek nyata (Fernando, 2013). Desain merupakan kegiatan kreatif untuk merencanakan dan merancang sesuatu yang umumnya fungsional dan tidak ada sebelumnya dalam rangka menyelesaikan suatu masalah tertentu agar memiliki nilai lebih dan menjadi lebih bermanfaat bagi penggunanya.

2.6 Desain *User Interface* (UI)

Desain *User Interface* (UI) adalah cara program dan pengguna untuk berinteraksi. *User Interface* memiliki fungsi untuk menghubungkan atau menerjemahkan informasi antara pengguna dengan sistem operasi, sehingga perangkat dapat digunakan (Lastiansah, 2012). Desain *interface* merupakan desain untuk komputer, peralatan, mesin, perangkat lunak, dan situs web yang berfokus pada pengalaman dan interaksi penggunanya dengan tujuan untuk membuat interaksi pengguna sesederhana dan seefisien mungkin. Desain *interface* ini sangat penting dikarenakan akan sangat berpengaruh pada user dalam menggunakan komputer, aplikasi maupun website, apabila suatu program sulit untuk digunakan maka hal ini akan memaksa pengguna untuk melakukan suatu kesalahan dan program tersebut tidak berjalan sesuai dengan semestinya.

2.7 Citra

Citra merupakan gambar berdimensi 2 sebagai representasi dari suatu objek. Sebuah citra diartikan sebagai suatu fungsi kontinu dalam dua dimensi dari intensitas cahaya (x,y) (Himamunanto dan Widiarti, 2013). Dari sudut pandang

matematika citra adalah fungsi kontinu dari intensitas cahaya pada bidang dua dimensi, di dalam sebuah citra mengandung banyak informasi yang sering mengalami derau (*noise*) mengakibatkan informasi yang diperoleh dari citra tersebut menjadi kurang akurat. Derau (*noise*) yang terjadi misalnya warna yang terlalu kontras, kurang tajam, kabur (*blurring*), dan sebagainya. Informasi yang terkandung dalam citra adalah warna, tekstur, bentuk, dan ukuran sehingga mata manusia dapat menganalisis informasi tersebut dengan tujuan yang diinginkan. Informasi pada citra didapatkan dengan dua cara yaitu diolah secara langsung tanpa proses terlebih dahulu untuk mendapatkan informasinya dan yang harus diolah terlebih dahulu dengan kata lain memerlukan beberapa proses pengolahan untuk mendapatkan informasi dari suatu citra tersebut.

Secara umum pengolahan citra digital menunjuk pada pemrosesan gambar dimensi dua menggunakan komputer. Pada konteks yang lebih luas, pengolahan citra digital mengacu pada pemrosesan setiap data berdimensi dua. Citra digital merupakan sebuah larik (*array*) yang berisi nilai-nilai *real* maupun kompleks yang direpresentasikan dengan deretan bit tertentu (Putra, 2010).

2.8 Citra Berwarna

Citra berwarna adalah citra digital yang setiap pikselnya mengandung informasi warna. Citra berwarna atau citra RGB, merupakan jenis citra yang menyajikan warna dalam bentuk komponen R (merah), G (hijau), dan B (biru) (Kadir dan Sudanto, 2012). Setiap warna memiliki jumlah bit sebanyak delapan yang berarti nilai pikselnya berkisar antara 0 sampai dengan 255. Dengan kemungkinan warna yang dapat disajikan mencapai $255 \times 255 \times 255$. Untuk warna merah memiliki nilai piksel sebesar $255 \times 0 \times 0$, untuk warna hijau memiliki nilai piksel sebesar $0 \times 255 \times 0$, dan untuk warna biru memiliki nilai piksel sebesar $0 \times 0 \times 255$.

2.9 Citra *Grayscale*

Citra *grayscale* merupakan citra yang setiap pikselnya hanya mengandung komponen warna *luminance*, umumnya Citra *grayscale* dikodekan 8 bit yang berarti memiliki skala keabuan dari 0 sampai 255, 0 merepresentasikan warna hitam dan 255 merepresentasikan warna putih (Madenda, 2015). Jenis citra ini memiliki satu jenis warna matriks yaitu abu-abu dengan nilai piksel yang berbeda-beda. Operasi *grayscale* bertujuan untuk merubah citra 24 bit RGB menjadi citra abu-abu. Pemilihan pemrosesan pada tingkat abu-abu ini dikarenakan lebih sederhana. Operasi *grayscale* sangat bermanfaat dikarenakan cukup signifikan untuk menyimpan detail gambar serta operasi *grayscale* yang hanya terdiri dari satu *channel* akan meringankan beban komputasi untuk memproses suatu gambar (Junaidi dkk., 2020).

2.10 Citra Biner

Citra biner adalah citra yang setiap pikselnya hanya mengandung dua level piksel yaitu 0 untuk warna hitam, dan 1 untuk warna putih, sehingga citra biner hanya memerlukan satu bit untuk mewakili setiap piksel yang ada. Citra biner juga disebut sebagai citra B&W (*Black and White*), atau citra monokrom. Citra biner sering kali muncul sebagai hasil dari proses pengolahan seperti segmentasi, pengambangan, morfologi ataupun *dithering* (Putra, 2010). Citra biner dihitung dengan menggunakan *threshold* (nilai ambang), jika suatu piksel nilainya dibawah batas ambang maka piksel tersebut akan diubah menjadi 0, jika nilai suatu pikselnya di atas batas nilai ambangnya maka piksel tersebut akan diubah menjadi 1. Jenis citra ini memiliki satu jenis matriks bertipe *logical* matriks yang berarti hanya memiliki dua nilai jika tidak bernilai A maka bernilai B.

2.11 *Character Recognition*

Character Recognition merupakan kemampuan komputer untuk mengenali karakter dari masukan berupa dokumen teks pada kertas, foto, layar sentuh dan perangkat lainnya (Uluwiyah dkk., 2013). Terdapat dua macam *Character Recognition*, yaitu *Offline* dan *Online Character Recognition*. Sistem *online* yaitu mengolah data atau dokumen yang dibuat langsung berupa inputan dengan bantuan alat optik berupa sensor, layar, dan pena yang nantinya digunakan untuk menulis setiap karakter. Sedangkan sistem *offline* menggunakan scan gambar tulisan tangan sebagai objek citra yang akan diteliti. Berdasarkan metode penulisannya, terdapat jenis pengenalan karakter dengan input berupa tulisan tangan yang disebut *Handwritten Character Recognition* (HCR) dan input berupa aksara cetak atau *Printed Character Recognition*.

2.12 **Akuisisi Gambar**

Akuisisi gambar adalah tahap pertama dalam *Character Recognition*. Akuisisi gambar merupakan tahapan yang bertujuan untuk mengubah data gambar optik menjadi sebuah *array* data numerik yang dimanipulasi oleh komputer dalam pemrosesan suatu gambar (Mishra dkk., 2017). Akuisisi gambar merupakan metode untuk mendapatkan gambar melalui kamera atau pemindai. Gambar yang telah diakuisisi berupa format *file* yang ditentukan, misalnya JPEG atau PNG.

2.13 *Preprocessing*

Preprocessing merupakan bagian dari pada proses pengenalan pola yang berguna untuk mempersiapkan citra atau gambar sehingga sesuai dengan kebutuhan. Salah satu hal yang dilakukan pada *preprocessing* misalnya mengubah suatu citra menjadi citra *grayscale* (Keabu-abuan) (Yodha dan Kurniawan, 2014). *Preprocessing* juga melakukan proses untuk merubah

ukuran suatu citra menjadi piksel yang lebih kecil agar dapat terlihat pola dari suatu citra atau disebut juga sebagai *resize* dengan beberapa metode seperti interpolasi Bilinear dan interpolasi Bicubic serta melakukan binerisasi pada citra tersebut yang dapat dilakukan dengan beberapa metode seperti Sauvola, Otsu, dan Niblack.

2.14 Binerisasi

Binerisasi adalah tahap di mana *file* citra digital dikonversi menjadi citra biner. Citra biner merupakan citra yang hanya memiliki dua derajat keabuan, yaitu hitam dan putih. Piksel-piksel objek bernilai 1 yang berwarna putih dan piksel-piksel latar belakang bernilai 0 yang berwarna hitam. Dengan kata lain pada citra biner, latar belakang berwarna putih sedangkan objek berwarna hitam (Munir, 2004).

2.15 Metode Niblack

Metode ini merupakan metode binerisasi dengan tipe *local threshold*. Metode Niblack merupakan algoritma yang digunakan untuk menentukan nilai ambang batas (*threshold*) lokal untuk setiap piksel citra. Algoritma Niblack menghitung batas *piksel wise* dengan menggeser jendela persegi pada citra *grayscale*. Nilai *threshold* dihitung dengan menggunakan rata-rata atau *mean* dan standar deviasi dari semua piksel dalam jendela (Niblack, 1986).

2.16 Metode Sauvola

Metode Sauvola merupakan pengembangan dari metode Niblack. Algoritma Sauvola merupakan modifikasi dari algoritma Niblack yang diklaim mampu memberikan peningkatan kinerja pada citra yang berlatar belakang tekstur ringan, variasi yang banyak, dan pencahayaan tidak merata (Sauvola dan

Pietikainen, 1999) Metode Sauvola tidak bisa mengatasi kontras yang rendah, dan *local window* juga masih ditentukan secara manual.

2.17 Metode Otsu

Metode Otsu merupakan metode global *thresholding* yang mengklasifikasikan antara latar depan dan latar belakangnya dari nilai *thresholding* yang diambil dari variasi keseluruhan gambar yang dijadikan kelas-kelas sehingga ditemukan nilai bobot tiap kelasnya. Pendekatan yang dilakukan oleh metode Otsu yaitu dengan analisis diskriminan yang menentukan suatu variabel sehingga dapat membedakan antara dua atau lebih kelompok yang muncul secara alami. Untuk memisahkan objek dengan latar belakang analisis diskriminan akan memaksimumkan variabel tersebut (Putra, 2004).

2.18 Interpolasi Citra

Interpolasi adalah proses pembuatan ulang (*resample*) piksel baru dari piksel yang telah diketahui dengan suatu fungsi tertentu. Menambah ukuran dari suatu citra disebut *upsampling*, sedangkan mengurangi ukurannya disebut *down sampling* (Sachs, 2001). Interpolasi citra digital bekerja secara dua arah. Proses ini berusaha untuk mendapatkan perkiraan nilai piksel warna dan intensitas yang terbaik berdasarkan nilai pada piksel-piksel di sekitarnya.

2.19 Interpolasi Nearest Neighbor

Interpolasi ini merupakan metode paling sederhana dan pada dasarnya digunakan untuk membuat piksel menjadi lebih besar. Metode ini paling sederhana, karena memanfaatkan teknik replikasi piksel. Metode ini baik digunakan apabila warna piksel yang diinginkan tidak berubah, namun metode ini kurang baik jika digunakan untuk memperbesar gambar foto (Tinaliah dan

Triana, 2019). Metode ini baik digunakan karena kemudahan dan kecepatan eksekusinya, namun metode ini kurang baik untuk memperbesar gambar foto dikarenakan tidak adanya proses *smoothing* yang membuat gambar memiliki efek catur atau kotak-kotak pada gambar.

2.20 Interpolasi Bilinear

Interpolasi Bilinear adalah interpolasi dengan menentukan nilai piksel baru berdasarkan rata-rata, dengan memberi bobot dari 4 piksel dari ukuran 2x2 piksel tetangga terdekat dalam gambar asli. Nilai piksel dengan jarak terdekat dengan piksel yang baru mempunyai bobot yang lebih besar dibanding nilai-nilai piksel lainnya dengan jarak yang agak jauh (Tinaliah dan Triana, 2019). Gambar yang dihasilkan menggunakan interpolasi Bilinear akan tampak lebih *soft* atau sedikit *blur* sehingga gambar akan terlihat sama seperti gambar sebelum dilakukan proses interpolasi.

2.21 Interpolasi Bicubic

Interpolasi Bicubic adalah interpolasi dengan metode yang lebih canggih dan hasilnya lebih halus pada bagian tepi-tepinya daripada interpolasi Bilinear. Interpolasi Bicubic menggunakan 4x4 piksel atau dengan kata lain 16 piksel tetangga untuk mendapatkan informasi nilai dari tingkat keabuan tiap piksel. Interpolasi Bicubic dapat memberikan hasil penskalaan yang lebih baik dan menghasilkan gambar yang terasa lebih tajam karena mengambil sampel piksel yang lebih banyak dibandingkan metode interpolasi lain (Tinaliah dan Triana, 2019).

2.22 *Library OpenCV*

Library OpenCV secara resmi diperkenalkan pada tahun 1999 oleh Intel Research. Sebelumnya OpenCV ditulis dalam bahasa C. Namun sejak versi 2.0, OpenCV menyertakan *interface* pada bahasa C dan C++. *Library OpenCV* yang pertama kali dibangun untuk digunakan dalam sistem operasi Android dimulai pada versi 2.2. Kemudian dirilis versi OpenCV terbaru pada bulan Agustus 2011 yaitu OpenCV 2.3.1 (beta2). OpenCV versi 2.3.1 yang digunakan untuk sistem operasi Android menyertakan contoh kode pemrosesan gambar menggunakan kamera seperti, deteksi wajah, pencarian fitur dengan cepat, yang di mana menggunakan kombinasi Java dan C++ (Anuar dkk., 2011).

Agar dapat membuat aplikasi berbasis Android menggunakan OpenCV yang menggunakan kombinasi C++, bagian C++ harus dibangun terlebih dahulu. Cara paling populer untuk membangun bagian C++ adalah dengan menggunakan kit pengembangan Android atau *Native Development Kit* (NDK) bersama dengan Cygwin pada sistem operasi seperti Linux dan Windows.

III. METODOLOGI PENELITIAN

3.1 Waktu dan Tempat Pelaksanaan

Penelitian ini dilaksanakan pada periode ganjil tahun ajaran 2020/2021, di Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.

3.2 Alat Pendukung

Pengerjaan penelitian ini menggunakan peralatan pendukung sebagai berikut.

1. Perangkat Keras (*Hardware*)

Perangkat keras yang digunakan dalam penelitian ini berupa :

- Laptop dengan spesifikasi sebagai berikut.
 - *Processor* : AMD Ryzen 3 3200U
 - *Installed RAM* : 8,00 GB
 - *Sistem Type* : 64-bit *operating system*, x64- *processor*

- *Smartphone* dengan spesifikasi sebagai berikut :
 - CPU : Octa-Core Max 1.40GHz
 - *Installed RAM* : 2,00 GB
 - *Android version* : 6.0.1 Marshmallow

2. Perangkat Lunak (*Software*)

Perangkat lunak yang digunakan dalam penelitian ini adalah sebagai berikut.

- Android Studio 4.0.1 sebagai pengembangan aplikasi.
- *Library* OpenCV

3.3 Proses Pembuatan Aplikasi

Aplikasi ini memiliki proses pembuatan yang dilakukan berdasarkan diagram alir yang dapat dilihat pada Gambar 6.

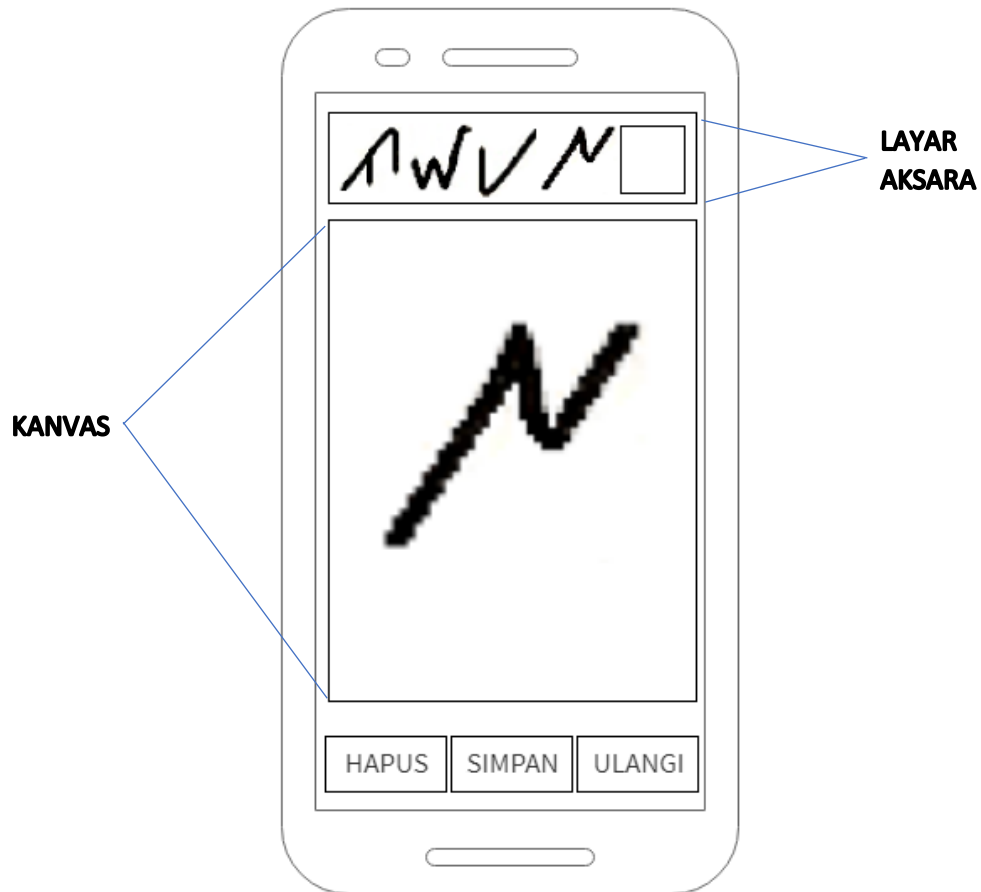


Gambar 6. Tahap pembuatan aplikasi.

3.4 Membuat Area Tulisan

Membuat Kanvas sebagai media untuk menuliskan aksara berupa layar kosong dengan ukuran yang dapat menyesuaikan panjang dan lebar *smartphone*.

Kanvas dibuat dengan ukuran yang dapat menyesuaikan ukuran layar *smartphone* pengguna. Kanvas pada layar *smartphone* dapat menangkap goresan tangan sebagai *input user* untuk menuliskan aksara Lampung yang akan dilakukan proses normalisasi dan binerisasi gambar aksara.



Gambar 7. Area tulisan dan layar aksara.

Pada Gambar 7 ditampilkan aksara Lampung yang telah dituliskan pengguna pada kanvas sebagai *input user*. Kanvas dapat menampung goresan tangan pengguna yang akan menghasilkan gambar aksara Lampung. Aksara Lampung akan ditampilkan pada kanvas dengan warna dasar hitam dan *background* dengan warna putih.

3.5 Menampilkan Aksara

Pada tahap menampilkan aksara, aplikasi akan membaca pergerakan tangan ataupun sentuhan pada kanvas yang telah disediakan. Pada saat pengguna

menyentuh kanvas dan mulai menuliskan aksara menggunakan tangan, maka aplikasi akan merekam informasi berupa koordinat gerakan tangan dari awal menyentuh kanvas hingga akhir pergerakan tangan meninggalkan kanvas pada layar. Informasi koordinat gerakan tangan tersebut kemudian ditampilkan pada kanvas berupa garis dengan warna hitam yang menjadi gambar aksara Lampung.

3.6 Menangkap Gambar

Menangkap gambar merupakan hasil dari penulisan aksara pada kanvas yang sudah ditampilkan pada layar aksara. Pada tahap ini aplikasi dapat mengolah data berupa tulisan aksara Lampung yang dibuat oleh pengguna pada kanvas secara langsung. Hasil dari tulisan pengguna di kanvas kemudian diproses saat itu juga menjadi *file* gambar dengan ekstensi PNG. Gambar tersebut akan tersimpan pada galeri *smartphone* lalu ditampilkan pada layar aksara.

3.7 Normalisasi Gambar

Pada tahap normalisasi gambar yaitu proses untuk memotong gambar aksara. Gambar yang sebelumnya terdapat gambar aksara beserta dengan kanvas/latar belakang putih penuh menjadi hanya gambar aksara saja. Aksara akan memenuhi gambar tersebut tanpa ada ruang kosong atau *padding*. Gambar yang dihasilkan berupa gambar dengan ukuran yang disesuaikan dengan panjang dan lebar aksara yang ditulis pada kanvas, sehingga aksara memenuhi ukuran kanvas yang telah dipotong.

3.8 Binerisasi Gambar

Pada tahap terakhir binerisasi gambar yaitu tahap untuk menghasilkan gambar dengan citra biner. Tahap ini melakukan proses konversi gambar berwarna menjadi skala abu-abu atau *grayscale* yang sudah dilakukan normalisasi pada

gambar aksara, kemudian dikonversi menjadi gambar biner. Hasil akhir dari tahapan ini berupa *file* gambar dengan ekstensi PNG. Pada proses binerisasi ini menggunakan metode Otsu yang berfungsi mempermudah proses binerisasi karena pada metode Otsu proses penentuan *thresholding* dilakukan secara otomatis.

3.9 Perancangan Aplikasi

Perancangan aplikasi merupakan tahapan rencana dalam pengembangan aplikasi kedalam bentuk desain yang akan digunakan untuk memudahkan pengguna melihat rancangan aplikasi yang akan dibuat. Desain aplikasi Pengenalan aksara Lampung berbasis Android sebagai berikut :

1. *Splash Screen*

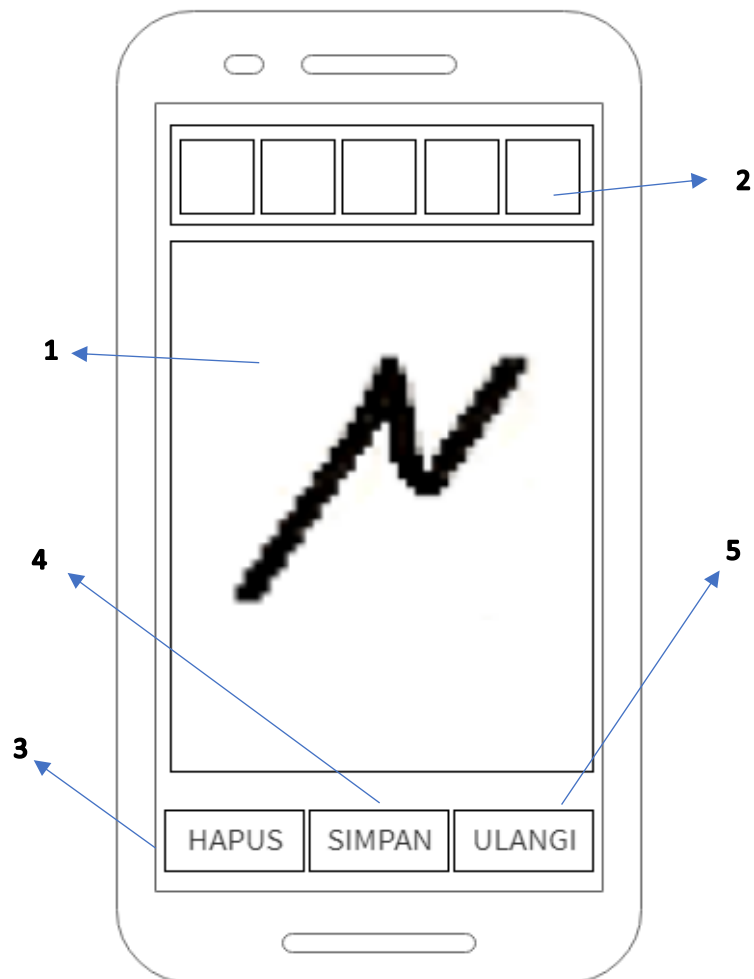
Gambar yang akan muncul pada saat aplikasi pengenalan aksara Lampung dalam proses *loading* pada saat awal dibuka. *Splash screen* pada aplikasi ini berupa gambar penuh yang menutupi layar dengan logo yang terdapat di tengah layar *smartphone* serta teks dari *splash screen* aplikasi dengan durasi 4 detik. *Splash screen* akan muncul pada saat pengguna membuka aplikasi, *splash screen* akan terlihat seperti pada Gambar 8.



Gambar 8. *Splash screen*.

2. Halaman utama

Pada halaman utama terdapat kanvas sebagai media yang berfungsi untuk menuliskan aksara Lampung. Layar aksara yang berfungsi untuk menampung dan menampilkan gambar aksara yang telah ditulis oleh pengguna. Pada halaman utama juga terdapat beberapa tombol yang memiliki fungsi tersendiri seperti tombol simpan yang berfungsi untuk menyimpan hasil akhir aksara berupa gambar yang telah dinormalisasi ukuran dan dilakukan binerisasi pada gambar aksara yang kemudian akan otomatis muncul pada layar aksara. Pada Gambar 9 akan dijelaskan bagian-bagian dari aplikasi serta fungsi dari setiap tombol yang ada pada aplikasi.



Gambar 9. Halaman utama.

Berikut ini keterangan dari tombol dan *view* pada aplikasi:

1. Kanvas

Berisi layar kosong yang digunakan pengguna sebagai media untuk menuliskan aksara Lampung dengan sentuhan tangan.

2. Layar Aksara

Berisi 5 *image view* yang akan menampilkan hasil dari tulisan aksara yang telah pengguna tuliskan pada kanvas setelah pengguna menekan tombol simpan pada *smartphone*.

3. Tombol Hapus

Berfungsi untuk menghapus tulisan aksara pada kanvas sehingga apabila pengguna salah menuliskan aksara pada kanvas maka kanvas akan kembali seperti semula berupa kanvas kosong.

4. Tombol Simpan

Berfungsi untuk menyimpan hasil akhir dari aksara yang telah digoreskan pada kanvas. Saat pengguna menekan tombol simpan maka aksara yang telah ditulis pada kanvas kemudian dilakukan normalisasi ukuran dan binerisasi pada gambar. Gambar aksara kemudian akan disimpan pada galeri *smartphone*.

5. Tombol ulangi

Berfungsi untuk mengulangi penulisan aksara pada kanvas setelah pengguna mencapai batas maksimal menuliskan 5 aksara dalam satu kali proses. Pada saat tombol ulangi ditekan maka kanvas dan layar aksara akan menjadi kosong dan siap untuk melakukan penulisan aksara ulang. Tombol ini tidak menyimpan gambar aksara melainkan hanya untuk mengulangi proses penulisan aksara Lampung.

V. SIMPULAN DAN SARAN

5.1 Simpulan

Kesimpulan yang diperoleh berdasarkan penelitian yang telah dilakukan adalah sebagai berikut :

1. Aplikasi dapat memproses aksara dengan hasil akhir berupa gambar dengan ekstensi PNG.
2. Aplikasi ini telah dapat digunakan untuk pengumpulan data aksara Lampung dengan skala terbatas.

5.2 Saran

Saran yang dapat diberikan untuk penelitian selanjutnya adalah sebagai berikut :

1. Mengembangkan penelitian tentang normalisasi ukuran dan binerisasi *input image* pada kanvas Android agar dapat menampung lebih dari 5 aksara.
2. Mengembangkan penelitian tentang normalisasi ukuran dan binerisasi *input image* Android agar dapat menampung aksara yang berasal dari kamera *smartphone*.

DAFTAR PUSTAKA

DAFTAR PUSTAKA

- Anuar, A., Saipullah, K. M., Ismail, N. A., dan Soo, Y. (2011). OpenCV Based Real-Time Video Processing Using Android Smartphone. *International Journal of Computer Technology and Electronics Engineering (IJCTEE)*, 1(3):58-63.
- Aryantio, A., dan Munir, R. (2015). Pengenalan Aksara Lampung Menggunakan Jaringan Syaraf Tiruan. Bandung: Jurnal Institut Teknologi Bandung bidang Teknik Informatika.
- Bhahri, S., dan Rachmat. (2018). Transformasi Citra Biner Menggunakan Metode Thresholding Dan Otsu Thresholding. *JURNAL SISTEM INFORMASI DAN TEKNOLOGI INFORMASI*, 195-203.
- eMarketer. (2019). *databoks.katadata.co.id*. Retrieved from (<https://databoks.katadata.co.id/datapublish/2016/08/08/pengguna-smartphone-di-indonesia-2016-2019>, diakses tanggal 29 mei 2020)
- Fernando, A. (2013). *Landasan Konseptual Perencanaan dan Perancangan Design Entrepreneur School di Yogyakarta*. Yogyakarta: Skripsi Program Sarjana Universitas Atmajaya Yogyakarta.
- Hara, E., Fitriawan, H., dan Mulyani, Y. (2016). Penggunaan Deteksi Tepi (Canny) pada Sistem Pengenalan Tulisan Tangan Aksara Lampung Berbasis Jaringan Syaraf Tiruan. *Electrician-Jurnal Rekayasa dan Teknologi Elektro*, Volume 10, No. 3.
- Himamunanto, A., dan Widiarti, A.R. (2013). *Teori dan Aplikasi Pengolahan Citra Digital*. Yogyakarta: Lintang Pustaka Utama.

- Junaidi, A. (2016). *Lampung Handwritten Character Recognition*. Dortmund. German: Disertasi Program Doktorat Dortmund University.
- Junaidi, A., Trianingsih, S., dan Iqbal, M. (2020). Writer Identification of Lampung Handwritten Documents Based on Selected Characters. *Khazanah Informatika : Jurnal Ilmu Komputer dan Informatika*, 1-8.
- Kadir, A., dan Sudanto, A. (2012). *Teori dan Aplikasi Pengolahan Citra*. Yogyakarta: ANDI Yogyakarta.
- Lastiansah, S. (2012). *Pengertian User Interface*. Jakarta: PT.Elex Media Komputindo.
- Madenda, S. (2015). *Pengolahan Citra & Video Digital*. Jakarta: Erlangga.
- Mishra, V., Kumar, S., dan Shukla, N. (2017). Image Acquisition and Techniques to Perform Image Acquisition. *A Journal of Physical Sciences, Engineering and Technology*, v9i01.8333.
- Munir, R. (2004). *Pengolahan Citra Digital dengan Pendekatan Algoritmik*. Bandung: Informatika.
- Niblack, W. (1986). An Introduction to Image Processing. *Prentice-Hall, Englewood Cliffs*, 115-116.
- Putra, D. (2004). *Binerisasi Citra Tangan dengan Metode Otsu*. Bali: Majalah Ilmiah Teknologi Elektro.
- Putra, D. (2010). *Pengolahan Citra Digital*. Yogyakarta: Andi.
- Rojatkar, D. V., Jengathe, G., Khaimar, A., dan Lengure, S. (2016). Android Application Development Software - Android Studio And Eclipse. *International Journal For Engineering Applications and Technology*, 1(1).
- Sachs, J. (2001). *Image Resampling*. Digital Light & Color.
- Safaat, H. N. (2011). *Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android*. Bandung: Informatika Bandung.

- Sauvola, J., dan Pietikainen, M. (1999). Adaptive document image binarization. *THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY*, 225-236.
- Tinaliah, dan Triana, E. (2019). Analisis Hasil Resolusi Citra Dengan Metode Interpolasi Nearest Neighbor, Interpolasi Bilinear, dan Interpolasi Bicubic. *JURNAL SISTEM & TEKNOLOGI INFORMASI KOMUNIKASI*, Vol 2 : 1.
- Uluwiyah, R. A., Kusumaningsih, A., dan Nur, O. (2013). Deteksi Manusia dengan Menggunakan Histogram of Oriented Gradients Naïve Bayes Classifier. Madura: Jurnal Universitas Trunojoyo Bidang Teknik Informatika.
- Yodha, J., dan Kurniawan, A. W. (2014). Pengenalan Motif Batik menggunakan Deteksi Tepi Canny dan K-Nearest Neighbor. *Techno.COM*, Vol.13, No.4.