



# INSYST

Journal of Intelligent System and Computation

p-ISSN: 2621-9220

e-ISSN: 2722-1962

Volume 2 Nomor 2, Oktober 2020



Published By **Lembaga Penelitian dan Pengabdian Masyarakat (LPPM)**  
**Institut Sains dan Teknologi Terpadu Surabaya (ISTTS)**  
formerly **Sekolah Tinggi Teknik Surabaya (STTS)**



Managed By  
**Departement of Informatics**  
**Institut Sains dan Teknologi Terpadu Surabaya (ISTTS)**

# INSYST

Journal of Intelligent System and Computation

Volume 02 Nomor 02 Oktober 2020

---

## **Editor in Chief:**

Dr. Yosi Kristian, S.Kom, M.Kom.  
*Institut Sains dan Teknologi Terpadu Surabaya, Indonesia*

## **Managing Editor:**

Dr. Esther Irawati Setiawan, S.Kom., M.Kom.  
*Institut Sains dan Teknologi Terpadu Surabaya, Indonesia*

Reddy Alexandro H., S.Kom., M.Kom.  
*Institut Sains dan Teknologi Terpadu Surabaya, Indonesia*

## **Editorial Board:**

Dr. Ir. Endang Setyati, M.T.  
*Institut Sains dan Teknologi Terpadu Surabaya, Indonesia*

Ir. Edwin Pramana, M.App.Sc, Ph.D  
*Institut Sains dan Teknologi Terpadu Surabaya, Indonesia*

Prof. Dr. Ir. Mauridhi Hery Purnomo, M.T.  
*Institut Sepuluh November, Indonesia*

Hindriyanto Dwi Purnomo, Ph.D.  
*Universitas Kristen Satya Wacana, Salatiga, Indonesia*

Hendrawan Armanto, S.Kom., M.Kom.  
*Institut Sains dan Teknologi Terpadu Surabaya, Indonesia*

Dr. Lukman Zaman PCSW, M.Kom.  
*Institut Sains dan Teknologi Terpadu Surabaya, Indonesia*

Dr. Diana Purwitasari, S.Kom., M.Sc.  
*Institut Sepuluh November, Indonesia*

Dr. Joan Santoso, S.Kom., M.Kom.  
*Institut Sains dan Teknologi Terpadu Surabaya, Indonesia*

# INSYST

Journal of Intelligent System and Computation

Volume 02 Nomor 02 Oktober 2020

---

## **Reviewer:**

Teguh Wahyono, S.Kom., M.Cs.

*Universitas Kristen Satya Wacana, Salatiga, Indonesia*

Dr. Anang Kukuh Adisusilo, ST, MT.

*Universitas Wijaya Kusuma, Surabaya, Indonesia*

Dr. I Ketut Eddy Purnama, ST., MT.

*Institut Sepuluh November, Indonesia*

Prof. Dr. Benny Tjahjono, M.Sc.

*Coventry University, United Kingdom*

Dr. Ir. Gunawan, M.Kom.

*Institut Sains dan Teknologi Terpadu Surabaya, Indonesia*

Dr. Umi Laili Yuhana S.Kom., M.Sc.

*Institut Sepuluh November, Indonesia*

Dr. Tita Karlita, S.Kom., M.Kom.

*Politeknik Elektronika Negeri Surabaya, Indonesia*

Dr. Ir. Rika Rokhana, M.T.

*Politeknik Elektronika Negeri Surabaya, Indonesia*

Dr. I Made Gede Sunarya, S.Kom., M.Cs.

*Universitas Pendidikan Ganesha, Indonesia*

Dr. Yuni Yamasari, S.Kom., M.Kom.

*Universitas Negeri Surabaya, Indonesia*

Dr. Adri Gabriel Sooai, S.T., M.T.

*Universitas Katolik Widya Mandira, Indonesia*

# INSYST

Journal of Intelligent System and Computation

Volume 02 Nomor 02 Oktober 2020

---

## Daftar Isi

|   |    |
|---|----|
| Aplikasi Manajemen Hutang Piutang dengan OCR Berbasis Swift<br>Amelia Alexandra Putri Warsito, Herry Pieter .....   | 47 |
| Ekstraksi Informasi Berbasis Rule untuk Proceeding, Jurnal, dan Technical Report<br>dengan Memanfaatkan Attribut Font dan Paragraf<br>Christian Aditya Santoso, Gunawan ..... | 56 |
| Pembuatan Turn Based Tactical Game "Hex World" Dilengkapi Dengan Artificial<br>Intelligence<br>Albert Rianka Kristian, Hendrawan Armanto .....                                | 63 |
| Steering, Throttle Dan Brake Prediction pada Simulator Self-Driving Car Memanfaatkan<br>CNN<br>Andre Nyoto Raharjo, Andreas .....   | 69 |
| Pemanfaatan 3D U-Net untuk Segmentasi 3 Dimensi Gelembung Penyebab Kanker Paru-<br>paru (Nodule) pada Lapisan Citra CT Scan<br>Agung Pribadi, Anang Kukuh Adisusilo .....     | 74 |
| Automatic Parental Guide Scene Classification Menggunakan Metode Deep<br>Convolutional Neural Network Dan Lstm<br>Riko Gunawan, Yosi Kristian .....                           | 86 |
| 2D Data Visualization Tools Menggunakan Flask dan AngularJS<br>Jerryl Jeovano .....   | 91 |

# Aplikasi Manajemen Hutang Piutang dengan OCR Berbasis Swift

Amelia Alexandra, *Teknik Informatika Institut Sains dan Teknologi Terpadu Surabaya*,  
Herry Pieter, *Teknik Informatika Institut Sains dan Teknologi Terpadu Surabaya*.

**Abstrak**—Perkembangan aplikasi mobile yang pesat juga mendorong banyaknya aplikasi manajemen hutang perseorangan yang mulai beredar di pasaran. Saat ini, aplikasi manajemen hutang perseorangan bisa dibilang sudah cukup banyak, tetapi banyak dari aplikasi tersebut yang tidak terhubung langsung dengan perbankan. Apabila aplikasi tersebut diterapkan di era sekarang ini akan menjadi agak merepotkan, mengingat 80% orang Indonesia menyimpan sebagian besar uangnya pada rekening bank. Selain itu, akan menjadi sangat merepotkan apabila aplikasi pencatatan hutang yang digunakan tidak bisa membantu pengguna untuk mengecek mutasi masuk atau hutang mana yang sudah terbayar dan mana yang belum

**Kata Kunci**— aplikasi mobile ios, image processing, manajemen hutang piutang, optical character recognition,

## I. PENDAHULUAN

Aplikasi pencatatan hutang piutang membuat proses pencatatan hutang akan menjadi lebih rapi dan teratur dikarenakan setiap hutang dapat dicatat dengan baik dan bahkan diverifikasi oleh kedua belah pihak (pihak yang memberi hutang dan pihak yang berhutang), serta setiap pembayaran hutang dapat didokumentasikan dengan baik. Berdasarkan perkembangan teknologi dan masalah di atas, akan dibuat sebuah aplikasi manajemen hutang untuk mencatat segala informasi mengenai hutang dan piutang orang tersebut, mulai dari hutang yang diberikan kepada orang lain, pembayaran yang telah dilakukan oleh pihak yang berhutang, sampai rekapitulasi serta laporan mengenai perhitungan hutang secara menyeluruh dalam suatu periode tertentu.

## II. TEORI DASAR

Aplikasi Manajemen Hutang Piutang dengan OCR Berbasis iOS, seperti swift dengan Xcode, OCR Library, Charts Library, Web Service, API Perbankan, dan Firebase. Berikut ini penjelasan framework dalam bentuk poin-poin.

### A. Swift

Swift adalah bahasa pemrograman yang kokoh dan intuitif yang diciptakan Apple untuk mengembangkan aplikasi untuk iOS, Mac, Apple TV, dan Apple Watch. Bahasa ini didesain untuk memberikan lebih banyak

kebebasan kepada pengembang. Swift mudah digunakan dan bersifat sumber terbuka (open source).

Swift adalah hasil penelitian terbaru tentang bahasa pemrograman, dikombinasikan dengan pengalaman puluhan tahun membangun platform Apple. Parameter diekspresikan dalam sintaks bersih yang membuat API di Swift lebih mudah dibaca. Lebih baik lagi, tidak perlu mengetikkan titik koma setiap akhir sintaks.

Pada pembuatan aplikasi Ios menggunakan swift, pengguna disarankan untuk menggunakan IDE bawaan yaitu *xcode*. Pada *xcode* pengguna akan diberikan peringatan jika variabel yang dibuat tidak efisien, contohnya pada saat pembuatan variabel menggunakan *var* dan *let*. Variabel yang menggunakan *var* memakan memory yang lebih besar dibanding *let*, karena isi dari *var* dapat diubah oleh pengguna. Sedangkan isi dari variabel *let* sudah tidak dapat diubah atau hanya sekali pengisian.

Pada Swift language pengguna juga dapat menggunakan library yang berbasis pada bahasa C, sehingga library yang dapat digunakan sangat banyak. Untuk menggunakan library berbasis C, pengguna perlu membuat file header sebagai jembatan agar bisa digunakan oleh swift. File header tersebut bertujuan untuk mengubah library menjadi object yang bisa digunakan[4].

Salah satu kelemahan dari swift language ialah pada saat terdapat versi baru, function yang digunakan di versi lama, sebagian akan menjadi deprecated dan menyebabkan kebanyakan kode yang telah dibuat menjadi error dan perlu diupdate, sehingga pembuat aplikasi harus mempelajari function baru setiap adanya versi baru pada swift.

### B. OCR

Pada subbab ini akan dijelaskan secara singkat mengenai OCR. OCR merupakan singkatan dari *Optical Character Recognition* merupakan teknik yang digunakan untuk mengubah tulisan pada gambar menjadi sebuah teks yang dapat dikenali atau dibaca oleh aplikasi. Pada pembuatan OCR terdapat dua algoritma yang sering digunakan, berikut ini algoritma tersebut[1].

#### 1. Pattern Recognition

Algoritma ini membutuhkan teks atau tulisan yang digunakan sebagai pembanding, sehingga pada saat proses pengenalan karakter dapat membedakan mana yang merupakan teks dan mana yang bukan merupakan teks.

#### 2. Feature Detection

Algoritma ini digunakan untuk mendeteksi sebuah karakter yang lebih spesifik. Misalnya sebuah garis lurus, garis melengkung ataupun mengerucut. Sehingga hasil yang didapatkan akan lebih bagus dan optimal.

Herry Pieter, Departemen Teknik Informatika, Institut Sains dan Teknologi Terpadu Surabaya, Surabaya, Jawa Timur, Indonesia (e-mail: herrypieter30@gmail.com)

Amelia Alexandra, Departemen Teknik Informatika, Institut Sains dan Teknologi Terpadu Surabaya, Surabaya, Jawa Timur, Indonesia (e-mail: amelia@stts.edu)

Saat ini OCR telah banyak digunakan dalam aplikasi sehari-hari seperti aplikasi editing, scanner gambar, aplikasi lalu lintas dll. OCR mempunyai beberapa tahapan proses yang sering digunakan atau dijadikan dasar dalam pembuatan aplikasi OCR, berikut tahapan kerja tersebut.

1. Meratakan Gambar  
Pada proses ini gambar yang miring akan dibuat sejajar atau lurus.
2. Melakukan Analisa  
Bagian ini merupakan proses untuk mengidentifikasi mana yang gambar dan mana yang teks.
3. Mengatur Arah Gambar  
Pada bagian ini aplikasi akan mengambil sebagian area pada gambar untuk mengidentifikasi sudut gambar.
4. Memisahkan Karakter Huruf dan Angka  
Pada bagian ini proses yang dilakukan adalah memisahkan mana yang merupakan huruf dan angka.
5. Mengidentifikasi Gambar  
Pada bagian ini merupakan proses mengidentifikasi satu persatu gambar yang sudah dipisahkan dan mengubah menjadi teks.
6. Membuat Hasil Akhir  
Setelah angka dan huruf dipisahkan, proses yang terakhir dilakukan ialah menyatukan kembali bagian yang merupakan teks dan menjadikan file yang diinginkan.

### C. Charts

Pada subbab ini akan dijelaskan mengenai charts dari bahasa pemrograman swift yang akan digunakan. Charts ini digunakan untuk menampilkan sekumpulan data yang ingin dikonversikan menjadi grafik-grafik sehingga memudahkan data tersebut untuk dibaca dan dipahami. Berikut ini adalah fitur-fitur yang disediakan oleh Charts ini.

- 1) Penskalaan pada kedua sumbu yang dapat dilakukan dengan touch-gesture, atau pinch zoom
- 2) Charts dapat digeser-geser
- 3) Charts dapat dikombinasikan contoh (line-, bar-, scatter-, candle-stick-, bubble-)
- 4) Sumbu dapat dikustom sesuai dengan kebutuhan
- 5) Charts dapat disimpan menjadi gambar yang dapat berupa PNG atau JPG
- 6) Charts ini dapat memiliki animasi sesuai keinginan
- 7) Sangat dapat dikustom dari warna, jenis font, legenda, warna, latar belakang, gesture, dan lain sebagainya

Charts mempunyai kelebihan dalam proses implementasi yang disebabkan karena charts merupakan library bawaan dari swift, sehingga pembuat tinggal memanggil library tersebut untuk menggunakan charts. Charts juga mempunyai delapan macam bentuk grafik, dimana grafik tersebut pada umumnya dibutuhkan dalam laporan. Delapan grafik tersebut yaitu :

1. Line Chart (with legend, simple design)
2. Bar Chart (simple design)
3. Combined Chart (bar and line)
4. Pie Chart
5. Scatter Chart
6. Candle Stick Chart
7. Buble Chart
8. Radar Chart

Pada masing-masing jenis chart juga terdapat pilihan yang bisa dicustom, seperti pada bar chart, dimana pengguna bisa membuat simple, horizontal dan dataset bar chart. Pada aplikasi manajemen hutang piutang, jenis grafik yang dipakai ialah pie chart. Cara penggunaan pie chart cukup dengan memasukkan array yang berisi label beserta nominal yang didapatkan ke dalam dataset charts. Kemudian setelah mengatur dataset, pengguna perlu melakukan reload ke object chart agar charts menampilkan data yang didapatkan.

### D. Web Service

Pada subbab ini akan dijelaskan mengenai web service. Web service pada aplikasi manajemen hutang piutang ini adalah sebagai perantara antara database dengan aplikasi manajemen hutang piutang ini. Web service yang digunakan berbasis REST API. Berikut adalah penjelasan mengenai web service yang digunakan pada tugas akhir ini.

REST (Representational State Transfer) adalah suatu arsitektur metode komunikasi yang menggunakan protokol HTTP untuk pertukaran data dan metode ini sering diterapkan dalam pengembangan aplikasi. Dimana tujuannya adalah untuk menjadikan sistem yang memiliki performa yang baik, cepat dan mudah untuk di kembangkan terutama dalam pertukaran dan komunikasi data[5].

Pada subbab ini akan dijelaskan mengenai web service. Web service pada aplikasi manajemen hutang piutang ini adalah sebagai perantara antara database dengan aplikasi manajemen hutang piutang ini. Web service yang digunakan berbasis REST API. Berikut adalah penjelasan mengenai web service yang digunakan pada tugas akhir ini.

REST (Representational State Transfer) adalah suatu arsitektur metode komunikasi yang menggunakan protokol HTTP untuk pertukaran data dan metode ini sering diterapkan dalam pengembangan aplikasi. Dimana tujuannya adalah untuk menjadikan sistem yang memiliki performa yang baik, cepat dan mudah untuk di kembangkan terutama dalam pertukaran dan komunikasi data.

Slim framework sangat mudah digunakan oleh pengguna, karena memiliki kerangka yang telah disediakan. Sehingga pengguna hanya perlu mengatur username, password, dan database yang akan digunakan. Slim framework sangat cocok digunakan untuk proyek skala kecil hingga menengah, karena proses pemasangan yang tidak lebih dari 5 menit[6].

Kelemahan dari slim framework pada saat pengguna ingin membuat object oriented. Proses pengaturan yang diperlukan lebih rumit dibanding dengan pembuatan menggunakan laravel atau lumen. Slim framework menggunakan query SQL dan tidak menyediakan query builder, seperti web service framework lainnya.

### E. Face ID

Pada subbab ini akan dijelaskan mengenai Face ID yang merupakan fitur keamanan buatan Apple. Face ID dirilis pada November 2017 bersamaan dengan dirilisnya iPhone X. Fitur ini dikembangkan untuk menjadi pengganti Touch ID. Face ID bekerja dengan menggunakan camera *true depth* yang memindai 30.000 titik infra merah yang digunakan untuk memetakan wajah pengguna menjadi gambar 2D.

Gambar ini disimpan di dalam *secure enclave* secara lokal dalam ponsel tiap pengguna. Face ID menggunakan

beberapa sistem keamanan yang bertujuan untuk mencegah *spoofing*, sehingga Face ID dapat mencegah pengguna lain yang ingin membuka ponsel dengan topeng dan foto [8].

Face ID dapat mengenali pengguna walaupun menggunakan aksesoris tambahan seperti topi, kacamata, bahkan kumis dan jenggot. Kelemahan utama Face ID ialah tidak dapat membedakan pengguna apabila memiliki kembaran yang mirip. Face ID diklaim hanya dapat dibobol oleh satu orang berbanding satu juta karena menyimpan hingga 300.000 titik muka yang sangat unik.

Pada aplikasi manajemen hutang piutang, Face ID digunakan sebagai salah satu security untuk dapat membuka aplikasi. Untuk dapat menggunakan Face ID dalam aplikasi manajemen hutang piutang, pengguna harus mengizinkan aplikasi untuk mengakses Face ID pada perangkat dan juga pengguna harus mengaktifkan Face ID di perangkat tersebut.

Pada Swift Language, implementasi Face ID dan Touch ID menggunakan module yang sama. Namun, untuk dapat menggunakan Touch ID, pengembang perlu mengatur hak akses yang berbeda dengan Face ID, dan juga handler yang digunakan untuk mengolah data yang dikembalikan Touch ID berbeda dengan Face ID.

#### F. Api Perbankan

Pada subbab ini akan dijelaskan mengenai API Perbankan. API Perbankan adalah API yang digunakan untuk mengecek status transaksi dari pengguna. Untuk aplikasi manajemen hutang piutang ini, API Perbankan oada aplikasi manajemen hutang piutang ini digunakan untuk validasi pembayaran utang atau piutang via transfer.

Cekmutasi.co.id merupakan sebuah web-based software terintegrasi untuk mengelola banyak rekening dalam satu dashboard. Sistem ini dilengkapi robot mutasi yang bekerja memeriksa transaksi terbaru dalam rekening pengguna dan menyajikannya dalam sebuah laporan sehingga pengguna tidak perlu melakukan login internet banking berulang-ulang hanya untuk mengecek pembayaran.

Sistem cekmutasi juga mendukung proses validasi pembayaran otomatis yang dapat diintegrasikan ke bisnis pengguna melalui Restful API. Cekmutasi sendiri sudah support dengan Bank BRI, BNI, BCA, Mandiri, PayPal, OVO & GO-PAY [10]. Cekmutasi.co.id juga sudah menyediakan url callback untuk memudahkan proses integrasi, berikut contoh penggunaan API Perbankan pada aplikasi manajemen hutang piutang ini.

Pada aplikasi manajemen hutang piutang, fitur cekmutasi.co.id yang digunakan ialah notifikasi callback. Pembuat aplikasi dapat memasukkan url API yang telah dibuat pada saat pendaftaran rekening di dashboard cekmutasi.co.id, kemudian mengatur interval bot cekmutasi.co.id untuk melakukan crawl ke internet banking masing-masing bank. Jika bot pada mutasi.co.id menemukan mutasi baru pada internet banking, cekmutasi.co.id akan mengirim notifikasi ke url callback yang telah diatur. Data JSON yang dikirimkan oleh cekmutasi.co.id akan diproses oleh web service untuk melakukan pengecekan request pembayaran oleh pengguna.

Kelemahan dari penggunaan cekmutasi.co.id ialah proses yang dilakukan tidak real time, karena terdapat delay pada interval crawl yang dilakukan. Jika interval crawl terlalu kecil, resiko akun internet banking pengguna akan diblock

lebih besar. Interval yang disarankan untuk melakukan crawl ialah setiap 3-5 menit sekali.

#### G. Firebase Notification

Menggunakan Firebase Cloud Messaging (FCM) dapat memberi tahu aplikasi klien bahwa email baru atau data lain tersedia untuk disinkronkan. Pengguna dapat mengirim pesan notifikasi untuk mendorong keterlibatan dan retensi pengguna[3].

Untuk kasus-kasus penggunaan seperti pengiriman pesan instan, sebuah pesan dapat mentransfer muatan hingga 4KB ke aplikasi klien. Implementasi FCM mencakup dua komponen utama untuk mengirim dan menerima pesan. Dua komponen utama tersebut ialah.

- 1) Environment yang terpercaya seperti firebase cloud, atau sebuah server aplikasi yang digunakan untuk membuat, dan mengirim pesan.
- 2) Aplikasi klien berbasis iOS, Android, atau web (JavaScript) yang menerima pesan melalui layanan platform terkait.

Firestore memiliki kelebihan yaitu proses implementasi yang mudah dan dokumentasi yang lengkap. Firestore juga memiliki kecepatan dalam proses pengirim notifikasi yang sangat cepat dan menggunakan data yang lebih kecil dari yang lain. Untuk melakukan pemasangan notifikasi firestore ke aplikasi iOS pengguna memerlukan akun developer dari Apple. Akun tersebut digunakan untuk mengeneratekan certificate untuk push notification.

Certificate tersebut akan dimasukkan ke dashboard firestore agar pengguna dapat mengirim push notification. Firestore juga menyediakan fitur scheduled notification, dimana pembuat aplikasi dapat mengatur tanggal serta jam yang diinginkan[2].

#### H. Preprocessing Gambar

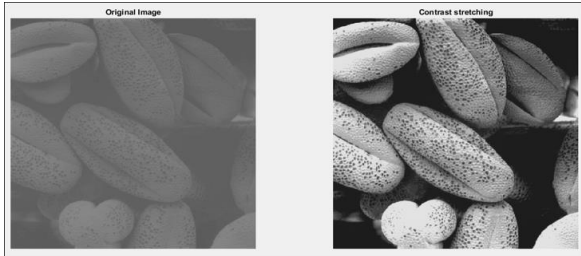
Pada subbab ini akan dijelaskan mengenai preprocessing gambar. Preprocessing merupakan suatu bentuk pengolahan atau pemrosesan dengan input gambar dan diubah menjadi suatu gambar lain dengan menggunakan teknik tertentu. Image preprocessing dilakukan untuk memperbaiki serta meningkatkan kualitas gambar agar lebih mudah dilihat oleh manusia dan proses manipulasi terhadap gambar [9]. Image preprocessing terbagi menjadi beberapa kelompok berdasarkan dari tujuan transformasi yang diinginkan. Berikut ini kelompok dari Image Processing.

1. Image Enhancement  
Proses ini berfungsi untuk meningkatkan fitur tertentu pada citra, sehingga tingkat keberhasilan dalam pengolahan gambar menjadi lebih tinggi. Contoh metode image enhancement seperti contrast stretching, gray scalling, linear contrast adjusment dan histogram equalization.
2. Image Restoration  
Proses ini bertujuan untuk mengembalikan kondisi gambar yang sudah rusak atau cacat karena adanya gangguan yang menyebabkan penurunan kualitas gambar. Contoh metode yang bisa digunakan dalam kelompok ini adalah median filter, adaptive filter, linear filter dan block matching.
3. Image Compression  
Proses ini bertujuan untuk membuat ukuran gambar menjadi lebih kecil dengan memadatkan bit pada citra. Proses ini dilakukan agar pengiriman gambar

menjadi lebih cepat. Contoh teknik yang umum digunakan untuk image compression ialah. Fractal, wavelets, chroma sub-sampling dan transform coding.

4. Image Representation & Modelling

Proses ini merupakan pengelompokan batas-batas gambar yang ingin digunakan. Metode yang sering digunakan untuk image representation & modelling ialah block pattern.



Gambar. 1. Contoh Sesudah dan Sebelum Contrast Stretching

Pada gambar 1 merupakan perbedaan gambar asli dan setelah dilakukan proses contrast stretching, gambar yang menggunakan contrast stretching membuat content di dalam gambar lebih jelas dan mudah dikenali. Oleh karena itu pre processing gambar sangat dibutuhkan didalam aplikasi ini.

Pada aplikasi manajemen hutang piutang, kelompok image preprocessing yang digunakan ialah image enhancement. Metode ini digunakan karena bertujuan untuk meningkatkan kualitas gambar nota sebelum dikirim ke Google Vision dan memberi hasil yang lebih optimal. Metode image enhancement yang digunakan ialah gray scalling dan contrast stretching.

III. ANALISA SISTEM

Pada tahap ini dijelaskan tentang analisa sistem dari aplikasi manajemen hutang piutang. Penjelasan berupa deskripsi sistem yang dibuat, deskripsi dari sistem pada aplikasi-aplikasi referensi analisa permasalahan yang ada, dan spesifikasi kebutuhan dari sistem aplikasi yang dibuat.

Aplikasi Splitwise memiliki beberapa fitur, seperti menambah utang, melihat total utang dan piutang keseluruhan, selain itu terdapat fitur untuk menyederhanakan utang piutang di dalam group. Pada fitur dari Splitwise user dapat melakukan register melalui email dan login menggunakan google, setelah user melakukan login, user dapat melihat total hutang dan piutang pada halaman dashboard aplikasi tersebut.

Untuk membayar hutang ke pengguna lain, pengguna dapat masuk ke halaman teman atau grup dan terdapat tombol "Settle up" yang berfungsi melunaskan hutang ke pengguna lain seperti pada gambar 3.4, setelah itu pengguna satu nya akan mendapat notifikasi bahwa hutang sudah dibayar oleh pengguna lain. Jika pengguna tersebut merasa tidak menerima pembayaran, maka si pengguna dapat menghapus pembayaran yang dibuat oleh pengguna lain.

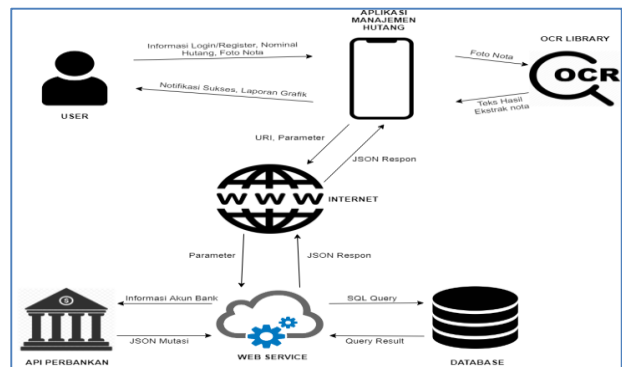
akan dijelaskan hasil spesifikasi kebutuhan yang diperlukan sistem aplikasi manajemen utang piutang. Spesifikasi kebutuhan ini berupa fitur-fitur yang menjadi solusi dari aplikasi referensi. Fitur-fitur yang menjadi solusi ini menjadi fitur utama pada aplikasi manajemen hutang piutang. Berikut dijelaskan fitur-fitur yang menjadi solusi dan fitur pendukung lain tersebut dalam bentuk poin-poin.

- Input Cicilan

- Pelunasan Hutang Otomatis
- Konfirmasi Hutang
- Kategori dan Keterangan Hutang
- Laporan Hutang
- Verifikasi Pembayaran Hutang Otomatis
- Chat
- Scan Nota
- Scan Bukti Pembayaran
- Promo
- Security Password

IV. DESAIN

Pada tahap ini akan dijelaskan desain-desain yang diterapkan pada program aplikasi manajemen hutang piutang untuk aplikasi manajemen hutang piutang ini. Penjelasan berupa desain arsitektur, desain database, desain tampilan.



Gambar. 2. Arsitektur Sistem

Seperti dapat dilihat pada Gambar 2, alur dari aplikasi ini dimulai ketika user membuka aplikasi. Saat aplikasi dibuka pertama kali, maka akan diberikan tawaran apakah user sudah memiliki akun atau belum. Apabila user belum memiliki akun, maka user dapat melakukan register terlebih dahulu. Namun, ketika user telah memiliki akun, maka user dapat langsung masuk dengan akun yang telah dimiliki. Ketika user telah berada dalam halaman home, maka user sudah dapat menggunakan fitur dari aplikasi ini dengan utuh. User mulai dapat menambahkan hutang dengan cara klik menu tambah hutang, pilih teman yang hendak ditunjuk sebagai pihak yang berhutang, dan nominal. Selain dengan menginputkan manual, user juga dapat melampirkan foto dari nota belanjanya. Dengan melampirkan foto nota belanja tersebut, maka aplikasi akan secara otomatis memprosesnya sebagai hutang baru. Foto yang dilampirkan di dalam aplikasi tersebut akan diproses oleh OCR Library untuk diekstrak menjadi informasi yang lebih siap diolah dalam aplikasi[7].

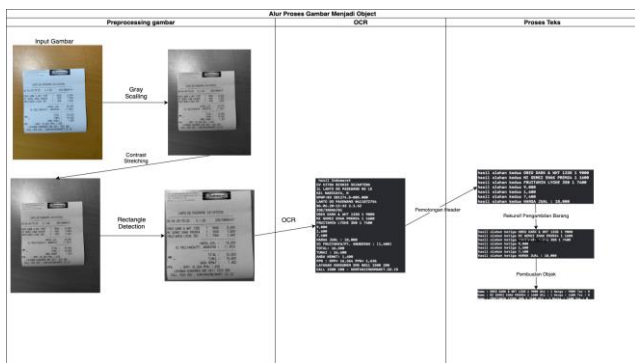
Setiap data hutang tersebut akan disinkronisasikan secara berkala dengan bantuan internet. Internet akan berguna untuk menghubungkan aplikasi dengan server. Koneksi dengan server sangat diperlukan untuk melihat dan menyimpan daftar hutang yang tersimpan di dalam server. Selain untuk mengelola daftar hutang, koneksi dengan internet juga diperlukan apabila aplikasi membutuhkan koneksi dengan akun bank yang dibantu oleh web service dan API Perbankan yang memudahkan pengguna.

Seperti dapat dilihat pada Gambar 1, alur dari aplikasi ini dimulai ketika user membuka aplikasi. Saat aplikasi dibuka pertama kali, maka akan diberikan tawaran apakah user



sudah memiliki akun atau belum. Apabila user belum memiliki akun, maka user dapat melakukan register terlebih dahulu. Namun, ketika user telah memiliki akun, maka user dapat langsung masuk dengan akun yang telah dimiliki. Ketika user telah berada dalam halaman home, maka user sudah dapat menggunakan fitur dari aplikasi ini dengan utuh. User mulai dapat menambahkan hutang dengan cara klik menu tambah hutang, pilih teman yang hendak ditunjuk sebagai pihak yang berhutang, dan nominal. Selain dengan menginputkan manual, user juga dapat melampirkan foto dari nota belanjanya. Dengan melampirkan foto nota belanja tersebut, maka aplikasi akan secara otomatis memprosesnya sebagai hutang baru. Foto yang dilampirkan di dalam aplikasi tersebut akan diproses oleh OCR Library untuk diekstrak menjadi informasi yang lebih siap diolah dalam aplikasi.

Setiap data hutang tersebut akan disinkronisasikan secara berkala dengan bantuan internet. Internet akan berguna untuk menghubungkan aplikasi dengan server. Koneksi dengan server sangat diperlukan untuk melihat dan menyimpan daftar hutang yang tersimpan di dalam server. Selain untuk mengelola daftar hutang, koneksi dengan internet juga diperlukan apabila aplikasi membutuhkan koneksi dengan akun bank yang dibantu oleh web service dan API Perbankan yang memudahkan pengguna sehingga tidak diperlukan untuk mengecek sendiri.



Gambar. 5. Flow Gambar Menjadi Object

Pada subbab ini akan dijelaskan mengenai alur input gambar menjadi objek yang akan digunakan di dalam aplikasi manajemen hutang piutang. Pada gambar 5 terdapat tiga bagian proses, yaitu preprocessing image, OCR yang dilakukan oleh Google Vision, dan proses yang dilakukan terhadap teks hasil OCR.

Pada proses preprocessing, langkah pertama yang dilakukan adalah gambar akan diubah dengan gray scalling, kemudian akan dilakukan contrast stretchcing agar gambar yang blur menjadi lebih jelas, serta garis pada nota menjadi lebih terlihat. Proses terakhir yang dilakukan pada preprocessing ialah rectangle detection untuk membuang background pada gambar dan berfokus ke nota. Pada rectangle detection, batasan gambar nota yang bisa diproses apabila k empat sudut nota tidak sampai ujung gambar.

Pada proses selanjutnya, gambar yang telah diproses akan dikirim ke Google Vision untuk melakukan OCR. Google Vision akan melakukan proses layouting dan pengenalan karakter pada gambar yang dikirim, kemudian Google Vision akan mengembalikan hasil berupa string yang berhasil dibaca. String tersebut akan diproses untuk membuang baris yang tidak terpakai.

Proses pengolahan teks yang pertama ialah pemotongan header. Proses ini bertujuan untuk membuang semua kalimat yang ada pada header nota, sehingga mempermudah dalam pengambilan nama barang. Dalam proses pemotongan header terdapat beberapa step yang dilakukan, step tersebut sebagai berikut.

1. Pencarian index kalimat pertama yang mengandung harga dan index lebih besar dari 3 dalam teks yang dihasilkan oleh Google Vision dengan menggunakan regex `'[0-9][.,][0-9]'` dan regex `'@[0-9][.,][0-9]'`.
2. Index kalimat pertama yang mengandung harga menjadi batas akhir dalam proses looping mencari index terbesar dari kalimat yang mengandung pola jam, tanggal, kode nota dan salah huruf akibat kesalahan pengenalan karakter oleh Google Vision. Regex yang digunakan pencarian format header yang sudah ditentukan ialah `'[0-9][:~/][0-9]'`, `'[*?!~/]'` dan `'^\\s*([a-zA-z]{1})$'`
3. Index yang didapatkan dari proses looping point nomor 2 akan ditambahkan satu, kemudian hasil tersebut akan digunakan untuk menjadi batasan akhir dalam proses penghapusan isi array.

Kelemahan utama dalam proses pemotongan header, yaitu jika pada header nota terdapat pola yang sesuai dengan regex yang sudah ditentukan. Seperti terdapat nomor npwp pada header atau kode nota yang mempunyai format seperti harga, yang menyebabkan index batasan yang didapat bukan index harga pertama dalam suatu nota.

Setelah proses pemotongan header selesai, maka akan dilanjutkan ke pemotongan footer, yang bertujuan untuk membuang kalimat yang ada pada footer, sehingga data yang terdapat pada array hanya data barang beserta atributnya. Dalam proses pemotongan footer akan dilakukan looping untuk mencari kalimat pertama atau index terkecil dalam array yang mengandung pola atau kata yang telah ditentukan, pola atau kata yang digunakan sebagai berikut.

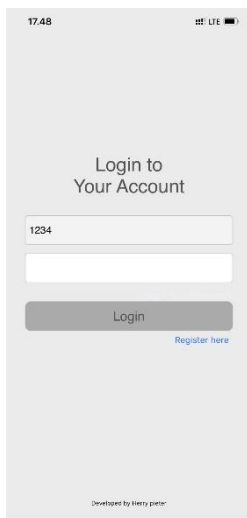
1. Mencari kalimat yang mengandung "Total", "Diskon" dan "Subtotal".
2. Menggunakan regex `"[-]{2}"`, yang bertujuan untuk mencari pembatas garis nota yang mempunyai pola "----" yang berhasil terbaca oleh Google Vision.
3. Menggunakan regex `"[a-zA-Z]:"`, yang bertujuan untuk mencari pola "Total:" yang terdapat pada bagian footer.
4. Menggunakan regex `"^([a-zA-z])$"`, yang bertujuan untuk mencari karakter salah pengenalan dari Google Vision.

Proses looping bertujuan untuk mendapatkan index kalimat yang akan digunakan sebagai batasan dalam pemotongan data array. Setelah data melalui dua proses pemotongan, maka proses terakhir yang dilakukan adalah pengambilan data barang. Dalam proses pengambilan data, akan dilakukan looping terhadap array untuk memisahkan data menjadi 4 bagian, yaitu nama, kuantiti, diskon, dan harga. Terdapat beberapa step yang dilakukan dalam pengambilan data, proses tersebut yaitu:

1. Setiap baris kalimat akan dilakukan pengecekan apa kalimat tersebut nama barang atau bukan dengan menggunakan regex `"[a-zA-Z0-9]{2} [a-zA-Z0-9]{2}"`.
2. Apabila kalimat tersebut sesuai dengan regex pada poin 1, maka kalimat tersebut akan di split dengan karakter spasi.

3. Kata yang dipecah akan dilakukan pengecekan untuk mencari apabila ada harga, kuantiti atau harga.
4. Regex yang digunakan untuk pengecekan kuaniti yaitu “[0-9]”.
5. Regex yang digunakan dalam aplikasi untuk pengecekan harga, yaitu “[1-9]{1}[\.\,][0-9]{3}”.
6. Regex yang digunakan untuk melakukan pengecekan diskon yaitu “-[0-9]{2}”.
7. Apabila kalimat tidak sesuai dengan regex pada poin pertama, maka kalimat tersebut akan displit dengan pemisah karakter spasi. Kemudian, setiap kata yang merupakan hasil split akan dilakukan pengecekan dengan menggunakan pola atau regex untuk kuantiti, harga dan diskon. Apabila kata yang didapat tidak sesuai dengan pola yang sudah ditentukan, maka kata tersebut akan diabaikan.
8. Setelah semua data pada array selesai dipisahkan menjadi empat bagian, selanjutnya data pada array harga akan dinormalisasi untuk menghilangkan karakter ‘,’ dan ‘.’, yang bertujuan agar dapat diconvert menjadi tipe data Int.
9. Setelah harga dinormalisasi, maka keempat bagian yang sudah dipisahkan akan disatukan menjadi object yang akan digunakan pada aplikasi.

Proses pengambilan data yang dilakukan memiliki beberapa kelemahan, yaitu apabila produk hanya terdiri dari 1 kata, maka produk tersebut tidak dapat terbaca. Selain itu, apabila nama produk hanya mengandung angka, maka nama produk tersebut akan terbaca sebagai harga.



Gambar. 6. Halaman Login

Pada gambar 6 adalah tampilan login saat user sudah mendaftar pada aplikasi. Untuk login user perlu memasukkan email / nomor telepon dan password. User dapat menekan tombol masuk ketika email dan password terisi dengan benar. Jika pengguna salah memasukkan nomor telepon yang digunakan, maka aplikasi akan memunculkan pesan error “Nomor hp / password salah”.



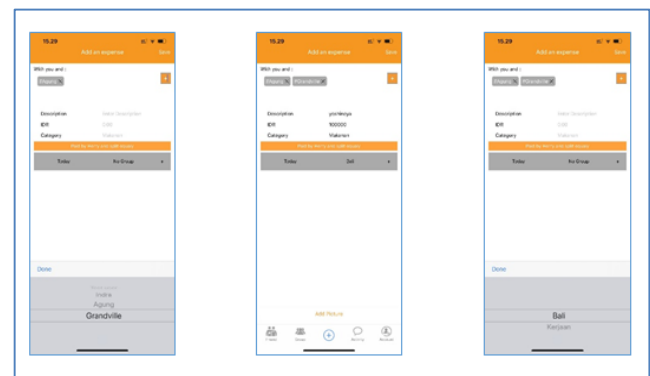
Gambar. 7. Halaman Utama

Pada gambar 7 adalah halaman daftar teman dan total hutang atau piutang dari pengguna dan juga halaman utama ketika pengguna melakukan login. Gambar 7 adalah daftar teman yang dimiliki oleh pengguna beserta total hutang dan piutang dari masing-masing teman yang ada.



Gambar. 8. Halaman Group

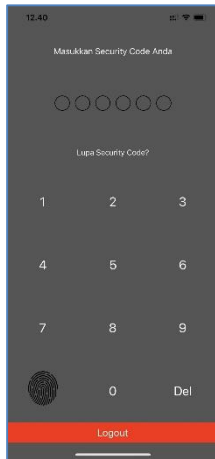
Pada gambar 8 adalah daftar keseluruhan group yang dimiliki oleh pengguna beserta total hutang atau piutang yang di miliki pengguna di group tersebut. Pada saat pengguna baru melakukan register dan login, maka halaman ini akan menjadi kosong.



Gambar. 9 Tambah Hutang atau piutang

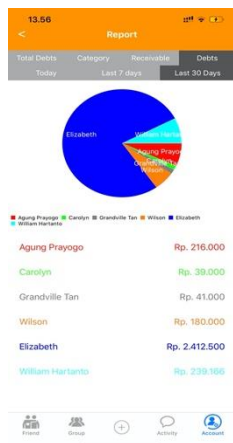
Gambar 9 merupakan tampilan untuk menambahkan transaksi di dalam aplikasi manajemen hutang piutang. Pengguna dapat memilih dengan siapa saja ingin melakukan

transaksi, atau dengan group mana transaksi tersebut. Pengguna juga dapat memilih kategori, mengisi tanggal, deskripsi dan nominal transaksi.



Gambar. 10 Halaman Passcode

Pada gambar 10 merupakan halaman dari passcode pada aplikasi manajemen hutang piutang. Untuk memunculkan halaman ini, pengguna harus mengaktifkan passcode pada halaman my account. Face ID akan otomatis dapat digunakan setelah pengguna mengaktifkan passcode.



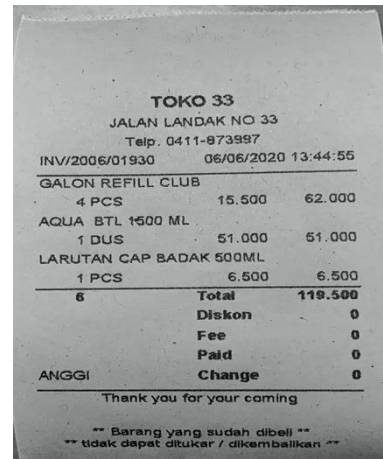
Gambar. 11 Halaman Laporan

Gambar 11 merupakan contoh laporan pada aplikasi manajemen hutang piutang. Laporan pada gambar 11 merupakan laporan total hutang yang dikelompokkan berdasarkan teman. Selain itu, terdapat juga laporan total hutang dan piutang serta laporan total piutang yang dikelompokkan berdasarkan dengan teman yang dimiliki pengguna.

### V. UJI COBA

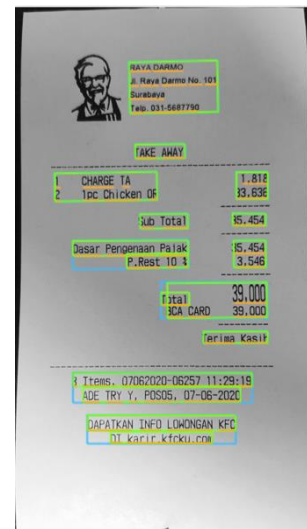
Pada tahap ini akan dijelaskan mengenai uji coba yang dilakukan pada aplikasi manajemen hutang piutang ini. Uji coba ini bertujuan untuk mengetahui semua fungsi berjalan dengan baik dan uji kelayakan bertujuan untuk mengetahui kekurangan dari calon pengguna.

Fitur yang diuji coba merupakan fitur-fitur utama di dalam aplikasi meliputi OCR, laporan, passcode, serta menambahkan hutang atau piutang pada aplikasi.



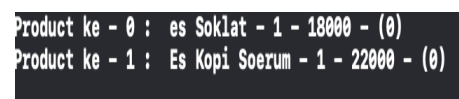
Gambar. 12. Output Preprocessing

Pada gambar 12 merupakan hasil setelah melakukan preprocessing pada gambar. Proses ini terdiri dari gray scalling, contrast stretching dan rectangle detection. Proses ini bertujuan untuk membuat gambar yang blur menjadi lebih jelas dan juga membuat output dari Google Vision menjadi lebih optimal.



Gambar. 6. Output Segmentasi Google Vision

Gambar 6 merupakan hasil uji coba segmentasi Google Vision. Pada hasil uji coba segmentasi yang dilakukan terhadap 30 nota, terdapat total 15 nota yang mempunyai akurasi kurang dari 100%, yaitu 4 nota Alfamart, 2 nota Indomaret, 7 nota Toko 33, dan 2 nota Janji Jiwa. Sehingga akurasi segmentasi yang dihasilkan Google Vision ialah 94.79%. Penyebab utama kesalahan yang terjadi ialah Google Vision tidak dapat mendeteksi satu digit angka yang posisinya berada di ujung nota. Penyebab lainnya karena Google Vision mendeteksi watermark pada nota, sehingga akan berpengaruh pada hasil pembacaan karakter yang dilakukan



Gambar. 7. Hasil Uji Coba Pengambilan Barang

Gambar 7 merupakan input dari proses pengambilan barang beserta atributnya pada aplikasi manajemen hutang piutang. Pada uji coba proses pengambilan nama barang

terhadap 30 nota. Tingkat akurasi yang dihasilkan sebesar 89.99%, dengan perbandingan jumlah nama, kuantitas, dan harga yang berhasil diambil dibagi dengan jumlah nama, kuantitas, dan harga keseluruhan yang terdapat pada nota. Penyebab kesalahan yang terjadi yaitu :

1. Pada saat proses pengambilan nama beserta atributnya karena kesalahan posisi dari atribut barang yang terdapat di bagian footer.
2. Data yang diterima masih terdapat footer dan header, sehingga data tersebut dibaca menjadi data barang beserta atributnya.
3. Nama barang pada nota berpindah menjadi dua baris, yang menyebabkan aplikasi manajemen hutang piutang mengganggu dua produk yang berbeda.

Kuisisioner ini akan dijelaskan hasil kuisisioner yang dibagikan kepada minimal lima belas pengguna sesuai kriteria pada proposal yaitu usia 17 sampai 40 tahun. Terdapat lima belas responden yang menggunakan aplikasi minimal tiga kali kemudian menjawab kuisisioner yang diberikan.

Tabel 1  
Pertanyaan dan Presentase Jawaban Untuk Aplikasi Yang Dibuat

| No | Pertanyaan  | Sangat Kurang     | Kurang             | Cukup               | Baik                 | Sangat Baik     |
|----|---|-------------------|--------------------|---------------------|----------------------|-----------------|
| 1  | Seberapa mudah menggunakan aplikasi ini?                                    | 4,2%<br>(0 orang) | 0%<br>(0 orang)    | 33,33%<br>(5 orang) | 66,67%<br>(10 orang) | 0%<br>(0 orang) |
| 2  | Apakah aplikasi ini mempermudah dalam proses pencatatan hutang dan piutang? | 0%<br>(0 orang)   | 26,6%<br>(4 orang) | 46,6%<br>(7 orang)  | 26,8%<br>(4 orang)   | 0%<br>(0 orang) |
| 3  | Apakah fitur aplikasi sudah cukup ?   | 0%<br>(0 orang)   | 13,3%<br>(2 orang) | 66,7%<br>(10 orang) | 20%<br>(3 orang)     | 0%<br>(0 orang) |
| 4  | Bagaimana dengan tampilan keseluruhan aplikasi ?                            | 0%<br>(0 orang)   | 46,7%<br>(7 orang) | 46,7%<br>(7 orang)  | 6,6%<br>(1 orang)    | 0%<br>(0 orang) |
| 5  | Seberapa puas menggunakan aplikasi ?  | 0%<br>(0 orang)   | 26,7%<br>(4 orang) | 53,4%<br>(8 orang)  | 19,9%<br>(3 orang)   | 0%<br>(0 orang) |

Keterangan hasil yang dirinci pada tabel 1 membuktikan bahwa kenyamanan user untuk pemakaian aplikasi cukup baik. Persentase kemudahan aplikasi memiliki penilaian yang lebih tinggi dibanding penilaian buruk. Kemudian presentasi kepuasan responden dalam menggunakan aplikasi sudah cukup baik karena memiliki nilai yang lebih tinggi dibanding pengguna yang kurang puas.

## VI. KESIMPULAN

Pada subbab ini akan dijelaskan mengenai kesimpulan yang didapat selama mengerjakan tugas akhir. Adapun beberapa proses pengerjaan seperti penggunaan swift language, penggunaan firebase, pengguna web service, alur sistem dan semua masalah yang dihadapi semasa

pengerjaan. Berikut ini penjabaran kesimpulan yang didapatkan dalam bentuk poin-poin:

- 1) Aplikasi sudah cukup memenuhi kebutuhan pengguna dalam pencatatan hutang dan piutang, dengan presentasi 26.6% menyatakan kurang, 46,6% menyatakan cukup dan 26,8% menyatakan baik.
- 2) Pengerjaan aplikasi manajemen hutang piutang menggunakan Swift cukup mempermudah dalam pengerjaan aplikasi manajemen hutang piutang. Selain itu, Swift juga mempunyai banyak plugin bawaan seperti charts, json decoder, alamofire yang membantu dalam pengerjaan aplikasi manajemen hutang piutang. Selain itu banyak library open source yang tersedia dan sangat membantu dalam proses pengerjaan aplikasi manajemen hutang piutang.
- 3) Penggunaan storyboard sangat mempermudah dalam melihat alur program, tetapi sangat sulit untuk melakukan design UI/UX yang bagus karena keterbatasan komponen dan bahasa yang sangat cepat berubah, sehingga mempersulit dalam hal desain tampilan.
- 4) Penggunaan bahasa pemrograman PHP untuk web service sudah cukup membantu dalam pembuatan web service. Namun slim framework masih kurang baik dalam menerapkan *object oriented* dan *eloquent*, sehingga banyak proses yang dilakukan.
- 5) Tingkat keberhasilan segmentasi Google Vision ialah 94.79%.
- 6) Tingkat akurasi pengenalan nama beserta harga dan kuantitas produk oleh Google Vision sebesar 93.72%.
- 7) Tingkat akurasi proses pemotongan header pada aplikasi manajemen hutang piutang sebesar 97.99%.
- 8) Tingkat akurasi proses pemotongan footer pada aplikasi manajemen hutang piutang sebesar 94.62%.
- 9) Tingkat akurasi pengambilan nama barang beserta kuantitas dan harga terhadap 30 nota sebesar 89.99%.
- 10) Rata-rata tingkat akurasi dari proses ocr hingga pengambilan nama barang agar dapat digunakan sebesar 94.2%
- 11) Implementasi cekmutasi.co.id untuk OVO, BCA, dan Mandiri memiliki proses yang sama. Namun, untuk menggunakan OVO mempunyai tingkat bahaya lebih tinggi, yang disebabkan adanya fitur transfer yang disediakan.

## DAFTAR PUSTAKA

- [1] G. Nagy, T. A. Nartker, and S. V Rice, "Optical character recognition: An illustrated guide to the frontier," in *Document Recognition and Retrieval VII*, 1999, vol. 3967, pp. 58–69.
- [2] L. Moroney, Moroney, and Anglin, *Definitive Guide to Firebase*. Springer, 2017.
- [3] M. A. Mekar, S. O. Fageeri, and S. E. Fattoh, "Using firebase cloud messaging to control mobile applications," in *2019 International Conference on Computer, Control, Electrical, and Electronics Engineering (ICCEEE)*, 2019, pp. 1–5.
- [4] M. Mathias and J. Gallagher, *Swift Programming: The Big Nerd Ranch Guide*. Pearson Technology Group, 2016.
- [5] D. K. Barry, *Web services, service-oriented architectures, and cloud computing*. Elsevier, 2003.
- [6] A. Sunardi and others, "MVC architecture: A comparative study

- between laravel framework and slim framework in freelancer project monitoring system web based,” *Procedia Comput. Sci.*, vol. 157, pp. 134–141, 2019.
- [7] D. Mulfari, A. Celesti, M. Fazio, M. Villari, and A. Puliafito, “Using Google Cloud Vision in assistive technology scenarios,” in *2016 IEEE symposium on computers and communication (ISCC)*, 2016, pp. 214–219.
- [8] A. Bud, “Facing the future: The impact of Apple FaceID,” *Biometric Technol. today*, vol. 2018, no. 1, pp. 5–7, 2018.
- [9] J. Chaki and N. Dey, *A beginner’s guide to image preprocessing techniques*. CRC Press, 2018.
- [10] A. S. B. Nugroho and M. S. Al Ihsan, “Aplikasi Startup Dokter Sepatu Menggunakan Pembayaran Online Realtime Berbasis Web Dan Android,” in *Seminar Nasional Riset Terapan*, 2019, vol. 4, pp. A25--A31.

# Ekstraksi Informasi Berbasis Rule untuk Proceeding, Jurnal, dan Technical Report dengan Memanfaatkan Attribute Font dan Paragraf

Christian Aditya Santoso, *Teknik Informatika, Institut Sains dan Teknologi Terpadu Surabaya, Gunawan, Teknik Informatika, Institut Sains dan Teknologi Terpadu Surabaya.*

**Abstrak**—Digital library merupakan solusi yang baik untuk dunia edukasi. Hal ini disebabkan karena buku yang sudah berevolusi menjadi digital. Awalnya dalam bentuk fisik sekarang sudah dalam bentuk digital dengan ekstensi PDF. Namun untuk membangun sebuah digital library merupakan system yang besar dan kompleks, sehingga diperlukan bagian yang banyak. Penelitian ini mengambil satu bagian dari pengembangan system digital library, yaitu pada bagian preprocessing atau persiapan sumber data digital library. Penyediaan sumber data digital library sangat luas dan banyak. Fokus dari penelitian ini adalah penyediaan data dimana data tersebut adalah jurnal, prosiding dan paper. Dokumen tersebut dipilih karena dinilai memiliki manfaat yang besar untuk edukasi karena peneliti mendokumentasikan hasil penelitian pada dokumen tersebut. Dalam 1 paper tentunya ada bagian yang menjadi kunci yang menggambarkan intisari dari penelitian tersebut. Pada penelitian ini diambil informasi Judul, Abstract, Keyword dan penulis. Informasi tersebut dipercaya mampu menggambarkan intisari dari suatu paper. Proses dilakukan dengan terbagi menjadi 3 bagian besar yaitu konversi file mentah dengan ekstensi PDF menjadi file JSON, Proses pengambilan fitur, Proses ekstraksi informasi. Ekstraksi informasi pada penelitian ini menggunakan kumpulan rule yang diimplementasikan pada software. Rule didapatkan dari hasil pengamatan selama penelitian. Hasil dari penelitian dilakukan perhitungan dengan memberikan bobot dimana hal yang terberat memiliki pengaruh yang lebih besar. Ketelitian yang dicapai adalah 81.32% dimana dari hipotesa awal pada ketelitian 80%. Namun masih banyak pengembangan yang bisa dilakukan agar lebih baik lagi pada penelitian selanjutnya

**Kata Kunci**—Ekstraksi Inforasmi, Rule, Jurnal, Prosiding, paper.

## I. PENDAHULUAN

Perpustakaan merupakan media untuk mencari informasi yang lengkap dan menjadi penunjang informasi dalam proses belajar mengajar pada dunia edukasi. Namun dalam hal penyediaan sumber pada perpustakaan tidaklah mudah.

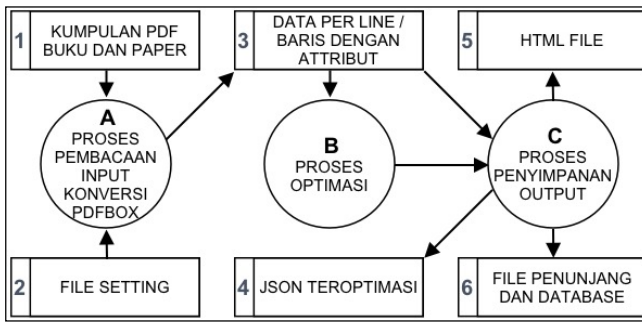
Christian Aditya Santoso, Informatics Department, Institut Sains dan Teknologi Terpadu Surabaya, Surabaya, Jawa Timur, Indonesia

Gunawan, Department of Information Technology, Institut Sains dan Teknologi Terpadu Surabaya, Surabaya, Jawa Timur, Indonesia (e-mail: gunawan@stts.edu)

Apabila menggunakan metode konvensional tentunya harus membeli banyak sekali buku. Permasalahan lebih rumit ketika sudah mengalami perkembangan teknologi dimana semua sudah serba digital dan kemudahan mendapatkan informasi menjadi prioritas. Hal ini juga terjadi pada perpustakaan maka tercetuslah perpustakaan digital. Perpustakaan digital tidak mudah untuk diwujudkan karena kompleks dan memiliki banyak bagian. Namun apabila tidak dimulai maka juga tidak akan terwujud. Penelitian ini mengambil salah satu bagian dari digital library yaitu penyediaan data sumber untuk diekstraksi informasinya guna dilakukan proses clustering yang digunakan untuk melakukan pencarian yang handal pada mesin pencarian perpustakaan digital. Proses yang akan dilakukan adalah konversi file pdf menjadi data terlebih dahulu pada penelitian ini dipilih JSON. Apabila file dinilai sebagai jurnal ataupun prosiding maka akan ada langkah lebih lanjut yaitu mencari nomor file JSON dari masing-masing prosiding atau jurnal. Langkah selanjutnya dilakukan dengan pengambilan fitur dari masing-masing element yang akan di ekstraksi dalam penelitian ini adalah judul, abstrak, keyword dan penulis. Setelah fitur terkumpul maka dilakukan proses ekstraksi informasi sehingga bisa menghasilkan data yang siap untuk di cluster.

## II. PROSES KONVERSI PDF DAN TAGGING PAPER TUNGGAL PADA JURNAL DAN PROSIDING

Seperti yang dibahas pada pendahuluan proses ini merupakan proses pertama yang dilakukan dalam penelitian ini. Konversi PDF dilakukan dengan menggunakan library PDFBox dimana PDFBox merupakan library cross platform. Pada penelitian ini software ditulis dalam Bahasa JAVA. Proses pada bagian ini akan dibahas menjadi 2 hal yaitu konversi PDF dan tagging paper tunggal pada jurnal dan prosiding.



Gambar. 1. Gambar Arsitektur Proses Konversi PDF

### A. Proses Konversi PDF

Gambar 1 merupakan alur kerja dari Proses Konversi PDF. Pada gambar 1 terdapat 3 Subproses yang terjadi pada proses Konversi PDF. Subproses pertama adalah Proses Pembacaan Input Konversi PDF Box pada gambar 1 diwakili Proses A. Subproses kedua adalah Subproses Optimasi pada gambar 1 diwakili oleh Proses B. Subproses ketiga adalah Subproses Penyimpanan Output pada gambar 1 diwakili oleh Proses C.

Proses A merupakan proses pertama yang dijalankan oleh sistem. Input dari proses A adalah file dengan ekstensi PDF pada suatu folder serta file setting dimana file setting ini ditujukan untuk memudahkan penggantian folder pembacaan. Proses konversi dari PDF menjadi file JSON dilakukan dengan bantuan PDF Box. Dimana kinerja PDFBox adalah membaca baris perbaris dari PDF File. Namun dari pembacaan baris-perbaris juga di dapatkan informasi seperti letak dalam x dan y koordinat, serta attribute font family dan font size. Dari informasi tersebutlah digunakan sebagai informasi yang dijadikan input proses B pada gambar 1.

Proses B, merupakan proses lanjutan dari proses Konversi PDF setelah menghasilkan array dengan isi text dan attribute untuk masing-masing baris pada masing-masing halaman PDF, maka masih diperlukan langkah selanjutnya yaitu optimasi. Secara umum optimasi yang dilakukan adalah untuk membuat data baris perbaris menjadi kumpulan paragraf, sehingga tidak terpisahkan berdasar baris.

Proses optimasi dilakukan perlu dilakukan karena informasi yang dihasilkan oleh Subsistem 1 ini tidak bisa hanya informasi dalam masing-masing baris contohnya adalah abstract, bisa kita ketahui bahwa abstract pasti memiliki lebih dari satu baris, sehingga akan sangat susah sekali apabila data yang akan dikonversi ini masih dalam wujud perbaris. Contoh lain adalah judul, banyak sekali paper penelitian yang memiliki judul penelitian lebih dari satu baris sangat jarang sekali bisa ditemukan ada judul penelitian yang hanya memiliki 1 baris. Dengan permasalahan tersebut maka diputuskan untuk mengembangkan proses optimasi ini. Harapan yang dihasilkan dari optimasi ini adalah data yang awalnya terpisah dari tiap barisnya, akan bisa menjadi bergabung dalam paragraf-paragraf. Hal ini tentunya akan mempermudah proses ekstraksi informasi yang akan dilakukan pada proses selanjutnya.

Proses C merupakan proses terakhir pada gambar 1, bagian ini merupakan proses penyimpanan hasil output. File yang disimpan cukup banyak yaitu, File PDF yang sudah di rename, File JSON dari masing-masing halaman yang telah

dikonversi, File txt non optimized, File HTML non Optimized serta gambar dari halaman pertama jurnal, prosiding dan paper. Hal ini dilakukan agar proses pemeriksaan bisa dilakukan dengan mudah dan cepat.

### B. Proses Tagging paper Tunggal pada Jurnal dan Prosiding

Jurnal dan prosiding memiliki model perlakuan yang berbeda dengan paper tunggal. Hal ini disebabkan oleh jurnal maupun prosiding merupakan kumpulan dari beberapa paper tunggal hal ini yang menjadi dasar dilakukannya proses ini. Proses ini masih ada melalui proses manual yaitu mendata semua judul-judul paper tunggal pada daftar isi dan dilakukan penyimpanan pada database. Setelah proses manual tersebut dilakukan maka dicarilah File JSON yang merupakan halaman pertama dari masing-masing paper tunggal.

## III. PROSES EKSTRAKSI FITUR

Proses Ekstraksi Fitur merupakan proses yang akan dijalankan terlebih dahulu sebelum dilakukan ekstraksi informasi [1]. Secara software yang diimplementasikan menjadi satu software mandiri dari software yang ada di section B, bahkan juga menjadi software yang berbeda dari Proses-proses sebelumnya. Ekstraksi Fitur ini merupakan bagian yang mencoba mengenali bagian-bagian element yang akan diekstraksi informasinya. Element yang akan diekstraksi adalah sebagai berikut :

- Judul Paper : merupakan nama yang dipakai untuk buku atau bab dalam buku yang dapat menyiratkan secara pendek isi atau maksud buku atau bab itu.
- Abstract Paper : merupakan sebuah ringkasan isi dari sebuah karya tulis ilmiah yang ditujukan untuk membantu seorang pembaca agar dapat dengan mudah dan cepat untuk melihat tujuan dari penulisannya.
- Kata Kunci Paper : merupakan kata-kata yang mengandung konsep pokok yang dibahas dalam karya tulis.
- Penulis : merupakan seseorang atau beberapa orang yang melakukan penelitian dan membuat laporan hasil penelitian dalam bentuk paper / karya ilmiah.

Dengan penjelasan tersebut maka dipilih empat informasi tersebut untuk dipilih karena tujuan dari penelitian ini adalah untuk menghasilkan informasi yang menggambarkan intisari dari isi suatu paper ilmiah sehingga bisa dilakukan clustering atau pengelompokan data pada subsistem tiga. Dari empat informasi tersebut hal yang dapat menggambarkan mengenai intisari dari paper pada sesungguhnya hanya pada informasi judul paper, abstrak paper, dan kata kunci paper. Penulis di ekstraksi untuk tujuan pembuatan front end aplikasi pencarian data. Namun pada penelitian ini hanya sebatas hanya bisa mengekstrasi data penulis.

Proses Ekstraksi fitur ini menggunakan metode pengenalan akan kata kunci. Masing-masing element yang akan diambil informasinya sehingga software memiliki kumpulan kata kunci untuk masing-masing element yang akan diekstraksi informasinya. Kata kunci tersebut didapatkan dengan proses pengamatan selama penelitian, sehingga ditemukan

kumpulan kata kunci tersebut untuk mengenali masing-masing element.

TABEL I  
DAFTAR KATA KUNCI

| Pendahuluan     | Keyword             | Abstrak           |
|-----------------|---------------------|-------------------|
| Introduction    | .key.word           | abstract          |
| 1. Introduction | key.word            | abstracts         |
| 1.              | (.*)key(.*)word(.*) | a b s t r a c t   |
| Pendahuluan     | keyword             | a b s t r a c t s |
| 1. Pendahuluan  | keywords            | abstrak           |
| I               | key-words           |                   |
| i               | key-word            |                   |
| I. Introduction | key words           |                   |
| i. Introduction | key word            |                   |
| I. Pendahuluan  | index terms         |                   |
| i. Pendahuluan  | index term          |                   |
| Background      | indexterms          |                   |
|                 | indexterm           |                   |
|                 | index-terms         |                   |
|                 | index-term          |                   |

Tabel 1 merupakan daftar kata kunci yang diterapkan untuk pencarian element. Pada table tidak terdapat untuk kata kunci judul dan juga kata kunci penulis. Hal ini dikarenakan untuk dua element tersebut dilakukan metode dan cara berbeda untuk mengekstraksi fitur yang dibutuhkan. Setelah dilakukan proses ekstraksi fitur maka dilakukan ekstraksi informasi pada masing-masing element.

#### IV. PROSES EKSTRAKSI INFORMASI

Proses Ekstraksi informasi merupakan inti dari penelitian ini. Proses ekstraksi informasi terbagi menjadi 4 bagian besar yaitu ekstraksi informasi pada bagian judul, abstract, keyword dan author. Proses ekstraksi informasi menggunakan rule dimana rule ini dihasilkan dan disimpulkan dari proses penelitian. Berbagai penelitian sebelumnya telah dilakukan untuk pengolahan teks [2], [3] dan ekstraksi informasi [4]

##### A. Ekstraksi Informasi Judul

Proses ekstraksi informasi dari element judul berbeda dengan element yang lain karena element ini tidak dilakukan pengenalan element terlebih dahulu [5]–[8]. Element judul langsung dilakukan ekstraksi informasi. Namun untuk melakukan ekstraksi tersebut selama penelitian tetap mengumpulkan fakta-fakta yang bisa dijadikan acuan untuk mengidentifikasi judul. Fakta-fakta yang berhasil dikumpulkan adalah sebagai berikut :

- Judul selalu berada dibagian awal
- Judul selalu memiliki font yang paling besar
- Judul selalu memiliki tipe font Bold
- Judul selalu dituliskan sebelum element author
- Judul dituliskan dengan Upper case.

Dengan didapatkan fakta mengenai tata cara penulisan serta peletakan element judul maka di berikan beberapa rules yang digunakan untuk melakukan ekstraksi informasi element judul. Berdasar 5 fakta yang disimpulkan dari semua fakta tersebut diinterpretasikan kedalam software. Implementasi ke dalam software dilakukan dengan membuat rule-rule yang apabila di eksekusi akan mampu menghasilkan output yang baik. Rule yang diterapkan pada bagian judul disajikan pada table 2.

TABEL 2  
DAFTAR RULE

| No | Rule  | Keterangan   |
|----|---|--|
| 1  | Pencarian Font Size Terbesar pada halaman pertama             | Disimpulkan bahwa judul di tuliskan dalam Font Size terbesar pada halaman pertama  |
| 2  | Apakah Abstrak sudah terdeteksi?                              | Rule ini untuk menentukan batas pencarian  |
| 3  | Apakah Keyword sudah terdeteksi?                              | Rule ini untuk menentukan batas pencarian  |
| 4  | Apakah Introduction sudah terdeteksi?                         | Rule ini untuk menentukan batas pencarian  |
| 5  | Pencarian paragraf dengan font size terbesar                  | Rule untuk mengenali paragraf yang memiliki font terbesar dan di jadikan sebagai Judul   |
| 6  | Apakah hasil dari no 5 memiliki karakter lebih dari 30        | Rule ini digunakan untuk mendeteksi apakah yang terdeteksi merupakan benar judul, karena judul tidak mungkin hanya 30 karakter |
| 7  | Pencarian dilanjutkan dengan mengambil fontsize ke 2 terbesar | Proses ini dilakukan sampai Langkah 6 menghasilkan jumlah karakter cukup   |

Dengan menggunakan 7 rule tersebut maka pencarian judul dilakukan. Masing-masing rule memiliki peran tersendiri. Rule ini masih terus bisa di eksplorasi dan ditambahkan agar ketelitian bisa terus meningkat.

##### B. Ekstraksi Informasi Abstrak

Metode yang digunakan untuk pengambilan informasi pada abstrak berbeda dengan metode pengambilan informasi pada judul [9]. Hal itu disebabkan pengenalan pada abstrak menggunakan output informasi dari subproses B. Namun prediksi dari subproses B masih belum selesai karena ketika diambil informasinya masih belum 100% lengkap dan benar. Berikut adalah masalah-masalah yang harus diselesaikan ketika melakukan ekstraksi informasi pada bagian abstrak.

- Terdapat penulisan kata kunci abstrak yang tidak tergabung dengan abstrak, sehingga hal ini akan membuat pengenalan dari subproses B masih belum memuat konten abstrak
- Terdapat penulisan kata kunci abstrak namun memiliki style berbeda dengan konten abstrak
- Terdapat penulisan abstrak yang diikuti oleh copyright penulisan
- Terdapat penulisan abstrak yang bergabung dengan keyword
- Terdapat penulisan abstrak yang keyword penanda abstrak bergabung dengan konten abstrak.

Dari 5 masalah tersebut yang harus diselesaikan pada subproses ekstraksi informasi untuk abstrak ini. Sehingga dapat dihasilkan abstrak yang rapi dan memuat semua konten dari abstrak. Dengan baiknya output yang dihasilkan tentunya akan membantu proses selanjutnya untuk bisa menghasilkan output yang optimal.

Permasalahan yang cukup banyak ditemui adalah banyak sekali keyword abstrak yang dituliskan sebagai judul subbab pada suatu paper ilmiah, penanganan untuk permasalahan ini adalah software akan membaca terlebih dahulu data dari table paperfeature dan melakukan filter data untuk pdf yang akan diekstrak informasinya kemudian dilakukan filter dengan jenisfeature abstract. Langkah pertama setelah mendapatkan data tersebut adalah dengan cara melihat jumlah karakternya apabila jumlah karakternya sedikit (dalam kasus ini di



lakukan pengecekan lebih kecil dari 10). Berikut adalah rule yang digunakan untuk mengekstraksi bagian abstrak.

TABEL 3  
DAFTAR RULE

| No | Rule  | Keterangan   |
|----|---|--|
| 1  | Pencarian Keyword Penentu Abstrak, daftar keyword terdapat pada tabel 5.5                               | Disimpulkan bahwa judul di tuliskan dalam Font Size terbesar pada halaman pertama  |
| 2  | Apakah Index Abstrak Lebih Kecil dari Index Introduction?   | Rule ini untuk menentukan batas pencarian  |
| 3  | Apakah Akhir pencarian lebih besar dari index keyword dan index keyword lebih besar dari index abstrak? | Rule ini untuk menentukan batas pencarian  |
| 4  | Apakah index keyword lebih besar dari index abstrak?  | Rule ini untuk menentukan batas pencarian  |
| 5  | Apakah start karakter keyword = 0?  | Apabila tidak 0 maka artinya keyword di tuliskan Bersama dengan keyword  |
| 6  | Apakah index abstrak sama dengan index pencarian?   | Apabila true maka system akan mencari sampai index abstrak + 1   |
| 7  | Apakah Index Abstrak = 0  | Kasus ini jarang terjadi namun apabila terjadi maka system akan menentukan pencarian adalah index email + 1  |
| 8  | Setelah di temukan start dan akhir pencarian system akan menggabungkan data text menjadi satu kesatuan  | Proses ini digunakan apabila abstrak terpecah menjadi beberapa object JSON   |
| 9  | Apakah Keyword kata kunci ditemukan dalam text gabungan?  | Apabila true maka akan ada proses substring data dari awal sampai index kata kunci   |
| 10 | Apakah Keyword abstrak ditemukan dalam text gabungan?   | Apabila ditemukan maka akan ada proses substring untuk menghilangkan keyword tersebut  |
| 11 | Apakah ada karakter © pada text gabungan?   | Apabila ditemukan maka akan ada proses substring karena karakter tersebut merupakan informasi copyright dari publisher dimana hal tersebut tidak termasuk di dalam abstrak |

Abstract merupakan rule yang paling banyak, karena bagian ini adalah bagian yang paling sukar untuk diekstraksi. Dengan menggunakan rule tersebut mendapat hasil yang cukup memuaskan. Dengan ditambah rule tentunya akan meningkatkan ketelitian dari ekstraksi informasi bagian abstrak.

### C. Ekstraksi Informasi Keyword

Keyword merupakan element yang juga bisa menggambarkan isi dari suatu paper ilmiah sehingga bagian ini juga penting untuk dilakukan ekstraksi informasi. Element keyword pada sesungguhnya juga merupakan bagian dari abstrak dan tidak selalu pada setiap paper ilmiah. Sama halnya terjadi pada abstrak element ini hasil output dari proses belum bisa langsung dilakukan ekstraksi informasi. Hal itu dikarenakan konsep tata tulis yang tidak 100% standar, sehingga pada bagian ini masih akan dilakukan pengolahan lebih lanjut agar hasil output untuk element ini lebih optimal. Hasil yang optimal tentunya akan meningkatkan performansi untuk proses selanjutnya. Berikut adalah permasalahan yang akan diselesaikan pada proses ekstraksi element keyword.

- Penulisan kata kunci penanda element keyword yang menjadi satu dengan konten keyword
- Penulisan keyword yang bergabung dengan abstrak

Selama proses penelitian ini dilakukan terdapat dua permasalahan ini yang harus diselesaikan proses ekstraksi informasi untuk element keyword. Pada subproses ini akan menyelesaikan dua permasalahan tersebut sistem mampu menghasilkan output yang baik. Berikut adalah rule yang digunakan untuk mengekstraksi keyword

TABEL 4  
DAFTAR RULE

| No | Rule   | Keterangan   |
|----|--|--|
| 1  | Pencarian Keyword Penentu Abstrak, daftar keyword terdapat pada tabel 5.3                            | Disimpulkan terdapat keyword tertentu yang selalu dituliskan sebelum dituliskan kata kunci   |
| 2  | Apakah index kata kunci == 0?  | Apabila True maka pada paper tersebut disimpulkan tidak ada kata kunci karena terdapat paper yang memang tidak ada kata kunci sebagai element informasi. |
| 3  | Apakah index kata kunci memiliki karakter < 30?  | Hal ini digunakan untuk memutuskan apakah kata kunci terbagi menjadi 2 baris atau tidak  |
| 4  | Apakah index introduction di temukan?  | Index introduction di butuhkan untuk menentukan batas pencarian.   |
| 5  | Sistem akan melakukan iterasi atau tidak sampai pada batas tertentu berdasar hasil dari rule nomor 3 | Iterasi ini digunakan untuk mengambil kumpulan text bagian kata kunci  |
| 6  | Apakah keyword kata kunci dimulai dari index 0?  | Apabila false maka dilakukan substring start dari awal keyword sampai akhir text   |
| 7  | Apakah terdapat keyword kata kunci pada text?  | Jika true maka system akan melakukan substring kembali dari akhir  |

Ekstraksi Keyword membutuhkan 7 rule. Keyword merupakan salah element yang memiliki nilai ketelitian baik. Namun masih ditemukan beberapa kendala dan tentunya pengembangan masih bisa terus dilakukan

### D. Ekstraksi Informasi Author

Proses ekstraksi element author merupakan ekstraksi informasi terakhir. Output dari proses ini adalah berupa satu paragraf yang berisi informasi tentang detil dari author. Untuk mengidentifikasi element ini menggunakan bantuan NER (Name Entity Recognizer) [10]. Berbeda dengan metode yang akan dipakai pada bagian ini. Sebelum dilakukan ekstraksi informasi terlebih dahulu dilakukan pengamatan tata letak penulisan element author ini. Berikut adalah beberapa fakta yang disimpulkan untuk element author.

- Author ditulis setelah penulisan title
- Author ditulis sebelum penulisan abstrak
- Pada element author juga terdapat email dari masing-masing author

Dengan fakta tersebut maka disusunlah software untuk mengadopsi fakta-fakta yang ditemukan sehingga bisa menjadi dasar penunjang ekstraksi informasi yang

akan dilakukan. Berikut adalah table rule yang digunakan untuk ekstraksi informasi pada author

TABEL 5  
DAFTAR RULE

| No | Rule  | Keterangan   |
|----|---|--|
| 1  | Apakah sudah mendapat index judul?                          | Syarat agar hasil menjadi optimal adalah proses ini dilakukan setelah proses ekstraksi judul     |
| 2  | Apakah indexabstrak < indexkeyword                          | Jika true maka batas pencarian sampai abstrak apabila false batas pencarian sampai pada keyword. |
| 3  | Proses iterasi penggabungan informasi author menjadi 1 text | Proses ini digunakan untuk menggabungkan text author dimana hasil selalu                         |

Rule untuk bagian author merupakan rule yang paling sedikit. Author di ekstraksi namun tidak terlalu mendetail karena informasi mengenai author hanya digunakan sebagai pelengkap informasi. Bagian ini masih bisa dibedah lebih jauh lagi.

### V. UJI COBA

Pada bagian ini akan membahas mengenai hasil dari penelitian yang telah dilakukan. Berawal dari data. Data yang dijadikan uji coba adalah sebesar 1080 data dimana dari 1080 data terdapat 4 pdf yang tidak berhasil di konversi sehingga menjadi 1076 data. Berikut adalah rincian mengenai data tersebut.

TABEL 6  
KOMPOSISI DATA

| No    | Jenis Dokumen      | Jumlah Dokumen        |
|-------|--------------------|-----------------------|
| 1     | Jurnal / Prosiding | 3 (167 Paper Tunggal) |
| 2     | Paper Tunggal      | 909                   |
| TOTAL |                    | 1076                  |

Dari 1076 dokument yang terekstraksi informasinya memiliki beberapa model penulisan paper yang diekstraksi. Variasi tersebut juga menjadi salah satu bagian dari tantangan pengujian system yang diteliti berikut adalah variasi yang dilakukan.

TABEL 7  
KOMPOSISI DATA

| No    | Jenis Dokumen | Jumlah Dokumen |
|-------|---------------|----------------|
| 1     | Tipe 1        | 91             |
| 2     | Tipe 2        | 21             |
| 3     | Tipe 3        | 55             |
| 4     | Tipe 4        | 119            |
| 5     | Tipe 5        | 408            |
| 6     | Tipe 6        | 65             |
| 7     | Tipe 7        | 36             |
| 8     | Tipe 8        | 11             |
| 9     | Tipe 9        | 270            |
| TOTAL |               | 1076           |

Pada Tabel 7 terdapat 9 tipe penulisan paper yang menjadi target ujicoba pada penelitian ini. Tipe 1 sampai tipe 3 mewakili masing-masing jurnal ataupun prosiding, Tipe 4 paper tunggal namun juga memiliki kemiripan, Tipe 6, namun pada tipe 6 masih terdapat varian di dalamnya namun dikategorikan menjadi 1 kategori. Tipe 7 juga merupakan paper tunggal namun memiliki kemiripan dan bukan hanya dari 1 publisher. Hal sama juga pada tipe 7. Berbeda dengan Tipe 8 terdapat 11 dokumen namun dari 11 dokumen tersebut memiliki ciri khas model format penulisan yang berbeda-

beda. Tipe 8 merupakan tipe yang digunakan sebagai kategori penulisan format namun pada waktu pencarian sumber data, paper yang masuk ke dalam tipe 8 ini penulis tidak menemukan banyak, sehingga di kategorikan menjadi 1 kategori, tipe 9 merupakan tipe terakhir. Berikut adalah table sampling dari hasil pengujian mengenai score yang diberikan pada masing-masing data.

TABEL 8  
SCORE SAMPLE

| No   | Id PDF | Id TOC | Judul (0/1/2) | Penulis (0/1/2) | Abstrak (0/1/2) | Kata kunci (0/1/2) |
|------|--------|--------|---------------|-----------------|-----------------|--------------------|
| 1    | 1      | 0      | 2             | 2               | 2               | 2                  |
| 2    | 2      | 0      | 2             | 2               | 2               | 2                  |
| 3    | 3      | 0      | 2             | 2               | 2               | 2                  |
| 4    | 4      | 0      | 2             | 2               | 2               | 2                  |
| 5    | 5      | 0      | 2             | 2               | 2               | 2                  |
| 6    | 6      | 0      | 2             | 2               | 2               | 2                  |
| 7    | 7      | 0      | 2             | 2               | 2               | 2                  |
| 8    | 8      | 0      | 2             | 2               | 1               | 0                  |
| 9    | 9      | 0      | 2             | 2               | 2               | 2                  |
| 10   | 10     | 0      | 2             | 2               | 2               | 2                  |
| **** | ****   | ****   | ****          |                 |                 |                    |
| 290  | 265    | 31     | 2             | 2               | 2               | 2                  |
| 291  | 265    | 32     | 2             | 2               | 2               | 2                  |
| 292  | 265    | 33     | 2             | 2               | 2               | 2                  |
| 293  | 265    | 34     | 2             | 2               | 2               | 2                  |
| 294  | 265    | 35     | 2             | 2               | 0               | 2                  |
| 295  | 265    | 36     | 2             | 2               | 2               | 2                  |
| 296  | 265    | 37     | 2             | 2               | 2               | 2                  |
| 297  | 265    | 38     | 2             | 2               | 2               | 2                  |
| 298  | 265    | 39     | 2             | 2               | 2               | 2                  |
| 299  | 265    | 40     | 2             | 2               | 2               | 2                  |
| **** | ****   | ****   | ****          |                 |                 |                    |
| 850  | 712    | 0      | 2             | 2               | 2               | 2                  |
| 851  | 713    | 0      | 2             | 2               | 2               | 2                  |
| 852  | 714    | 0      | 2             | 2               | 2               | 2                  |
| 853  | 715    | 0      | 2             | 2               | 2               | 2                  |
| 854  | 716    | 0      | 2             | 2               | 2               | 2                  |
| 855  | 717    | 0      | 2             | 2               | 2               | 2                  |
| 856  | 718    | 0      | 2             | 2               | 2               | 2                  |
| 857  | 719    | 0      | 2             | 2               | 2               | 2                  |
| 858  | 720    | 0      | 2             | 2               | 2               | 2                  |
| 859  | 721    | 0      | 2             | 2               | 2               | 2                  |

Tabel 6.6 merupakan potongan dari hasil pencocokan / koreksi data ekstraksi informasi dari system. no 1 – 10 merupakan paper tunggal ditandai dengan id\_toc bernilai 0, hal serupa sama dengan no 850 – 859 merupakan paper tunggal. Sedangkan no 500 – 509 memiliki id\_pdf yang sama namun id\_tocnya tidak 0 menandakan data pada bagian itu adalah prosiding atau jurnal. Poin terpenting ada pada kolom score. Pengisian score dilakukan dengan cara sebagai berikut:

- Apabila judul tidak terdeteksi sama sekali maka akan mendapat score sebesar 0
- Apabila judul terdeteksi namun tidak sempurna maka akan mendapat score sebesar 1
- Apabila judul terdeteksi dan menghasilkan informasi yang benar maka akan mendapat score sebesar 2

Dengan menggunakan acuan tersebut maka score akan dihitung dan diolah untuk menghasilkan nilai ketelitiannya. Pada proses ujicoba dilakukan lebih dari 1x running data (1) pengolahan dan pendeteksian 1000 data. Namun terdapat ditentukan dulu rumus untuk melakukan perhitungan ketelitian. Rumus yang dimaksud adalah

$$Percentage (\%) = \frac{Sum(Score\ Judul)}{2 * Jumlah\ Data} \times 100$$

Pada Rumus 6.1 merupakan cara perhitungan untuk mencari nilai percentage ketelitian. Secara umum perhitungan dilakukan dengan cara menjumlahkan semua score yang didapat dan kemudian dibagi dengan jumlah score penuh dalam kasus ini adalah 2 karena poin maximum untuk masing-masing judul dan jumlah data yang digunakan pada penelitian ini adalah 1076 data. Setelah dihitung maka dikalikan dengan 100 untuk mendapat nilai persen untuk ketelitian judul. Berikut adalah perhitungan final untuk penelitian ini

TABEL 8  
HASIL FINAL

| Keterangan    | Judul | Penulis | Abstrak | Kata Kunci   |
|---------------|-------|---------|---------|--------------|
| Score         | 1761  | 1873    | 1862    | 1933         |
| Score Penuh   | 2160  | 2160    | 2160    | 2160         |
| Percent       | 81,53 | 86,71   | 86,20   | 89,49        |
| Rule          | 7     | 3       | 11      | 7            |
| Total Rule    | 28    | 28      | 28      | 28           |
| Percent Bobot | 20,38 | 9,29    | 33,87   | 22,37        |
| <b>TOTAL</b>  |       |         |         | <b>85,91</b> |

Konsep perhitungan yang disajikan adalah persentase yang didapatkan dihitung dengan bobot jumlah rule yang mempengaruhi masing-masing elemen, sehingga elemen yang rumit menjadi penentu porsi paling besar dalam perhitungan. Dalam Tabel 6.8 dapat disimpulkan ketelitian yang dicapai pada penelitian ini adalah sebesar 85,91 %. Hipotesa dari penelitian ini adalah ketelitian mencapai angka 80 %. Sehingga disimpulkan bahwa penelitian ini memenuhi kriteria pada hipotesa awal.

## VI. KESIMPULAN

Setelah melakukan proses penelitian yang panjang dan menemukan beberapa solusi. Penelitian yang diawali dengan ide dan terealisasi dalam system yang dikembangkan. Kemudian juga dilakukan pengujian. Hal tersebut menghasilkan beberapa kesimpulan diantaranya adalah:

1. Penulisan paper / karya ilmiah memiliki format yang bisa di generalisasi dan memiliki kemiripan yang signifikan dari masing-masing penerbit. Hal ini menjadi penunjang yang baik untuk penyusunan rule base system karena system tidak dituntut untuk terus menyesuaikan dengan perubahan yang sangat dinamis.
2. Selain format penulisan urutan dari masing-masing elemen paper yang memiliki aturan yang general juga sangat membantu dalam proses ekstraksi informasi pada system.
3. Pengambilan informasi judul, abstrak, dan kata kunci menjadi pilihan yang sangat tepat untuk proses clustering dalam pencarian kesamaan isi dari paper. Pada 3 informasi tersebutlah tersimpan ini dan isi dari suatu paper.
4. Kumpulan Rule dan Algoritma yang disusun oleh pakar dengan menggunakan Attribut Font dan juga

Susunan Paragraf terbukti mampu mengekstraksi informasi untuk judul, abstrak, katakunci dan penulis dengan baik.

5. Fitur pencarian kata kunci juga menjadi penunjang yang sangat baik untuk mengenali bagian-bagian dari paper.
6. Deteksi Penulis dengan hanya menggunakan NER tidak akan pernah maksimal karena tata cara penulisan bagian penulis sangat bervariasi font attributnya sehingga mengakibatkan menjadi lebih rumit untuk di gabungkan menjadi 1 kesatuan
7. Subsystem konversi PDF dengan mengoptimasi pengenalan paragraph dengan menggunakan Attribut Font terbukti membantu mempermudah proses pengenalan element dan ekstraksi informasi
8. Subsystem Ekstraksi Informasi menggunakan rule dapat menjadi sebuah solusi yang cukup baik untuk ekstraksi informasi pada paper karena sudah memiliki struktur yang cukup baku, dengan ketelitian yang berhasil dicapai adalah 82.17%
9. Alat bantu PDFBox yang digunakan sangat membantu dalam proses konversi file sumber dengan ekstensi .pdf menjadi data yang bisa diolah lebih baik.
10. Encoding yang digunakan untuk penyimpanan pada database harusnya UTF-8 karena pada paper terdapat karakter-karakter khusus yang tentunya akan mengganggu proses penyimpanan.
11. Format JSON yang bisa di sematkan dalam bentuk object sangat membantu dalam proses ekstraksi informasi karena proses pencarian tidak diperlukan lagi query ke database namun bisa langsung dilakukan pada level program dengan memanfaatkan data yang sudah berbentuk JSON.

## DAFTAR PUSTAKA

- [1] L. Chiticariu, Y. Li, and F. R. Reiss, "Rule-based information extraction is dead! Long live rule-based information extraction systems!," 2013.
- [2] E. Lim, E. I. Setiawan, and J. Santoso, "Stance Classification Post Kesehatan di Media Sosial Dengan FastText Embedding dan Deep Learning," *J. Intell. Syst. Comput.*, vol. 1, no. 2, pp. 65–73, 2019.
- [3] M. A. Rahman, H. Budiarto, and E. I. Setiawan, "Aspect Based Sentimen Analysis Opini Publik Pada Instagram dengan Convolutional Neural Network," *J. Intell. Syst. Comput.*, vol. 1, no. 2, pp. 50–57, 2019.
- [4] S. N. Soenardjo and G. Gunawan, "Information Extraction Berbasis Rule Untuk Soal Ujian," *J. Intell. Syst. Comput.*, vol. 2, no. 1, pp. 28–33, 2020.
- [5] K. Yao, "Header Extraction from Scientific Documents."
- [6] J. Beel, B. Gipp, A. Shaker, and N. Friedrich, "SciPlore Xtract: extracting titles from scientific PDF documents by analyzing style information (Font Size)," in *International Conference on Theory and Practice of Digital Libraries*, 2010, pp. 413–416.
- [7] J. Beel, S. Langer, M. Genzmehr, and C. Müller, "Docear's PDF Inspector: Title Extraction from PDF Files," in *Proceedings of the 13th ACM/IEEE-CS Joint Conference on Digital Libraries*, 2013, pp. 443–444, doi: 10.1145/2467696.2467789.
- [8] D. Meyerzon, Y. Cao, H. Li, Q. Zheng, and Y. Hu, "Automatic extraction of titles from general documents using machine learning," in *Proceedings of the 5th ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL '05)*, 2005, pp. 145–154, doi: 10.1145/1065385.1065418.
- [9] L. Kovriguina, A. Shipilo, F. Kozlov, M. Kolchin, and E. Cherny, "Metadata extraction from conference proceedings using template-based approach," in *Communications in Computer and*

*Information Science*, 2015, vol. 548, doi: 10.1007/978-3-319-25518-7\_13.

- [10] F. Peng and A. McCallum, "Information extraction from research papers using conditional random fields," *Inf. Process. \& Manag.*, vol. 42, no. 4, pp. 963–979, 2006.

# Pembuatan Turn Based Tactical Game “Hex World” Dilengkapi Dengan Artificial Intelligence

Albert Rianka Kristian, *Teknik Informatika Institut Sains dan Teknologi Terpadu Surabaya*,  
Hendrawan Armanto, *Teknik Informatika Institut Sains dan Teknologi Terpadu Surabaya*

**Abstrak**—Tactical video game merupakan salah satu genre game yang mengutamakan perencanaan dan pemikiran yang mahir sebagai daya tarik game tersebut. Game tactical sering disertai dengan artificial intelligence sebagai lawan bermain agar pemain bisa menikmati game tersebut secara keseluruhan tanpa perlu bantuan pemain lain. Selain artificial intelligence, ada beberapa fitur-fitur lain yang menunjang pengalaman bermain, seperti skenario permainan beserta batasan-batasannya, goal yang harus dicapai, terrain dari papan permainan, dan karakter yang unik yang disediakan oleh game. Artificial intelligence merupakan komponen utama dari game tactical sehingga harus dibuat sedemikian rupa untuk dapat meningkatkan pengalaman bermain dari pemain. Kombinasi dari algoritma Behavior Tree, Influence Map, dan Utility AI dapat membuat artificial intelligence yang memadai yang dapat digunakan dalam game tactical. Dengan adanya fitur tambahan seperti terrain dan karakter yang unik, programmer sudah bisa membuat game tactical yang cukup menarik dan menantang.

**Kata Kunci**—Behavior Tree, Influence Map, Tactical Video Game, Utility AI.

## I. PENDAHULUAN

Tactical video game merupakan salah satu genre game yang mulai muncul sekitar tahun 1980 di Jepang. Tactical video game adalah genre game yang mengutamakan perencanaan dan pemikiran yang mahir untuk dapat memenangkan permainan. Biasanya tipe game ini dimainkan dalam bidang *isometric* dan lawan bermainnya merupakan *artificial intelligence* yang disediakan oleh game. Tidak seperti genre video game yang lain, tactical video game cenderung tidak menyediakan fitur *multiplayer* dimana seorang pemain bisa bermain dengan pemain lain dalam suatu skenario dan waktu yang sama.

Permainan taktik yang disediakan oleh genre game tactical adalah daya tarik utama dari game tersebut. Permainan taktik tersebut didapatkan saat pemain melakukan *battle* terhadap musuh. Untuk memaksimalkan pengalaman bermain game tactical, maka artificial intelligence dari lawan bermain harus dibuat sedemikian rupa sehingga pemain dapat menikmati permainan taktik tersebut. Selain dari artificial intelligence

dari lawan bermain, game tactical juga menawarkan fitur-fitur lain yang menunjang permainan taktik. *Terrain* atau tempat bermain yang unik dari setiap skenario, *goal* yang harus dicapai dalam suatu skenario, batasan-batasan dari suatu skenario, keuntungan atau kerugian yang didapat dalam awal skenario dan karakter beserta *skill* yang unik adalah beberapa fitur yang juga ada dalam game tactical sebagai penunjang permainan taktik.

Bermain game tactical merupakan salah satu cara untuk mengembangkan daya pikir dari pemain. Dengan tipe permainan taktik, pemain diharuskan untuk berpikir cepat dan efisien selama battle berlangsung agar bisa memenangkan battle tersebut. Bila musuh bermain semakin pintar, pemain juga dituntut untuk berpikir lebih keras. Melalui game tactical ini, pemain tidak hanya mendapat kesenangan dari bermain game, tetapi juga melatih daya berpikir kritis serta pengambilan keputusan dari pemain.

## II. TEORI DASAR

Bab ini akan membahas tentang teori yang berhubungan dengan judul penulisan buku. Bab ini menjelaskan tentang apa itu game mobile, Unity, Influence Map, dan Behavior Tree.

### A. Pengertian Game Mobile

Game mobile adalah salah satu jenis video game yang dimainkan di smartphone, smartwatch, tablet computer, dan media elektronik portabel lainnya. Game mobile yang terkenal pertama kali adalah Tetris pada alat Hagenuk MT-2000 pada tahun 1994. Game mobile terdahulu sudah langsung terinstal ke dalam media portabel sejak media portabel tersebut dirilis. Sekarang, game mobile bisa didapatkan dengan download dari *app store* ataupun dari internet.

Terdapat banyak macam game mobile, ada yang bisa dimainkan oleh banyak pemain sekaligus, ada juga yang memanfaatkan teknik lokalisasi seperti GPS, ataupun lewat media bluetooth. Untuk kali ini, game mobile yang akan dibuat adalah game mobile yang bisa dimainkan oleh satu orang saja. Hal ini dikarenakan tema game mobile yang akan

Albert Rianka Kristian, Departemen Informatika, Institut Sains dan Teknologi Terpadu Surabaya, Surabaya, Jawa Timur, Indonesia (e-mail: albert.rianaka@gmail.com)

Hendrawan Armanto, Departemen Informatika, Institut Sains dan Teknologi Terpadu Surabaya, Jawa Timur, Indonesia (e-mail: hendrawan@sts.edu)

dibuat adalah jenis RPG yang menekankan peranan seorang pemain di dalam jalan cerita game mobile ini.

**B. Unity**

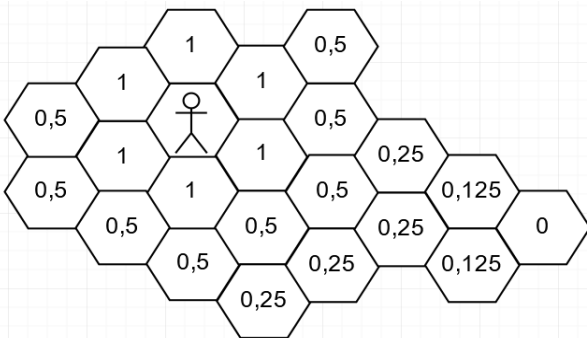
Unity adalah salah satu game engine untuk banyak platform yang dikembangkan oleh Unity Technologies dan digunakan untuk mengembangkan video game berbasis PC, konsol, mobile, dan juga website. Pada awalnya, Unity hanya ditujukan untuk platform OS X pada tahun 2005, dan sampai sekarang sudah merambah ke 21 jenis platform. Nintendo menyediakan lisensi gratis dari Unity 5 kepada semua developer game Nintendo beserta dengan *software development kits* (SDK) untuk Wii U dan Nintendo 3DS.

**C. Influence Map**

*Influence Map* adalah sebuah algoritma yang mengubah dunia game yang kompleks menjadi serangkaian data yang dapat digunakan atau dipakai dalam pengambilan keputusan oleh artificial intelligence. Influence map menyediakan tiga tipe informasi yang berguna dalam pengambilan keputusan:

- 1) *Situation Summary*: informasi hasil rangkuman dari semua detail-detail kecil yang mudah untuk dipahami.
- 2) *Historical Statistic*: informasi yang didapatkan dari kejadian beberapa waktu lalu.
- 3) *Future Prediction*: informasi yang berisi prediksi masa depan, seperti kemana musuh akan bergerak dan dampak dari gerakan tersebut.

Dalam game ini Influence Map menggambarkan bagaimana setiap karakter mempunyai pengaruh terhadap area di sekitarnya. Pengaruh suatu karakter semakin besar pada area yang dekat dengan dirinya dan akan memudar pada area yang lebih jauh. Ilustrasi pengaruh Influence Map dapat dilihat pada Gambar 1.



Gambar 1. Contoh nilai pengaruh pada Influence Map

**D. Behavior Tree**

*Behavior Tree* adalah program yang menyerupai struktur pohon yang berisi banyak *node* yang mengontrol pengambilan keputusan dari sebuah AI [1]–[5]. Anak paling kecil dari sebuah pohon, yaitu *leaf*, adalah aksi yang menggerakkan AI, dan *branch* adalah berbagai macam node yang mengontrol AI untuk mencapai aksi yang paling baik untuk kondisi saat itu.

Behavior Tree terdiri dari beberapa tipe node, tiga tipe utama dari node Behavior Tree yaitu *Composite*, *Decorator*, *Leaf*. Ketiga tipe utama tersebut memiliki fungsi utama, yaitu

tiap node mengembalikan *status*. Terdapat tiga status utama yang dikembalikan oleh node, *Success*, *Failure*, *Running*. Status Success atau Failure mengembalikan nilai berhasil atau gagal kepada *parent* node tersebut, dan status Running menandakan bahwa node tersebut masih dalam proses menentukan status Success atau Failure.

III. ANALISA GAME SEJENIS

Bab ini akan membahas tentang game-game taktik yang sejenis dengan game yang dibuat dalam pengerjaan penelitian ini. Game-game tersebut memiliki beberapa fitur yang sama dan ada juga perbedaan dari game yang dibuat. Game-game tersebut antara lain *Legion War*, *Langrisser Mobile*, dan *Brigandine*.

**A. Legion War**

*Legion War* merupakan salah satu game taktik mobile dengan sistem *turn-based* yang dikembangkan oleh Bear&Cat Studio. Game *Legion War* ini memiliki grafis 2D dengan tile segi enam sebagai terrainnya. Gambar 2 merupakan contoh tampilan battle dari game *Legion War*. Dalam game tersebut pemain diberikan beberapa unit awal dan pemain bisa membeli unit baru dengan uang yang tersedia. Kondisi pemain memenangkan permainan adalah dengan mengalahkan semua musuh dan pemain akan kalah bila semua unit pemain habis.



Gambar 2. Battle Game Taktik “Legion War”

**B. Langrisser Mobile**

*Langrisser Mobile* adalah game penerus dari seri game *Langrisser* yang rilis pertama kali pada tahun 1987 di konsol *Mega Drive*. *Langrisser Mobile* ini merupakan game taktik yang bisa dimainkan pada *smartphone* dengan sistem operasi Android ataupun iOS. Game ini memiliki alur cerita dan banyak skenario permainan yang bisa dimainkan.



Gambar 3. Tampilan Battle “Langrisser Mobile”

Gambar 3 merupakan contoh tampilan dari game Langrisser Mobile. Setiap unit memiliki giliran masing-masing untuk beraksi. Pemain bisa memilih dari beberapa aksi yang tersedia, yaitu bergerak, menyerang biasa, menggunakan skill, dan diam di tempat. Pergerakan dan jangkauan serang dari tiap unit bisa berbeda-beda tergantung dari tipe dan class unit tersebut.

C. Brigandine

Brigandine merupakan game taktik yang dirilis oleh Atlus pada tahun 1998 pada konsol PlayStation. Game ini menggunakan tile segi enam sebagai papan bermainnya dan memiliki beberapa tipe terrain dasar seperti tanah dan air. Sebelum memainkan game ini, pemain harus memilih dari enam negara untuk dikendalikan oleh pemain. Setiap negara memiliki satu unit raja, beberapa unit kapten beserta monster yang dibawahinya kapten tersebut, dan beberapa kota yang dikuasai. Lima negara lain yang tidak dipilih oleh pemain akan dikendalikan oleh AI. Tujuan akhir game ini adalah menguasai seluruh kota dalam dunia game ini.



Gambar. 4. Tampilan Battle pada “Brigandine”

Sebelum memulai pertarungan, pemain dan AI akan memilih maksimal tiga unit kapten untuk dimainkan. Setiap kapten memiliki maksimal enam unit monster yang dibawahinya. Kelompok unit akan bergerak secara bergiliran sesuai dengan urutan level dari kapten. Kapten yang memiliki level paling tinggi akan mendapat giliran lebih utama. Saat giliran suatu kelompok unit tiba, kapten dan seluruh unit yang dibawahinya bisa melakukan aksi seperti bergerak, menyerang, atau menggunakan skill. Setiap unit memiliki nomor di samping gambar unit tersebut yang mendandakan unit tersebut termasuk dalam kelompok mana. Setelah semua unit dalam kelompok unit itu sudah melakukan aksi, maka kelompok unit dengan level kapten tertinggi selanjutnya yang akan melakukan aksi. Permainan akan berakhir saat salah satu kubu kehabisan pasukan karena dikalahkan oleh musuh atau melarikan diri. Contoh tampilan pertarungan dapat dilihat pada Gambar 4.

D. Perbandingan dengan Game Sejenis

Ketiga game yang sudah dibahas sebelumnya merupakan game taktik yang memiliki fitur-fitur yang unik. Fitur-fitur dari ketiga game tersebut akan dibandingkan dengan fitur dari game yang akan dibuat pada penelitian ini. Perbedaan dan persamaan fitur-fitur tersebut bisa dilihat pada Tabel 1.

TABEL I  
PERBANDINGAN FITUR

| Fitur           | Legion War                           | Langrisser Mobile          | Brigandine                                 | Hex World                                  |
|-----------------|--------------------------------------|----------------------------|--|--|
| Kondisi Terrain | Statis, ditentukan di awal           | Statis, ditentukan di awal | Statis, ditentukan di awal                 | Dinamis, bisa berubah saat bermain         |
| Sisi Terrain    | 6 sisi                               | 4 sisi                     | 6 sisi                                     | 6 sisi                                     |
| Unit            | Non-unik                             | Unik                       | Unik dan non-unik                          | Non-unik                                   |
| Level Unit      | Ada                                  | Ada                        | Ada  | Tidak ada                                  |
| GerakUnit       | Bebas dengan batasan                 | Bebas dengan batasan       | Bebas dengan batasan                       | Bebas dengan batasan                       |
| Serangan Unit   | Sesuai tipe unit dan skill           | Sesuai tipe unit dan skill | Sesuai tipe unit dan skill                 | Sesuai tipe unit dan skill                 |
| Sprite Unit     | 2 sisi                               | 2 sisi                     | 6 sisi                                     | 2 sisi                                     |
| Animasi Unit    | Attack, Skill                        | Idle, Move, Attack, Skill  | Idle, Move, Attack, Skill                  | Idle, Attack, Skill                        |
| Perputaran Aksi | Seluruh unit milik pemain lalu musuh | Masing-masing unit         | Setiap unit dalam satu grup lalu grup lain | Setiap unit dalam satu grup lalu grup lain |

IV. DESAIN GAME

Pada bab ini akan dijelaskan mengenai desain game dari penelitian ini. Desain game akan dibagi menjadi tiga bagian. Bagian pertama akan menjelaskan tentang *main menu*. Bagian kedua akan menjelaskan tentang *gameplay*. Bagian ketiga akan menjelaskan tentang *tactic*.

A. Main Menu

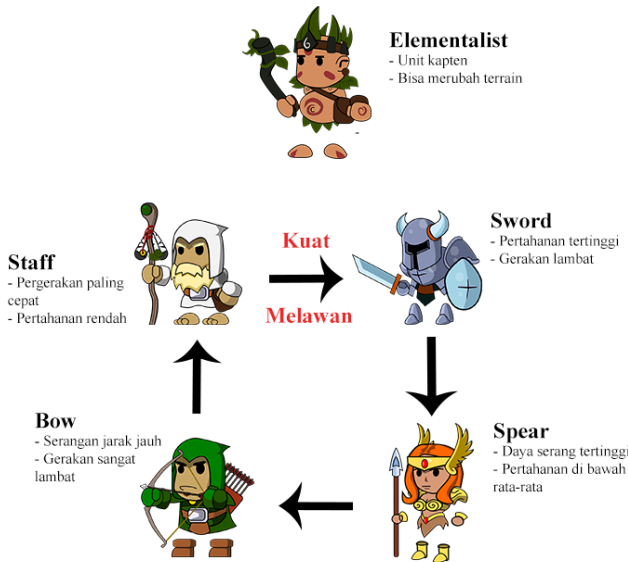
Pada main menu terdapat logo dari game yang dibuat dan tiga tombol yang berfungsi untuk berpindah scene serta satu tombol yang berfungsi untuk keluar dari permainan. Tampilan main menu bisa dilihat pada Gambar 5.



Gambar. 5. Tampilan Main Menu

**B. Gameplay**

Bagian gameplay merupakan bagian utama dari game yang dibuat. Pemain akan melawan AI sesuai dengan tingkat kesulitan yang telah dipilih sebelumnya. Terdapat lima tipe unit yang bisa dipakai oleh pemain untuk mengalahkan musuh. Kelima tipe unit tersebut adalah *Elementalist*, *Sword*, *Spear*, *Bow* dan *Staff*. Masing-masing tipe unit memiliki kelebihan dan kekurangannya sendiri. Tampilan kelima tipe unit tersebut bisa dilihat pada Gambar 6.



Gambar. 6. Deskripsi Singkat Tipe Unit

Sebelum memulai permainan, pemain diharuskan untuk memilih unit mana yang akan digunakan untuk bermain. Pertama-tama pemain menempatkan unit *Elementalist* terlebih dahulu di area yang sudah disediakan, dilanjutkan dengan menempatkan unit bawahan lain di area sekitar *Elementalist*. Ilustrasi dari formasi ini bisa dilihat pada Gambar 7.

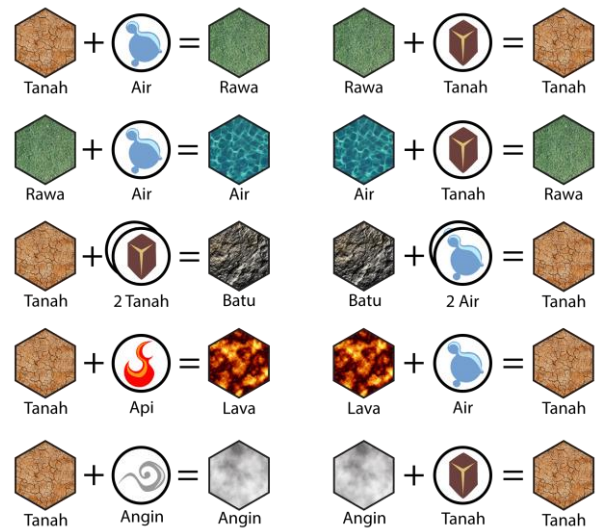


Gambar. 7. Pengaturan Formasi

Papan permainan terdiri dari beberapa tipe tile. Kondisi awal papan berbeda-beda sesuai dengan tingkat kesulitan yang dipilih oleh pemain. Pemain dan AI bisa merubah tile dengan unit *Elementalist* yang dimilikinya. Tiap tingkat kesulitan memiliki batasan tipe tile yang bisa dimainkan. Untuk tingkat kesulitan Easy, kondisi awal papan serta perubahan yang bisa dilakukan terhadap tile di papan hanya

dibatasi tiga tipe tile, yaitu tile Tanah, tile Rawa, dan Tile Air. Setiap giliran *Elementalist* bisa merubah salah satu tile papan sesuai dengan pilihan pemain. Perubahan tersebut memerlukan *Element* yang telah disediakan pada awal permainan. *Element* yang dibutuhkan untuk merubah tile berbeda-beda, tergantung dari tipe tile sebelum dirubah dan tipe tile tujuan yang dipilih. Kombinasi tipe tile awal dan *Element* bisa dilihat pada Gambar 8.

Jumlah *Element* yang diperlukan bersifat kumulatif. Sebagai contoh, bila pemain ingin merubah tile Tanah menjadi tile Air, maka pemain memerlukan total dua *Element* air untuk bisa merubah tile Tanah menjadi tile Air. Pemain perlu memperhatikan jumlah *Element* yang dibutuhkan karena jumlah dari *Element* yang disediakan di awal permainan terbatas dan tidak bisa didapatkan selama game berlangsung.



Gambar. 8. Kombinasi Tipe Tile dan Element

**C. Tactic**

Menu *Tactic* merupakan menu *tutorial* dari game ini dimana pemain akan dijelaskan cara bermain serta kontrol dari game ini. Pemain juga diberitahukan info mengenai unit dan tile yang ada. Info dan penjelasan mengenai game ini diberikan melalui gambar-gambar disertai petunjuk. Contoh tampilan dari *Tactic* bisa dilihat pada Gambar 9.



Gambar. 9. Tampilan Tactic

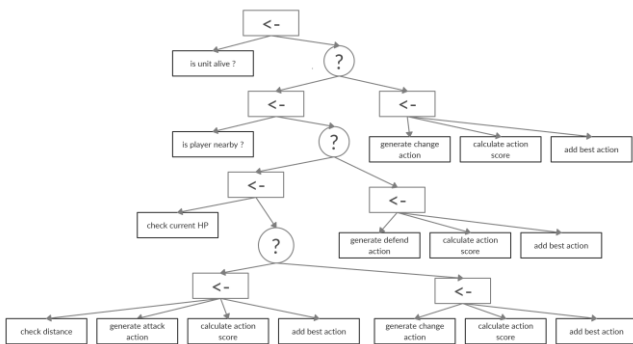


V. DESAIN ARTIFICIAL INTELLIGENCE

Pada bab ini akan dijelaskan mengenai desain AI dari penelitian ini. Desain AI akan dibagi menjadi empat bagian. Bagian-bagian tersebut antara lain implementasi Behavior Tree, Utility AI, Influence Map dan keseluruhan dari AI.

A. Desain Behavior Tree

Elementalist memiliki Behavior Tree yang berbeda dibanding dengan unit lain. Perbedaan ini dikarenakan Elementalist memiliki satu *action* unik yang tidak dimiliki oleh unit lainnya, yaitu *action change*. Behavior Tree Elementalist dapat dilihat pada Gambar 10.



Gambar. 10. Behavior Tree Elementalist

Behavior Tree kedua yang dibuat dalam game ini adalah Behavior Tree untuk unit selain Elementalist. Perbedaan antara kedua Behavior Tree ini terletak pada pemilihan actionnya, dimana pada Behavior Tree non Elementalist tidak terdapat pemilihan *action change* pada semua anak tree karena unit selain Elementalist tidak bisa menggunakan *action change*.

B. Desain Utility AI

Utility AI pada game ini digunakan pada penilaian list *action change*, *attack* maupun *defend*. Penilaian dilakukan berdasarkan berbagai macam faktor, antara lain sisa hit point dari unit, jarak yang ditempuh terhadap tile tujuan, tipe unit yang diserang, sisa hit point dari unit yang diserang, tipe tile tujuan yang akan diubah, dan *influence map*.

C. Desain Influence Map

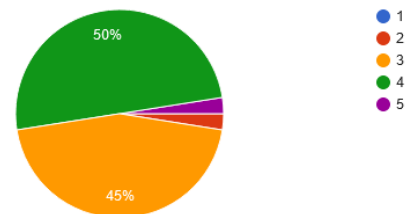
Nilai pada papan bergantung pada posisi tiap unit, baik unit AI maupun unit pemain, dan tipe tile papan tersebut. Pemberian nilai *Influence Map* didasarkan pada jarak suatu tile terhadap semua unit. Setiap unit pemain dan AI memberi pengaruh terhadap nilai *Influence Map* suatu tile. Unit pemain memberi nilai positif, sedangkan unit AI memberi nilai negatif.

VI. UJI COBA

Pada bab ini akan dijelaskan mengenai uji coba yang telah dilakukan pada game yang telah dibuat dalam pengerjaan penelitian ini. Uji coba yang dilakukan adalah perekaman permainan, dan uji coba berupa kuisisioner yang diberikan kepada beberapa *tester* melalui *Google Form*. Uji coba dilakukan untuk mengetahui apakah game yang telah dibuat sudah memiliki *gameplay* dan AI yang memadai.

Uji coba ini dilakukan dengan memberikan game kepada 40 tester lalu tester diberikan kuisisioner melalui *Google Form* untuk merekam *feedback* dari tester. Pertanyaan yang diajukan meliputi aspek tampilan, *gameplay*, AI dan panduan game. Pilihan jawaban merupakan nilai 1 sampai 5 dimana nilai 1 merupakan nilai terendah sedangkan nilai 5 merupakan nilai tertinggi.

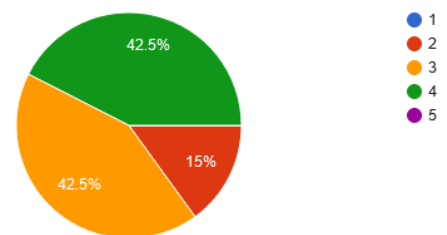
Berapakah nilai *gameplay* (sistem *gameplay*, pilihan unit) dari game ini ?  
40 responses



Gambar. 11. Nilai *Gameplay* Menurut Tester

Gambar 11 merupakan grafik response dari tester mengenai *gameplay* dari game yang telah dibuat. *Gameplay* yang dinilai oleh tester meliputi keseluruhan sistem dari battle dan pilihan unit yang tersedia. Pada grafik tersebut dapat dilihat bahwa sebagian besar dari tester menilai *gameplay* yang dibuat sudah cukup baik.

Berapakah nilai AI (tingkat kesulitan, pemilihan aksi) dari game ini ?  
40 responses



Gambar. 12. Nilai AI Game Menurut Tester

Gambar 12 merupakan grafik response dari tester mengenai AI dari game yang telah dibuat. AI yang dinilai oleh tester meliputi tingkat kesulitan dari AI dan pemilihan aksi yang diambil oleh AI. Pada grafik tersebut dapat dilihat bahwa sebagian besar dari tester menilai AI yang dibuat sudah cukup baik.

## VII. KESIMPULAN

Pada bagian ini dijelaskan kesimpulan yang didapat selama pembuatan penelitian dimana kesimpulan ini diperoleh dari hasil uji coba yang telah dilakukan. Kesimpulan tersebut adalah sebagai berikut:

- 1) Penggunaan Behavior Tree mempermudah implementasi AI dalam aspek pemilihan aksi secara garis besar.
- 2) Utility AI yang dikombinasikan dengan Influence Map sangat mempengaruhi tingkat kepintaran dari AI.
- 3) Aspek tampilan pada game yang telah dibuat sudah cukup bagus menurut tester, tetapi aspek AI dari game masih kurang memenuhi ekspektasi dari tester. Hal ini diketahui dari hasil kuisioner yang diperoleh, dimana 42.5% dari tester memberi nilai 3 poin dari nilai maksimum 5 poin.
- 4) Aspek panduan game yang telah dibuat dinilai kurang menurut tester. Hal ini diketahui dari hasil kuisioner yang diperoleh, dimana 55% dari tester member nilai 3 poin dari nilai maksimum 5 poin. Panduan game harus dibuat sedetail dan sejelas mungkin agar pemain dapat memahami secara menyeluruh sistem gameplay dari game yang telah dibuat.

## DAFTAR PUSTAKA

- [1] B. Schwab and B. Schwab, *AI game engine programming*. Charles River Media Hingham, 2004.
- [2] D. M. Bourg and G. Seemann, *AI for game developers*. “O’Reilly Media, Inc.,” 2004.
- [3] I. Millington and J. Funge, *Artificial intelligence for games*. CRC Press, 2018.
- [4] M. Buckland, *Programming game AI by example*. Jones & Bartlett Learning, 2005.
- [5] R. Hunicke, “The case for dynamic difficulty adjustment in games,” in *Proceedings of the 2005 ACM SIGCHI International Conference on Advances in computer entertainment technology*, 2005, pp. 429–433.

# STEERING, THROTTLE DAN BRAKE PREDICTION PADA SIMULATOR SELF-DRIVING CAR MEMANFAATKAN CNN

Andre Nyoto Raharjo, *Departemen Teknik Informatika, Institut Sains dan Teknologi Terpadu Surabaya,*  
Andreas, *Fakultas Ekonomi dan Bisnis, Universitas Pelita Harapan Surabaya.*

**Abstrak**— Manusia dapat mengendarai mobil dengan cara belajar sendiri, belajar dari orang lain atau dengan menirukan cara mengemudi orang lain. Dalam mengemudikan mobil pengemudi akan melihat kondisi jalan dan kondisi mobil sebagai pertimbangan untuk menentukan kontrol kemudi yang akan dijalankan seperti nilai steering angle, throttle dan brake. Dengan pembelajaran atau menirukan kebiasaan manusia, mesin juga mampu untuk melakukan hal yang sama yang dapat dilakukan oleh manusia seperti mengemudikan mobil. Maka dari itu akan dibuat program yang dapat mengemudikan mobil dengan menentukan nilai steering angle, throttle dan brake menggunakan data gambar kamera depan mobil, nilai steering angle, throttle, brake dan speed terakhir mobil. Aplikasi akan dibuat menggunakan Keras untuk pembuatan model, training dan testing. Berbagai teknik augmentasi gambar digunakan untuk menghasilkan gambar baru dengan jumlah tidak terbatas dari data gambar asli yang dikumpulkan. Program menggunakan simulator mobil untuk melakukan pengumpulan data dan uji coba program. Simulator dipilih sebagai media karena mudah dalam hal pengumpulan data dan tidak ada risiko terjadi kecelakaan seperti pada dunia nyata. Dengan adanya program ini, mesin dapat mengemudikan mobil secara otonom. Dari hasil eksperimen disimpulkan bahwa performa terbaik model tercapai pada model yang menggunakan dataset Lake dan Mountain, menggunakan augmentasi data, menggunakan channel warna YUV, menggunakan gambar dan state dari mobil sebagai input, learning rate 0.0001, dan drop out 0.5.

**Kata Kunci**—convolutional neural network, deep learning, self-driving car.

## I. PENDAHULUAN

Perkembangan dan penelitian di bidang informasi dan komputer menghasilkan banyak penemuan di berbagai bidang. Salah satu bidang yang mendapat efek besar adalah industri otomotif dan pengembangan kendaraan yang otonom sepenuhnya. Dengan adanya teknologi self-driving car dan sistem robot otomatis kompleks yang lain semakin banyak dijumpai pada kehidupan sehari-hari [1]. Banyak penelitian yang menyatakan bahwa teknologi tersebut dapat memudahkan kehidupan manusia sehari-hari. Sistem robot otomatis cepat digunakan pada kondisi yang berbahaya bagi manusia [2].

Data dalam jumlah besar dan komputasi yang besar dibutuhkan untuk menyelesaikan masalah pergerakan otomatis pada lingkungan. Solusi modern yang ada sekarang ini adalah dengan menggunakan sensor, perangkat keras dan program [3].

Salah satu solusi untuk menyelesaikan masalah tersebut adalah dengan menggunakan teknik algoritma Artificial Intelligence yang mampu mengontrol self-driving car dengan tidak lebih buruk dibandingkan dengan manusia berdasar data dalam jumlah besar yang disediakan[4].

Pengumpulan data dari jalan yang ada di dunia nyata tidaklah mudah dan membutuhkan waktu dan biaya yang banyak. Oleh sebab itu penggunaan media simulator adalah solusi terbaik untuk mengatasi masalah pengumpulan data. Dengan adanya simulator juga untuk menguji kemampuan self-driving car tidak perlu memikirkan risiko kecelakaan seperti pada dunia nyata

Makalah ini bertujuan untuk melakukan pengumpulan artificial data dan mensimulasikan proses mengendarai mobil menggunakan simulator. Diperlukan pembuatan arsitektur network yang mampu memprediksi nilai steering angle, throttle dan brake dari data gambar dan kondisi terakhir mobil yang terdiri dari data steering angle, throttle, brake dan speed terakhir dari mobil.

## II. CONVOLUTIONAL NEURAL NETWORK

Convolutional Neural Network (CNN) sangat berguna dalam hal pemrosesan gambar dan jenis neural network ini dapat diaplikasikan pada berbagai bidang. Contoh penggunaan CNN adalah computer vision, pengenalan wajah, pelabelan scene untuk pengenalan Tindakan, dan lain sebagainya. CNN merupakan arsitektur neural network yang khusus dibuat dengan tujuan untuk pengenalan gambar yang efektif [5]. CNN terdiri dari beberapa layer yang berbeda, seperti convolutional layer, subsampling layer dan fully connected layer [6].

Convolutional Neural Network berhasil digunakan pada masalah pengenalan pola pada video, dimana sistem tersebut sangat penting pada self-driving car [7]. Pada makalah ini penggunaan CNN adalah untuk melakukan ekstraksi fitur pada gambar kamera depan dari mobil [8]. Ekstraksi fitur tersebut ditujukan untuk sistem dapat mengenali bagian gambar yang merupakan jalan dan yang bukan jalan. Hasil dari ekstraksi tersebut kemudian akan digabungkan dengan data kondisi mobil untuk digunakan sebagai pertimbangan sistem untuk menentukan kontrol mobil.

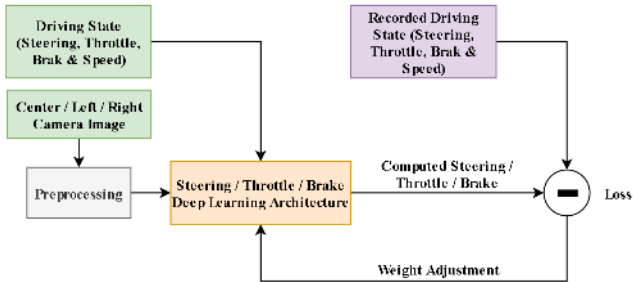
## III. METODOLOGI

Penelitian ini menggunakan bahasa pemrograman Python

<sup>1</sup> Andre Nyoto Raharjo, Departemen Teknik Informatika, Institut Sains dan Teknologi Terpadu Surabaya, Jl. Ngagel Jaya Tengah 73-77, Surabaya, Jawa Timur, Indonesia 60284 (e-mail: andrenyoto22@gmail.com)

Andreas, Department of Management, Faculty of Economy and Business, Universitas Pelita Harapan, Surabaya, Jawa Timur, Indonesia (e-mail: andreas.jodhinata@gmail.com)

yang dijalankan pada Anconda Virtual Environment. Proses pelatihan dan testing akan menggunakan library Keras [9] yang berjalan pada Tensorflow [10] backend.



Gambar. 1. Diagram proses training

Metode training yang digunakan dapat dilihat pada Gambar. 1. Detail proses dari metode tersebut adalah sebagai berikut:

1. Data gambar melalui tahap preprocessing
2. Data gambar hasil preprocessing dan data driving state menjadi input network
3. Network akan memprediksi nilai steering angle / throttle / brake
4. Hasil nilai prediksi model akan dibandingkan dengan nilai dari recorded driving state untuk mendapat nilai loss
5. Nilai loss yang didapat digunakan untuk mengubah weight pada network

Proses Training yang dilakukan memiliki jumlah iterasi maksimal 200, ukuran batch size 40, ukuran sampel setiap iterasi untuk data training adalah 400 dan sampel untuk data validasi training adalah 40. Training akan berakhir ketika mencapai iterasi maksimal atau selama 3 iterasi berurutan tidak terjadi perbaikan nilai validation loss.

| Layer (type)                | Output Shape        | Param # | Connected to                     |
|-----------------------------|---------------------|---------|----------------------------------|
| input_7 (InputLayer)        | (None, 66, 200, 3)  | 0       |                                  |
| lambda_4 (Lambda)           | (None, 66, 200, 3)  | 0       | input_7[0][0]                    |
| convolution0 (Conv2D)       | (None, 33, 100, 24) | 1824    | lambda_4[0][0]                   |
| convolution1 (Conv2D)       | (None, 15, 48, 36)  | 21636   | convolution0[0][0]               |
| convolution2 (Conv2D)       | (None, 6, 22, 48)   | 43248   | convolution1[0][0]               |
| convolution3 (Conv2D)       | (None, 4, 20, 64)   | 27712   | convolution2[0][0]               |
| convolution4 (Conv2D)       | (None, 2, 18, 64)   | 36928   | convolution3[0][0]               |
| dropout_10 (Dropout)        | (None, 2, 18, 64)   | 0       | convolution4[0][0]               |
| flatten_4 (Flatten)         | (None, 2304)        | 0       | dropout_10[0][0]                 |
| input_8 (InputLayer)        | (None, 4)           | 0       |                                  |
| concatenate_4 (Concatenate) | (None, 2308)        | 0       | flatten_4[0][0]<br>input_8[0][0] |
| dense1 (Dense)              | (None, 100)         | 230900  | concatenate_4[0][0]              |
| dropout_11 (Dropout)        | (None, 100)         | 0       | dense1[0][0]                     |
| dense2 (Dense)              | (None, 50)          | 5050    | dropout_11[0][0]                 |
| dropout_12 (Dropout)        | (None, 50)          | 0       | dense2[0][0]                     |
| dense3 (Dense)              | (None, 10)          | 510     | dropout_12[0][0]                 |
| output (Dense)              | (None, 1)           | 11      | dense3[0][0]                     |
| Total params: 367,819       |                     |         |                                  |
| Trainable params: 367,819   |                     |         |                                  |
| Non-trainable params: 0     |                     |         |                                  |

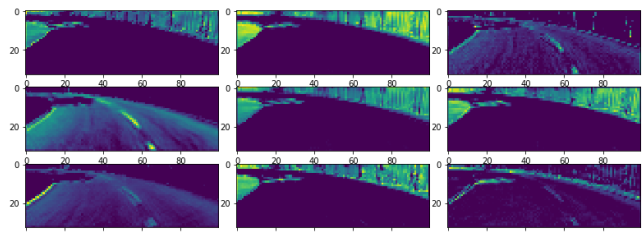
Gambar. 2. Arsitektur Network

### A. Arsitektur Deep Learning

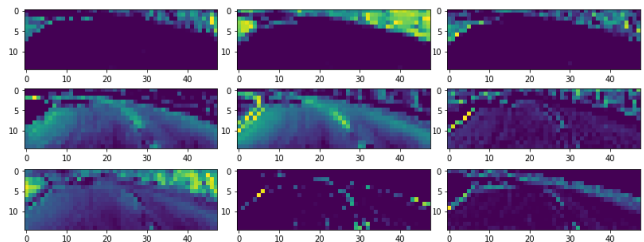
Arsitektur deep learning terbaik yang digunakan memiliki

17 layer. Detail susunan setiap layer dapat dilihat pada gambar summary model pada gambar. 2.

Pada gambar 2, Arsitektur network akan memiliki 2 input layer, 1 lambda layer, 5 convolutional layer, 1 flatten layer, 1 concatenate layer, 3 dropout layer, 3 fully connected layer dan 1 output layer. Convolutional0 sampai convolutional 4 dan dense1 sampai dense3 akan menggunakan ReLU sebagai activation function. Arsitektur tersebut akan menggunakan Adam sebagai optimizer. Input layer1 berupa gambar berukuran 66x200x3, 66 adalah tinggi gambar, 200 lebar gambar dan 3 adalah color channel. Contoh output layer Convolutional0 dan Convolutional1 dapat dilihat pada gambar 3 dan gambar 4



Gambar. 3. Output Convolutional0



Gambar. 4. Output Convolutional1

### B. Dataset

Dataset yang digunakan ada 2 macam, yaitu data gambar dan data numerik. Dataset didapatkan menggunakan simulator pada lintasan yang digunakan. Dataset yang gambar yang didapatkan berupa gambar kamera depan bagian kiri, tengah dan kanan. Kemudian untuk data numerik yang digunakan adalah data steering angle, throttle brake, dan speed. Dataset yang digunakan akan berasal dari 3 buah lintasan yang berbeda, yaitu lintasan lake, mountain dan castle. Contoh dataset lintasan lake dapat dilihat pada gambar .5 .



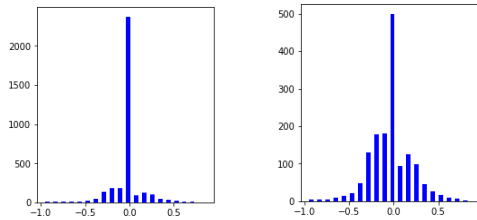
Gambar. 5. Diagram proses training

TABEL I  
PERBANDINGAN DATASET

| No | Data set | Ukuran Image | Total |
|----|----------|--------------|-------|
| 1  | Lake     | 320x160      | 3392  |
| 2  | Mountain | 320x160      | 6493  |
| 3  | Castle   | 320x160      | 6479  |

Pada tabel I dapat dilihat perbandingan jumlah dataset

yang digunakan. Dataset yang digunakan pada saat training akan melalui tahap perataan persebaran data. Hal ini dilakukan untuk mengurangi data dengan frekuensi yang sangat besar dibanding data yang lain. Setelah proses tersebut data akan dibagi menjadi 2, yaitu data training dan validation dengan perbandingan 80% data testing dan 20% data validation. Contoh perbandingan data asli dan hasil perataan dapat dilihat pada gambar.6 .



Gambar. 6. Distribusi data asli(kiri) dan hasil perataan(kanan)

### C. Preprocessing

Preprocessing adalah tahap yang harus dilakukan sebelum gambar digunakan sebagai input. Ada tiga tahap preprocessing yang digunakan. Pertama adalah crop untuk membuang bagian depan mobil dan langit pada gambar, pemotongan dilakukan sebesar 25px dari bawah dan 60px dari atas. Kedua adalah resize untuk mengubah ukuran gambar menjadi 200px lebar dan 66px tinggi. Proses terakhir adalah pengubahan color channel.

### D. Augmentation

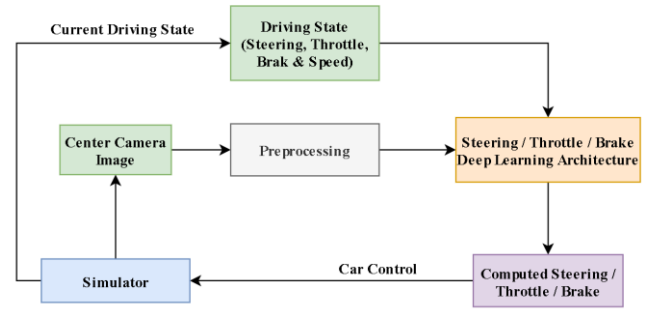
Augmentasi adalah teknik untuk membuat gambar baru dari gambar yang sudah ada. Teknik augmentasi yang dilakukan ada 2 macam, yaitu augmentasi dengan pengubahan data numerik dan tanpa pengubahan data numerik. Augmentasi dengan pengubahan data numerik adalah random choose image, flip, dan translate.

Untuk random choose gambar akan dipilih random dari gambar kiri, tengah dan kanan dan dilakukan pengubahan nilai steering sebesar +0.15 jika gambar kiri yang terpilih dan -0.15 jika gambar kanan yang terpilih. Pada augmentasi flip akan dilakukan pembalikan nilai steering angle, jika nilai awal positif akan dijadikan negatif dan sebaliknya.

Pada augmentasi translate akan dilakukan penambahan nilai steering sebesar  $-0.0015 \times$  jumlah pixel pergeseran ke kiri dan  $+0.0015 \times$  jumlah pixel pergeseran ke kanan. Augmentasi tanpa ada pengubahan data numerik meliputi shadow, brightness, fog, flare, rain, snow dan gravel.

### E. Ilustrasi Testing

Proses testing akan menggunakan model steering, throttle dan brake yang sudah dilatih untuk mengemudi mobil pada simulator.



Gambar. 7. Diagram proses testing

Metode testing yang digunakan pada makalah ini dapat dilihat pada Gambar. 7. Detail proses dari metode tersebut adalah sebagai berikut:

1. Program akan mengambil data gambar kamera depan, nilai steering angle, throttle, brake, dan speed terakhir mobil
2. Data gambar kamera depan melalui tahap preprocessing
3. Data gambar hasil preprocessing dan data driving state menjadi input network
4. Network akan memprediksi nilai steering angle / throttle / brake
5. Hasil nilai prediksi akan dikirimkan ke simulator sebagai car control
6. Proses berlangsung selamanya sampai diberhentikan secara manual

## IV. EKSPERIMEN

Terdapat 6 macam eksperimen yang dilakukan dan dari 5 macam tersebut terdapat 15 skenario eksperimen. Eksperimen yang dilakukan meliputi perbedaan input yang digunakan, perbedaan dataset yang digunakan, perbedaan color channel yang digunakan, perbedaan augmentasi yang digunakan, perbedaan parameter learning rate, dan perbedaan parameter dropout. Detail skenario eksperimen yang dilakukan dapat dilihat pada tabel II.

Uji coba dilakukan pada lintasan lake dan mountain. Setiap skenario uji coba akan diamati performa program untuk mengendarai mobil. Performa diukur dengan kemampuan program mengenali jalan untuk menentukan kendali yang diperlukan. Jumlah kejadian mobil keluar jalur atau menabrak dan jumlah kendali manual yang digunakan juga akan diamati. Semakin sedikit kendali manual yang diperlukan dan semakin sedikit mobil berjalan keluar jalur atau menabrak dianggap memiliki performa yang baik.

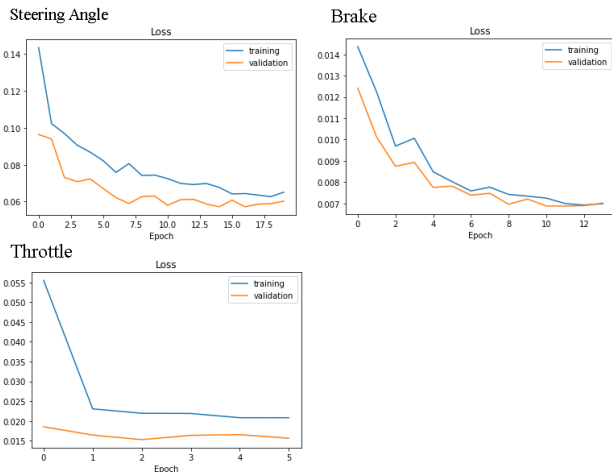
## V. HASIL EKSPERIMEN

Subbab ini akan membahas hasil eksperimen dari setiap skenario yang ada pada tabel. Dari semua skenario didapatkan bahwa skenario no 2 memiliki performa terbaik dari semua skenario.

TABEL II  
SKENARIO EKSPERIMEN

| No | Data set        | Aug-ment-ation | Color Channel | Input        | Learn-ing Rate | Drop Out   |
|----|-----------------|----------------|---------------|--------------|----------------|------------|
| 1  | LK+MT           | Some           | YUV           | <i>Im</i>    | 0.0001         | 0.5        |
| 2  | <i>LK+MT</i>    | <i>Some</i>    | <i>YUV</i>    | <i>Im+St</i> | <i>0.0001</i>  | <i>0.5</i> |
| 3  | <i>LK</i>       | Some           | YUV           | Im+St        | 0.0001         | 0.5        |
| 4  | <i>MT</i>       | Some           | YUV           | Im+St        | 0.0001         | 0.5        |
| 5  | <i>CS</i>       | Some           | YUV           | Im+St        | 0.0001         | 0.5        |
| 6  | <i>LK+MT+CR</i> | Some           | YUV           | Im+St        | 0.0001         | 0.5        |
| 7  | LK+MT           | <i>None</i>    | YUV           | Im+St        | 0.0001         | 0.5        |
| 8  | LK+MT           | <i>Full</i>    | YUV           | Im+St        | 0.0001         | 0.5        |
| 9  | LK+MT           | Some           | <i>RGB</i>    | Im+St        | 0.0001         | 0.5        |
| 10 | LK+MT           | Some           | <i>YCbCr</i>  | Im+St        | 0.0001         | 0.5        |
| 11 | LK+MT           | Some           | <i>CIELAB</i> | Im+St        | 0.0001         | 0.5        |
| 12 | LK+MT           | Some           | YUV           | Im+St        | <i>0.001</i>   | 0.5        |
| 13 | LK+MT           | Some           | YUV           | Im+St        | <i>0.0005</i>  | 0.5        |
| 14 | LK+MT           | Some           | YUV           | Im+St        | 0.0001         | <i>0.1</i> |
| 15 | LK+MT           | Some           | YUV           | Im+St        | 0.0001         | <i>1</i>   |

LK adalah Lake, MT adalah mountain, CS adalah Castle, CR adalah crash recovery, Im adalah Image, dan St adalah State(Steering, Throttle, Brake dan Speed).



Gambar. 8. Grafik loss dan val\_loss training model terbaik

Pada gambar.8 . adalah gambar grafik hasil training untuk steering angle, throttle, dan brake. Pada pelatihan steering angle proses traing berakhir pada epoch 20 dan hasil terbaik pada epoch 17 dengan nilai validation loss 0.0571828. Throttle selesai pada epoch 6 dan hasil terbaik pada epoch 3 dengan nilai validation loss 0.00152912. Brake selesai pada epoch 14 dengan hasil terbaik pada epoch 11 dengan nilai validation loss 0.0068767.

Hasil dari ketiga model tersebut dapat mengemudikan mobil pada kedua lintasan tanpa ada bantuan kendali manual. Pada lintasan lake mobil dapat menyelesaikan 1 lap dengan waktu 1 menit 19 detik dan pada lintasan mountain dengan waktu 2 menit 24 detik.

Eksperimen perbedaan dataset program tidak dapat mengemudikan mobil dengan baik pada lintasan yang belum diketahui sebelumnya. Penggunaan dataset castle memiliki performa terburuk dari dataset yang lain. Penambahan teknik crash recovery tidak memberikan kenaikan performa dibandingkan tanpa teknik crash recovery.

Eksperimen color channel, YCbCr memiliki performa yang sedikit di bawah color channel lain pada lintasan lake. RGB, CIELAB dan YUV memiliki performa yang cukup serupa pada lintasan lake. Pada lintasan mountain penggunaan color channel YUV memiliki performa terbaik dibanding yang lain.

Eksperimen perbedaan augmentasi, penggunaan augmentasi data dapat meningkatkan performa program untuk mengendarai mobil. Akan tetapi penggunaan augmentasi rain, snow, fog, flare dan gravel tidak memberikan peningkatan performa. Augmentasi tersebut tidak membantu banyak karena tidak sesuai dengan lintasan dan data gambar yang dipakai.

Eksperimen perbedaan nilai learning rate, semakin tinggi learning rate maka semakin cepat proses training dengan risiko penurunan performa. Eksperimen nilai perbedaan drop rate, nilai drop rate yang terlalu tinggi dan terlalu rendah dapat menurunkan performa. Hasil eksperimen pada setiap skenario dapat dilihat pada tabel III. Cuplikan hasil eksperimen dapat dilihat pada gambar 9 dan 10.

TABEL III  
PERFORMA HASIL EKSPERIMEN

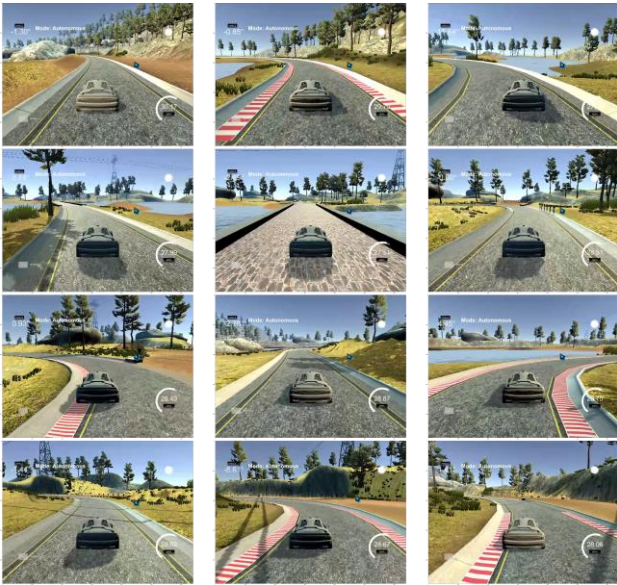
| No | Lake Track |           | Mountain Track |           |
|----|------------|-----------|----------------|-----------|
|    | No Crash   | No Manual | No Crash       | No Manual |
| 1  | X          | X         | X              | X         |
| 2  | ✓          | ✓         | ✓              | ✓         |
| 3  | X          | X         | X              | X         |
| 4  | X          | X         | X              | X         |
| 5  | X          | X         | X              | X         |
| 6  | X          | X         | X              | X         |
| 7  | X          | X         | X              | X         |
| 8  | X          | X         | X              | X         |
| 9  | ✓          | ✓         | X              | X         |
| 10 | X          | X         | X              | X         |
| 11 | ✓          | ✓         | X              | X         |
| 12 | X          | X         | X              | X         |
| 13 | X          | X         | X              | X         |
| 14 | X          | X         | X              | X         |
| 15 | X          | X         | X              | X         |

## VI. PENUTUP

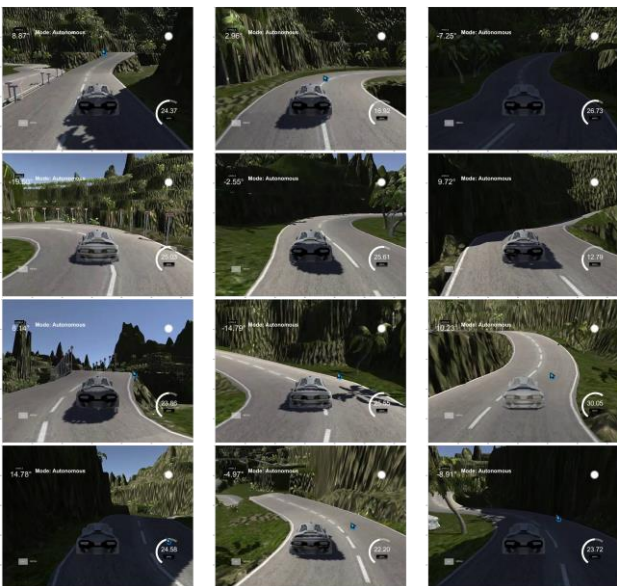
Bab ini akan berisi kesimpulan dan saran. Kesimpulan didapatkan berdasarkan hasil pengamatan proses uji coba yang telah dilakukan. Dari beberapa kesimpulan yang diambil, dapat dilihat keunggulan atau kekurangan dari program yang dibuat. Maka dari itu, bab ini akan memberikan saran untuk mengembangkan aplikasi serupa atau aplikasi lain dengan metode yang sama. Saran tersebut juga akan diberikan per poin.

### A. Kesimpulan

Pada subbab ini akan di bahas mengenai kesimpulan yang di dapat setelah pembuatan penelitian ini. Kesimpulan merupakan hasil yang didapatkan dari pengamatan performa program selama proses uji coba. Kesimpulan tersebut akan dijelaskan dalam beberapa poin sebagai berikut:



Gambar. 9. Gambar cuplikan self-driving dengan model scenario 2 pada lintasan lake



Gambar. 10. Gambar cuplikan self-driving dengan model scenario 2 pada lintasan mountain

1. Penggunaan teknik augmentasi gambar sangat membantu dalam hal mengatasi masalah terbatasnya jumlah training data. Augmentasi gambar mampu menghasilkan gambar-gambar baru dengan jumlah tak terbatas yang dapat memperkaya dataset yang digunakan untuk melatih model.
2. Teknik augmentasi flip, translate, shadow dan brightness dan penyesuaian nilai steering angle dapat meningkatkan performa program untuk mengendalikan mobil.
3. Teknik augmentasi snow, fog, rain, gravel, dan flare tidak memberikan perubahan performa yang signifikan pada lintasan danau dan penurunan performa pada lintasan gunung.
4. Penggunaan color channel YUV memiliki performa yang paling baik pada kedua lintasan yang digunakan. RGB, YCbCr dan CIELAB memiliki performa yang kurang lebih sama baik pada lintasan danau, dan pada lintasan gunung YCbCr dan CIELAB memiliki performa yang mirip dan RGB memiliki performa yang paling rendah.

5. Penggunaan input gambar dan input kondisi mobil meningkatkan kemampuan program untuk menentukan nilai steering angle, throttle, dan brake dibandingkan dengan hanya menggunakan input gambar saja.
6. Program belum berjalan baik pada lintasan yang baru/berbeda dimana data tersebut tidak terdapat pada saat training dan masih memerlukan pengembangan lebih lanjut
7. Penggunaan nilai drop out terlalu tinggi atau terlalu rendah dapat menurunkan performa model yang dilatih
8. Nilai learning rate semakin tinggi akan mempercepat proses training, tetapi pada saat yang bersamaan juga menurunkan performa model yang dilatih

**B. Saran**

Selain menarik kesimpulan dari pembuatan program ini, berdasarkan proses pembuatan dan uji coba terdapat saran yang mungkin berguna bagi pembaca dalam pembuatan aplikasi serupa. Beberapa saran yang ada antara lain :

1. Menggunakan gabungan arsitektur Convolutional Neural Network dengan arsitektur lain seperti Long-Short Term Neural Network dan Recurent Neural Network.
2. Menggunakan gambar dengan resolusi dan ukuran yang lebih besar.
3. Untuk pengembangan selanjutnya dapat menambahkan jenis-jenis lintasan lain sebagai data set, supaya program dapat mengenali berbagai macam lintasan.
4. Menggunakan ukuran batch yang lebih besar supaya program dapat mengenali fitur-fitur penting dengan lebih baik.

**DAFTAR PUSTAKA**

- [1] M. V Smolyakov, A. I. Frolov, V. N. Volkov, and I. V Stelmashchuk, "Self-driving car steering angle prediction based on deep neural network an example of CarND udacity simulator," in *2018 IEEE 12th international conference on application of information and communication technologies (AICT)*, 2018, pp. 1–5.
- [2] J. Kocić, N. Jovičić, and V. Drndarević, "An end-to-end deep neural network for autonomous driving designed for embedded automotive platforms," *Sensors*, vol. 19, no. 9, p. 2064, 2019.
- [3] M. Bojarski *et al.*, "End to end learning for self-driving cars," *arXiv Prepr. arXiv1604.07316*, 2016.
- [4] Y. Tian, K. Pei, S. Jana, and B. Ray, "Deeptest: Automated testing of deep-neural-network-driven autonomous cars," in *Proceedings of the 40th international conference on software engineering*, 2018, pp. 303–314.
- [5] T.-D. Do, M.-T. Duong, Q.-V. Dang, and M.-H. Le, "Real-time self-driving car navigation using deep neural network," in *2018 4th International Conference on Green Technology and Sustainable Development (GTSD)*, 2018, pp. 7–12.
- [6] A. Navarro, J. Joerdening, R. Khalil, A. Brown, and Z. Asher, "Development of an autonomous vehicle control strategy using a single camera and deep neural networks," 2018.
- [7] M. A. Nielsen, *Neural networks and deep learning*, vol. 25. Determination press San Francisco, CA, 2015.
- [8] M. Spryn, "Autonomous Driving using End-to-End Deep Learning: an AirSim tutorial," 2017. <https://github.com/Microsoft/AutonomousDrivingCookbook/tree/master/AirSimE2EDeepLearning>.
- [9] F. Chollet, "Keras: The Python Deep Learning library," *Keras.Io*, 2015.
- [10] M. Abadi *et al.*, "TensorFlow: A system for large-scale machine learning," 2016.

# Pemanfaatan 3D U-Net untuk Segmentasi 3 Dimensi Gelembung Penyebab Kanker Paru-paru (Nodule) pada Lapisan Citra CT Scan

Agung Pribadi, *Departemen Informatika, Institut Sains dan Teknologi Terpadu Surabaya,*  
Anang Kukuh Adisusilo, *Departemen Informatika, Universitas Wijaya Kusuma Surabaya.*

**Abstrak-** Kanker paru-paru adalah salah satu tipe kanker yang mematikan di seluruh dunia. Proses pendeteksian manual memakan banyak waktu dan energi untuk dituangkan. Kerja keras sekalipun tidaklah cukup, maka telat dalam pendeteksian telah menjadi faktor utama dalam tingkat kesembuhan. Menggunakan CT Scan adalah salah satu alat skrining yang potensial, tetapi program skrining otomatis dibutuhkan untuk menghemat waktu. Pendeteksian, menyegmentasi dan penggolongan nodule adalah tiga poin utama untuk membuat ini menjadi nyata. Oleh karena itu, dalam mengambil bagian untuk membuat program itu, tugas akhir ini akan melakukan fokus menyegmentasi nodule. Pada awal program, pertama dibutuhkan hasil CT Scan dan segmentasi nodulanya. Selain itu, juga dibutuhkan titik koordinat centroid dari nodule dari hasil scan tersebut. Setelah mengetahui titik centroid dari nodule, maka proses preprocessing akan menjadi mudah. Program akan mengambil daerah sekitar nodule dan nodule sebagai titik tengahnya, hasil dari proses ini adalah sebuah kubus yang memiliki ukuran yang sama untuk semua hasil preprocess yang sebelumnya sudah ditentukan. Setelah mendapatkan kubus daerah sekitar nodule, proses normalisasi akan dilakukan sehingga data siap untuk ditraining oleh neural network. 2 Model dibuat dengan arsitektur yang berbeda, seperti menggunakan U-Net biasa dan U-Net++. Hasil dari program akan menghasilkan skor Jaccard index dengan tingkat keakuratan tertinggi sebesar 79.8%. Didapatkan kesimpulan bahwa penggunaan U-Net biasa dan U-Net++ tidak memiliki perbedaan akurasi yang signifikan, meskipun memiliki komponen layer dan aktivasi yang sama.

**Kata Kunci-** CT Scan, Kanker Paru-paru, Nodule, Segmentasi, U-Net.

## I. PENDAHULUAN

Dalam era digital sekarang ini, teknologi sudah melingkupi banyak bidang pekerjaan, salah satunya adalah bidang kesehatan. Dalam bidang kesehatan, peralatan kedokteran terus diperbarui untuk menambah fungsi alat ataupun meningkatkan performa alat. Dalam kegiatan operasi, dokter spesialis selalu melihat bagian mana yang perlu dibedah, dan melihat apakah kegiatan pembedahan perlu dilakukan atau tidak melalui hasil scanner.

Agung Pribadi, Departemen Informatika, Institut Sains dan Teknologi Terpadu Surabaya, Surabaya, Jawa Timur, Indonesia (e-mail: omegaudayana777@gmail.com)

Anang Kukuh Adisusilo, Departemen Informatika, Universitas Wijaya Kusuma Surabaya, Jawa Timur, Indonesia (e-mail: anang@uwks.ac.id.)

Dengan begitu dokter dapat bekerja secara efektif dan tepat sasaran.

Kanker adalah penyakit yang perlu ditangani dengan cepat untuk meningkatkan tingkat penyembuhan. Namun untuk mendiagnosa kanker diperlukan alat screening seperti Computed Tomography (CT) Scan ataupun MRI sebelum dilakukan pengobatan. Di daerah terpencil dimana jumlah ahli atau radiologist berbanding terbalik dengan pasien, sehingga untuk melakukan manual checking hasil screening membutuhkan waktu yang lama. Selain itu, radiologist juga harus memeriksa gambar yang kompleks dengan tingkat abnormality yang beragam, sehingga kesalahan mungkin saja terjadi. Oleh karena itu, hal ini menyebabkan penggunaan komputer tidak dapat dihindari, dengan bantuan program [1] (CAD), dimana komputer dapat memproses dengan cepat dan tanpa lelah.

Dengan berkembangnya machine learning, program CAD pun ikut berkembang dengan pesat. Meskipun hasil dari CAD masih memerlukan pengecekan ulang oleh radiologist untuk memastikan ulang, program CAD sangatlah membantu memangkas waktu pemeriksaan. Kanker paru-paru adalah salah satu penyakit yang mematikan di seluruh dunia. Telatnya pendeteksian diagnosa menjadi salah satu faktor rendahnya tingkat kesembuhan penyakit ini. Untuk melakukan CAD dalam permasalahan kanker paru-paru diperlukan tiga langkah, yaitu pendeteksian gelembung paru-paru (nodules), lalu melakukan segmentasi pada nodule yang telah terdeteksi, dan proses selanjutnya adalah mengklasifikasi tekstur dari nodule tersebut. Ketiga langkah tersebut adalah fokus utama pada LNDB challenge yang dipublikasikan online.

Untuk mengatasi masalah tersebut, tugas akhir ini akan mengambil bagian tugas segmentasi saja. Dengan demikian hasil dari tugas akhir ini dapat dilanjutkan ke tahap selanjutnya. Tujuan pembuatan program ini dapat dilihat pada beberapa poin berikut:

- Menyegmentasi semua nodule yang diberikan ke program, nodule telah dipreproses sebelum menuju proses prediksi.
- Membandingkan kegunaan dari generator saat training dan berbagai tipe residual pada model training deep learning.

## II. TINJAUAN PUSTAKA

Untuk mendukung program yang dibuat yaitu segmentasi nodule akan menggunakan referensi beberapa jurnal, seperti



pada penelitian Fabian Isensee tahun 2019 tentang segmentasi tumor pada ginjal [2] dan pada penelitian Zongwei Zhou tahun 2018 tentang perbandingan kemampuan antara berbagai variasi U-Net untuk melakukan beberapa tugas segmentasi [3]. Pada jurnal pertama, peneliti menjelaskan bahwa kanker ginjal adalah masalah kesehatan serius dengan 400.000 kasus tiap tahun. Peneliti menyatakan bahwa semantic segmentation adalah langkah awal yang menjanjikan terhadap peningkatan hasil perawatan. Peneliti juga menyatakan bahwa sebagian mayoritas keberhasilan pada 3D image segmentation didasari oleh variasi dari arsitektur U-Net. Untuk itu peneliti mengembangkan dan mencoba melatih 3 tipe U-Net yaitu U-Net biasa tanpa residual, U-Net dengan residual dan pre-activation residual U-Net. Peneliti melatih model dengan stochastic gradient descent dan batch size dengan ukuran 2, juga mengaplikasikan data augmentation ke semua data training selama proses training berjalan. Data augmentasi yang diaplikasikan adalah scalling, rotation, brightness, gamma dan Gaussian noise. Berdasarkan pembahasan dan penelitian tersebut, maka dapat diambil kesimpulan bahwa data augmentasi dan berbagai macam tipe residual dapat dimanfaatkan untuk proses training.

Pada jurnal kedua, peneliti menyatakan bahwa skip connection yang menyambungkan encoder ke decoder telah terbukti ampuh menghasilkan segmentation mask dengan detail yang baik meskipun memiliki background yang kompleks, hal ini terlihat pada arsitektur U-Net biasa dan FCN yang memanfaatkan skip connection. Peneliti juga menyatakan bahwa menyegmentasi pada medical image menuntut akurasi tinggi karena jika terdapat error sedikit saja pada segmentasi dapat mengarah ke user experience yang rendah, sebagai contoh pola disekitar nodule paru-paru dapat mengindikasikan keganasan nodule. Oleh karena itu, diharapkan untuk adanya arsitektur image segmentation yang secara efektif memulihkan detail yang jelas pada objek target. Peneliti lalu merancang UNet++ yang memiliki arsitektur bersarang. Peneliti juga memiliki hipotesis bahwa model ini dapat secara efektif menangkap detail target secara jelas. Yang membedakan ini dari U-Net biasa yang secara langsung melempar hasil feature maps encoder ke decoder yang lalu menghasilkan perpaduan feature maps yang secara sementic tidak sama adalah pelemparan hasil feature maps pada encoder tidak langsung dilemparkan ke decoder tetapi melalui convolution block. Berdasarkan pembasahan dan penelitian tersebut, maka dapat diambil kesimpulan bahwa terdapat variasi lainnya selain U-Net yang dapat dicoba untuk training nantinya.

Untuk mencapai target, diperlukan beberapa komponen pendukung. Berikut adalah beberapa komponen pendukung untuk membuat program. Teori singkat diberikan agar pembaca dapat memahami secara singkat komponen tersebut.

### A. Convolutional Neural Network

Convolutional Neural Network (CNN) adalah perkembangan dari multi level perceptron. CNN terinspirasi dari proses biologi pada pola konektivitas antara banyak neuron sehingga menyerupai visual cortex pada otak. CNN dapat dipakai pada input berdimensi 2 atau lebih, contohnya adalah gambar. Gambar jika diubah menjadi multi level perceptron (MLP) maka akan mendapat akurasi yang jelek.

Oleh karena MLP mengabaikan spatial information dan memiliki banyak parameter yang redundant karena memakai fully connected layer sehingga tidak efisien.

$$P = (F_w * F_h * F_{n-1} + 1) * F_n \tag{1}$$

Pada formula 1, Parameter adalah weight yang dipelajari pada suatu layer. Penambahan angka 1 adalah karena pengaruh bias. Variabel  $F_w$  adalah ukuran lebar filter, variabel  $F_h$  adalah ukuran tinggi filter, variabel  $F_{n-1}$  adalah jumlah filter pada layer sebelumnya, variabel  $F_n$  adalah jumlah filter pada layer sekarang. Semakin besar total parameter sebuah model, maka semakin besar waktu yang diperlukan untuk melakukan training..

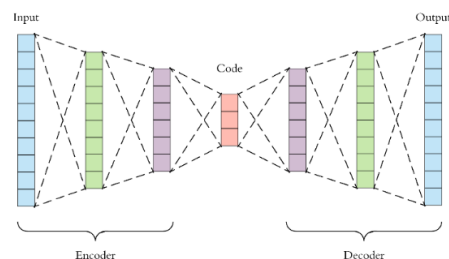
Karena gambar bergantung pada pixel, dan jika diubah menjadi 1 dimensi dan dilakukan proses dot matrix maka perhitungan akan kacau. Feed forward pada weight juga akan menjadi kacau karena posisi pixel, maka dari itu penggunaan multi perceptron jarang dipakai pada gambar. Layer convolutional memanfaatkan filter size untuk masalah ini.

Convolutional neural network terdiri dari layer convolutional dan layer-layer tambahan lainnya seperti pooling layer, activation function, layer upsampling, layer dropout[4], layer normalization[5] yang memiliki fungsi yang berbeda-beda. Layer-layer ini kemudian akan disusun dan membentuk model deep learning yang bisa dipakai untuk melatih tugas tertentu seperti klasifikasi.

### B. Autoencoder

Autoencoder[6] adalah salah satu tipe neural network yang digunakan untuk feature learning dan biasanya bersifat unsupervised, dan dapat bersifat supervised juga. Feature learning didasari oleh keinginan agar tugas machine learning seperti klasifikasi menjadi lebih mudah, fitur dalam supervised dipelajari menggunakan input data yang sudah dilabeli, fitur dalam unsupervised dipelajari tanpa menggunakan label. Tujuan dari Autoencoder adalah mempelajari perwakilan atau fitur yang penting dari suatu data, biasanya seperti dimensionality reduction. Autoencoder berusaha menghasilkan data baru dari representasi data yang penting tersebut.

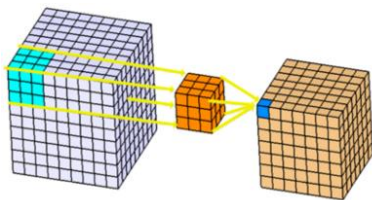
Autoencoder terdiri dari encoder dan decoder. Encoder berguna untuk mempelajari dan mengecilkan dimensi input. Decoder berguna untuk mempelajari dan memperbesar dimensi bottleneck hingga ke dimensi awal input. Hasil visual dari Autoencoder dapat lebih baik (seperti penggunaan denoising Autoencoder) atau lebih jelek (karena mengambil fitur yang penting saja). Autoencoder dapat bersifat stacked atau tidak, jika bersifat stacked maka neural network mempunyai banyak hidden layer, jika tidak maka neural network hanya memiliki 1 hidden layer.



Gambar 1. Struktur Autoencoder

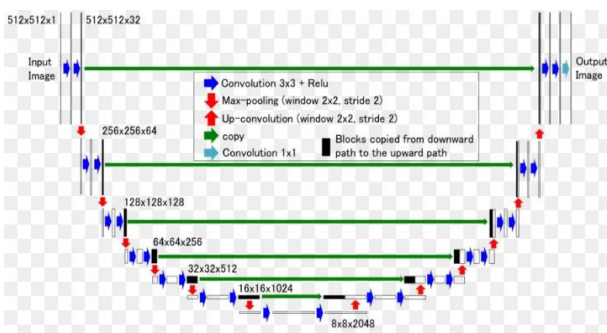
C. U-Net

Tujuan dari U-Net[7] adalah mapping fitur dari gambar (segmentasi), neural network ini dikhususkan untuk masalah klasifikasi untuk setiap pixel karena vector label untuk setiap pixelnya sudah disediakan oleh radiologist, dimana setiap file CT Scan bisa memiliki lebih dari 1 nodule, label dalam kasus ini adalah area nodule dengan nilai Id Finding yang dapat dilihat pada file csv. Target dari neural network adalah menghasilkan vector label. Unet memiliki encoder (down sampling) dan decoder (up sampling) yang sama seperti Autoencoder. Tiap layer encoder yang ada akan mengecilkan dimensi input layer tersebut, sehingga hal ini disebut down sampling. Tiap layer decoder yang ada akan membesarkan dimensi input layer tersebut, sehingga hal ini disebut up sampling. Hal yang membedakan kedua jenis adalah adanya layer penghubung encoder menuju decoder (layer concatenate). Unet biasanya menggunakan layer convolutional untuk bagian encoder dan decodernya ketimbang menggunakan layer Dense.



Gambar 2. Layer Convolutional 3 Dimensi

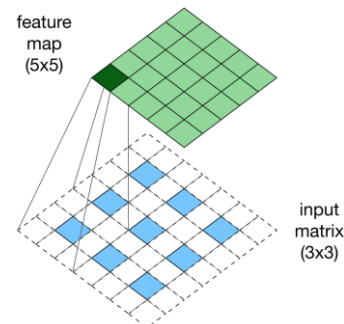
Unet dapat menerima input gambar 3 dimensi maupun 2 dimensi, dengan bantuan layer convolutional, hal ini menjadi mudah. Karena layer convolutional berupa 3 dimensi, maka weight juga berupa 3 dimensi. Proses perkalian input (bukan dot matrix) dengan weight akan menghasilkan output 3 dimensi juga. Dimensi output dapat diatur menggunakan stride dan padding, dimensi output dapat sama dengan dimensi input dengan bantuan padding dengan tipe same. Hal yang membedakan U-Net dari Autoencoder adalah adanya layer skip connection yang menghubungkan hasil feature map sebagai input tambahan untuk layer decoder pada level yang sama dengan menggunakan bantuan layer concatenate, sehingga dimensi input akan bertambah, bukannya nilai dari input decoder yang bertambah.



Gambar 3. Struktur U-Net

Pada gambar 3, garis hijau adalah layer concatenate yang menghubungkan encoder ke decoder. Garis merah ke bawah adalah layer downsampling yang biasanya menggunakan layer max pooling ataupun dapat memanfaatkan atribut stride

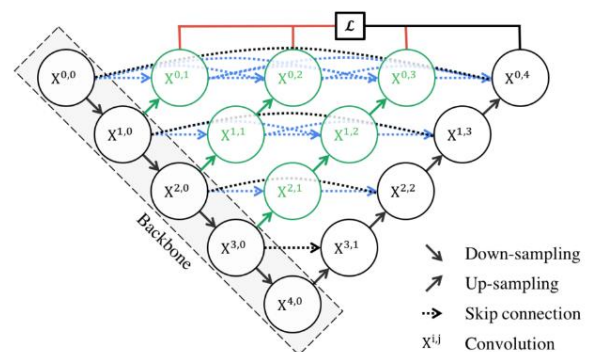
pada layer convolutional. Garis merah ke atas adalah layer transpose. Transpose layer adalah teknik lain selain up sampling biasa. Transpose layer seringkali salah dikenal sebagai layer deconvolutional, padahal kedua layer adalah jenis yang berbeda. Layer deconvolutional memutar balikan operasi dari layer convolutional biasa. Sedangkan layer transposed meningkatkan spatial dimensi dari input dengan bantuan stride dan padding. Selain itu, layer transposed memiliki weight yang dapat dipelajari, cara mempelajari weight sama dengan layer convolutional pada umumnya, sehingga layer ini mempunyai ukuran filter dan jumlah channel. Hal ini lah yang membedakan dari layer up sampling biasa. Hasil output dari layer deconvolutional dan layer transpose memiliki spatial dimension yang sama, sehingga hal ini yang membuat pemahaman kebanyakan orang salah.



Gambar 4. Visualisasi Layer Transpose

Pada gambar 4, input matrix awalnya adalah 3x3, tetapi karena terdapat padding 1 dan stride 2 maka akan keluar hasil seperti pada gambar. Kotak berwarna biru adalah kotak dengan nilai input awal matrix, sedangkan kotak tidak berwarna memiliki nilai 0. Langkah selanjutnya adalah dilakukan perkalian dengan weight seperti layer convolutional. Jika stride adalah 3 dan padding bertipe same, maka dimensi input awal akan dikalikan dengan 3.

D. U-Net++



Gambar 5. Struktur Unet++

Struktur Unet++ terdiri dari Unet sebagai bagian utama encoder dan decodernya (backbone dan hasil upsampling tiap decoder). Layer diantara encoder dan decoder utama adalah layer hasil concatenate layer upsampling layer convolutional level x+1 dengan layer convolutional yang ada pada level x

(level dimulai dari paling atas), layer ini ditandai dengan warna hijau pada gambar 5. Selain itu, layer hijau juga menerima concatenate dari layer-layer sebelumnya pada level yang sama. Sebagai contoh layer X0,3 pada gambar adalah hasil up sampling dari layer X1,2 dan menerima concatenate dari X0,0 dan X0,1 dan X0,2. Setelah itu biasanya hasil tersebut akan di feed forward ke layer convolutional. Hal ini untuk memberikan input ke layer-layer setelahnya sehingga dapat di-feed forward untuk mendapatkan informasi gabungan. Jumlah filter pada layer convolutional untuk setiap level memiliki jumlah yang sama dan spatial dimension output yang sama (selain hasil concatenate dan upsampling).

### III. SEGMENTASI NODULE

Pada bab ini akan dijabarkan, yaitu mengenai dataset, arsitektur program dan struktur model.

#### A. Dataset

Tersedia 236 file gambar dan 1527 baris csv yang dapat dijadikan menjadi dataset gambar, dengan atribut id file yang dipakai, id radiologis, id nodule yang ada pada 1 gambar, lokasi centroid x, lokasi centroid y, lokasi centroid z, apakah ini nodule, volume nodule dan tekstur nodule. Pada kasus segmentasi, atribut tekstur nodule dan apakah ini nodule tidak digunakan. File-file tersebut didapat dari LNDb challenge[8] yang dimulai pada tahun 2019. Setelah proses prediksi segmentasi berhasil dilakukan, dapat dilanjutkan proses klasifikasi tesktur nodule mulai dari nilai 0 sampai 5, dimana 0 adalah bukan nodule. Tetapi, proses klasifikasi tekstur tidak dilakukan karena bukan fokus utama pengerjaan.

| LNDbID | RadID | FindingID | x        | y        | z      | Nodule | Volume   | Text |
|--------|-------|-----------|----------|----------|--------|--------|----------|------|
| 1      | 1     | 1         | -44.6084 | -119.073 | -37.5  | 1      | 440.9088 | 5    |
| 1      | 1     | 2         | 25.85254 | -126.97  | -45.5  | 1      | 152.381  | 4    |
| 1      | 2     | 1         | -44.001  | -118.466 | -37.5  | 1      | 56.82005 | 5    |
| 1      | 3     | 1         | -44.001  | -119.681 | -37.5  | 1      | 169.3533 | 5    |
| 2      | 1     | 1         | 88.89551 | -123.626 | -129.5 | 1      | 339.188  | 5    |
| 2      | 1     | 2         | 63.53418 | -112.757 | -117.5 | 1      | 163.2933 | 5    |
| 2      | 1     | 3         | -103.851 | -117.104 | -253.5 | 1      | 357.0399 | 5    |
| 2      | 1     | 4         | 67.88184 | -95.3662 | -81.5  | 1      | 54.08105 | 3    |
| 2      | 1     | 5         | 44.69434 | -123.626 | -63.5  | 1      | 22.05247 | 4    |
| 2      | 2     | 1         | 88.89551 | -124.351 | -129.5 | 1      | 282.4816 | 5    |

Gambar 6. Gambaran Isi File Dataset

Pada gambar 6, terdapat nodule dengan titik centroid x,y dan z yang sama yang telah disegment oleh lebih dari 1 radiologist. Setiap radiologist memberikan hasil segmentasi menurut pendapat mereka masing-masing, sehingga hasil tersebut tidak sama satu sama lain. Hal ini dapat dimanfaatkan untuk melihat distribusi nodule pada dataset utama sebelum di split untuk training, validasi dan testing.

| LNDbID | RadID | RadFinding | FindingID | x        | y        | z      | AgrLevel | Nodule | Volume   | Text |
|--------|-------|------------|-----------|----------|----------|--------|----------|--------|----------|------|
| 1      | 1,2,3 | 1,1,1      | 1         | -44.2035 | -119.073 | -37.5  | 3        | 1      | 222.3607 | 5    |
| 1      | 1     | 1          | 2         | 25.85254 | -126.97  | -45.5  | 1        | 1      | 152.381  | 4    |
| 2      | 1,2,3 | 1,1,3      | 1         | 88.89551 | -123.668 | -129.5 | 3        | 1      | 378.0423 | 5    |
| 2      | 1,3   | 2,2        | 2         | 63.53418 | -112.757 | -117.5 | 2        | 1      | 174.8446 | 5    |
| 2      | 1,3   | 3,5        | 3         | -103.851 | -116.742 | -253   | 2        | 1      | 297.7083 | 5    |
| 2      | 1     | 4          | 4         | 67.88184 | -95.3662 | -81.5  | 1        | 1      | 54.08105 | 3    |
| 2      | 1     | 5          | 5         | 44.69434 | -123.626 | -63.5  | 1        | 1      | 22.05247 | 4    |
| 2      | 2     | 2          | 6         | 117.8799 | -121.452 | -209.5 | 1        | 1      | 14.13717 | 5    |
| 2      | 3     | 1          | 7         | 87.44629 | -121.452 | -107.5 | 1        | 1      | 384.8681 | 4    |

Gambar 7. File Kumpulan Nodule yang Sama

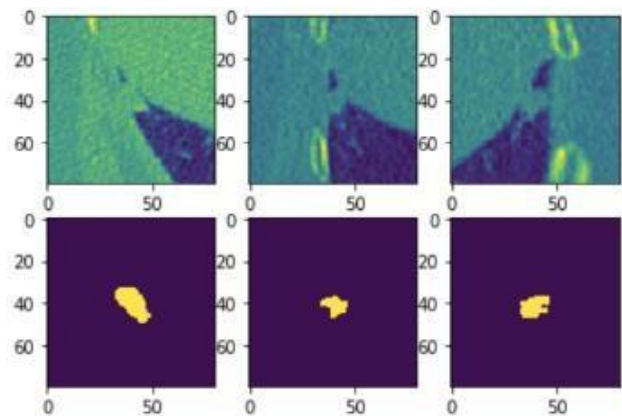
Pada gambar 7, informasi tentang nodule yang sama telah diringkas sehingga dapat dengan mudah dibuat distribusinya. Nilai seperti x, y, z, volume dan texture adalah nilai rata-rata dari kumpulan data awal. Nilai rata-rata pada informasi di file ini tidak dipakai, program akan memakai nilai x, y, z

awal pada gambar 6. Dari file csv gambar 7, akan dapat sebuah kesimpulan yaitu, jumlah nodule dengan frekuensi 0 adalah 594, dengan frekuensi 1 adalah 432, dengan frekuensi 2 adalah 151, dengan frekuensi 3 adalah 42.

1528 data akan dipangkas menjadi 860 data karena perhitungan volume nodule yang akan diambil adalah dengan lebar diameter lebih besar sama dengan 3 mm. Angka 3 mm dipilih karena angka dibawah itu adalah micronodule, micronodule dapat berevolusi menjadi nodule dan jika micronodule terus membesar diameternya maka dapat bersifat (ada kemungkinan) berbahaya dan dapat bersifat kanker. Perhitungan diameter nodule menggunakan rumus volume sphere.

$$V = \frac{4}{3}\pi r^3 \tag{2}$$

Pada dataset, volume sphere sudah diketahui maka variabel yang harus dicari adalah r (radius atau jari-jari), setelah menemukan jari-jari, maka langkah selanjutnya adalah hasil jari-jari dikalikan dengan 2 agar menghasilkan diameter sphere. Pembagian data adalah 70% untuk training, 20% untuk validasi, 10% untuk testing.



Gambar 8. Contoh Nodule dan Segmentasinya

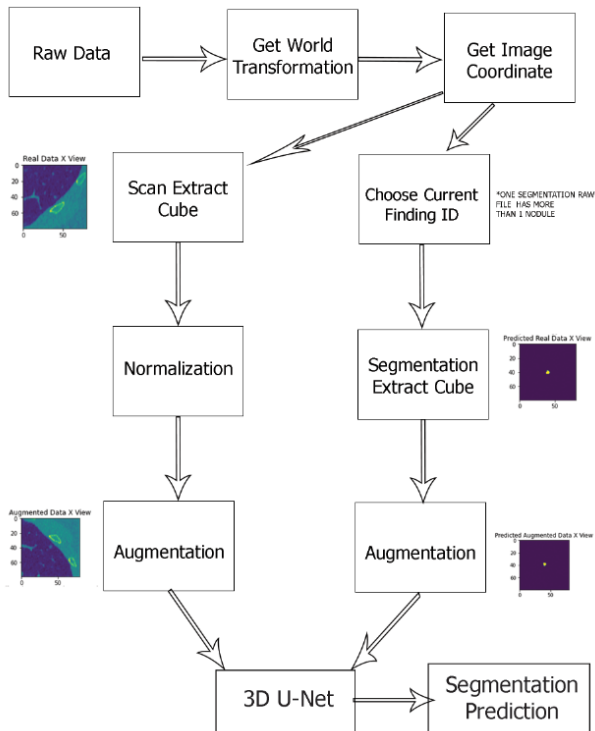
Pada gambar 8, gambar sudah dilakukan preprocessing dan hasil outputnya berupa kubus dengan dimensi 80x80x80 pixel. Dapat terlihat sebuah objek berbentuk menyerupai gelembung. Posisi gelembung untuk semua dataset pasti berada pada posisi tengah karena pengaruh titik centroid x, y dan z yang ada di dataset.

#### B. Arsitektur Program

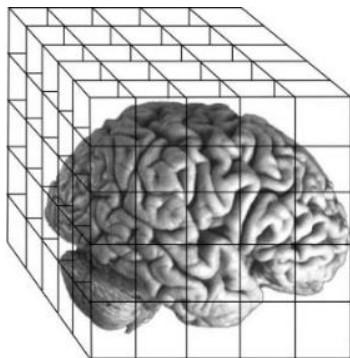
Program akan mengambil raw data dari dataset menggunakan library simpleITK sesuai dengan informasi dari file csv. Hasil rawdata akan diubah menjadi tipe numpy sehingga mudah diolah oleh library lainnya. Setelah itu, titik-titik centroid akan digunakan untuk membuat area kubus untuk nodule yang ditunjuk dan juga dilakukan hal yang sama untuk target segmentasi nodule. Setelah itu kubus nodule dan segmentasinya akan menjadi input untuk model training.

Input dari program ini adalah raw data. Library simpleITK membantu pengolahan raw data sehingga dapat dibaca, seperti mendapatkan nilai spacing, origin dan rotation matrix. Variabel-variabel tersebut sangat penting untuk mengubah sistem koordinasi CT scan menjadi sistem koordinasi gambar biasa. Raw data memiliki Anatomical

Coordinate System, simpleITK sudah mengubah raw data menjadi Image Coordinate System, tetapi titik centroid pada file csv belum dirubah menjadi Image Coordinate System.



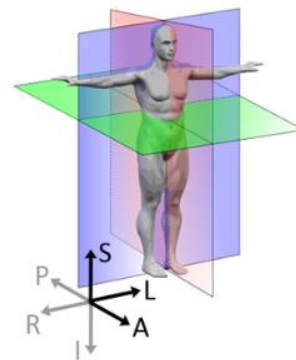
Gambar 9. Arsitektur Program



Gambar 10. Contoh Voxel Pada Gambar 3 Dimensi

Voxel adalah volumetric pixel penyusun gambar 3 dimensi. Voxel adalah panggilan bongkahan pixel dalam gambar 3 dimensi, dalam 1 voxel terkandung informasi yang tidak dimiliki oleh pixel, karena voxel memiliki banyak pixel, sedangkan pixel hanya memiliki 1 pixel itu sendiri. Voxel tidak harus berbentuk kubus, namun bisa berbentuk segitiga, diamond, bola, maupun bidang lainnya yang dapat ditumpuk menghasilkan voxel.

Untuk pengerjaan tugas ini, sistem koordinat dibagi menjadi 3 yaitu world coordinate system, anatomical coordinate system dan image coordinate system[9]. World coordinate system adalah sistem koordinat kartesian dimana kita ingin menentukan dimana gambar ditampilkan, posisi originnya biasanya berada di pojok kiri bawah, tetapi nilai originnya dapat dirubah.

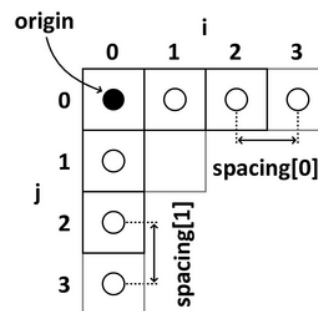


Gambar 11. Contoh Anatomical Coordinate System

Anatomical Coordinate System adalah sistem koordinat yang dipakai untuk teknik pencitraan medis. Sistem ini terdiri dari 3 bidang, yaitu:

- Bidang axial yang berdiri tegak dari tanah menuju kepala. Pada gambar 11, bidang ini diwakilkan oleh variabel S (Superior) dan I (Inferior).
- Bidang coronal yang memisahkan antara depan dan belakang. Pada gambar 11, bidang ini diwakilkan oleh variabel A (Anterior) dan P (Posterior).
- Bidang sagittal yang membentang dari kiri hingga kanan. Pada gambar 11, bidang ini diwakilkan oleh variabel L (Left) dan R (Right).

Image coordinate system menjelaskan bagaimana nilai pixel dalam gambar diletakkan di dalam gambar. Format foto JPEG atau format lainnya mempunyai standard untuk pengkoordinasian. Selain itu, sistem ini juga menjelaskan orientasi (landscape atau portrait). Perbedaan antara image coordinate dengan world coordinate adalah origin, image coordinate tidak dipengaruhi oleh origin, sedangkan world coordinate dipengaruhi



Gambar 12. Gambar Penjelasan Spacing dan Origin

Spacing adalah jarak antara titik centroid voxel dengan titik centroid voxel lainnya. Origin adalah posisi awal voxel pertama, biasanya berada di pojok kiri atas. Sumbu i akan bergerak ke kanan dan sumbu j akan bergerak ke bawah. Setiap gambar raw memiliki 1 buah nilai origin, spacing dan rotation matrix

$$R = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \quad (3)$$

Pada formula 3, Rotation matrix biasanya berupa matrix identitas. Matrix ini mengatur apakah titik koordinat voxel memiliki rotasi dari raw data. Nilai  $\theta$  biasanya adalah 0

karena tidak terjadi rotasi, jika nilai 0 maka secara matematis matrix akan menjadi matrix identitas. Input masih memiliki sistem perkoordinasian anatomical, sehingga langkah selanjutnya adalah mengubahnya menjadi image coordinate system dan mengekstrak daerah sekitar nodule sehingga input deep learning tidak perlu menerima banyak informasi tidak penting, seperti mempelajari semua area CT Scan.

Preprocessing diperlukan agar input data dapat diringkas sehingga dapat digunakan untuk deep learning. Sistem koordinat pada titik centroid x, y, z yang ada pada dataset perlu dirubah menjadi Image Coordinate System. Berikut adalah penjelasan dan langkah-langkah preprocessing:

- Mendapatkan World Transformation

Setelah mendapatkan nilai variabel origin, spacing dan rotation matrix maka langkah selanjutnya adalah mengubah sistem koordinasi dataset menuju ke world coordinate system untuk mendapatkan transformasi matrixnya, karena jika transformasi matrix milik world coordinate di inverse maka akan menjadi transformasi matrix image coordinate.

$$\begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \\ w_{31} & w_{32} & w_{33} \end{bmatrix} = \begin{bmatrix} A_{11} * S_1 & A_{12} * S_2 & A_{13} * S_3 \\ A_{21} * S_1 & A_{22} * S_2 & A_{23} * S_3 \\ A_{31} * S_1 & A_{32} * S_2 & A_{33} * S_3 \end{bmatrix} \quad (4)$$

Pada formula 3, variabel A adalah rotation matrix dan variabel S adalah nilai spacing untuk tiap sumbu atau axis. Rotation matrix dikalikan dengan spacing karena pada world coordination system dan image coordination system tidak menyimpan nilai spacing. Langkah selanjutnya adalah menginverse transformasi matrix yang didapatkan.

- Mengubah Titik Centroid Menjadi Image Coordinate System

Titik-titik centroid yang terdapat pada file csv perlu diubah sistem koordinatnya. Hal ini karena gambar raw sudah berubah menjadi Image Coordinate System (bertipe numpy) setelah dibaca oleh library simpleITK. Titik centroid ini akan memberitahu lokasi titik tengah dari nodule, setelah diubah sistem koordinatnya, maka crop daerah sekitar nodule akan menjadi mudah.

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} - \begin{bmatrix} T_1 \\ T_2 \\ T_3 \end{bmatrix} \quad (5)$$

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} W_{11} & W_{12} & W_{13} \\ W_{21} & W_{22} & W_{23} \\ W_{31} & W_{32} & W_{33} \end{bmatrix}^{-1} \cdot \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (6)$$

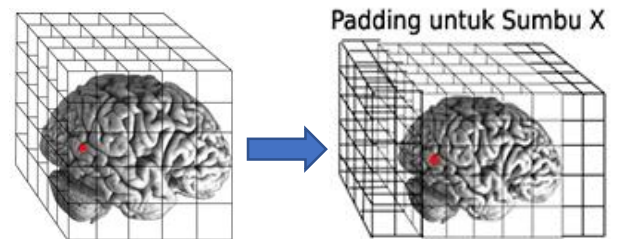
Pada formula 5 dan 6 digunakan untuk mengubah posisi menjadi koordinat Image Coordinate System, variabel T adalah nilai origin untuk masing-masing sumbu atau axis. Image Coordinate System tidak memiliki nilai origin sehingga, titik-titik koordinat dikurangi dengan nilai origin. Setelah dikurangi, hasil tersebut dikalikan matriks dengan transformasi matriks World Coordinate System yang sudah dilakukan inverse. Setelah mengubah sistem koordinat, langkah selanjutnya adalah mengambil daerah sekitar nodule.

- Ekstrak Kubus

Langkah pertama untuk mengekstrak daerah sekitar nodule adalah menentukan ukuran untuk 1 voxel, karena voxel pada dataset adalah kubus maka 1 ukuran untuk semua sisi, dan menentukan dimensi output yang diinginkan. Semakin besar ukuran voxel maka gambar nodule yang ditampilkan akan semakin kecil, karena 1 voxel yang lebih besar menampung lebih banyak informasi gambar, dalam kasus ini nodule ada pada titik tengah voxel.

$$HC_i = \frac{1}{2} * S_i * L \quad (7)$$

Pada formula 7, rumus tersebut untuk mendapatkan setengah bagian dari voxel atau cube untuk masing-masing axis atau sumbu. Rumus tersebut akan menghasilkan 3 hasil untuk masing-masing axis. Hal ini dilakukan agar nodule bisa berada di tengah voxel. Variable Si adalah nilai spacing untuk axis yang ke i. Variabel L adalah nilai ukuran untuk 1 voxel. Setelah mendapatkan ukuran setengah voxel, langkah selanjutnya adalah mengecek apakah posisi centroid voxel jika ditambah atau dikurangi nilai setengah apakah melewati batas resolusi gambar.



Gambar 13.. Ilustrasi Padding untuk Sumbu X

Jika melewati maka akan dilakukan padding untuk semua sumbu, padding dengan nilai 0 dilakukan sebanyak nilai maximum dari nilai Hc untuk sisi kiri, bawah dan depan, nilai maximum juga untuk sisi kanan, atas dan belakang. Karena pengaruh dari padding padding maka titik centroid yang sudah diubah menjadi Image Coordinate System ditambah dengan nilai maximum tersebut (lihat posisi titik merah pada gambar, gambar sudah menjadi Image Coordinate System sejak dibaca dengan library simpleITK).

Jika tidak melewati dimensi gambar, maka dilakukan pemotongan gambar dimulai dari titik centroid x-Hcx sampai x + Hcx, dari titik centroid y-Hcy sampai y+Hcy, dari titik centroid z sampai z + Hcz. Hasil pemotongan bisa saja tidak sesuai dengan dimensi output yang diinginkan(80x80x80), maka dilakukan proses zoom sengan skala dibagi dengan dimensi hasil x untuk proses zoom sumbu x, sumbu lain pun dilakukan proses yang sama.

- Memilih Finding Id pada Data Segmentasi

Masing-masing CT Scan memiliki informasi segmentasinya sendiri-sendiri yang berada pada file yang berbeda dengan file CT Scan. Setelah data segmentasi telah melalui proses ekstrak cube dengan dimensi 80x80x80. Dalam informasi kubus tersebut dapat memiliki lebih dari 1 nodule, hal ini ditandai dengan angka. Seperti angka 0 menandakan bukan area nodule, angka 1 dan seterusnya adalah id nodule pada kubus tersebut. Id finding dapat dilihat

pada file csv dataset, selain id finding pada iterasi pembacaan file akan dijadikan nilai 0 dan nilai id finding pada kubus akan dijadikan angka 1.

- Normalisasi

Tipe dari gambar biasanya adalah RGB (Red, Green, Blue), yang terdiri dari nilai 0 sampai 255. Namun, tipe pada dataset ini adalah tipe Hounsfield Unit (HU) yang terdiri dari nilai antara -1000 sampai 4000. Nilai-nilai tersebut menentukan bagian dari organ tubuh seperti tulang memiliki nilai sekitar 100 HU sampai 2200 HU dan lemak ditandai dengan nilai sekitar -120 sampai -90.

Normalisasi dibutuhkan agar saat training jika misal value pada dataset ke 1 index x adalah 100 dan dataset ke 2 index x adalah 3, maka perubahan weight saat backpropagation akan sangat besar. Oleh karena itu, dibutuhkan normalisasi data input, meskipun value pada index x berbeda, namun tidak akan mengakibatkan perubahan weight yang sangat besar. Normalisasi yang dipakai pada dataset adalah normalisasi min max, dengan memanfaatkan range nilai min dan max pada 1 data dapat digunakan untuk melakukan normalisasi. Berikut adalah rumus normalisasi min max.

$$t_{range} = t_{max} - t_{min} \tag{8}$$

$$s_{range} = s_{max} - s_{min}$$

$$scale = \frac{(arrayimage - s_{min})}{s_{range}}$$

$$result = t_{min} + (scale * t_{range})$$

Pada formula 8, target normalisasi adalah 0 untuk nilai minimumnya dan 1 untuk nilai maximumnya. Variabel t adalah target, s adalah sumber, sehingga  $s_{max}$  dan  $s_{min}$  ditentukan sendiri nilainya. Rumus diatas dapat bekerja jika keempat parameter sudah diketahui yaitu: nilai maximum target, nilai minimum target, nilai maximum sumber image, nilai minimum sumber image. Rumus ini juga dapat dipakai untuk denormalisasi, yaitu mengembalikan nilai ke asalnya jika mengetahui keempat parameternya. Variabel  $arrayimage$  adalah sebuah array yang mengandung informasi gambar. Variabel  $scale$  akan menghasilkan output sama dengan dimensi gambar.

Pada saat pengerjaan, normalisasi dilakukan untuk masing-masing gambar, sehingga nilai maximum dan minimum sumber akan berbeda-beda seperti karena data CT scan tidak semuanya terdapat udara yang nilainya HU nya -1000 atau nilai HU lainnya. Normalisasi ini tidak mengikuti nilai maximum dan minimum Hounsfield Unit yaitu -1000 untuk minimum dan 4000 untuk maximum. Hal ini dilakukan untuk mengacak nilai (hasil normalisasi) nodule pada gambar untuk deep learning.

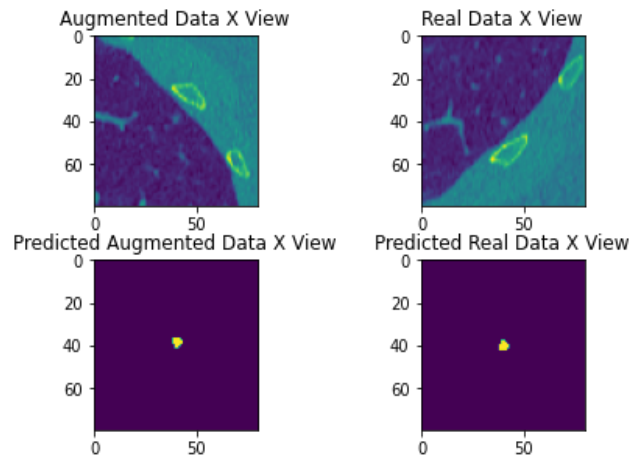
- Data Augmentation

Data augmentation digunakan untuk menambah variasi dari nodule. Karena nodule selalu berada di tengah maka augmentasi yang akan dilakukan adalah rotasi, flip dan brightness. Akan ada 400 data augmentasi tambahan untuk

dataset utama. Beberapa augmentasi yang digunakan adalah sebagai berikut.

1. Rotasi Gambar

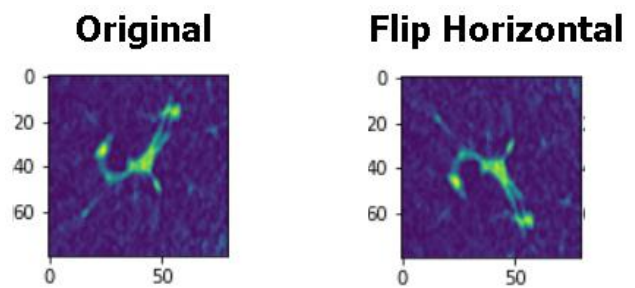
Proses ini akan memutar gambar secara 3 dimensi dengan titik centroid kubus sebagai pusat perputaran. Sudut perputaran dipilih secara acak dengan range antara 30 derajat hingga -30 derajat. Rotasi gambar dapat dilakukan dengan mudah dengan bantuan library Scipy. Contoh gambar yang telah diaplikasikan dengan rotasi dapat dilihat pada gambar 14.



Gambar 14 Contoh Augmentasi Rotate

2. Membalikkan Gambar

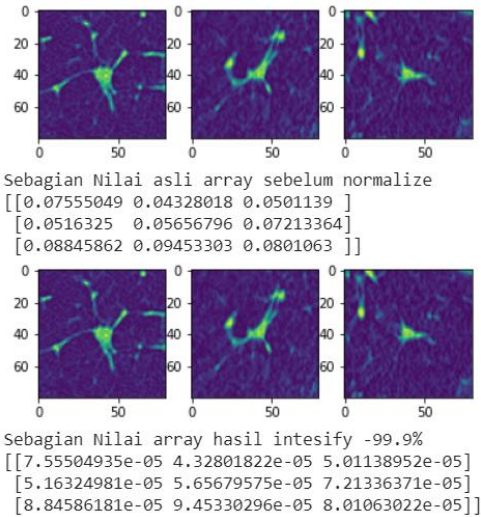
Proses ini akan membalikkan gambar secara horizontal ataupun vertical. Pemilihan sumbu tersebut akan dipilih secara acak. Proses ini dapat dengan mudah dilakukan juga dengan menggunakan library Scipy. Contoh gambar yang telah diaplikasikan dengan membalikkan gambar dapat dilihat pada gambar 15.



Gambar 15. Contoh Augmentasi Flip Horizontal

3. Intensitas Pixel

Proses ini akan mengurangi nilai setiap pixel sebanyak nilai persen yang ditentukan. Proses ini dapat diselesaikan tanpa menggunakan library apapun, hanya menggunakan perkalian biasa. Meskipun nilai pixel telah berubah drastis, terang gelapnya tidak dapat dilihat pada gambar. Perubahan intensitas gambar pada program berada di range antara 10% hingga -10%. Contoh gambar yang telah diaplikasikan dapat dilihat pada gambar 16.

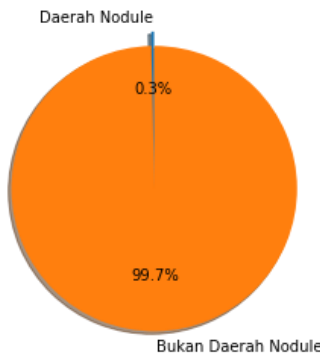


Gambar 16. Contoh Augmentasi Intensity.

Dengan adanya data augmentasi maka distribusi data akan berubah menjadi: frekuensi 0 adalah 594, dengan frekuensi 1 adalah 234, dengan frekuensi 2 adalah 242, frekuensi 3 adalah 79, dengan frekuensi 4 adalah 48, dengan frekuensi 5 adalah 19, dengan frekuensi 6 adalah 3.

• Training

Proses training menerima input dataset yang sudah melewati preprocessing dan data hasil augmentasi, juga menerima target segmentasi dataset asli dan segmentasi hasil augmentasi. Setiap dataset pasti memiliki nodule. Training dilakukan dengan menggunakan dice loss, penggunaan loss ini karena kelas untuk nodule dan bukan nodule sangat tidak seimbang.



Gambar 17. Distribusi Nilai Target

Pada gambar 17, data distribusi berasal dari input training yang berjumlah 70% dari 1260 gambar (860 gambar original, 400 gambar augmentasi). Terlihat bahwa distribusi dikuasai oleh daerah bukan nodule atau nilai 0 pada target sebanyak 99.7%, dan distribusi daerah nodule atau nilai 1 pada target sangat kecil sebanyak 0.3%. Jika menggunakan loss function binary crossentropy, hasil prediksi akan jelek karena pengaruh nilai 0 pada target.

$$Loss\ Function = 1 - \frac{2 \sum_{i=0}^h \sum_{j=0}^w (p_{ij} * \hat{p}_{ij})}{\sum_{i=0}^h \sum_{j=0}^w (p_{ij} + \hat{p}_{ij})} \quad (9)$$

Pada formula 9, variabel  $\hat{p}_{ij}$  adalah nilai prediksi model dan variabel  $p_{ij}$  adalah nilai target. terjadi proses perkalian pada pecahan pembilang. Hal ini sangat penting karena dengan mengkalikan, maka jika salah satu memiliki nilai 0 maka nilai Dice Loss tidak akan terpengaruh. Pada bagian penyebut pecahan pun nilai 0 tidak memiliki efek apapun yang mempengaruhi nilai dice loss. Dice coefficient pada training akan digunakan sebagai metric.

• Testing

Pengukuran akurasi testing akan menggunakan Jaccard Index. Daerah interseksi akan dimanfaatkan untuk mengukur kesamaan antara data testing dan memiliki rumus daerah interseksi dibagi dengan union dari data. Formula ini cocok untuk membandingkan kesamaan antara data prediksi dengan data target.

$$J(A, B) = \frac{|A \cap B|}{|A| + |B| - |A \cap B|} \quad (10)$$

Pada formula 10, variabel A adalah data target variabel B adalah data prediksi. Nilai dari interseksi akan dibagi dengan total penjumlahan area nodule A dengan area nodule B lalu dikurangi dengan daerah interseksi. Formula seperti ini juga dikenal sebagai Intersection Over Union (IoU).

C. Struktur Model

Struktur model akan ditulis dengan bantuan block. Block terhubung dengan struktur utama. Output dimensi block ditampilkan pada struktur model utama. Sebuah layer terhubung dengan layer yang ada pada kolom 'Connected To'. Dimensi input dan output memiliki nilai yang sama karena menggunakan tipe padding same, kecuali layer shortcut projection, max pooling dan transpose.

- Model U-Net (Dua Conv32-Instnorm-ReLU Max Pooling Dice Loss)

Berikut adalah struktur model pertama:

TABEL 1  
STRUKTUR BLOCK MODEL U-NET

| Block  |                            |
|--|----------------------------|
| Layer  | Connected To               |
| conv3dBlock (Conv3d, ?, 3x3x3)                     | inputBlock                 |
| instancenormalizationBlock (InstanceNormalization) | conv3dBlock                |
| reluBlock  | instancenormalizationBlock |

Pada tabel 1, simbol '?' merupakan parameter untuk banyaknya layer convolutional. Nilai ini didapat dari parameter struktur layer di model utama. Setiap layer convolutional pada block memiliki convolutional size 3x3x3 dan stride 1x1x1. Model ini tidak memiliki residual, yang

membedakan stuktur block dengan model E adalah jumlah stack atau tumpukan block.

TABEL 2  
STRUKTUR MODEL U-NET

| Layer                             | Output         | Connected To            |
|-----------------------------------|----------------|-------------------------|
| input1(InputLayer)                | [(80,80,80,1)] | -                       |
| Block1 (32)                       | (80,80,80,32)  | input1                  |
| Block2 (32)                       | (80,80,80,32)  | Block1                  |
| maxpooling3d(MaxPooling3D)        | (40,40,40,32)  | Block2                  |
| Block3 (64)                       | (40,40,40,64)  | maxpooling3d            |
| Block4 (64)                       | (40,40,40,64)  | Block3                  |
| maxpooling3d1(MaxPooling3D)       | (20,20,20,64)  | Block4                  |
| Block5 (128)                      | (20,20,20,128) | maxpooling3d1           |
| Block6 (128)                      | (20,20,20,128) | Block5                  |
| maxpooling3d2(MaxPooling3D)       | (10,10,10,128) | Block6                  |
| Block7 (256)                      | (10,10,10,256) | Maxpooling3d2           |
| Block8 (256)                      | (10,10,10,256) | Block7                  |
| maxpooling3d3(MaxPooling3D)       | (5,5,5,256)    | Block8                  |
| Block9 (512)                      | (5,5,5,512)    | maxpooling3d3           |
| Block10 (512)                     | (5,5,5,512)    | Block9                  |
| conv3dtranspose(Conv3DTranspose)  | (10,10,10,256) | Block10                 |
| concatenate(Concatenate)          | (10,10,10,512) | conv3dtranspose Block8  |
| Block11 (256)                     | (10,10,10,256) | concatenate             |
| Block12 (256)                     | (10,10,10,256) | Block11                 |
| conv3dtranspose1(Conv3DTranspose) | (20,20,20,128) | Block12                 |
| concatenate1(Concatenate)         | (20,20,20,256) | conv3dtranspose1 Block6 |
| Block13 (128)                     | (20,20,20,128) | concatenate1            |
| Block14 (128)                     | (20,20,20,128) | Block13                 |
| conv3dtranspose2(Conv3DTranspose) | (40,40,40,64)  | Block14                 |
| concatenate2(Concatenate)         | (40,40,40,128) | conv3dtranspose2 Block4 |
| Block15 (64)                      | (40,40,40,64)  | concatenate2            |
| Block16 (64)                      | (40,40,40,64)  | Block15                 |

TABEL 2 (LANJUTAN)  
STRUKTUR MODEL U-NET

| Layer                             | Output        | Connected To            |
|-----------------------------------|---------------|-------------------------|
| conv3dtranspose3(Conv3DTranspose) | (80,80,80,32) | Block16                 |
| concatenate3(Concatenate)         | (80,80,80,64) | conv3dtranspose3 Block2 |
| Block17 (32)                      | (80,80,80,32) | concatenate3            |
| Block18 (32)                      | (80,80,80,32) | Block17                 |
| conv3d(Conv3D, 1, 1x1x1)          | (80,80,80,1)  | Block18                 |

Penumpukan block pada model ini adalah 2. Model ini memiliki 22,581,217 parameter sehingga membutuhkan waktu yang lama untuk training dan memory yang besar. Penumpukan block dilakukan karena dengan penumpukan blok jenis ini memiliki akurasi yang tinggi, sehingga ingin dilihat apakah dengan penumpukan 2 block akan menghasilkan akurasi yang lebih tinggi atau tidak.

- Model U-Net++ (Uplus Dua Conv32-Instnorm-ReLU Max Pooling Dice Loss)

Berikut adalah struktur model kedua:

TABEL 3  
STRUKTUR RESIDUAL BLOCK MODEL U-NET++

| ResidualBlock                                       |                             |
|---|-----------------------------|
| Layer   | Connected To                |
| conv3dBlock(Conv3d, ?, 3x3x3)                       | inputBlock                  |
| instancenormalizationBlock (InstanceNormalization)  | conv3dBlock                 |
| reluBlock (ReLU)                                    | instancenormalizationBlock  |
| conv3dBlock1(Conv3d, ?, 3x3x3)                      | reluBlock                   |
| instancenormalizationBlock1 (InstanceNormalization) | conv3dBlock1                |
| reluBlock1 (ReLU)                                   | instancenormalizationBlock1 |

Pada tabel 3. Setiap layer convolutional pada block memiliki convolutional size 3x3x3 dan stride 1x1x1. Model ini tidak memiliki residual. Hal yang membedakan stuktur block dengan model G adalah layer activation dan layer normalisasi.

TABEL 4  
STRUKTUR MODEL U-NET++

| Layer              | Output         | Connected To |
|--------------------|----------------|--------------|
| input1(InputLayer) | [(80,80,80,1)] | -            |
| Block_1 (32)       | (80,80,80,32)  | input1       |



TABEL 4 (LANJUTAN)  
Struktur model U-Net++

| Layer                             | Output         | Connected To                              |
|-----------------------------------|----------------|---|
| maxpooling3d(MaxPooling3D)        | (40,40,40,32)  | Block_1                                   |
| Block_2 (64)                      | (40,40,40,64)  | maxpooling3d                              |
| maxpooling3d1(MaxPooling3D)       | (20,20,20,64)  | Block_2                                   |
| Block_3 (128)                     | (20,20,20,128) | maxpooling3d1                             |
| maxpooling3d2(MaxPooling3D)       | (10,10,10,128) | Block_3                                   |
| Block_4 (256)                     | (10,10,10,256) | maxpooling3d2                             |
| maxpooling3d3(MaxPooling3D)       | (5,5,5,256)    | Block_4                                   |
| Block_5 (512)                     | (5,5,5,512)    | maxpooling3d3                             |
| conv3dtranspose(Conv3DTranspose)  | (10,10,10,256) | Block_5                                   |
| concatenate(Concatenate)          | (10,10,10,512) | conv3dtranspose Block_4                   |
| Block_6 (256)                     | (10,10,10,256) | Concatenate                               |
| conv3dtranspose1(Conv3DTranspose) | (20,20,20,128) | Block_4                                   |
| Concatenate1(Concatenate)         | (20,20,20,256) | conv3dtranspose1 Block_3                  |
| Block_7 (128)                     | (20,20,20,128) | Concatenate1                              |
| conv3dtranspose2(Conv3DTranspose) | (20,20,20,128) | Block_6                                   |
| Concatenate2(Concatenate)         | (20,20,20,384) | conv3dtranspose2 Block_3 Block_7          |
| Block_8 (128)                     | (20,20,20,128) | Concatenate2                              |
| conv3dtranspose3(Conv3DTranspose) | (40,40,40,64)  | Block_3                                   |
| Concatenate3(Concatenate)         | (40,40,40,128) | conv3dtranspose3 Block_2                  |
| Block_9 (64)                      | (40,40,40,64)  | Concatenate3                              |
| conv3dtranspose4(Conv3DTranspose) | (40,40,40,64)  | Block_7                                   |
| Concatenate4(Concatenate)         | (40,40,40,192) | conv3dtranspose4 Block_2                  |
| Block_10 (64)                     | (40,40,40,64)  | Concatenate4                              |
| conv3dtranspose5(Conv3DTranspose) | (40,40,40,64)  | Block_8                                   |
| Concatenate5(Concatenate)         | (40,40,40,256) | conv3dtranspose5 Block_2 Block_9 Block_10 |
| Block_11 (64)                     | (40,40,40,64)  | Concatenate5                              |

TABEL 4 (LANJUTAN)  
STRUKTUR MODEL U-NET++

| Layer                             | Output         | Connected To                               |
|-----------------------------------|----------------|--|
| conv3dtranspose6(Conv3DTranspose) | (80,80,80,32)  | Block_2                                    |
| Concatenate6(Concatenate)         | (80,80,80,64)  | conv3dtranspose6 Block_1                   |
| Block_12 (32)                     | (80,80,80,32)  | Concatenate6                               |
| conv3dtranspose7(Conv3DTranspose) | (80,80,80,32)  | Block_9                                    |
| Concatenate7(Concatenate)         | (80,80,80,96)  | conv3dtranspose7 Block_1                   |
| Block_13 (32)                     | (80,80,80,32)  | Block_12 Concatenate7                      |
| conv3dtranspose8(Conv3DTranspose) | (80,80,80,32)  | Block_10                                   |
| Concatenate8(Concatenate)         | (80,80,80,128) | conv3dtranspose8 Block_1 Block_12          |
| Block_14 (32)                     | (80,80,80,32)  | Block_13 Concatenate8                      |
| conv3dtranspose9(Conv3DTranspose) | (80,80,80,32)  | Block_11                                   |
| Concatenate9(Concatenate)         | (80,80,80,128) | conv3dtranspose9 Block_1 Block_12 Block_13 |
| Block_15 (32)                     | (80,80,80,32)  | Block_14 Concatenate9                      |
| conv3(Conv3d, 1, 1x1x1)           | (80,80,80,1)   | Block_15                                   |

Penumpukan layer convolutional beserta activationnya pada layer ini adalah 2 (terlihat pada tabel 3). Layer ini memiliki 30,563,713 parameter sehingga membutuhkan waktu dan memory yang besar. Pada model ini ingin diketahui apa tidak perbedaan arsitektur biasa dengan arsitektur U-Net++. Unet++ berusaha agar informasi dari layer encoder tidak hilang hingga sampai ke decoder. Sehingga dibuat banyak layer baru diantara encoder dan decoder pada kedalaman yang sama untuk menampung dan memproses informasi tersebut dengan bantuan concatenate.

#### IV. UJI COBA

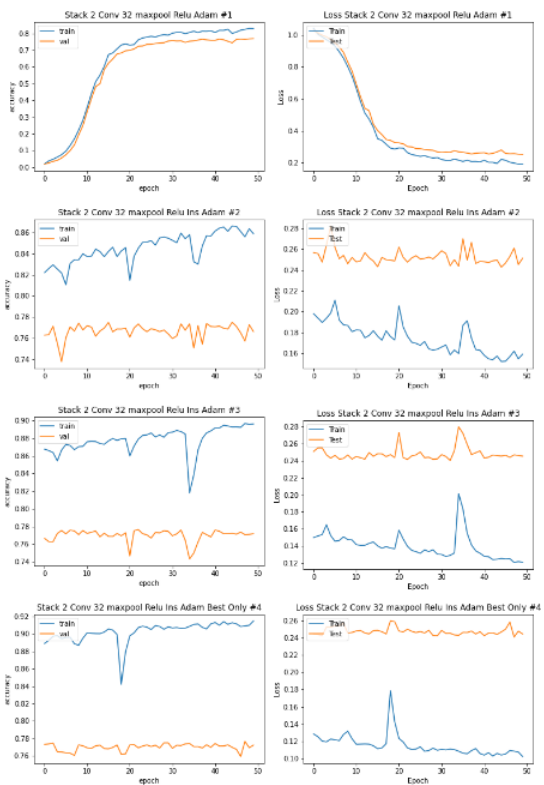
Uji coba dibagi menjadi 2 bagian yaitu model unet dan model Unet++. Saat melakukan sesi training akan dipecah menjadi beberapa bagian karena kemampuan GPU dan RAM yang tersedia tidak mendukung proses training secara penuh. Saat training digunakan memory yang ideal sebesar 25 GB dan menggunakan Tesla P100-PCIE sebagai GPU. Saat testing, index 81 dari dataset utama sebelum dilakukan pembagian split set akan digunakan. Pemilihan index tersebut karena terdapat perbedaan antara model generator dan bukan generator pada jumlah datasetnya. Array 81 adalah index yang terdapat pada testing untuk semua model karena menggunakan random seed yang sama saat pembagian set, tetapi pemilihan index data training kedua jenis model berbeda. Karena gambar berupa 3 dimensi dan nodule selalu berada di posisi tengah, maka ditampilkan gambar 2 dimensi

untuk setiap axis dengan fokus pada sisi tengah axis (jumlah pixel axis terfokus / 2).

Akan ada 4 bagian tersedia pada plot prediksi. Bagian  $x\_test$  adalah data yang tidak diikuti dalam proses training. Bagian ground truth adalah bagian nodule yang sudah ditentukan oleh radiologist, bagian predicted adalah hasil prediksi yang ditebak oleh model training dan bagian blended adalah ground truth yang overlapped dengan bagian predicted. Semua perhitungan akurasi testing akan menggunakan jaccard index. Model pertama akan menggunakan arsitektur U-Net biasa dan model kedua menggunakan U-Net++

Uji coba hanya akan menggunakan Adam[10] sebagai optimizer saat training. Pemilihan optimizer ini dikarenakan momentum yang dimiliki Adam bersifat dinamis sehingga terdapat lebih sedikit hyperparameter yang perlu disesuaikan dibandingkan optimizer lainnya. Momentum ini akan dipakai untuk mempengaruhi learning rate saat training. Adam bersifat stochastic, yang mana Adam akan memilih salah satu data dari data training untuk dihitung gradientnya lalu melakukan update parameter.

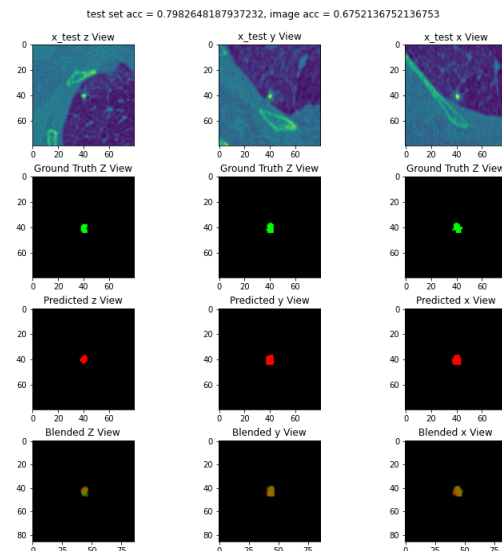
A. Model U-Net



Gambar 18. Hasil Training Model F

Pada Gambar 18, sesi training dibagi menjadi 4 bagian yang mana terdapat 50 epoch untuk setiap bagian. Proses training dilakukan tanpa menggunakan generator data augmentasi. Dataset yang digunakan adalah 860 gambar dan 400 gambar hasil augmentasi, dengan pembagian 70% untuk training, 20% untuk validation, 10% untuk testing. Saat selesai menjalani 200 epoch, akurasi training berada dikisaran  $\pm 91\%$  dan akurasi validasi berada dikisaran  $\pm 77\%$ . Pada saat dilakukan prediksi, menggunakan model bagian pertama (50 epoch pertama) menghasilkan akurasi testing

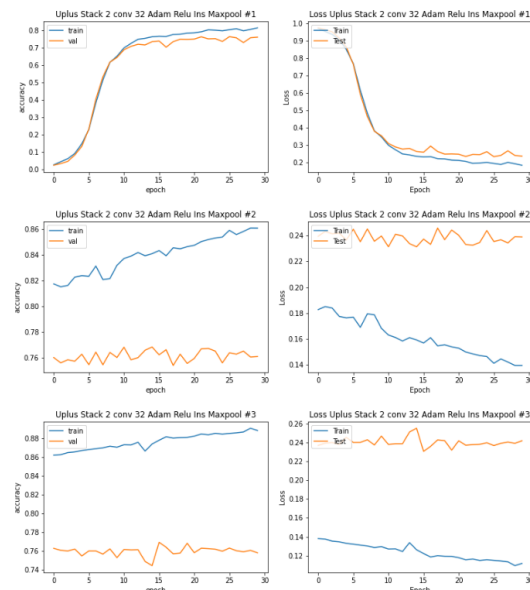
sebesar 77.0592 %, bagian kedua menghasilkan akurasi testing sebesar 77.5319%, bagian ketiga menghasilkan akurasi testing sebesar 79.0632%, bagian keempat menghasilkan akurasi testing sebesar 79.8264%. Pada model ini, dilakukan testing menggunakan parameter callback nilai maximum training saat training dan tidak menggunakan parameter, akurasi testing model tanpa callback adalah 79.3363% dan saat menggunakan callback adalah 79.8264. Berdasarkan hasil testing, maka akan dipakai model keempat karena memiliki akurasi terbesar untuk melakukan plotting.



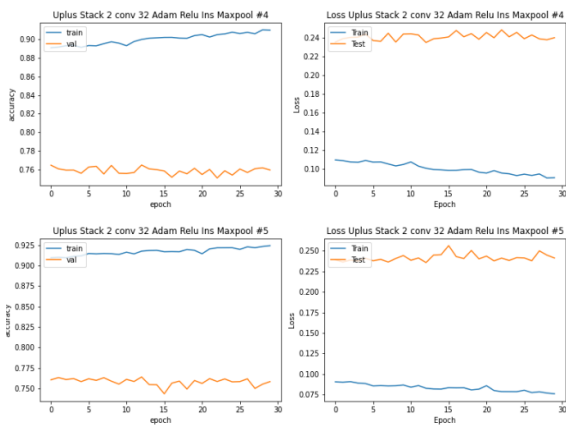
Gambar 19. Contoh Hasil Prediksi Model F

Pada gambar 19, gambar yang digunakan adalah array ke 81 dari dataset utama. Hasil prediksi menggunakan Jaccard Index menghasilkan akurasi sebesar 67.5213%. Warna hijau adalah ground truth (segmentasi yang diberikan oleh radiologist), warna merah adalah hasil prediksi, warna coklat adalah bagian intersect antara ground truth dan hasil prediksi. Area ground truth sangat penting karena menentukan tingkat akurasi dari gambar.

B. Model U-Net++

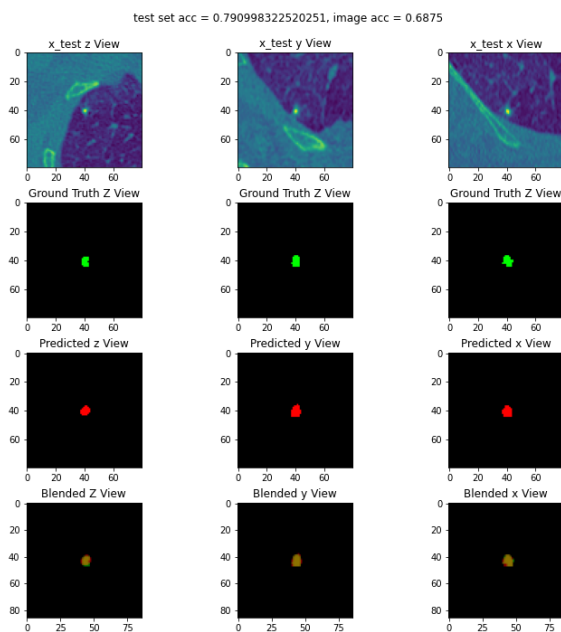


Gambar 20. Hasil Training Model H



Gambar 20. Hasil Training Model H (Lanjutan)

Pada Gambar 20, sesi training dibagi menjadi 5 bagian yang mana terdapat 30 epoch untuk setiap bagian. Proses training dilakukan tanpa menggunakan generator data augmentasi. Dataset yang digunakan adalah 860 gambar dan 400 gambar hasil augmentasi, dengan pembagian 70% untuk training, 20% untuk validation, 10% untuk testing. Saat selesai menjalani 150 epoch, akurasi training berada dikisaran  $\pm 92\%$  dan akurasi validasi berada dikisaran  $\pm 76\%$ . Pada saat dilakukan prediksi, menggunakan model bagian pertama (50 epoch pertama) menghasilkan akurasi testing sebesar 75.8763%, bagian kedua menghasilkan akurasi testing sebesar 76.3644%, bagian ketiga menghasilkan akurasi testing sebesar 78.4958%, bagian keempat menghasilkan akurasi testing sebesar 79.0998%, bagian kelima menghasilkan akurasi testing sebesar 78.1955%. Berdasarkan hasil testing, maka akan dipakai model keempat karena memiliki akurasi terbesar untuk melakukan plotting.



Gambar 21. Contoh Hasil Prediksi Model H (Lanjutan)

Pada Gambar 21, gambar yang digunakan adalah array ke 81 dari dataset utama. Hasil prediksi menggunakan Jaccard Index menghasilkan akurasi sebesar 68.75%. Warna hijau adalah ground truth, warna merah adalah hasil prediksi,

warna coklat adalah bagian intersect antara ground truth dan hasil prediksi. Kebenaran area ground truth yang diberikan radiologist sangat penting karena menentukan tingkat akurasi dari gambar.

### V. KESIMPULAN

Pada subbab ini dijelaskan mengenai kesimpulan dari pembuatan 2 model untuk program prediksi nodule. Nilai skor jaccard index untuk testing model U-Net++ tidak terlalu berbeda jauh hasilnya dari U-Net biasa. Meskipun U-Net++ memiliki komponen layer dan aktivasi yang sama dan memiliki banyak layer daripada U-Net biasa tetap tidak memiliki pengaruh yang signifikan pada akurasi testing.

### DAFTAR PUSTAKA

- [1] K. Mori, "CAD in lung," in *Handbook of Medical Image Computing and Computer Assisted Intervention*, Elsevier, 2020, pp. 91–107.
- [2] F. Isensee and K. H. Maier-Hein, "An attempt at beating the 3D U-Net," *arXiv Prepr. arXiv1908.02182*, 2019.
- [3] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "Unet++: A nested u-net architecture for medical image segmentation," in *Deep learning in medical image analysis and multimodal learning for clinical decision support*, Springer, 2018, pp. 3–11.
- [4] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.
- [5] Y. Wu and K. He, "Group normalization," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 3–19.
- [6] P. Baldi, "Autoencoders, unsupervised learning, and deep architectures," in *Proceedings of ICML workshop on unsupervised and transfer learning*, 2012, pp. 37–49.
- [7] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*, 2015, pp. 234–241.
- [8] J. Pedrosa *et al.*, "LNDb: a lung nodule database on computed tomography," *arXiv Prepr. arXiv1911.08434*, 2019.
- [9] G. L. Kindlmann, "An self-contained explanation of image orientation and the 'measurement frame', with connections to the NRRD format (Version 0.5)," 2010.
- [10] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv Prepr. arXiv1412.6980*, 2014.

# AUTOMATIC PARENTAL GUIDE SCENE CLASSIFICATION MENGGUNAKAN METODE DEEP CONVOLUTIONAL NEURAL NETWORK DAN LSTM

Riko Gunawan, *Teknologi Informasi Institut Sains dan Teknologi Terpadu Surabaya*,  
 Yosi Kristian, *Teknologi Informasi Institut Sains dan Teknologi Terpadu Surabaya*

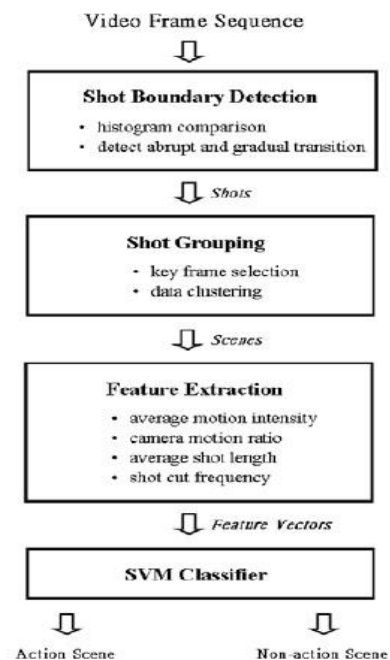
**Abstrak**— Menonton film merupakan salah satu hobi yang paling digemari oleh berbagai kalangan. Seiring dengan semakin bertambahnya film yang beredar di pasaran, semakin banyak pula konten tidak pantas pada film-film tersebut. Oleh karena itu, dibutuhkan sebuah metode untuk mengklasifikasikan film agar konten yang ditonton sesuai dengan usia penonton. Konten film yang kurang cocok untuk pengguna di bawah umur yang akan diklasifikasikan pada penelitian ini antara lain: kekerasan, pronografi, kata-kata kasar, minuman keras, penggunaan obat-obatan terlarang, merokok, adegan mengerikan (horror) dan intens. Metode klasifikasi yang digunakan berupa modifikasi dari convolutional neural network dan LSTM. Gabungan kedua metode ini dapat mengakomodasi data training dalam jumlah yang kecil, serta dapat melakukan multi klasifikasi berdasarkan video, audio, dan subtitle film. Penggunaan multi klasifikasi ini dikarenakan sebuah film selalu memiliki lebih dari satu klasifikasi. Dalam proses training dan testing pada penelitian ini digunakan sebanyak 1000 data untuk klasifikasi video, 600 data klasifikasi audio, dan 400 data klasifikasi subtitle yang didapatkan dari internet. Dari hasil percobaan dihasilkan tingkat akurasi yang diukur dengan menggunakan F1-Score sebesar 0.922 untuk klasifikasi video, 0.741 untuk klasifikasi audio, dan 0.844 untuk klasifikasi subtitle dengan rata-rata akurasi sebesar 0.835. Pada penelitian berikutnya akan dicoba dengan menggunakan metode Deep Convolutional Neural Network yang lain serta dengan memperbanyak jumlah dan variasi dari data testing.

**Kata Kunci**—Convolutional Neural Network, Deep Learning, LSTM, Movie Classification

## I. PENDAHULUAN

Pesatnya perkembangan dalam dunia perfilman menciptakan sebuah tantangan baru bagi penikmat film, khususnya mereka yang telah berkeluarga dan memiliki anak. Hal ini dikarenakan dalam proses pembuatannya, film yang dibuat dapat memiliki beragam konten yang kurang sesuai untuk dinikmati oleh semua kalangan [1]–[3]. Konten yang dimaksud dalam hal ini dapat berupa adegan kekerasan [4], [5], horror, pornografi, dan lain-lain. Studi lebih lanjut menunjukkan, bahwa anak-anak berumur 8 tahun yang terpapar tontonan dengan unsur kekerasan memiliki kecenderungan perilaku agresif ketika mereka mencapai umur 18 tahun [1]. Selain konten kekerasan, konten horror juga memiliki dampak negatif khususnya secara psikologis. Semakin sering seorang anak terpapar dengan konten horror, maka anak akan lebih mudah memiliki phobia [6]. Pada penelitian ini, peneliti bermaksud untuk memberikan sebuah metode klasifikasi konten film, yang lebih lanjut akan disebut sebagai scene. Klasifikasi scene ini diharapkan dapat

bermanfaat bagi orang tua untuk membantu dalam membatasi paparan scene film, khususnya yang kurang sesuai bagi anak-anak. Pendekatan klasifikasi yang akan digunakan berupa visual, audio, dan teks [7] dalam scene film. Jenis klasifikasi yang digunakan adalah: Violence & Gore [8], [9], Sex & Nudity [10], Profanity [11], [12], Alcohol, Drugs, and Smoking, dan Frightening & Intense.



Gambar 1. Alur klasifikasi action scene menggunakan SVM

## II. TINJAUAN PUSTAKA

### A. Action Scene Detection with Support Vector Machines

Pada paper ini dilakukan pendekatan metode berdasarkan konstruksi dari struktur video. Dalam hal ini penulis hanya menggunakan fitur visual dari video sebagai input dari SVM. SVM digunakan oleh penulis sebagai metode klasifikasi dikarenakan SVM memiliki tingkat akurasi yang cukup tinggi dengan training sample yang relatif kecil. Hasil klasifikasi dari input video akan berupa action scenes atau bukan. Alur klasifikasi action scene dapat dilihat pada Gambar 1.

Tahap pertama yang dilakukan oleh penulis dalam melakukan klasifikasi ini adalah dengan membagi video input menjadi shots atau frame, kemudian akan dipilih beberapa key frame dari hasil segmentasi tersebut. Key frame yang dimaksud adalah sebuah frame yang dianggap

dapat mewakili konten secara utuh dari sebuah video. Untuk mendapatkan keyframe ini, penulis menggunakan metode clustering.

Setelah didapatkan keyframe yang sesuai, maka hasil akan diteruskan kedalam SVM. Penulis memilih metode SVM ini dikarenakan memiliki structural risk minimalization principle, sehingga dapat mengurangi jumlah kesalahan pada training set meskipun data yang digunakan tidak terlalu banyak. Hasil dari penelitian ini akan ditunjukkan pada tabel berikut, dimana penulis menggunakan 4 buah film sebagai test video.

TABEL 1  
TINGKAT AKURASI KLASIFIKASI TEST VIDEO

| Movie ID | No of Action Scene | Correct Detection | Missed Detection | False Detection |
|----------|--------------------|-------------------|------------------|-----------------|
| 1        | 21                 | 17                | 4                | 6               |
| 2        | 16                 | 14                | 2                | 3               |
| 3        | 14                 | 13                | 1                | 3               |
| 4        | 19                 | 16                | 3                | 5               |

**B. Violence Detection in Video Using Computer Vision Techniques**

Tujuan dari penelitian ini adalah agar dapat menemukan sebuah action recognition yang dapat digunakan untuk mendeteksi perkelahian atau tindak kekerasan, baik dalam film maupun hasil dari video surveillance. Selain itu peneliti juga bertujuan untuk menghasilkan sebuah dataset video tindak kekerasan dengan menggunakan metode action recognition STIP dan MoSIFT.

TABEL 2  
TINGKAT AKURASI DETECTION MENGGUNAKAN CLIP HOCKEY

| Vocabulary | STIP (HOG) + HIK | STIP (HOF) + HIK | MoSIFT + HIK |
|------------|------------------|------------------|--------------|
| 50         | 87.8%            | 83.5%            | 87.5%        |
| 100        | 89.1%            | 84.3%            | 89.4%        |
| 150        | 89.7%            | 85.9%            | 89.5%        |
| 200        | 89.4%            | 87.5%            | 90.4%        |
| 300        | 90.8%            | 87.2%            | 90.4%        |
| 500        | 91.4%            | 87.4%            | 90.5%        |
| 1000       | 91.7%            | 88.6%            | 90.0%        |

Agar dapat mencapai tujuan di atas, peneliti mengumpulkan 1000 video clip dari olahraga hockey. Setiap video clip berisi 50 frames dengan ukuran 420x576 pixels dan proses pemilahan apakah video termasuk kekerasan atau tidak dilakukan secara manual oleh peneliti. Video clip yang telah dikumpulkan akan diteruskan kedalam metode STIP dan MoSIFT sebagai model dari action recognition. Selain menggunakan kedua metode di atas, peneliti juga menggunakan Bag of Words yang diadaptasi sehingga menghasilkan cluster yang didapat dari metode k-means clustering. Dalam penelitian ini cluster yang digunakan sebagai test sebanyak 50, 100, 150, 200, 300, 500 dan 1000 cluster center.

TABEL 3  
TINGKAT AKURASI DETECTION MENGGUNAKAN CLIP FILM

| Vocabulary | STIP (HOG) + HIK | STIP (HOF) + HIK | MoSIFT + HIK |
|------------|------------------|------------------|--------------|
| 50         | 44.5%            | 51.2%            | 76.0%        |
| 100        | 45.0%            | 56.5%            | 79.5%        |
| 150        | 49.0%            | 59.0%            | 80.0%        |
| 200        | 46.5%            | 53.5%            | 80.0%        |
| 300        | 44.5%            | 52.5%            | 87.5%        |
| 500        | 44.5%            | 50.5%            | 89.5%        |
| 1000       | 38.5%            | 52.5%            | 89.0%        |

Dari kedua tabel di atas, peneliti mengambil kesimpulan bahwa metode MoSIFT memiliki tingkat akurasi yang lebih baik jika dibandingkan dengan metode STIP. Peneliti juga menyimpulkan bahwa penambahan jumlah vocabulary tidak menghasilkan peningkatan akurasi yang berarti.

**C. Horror Video Scene Recognition via Multiple Instance Learning**

Penelitian yang dilakukan dalam paper ini akan melalui tiga tahapan utama, yaitu video segmentation, feature extraction, dan multiple instance learning method. Dalam proses video segmentation, peneliti menggunakan metode mutual information (MI) berdasarkan dari [13]. Metode ini menggunakan video transition dan fades sebagai dasar acuan.

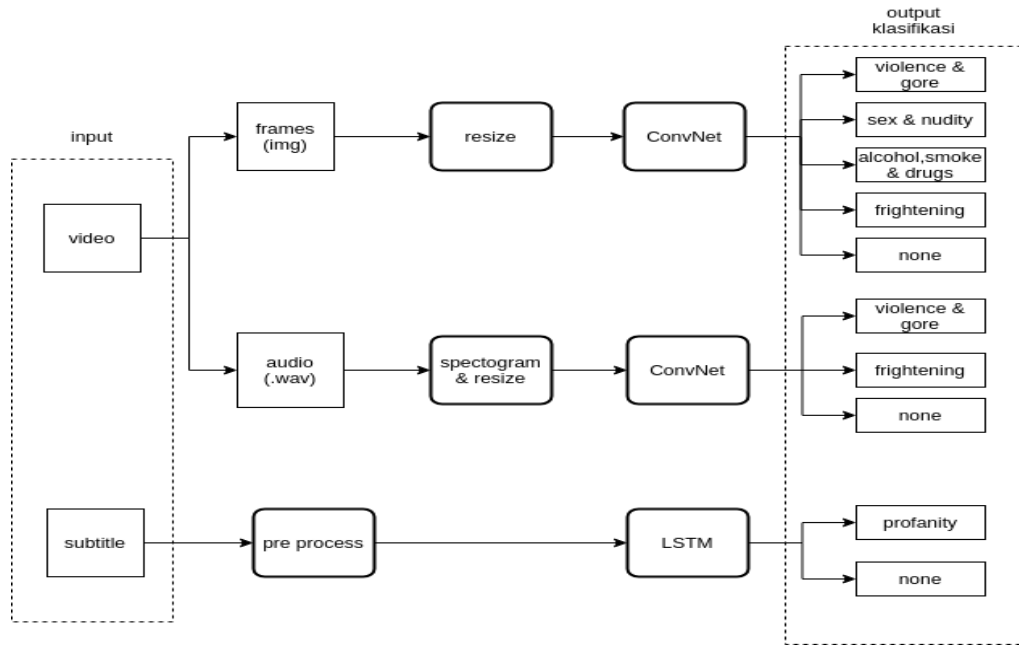
Ketika proses segmentasi telah selesai, maka hasil dari setiap segmen akan diambil sebuah frame utama yang akan dijadikan input. Visual feature extraction akan dilakukan pada tiap keyframe dengan mencari variasi warna dengan rumus covariance L (Luv coor frame) sebagai berikut:

$$\rho = \begin{bmatrix} \sigma_L^2 & \sigma_{Lu}^2 & \sigma_{Lv}^2 \\ \sigma_{Lu}^2 & \sigma_L^2 & \sigma_{uv}^2 \\ \sigma_{Lv}^2 & \sigma_{uv}^2 & \sigma_v^2 \end{bmatrix} \tag{1}$$

Selain feature extraction pada keyframe, peneliti juga melakukan feature extraction pada audio dan color emotional features sebelum akhirnya akan diteruskan kedalam tahapan multiple instance learning. Multiple-Instace Learning (MIL)[6] merupakan sebuah metode supervised classification dimana setiap training class memiliki asosiasi terhadap sebuah pola. Pada kasus ini, sebuah film akan diklasifikasikan sebagai film horror jika salah satu segmentasi terklasifikasi sebagai film horror. Sebagai perbandingan, peneliti juga menggunakan metode Continuous K-Nearest Neighbour (CKNN), EM-DD, serta SI.

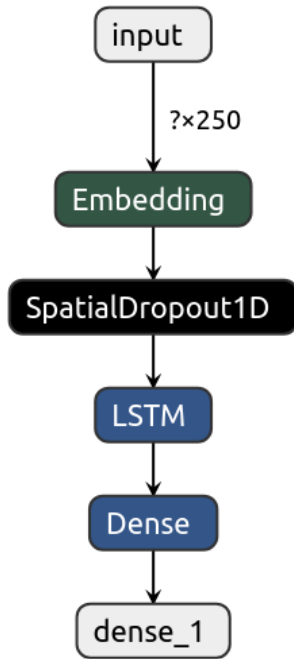
**III. METODE PENELITIAN**

Proses yang dilakukan dalam penelitian ini terbagi menjadi 2 bagian yaitu pengumpulan data dan pembuatan model Deep Convolutional Neural Network dan LSTM sebagai penentu klasifikasi. Skema proses ini lebih jelas dapat dilihat pada Gambar 2.

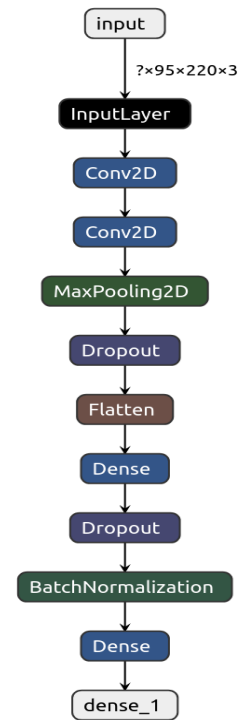


Gambar. 2. Skema metode penelitian

smoke and drugs, profanity, dan none. Klasifikasi ini didasarkan pada klasifikasi parental guidance dari imdb.



Gambar. 3. Model arsitektur LSTM untuk klasifikasi subtitle



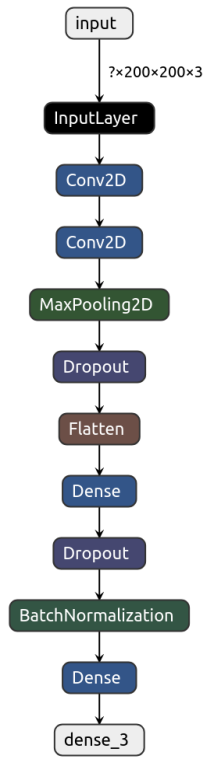
Gambar. 4. Model Arsitektur ConvNet untuk klasifikasi audio

**A. Proses Pengumpulan Data**

Data yang dikumpulkan oleh peneliti berupa video clip didapatkan melalui beberapa sumber, baik diunduh melalui situs youtube maupun pemotongan film secara manual dari file film digital. Film yang dikumpulkan oleh peneliti merupakan film mancanegara dengan subtitle berbahasa Inggris dengan berbagai genre film kecuali animasi. Data film yang berhasil dikumpulkan oleh peneliti lebih lanjut dipotong sehingga setiap video clip akan memiliki durasi sepanjang 10 detik beserta dengan pemotongan subtitle yang sesuai dengan durasi video. Potongan video dan subtitle kemudian diberi label secara manual dengan 6 klasifikasi yaitu violence and gore, sex and nudity, frightening, alcohol,

Setelah setiap potongan video dan subtitle berhasil dikumpulkan dan diberi label, maka dilakukan proses lebih lanjut. Potongan video kemudian dikonversi menjadi image dan spectrogram. Proses konversi dari potongan video menjadi image adalah dengan mengambil 1 frame untuk setiap detik dari video, maka setiap video menghasilkan 10 image yang kemudian diresize ulang menjadi berukuran

200x200 pixel. Sedangkan spectrogram didapat dari mengkonversi video menjadi file .wav yang kemudian dikonversi menjadi gambar spectrogram dengan jenis amplitude. Hasil gambar spectrogram ini akan diresize menjadi berukuran 95x220 pixel.



Gambar. 5. Model Arsitektur ConvNet untuk klasifikasi video

Sedangkan subtitle dari video akan melalui proses normalisasi, case folding dan tokenisasi. Proses ini bertujuan agar didapatkan data subtitle yang bebas dari noise seperti simbol khusus dan angka yang terdapat pada subtitle. Dimana subtitle yang dikumpulkan oleh peneliti berupa subtitle film dengan bahasa inggris dan subtitle yang masih belum disensor agar klasifikasi yang didapat lebih akurat

**B. Pembuatan Model**

Model arsitektur yang digunakan oleh peneliti merupakan model custom yang dibuat oleh peneliti berdasarkan dari simple CNN dan simple LSTM. Model arsitektur untuk setiap tahapan dapat dilihat pada Gambar 3,4, dan 5. Dalam pembuatan model ini juga diberikan dropout layer sebesar 0.3 agar tidak terjadi overfitting pada data pelatihan. Layer ini bertujuan untuk memilih neuron secara acak untuk tidak dipakai dalam proses training. Selain itu peneliti juga menambahkan layer batch normalization hal ini dilakukan untuk meningkatkan akurasi dari model serta mengurangi kebutuhan data training yang besar. Namun dropout layer ini tidak digunakan pada model LSTM, hanya pada model ConvNet.

**IV. HASIL DAN PEMBAHASAN**

Dalam bagian ini dijelaskan tentang hasil dari uji coba yang dilakukan peneliti untuk mendapatkan akurasi terbaik dari model yang dibuat. Testing yang dilakukan oleh peneliti

adalah dengan melakukan variasi pada jumlah cluster data testing, alat ukur tingkat akurasi yang digunakan oleh peneliti adalah F1 score dari hasil model yang digunakan. Data yang digunakan terbagi menjadi data training dan data testing dengan 90% sebagai training data dan 10% sebagai testing dikarenakan terbatasnya jumlah data yang dikumpulkan oleh peneliti.

Jumlah variasi cluster data yang digunakan oleh peneliti adalah dengan variasi 30,100,150, dan 200 cluster data untuk setiap kategori tag yang bersangkutan pada setiap model. Sebagai contoh pada model klasifikasi subtitle yang menghasilkan 2 label klasifikasi profanity dan none, dengan menggunakan cluster data sebesar 30 maka data yang akan digunakan adalah 30 data dengan label profanity dan 30 data dengan label none.

TABEL 4  
TINGKAT AKURASI DETECTION DENGAN BATCH NORMALIZATION

| Vocabulary per Label | Subtitle | Video | Audio | Rata-rata |
|----------------------|----------|-------|-------|-----------|
| 30                   | 0.667    | 0.848 | 0.303 | 0.606     |
| 100                  | 0.649    | 0.881 | 0.707 | 0.745     |
| 150                  | 0.756    | 0.877 | 0.733 | 0.788     |
| 200                  | 0.844    | 0.922 | 0.741 | 0.835     |

Tingkat akurasi klasifikasi test video, akurasi detection menggunakan clip hockey, akurasi menggunakan clip film, dan akurasi dengan batch normalization dapat dilihat pada Tabel 1, 2, 3, dan 4. Dari hasil percobaan yang dilakukan terhadap model ditemukan bahwa penggunaan batch normalization layer dapat meningkatkan akurasi dari data training, khususnya pada data spectrogram audio yang memiliki total data yang jauh lebih kecil jika dibandingkan dengan data video. Selain itu peningkatan vocabulary pada model juga memiliki dampak yang cukup signifikan pada peningkatan akurasi secara keseluruhan dalam mengklasifikasikan jenis label pada cuplikan film.

**V. KESIMPULAN**

Dengan menggunakan gabungan dari model yang telah dibuat oleh peneliti khususnya dengan menggunakan batch normalization layer dapat memberikan nilai akurasi yang tinggi dengan rata-rata sebesar 0.835 meskipun dengan jumlah data training yang relatif kecil. Namun pada penelitian mendatang akan dicoba dengan menggunakan metode Deep Convolutional Neural Network yang lain serta dengan memperbanyak jumlah dan variasi dari data testing.

**DAFTAR PUSTAKA**

- [1] L. H. Chen, H. W. Hsu, L. Y. Wang, and C. W. Su, "Violence detection in movies," 2011, doi: 10.1109/CGIV.2011.14.
- [2] U. A. Khan, N. Ejaz, M. A. Martinez-Del-Amor, and H. Sparenberg, "Movies tags extraction using deep learning," 2017, doi: 10.1109/AVSS.2017.8078459.
- [3] Q. Dai *et al.*, "Fudan-Huawei at MediaEval 2015: Detecting violent scenes and affective impact in movies with deep learning," in *CEUR Workshop Proceedings*, 2015, vol. 1436.
- [4] L. H. Chen, C. W. Su, C. F. Weng, and H. Y. M. Liao, "Action scene detection with support vector machines," *J. Multimed.*, vol.

- 4, no. 4, 2009, doi: 10.4304/jmm.4.4.248-253.
- [5] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Advances in Neural Information Processing Systems*, 2014, vol. 1, no. January.
- [6] J. Wang, B. Li, W. Hu, and O. Wu, "Horror video scene recognition via multiple-instance learning," 2011, doi: 10.1109/ICASSP.2011.5946656.
- [7] L. Li, L. Xiao, N. Wang, G. Yang, and J. Zhang, "Text classification method based on convolution neural network," in *2017 3rd IEEE International Conference on Computer and Communications, ICC3 2017*, 2018, vol. 2018-January, doi: 10.1109/CompComm.2017.8322884.
- [8] E. Acar, M. Irrgang, D. Maniry, and F. Hopfgartner, "Detecting violent content in hollywood movies and user-generated videos," *Adv. Comput. Vis. Pattern Recognit.*, vol. 66, 2015, doi: 10.1007/978-3-319-14178-7\_11.
- [9] E. Bermejo Nievas, O. Deniz Suarez, G. Bueno García, and R. Sukthankar, "Violence detection in video using computer vision techniques," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2011, vol. 6855 LNCS, no. PART 2, doi: 10.1007/978-3-642-23678-5\_39.
- [10] M. Moustafa, "Applying deep learning to classify pornographic images and videos," *arXiv Prepr. arXiv1511.08899*, 2015.
- [11] H. Yenala, A. Jhanwar, M. K. Chinnakotla, and J. Goyal, "Deep learning for detecting inappropriate content in text," *Int. J. Data Sci. Anal.*, vol. 6, no. 4, 2018, doi: 10.1007/s41060-017-0088-4.
- [12] N. Nikhil, R. Pahwa, M. K. Nirala, and R. Khilnani, "Lstms with attention for aggression detection," in *Proceedings of the first workshop on trolling, aggression and cyberbullying (TRAC-2018)*, 2018, pp. 52–57.
- [13] Z. Cernekova, C. Kotropoulos, N. Nikolaidis, and I. Pitas, "Video shot segmentation using fusion of SVD and mutual information features," in *2005 IEEE International Symposium on Circuits and Systems*, 2005, pp. 3849–3852.



# 2D Data Visualization Tools Menggunakan Flask dan AngularJS

Jerryl Jeovano, *Departemen Teknik Informatika Institut Sains dan Teknologi Terpadu Surabaya*

**Abstrak** — Data merupakan bagian penting dari semua aplikasi. Namun seiring perkembangan aplikasi dan kecepatan stream data, volume data yang dihasilkan telah melebihi milyaran data. Semakin besar volume data, maka semakin sulit juga data tersebut dianalisa. Tujuan pembuatan software ini untuk memvisualisasikan data user dari file yang diupload kepada sistem. Memudahkan tim analis untuk memprediksi dan memajukan sebuah usaha. Visualisasi data berjalan fleksibel, sehingga user dapat menentukan sendiri data pada bagian axis-axisnya. Pembuatan website ini menggunakan framework Flask yang berbasis pada bahasa pemrograman Python, database cassandra merupakan database nosql yang mana digunakan untuk penyimpanan data website ini. Sedangkan visualisasi, website ini menggunakan library DevExpress dan Google Chart yang berbasis AngularJS. AngularJS sendiri merupakan perluasan dari bahasa pemrograman HTML. Metodologi yang akan digunakan pada pembuatan website ini adalah Scrum. Prose pembuatan website ini dilakukan dalam 4 sprint. Masing-masing sprint dalam penelitian ini berlangsung dalam jangka waktu 14 hari. Dari hasil penelitian ini dapat disimpulkan bahwa penggunaan metode scrum untuk membuat web untuk visualisasi data berjalan dengan efektif karena dalam proses pembuatan web ini lebih terstruktur sehingga time management menjadi lebih efisien. Library DevExpress lebih mudah digunakan daripada menggunakan library Google Chart. Dalam menampilkan chart, library DevExpress hanya dapat menampilkan sebuah chart dalam satu halaman.

**Kata Kunci**—Data Visualization, Big Data, ETL, Python Flask.

## I. PENDAHULUAN

**D**ata merupakan bagian penting dari semua aplikasi.

Namun seiring perkembangan aplikasi dan kecepatan stream data, volume data yang dihasilkan telah melebihi milyaran data. Semakin besar volume data, maka semakin sulit juga data tersebut dianalisa. Penganalisaan data dilakukan agar dapat digunakan oleh tim analis untuk mendapatkan informasi-informasi yang dapat memajukan suatu bisnis.<sup>[1]</sup>

Visualisasi data memiliki peranan penting dalam menganalisa data. Visualisasi data yang di tampilkan harus divalidasi terlebih dahulu oleh user agar data yang tampil sesuai dengan yang diinginkan. Sering kali, visualisasi data yang digunakan tidak sesuai dengan kebutuhan user atau sistem sudah berkembang sehingga laporan-laporan yang disediakan tidak sesuai dengan kebutuhan user sekarang. Oleh karena itu, visualisasi data merupakan aktivitas yang seharusnya diatur oleh user dan berguna untuk kepentingan user itu sendiri.

Jerryl Jeovano, Departemen Teknik Informatika, Institut Sains dan Teknologi Terpadu Surabaya, Surabaya, Jawa Timur, Indonesia (e-mail: jerryl1@stts.edu)

Dengan adanya permasalahan tersebut maka munculah ide untuk membuat sebuah visualisasi data 2D dalam bentuk sebuah website.

Website ini diharapkan dapat membantu user dalam pembuatan laporan. Laporan ini nantinya dapat membantu tim analisis dalam hal mengambil sebuah keputusan yang mampu memajukan sebuah peluang usaha.

Pengolahan data pada website ini mengacu pada teori pengolahan data OLAP dan ETL yang akan dijelaskan pada bagian teori penunjang. Proses pengolahan data ini diperlukan sebelum memvisualisasikan data, agar data yang ditampilkan tidak ada yang redundant (sama). Dengan adanya pengolahan data, maka user akan mendapatkan hasil visual yang memiliki *value* terbaik.

## II. TINJAUAN PUSTAKA

Di dalam bab ini akan dibahas beberapa teori terkait dengan penelitian ini. Detail pembahasan adalah sebagai berikut:

### A. *Big Data* [1]

Data adalah jumlah, karakter, atau simbol tempat operasi dilakukan oleh komputer, yang dapat disimpan dan dikirim dalam bentuk sinyal listrik dan direkam pada media perekaman magnetik, optik, atau mekanis. Big Data juga data tetapi dengan ukuran besar. Big Data adalah istilah yang digunakan untuk mendeskripsikan kumpulan data yang berukuran sangat besar namun tumbuh secara eksponensial seiring waktu. Singkatnya, data tersebut sangat besar dan kompleks sehingga tidak ada alat manajemen data tradisional yang dapat menyimpan atau memprosesnya secara efisien.

### B. *OLAP*

OLAP (*Online Analytical Processing*) [2] adalah metode pendekatan untuk menyajikan jawaban dari permintaan proses analisis yang bersifat dimensional secara cepat, yaitu desain dari aplikasi dan teknologi yang dapat mengoleksi, menyimpan, memanipulasi suatu data multidimensi untuk tujuan analis. OLAP adalah bagian dari kategori yang lebih global dari pemikiran bisnis, yang juga merangkum hubungan antara pelaporan dan penggalian data.

### C. *Extract, Transform, Load (ETL)*

Extract Transform and Load (ETL) adalah sekumpulan proses yang harus dilalui dalam pembentukan data warehouse. Dengan tujuan mengumpulkan, menyaring, mengolah dan menggabungkan data-data yang relevan dari berbagai sumber untuk disimpan ke dalam data warehouse.

### D. *Python Flask*

Python Flask [3] adalah *framework* atau API Python untuk aplikasi web. Flask juga merupakan aplikasi web WSGI yang

ringan. Yang berarti Flask memberi bantuan berupa *tools*, *library* dan *technology* yang memungkinkan untuk membangun aplikasi web. Flask merupakan bagian dari kategori kerangka kerja mikro. Kerangka kerja micro biasanya kerangka kerja dengan sedikit atau tidak ada ketergantungan ke *library* eksternal. Kerangka kerja flask lebih cepat dan mudah dengan kemampuan meningkatkan aplikasi yang kompleks daripada kerangka kerja Django dan juga lebih mudah dipelajari.

#### E. AngularJS

AngularJS adalah kerangka kerja struktural untuk aplikasi web dinamis dan open-source. Ini memungkinkan untuk menggunakan HTML sebagai bahasa template dan memungkinkan untuk memperluas sintaksis HTML untuk mengekspresikan komponen aplikasi dengan jelas dan ringkas. Pengikatan data dan injeksi ketergantungannya menghilangkan banyak kode yang saat ini harus ditulis. Dan itu semua terjadi dalam browser, menjadikannya mitra yang ideal dengan teknologi server apa pun.

#### F. Apache Cassandra

Apache Cassandra adalah sistem basis data terdistribusi open-source yang sangat kuat yang bekerja sangat baik untuk menangani volume besar catatan yang tersebar di beberapa server komoditas. Dapat dengan mudah ditingkatkan untuk memenuhi peningkatan permintaan yang tiba-tiba dengan menggunakan multi-node Cassandra cluster dan memenuhi persyaratan ketersediaan tinggi, tanpa satu titik kegagalan. Ini adalah salah satu database NoSQL paling efisien yang tersedia saat ini. DataStax menawarkan distribusi paket gratis dari Apache Cassandra. Ini juga mencakup berbagai alat lain seperti Pemasang Windows, DevCenter, dan dokumentasi profesional DataStax.

#### G. Trello

Trello adalah alat kolaborasi untuk mengatur dan kolaborasi proyek berbasis *cloud* untuk membantu pekerjaan tim serta memberi gambaran visual tentang apa yang sedang dikerjakan, siapa yang mengerjakannya, dan seberapa jauh mereka telah mencapai itu. Hal ini memberi pengguna visibilitas penuh dan kontrol atas proyek mereka, memungkinkan mereka untuk mengirimkan proyek tepat waktu dan sesuai anggaran.

#### H. Data Visualization

Visualisasi data adalah penyajian data dalam format gambar atau grafik. Ini memungkinkan para pembuat keputusan untuk melihat analitik yang disajikan secara visual, sehingga dapat memahami konsep-konsep yang sulit atau mengidentifikasi pola-pola baru. Dengan visualisasi interaktif, maka dengan mengambil konsep ini selangkah lebih maju dengan menggunakan teknologi untuk menelusuri bagan dan grafik untuk lebih detail. Tujuan utamanya adalah untuk menyaring dataset besar ke dalam grafik visual untuk memudahkan pemahaman hubungan yang kompleks dalam data. [4]

#### I. Library *werkzeug*

Library *werkzeug.utils* (*secure\_filename*) adalah library yang digunakan untuk mengamankan nama file. Sehingga ketika file disimpan di sistem, pengaksesan file lebih mudah.

Library *werkzeug.security* (*generate\_password\_hash*, *check\_password\_hash*) adalah library yang digunakan untuk mengencrypt password-password pengguna website. Sedangkan *check\_password\_hash* digunakan untuk melakukan pengecekan terhadap password yang terdaftar dengan password yang diinput oleh pengguna website ketika akan masuk ke dalam website.

#### J. Library *flask\_mail*

Library *flask\_mail* adalah library yang disediakan oleh Matt Wright yang akan memudahkan untuk menghubungkan aplikasi dengan provider email. Library ini merupakan API yang digunakan untuk mengatur SMTP pada aplikasi. Meskipun disebut sebagai API tetapi tidak sepenuhnya sebagai API pada umumnya, masih diperlukan melakukan beberapa perubahan opsi agar mailer dapat berjalan dengan baik. Library ini juga support untuk mengirimkan email dengan format HTML.

#### K. Library *Pandas*

Library *pandas* [5] adalah paket Python yang menyediakan struktur data yang cepat, fleksibel, dan ekspresif yang dirancang untuk membuat bekerja dengan data "relasional" atau "berlabel" mudah dan intuitif. Ini bertujuan untuk menjadi blok bangunan tingkat tinggi yang mendasar untuk melakukan analisis data dunia nyata yang praktis dengan Python. Selain itu, ia memiliki tujuan yang lebih luas untuk menjadi alat analisis / manipulasi data open source yang paling kuat dan fleksibel yang tersedia dalam bahasa apa pun.

#### L. Library *Datetime*

Library *datetime* digunakan untuk memanipulasi tanggal dan waktu. Fokus implementasi adalah pada ekstraksi atribut yang efisien untuk pemformatan dan manipulasi output. Objek *naive* tidak mengandung informasi yang cukup untuk secara jelas menemukan dirinya relatif terhadap objek tanggal / waktu lainnya.

#### M. Library *gviz\_api*

Library *gviz\_api* adalah library bantuan untuk pembentukan chart dari library Google Chart. Library ini membantu terhadap pemrosesan data dari json kepada data table. Selain membantu mempermudah proses data, library ini juga berfungsi untuk mengirimkan data tersebut ke dalam halaman HTML dan JavaScript. Meskipun terdapat library ini, tetap membutuhkan pemrosesan data pada Python yang tergolong cukup rumit. Dikarenakan banyak requirement yang harus dipenuhi agar grafik dapat terbentuk.

#### N. *The Truthful Art Data, Charts, and Maps for Communication* [6]

Aspek dari pemikiran kontekstual menyangkut dalam mengidentifikasi keadaan proyek yang memiliki beberapa faktor yang mempengaruhi dan mencirikan kondisi proyek yang dilakukan. Hal ini akan menentukan batas-batas ambisi kreatif dalam sebuah proyek visualisasi. Dalam melakukan visualisasi data, terdapat tiga buah role yang bertugas dalam menentukan visualisasi data yang diinginkan yaitu stakeholder, audience, dan visualiser. Berikut ini merupakan penjelasan untuk setiap rolanya:

##### 1) Stakeholder

Dalam sebuah situasi mengembangkan visualisasi, sangat penting sekali untuk membangun pemahaman tentang siapa yang terlibat. Hal ini merupakan sebuah

pertanyaan kunci untuk dijawab. Pelanggan akan selalu diinvestasikan secara khusus dalam suatu hal yang akan dikembangkan. Pelanggan biasanya akan menentukan apakah pekerjaan itu berada di jalur yang benar, menurut pandangan mereka. Mereka juga akan menentukan secara kritis kapan pekerjaan pada akhirnya memiliki kualitas yang cukup untuk dipertimbangkan selesai. Jika tidak ada pelanggan, maka hal ini merupakan upaya tunggal dan akan ada lebih banyak otonomi yang dapat membebaskan yang akan menjadi tanggung jawab lebih besar untuk mengarahkan semua masalah sendiri.

2) Audience

Terdapat beberapa karakteristik target yang diharapkan dari audience yang perlu dipertimbangkan dengan cermat. Hal ini berguna agar proyek yang dibuat tidak akan menjadi irasional dan inkonsisten. Terkadang jika kita memiliki pengetahuan langsung tentang audience, hal tersebut dapat dengan mudah menandai kebutuhan mereka dan membentuk pilihan yang sesuai dengan keinginan audience. Misalnya, jika memiliki grup pemirsa tetap yang kita kenal akan memahami konteks teknis dari data yang disajikan. Sehingga kita tidak perlu menyertakan penjelasan terperinci yang diperlukan untuk audience yang kurang berpengetahuan.

3) Visualiser

Desain visualisasi data benar-benar merupakan upaya multidisiplin. Variasi inilah yang memicu kekayaan subjek dan menjadikannya tantangan yang sangat menarik. Untuk menguasainya diperlukan serangkaian keterampilan, pengetahuan, dan sikap berbeda yang mendominasi berbagai tahap proses ini. Jika kolaborasi dimungkinkan, ada keuntungan yang jelas dalam menggabungkan berbagai kemampuan menjadi tantangan bersama. Visualisasi yang berfungsi paling baik akan menawarkan perpaduan keterampilan yang seimbang. Keberhasilan akan sulit dicapai jika sebuah tim terdiri dari dominasi yang condong pada keragaman kemampuan individu.

Saat memulai suatu proyek, mungkin tidak semua keadaan yang berpotensi mempengaruhi pekerjaan Anda akan dapat ditentukan. Banyak hal berubah, hal itu sebabnya kita perlu siap untuk mengakomodasi dampak faktor-faktor baru di setiap titik dalam proses. Tentu saja, semakin banyak hal yang dapat Anda definisikan, semakin banyak hal diperbaiki dan ini mengurangi ketidakpastian. Idealnya, hal ini digunakan untuk menghilangkan sebanyak mungkin hal yang tidak diketahui. Kendala tidak selalu merupakan hal yang buruk, namun sering dapat membantu untuk berinovasi. Berikut ini merupakan kendala-kendala yang terdapat dalam sebuah proyek visualisasi data:

1) Timescale

Kendala utama biasanya adalah berapa banyak waktu yang dibutuhkan untuk mengembangkan solusi. Sebagian besar proyek memiliki tenggat waktu, apakah ini dipaksakan, disepakati bersama atau ditetapkan sendiri. Bahkan jika tidak perlu mematuhi tenggat waktu, masih dapat berguna untuk menentukan tanggal target untuk membantu mempertajam kemajuan Anda. Di ujung timeline, ada tanggal mulai. Mungkin harus menunggu untuk kondisi tertentu sebelum memulai

pekerjaan. Memperkirakan durasi proyek hingga tingkat yang dapat dipercaya adalah hal yang sulit untuk dinilai. Biasanya kita tidak tahu berapa lama proyek akan berlangsung sampai selesai, yang seringkali terlambat untuk berguna. Bahkan dengan pengalaman bekerja di beragam proyek, proyek yang tampaknya serupa dapat berakhir dengan durasi tugas yang sangat berbeda.

2) Pressure

Pressure bergantung pada konteks proyek, faktor biaya tertentu mungkin ada. Beberapa proyek mungkin memiliki anggaran yang dialokasikan dan oleh karena itu biaya kegiatan staf terkait perlu dikelola secara bijaksana. Biaya apa yang akan dikeluarkan untuk membayar suatu barang yang dibutuhkan dalam membuat proyek tersebut. Tekanan lebih lanjut dapat muncul dari politik di sekitar kita sehingga kita dapat menerima panduan dari orang lain bahwa pesan-pesan tertentu perlu diremehkan atau diperkuat. Ini bisa menjadi masalah sulit untuk ditangani karena kita ingin menghormati persyaratan yang diterima, tetapi kita juga tidak ingin merusak integritas apa yang kita wakili. Mungkin ada kepekaan budaya untuk dipertimbangkan dalam membuat karya untuk audience dari berbagai daerah. Masalah seputar penggunaan citra, konotasi warna, atau simbolologi bentuk-bentuk tertentu mungkin perlu ditangani dengan hati-hati. Mungkin juga ada pertimbangan lingkungan, terutama yang menyangkut hasil pekerjaan kita yang perlu diperhatikan.

3) Design

Pembatasan sekitar pilihan desain tertentu merupakan hal yang umum, sering diinformasikan oleh pedoman gaya yang harus dipatuhi melalui penggunaan warna, jenis huruf dan font tertentu. Mungkin seperti perlu memasukkan logo, yang dapat mengambil ruang berharga dan tidak seimbang komposisinya, tetapi aturan adalah aturan dan oleh karena itu kita perlu tahu tentang hal-hal ini di awal. Pembatasan layout atau size mungkin juga ada untuk menentukan ruang di mana Anda harus bekerja. Misalnya, ketika memproduksi grafik untuk jurnal atau untuk keluaran digital yang perlu bekerja pada tablet atau smartphone, mungkin memiliki ruang yang cukup kecil untuk digunakan. Sebaliknya, output Anda mungkin harus sangat besar, yang dapat memperkenalkan berbagai tantangan dengan keterbacaan dan kualitas resolusi.

4) Technology

Terdapat banyak sekali alat, aplikasi, dan perpustakaan pemrograman dalam visualisasi data, yang menawarkan lanskap beragam kemampuan. Teknologi yang dimiliki aksesnya akan memengaruhi seberapa ambisius pekerjaan secara digital dan seberapa efisien yang dapat membuatnya.

Bagian berikutnya adalah penggunaan data yang akan divisualisasikan dengan melakukan empat langkah berbeda yang secara komprehensif memperkenalkan dengan potensi data mentah. Langkah pertama adalah data acquisition yang

melihat asal-usul dan metode berbeda untuk mengakses data, seperti melalui unduhan sistem, diperoleh dengan menggunakan API, atau melibatkan metode pengumpulan secara manual. Langkah berikutnya adalah data examination yang memetakan berbagai karakteristik yang menentukan jenis, ukuran, dan kondisi data. Untuk membedakan berbagai jenis data, mnemonik 'TNOIR' diusulkan yaitu textual, nominal, ordinal, interval, dan ratio.

Langkah ketiga adalah data transformation yang dibangun di atas data examination dimana hal ini berguna untuk mengidentifikasi cara-cara memodifikasi dan meningkatkan data untuk mempersiapkannya untuk digunakan. Data transformation meliputi cleaning yang menghapus data yang bermasalah, creating yang membuat sebuah kalkulas dan nilai konversi yang baru, dan consolidating yang berguna untuk memperkenalkan data lebih lanjut untuk memperluas atau menambahkan apa yang sudah dimiliki. Langkah terakhir adalah data exploration yang membahas nilai menggunakan teknik visualisasi untuk melengkapi pendekatan statistik sebagai cara untuk menemukan lebih banyak tentang kualitas dan wawasan yang terdapat pada data yang akan divisualisasikan.

*O. Data Visualization A Handbook for Data Driven Design* [6]

Selama beberapa dekade, banyak yang beranggapan bahwa visualisasi bekerja; bahwa mempelajari cara melakukannya dengan benar dapat membuat siapa pun menjadi lebih baik tidak hanya dalam mengomunikasikan fakta yang relevan tentang dunia, tetapi juga dalam mengeksplorasinya dan bahwa itu adalah kerajinan yang tidak didasarkan pada kecenderungan artistik, tetapi pada merangkul prinsip-prinsip dan heuristik tertentu yang berasal dari pengalaman dan penyelidikan ilmiah. Beberapa orang percaya ide-ide tersebut benar. Salah satu contohnya adalah The Hockey Stick Chart [7], yang dirancang oleh profesor Michael E. Mann, Raymond S. Bradley, dan Malcolm K. Hughes.

The hockey stick chart adalah salah satu visualisasi paling ikonik dan persuasif. Visualisasi data ini menjadi yang sangat baik dikarenakan memiliki kualitas tertentu. Tanpa adanya kualitas-kualitas ini dapat mengurangi nilai visualisasi sebuah data seperti kurang memahami arti dari sebuah data yang divisualisasikan. Berikut ini merupakan kualitas yang harus dimiliki dalam sebuah data yang akan divisualisasikan:

1) Truthful

Hal ini merupakan bagian yang paling penting dalam melakukan visualisasi data. Seluruh data yang dimiliki merupakan data yang asli secara menyeluruh. Menjadi jujur melibatkan dua strategi yang berbeda tetapi terhubung erat. Strategi yang pertama adalah menghindari penipuan diri. Kita sebagai manusia berevolusi untuk melihat pola noise yang tidak berarti. Kami langsung mengambil kesimpulan. Kami melihat hubungan sebab-akibat di mana hanya ada korelasi. Grafik yang jujur dibuat oleh orang-orang yang melakukan yang terbaik untuk mengatasi kelemahan intelektual dan bias kognitif dan ideologis dengan menerapkan pemikiran yang kritis. Strategi yang kedua adalah jujur dengan audiens. Tunjukkan pada mereka pemahaman terbaik tentang apa kenyataannya.

2) Functional

Hal berikutnya yang penting dalam sebuah data yang akan divisualisasikan adalah fungsionalitas dari sebuah data. Fungsional dari sebuah data dapat membantu dalam penggambaran data yang akurat, dan dibuat dengan cara yang memungkinkan orang melakukan operasi yang bermakna berdasarkan data seperti melihat perubahan waktu. Tujuan dari visualisasi sebuah data dalam bentuk grafik adalah memandu dalam mengambil keputusan untuk membentuk sebuah informasi.

3) Beautiful

Dalam melakukan visualisasi data harus menarik, dan menyenangkan secara estetika bagi audiens yang dituju. Beberapa hal bisa menjadi indah hanya karena menjadi efisien. Salah satunya yaitu dengan memberikan banyak hal menggunakan jumlah elemen minimum. Banyak perancang visualisasi dan ahli teori menekankan pentingnya kesederhanaan.

4) Insightful

Tujuan visualisasi adalah wawasan, bukan gambar [8]. Hal ini berarti bahwa visualisasi yang baik membersihkan jalan untuk membuat penemuan-penemuan berharga yang tidak akan dapat diakses jika informasi disajikan dengan cara yang berbeda. Visualisasi yang menawarkan pesan yang jelas dan sepele tidak berharga. Secara implisit, desainer meminta untuk berinvestasi dalam upaya membacanya dengan imbalan sedikit pengembalian.

5) Enlightening

Grafik yang mencerahkan adalah konsekuensi dari memperhatikan empat kualitas sebelumnya. Grafik yang jujur, fungsional, indah, dan berwawasan luas juga berpotensi menjadi pencerahan. Dengan memperhatikan empat kualitas sebelumnya, audience juga dapat memahami data yang divisualisasikan.

*P. Visualize This: The FlowingData Guide to Design, Visualization, and Statistics* [9]

Sebelum memulai bekerja untuk bagian visualisasi, hal yang pertama kali dibutuhkan adalah melakukan pengumpulan data yang akan divisualisasikan. Data merupakan hasil yang sangat penting untuk membuat sebuah visualisasi menjadi menarik. Jika tidak memiliki data yang menarik, maka visualisasi ini hanya akan menjadi sebuah graph yang terlupakan dan sebuah gambar yang tidak berguna.

Sebuah data yang menarik tidak akan datang secara langsung dan didapatkan secara cuma-cuma. Untuk memperolehnya memerlukan sebuah usaha yang keras dan biaya yang besar. Sebuah data yang menarik didapatkan dengan melakukan pengumpulan data yang didapatkan dari search engine, universitas, olahraga, dan sebagainya.

Setelah mendapatkan data yang diinginkan, selanjutnya adalah melakukan formatting pada data yang akan divisualisasikan. Bagian kedua ini merupakan melakukan format pada sebuah data agar dapat dibaca oleh komputer sehingga dapat dipahami dan diproses oleh komputer. Tiga buah format yang cocok agar komputer dapat memproses adalah dengan menggunakan delimiter text, JavaScript Object Notation (JSON), dan Extensible Markup Language (XML). Pada bagian ini format yang akan digunakan adalah JSON untuk melakukan proses agar data dapat divisualisasikan

*Q. Perbandingan Sistem yang Dibuat dengan Tableau dan PowerBI*

Perbedaan yang mendasar dari PowerBI [10] dan Tableau [11] adalah interface dari setiap software, target pasar, penggunaan software, dan jumlah data yang diproses. Tabel I menunjukkan perbedaan untuk PowerBI dan Tableau.

Tabel 1  
Tabel Perbandingan PowerBI dan Tableau

| Keterangan          | PowerBI   | Tableau   |
|---------------------|---|---|
| Volume Data         | Volume data yang dapat diproses terbatas max 250 MB.              | Volume data yang dapat diproses sangat besar.               |
| Interface           | Interface sederhana dan mudah untuk dipahami                      | Interface kurang baik dan sulit untuk dipahami              |
| Target Pasar        | Tidak direkomendasi untuk user yang memprioritaskan analisis data | Direkomendasi untuk user yang memprioritaskan analisis data |
| Penggunaan software | Mudah dipahami  | Sulit dipahami  |

Perbedaan yang pertama antara PowerBI dengan Tableau adalah volume data yang dapat diakses. PowerBI hanya dapat memproses data maksimal sebesar 250 MB. Sedangkan untuk Tableau tidak memiliki batasan volume data yang dapat diproses. Perbedaan berikutnya yaitu design interface yang dimiliki setiap software. PowerBI memiliki interface yang sederhana sehingga lebih mudah dipahami oleh pengguna. Sedangkan untuk software Tableau, interface lebih rumit sehingga pengguna yang pemula akan sedikit lebih bingung ketika menggunakan software Tableau. Perbedaan yang ketiga adalah target pasar yang dituju. Software PowerBI lebih mengutamakan untuk kelas menengah kebawah dikarenakan jumlah data yang dapat diproses terbatas. Sedangkan untuk software Tableau lebih mengutamakan untuk melakukan pemrosesan untuk big data dikarenakan tidak memiliki batas data yang dapat diproses

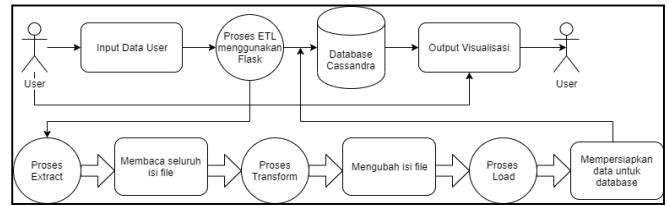
III. IMPLEMENTASI

A. Arsitektur sistem

Untuk menggunakan fitur-fitur yang disediakan pada website, customer memerlukan ID yang terdaftar. Untuk mendaftar, customer bisa melakukan registrasi sendiri dengan memasukan data-data yang diminta oleh website. Data-data yang diminta antara lain adalah alamat email, nama lengkap, dan lain sebagainya. Setelah terdaftar barulah customer dapat melakukan login kedalam sistem. Setelah login, user dapat melakukan pembuatan project baru dengan melakukan pemberian nama pada project tersebut dan melakukan upload file(s) yang akan digunakan.

Setelah melakukan pembuatan project, customer akan diarahkan ke halaman baru yang akan berisikan list file yang telah diupload dan 10 baris teratas data dari file tersebut. Pada halaman yang sama juga user dapat melakukan penguploadan file baru dan beberapa fitur terhadap file yang ada. Setelah proses ETL maka data yang telah diproses tadi akan diupload ke dalam database. Kemudian customer dapat melakukan pemilihan grafik yang diinginkan beserta data yang akan digunakan pada file tersebut. Jika data telah terupload ke database, maka customer tidak perlu mengulangi proses ETL

ketika akan membuat grafik lagi, melainkan customer langsung memilih grafik dan data yang akan digunakan. Arsitektur sistem ini dapat dilihat pada gambar 1.



Gambar 1. Arsitektur Sistem

Proses ETL akan berjalan dalam beberapa langkah. Langkah pertama adalah ekstraksi file, sistem akan membaca file yang telah diupload dan dipilih. Selanjutnya sistem akan melakukan langkah kedua yaitu transform. Pada langkah kedua ini, data akan diproses pembersihan dan penyesuaian data. Setelah kedua langkah tadi, maka data bisa dibuang siap untuk dimasukkan kedalam database. Langkah terakhir adalah proses penguploadan kedalam database.

B. ETL Proses

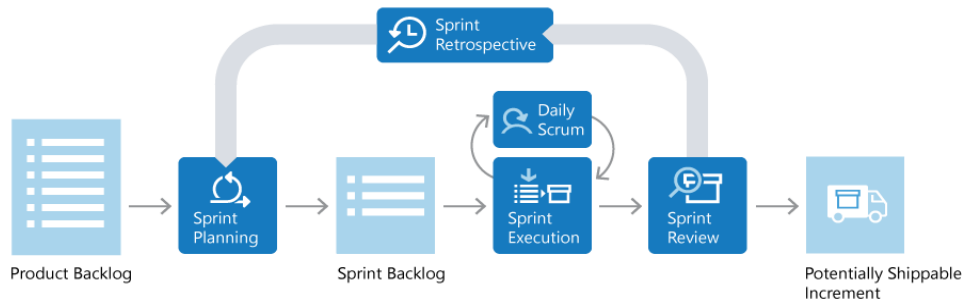
Extract Transform and Load (ETL) adalah sekumpulan proses yang harus dilalui dalam pembentukan data warehouse. Dengan tujuan mengumpulkan, menyaring, mengolah dan menggabungkan data-data yang relevan dari berbagai sumber untuk disimpan ke dalam data warehouse.

- 1) Extract
 

Pada langkah ini, data diekstraksi dari sistem sumber ke area pementasan. Transformasi jika ada dilakukan di staging area sehingga kinerja sistem sumber tidak terdegradasi. Juga, jika data yang rusak disalin langsung dari sumber ke dalam database gudang data, rollback akan menjadi tantangan. Staging area memberi peluang untuk memvalidasi data yang diekstraksi sebelum pindah ke gudang Data.
- 2) Transform
 

Pada langkah ini, akan menerapkan serangkaian fungsi pada data yang diekstraksi. Data yang tidak memerlukan transformasi apa pun disebut sebagai pemindahan langsung atau melalui data. Pada langkah transformasi, proses dapat melakukan operasi khusus pada data. Misalnya, jika pengguna menginginkan jumlah total penjualan yang tidak ada dalam basis data. Atau jika nama depan dan nama belakang dalam sebuah tabel ada di kolom yang berbeda. Dimungkinkan untuk menggabungkannya sebelum memuat.
- 3) Load
 

Memuat data ke dalam basis data house data target adalah langkah terakhir dari proses ETL. Dalam data warehouse yang khas, volume data yang besar perlu dimuat dalam periode yang relatif singkat. Karenanya, proses pemuatan harus dioptimalkan untuk kinerja. Sehingga dalam hal kegagalan pemuatan, mekanisme pemulihan harus dikonfigurasi untuk memulai kembali dari titik kegagalan tanpa kehilangan integritas



Gambar 2. Scrum Lifecycle

data. Admin database perlu memantau, melanjutkan, membatalkan beban sesuai kinerja server yang berlaku.

C. *Fitur*

Fitur – fitur yang disediakan oleh website adalah user dapat masuk atau mendaftarkan diri dengan menggunakan emailnya sendiri. Selain itu, user juga dapat masuk dengan akun Google miliknya untuk kemudahan dan kepraktisan pengguna. Website ini juga sudah mendukung untuk big data, sehingga user tidak perlu kebingungan untuk masalah jumlah data yang besar.

Website juga memfasilitasi penguploadan file dalam jumlah yang banyak dalam sekali upload. Setelah proses upload, user akan diminta untuk memilih proses mengeskrak, merubah, dan memuat (ETL) data agar data siap untuk di visualisasikan. Grafik yang dapat dibuat oleh sistem dalam bentuk 2D dengan opsi – opsi chart tertentu sesuai keinginan user itu sendiri.

D. *Desain dan Garis Besar Scrum*

Scrum [12] adalah kerangka kerja proses yang telah digunakan untuk mengelola pengembangan produk kompleks sejak awal tahun 1990-an. Scrum bukanlah sebuah proses, ataupun teknik. Akan tetapi Scrum adalah sebuah kerangka kerja dimana dapat menggunakan bermacam proses dan teknik di dalamnya. Scrum mengeksps ketidak efektifan dari manajemen produk dan teknik kerja yang ada, sehingga dapat secara terus menerus meningkatkan kinerja produk, tim, dan lingkungan kerja. Lifecycle scrum dapat dilihat pada gambar 2.

Kerangka kerja Scrum terdiri dari Scrum team dan peran-peran, acara-acara, artefak-artefak, dan aturan-aturan terkait. Setiap komponen di dalam kerangka kerja ini memiliki tujuan tertentu dan sangat penting bagi keberhasilan penggunaan Scrum. Aturan Scrum mengikat peran-peran, acara-acara, artefak-artefak, serta menjaga hubungan dan interaksi antar komponen tersebut Ujicoba

Uji coba terhadap website dilakukan secara bertahap mengikuti jumlah sprint yang telah ditetapkan. Untuk website proses pembuatan website ini dibutuhkan 4 buah sprint. Yang masing – masing sprint memiliki waktu pengerjaan 14 hari. Dari 1 sprint dibagi menjadi bagian yang terdiri dari Sprint Planning, Daily Scrum, Sprint Test, Sprint Deploy, dan Sprint Review.

Sprint 1 merupakan permulaan pengerjaan website ini. Pada sprint ini, tim memulai untuk pembelajaran, dan pemahaman terhadap bahasa permrograman, database, dan tools yang akan digunakan untuk membantu proses pengerjaan.

Sprint 2 merupakan permulaan untuk membangun website penelitian ini. Pada sprint ini, pekerjaan tim adalah memulai pembuatan prototype halaman – halaman HTML dan penggabungan modul yang telah dipelajari pada sprint 1.

Sprint 3 merupakan peralihan halaman – halaman HTML dari prototype menjadi menggunakan templates. Dan penambahan beberapa halaman baru.

Sprint 4 merupakan sprint terakhir dari proses pengerjaan website ini, pada sprint ini dapat disebut sebagai finishing dari website ini. Ringkasan uji coba yang dilakukan dari sprint 1 hingga sprint 4 dapat dilihat pada tabel 2.

Tabel 2  
Ringkasan uji coba yang dilakukan dari sprint 1 hingga sprint 4

| Skenario Pengujian                  | Yang Diharapkan   | Kesimpulan  |
|-------------------------------------|---|---|
| Navigasi halaman                    | Sistem dapat melakukan navigasi halaman                                       | Berhasil  |
| Pengubahan format penulisan tanggal | Sistem dapat mengubah berbagai format tanggal menjadi ‘YYYY-MM-DD’            | 3 format tanggal Berhasil<br>3 format tanggal Gagal |
| Pengiriman data dalam bentuk JSON   | Sistem dapat mengirimkan data dalam bentuk JSON dari Python menuju JavaScript | Berhasil  |
| Pembentukan Grafik                  | Sistem dapat membuat dan menampilkan chart tersebut                           | Berhasil  |

IV. KESIMPULAN DAN SARAN

Berdasarkan hasil tersebut dapat disimpulkan beberapa poin yang akan disebutkan dibawah ini.

- 1) Penggunaan metode scrum untuk membuat web untuk visualisasi data berjalan dengan efektif karena dalam proses pembuatan web ini lebih terstruktur sehingga time management menjadi lebih efisien.
- 2) Library DevExpress lebih mudah digunakan daripada menggunakan library Google Chart.
- 3) Dalam menampilkan chart, library DevExpress hanya dapat menampilkan sebuah chart dalam satu halaman.

Berdasarkan hasil tersebut mendapatkan saran beberapa poin yang akan disebutkan dibawah ini.

- 1) Library visualisasi Google Chart lebih baik digantikan dengan library lainnya yang dapat melakukan proses zoom in dan zoom out lebih baik.
- 2) Library ETL yang dapat melakukan prosesnya secara

otomatis dan memiliki kemampuan mempelajari pola – pola user.

#### DAFTAR PUSTAKA

- [1] J. D. Miller, *Big data visualization*. Packt Publishing Ltd, 2017.
- [2] S. Chaudhuri and U. Dayal, “An overview of data warehousing and OLAP technology,” *ACM Sigmod Rec.*, vol. 26, no. 1, pp. 65–74, 1997.
- [3] M. Grinberg, *Flask web development: developing web applications with python*. “ O’Reilly Media, Inc.,” 2018.
- [4] J. Bertin, “Semiology of graphics: diagrams, networks, maps.,” 1984.
- [5] W. McKinney and others, “pandas: a foundational Python library for data analysis and statistics,” *Python high Perform. Sci. Comput.*, vol. 14, no. 9, pp. 1–9, 2011.
- [6] A. Cairo, *The truthful art: Data, charts, and maps for communication*. New Riders, 2016.
- [7] M. E. Mann, *The Hockey Stick and the Climate Wars*. 2012.
- [8] S. K. Card, J. Mackinlay, and B. Shneiderman, “Reading in information visualization: using vision to Think (Chapter 1),” *Morgan Kaufmann*, vol. 6, no. 4, 1999.
- [9] N. Yau, *Visualize this: the FlowingData guide to design, visualization, and statistics*. John Wiley & Sons, 2011.
- [10] A. Ferrari and M. Russo, *Introducing Microsoft Power BI*. Microsoft Press, 2016.
- [11] S. Batt, T. Grealis, O. Harmon, and P. Tomolonis, “Learning Tableau: A data visualization tool,” *J. Econ. Educ.*, vol. 51, no. 3–4, pp. 317–328, 2020.
- [12] L. Gonçalves, “Scrum,” *Control. & Manag. Rev.*, vol. 62, no. 4, pp. 40–42, 2018.

# INSYST

Journal of Intelligent System and Computation

Volume 02 Nomor 02 Oktober 2020

---

## Author Guidelines

- Manuscript should be written in Indonesia and be submitted online via journal website. Online Submission will be charged at no Cost
- Manuscript should not exceed 15 pages including embedded figures and tables, without any appendix, and the file should be in Microsoft Office (.doc/.docx). [download template](#)
- Title, Abstract and Keywords must be written in bilingual
- Title should be less than 15 words
- Abstracts consists of no more than 200 words, contains the essence of the article and includes a brief background, objectives, methods and results or findings of the study. Abstract is written in one paragraph.
- Keywords are written in Indonesia and English three to five words/phrases, separated with coma and consist of important words/phrases from the article.
- Author's name, affiliation, affiliation address and email. State clearly and include country's name on your affiliation address.
- The main text of the writing should be consists of: Introduction, Method, Result and Discussion, and Conclusion; followed by Acknowledgment and Reference
- Introduction State adequate background, issues and objectives, avoiding a detailed literature survey or a summary of the results. Explain how you addressed the problem and clearly state the aims of your study.
- Used method is the scientific in the form of study of literature, observation, surveys, interviews, Focus Group Discussion, system testing or simulation and other techniques commonly used in the world of research. It is also recommended to describe analysis techniques used briefly and clearly, so that the reader can easily understand.
- Results should be clear, concise and not in the form of raw data. Discussion should explore the significance of the results of the work, not repeat them. Avoid extensive citations and discussion of published literature. INSYST will do the final formatting of your paper.
- Conclusion should lead the reader to important matter of the paper. Authors are allowed to include suggestion or recommendation in this section. Write conclusion, suggestion and/or recommendation in narrative form (avoid of using bulleting and numbering)
- Acknowledgments. It is highly recommended to acknowledge a person and/or organizations helping author(s) in many ways. Sponsor and financial support acknowledgments should be included in this section. Should you have lots of parties



to be acknowledged, state your acknowledgments only in one paragraph. Avoid of using bulleting and numbering in this section

- The number of references are not less than 10 with at least 8 primary references. Primary references are include journal, thesis, disertasion and all kinds of research reports. All refferences must come from source published in last 7 years.
- Figure and table should be in black and white, and if it is made in color, it should be readable when it is later printed in black and white.
- Figure and table should be clearly readable and in a proportional measure to the overall page.

### **Tim Redaksi**

Journal of Intelligent System and Computation

Departement of Informatics

Institut Sains dan Teknologi Terpadu Surabaya

Jl. Ngagel Jaya Tengah 73-77 Surabaya

Email: [insyst@istts.ac.id](mailto:insyst@istts.ac.id)

Website: <https://jurnal.stts.edu/index.php/INSYST/index>