

Implementasi Metode *Recurrent Neural Network* pada Pengklasifikasian Kualitas Telur Puyuh

Exel Defrisco Tarkus, Sherwin R.U.A. Sompie, Agustinus Jacobus
Teknik Elektro Universitas Sam Ratulangi Manado, Jl. Kampus Bahu, 95115, Indonesia
email : exeldtarkus@gmail.com, aldo@unsrat.ac.id, a.jacobus@unsrat.ac.id
Diterima: 24 Juni 2020 ; direvisi: 26 Juni 2020 ; disetujui: 30 Juni 2020

Abstrak — Telur merupakan salah satu sumber makanan yang banyak mengandung protein sekaligus disukai oleh banyak orang dan juga mempunyai berbagai macam manfaat bagi manusia. *Recurrent Neural Network* (RNN) adalah jaringan saraf berulang atau jaringan saraf tiruan yang pemrosesannya di panggil secara berulang-ulang untuk memproses masukan yang biasa adalah data sekuensial. Setelah mengambil suara ketukan, pemrosesan akan memasuki tahap ekstraksi fitur, *training data* dan prediksi data. Hasil dari proses ekstraksi fitur berupa nilai statistika, *Fast Fourier Transform* (FFT) dan *Mel-Frequency Cepstral Coefficients* (MFCC) yang kemudian akan di *training* dengan metode RNN dan menghasilkan file *model.h5*. Untuk hasil prediksi *precision* diatas 75%, *recall* diatas 81% dan akurasi diatas 87%.

Kata Kunci — *Fast Fourier Transform*; Kualitas Telur; *Mel-Frequency Cepstral Coefficients*; *Recurrent Neural Network*.

Abstract — Eggs are one of the food sources that contain lots of protein and at the same time favored by many people and also have a variety of benefits for humans. *Recurrent Neural Network* (RNN) is a recurrent neural network or artificial neural network that the processing is called repeatedly to process input which is usually sequential data. After taking the sound of quail, processing will enter the feature extraction, training data and data prediction stages. The results of the feature extraction process are statistical values, *Fast Fourier Transform* (FFT) and *Mel-Frequency Cepstral Coefficients* (MFCC) which will then be trained with the RNN method and produce a *model.h5* file. For precision prediction results above 75%, recall above 81% and accuracy above 87%.

Keywords — *Fast Fourier Transform*; Egg Quality; *Mel-Frequency Cepstral Coefficients*; *Recurrent Neural Network*.

I. PENDAHULUAN

Telur puyuh merupakan salah satu sumber protein hewani yang lebih baik dibanding dengan kandungan protein telur ayam dan telur itik, juga relatif murah dibandingkan dengan telur ayam, daging sapi, dan lain-lain. Dilihat dari nilai gizinya, telur puyuh mengandung 13,6% protein dan 8,2% lemak. [1]Struktur dari telur puyuh secara umum tidak berbeda dengan struktur telur ayam yang terdiri atas tiga bagian, yaitu putih telur (58%), kuning telur (31%), dan kerabang telur (11%).

Kekuatan dan ketebalan kulit telur puyuh menjadi pelindung kualitas telur tersebut terhadap serangan-serangan bakteri maupun hal yang mengganggu kualitas telur puyuh itu sendiri. Dan terkadang masih terdapat telur-telur yang retak yang juga masih di jual dipasaran sehingga dapat merugikan konsumen dan juga produsen penyedia telur, maka dari itu

teknik klasifikasi yang baik dapat menjawab hal tersebut.

Teknik dan algoritma dalam Pembelajaran dalam dapat digunakan baik untuk kebutuhan pembelajaran terarah, pembelajaran tak terarah dan semi-terarah (*semi-supervised learning*) dalam berbagai aplikasi seperti pengenalan citra, pengenalan suara, klasifikasi teks, dan sebagainya. Teknik klasifikasi telur dengan menggunakan teknologi *deep learning* yang semakin berkembang menjadi pilihan yang tepat dan cepat dalam melakukan klasifikasi pada telur, salah satu teknik penyortiran yang ditawarkan adalah dengan menggunakan teknik pengolahan suara sehingga dapat mengklasifikasikan telur yang baik untuk konsumen.

Tujuan penelitian ini adalah untuk mengklasifikasikan telur puyuh dengan pengolahan suara berdasarkan metode *Recurrent Neural Network* (RNN).

A. Penelitian Terkait

Terdapat beberapa penelitian sebelumnya yang berhubungan dengan implementasi metode RNN pada penentuan kualitas telur dan dijabarkan sebagai berikut:

- 1) Klasifikasi Telur Ayam Berbasis Pengolahan Suara, oleh Devi Sahati. [2]Penelitian yang menggunakan metode *Support Vector Machine*. Dengan pengolahan suara, telur yang diteliti akan diambil sampel suaranya dengan cara diketuk cangkangnya. Setelah sampel didapat, pengolahan suara akan dilakukan, yang terdiri dari pengekstrasian ciri telur yang bagus maupun ciri telur yang buruk, kemudian akan dilakukan pengklasifikasian antara telur yang bagus dan telur yang buruk, dengan hasil yang didapat yaitu akurasi dari klasifikasi pada penelitian ini adalah 95% pada pengujian dengan menggunakan data latih, dan 89.47% pada pengujian dengan menggunakan data baru.
- 2) Prediksi Kunjungan Wisatawan Dengan *Recurrent Neural Network Extended Kalman Filter* oleh Rizal, Ahmad Ashril. [3]Untuk prediksi kunjungan wisatawan penelitian ini memiliki beberapa tahap diantaranya *pre-processing*, *training* RNN, dan proses *testing* dengan hasil *training* RNN dengan *Extended Kalman Filter* untuk prediksi memiliki tingkat akurasi terbaik sebesar 64.37% dan hasil *testing* RNN-EKF pada kasus prediksi memberikan hasil akurasi terbaik sebesar 62.91%.
- 3) Aplikasi *Model Recurrent Neural Network dan Recurrent Neuro Fuzzy* Untuk Peramalan Banyaknya Penumpang Kereta Api Jabotabek oleh Hermawan. [4]Penelitian ini menggunakan dua metode untuk proses peramalan data, peramalan dilakukan dengan memilih input, memilih struktur model yang terbaik setelah itu memilih

menyederhanakan koefisien dan melakukan kesesuaian model dan didapatkan hasil MAPE *training* 1,2617% dan MSE *training* 29500000. Serta MAPE *testing* 3,785% dan MSE *testing* 2610000000.

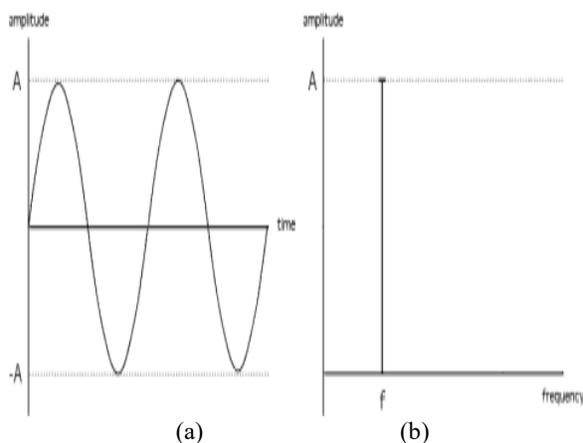
- 4) Identifikasi Judul Berita *Clickbait* Berbahasa Indonesia Dengan Algoritma *Long Short Term Memory* (LSTM) *Recurrent Neural Network* oleh Ibnu. [5] Penelitian ini akan mengidentifikasi berita dengan LSTM-RNN dengan tahap *input* data, normalisasi *text*, proses *training*, dan pengujian dengan hasil pelatihan dan validasi model dengan menambahkan *Dropout Layer* dan *Callback Function* mampu menghindari model dari *overfitting* dengan menghasilkan akurasi 82% dan loss 38%.
- 5) Dalam Publikasi oleh Suyudi, Esmeralda, Muspupah. [6] Prediksi Harga Saham Menggunakan Metode *Recurrent Neural Network*. Sistem prediksi harga saham terdiri dari tiga tahap. Tahap pertama yaitu pra-proses data yang terdiri dari ekstraksi fitur, normalisasi, *ascending* dan segmentasi. Tahap kedua yaitu proses pelatihan data menggunakan RNN, kemudian tahap ketiga yaitu proses pengujian. Hasil dari penelitian ini menunjukkan bahwa prediksi harga saham menggunakan RNN mendapatkan hasil kurang baik. Menggunakan optimasi Adam dengan nilai *learning rate* 0.001 dan *epoch* 200 memperoleh akurasi untuk data latih sebesar 94.16% dan data uji sebesar 55.26%.

B. Gelombang Suara

[7] Gelombang suara merupakan gelombang analog yang apabila diolah menggunakan peralatan elektronik, gelombang tersebut harus melalui tahap digitalisasi sehingga gelombang tersebut berupa data *digital*.

[8] Pendengaran manusia terbatas pada daerah frekuensi 16 Hz sampai 20 kHz. Namun, pendengaran akan lebih sensitif pada suara yang frekuensinya antara 1 kHz dan 4 kHz. Suara di atas 20 kHz tidak akan bisa didengar, sedangkan di bawah 16 Hz sinyal suara akan terdengar lambat dan berderak-derak. Antara 0 dB sampai 130 dB adalah level suara yang aman didengar. Di atas level 130 dB, suara tersebut akan menyakitkan telinga, dan dapat pula mengakibatkan selaput telinga pecah.

Sinyal bisa dilihat dari dua sudut pandang, yaitu *time domain* dan *frequency domain*. Gambar 1 (a), *time-domain*



Gambar 1. Sinyal dalam *Time-domain* (a), dan *Frequency-domain* (b)

menunjukkan bagaimana bentuk sinyal dalam rentang waktu tertentu. Dan untuk Gambar 1 (b), *frequency-domain* menunjukkan seberapa besar sinyal yang ada dalam setiap *frequency band* dalam rentang beberapa frekuensi.

C. Fast Fourier Transform

FFT adalah suatu algoritma untuk menghitung transformasi *fourier* diskrit dengan cepat dan efisien. Karena banyak sinyal-sinyal dalam sistem komunikasi yang bersifat kontinu, sehingga untuk kasus sinyal kontinu kita gunakan transformasi *fourier*.

Algoritma FFT berdasarkan atas prinsip pokok dekomposisi perhitungan *discrete fourier transform* dari suatu sekuen sepanjang N kedalam transformasi diskrit *fourier* secara berturut-turut lebih kecil. Cara prinsip ini diterapkan memimpin ke arah suatu variasi dari algoritma yang berbeda, di mana semuanya memperbandingkan peningkatan kecepatan perhitungan.

D. Mel-Frequency Cepstrum Coefficient

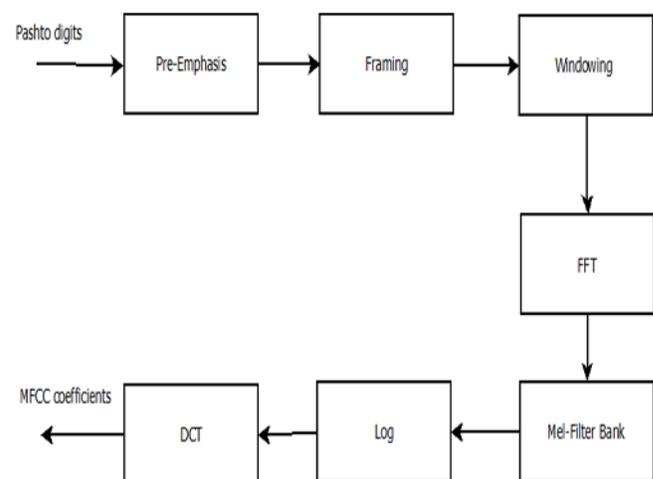
MFCC adalah salah satu metode ekstraksi fitur dalam bidang speech technology yang banyak digunakan untuk pengenalan suara. MFCC merupakan proses feature yang mengkonversi sinyal suara menjadi beberapa parameter, Keunggulan dari metode ini adalah:

- 1) Mampu untuk menangkap karakteristik suara yang sangat penting bagi pengenalan suara. Atau dengan kata lain mampu menangkap informasi-informasi penting yang terkandung dalam sinyal suara..
- 2) Menghasilkan data seminimal mungkin, tanpa menghilangkan informasi-informasi penting yang ada.
- 3) Dapat mengadaptasi organ pendengaran manusia dalam melakukan persepsi terhadap sinyal suara.

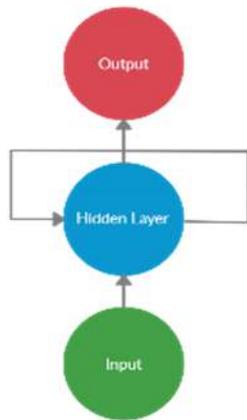
Perhitungan yang dilakukan dalam MFCC menggunakan dasar perhitungan *short-term analysis*. Hal ini dilakukan mengingat sinyal suara bersifat *stationary*. [9] MFCC *features extraction* sebenarnya merupakan adaptasi dari sistem pendengaran manusia, dimana sinyal suara akan di *filter* secara linear dengan frekuensi rendah (kurang dari 1000Hz) dan secara logarimik untuk frekuensi yang tinggi (diatas 1000Hz).

E. Recurrent Neural Network

RNN termasuk dalam jenis jaringan pemodelan *Neural Network*. Selain RNN, *Multilayer Perceptron* (MPL) dan



Gambar 2. Blok Diagram MFCC



Gambar 3. Arsitektur RNN

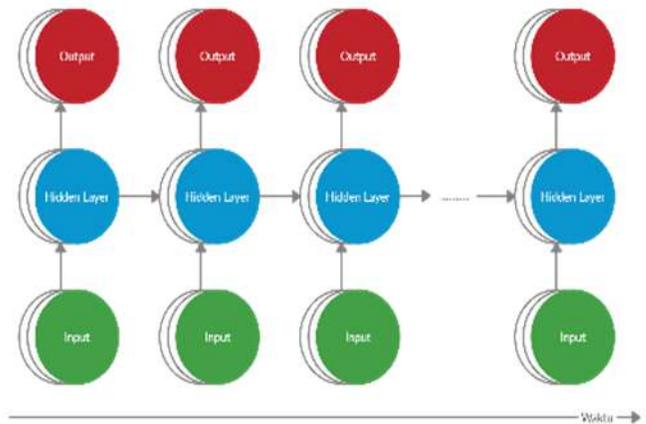
Feedforward Neural Network (FNN) termasuk dalam jenis pemodelan *Neural Network*. [10]FNN telah banyak diaplikasikan untuk melakukan prediksi atau klasifikasi data selama bertahun-tahun yang lalu. Salah satu tipe dari *neural network* yang dikembangkan dari dasar pemikiran FNN adalah RNN.

RNN yang juga disebut jaringan umpan balik adalah jenis jaringan pada *neural network* dimana terdapat *loop* sebagai koneksi umpan balik dalam jaringan. [11]Jaringan RNN adalah jaringan yang mengakomodasi *output* jaringan untuk menjadi *input* pada jaringan tersebut yang kemudian digunakan untuk menghasilkan *output* yang baru.

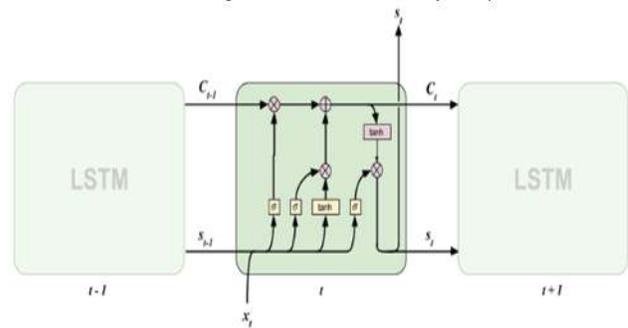
Pembuatan RNN sebenarnya dibuat untuk data-data yang bersifat *sequential* atau bertahap. Penggunaan *neural network* [12]biasanya semua *input* dan *output* tidak bergantung satu sama lain, maka akan terjadi penumpukan tugas pada *neural network* sangat banyak dan bertumpuk, sedangkan jika ingin membangun sebuah aplikasi yang mengelola data ilmiah dengan data sampel berbentuk *time series* diperlukan algoritma yang dapat *handle* data tersebut dengan cepat dan tepat, maka RNN dapat melakukan hal itu karena RNN melakukan tugas yang sama pada setiap elemen data di sebuah urutan, lalu memproses *outputnya* yang mengacu pada hasil komputasi sebelumnya. Maka secara teori RNN mampu menggunakan informasi yang telah direkam sebelumnya yang panjang urutannya beragam. [13]Jaringan *node* pada RNN dimasukan ke dalam *layer* yang berurutan. Setiap *node* dalam lapisan tertentu terhubung dengan koneksi terarah ke setiap *node* lain dilapisan *layer* berikutnya secara beturut-turut, setiap *node* juga memiliki aktivasi yang bernilai nyata yang bervariasi tergantung waktu.

Pada Gambar 3 merupakan diagram arsitektur pada pemrosesan RNN, terlihat bahwa lingkaran hijau merupakan *input layer* atau masuknya data yang akan dilakukan komputasi pada RNN, setelah itu masuk ke lingkaran biru yang merupakan *hidden layer* terlihat ada garis *looping* yang kembali pada lingkaran biru tersebut, pola *looping* ini memungkinkan RNN untuk menyimpan ingatan sementara yang nantinya akan dipakai dalam pemrosesan data berikutnya, dan juga yang terakhir ada *output layer* yang nantinya adalah hasil dari proses komputasi RNN.

Arsitektur RNN menunjukkan bagaimana RNN yang bekerja dengan banyak layer dengan perulangan *hidden state*



Gambar 4. Diagram Arsitektur RNN Banyak Layer



Gambar 5. Diagram Arsitektur LSTM

sebanyak data waktu yang disediakan $x_t = input$, $s_t = hidden layer$, $O_t = output$. Maka dalam *hidden layer* dapat menyimpan memori dari kalkulasi dengan rumus (1)

Dimana fungsi f biasanya adalah non-linearitas *tanh* atau *ReLU*, s_{t-1} digunakan untuk menghitung nilai *hidden layer* yang pertama, biasanya pada inialisasi selalu diawali dari 0 (no), sedangkan u adalah data vektor *input* pada RNN dan w adalah data vektor *output* pada RNN.

F. Long Short Term Memory

LSTM merupakan salah satu arsitektur RNN. LSTM diciptakan oleh Hochreiter & Schmidhuber dan kemudian dikembangkan dan dipopulerkan oleh banyak periset. Seperti RNN, jaringan LSTM (*LSTM network*) juga terdiri dari modul-modul dengan pemrosesan berulang. Ringkasnya, [14]LSTM menambahkan sebuah proses seleksi di dalam kotak kontrol (*cell*) sehingga bisa menyeleksi informasi mana yang layak untuk diteruskan, sekaligus menjadi solusi bagi permasalahan *vanishing gradient*.

Gambar arsitektur dari RNN menjadi LSTM maka *hidden layer* pada RNN akan berbentuk seperti *cell* yang berfungsi untuk mengingat peristiwa masa lalu. Dengan memperhatikan lebih detail apa yang terjadi di dalam *hidden layer*, maka perlu membedah isi dari perceptron X_t . Juga terdapat notasi baru I_t (dibaca *input* saat waktu t) yang merupakan input untuk perceptron X_t . Fungsi aktivasi di dalamnya yang digunakan bisa *sigmoid* atau *hyperbolic (tanh)*. [15]LSTM dibagi menjadi beberapa gerbang diantaranya *Forget Gate*, *Input Gate*, *Cell Gate* dan *Output Gate* yang mempunyai tugasnya masing-masing.

II. METODE PENELITIAN

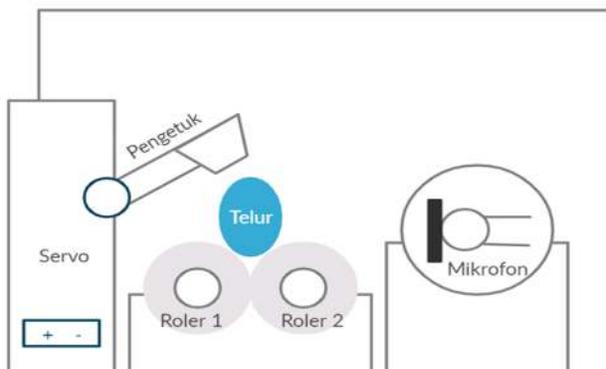
Tahapan penelitian yang dilalui dimulai dari desain dan pembuatan alat perekam bunyi ketukan, pengumpulan data suara, pengolahan data suara, pengujian dan evaluasi. Tahapan tersebut diuraikan lebih rinci sebagai berikut.

A. Perancangan Alat

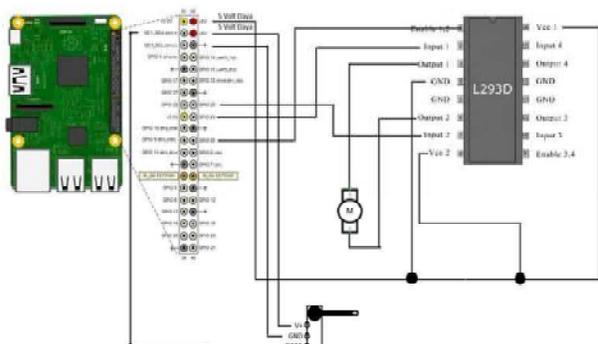
Tahap ini merupakan tahap perancangan sistem bagaimana cara kerja agar alat dapat merekam telur pada proses klasifikasi telur puyuh.

Gambar 6 merupakan gambar alat tampak depan dimana dalam alat tersebut terdapat servo yang sudah dipasangi pengetuk untuk mengetuk telur puyuh. Roler 1 dan roler 2 adalah penampang tempat. Terdapat juga mikrofon kondensator yang akan merekam ketukan telur pada saat proses perekaman telur. Dan juga terdapat rangkaian alat dimana alat akan dikoneksikan dengan *raspberrypi*.

Pada Gambar 7 terlihat pada *raspberrypi* untuk menggerakkan servo digunakan pin 3 (GPIO 2) yang mentransfer data ke servo, pin 2 (power 5v) sebagai daya 5 volt pada servo dan pin 6 (GND) sebagai *ground* pada servo. Sedangkan untuk menggerakkan motor diperlukan IC (*Intergrate Circuit*) dengan menggunakan pin 4 (power 5v) yang dihubungkan dengan pin 14 pada IC sebagai sumber daya pada motor, pin 16 (GPIO23) yang terhubung dengan pin pada IC sebagai *input* negatif yang juga terhubung dengan kabel negatif motor DC, pin 18 (GPIO 24) yang terhubung dengan pin 2 IC sebagai *input* positif dan juga terhubung dengan kabel positif motor DC, pin 22 (GPIO 25) yang terhubung dengan pin 1 IC sebagai *input enable* 1,2 dan pin 14 (GND) yang terhubung dengan pin 4 IC sebagai *ground* data.



Gambar 6. Desain Alat



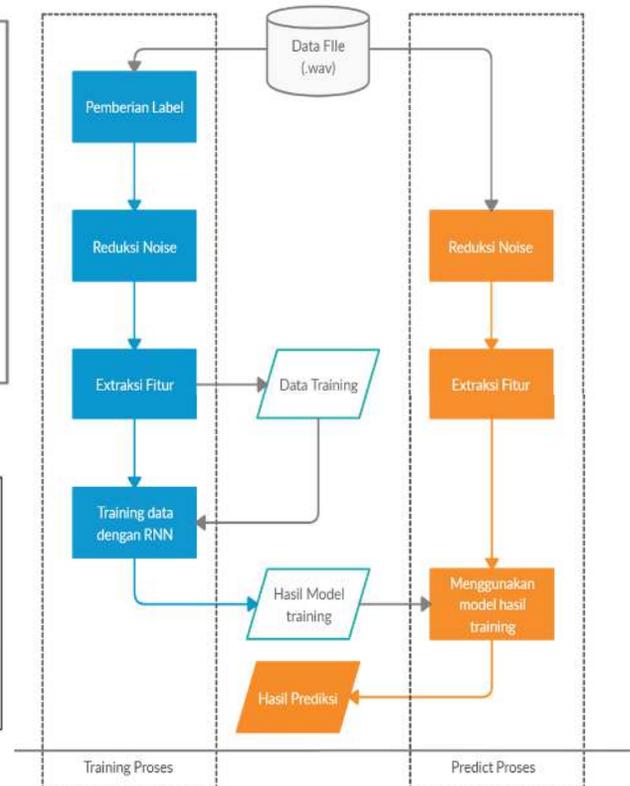
Gambar 7. Rangkaian Alat

B. Pengumpulan Data

Pada tahap pengumpulan data merupakan tahap pengambilan dataset yang kemudian akan di *training* dengan menggunakan algoritma RNN. Dimana telur yang diambil terdiri dari sampel telur dengan kualitas cangkang baik dan sampel telur dengan kualitas cangkang buruk, proses pengambilan sampel telur dilakukan dengan merekam suara hasil ketukan telur dan disimpan dengan format (.wav).

C. Prinsip Kerja Sistem

Prinsip kerja sistem memiliki dua proses alur data yang berbeda dimana proses dengan warna biru adalah proses dengan *training* dan warna *orange* adalah proses prediksi data dengan data yang baru. Proses *training* data terdapat satu proses yaitu data *training* yang merupakan proses untuk membuat data-data untuk pemrosesan model, dengan menginput kumpulan data *audio* yang belum diolah dan belum terstruktur. Kemudian memberikan label pada data yang dikumpulkan tersebut, selanjutnya data tersebut akan mengalami reduksi *noise* yang dapat mengganggu tingkat akurasi. Setelah menghilangkan *noise* data tersebut akan diekstraksi fitur-fiturnya dengan menggunakan FFT dan MFCC sehingga fitur-fitur tersebut akan di *training* dengan menggunakan metode RNN dan menghasilkan model yang akan disimpan untuk proses prediksi. Berbeda dengan proses *training* data. Proses prediksi dimulai dengan menginput data suara telur yang baru yang kemudian akan di hilangkan *noise*-nya, kemudian fitur dari suara baru tersebut akan di ekstraksi dan akan dibandingkan dengan model yang didapat pada proses *training* data. Setelah perbandingan model selesai maka akan didapatkan *output* berupa klasifikasi telur “good” atau “bad”.



Gambar 8. Prinsip Kerja Sistem

D. Pengujian dan Evaluasi

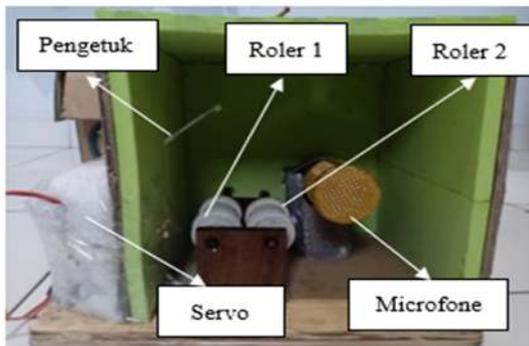
Pengujian dan evaluasi sistem data akan dihitung dengan menggunakan *Confusion Matrix*. Pada *confusion matrix* terdapat dua pengukuran rasio pada setiap kelas yaitu *positive rate* adalah nilai yang diprediksi sebagai label *good* pada kedua label sedangkan *negative rate* adalah nilai yang diprediksi sebagai label *bad* di kedua label, yang kemudian dibagi lagi menjadi empat bagian nilai diantaranya adalah *True Positive Rate* (TPR), *True Negative Rate* (TNR), *False Positive Rate* (FPR) dan *False Negative Rate* (FNR).

Tabel *confusion matrix* berdasarkan *positive rate* dan *negative rate* dari hasil prediksi data. TPR merupakan data label *good* yang terdeteksi benar dengan perhitungan membagi semua nilai hasil prediksi label *good* yang benar dengan semua label *good*. Hasil akurasi (rumus (1)) diperoleh dari pembagian jumlah prediksi yang tepat dibagi dengan jumlah seluruh data.

$$\text{akurasi} = \frac{\text{jumlah prediksi yang tepat}}{\text{jumlah seluruh data}} \quad (1)$$

Hasil *precision* (rumus (2)) dari *confusion matrix* dapat menggunakan perhitungan persentase. Hasil *precision* diperoleh dari perbandingan rasio prediksi tepat positif dengan keseluruhan hasil positif.

$$\text{precision}_{\text{label}} = \frac{\text{jumlah data yang diprediksi dengan tepat}}{\text{jumlah data per label}} \quad (2)$$



Gambar 9. Alat Perekam Telur

localhost:5000



Gambar 10. Tampilan Antarmuka *Index.html*

Hasil *recall* dari *confusion matrix* dapat menggunakan perhitungan persentase. Hasil *recall* (rumus (3)) diperoleh dari pembagian jumlah data yang benar *true positive* dibagi dengan jumlah data yang dipisahkan.

$$\text{recall}_{\text{label}} = \frac{\text{jumlah data yang diprediksi dengan tepat}}{\text{jumlah data prediksi per label}} \quad (3)$$

III. HASIL DAN PEMBAHASAN

A. Implementasi Perancangan Alat

Alat yang digunakan berbentuk kubus dengan alas memiliki luas sekitar 20 cm x 20 cm dimana untuk bagian dalam terdapat mikrofon untuk mengambil suara telur puyuh dan penampang untuk meletakkan telur puyuh.

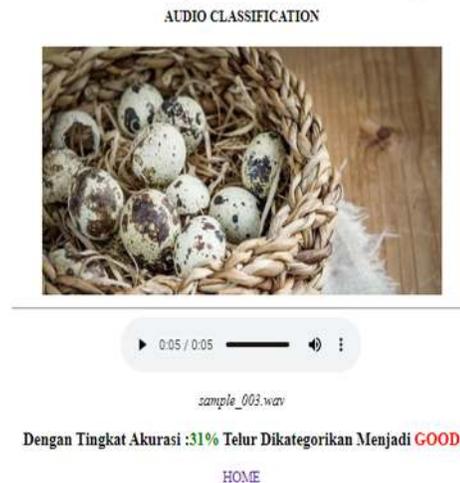
Pada Gambar 9 merupakan gambar tampak depan pada alat dimana terdapat pengetuk yang berfungsi untuk mengetuk telur, roler 1 dan roler 2 yang berfungsi untuk memutar telur dalam proses perekaman suara, selain itu terdapat servo yang akan menggerakkan pegetuk dalam proses perekaman suara dan terakhir terdapat *mikrofone* yang akan menangkap suara hasil ketukan telur.

B. Implementasi Rancangan Antarmuka

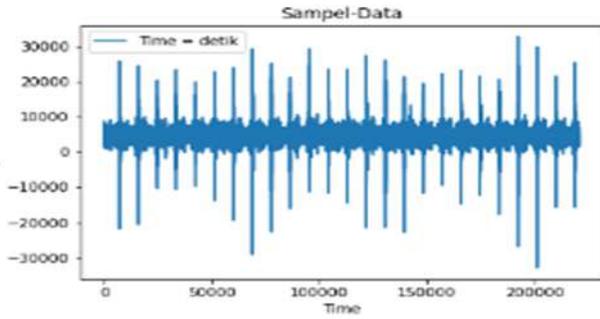
Pada tahap ini adalah tahap implementasi rancangan aplikasi yang digunakan untuk prediksi data dalam penelitian ini aplikasi ini berbasis website dan memiliki *python* sebagai pengelolah data atau *backend* dan HTML sebagai *frontend*.



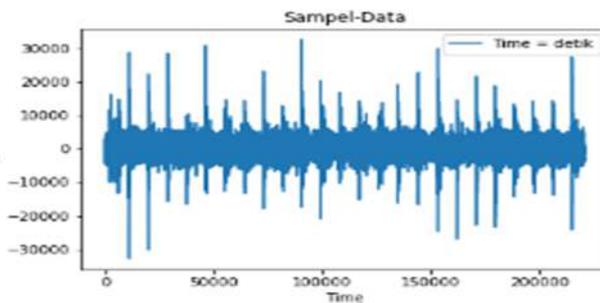
Gambar 11. Tampilan Antarmuka *Record Page*



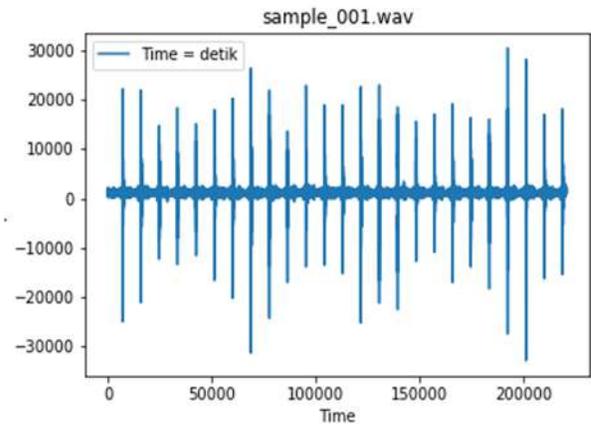
Gambar 12. Tampilan Antarmuka *Predict*



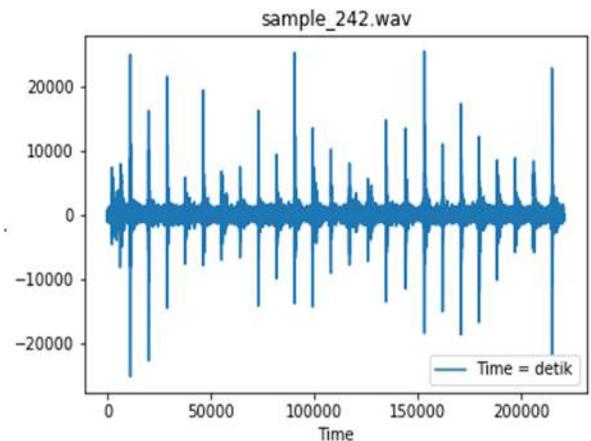
Gambar 13 Sampel Telur Label *Good*



Gambar 14 Sampel Telur Label *Bad*



Gambar 15 Reduksi Telur *Good*



Gambar 16 Reduksi Telur *Bad*

TABEL I
EXTRAKSI CIRI AUDIO

<i>Fname</i>	<i>OriMean</i>	<i>OriMedian</i>	<i>OriMax</i>	<i>OriMin</i>
<i>sample_001.wav</i>	0.13962	0.13907	0.89371	-0.58851
<i>sample_002.wav</i>	0.02101	0.02065	0.94055	-0.90486
<i>sample_003.wav</i>	0.03889	0.03870	0.86517	-0.77941
<i>sample_004.wav</i>	0.08123	0.08143	0.84413	-1.02902
<i>sample_005.wav</i>	-0.00181	-0.0017	0.86699	-0.97957
<i>sample_006.wav</i>	0.06677	0.06694	0.89413	-0.86549
<i>sample_007.wav</i>	0.07822	0.07849	0.89931	-0.89039
<i>sample_008.wav</i>	-0.07026	-0.0698	0.96958	-0.98938
<i>sample_009.wav</i>	-0.18639	-0.1845	0.93671	-0.98665
<i>sample_010.wav</i>	-0.58815	-0.5874	0.29296	-0.98454
<i>sample_011.wav</i>	0.08681	0.08671	1.00297	-0.98606
<i>sample_012.wav</i>	0.0445	0.0449	0.84143	-0.75824
<i>sample_010.wav</i>	-0.58815	-0.5874	0.29296	-0.98454
<i>sample_011.wav</i>	0.08681	0.08671	1.00297	-0.98606
<i>sample_005.wav</i>	0.0445	0.0449	0.84143	-0.75824
<i>sample_220.wav</i>	0.01037	0.01051	0.89080	-0.87143
<i>sample_221.wav</i>	-0.07911	-0.0789	1.01114	-0.92840
<i>sample_222.wav</i>	-0.07026	-0.0698	0.96958	-0.98938
<i>sample_223.wav</i>	-0.18639	-0.1845	0.93671	-0.98665
<i>sample_224.wav</i>	-0.58815	-0.5874	0.29296	-0.98454
<i>sample_225.wav</i>	0.08681	0.08671	1.00297	-0.98606
<i>sample_226.wav</i>	0.0445	0.0449	0.84143	-0.75824
<i>sample_227.wav</i>	0.2214	0.2214	0.99532	-0.95094
<i>sample_228.wav</i>	0.01037	0.01051	0.89080	-0.87143

TABEL II
HASIL *CONFUSION MATRIX* UNTUK PENGUJIAN DATASET

<i>True Class</i>	<i>Predict Class</i>	
	<i>X</i>	<i>Good</i>
<i>Good</i>	27	8
<i>Bad</i>	2	33

TABEL III
HASIL KESELURUHAN

<i>Label</i>	<i>Positive rate</i>	<i>Negative Rate</i>	<i>Precision</i>	<i>Recall</i>	<i>Akurasi</i>
<i>Good</i>	78 %	22 %	78 %	80 %	85 %
<i>Bad</i>	95 %	5 %	95 %	93 %	

C. Implementasi Pelabelan Data

Pada implementasi pelabelan data adalah hasil proses pelabelan data yang dikategorikan menjadi dua label yaitu label *good* dan label *bad*. Pelabelan data mengkategorikan telur menjadi 2 kategori, pertama telur dengan kondisi cangkang masih utuh, tidak retak dan tidak terlihat cacat dikategorikan telur “*Good*” dan telur dengan kondisi cangkang rusak, pecah dan retak dikategorikan telur “*Bad*”.

D. Implementasi Reduksi Noise

Merupakan suatu proses untuk mengurangi noise. Dari data yang telah di dapatkan dalam proses pengumpulan dan diberi label maka normalisasi pada data diperlukan untuk menghasilkan data yang bersih sehingga meningkatkan akurasi pada saat training data.

E. Implementasi Ekstraksi Fitur Data

Dalam mengelola file suara sangat diperlukan proses ekstraksi fitur atau ekstraksi ciri dengan menggunakan FFT dan MFCC yang berguna untuk mengambil fitur-fitur yang terdapat dalam satu file audio (.wav) yang kemudian dari hasil ekstraksi tersebut nantinya akan digunakan untuk proses pengklasifikasian data suara dalam prosesnya akan mengambil beberapa fitur audio yang nantinya akan di simpan dalam format excel. Pada table I dilihat hasil dari ekstraksi fitur dari beberapa audio berlabel “Good” dan “Bad” fitur-fitur yang diekstraksi berupa static value of original data yang berisi nilai-nilai statistic mean, median, maximum, minimum value dan range. OriMean adalah nilai dari mean dari original signal, OriMedian yang adalah median dari original signal, OriMax yang adalah nilai dari maximum original value dan OriRange yang adalah jangkauan dari original signal.

F. Evaluasi Sistem

Evaluasi sistem data akan dihitung dengan menggunakan Confusion Matrix Pengujian dari model yang didapat, dengan data baru dan diambil 70 dataset baru untuk dihitung hasil akurasi, precision dan recall dengan confusion matrix.

Pada Tabel I *confusion matrix* didapatkan hasil prediksi dari 70 data pengujian, 60 data yang berhasil diprediksi dengan tepat dan 10 data sisanya salah diprediksikan. Untuk kategori *good* dari 35 data, 27 data berhasil diprediksi dengan tepat, sedangkan 8 data salah diprediksikan menjadi kategori *bad*. Sedangkan untuk kelas *bad* dari 33 data berhasil diprediksikan dengan benar dan sisanya 2 data salah diprediksikan menjadi *good*.

$$akurasi = \frac{27 + 33}{70} \times 100\% = 85\%$$

Hasil akurasi dari percobaan dengan menggunakan rumus (1) maka didapatkan dengan hasil akurasi 85%, menunjukan model yang dibuat untuk memprediksi data menghasilkan nilai tingkat kedekatan antara nilai prediksi dengan nilai yang sebenarnya dengan prentase yang tinggi.

$$precision_{good} = \frac{27}{27 + 8} \times 100\% = 78\%$$

Hasil $precision_{good}$ dari percobaan dengan menggunakan rumus (2) maka didapatkan hasil $precision_{good}$ sebesar 78% yang menunjukan model yang dibuat untuk memprediksi data menghasilkan nilai ketepatan antara label *good* yang diminta oleh pengguna dengan jawaban label *good* yang diberikan oleh sistem mempunyai persentase ketepatan 78%.

$$precision_{bad} = \frac{33}{33 + 2} \times 100\% = 95\%$$

Hasil $precision_{bad}$ dari percobaan dengan menggunakan rumus (2) maka didapatkan hasil $precision_{bad}$ sebesar 95% yang menunjukan model yang dibuat untuk memprediksi data menghasilkan nilai ketepatan antara label *bad* yang diminta oleh pengguna dengan

jawaban label *bad* yang diberikan oleh sistem mempunyai persentase ketepatan 95%.

$$recall_{good} = \frac{33}{33 + 8} \times 100\% = 80\%$$

Hasil $recall_{good}$ dari percobaan Tabel II dengan menggunakan rumus (3) maka didapatkan hasil $recall_{good}$ sebesar 80% yang menunjukan model yang dibuat untuk memprediksi data menghasilkan nilai tingkat keberhasilan sistem dalam menemukan kembali sebuah label *good* sebesar 80%.

$$recall_{bad} = \frac{27}{27 + 2} \times 100\% = 93\%$$

Hasil $recall_{bad}$ percobaan Tabel II dengan menggunakan rumus (3) maka didapatkan hasil $recall$ 93%. yang menunjukan model yang dibuat untuk memprediksi data menghasilkan nilai tingkat keberhasilan sistem dalam menemukan kembali sebuah label *good* sebesar 93%.

Pada tabel III adalah tabel hasil keseluruhan pengujian, pada pengujian pertama yang dilakukan dengan mengambil data suara dari dataset sebanyak 70 sampel dengan 35 sampel telur *good* dan 35 sampel telur *bad*. Dari sampel telur pada pengujian maka dipadatkan hasil positive rate yang merupakan nilai ketepatan sistem dalam memprediksi data dengan benar pada setiap kelas sedangkan negative rate adalah nilai kesalahan sistem dalam memprediksi data per kelas yang dimana pada pengujian ini didapatkan hasil positive rate untuk kategori label *good* adalah sebesar 78% sedangkan kategori label *bad* adalah sebesar 95 %.

Untuk negative rate kategori label *good* adalah sebesar 22% sedangkan kategori label *bad* adalah sebesar 5 %. Perhitungan precision dimaksudkan mencari ketepatan antara informasi yang diminta oleh pengguna dengan jawaban yang diberikan oleh sistem sehingga dari pengujian yang dilakukan didapatkan hasil untuk kategori label *good* adalah sebesar 78% sedangkan kategori label *bad* adalah sebesar 95 %, selanjutnya recall dimaksudkan untuk melakukan perhitungan untuk kategori label *good* adalah sebesar 80% sedangkan kategori label *bad* adalah sebesar 93 % dan nilai akurasi dimaksudkan untuk mencari nilai tingkat kedekatan antara nilai prediksi dengan nilai yang sebenarnya, yang dalam pengujian ini didapatkan hasil 85 %. Sehingga model dengan metode recurrent neural network yang dibuat pada proses training sudah dapat mengklasifikasikan telur puyuh yang mempunyai cangkang bagus dan cangkang retak melalui suara dengan sangat baik dengan menunjukan nilai precision diatas 78 %, recall diatas 85 % dan akurasi 85 %.

IV. PENUTUP

A. Kesimpulan

Setelah melakukan pengumpulan data hingga menyelesaikan semua proses dalam penelitian ini, maka dapat menarik kesimpulan bahwa ekstraksi fitur dengan menggunakan *Mel-Frequency Cepstral Coefficients* dan ekstraksi fitur data *statistic* dari 252 data suara yang menghasilkan 8.316 data ekstraksi sudah mampu menghasilkan model yang baik dalam klasifikasi telur puyuh menggunakan metode RNN. Dan dengan menggunakan metode *Recurrent Neural Network*, klasifikasi telur puyuh berbasis pengolahan suara berhasil dilakukan dan memiliki hasil yang baik, serta dapat mengklasifikasikan telur puyuh yang mempunyai cangkang bagus dan cangkang retak

melalui suara, dan menunjukkan nilai *precision* diatas 78%, *recall* diatas 85% dan akurasi 85%.

B. Saran

Dalam penelitian dari hasil pengumpulan, labeling, training, dan testing yang telah dilakukan masih ada hal – hal yang perlu dikaji agar dapat menjadi lebih baik. *Noise* perekaman suara dengan menggunakan alat masih sangat banyak dan belum tereduksi dengan baik, maka untuk penelitian selanjutnya diperlukan alat yang dapat meredam *noise* yang dihasilkan pada alat atau pada menemukan metode reduksi *noise* yang lebih baik sehingga proses klasifikasi boleh menghasilkan kualitas yang lebih baik.

KUTIPAN

- [1] A. . Sukma, A.hintono, B.E.Stiani, “Perubahan Mutu Hedonik Telur asin Sangrai Selama Penyimpanan,” *J. Anim. Agric.*, vol. 1, no. 1, pp. 585–598, 2012.
- [2] A. J. Devi Sahati, Dringhuzen Mamahit, “Klasifikasi Telur Ayam Berbasis Pengelolaan Suara,” pp. 1–8, 2018.
- [3] A. A. Rizal and S. Hartati, “PREDIKSI KUNJUNGAN WISATAWAN DENGAN *RECURRENT NEURAL NETWORK EXTENDED KALMAN FILTER* Program Studi Informatika , STMIK Bumigora Mataram Jurusan Ilmu Komputer dan Elektronika , FMIPA UGM , Yogyakarta,” vol. X, no. 1, pp. 7–18, 2017.
- [4] Nanang Hermawan, “Aplikasi Model *Recurrent Neural Network*,” 2014.
- [5] I. Habibie, “Identifikasi Judul Berita Clickbait Berbahasa Indonesia dengan Algoritma *Long Short Term Memory (LSTM) Recurrent Neural Network*,” *Repos. Institusi Univ. Sumatra Utara*, p. 59, 2018, [Online]. Available: <https://repositori.usu.ac.id/handle/123456789/8874>.
- [6] V. K. Ayyadevara and V. K. Ayyadevara, “*Recurrent Neural Network*,” *Pro Mach. Learn. Algorithms*, pp. 217–257, 2018, doi: 10.1007/978-1-4842-3564-5_10.
- [7] L. Ayu Rahning Putri and S. Hartati, *Klasifikasi Genre Musik Menggunakan Learning Vector Quantization dan Self Organizing Map*, vol. 9, no. 1. 2017.
- [8] A. Setiawan, A. Hidayatno, R. R. Isnanto, J. T. Elektro, F. Teknik, and U. Diponegoro, “Angga Setiawan, Achmad Hidayatno, R. Rizal Isnanto Jurusan Teknik Elektro, Fakultas Teknik, Universitas Diponegoro, Jln. Prof. Sudharto, Tembalang, Semarang, Indonesia,” pp. 1–7.
- [9] R. Sariningsih and I. Herdiman, “Mengembangkan kemampuan penalaran statistik dan berpikir kreatif matematis mahasiswa di Kota Cimahi melalui pendekatan open-ended,” *J. Ris. Pendidik. Mat.*, vol. 4, no. 2, p. 239, 2017, doi: 10.21831/jrpm.v4i2.16685.
- [10] A. Budiarti, “Bab 2 landasan teori,” *Apl. dan Anal. Lit. Fasilkom UI*, pp. 4–25, 2006.
- [11] A. A. Rizal and S. Soraya, “Multi Time Steps Prediction Dengan *Recurrent Neural*,” vol. 18, no. 1, pp. 115–124, 2018.
- [12] P. Li, Q. Wang, Q. Zhang, S. Cao, Y. Liu, and T. Zhu, “Non-destructive Detection on the Egg Crack Based on Wavelet Transform,” *IERI Procedia*, vol. 2, pp. 372–382, 2012, doi: 10.1016/j.ieri.2012.06.104.
- [13] A. Jain, A. R. Zamir, S. Savarese, and A. Saxena, “Jain Structural-RNN Deep Learning CVPR 2016 paper,” *Cvpr*, pp. 5308–5317, 2016, doi: 10.1109/CVPR.2016.573.
- [14] S. Hochreiter and J. Uergen Schmidhuber, “Long Shortterm Memory,” *Neural Comput.*, vol. 9, no. 8, p. 17351780, 1997, [Online]. Available: <http://www7.informatik.tu-muenchen.de/~hochreit%0Ahttp://www.idsia.ch/~juergen>.

- [15] F. A. Gers, J. Schmidhuber, and F. Cummins, “Learning to forget: Continual prediction with LSTM,” *Neural Comput.*, vol. 12, no. 10, pp. 2451–2471, 2000, doi: 10.1162/089976600300015015.

Exel Defrisco Tarkus, lahir di Palu 8 Febuari 1998. Penulis merupakan anak ke-1 dari 2 bersaudara, dari pasangan I Ketut Aris Tarkus dan Siska Rahel Memah. Penulis mulai menempuh Pendidikan di Taman Kanak-Kanak GPID Sumbersari (2002-2003), lalu melanjutkan ke Sekolah Dasar Negeri Sumbersari (2003-2009). Kemudian melanjutkan Sekolah Menengah Pertama Negeri 1 Langowan (2009-2012). Penulis melanjutkan Sekolah Menengah Atas Kristen 2 Binsus Tomohon (2012-2015). Pada tahun 2015, penulis melanjutkan Pendidikan ke salah satu Perguruan Tinggi Negeri di Kota Manado yaitu Universitas Sam Ratulangi (UNSRAT) dengan mengambil Program Studi S-1 Teknik Informatika di Jurusan Elektro Fakultas Teknik. Selama berada di bangku kuliah, penulis tergabung dalam Organisasi Kemahasiswaan skala Jurusan yaitu Himpunan Mahasiswa Elektro Fakultas Teknik UNSRAT (HME-FT UNSRAT), dimana Penulis mengambil peran sebagai Panitia di berbagai kegiatan HME FT-UNSRAT, dan menjadi Ketua Umum HME-FT UNSRAT 1 Periode (2017-2018). Selain itu, penulis juga menjadi Pengurus Badan Eksekutif Mahasiswa Fakultas Teknik UNSRAT (BEM-FT UNSRAT) 1 Periode (2018-2019). Pada bulan September tahun 2019 Penulis mengajukan proposal Skripsi untuk memenuhi syarat meraih gelar sarjana (S1) dengan judul Implementasi Metode *Recurrent Neural Network* Pada Kualitas Telur Puyuh yang kemudian disetujui dan melanjutkan pembuatan penelitian skripsi. Pembuatan skripsi ini dibimbing oleh dua dosen pembimbing, yaitu Sherwin R.U. Sompie, ST., MT dan Agustinus Jacobus, ST, M.Cs. Pada tanggal 1 Juli 2020.

