

# Identifikasi Penulis Berdasarkan Pola Tulisan Tangan Menggunakan Convolutional Autoencoder dan KNN

Muhammad Turmudzi<sup>1</sup>, Endang Setyati<sup>2</sup>

<sup>1,2</sup>Program Studi Pasca Sarjana Teknologi Informasi  
Institut Sains dan Teknologi Terpadu Surabaya

E-mail: <sup>1</sup> [muhammادتurmudzi@gmail.com](mailto:muhammادتurmudzi@gmail.com), <sup>2</sup> [endang@stts.edu](mailto:endang@stts.edu)

Identifikasi tulisan tangan dilakukan dengan beberapa tahapan, yaitu Akuisisi Citra dengan memanfaatkan mesin scanner dengan kualitas gambar 300dpi, Segmentasi dilakukan dengan metode threshold dan seleksi kontour dari gambar, penggabungan gambar hasil segmentasi, proses citra dari hasil segmentasi ke dalam Convolutional Autoencoder yang hasilnya diteruskan ke Transfer Learning (Lazy Learning) dalam hal ini penulis menggunakan metode KNN untuk mencocokkan tulisan tangan dari penulis. Penelitian dilakukan dengan menggunakan 100 dataset dari 20 penulis yang masing-masing penulis menulis 5 kali. Dataset yang digunakan di ujicoba pertama menggunakan penggalan kalimat pada tulisan tangan yaitu Judul dari Puisi Chairil Anwar. Ujicoba dilakukan dengan membandingkan Training menggunakan Convolutional Autoencoder dan tanpa menggunakan Convolutional Autoencoder. Hasil dari ujicoba dengan Convolutional Autoencoder memperoleh nilai akurasi sebesar 89% dan tanpa menggunakan Convolutional Autoencoder, didapatkan nilai akurasi sebesar 88%. Pada ujicoba menggunakan tulisan tangan full, diperoleh hasil akurasi rata-rata 50% jauh di bawah hipotesa sehingga tidak cocok untuk diterapkan pada identifikasi tulisan tangan. Perlu ada nya pembatasan tulisan tangan yang akan digunakan sebagai dataset dalam identifikasi tulisan tangan.

Kata kunci : *Convolutional Autoencoder, KNN, identifikasi tulisan tangan.*

Handwriting identification is carried out in several stages, namely Image Acquisition using a scanner with 300dpi image quality, Segmentation is carried out by the threshold method and contour selection of the image, merging the image segmentation results, processing the image from the segmentation results into Convolutional Autoencoder which results are forwarded to Transfer Learning (Lazy Learning), in this case the writer uses the KNN method to match the writer's handwriting. The research was conducted using 100 datasets from 20 authors, each of whom wrote 5 times. The dataset used in the first trial used a handwritten sentence fragment, namely the Title of Chairil Anwar's Poetry. Testing was conducted by comparing Training using Convolutional Autoencoder and without using Convolutional Autoencoder. The results of the trials with Convolutional Autoencoder obtained an accuracy value of 89% and without using Convolutional Autoencoder, an accuracy value of 88% was obtained. In the trial using full handwriting, the results obtained an average accuracy of 50%, far below the hypothesis so that it is not suitable for application in handwriting identification. There should be restrictions on handwriting that will be used as a dataset for handwritten identification.

**Keyword:** Convolutional Autoencoder, KNN, Handwritten Identification.

## I. PENDAHULUAN

Convolutional Neural Network adalah salah satu metode machine learning dari pengembangan Multi Layer Perceptron (MLP) yang didesain untuk mengolah data dua dimensi. CNN termasuk dalam jenis Deep Neural Network karena dalamnya tingkat jaringan dan banyak diimplementasikan dalam data citra. CNN memiliki dua metode; yakni klasifikasi menggunakan feedforward dan tahap pembelajaran menggunakan backpropagation. Cara kerja CNN memiliki kesamaan pada MLP, namun dalam CNN setiap neuron dipresentasikan dalam bentuk dua dimensi, tidak seperti MLP yang setiap neuron hanya berukuran satu dimensi.

Autoencoder adalah model neural network yang memiliki input dan output yang sama. Autoencoder mempelajari data input dan berusaha untuk melakukan rekonstruksi terhadap data input tersebut. Autoencoder biasa digunakan untuk mengurangi dimensi dari features (Dimensionality Reduction). Jika data mempunyai dimensi yang sangat tinggi (data dengan jumlah features yang sangat banyak) bisa jadi tiap features yang ada tersebar pada setiap dimensi dari data sehingga setiap data yang ada terlihat sangat berbeda. Untuk mengatasi masalah tersebut kita membutuhkan data yang sangat banyak atau mengurangi dimensi data tersebut.

## II. TINJAUAN PUSTAKA

Dalam melakukan penelitian akan selalu disertakan beberapa paper atau penunjang sebagai bahan referensi dan penunjang dalam melakukan penelitian berdasarkan penelitian sebelumnya. Sehingga apa yang akan kita lakukan nanti apakah meneruskan apa yang sudah dibuat mereka atau kita membuat sesuatu hal baru yang belum pernah dikerjakan dan dipublikasikan oleh orang lain

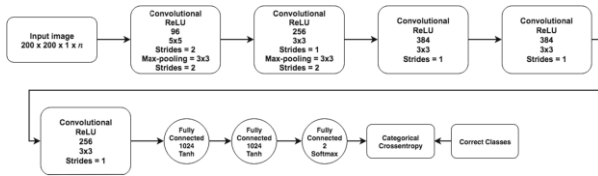
### A. Handwritten Texts for Personality Identification Using Convolutional Neural Networks.

Pada paper ini, dijelaskan tentang sistem biometrik penulis yang digunakan untuk identifikasi sifat personal penulis. Peneliti menggunakan 418 data set asli yang berupa gambar berwarna. Dari 418 data set asli tersebut diperoleh 216 bidang dari masing-masing dengan warna abu-abu. Dan kemudian dibinarikan menjadi 90.000 gambar. Dataset ini terdiri dari esai bahasa Spanyol tulisan tangan dari mahasiswa sarjana Meksiko.

Resolusi gambar-gambar ini bervariasi, jadi kami

memutuskan untuk menormalkan masing-masing hingga  $2000 \times 2600$  piksel. Karena jumlah gambar kecil, dataset diperluas, gambar di potong dengan ukuran masing-masing  $200 \times 200$  piksel dan menggeser 50 piksel ke kanan dan 20 ke bawah, untuk akhirnya memperoleh 216 potongan gambar per setiap gambar, sehingga diperoleh total 90.288 gambar untuk melatih model ini.

Model alur CNN yang pada paper ini adalah:



Gbr 1 Model alur CNN pada paper 1

Penulis melatih model dari awal menggunakan dataset yang disebutkan sebelum menggunakan metode Backpropagation dan Stochastic Gradient Descent (SGD) dengan tingkat pembelajaran sebesar 0,001. Dengan menggunakan 100 iterasi dan ukuran batch 50. Dari set lengkap patch menggunakan 80% untuk pelatihan, 10% untuk validasi dan 10% untuk pengujian. Gambar input memiliki ukuran  $200 \times 200$  piksel dan  $n = 50$  (ukuran batch).

Hasilnya ditunjukkan pada Tabel di bawah ini :

Tabel I  
Hasil Ujicoba

	Baseline (validationse t)	Ourmethod (validationse t)	Ourmethod (finalevaluatio n)
AUC	0.5014	0.5023	0.5009
F1 (multiclass)	0.2629	0.1931	0.2025
F1 (multilabel)	0.4994	0.2916	0.2738
BAC(multiclas)	0.2091	0.1888	0.2011
BAC(multilab)	0.5014	0.5023	0.5009
PAC(multiclas)	0.0136	0.0136	0.0135
PAC(multilabe)	0.5028	0.5028	0.5063
RegressionAB	-0.0004	-0.0876	-0.1384
RegressionR2	-1.0008	-1.1753	-1.2769

**B. Handwritten Recognition Using Deep-Learning.**

Dalam makalah ini penulis menyajikan metode inovatif untuk deteksi karakter tulisan tangan offline menggunakan Deep Learning. Di dunia sekarang ini telah menjadi lebih mudah untuk melatih jaringan deepneural karena ketersediaan sejumlah besar data dan berbagai inovasi Algoritma yang sedang berlangsung. Sekarang ini, jumlah daya komputasi yang diperlukan untuk melatih jaringan saraf telah meningkat karena ketersediaan GPU dan layanan berbasis cloud lainnya seperti Google Cloud platform dan Amazon Web Services yang menyediakan sumber daya untuk melatih Neural Network di cloud. Kami telah merancang sistem pengenalan karakter tulisan tangan berbasis segmentasi gambar. Dalam sistem kami, kami telah menggunakan OpenCV untuk melakukan pemrosesan Gambar dan telah menggunakan

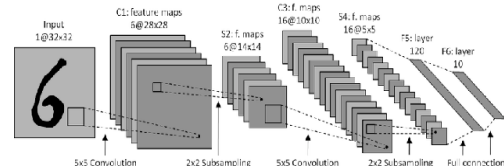
Tensorflow untuk melatih Jaringan saraf. Penulis telah mengembangkan sistem ini menggunakan bahasa pemrograman python.

Sistem ini terdiri dari dua bagian:

1. Aplikasi Android: Ini adalah tampilan depan sistem kami. Aplikasi Android membantu pengguna untuk mengklik gambar teks yang akan dikenali, menggunakan kamera ponsel pintar. Gambar ini diteruskan ke skrip python yang berjalan di server yang selanjutnya memproses gambar ini untuk mengekstrak informasi yang relevan
2. Server: Ini adalah backend dari sistem kami. Server ini adalah komputer yang mampu mengeksekusi skrip python. Ini diperlukan karena ponsel pintar android tidak memiliki daya komputasi yang diperlukan untuk menjalankan jaringan saraf dan melakukan operasi pemrosesan gambar. Juga penggunaan server untuk melakukan tugas-tugas intensif komputasi memungkinkan pengguna ponsel pintar yang lebih tua untuk menggunakan sistem kami. Kami menggunakan Model Jaringan Saraf Konvolusional dalam sistem kami. Kami menggunakan set data NIST yang tersedia untuk umum yang berisi sampel karakter tulisan tangan dari ribuan penulis. Model neural network yang digunakan adalah Convolutional Neural Network. Model jaringan saraf dilatih menggunakan Tensorflow yang merupakan perpustakaan open source yang digunakan untuk aplikasi pembelajaran mesin. OpenCV digunakan untuk melakukan berbagai operasi pemrosesan gambar seperti segmentasi, thresholding dan Operasi Morfologis.

Untuk merancang sistem pengenalan karakter tulisan tangan offline ini, kami telah menggunakan berbagai alat seperti Python, Android, OpenCV dan Tensorflow

Model CNN yang digunakan adalah sebagai berikut.



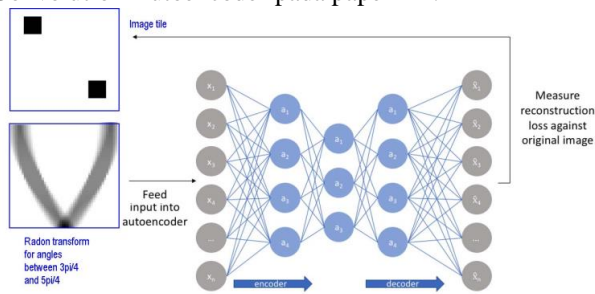
Gambar awalnya berukuran  $128 \times 128$  piksel. Gambar dalam set pelatihan dipotong hingga ukuran  $28 \times 28$ . Mengurangi ukuran gambar mengurangi waktu keseluruhan yang diambil untuk melatih model jaringan saraf. Setelah pelatihan model jaringan saraf, diperoleh akurasi hingga 94%.

Langkah-langkah dalam pemrosesan gambar: Pre-processing, Konversi to Grayscale, Thresholding dan Segmentasi gambar.

**C. Convolutional Autoencoder For Discriminating Handwriting Styles.**

Pada paper ini digunakan Convolutional Autoencoders

dengan input tulisan tangan. Input gambar di pre-processing dan di segmentasi menjadi ukuran 48x48x1. Berikut skema dari Convolution Autoencoder pada paper ini :



Gbr 3. Skema Convolution Autoencoder

Pada paper ini bisa digunakan untuk mendeteksi secara otomatis subnet gambar dari perbedaan warna, perbedaan orientasi dan ukuran.

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	(None, 48, 48, 1)	0
conv2d_1 (Conv2D)	(None, 48, 48, 16)	160
max_pooling2d_1 (MaxPooling2D)	(None, 24, 24, 16)	0
conv2d_2 (Conv2D)	(None, 24, 24, 8)	1160
max_pooling2d_2 (MaxPooling2D)	(None, 12, 12, 8)	0
conv2d_3 (Conv2D)	(None, 12, 12, 8)	584
max_pooling2d_3 (MaxPooling2D)	(None, 6, 6, 8)	0
conv2d_4 (Conv2D)	(None, 6, 6, 8)	584
max_pooling2d_4 (MaxPooling2D)	(None, 3, 3, 8)	0
flatten_1 (Flatten)	(None, 72)	0
reshape_1 (Reshape)	(None, 3, 3, 8)	0
conv2d_5 (Conv2D)	(None, 3, 3, 8)	584
up_sampling2d_1 (UpSampling2D)	(None, 6, 6, 8)	0
conv2d_6 (Conv2D)	(None, 6, 6, 8)	584
up_sampling2d_2 (UpSampling2D)	(None, 12, 12, 8)	0
conv2d_7 (Conv2D)	(None, 12, 12, 8)	584
up_sampling2d_3 (UpSampling2D)	(None, 24, 24, 8)	0
conv2d_8 (Conv2D)	(None, 24, 24, 16)	1168
up_sampling2d_4 (UpSampling2D)	(None, 48, 48, 16)	0
conv2d_9 (Conv2D)	(None, 48, 48, 1)	145
Total params: 5,553		
Trainable params: 5,553		
Non-trainable params: 0		

Gbr 4. Ringkasan model CAE

CNN adalah sebuah kelas dari Deep Neural Network yang pada umumnya digunakan untuk analisa citra visual. Pada subbab ini akan dijelaskan mengenai komponen-komponen penting penyusun CNN. Pada bab ini dijelaskan pula bagaimana CNN dapat menangani atau memproses input berupa gambar serta memperoleh hasil yang lebih baik jika dibandingkan dengan neural network biasa.

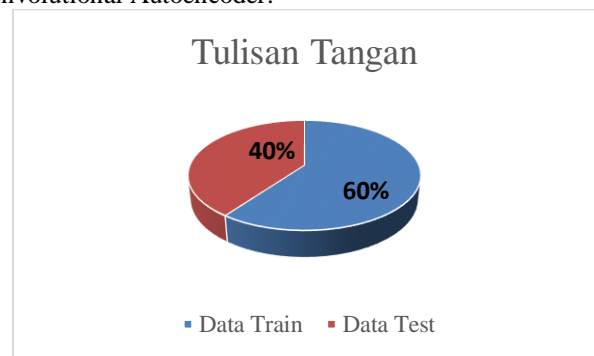
Cara paling mudah untuk memenuhi persyaratan format penulisan adalah dengan menggunakan dokumen ini sebagai template. Kemudian ketikkan teks Anda ke dalamnya

### III. BAHAN DAN METODE PENELITIAN

Pada penelitian ini akan dijelaskan beberapa tahapan sistem yang diusulkan yaitu penjelasan tentang dataset dan gambaran sistem proses Convolutional Autoencoder dan KNN.

#### A. Dataset

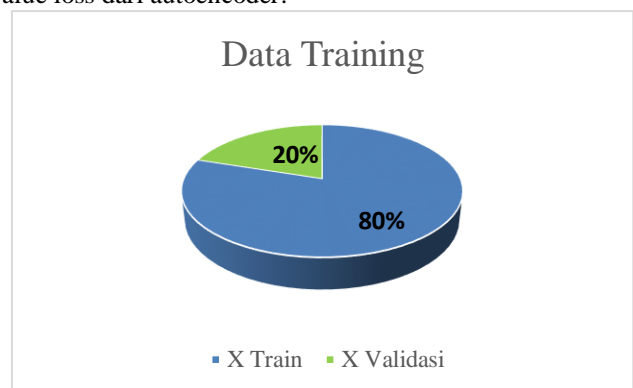
Dataset untuk ujicoba sistem menggunakan 100 gambar tulisan tangan yang ditulis oleh 20 orang dan masing-masing orang menulis 5 kali Puisi Chairil Anwar. pembagian dataset yang digunakan untuk proses training dan testing. Dataset dibagi menjadi 2 bagian. 60% dari tulisan tangan dipergunakan untuk data training dan 40% dipergunakan untuk data testing. Pada data training dibagi menjadi 80% untuk data training dan 20% untuk validasi data pada model Convolutional Autoencoder.



Gbr 5 Pembagian Data Set

Gambar 5 merupakan grafik yang menampilkan persentase data yang digunakan pada image classifier. Dataset yang digunakan untuk proses training adalah sebesar 60%. Dataset yang digunakan untuk melakukan testing adalah sebesar 40%.

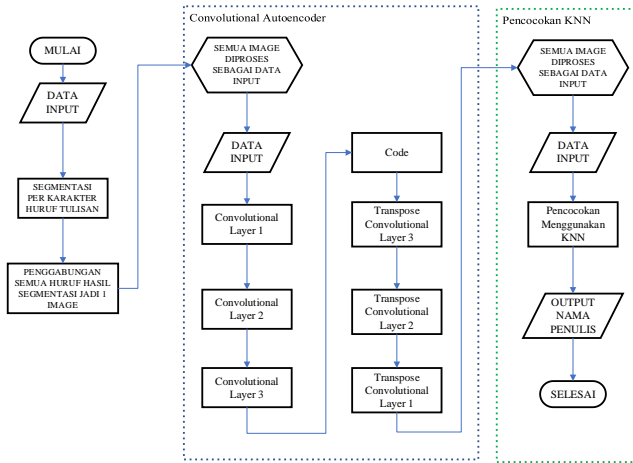
Dari pembagian dataset tersebut, pada data train dibagi lagi menjadi 2, yaitu data train dan data validasi untuk menentukan value loss dari autoencoder.



Gbr 6. Presentase Pembagian Data Set Training

Pembagian dari data train tersebut 80% untuk data train dan 20% untuk validasi nya.

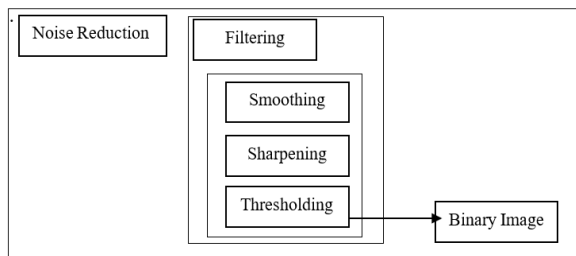
B. Gambaran Sistem



Gbr 7. Sistem yang diusulkan

1. Preprocessing

Gambar yang diambil (berupa file) diproses lebih lanjut agar dapat diolah pada tahap berikutnya dengan lebih mudah.



Gbr 8. Proses Preprocessing

2. Convolutional Autoencoder

Convolutional Autoencoder merupakan arsitektur neural network yang berusaha untuk menghasilkan output yang sama dengan inputnya. Autoencoder terdiri dari 2 bagian utama yaitu encoder dan decoder. Encoder pada autoencoder berfungsi untuk melakukan pengurangan ukuran pada data dengan mempertahankan fitur-fitur penting pada encoded data. Bagian Decoder berfungsi untuk melakukan rekonstruksi encoded data menjadi seperti input awal. Tujuan penggunaan autoencoder adalah untuk memperoleh encoded data yang mengandung fitur-fitur penting dari data asli.

Tabel II  
Model Convolutional Autoencoder

Encoder		Decoder	
Layer	Model	Layer	Model
Layer1	Conv 3x3, 32	Layer6	ConvTransposed 3x3, 64
Layer2	MaxPool 2x2	Layer7	ConvTransposed 3x3, 64
Layer3	Conv 3x3, 64	Layer8	ConvTransposed 3x3, 32
Layer4 (encoded)	MaxPool 2x2	Layer 9 (output)	ConvTransposed 3x3, 1
Layer5	Conv 3x3, 64	-	-

Model CAE di Bagian encoder dari arsitektur terdiri dari 4 layer. Penyusunan layer pada bagian encoder terdiri dari convolutional layer dan max pooling layer yang disusun secara bergantian. Bagian decoder terdiri dari 3 layer. Seperti penyusunan pada bagian encoder, bagian decoder juga disusun menggunakan convolutional Transposed layer yang disusun secara bergantian.

3. KNN

Hasil dari Convolutional Autoencoder kemudian diteruskan ke Lazy Learning (KNN) untuk dilakukan proses pencocokan tulisan / proses testing yang nantinya akan menghasilkan output nama penulis dari tulisan tangan yang sudah di proses.

Algoritma kNN (k-Nearest Neighbor) ini adalah algoritma klasifikasi berdasarkan tetangga terdekat. Dalam proses pencocokan tulisan tangan ini, untuk menghitung jaraknya akan digunakan 2 jenis untuk perbandingan yaitu Euclidean Distance dan Chebychev Distance. Euclidean Distance mirip dengan Pythagoras, hanya saja Euclidean Distance memiliki dimensi lebih dari 2.

$$d(x, y) = \sqrt{\sum_{i=1}^m (x_i - y_i)^2} \quad (1)$$

Berbeda dengan Chebychev untuk menghitung jarak yang dicari adalah jarak yang terbesar.

$$d(x, y) = \max_{i=1} |x_i - y_i| \quad (2)$$

IV. HASIL DAN PEMBAHASAN

Pada bab ini akan dijelaskan mengenai percobaan-percobaan yang dilakukan dalam penelitian untuk mengetahui seberapa besar tingkat keberhasilan pada penelitian yang dilakukan. Selanjutnya akan diketahui apakah sistem akan benar - benar dapat menghasilkan keluaran yang sesuai dengan tujuan yang diinginkan sebelum program ini diterapkan.

A. Ujicoba Convolutional Autoencoder

Uji coba Autoencoder ini menggunakan beberapa model Convolutional Autoencoder beberapa layer yang berbeda. Uji coba identifikasi Penulis akan menggunakan Image Encoder dan K Nearest Neighbor sebagai identifikasi nya.

Sistem akan dibangun menggunakan kombinasi dari Autoencoder dan identifikasi tulisan tangan yang memiliki performa terbaik pada proses training dan evaluasi masing-masing model.

Layer (type)	Output Shape	Param #
image_input (InputLayer)	[(None, 200, 900, 1)]	0
Layer1 (Conv2D)	(None, 200, 900, 32)	320
Layer2 (MaxPooling2D)	(None, 100, 450, 32)	0
Layer3 (Conv2D)	(None, 100, 450, 64)	18496
Layer4 (MaxPooling2D)	(None, 50, 225, 64)	0
Layer5 (Conv2D)	(None, 50, 225, 64)	36928
Layer6 (Conv2DTranspose)	(None, 100, 450, 64)	36928
Layer7 (Conv2DTranspose)	(None, 200, 900, 64)	36928
Layer8 (Conv2DTranspose)	(None, 200, 900, 32)	18464
Layer9 (Conv2D)	(None, 200, 900, 1)	289

Total params: 148,353  
 Trainable params: 148,353  
 Non-trainable params: 0

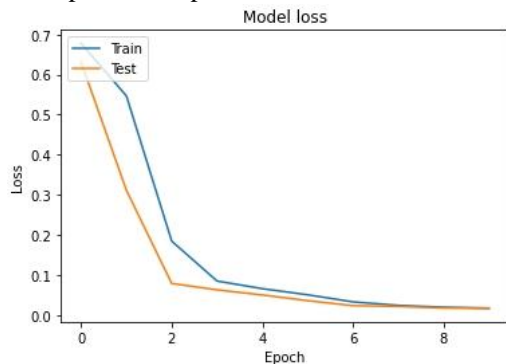
Gbr 9 Convolutional Autoencoder simple model

Pada gambar di atas adalah simple model Convolutional Autoencoder dengan Conv. Transposed.

```
Epoch 5/10
5/5 [=====] - 56s 11s/step - loss: 0.0658 - val_loss: 0.0502
Epoch 6/10
5/5 [=====] - 57s 11s/step - loss: 0.0506 - val_loss: 0.0359
Epoch 7/10
5/5 [=====] - 56s 11s/step - loss: 0.0329 - val_loss: 0.0235
Epoch 8/10
5/5 [=====] - 57s 11s/step - loss: 0.0241 - val_loss: 0.0214
Epoch 9/10
5/5 [=====] - 56s 11s/step - loss: 0.0196 - val_loss: 0.0175
Epoch 10/10
5/5 [=====] - 57s 11s/step - loss: 0.0173 - val_loss: 0.0164
```

Gbr 10 Value Loss CAE Conv. Transposed

Gambar 10 menjelaskan dari 10 epoch yang dilakukan Conv. Transposed memperoleh 0.0164 Value Loss.



Gambar 11 Grafik Value Loss CAE Conv. Transposed

Dari hasil ujicoba beberapa model Convolutional Autoencoder, hasil yang terbaik yang akan digunakan untuk Identifikasi Tulisan Tangan adalah Model Convolutional Autoencoder dengan Decoder menggunakan Conv. Transposed.

**B. Ujicoba Identifikasi Penulis**

Dilakukan Ujicoba dengan beberapa metode yang berbeda untuk mendapatkan hasil yang maksimal.

**1. Testing untuk Potongan Gambar Tulisan Tangan**

Ujicoba pertama dilakukan pada potongan gambar Tulisan tangan. Dalam hal ini diambil potongan dari Judul Puisi Chairil Anwar. Berikut hasil dari Ujicoba Identifikasi Tulisan Tangan yang melalui Proses Convolutional Autoencoder dan KNN dengan menggunakan Euclidean Distance dan mengambil 9 Tetangga terdekat :

Tabel III  
 Hasil Ujicoba Menggunakan Convolutional Autoencoder

No	Test	Acc	No	Test	Acc
1	AgusWibow	0.66	21	RinjaniPutra	1
2	AgusWibow	1	22	RinjaniPutra	1
3	DevinkaPa	1	23	Turmudzi	1
4	DevinkaPa	0.33	24	Turmudzi	1
5	DewiWulan	1	25	WahyuIsm	1
6	DewiWulan	1	26	WahyuIsm	0.33
7	EkaNovita	0.33	27	Wahyu K.	1
8	EkaNovita	1	28	Wahyu K.	1
9	EkaYufita	1	29	YulianKrist	0.66
10	EkaYufita	1	30	YulianKrist	1
11	Lailatul M.	1	31	aziz	1
12	Lailatul M.	1	32	aziz	0.33
13	LisviaPutri	1	33	enzeli	1
14	LisviaPutri	1	34	enzeli	1
15	MuchChafi	1	35	ervina	1
16	MuchChafi	1	36	ervina	1
17	Nisrina Z.	0.33	37	fatchurBeny	1
18	Nisrina Z.	1	38	fatchurBeny	1
19	Pratama	1	39	rizky	1
20	Pratama	1	40	rizky	0.66
Jumlah					35.66
Nilai Akurasi					89%

Hasil dari Ujicoba mendapatkan Nilai Akurasi 89%. Di samping itu juga dilakukan Ujicoba Identifikasi Tulisan Tangan tanpa melalui Proses Convolutional Autoencoder dan langsung KNN dengan menggunakan Euclidean Distance dan mengambil 9 Tetangga terdekat. Berikut hasil dari ujicoba nya:

Tabel IV  
 Hasil Ujicoba Tanpa Convolutional Autoencoder

No	Test	Acc	No	Test	Acc
1	AgusWibowo	1	21	RinjaniPutra	1
2	AgusWibowo	1	22	RinjaniPutra	1

3	DevinkaPasila	0.66	23	Turmudzi	0.66
4	DevinkaPasila	0.33	24	Turmudzi	1
5	DewiWulandari	1	25	WahyuIsmail	1
6	DewiWulandari	1	26	WahyuIsmail	0.33
7	EkaNovita	1	27	WahyuKurniawan	1
8	EkaNovita	1	28	WahyuKurniawan	0.66
9	EkaYufita	1	29	YulianKristian	1
10	EkaYufita	0.33	30	YulianKristian	1
11	LailatulMuhayarah	1	31	aziz	1
12	LailatulMuhayarah	1	32	aziz	0.66
13	LisviaPutri	1	33	enzeli	1
14	LisviaPutri	1	34	enzeli	1
15	MuchChafi	1	35	ervina	1
16	MuchChafi	1	36	ervina	0.66
17	NisrinaZubaidah	0.33	37	fatchurBeny	1
18	NisrinaZubaidah	1	38	fatchurBeny	1
19	PratamaEka	1	39	rizky	0.66
20	PratamaEka	1	40	rizky	1
<b>Jumlah</b>					<b>35.33</b>
<b>Nilai Akurasi</b>					<b>88%</b>

Hasil dari ujicoba tanpa menggunakan Convolutional Autoencoder, didapatkan nilai akurasi sebesar 88%. Jadi Hasil Akurasi yang lebih baik adalah dengan menggunakan Convolutional Autoencoder.

2. Testing Menggunakan Keseluruhan Gambar

Ujicoba selanjutnya dilakukan pada keseluruhan gambar Tulisan tangan. Berikut hasil dari Ujicoba Identifikasi Tulisan Tangan yang melalui Proses Convolutional Autoencoder dan KNN dengan menggunakan Euclidean Distance dan mengambil 9 Tetangga terdekat :

Tabel V  
Hasil Ujicoba dengan Berbagai Model

Model	Distance	Accuracy
Conv. Transposed 9 Layer	Euclidean	52,5%
Upsampling (512 x 512) 5 layer	Euclidean	59,1%
Upsampling (128 x 128) 5 layer	Euclidean	45%

Dari hasil ujicoba tabel 5.3, diperoleh hasil akurasi rata-rata 50% jauh di bawah hipotesa sehingga tidak cocok diterapkan pada identifikasi tulisan tangan. Perlu ada nya pembatasan tulisan tangan yang akan digunakan sebagai dataset dalam identifikasi tulisan tangan.

Grafik dan tabel harus terletak di tengah (*centered*). Grafik dan tabel yang besar dapat direntangkan pada kedua kolom. Setiap tabel atau gambar yang mencakup lebar lebih dari 1 kolom harus diposisikan di bagian atas atau di bagian bawah halaman.

V. KESIMPULAN

1. Dari hasil ujicoba pada tabel 5 dan 6 yang menggunakan potongan dari gambar tulisan tangan, bahwa hasil yang lebih baik adalah menggunakan Convolutional Autoencoder dan KNN sebagai identifikasi tulisan tangan dengan hasil 89%.
2. Penerapan Model Convolutional Autoencoder dan KNN pada keseluruhan gambar tulisan tangan mendapatkan hasil yang kurang bagus dengan rata-rata Nilai akurasi nya adalah 50%. Ini disebabkan karena semakin banyaknya feature yang di proses di KNN sehingga hasil perhitungan Jarak terdekat menjadi lebih tidak akurat.
3. Metode Convolutional Autoencoder dan KNN dapat diimplementasikan pada identifikasi tulisan tangan dengan syarat dibatasi dengan menggunakan dataset yang akan diproses tidak memiliki feature yang terlalu kompleks.

REFERENSI

- [1] Jose E. Valdez-Rodrigues, Hiram Calvo, And Edgardo M. Felipe-Riveron. Handwritten Texts for Personality Identification Using Convolutional Neural Networks. International Conference on Pattern Recognition. 2018.
- [2] Rohan Vaidya, Darshan Trivedi, Sagar Satra. Handwritten Recognition Using Deep-Learning. Proceedings of the 2nd International Conference on Inventive Communication and Computational Technologies. 2018.
- [3] Anamika Sen, Harsh Shah. Automated Handwriting Analysis System using Principles of Grapology and Image Processing. International Conference on Innovations in Information, Embedded and Communication System. 2017.
- [4] Sanae BOUTARFASS, Bernard BESSERER : Convolutional Autoencoder For Discriminating Handwriting Styles. European Workshop on Visual Information Processing. 2019.
- [5] Jose L. Vasquez, Antonio G. Ravelo-Garcia, Jesus B. Alonso, Malay Kishore Dutta, Carlos M. Travieso. Writer identification approach by holistic graphometric features using off-line handwritten words. The Natural Computing Forum. 2018.
- [6] Omar Santana, Carlos M. Travieso, Jeana B. Alonso, Miguel A. Ferrer. Writer Identification Based on Graphology Techniques. IEEE A&E System Magazines. 2010.
- [7] Mahesh Jangid, Sumit Srivastava. Handwritten Devanagari Character Recognition Using Layer-Wise Training of Deep Convolutional Neural Networks and Adaptive Gradient Methods. Journal of Imaging. 2018.
- [8] Champa and Ananda Kumar, Artificial Neural Network For Human Behavior Prediction Through Handwriting Analysis, International Journal of Computer Applications (0975-8887) Volume. 2010.
- [9] Champa and Ananda Kumar, Automated Human Behavior Prediction Through Handwriting Analysis, In Integrated Intelligent Computing (IChC), First International Conference on Integrated Intelligent Computing, pp. 160-165. IEEE,2010