

IDENTIFIKASI PENULIS MENGGUNAKAN CONVOLUTIONAL NEURAL NETWORK BERDASARKAN KARAKTER TULISAN TANGAN

(AUTHOR IDENTIFICATION USING CONVOLUTIONAL NEURAL NETWORK BASED ON HANDWRITTEN CHARACTERS)

Much Chafid¹⁾, Muhammad Turmudzi²⁾, Agus Wibowo³⁾, Pratama Eskaluspita⁴⁾, Ervina Yuniati Rokhmah⁵⁾, dan Dinda Heidiyuan Agustalita⁶⁾

^{1, 2, 3, 4, 5, 6)} Fakultas Teknik Program Studi Ilmu Komputer Universitas Gresik
Jawa Timur Indonesia

e-mail: chafid53@gmail.com¹⁾, muhammادتurmudzi@gmail.com²⁾, say.aguzz@gmail.com³⁾,
tamaeska@gmail.com⁴⁾, eyrewh@gmail.com⁵⁾, dindaheidi@gmail.com⁶⁾

ABSTRAK

Untuk membedakan karakter tulisan tangan diperlukan sebuah aplikasi atau sistem yang bisa digunakan untuk mengidentifikasi siapa penulis berdasarkan karakter tulisan tangan. Tujuan dari penelitian ini adalah agar sistem yang dibuat bisa menentukan suatu tulisan tangan adalah karya dari penulis tertentu. Pembuatan teks tulisan tangan dapat dilakukan dengan cara menangkap citra menggunakan scanner dengan kualitas citra 300 dpi, melakukan segmentasi menggunakan metode thresholding dan seleksi kontur dari citra, menggabungkan citra hasil segmentasi dan mengolah citra hasil segmentasi. Hasil convolutional autoencoder dapat diinput ke dalam transfer learning (lazy learning) menggunakan metode KNN untuk dicocokkan dengan tulisan tangan penulis. Pada Penelitian ini data yang digunakan adalah 100 yang dilakukan oleh 20 penulis, tiap penulis menulis 5 kali. Pada uji coba pertama menggunakan Dataset yang terdapat pada potongan kalimat tulisan tangan dari Judul dari Puisi yang di karang oleh Chairil Anwar. Pengujian dilakukan dengan membandingkan proses machine learning dengan dan tanpa convolutional autoencoder. Hasil pengujian dengan convolutional autoencoder menunjukkan akurasi sebesar 89%, sedangkan hasil pengujian tanpa convolutional autoencoder menunjukkan akurasi sebesar 88%

Kata Kunci: Convolutional Neurel Network, tulisan tangan.

ABSTRACT

An application or system is needed to distinguish handwritten characters, and it can be used to identify author-based handwritten characters. The aim of this research is so that the system created can determine whether a piece of handwriting is the work of a particular author. The creation of handwritten text can be done by capturing the image using a scanner with an image quality of 300 dpi, segmenting it using the thresholding method and contour selection from the image, combining the segmented images and processing the image from the segmentation results. The results of the convolutional autoencoder can be input into transfer learning (lazy learning) using KNN method to match with the author's handwriting. The study used 100 data sets from 20 authors, each of whom wrote five times. In the first trial, we used the dataset in handwritten sentence fragments from the title of a poem by Chairil Anwar. Tests were performed by comparing the machine learning process with and without a convolutional autoencoder. The test results with a convolutional autoencoder showed an accuracy of 89%, while the results without a convolutional autoencoder showed an accuracy of 88%.

Keywords: Convolutional Neurel Network, hand writing.

I. PENDAHULUAN

Tulisan tangan adalah coretan yang terdiri dari digit dan karakter yang dilakukan oleh seseorang dan melibatkan anggota tubuh yaitu tangan. Tulisan tangan itu berbesa satu sama lain bisa dikatakan ini adalah hal yang unik Seperti juga tanda tangan, yang pastinya ada perbedaan antara yang satu dengan yang lain [1]. Kadang sekilas terlihat ada kemiripan, tapi banyak dimungkinkan

dalam membuat tanda tangan ataupun tulisan tangan ada perbedaan di dalam proses pembuatannya

Untuk mengenali guritan huruf, kata, dan karakter tulisan, maka diperlukan sebuah sistem yang direncanakan dan diperlukan untuk belajar dari data yang sudah ada. Data yang sudah ada ini adalah hasil dari ekstraksi fitur yang didapat dari tulisan tangan. Tujuan dari pembelajaran ini adalah agar sistem yang dibuat bisa menentukan suatu

tulisan tangan adalah karya penulis tertentu. Sistem yang dibuat perlu dilakukan pembelajaran agar bisa membedakan karakter yang ditulis, misal tekanan pada saat menulis dan sudut kemiringan tertentu. Tiap penulis mempunyai ciri khas yang berbeda antara penulis satu dengan penulis yang lain. Kemampuan dalam hal pembelajaran mengenali pola baru serta menyimpan hasil proses belajar ke dalam database adalah sesuatu hal yang sangat dibutuhkan sistem. Dalam hal ini banyaknya jumlah tulisan tangan dari penulis akan diupayakan ditambah seiring waktu berjalan.

CNN yaitu salah satu metode yang digunakan oleh machine learning, ini pengembangan dari Multi Layer Perceptron (MLP) yang dirancang guna mengolah data yang sudah ada [6]. Convolutional Neural Network juga bisa dikategorikan jenis *Deep Neural Network*. *Convolutional Neural Network* menggunakan beberapa metode; yaitu klasifikasi dengan cara menggunakan feedforward serta tahap pembelajaran pada mesin menggunakan backpropagation. Cara kerja *Convolutional Neural Network* memiliki kemiripan dengan *Multi Layer Perceptron*, tetapi di dalam *Convolutional Neural Network* setiap neuron dipaparkan ke bentuk dua dimensi, berbeda lagi dengan *Multi Layer Perceptron* setiap neuronnya memiliki nilai satu dimensi.

Salah satu model dari Neural Network dengan masukan dan hasil keluaran yang memiliki kesamaan, disebut dengan istilah Autoencoder. Autoencoder memeriksa data masukan dan mencoba merekonstruksinya. Autoencoder sering digunakan untuk mengurangi ukuran objek (pengurangan dimensionalitas). Jika data berdimensi tinggi, setiap fitur dalam setiap dimensi data dapat berbeda, dan karenanya setiap bagian data yang ada berbeda. Untuk mengatasi masalah ini, diperlukan sejumlah besar data, atau dimensionalitas data yang ada perlu dikurangi.

II. STUDI PUSTAKA

Dalam kurun waktu lima tahun terakhir ada beberapa penelitian serupa yang mengangkat tema hampir mirip tetapi mungkin yang diangkat di penelitian yang lain adalah hal-hal yang tentunya mempunyai unsur yang berbeda, berikut adalah beberapa ulasan tentang penelitian terkait selama kurun waktu lima tahun

Penelitian yang dilakukan Oleh Muhammad

Turmuzdi dkk, dalam penelitian tersebut bahwa pola tulisan tangan seorang penulis mempunyai karakter tersendiri sehingga mampu diidentifikasi siapa penulis dari tulisan tangan tersebut [1]. Penelitian yang dilakukan oleh Dechy Deswita Indriani dkk, dalam penelitian tersebut untuk mengidentifikasi tanda tangan [2]. Selanjutnya penelitian yang dilakukan Lenny Husna dkk, tentang keaslian pola tanda tangan [3]. Penelitian yang dilakukan oleh Raudlatul Jannah dkk, tentang pengenalan dokumen tanda tangan [4]. Menurut penelitian yang dilakukan oleh Nanang Kasim dkk, tentang pola tulisan tangan aksara arab [5].

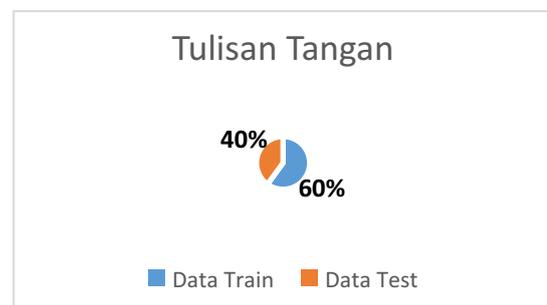
Dari penelitian yang sudah dilakukan hampir semuanya memiliki kemiripan meskipun ada hal-hal yang berbeda yang menonjolkan dalam penelitiannya.

III. METODE PENELITIAN

Pada penelitian kali ini penulis mencoba menjelaskan tahapan-tahapan yang digunakan oleh sistem yaitu :

A. Dataset

Pada uji coba sistem ini, 20 penulis melakukan tulisan tangan yang nantinya akan menghasilkan 100 citra. Dimana akan digunakan sebagai dataset pelatihan dan uji coba sistem, dengan kriteria setiap penulis menulis sebanyak 5 kali Judul dari Puisi Chairil Anwar. Komposisi yang digunakan dalam sistem ini menggunakan perbandingan 60% dataset digunakan untuk pelatihan, dan 40% digunakan untuk uji coba sistem.

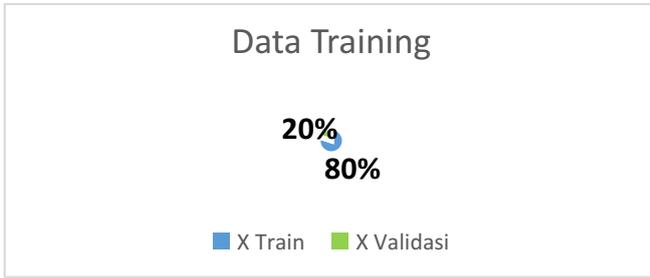


Gambar 1 Prosentase Dataset

Gambar 1 adalah jumlah prosentase dataset yang digunakan pada image classifier. Adapun pembagian dataset pada saat proses pembelajarannya mesin atau training, sebanyak 60% dari dataset sedangkan 40% digunakan untuk uji coba atau testing

Dengan menggunakan persentase pembagian

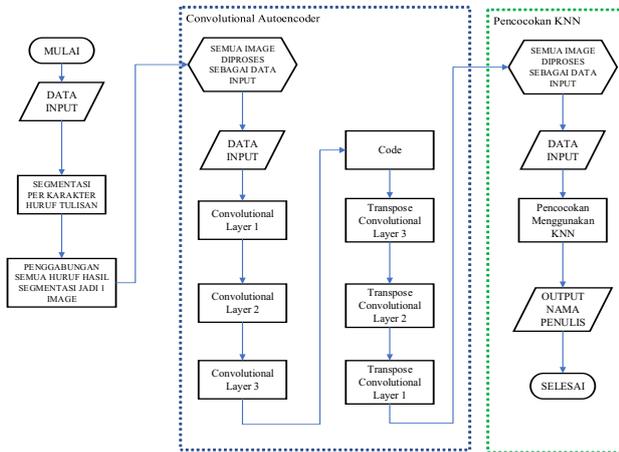
dataset diatas, 80% dari dataset tersebut akan digunakan sebagai data training dan 20% untuk data latih validasi penentuan value loss dari autoencoder.



Gambar 2. Diagram Data Set Ujicoba

Prosentase pembagian dataset 80% untuk data Ujicoba dan 20% untuk data validasi nya.

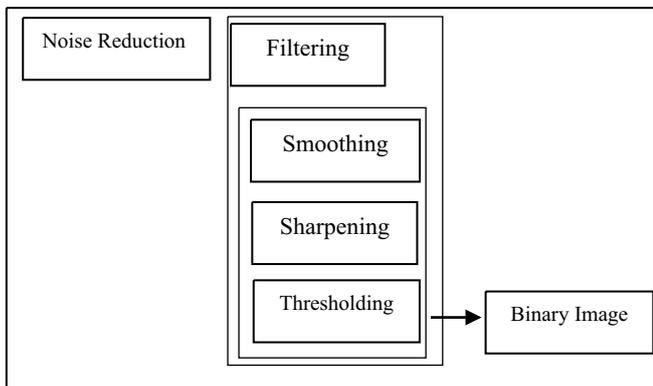
B. Gambaran Sistem



Gambar 3. Usulan pada sistem

a) Preprocessing

Citra yang digunakan dalam bentuk file akan diproses sehingga pengolahan ditahap selanjutnya akan lebih mudah



Gambar 4. Penjelasan Proses Preprocessing

b) Convolutional Autoencoder

Convolutional Autoencoder yaitu perancangan struktur neural network yang dapat menghasilkan output maupun input yang serupa. Autoencoder terbagi mendaji dua bagian yaitu yang pertama decoder dan yang kedua adalah encoder. Encoder mempunyai fungsi merubah ukuran atau resizeing dengan memperhatikan serta tidak meninggalkan fitur-fitur yang dianggap penting pada data. Sedangkan Decoder mempunyai fungsi mengembalikan seperti semula atau rekonstruksi data seperti pada saat awal kali input data. Autoencoder memiliki tujuan yaitu agar memperoleh encoded data dan tidak minggaikan fitur fitur yang dianggap penting dari data aslinya

Tabel 1. Model Convolutional Autoencoder

Encoder		Decoder	
Layer	Model	Layer	Model
Layer 1	Conv 3x3, 32	Layer 5	ConvTransposed 3x3, 64
Layer 2	MaxPool 2x2	Layer 6	ConvTransposed 3x3, 32
Layer 3	Conv 3x3, 64	Layer 7	ConvTransposed 3x3, 3
Layer 4	MaxPool 2x2	(Output)	-

Terdapat 4 layer bagian encoder yang terdapat pada arsitektur Model Convolutional Autoencoder. Convolutional dan Max Pooling merupakan layer pada bagian encoder yang disusun, Dimana proses penyusunannya dilakukan secara bergantian. Sedangkan ada tiga layer dibagian decoder. Seperti pada bagian encoder pada saat di susun, bagian decoder juga dilakukan penyusunan yang dilakukan secara bergantian saat penyusunannya, yaitu layer Convolutional Transposed.

c) KNN

Convolutional Autoencoder yang telah tersusun akan menghasilkan keluaran yang selanjutnya akan diaplikasikan metode Lazy Learning (KNN) untuk kelanjutan dari proses untuk pencocokan tulisan tangan/proses ujicoba sehingga nanti bisa mendapatkan hasil keluaran dari nama penulis beserta tulisan tangan dari masing-masing penulis dan akan diproses lebih lanjut.

K-Nearest Neighbor yaitu algoritma klasifikasi yang menggunakan perhitungan berdasarkan pada tetangga dengan posisi paling dekat. Dalam

pencocokan tulisan tangan, pada saat tahapan perhitungan jarak menggunakan dua jenis metode untuk dibandingkan yaitu Jarak Chebychev dan Jarak Euclidean. Perhitungan Euclidean Distance mempunyai kesamaan dengan perumusan Pythagoras, tetapi Euclidean Distance dapat menggunakan lebih dari dua dimensi

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

Terdapat perbedaan dengan Chebichev pada saat proses pencarian menghitung jarak adalah jarak yang mempunyai nilai terbesar.

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

IV. HASIL DAN PEMBAHASAN

Dibagian bab ini menjelaskan mengenai ujicoba yang dilakukan pada saat penelitian sehingga dapat memperoleh informasi dan bisa mendapatkan hasil yang baik pada tingkat keberhasilan dipenelitian yang dilakukan. Tahapan berikutnya akan menentukan system yang telah dibangun dapat memberikan hasil keluaran yang termasuk dalam harapan keluaran sistem jika dibandingkan dengan sebelum diterapkannya program ini.

A. Ujicoba Menggunakan Convolutional Autoencoder

Pada saat ujicoba Autoencoder pada penelitian kali ini terdapat beberapa layer yang berbeda dari model *Convolutional Autoencoder* pada saat melakukan testing untuk mengidentifikasi Penulis dengan menggunakan Encoder Citra dan KNN.

Sistem yang dibuat menggunakan gandingan dari Autoencoder serta mengidentifikasi tulisan tangan yang bisa menunjukkan performa yang baik pada saat melakukan proses pembelajaran mesin atau training serta bisa mengevaluasi dari model tersebut.

Layer (type)	Output Shape	Param #
image_input (InputLayer)	[(None, 32, 32, 1)]	0
Layer1 (Conv2D)	(None, 32, 32, 32)	320
Layer2 (MaxPooling2D)	(None, 16, 16, 32)	0
Layer3 (Conv2D)	(None, 16, 16, 64)	18496
Layer4 (MaxPooling2D)	(None, 8, 8, 64)	0
Layer5 (Conv2D)	(None, 8, 8, 64)	36928
Layer6 (Conv2DTranspose)	(None, 16, 16, 64)	36928
Layer7 (Conv2DTranspose)	(None, 32, 32, 64)	36928
Layer8 (Conv2DTranspose)	(None, 32, 32, 32)	18464
Layer9 (Conv2D)	(None, 32, 32, 1)	289
Total params: 148,353		
Trainable params: 148,353		
Non-trainable params: 0		

Gambar 5 Simple model Convolutional Autoencoder

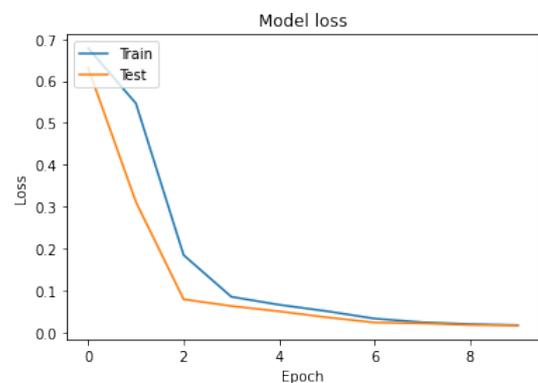
Gambar diatas merupakan hasil yang didapatkan pada Model Sederhana dari *Convolutional Autoencoder* dikombinasikan dengan *Conv. Transposed*.

```

Epoch 1/10
5/5 [=====] - 57s 11s/step - loss: 0.6782 - val_loss: 0.6310
Epoch 2/10
5/5 [=====] - 57s 11s/step - loss: 0.5466 - val_loss: 0.3113
Epoch 3/10
5/5 [=====] - 57s 11s/step - loss: 0.1844 - val_loss: 0.0788
Epoch 4/10
5/5 [=====] - 57s 11s/step - loss: 0.0849 - val_loss: 0.0629
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
    
```

Gambar 6 Value Loss CAE Conv. Transposed

Pada gambar di atas menggambarkan penerapan 10 epoch dari Conv. Transposed, dan memberikan 0.0164 Value Loss.



Gambar 7 Grafik Value Loss CAE Conv. Transposed

Berdasarkan hasil percobaan yang dilakukan, yang bisa menunjukkan nilai terbaik untuk

pengenalan tulisan tangan adalah model kombinasi antara *convolutional autoencoder* dengan menggunakan *decoder Conv.Transposed*.

B.Perhitungan Kuesioner

Percobaan dilakukan dengan cara mencoba mengaplikasikan dari metode-metode yang berbeda bertujuan agar menghasilkan output yang optimal.

Tabel 2 Hasil Testing Dengan Convolutional Autoencoder

No	Penulis	Akurasi	No	Penulis	Akurasi
1	Saniyatul	0.66	21	Saniyatul	1
2	Saniyatul	1	22	Yunia	1
3	Yunia	1	23	Muh Tur	1
4	Yunia	0.33	24	Muh Tur	1
5	Mufid	1	25	Darmawan	1
6	Mufid	1	26	Darmawan	0.33
7	Novita	0.33	27	Ervina.	1
8	Novita	1	28	Ervina.	1
9	Agus	1	29	Nabla	0.66
10	Agus	1	30	Nabila	1
11	Dimas.	1	31	Nugroho	1
12	Dimas.	1	32	Nugroho	0.33
13	Syafina	1	33	Amel	1
14	Syafina	1	34	Amel	1
15	Chafid	1	35	Andik	1
16	Chafid	1	36	Andik	1
17	Fashichul	0.33	37	Arinal	1
18	Fashichul	1	38	Arinal	1
19	Tama	1	39	Khuluq	1
20	Tama	1	40	Khuluq	0.66
Jumlah					35.66
Nilai Akurasi					89%

Hasil pengujian atau eksperimen menunjukkan nilai akurasi sebesar 89%. Selanjutnya, pada saat melakukan pengujian atau eksperimen, coretan tangan dikenali tidak harus berproses menggunakan *convolutional autoencoder*, dan KNN langsung menggunakan jarak Euclidean untuk mencari sembilan tetangga terdekat. Berikut adalah hasil pengujiannya:

Tabel 3 Hasil Testing tidak menggunakan Convolutional Autoencoder

No	Penulis	Akurasi	No	Penulis	Akurasi
1	Saniyatul	1	21	Saniyatul	1
2	Saniyatul	1	22	Yunia	1
3	Yunia	0.66	23	Muh Tur	0.66
4	Yunia	0.33	24	Muh Tur	1
5	Mufid	1	25	Darmawan	1
6	Mufid	1	26	Darmawan	0.33
7	Novita	1	27	Ervina.	1
8	Novita	1	28	Ervina.	0.66
9	Agus	1	29	Nabla	1
10	Agus	0.33	30	Nabila	1
11	Dimas.	1	31	Nugroho	1
12	Dimas.	1	32	Nugroho	0.66
13	Syafina	1	33	Amel	1
14	Syafina	1	34	Amel	1
15	Chafid	1	35	Andik	1
16	Chafid	1	36	Andik	0.66
17	Fashichul	0.33	37	Arinal	1
18	Fashichul	1	38	Arinal	1
19	Tama	1	39	Khuluq	0.66
20	Tama	1	40	Khuluq	1
Jumlah					35.3333
Nilai Akurasi					88%

Uji coba sistem tanpa diterapkannya *Convolutional Autoencoder*, mendapatkan akurasi sebesar 88%. Sehingga kesimpulan yang dapat ditarik adalah, bahwa *Convolutional Autoencoder* mampu meningkatkan akurasi dari sistem.

Pengujian yang selanjutnya yaitu di coba dilakukan pada gambar tulisan tangan secara keseluruhan. Di bawah ini adalah hasil pengujian pengenalan tulisan tangan dengan autoencoder konvolusional dan KNN menggunakan jarak Euclidean untuk memperoleh 9 tetangga :

Tabel 4. Output testing menggunakan 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 Testing di tabel 4, maka didapatkan hasil dengan akurasi rerata 50% jauh dari hipotesa sehingga tidak bisa digunakan untuk mengidentifikasi tulisan tangan. Diperkukan adanya batasan-batasan pada tulisan tangan untuk dataset di dalam mengidentifikasi tulisan tangan.

V. KESIMPULAN

Dari hasil percobaan atau pengujian pada bagian citra tulisan tangan pada Tabel 2 dan Tabel 3, citra yang memiliki performa lebih baik adalah citra yang menggunakan *convolutional autoencoder* dan KNN untuk pengenalan tulisan tangan, yaitu mencapai akurasi sebesar 89%.

Pada saat menggunakan KNN dan Model *Convolutional Autoencoder* pada seluruh Citra, hasil tulisan tangan yang didapat adalah kurang bagus rerata akurasi nilainya yaitu 50%. Ini dikarenakan semakin kompleksnya fitur yang ada di KNN sehingga memberikan hasil yang tidak akurat pada proses penghitungan jarak.

Pada saat mengimplementasikan Metode *Convolutional Autoencoder* dan KNN untuk mengidentifikasi tulisan tangan dengan beberapa persyaratan yang ditentukan dengan batasan-batasan dalam menggunakan dataset yang diproses lebih lanjut tidak memiliki banyak fitur.

UCAPAN TERIMA KASIH

Ungkapan terima kasih yang setinggi-tingginya kepada institusi kami yaitu Universitas Gresik yang banyak memberikan motivasi, dukungan, serta support sehingga penelitian dapat berjalan sesuai rencana ada halangan suatu apapun.

REFERENSI

- [1] Turmudzi, Muhammad, and Endang Setyati. "Identifikasi Penulis Berdasarkan Pola Tulisan Tangan Menggunakan Convolutional Autoencoder dan KNN." *JEECOM Journal of Electrical Engineering and Computer* 3.1 (2021): 8-13.
- [2] Sinaga, Elya Juni Arta, et al. "Identifikasi Tanda Tangan Dengan Menggunakan Metode Convolution Neural Network (CNN)." *J-INTECH (Journal of Information and Technology)* 12.1 (2024): 138-147.
- [3] Husna, Lenny, and Sestri Novia Rizki. "Pemanfaatan JST Pengenalan Keaslian Pola Tanda Tangan untuk Pencegahan Tindakan Pemalsuan Tanda Tangan." *Jurnal Teknik Informatika UNIKA Santo Thomas* (2023): 116-124
- [4] Jannah, Raudlatul, Miftahul Walid, and Hoiriyah Hoiriyah. "Sistem Pengenalan Citra Dokumen Tanda Tangan Menggunakan Metode CNN (Convolutional Neural Network)." *Energy: Jurnal Ilmiah Ilmu-Ilmu Teknik* 12.2 (2022): 54-61.
- [5] Kasim, Nanang, and Gibran Satya Nugraha. "Pengenalan pola tulisan tangan aksara arab menggunakan metode convolution neural network." *Jurnal Teknologi Informasi, Komputer, dan Aplikasinya (JTIKA)* 3.1 (2021): 85-95.
- [6] 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.
- [7] Rohan Vaidya, Darshan Trivedi, Sagar Satra. *Handwritten Recognition Using Deep-Learning.* Proceedings of the 2nd International Conference on Inventive Communication and Computational Technologies. 2018.
- [8] 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.
- [9] Sanae BOUTARFASS, Bernard BESSERER : *Convolutional Autoencoder For Discriminating Handwriting Styles.* European Workshop on Visual Information Processing. 2019.
- [10] 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.
- [11] Omar Santana, Carlos M. Travieso, Jeana B. Alonso, Miguel A. Ferrer. *Writer Identification Based on Graphology Techniques.* IEEE A&E System Magazines. 2010.
- [12] 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.
- [13] Champa and Ananda Kumar, *Artificial Neural Network For Human Behavior Prediction Through Handwriting Analysis,* International Journal of Computer Applications (0975–8887) Volume. 2010.
- [14] Champa and Ananda Kumar, *Automated Human Behavior Prediction Through Handwriting Analysis,* In Integrated Intelligent Computing (ICIIC), First International Conference on Integrated Intelligent Computing, pp. 160-165. IEEE,2010