

Klasifikasi Data Radar Menggunakan Algoritma Convolutional Neural Network (CNN)

Candra Nur Ihsan

*Pusat Sains dan Teknologi Atmosfer, LAPAN
Jl. Dr. Djunjunan No. 133 Bandung
candra.nurihsan9@gmail.com*

Abstrak. Penelitian ini berlandaskan pada masalah data radar di lokasi studi kasus, yaitu Lembaga Penerbangan dan Antariksa Nasional (LAPAN). Sistem sebelumnya yang sudah dibangun menggunakan radar kapal untuk mendeteksi cuaca yang diberi nama SANTANU. Terkadang saat pengamatan terjadi gangguan gangguan yang menyebabkan radar tidak menghasilkan data secara utuh sehingga sistem ini menghasilkan data yang tidak normal. Penelitian ini bertujuan untuk bisa mengenali data data yang tidak normal tersebut, sehingga jika dalam pengamatannya sering muncul data tidak normal dapat dilakukan penanganan pada radar cuaca tersebut. Hasil yang didapatkan dengan menggunakan algoritma CNN didapat nilai akurasi latih (training accuracy) sebesar 98.2% dan 96.6% pada akurasi validasi. Hasil ini menunjukkan data yang sudah diklasifikasi menggunakan algoritma CNN menghasilkan akurasi yang baik.

Kata Kunci - CNN, Klasifikasi, Data, Radar dan Cuaca

Abstract. This research is based on the problem of radar data at the case study location, namely the National Institute of Aeronautics and Space (LAPAN). The previous system that has been built using the ship's radar to detect the weather is named SANTANU. Sometimes when an observation occurs, there is a disturbance that causes the radar to not produce complete data, so the system produces abnormal data. This study aims to be able to identify the abnormal data, so that if the observations often appear abnormal data can be handled on the weather radar. The results obtained using the CNN algorithm obtained training accuracy of 98.2% and 96.6% of the validation accuracy. These results show that the data that has been classified using the CNN algorithm produces good accuracy.

Keywords - CNN, Classification, Data, Radar and Weather

Pendahuluan

Penyajian informasi cuaca saat ini dimanfaatkan oleh masyarakat untuk keperluan di berbagai sektor, misalnya dibutuhkan untuk prakiraan keberangkatan unuk kunjungan wisata, mencari ikan di laut dan sebagainya. Berbagai macam metode yang dikembangkan untuk melakukan prakiraan cuaca meliputi metode stokastik dan deterministik. Di Lembaga Penerbangan dan Antariksa Nasional (LAPAN) dikembangkan sistem bernama SANTANU (Sistem Pendeteksi Hujan Spasial) sistem ini dimanfaatkan untuk pengamatan cuaca secara real-time hujan yang sedang terjadi di sekitar area radar. Informasi yang disajikan kepada

pengguna dalam bentuk citra radar melalui website.

Masalah yang terjadi saat ini data yang dihasilkan oleh sistem SANTANU terkadang tidak normal ini disebabkan karena radar mengalami gangguan dan menghasilkan data yang tidak utuh sehingga sistem mengeluarkan data yang tidak normal dan menyajikan informasi yang tidak sesuai dengan kondisi cuaca di lapangan.

Sebelumnya gangguan ini hanya bisa dideteksi melalui pengamatan visual secara manual. Untuk mendeteksi gangguan lebih cepat diperlukan sebuah teknik untuk mengenali data data yang tidak normal sehingga bisa cepat tindak lanjuti apabila sistem mengeluarkan data tidak normal dalam jumlah yang banyak.

Teknik klasifikasi dapat digunakan untuk mengelompokkan data yang bertujuan untuk memudahkan untuk mendeteksi data data yang tidak normal. Klasifikasi merupakan bagian dari Machine Learning. Salah satu teknik klasifikasi yang umum digunakan untuk sebuah citra dapat menggunakan Algoritma Convolutional Neural Network (CNN). Algoritma CNN dapat melakukan pengenalan citra dan menghasilkan akurasi yang menyaingi manusia pada dataset tertentu

Berdasarkan kasus yang sudah dipaparkan, maka peneliti akan menyelesaikan permasalahan yang ada dengan cara melakukan "Klasifikasi Data Radar Menggunakan Algoritma CNN". Hasil penelitian ini diharapkan dapat bermanfaat dan membantu menangani gangguan gangguan yang terjadi pada radar. Penelitian dilakukan dengan menggunakan library tensorflow dan keras yang ada di python.

Landasan Teori

SANTANU

SANTANU adalah suatu sistem pemantauan hujan spasial, sebuah pengembangan teknologi x-band radar yang dikembangkan menjadi radar deteksi hujan oleh LAPAN (Ari Nugroho and Sinartra, 2018). Sistem ini mampu melakukan pengamatan dengan resolusi yang cukup tinggi dengan resolusi spasial sebesar 240 meter x 240 meter dan resolusi temporal sebesar dua menit, dengan daerah cakupan yang cukup luas kurang lebih 44 kilometer.

Sistem ini mengolah sinyal yang ditangkap oleh radar dan menyajikannya melalui website dengan alamat santanu.sains.lapan.go.id dalam bentuk citra spasial diatas peta sehingga pengguna dapat mengetahui kondisi hujan di daerah skala perkotaan.

Algoritma CNN

Convolutional Neural Network (CNN) adalah pengembangan dari Multilayer Perceptron (MLP) yang dirancang untuk mengolah data dua dimensi (Eka Putra, 2016). Convolutional Neural Network (CNN) termasuk dalam jenis deep learning karena kedalaman

jaringannya. Deep learning adalah cabang dari Machine learning yang dapat mengajarkan komputer untuk melakukan pekerjaan selayaknya manusia, seperti komputer dapat belajar dari proses training (Li Deng and Dong, 2013). CNN berbentuk sebuah layer yang memiliki susunan 3D (lebar, tinggi dan kedalaman) []. Ukuran layer digambarkan dengan sebuah lebar dan tinggi objek, sedangkan jumlah layer digambarkan dengan kedalaman objek. Secara umum CNN dibagi menjadi 2 layer, yaitu : Layer ekstraksi : layer ini berada pada awal arsitektur tersusun atas beberapa layer. Setiap layer tersusun atas neuron yang saling terhubung dengan local region sebelumnya. Layer Klasifikasi : layer ini tersusun atas beberapa layer. Setiap layer tersusun atas neuron yang saling terhubung secara penuh dengan layer lainnya.

Penelitian sebelumnya melakukan klasifikasi citra menggunakan CNN pada Caltech 101 (Eka Putra, 2016). Hasil yang didapat dengan menggunakan CNN Secara keseluruhan, klasifikasi citra yang dilakukan menghasilkan persentase kebenaran antara 20% sampai 50%. Penelitian lainnya melakukan penerapan algoritma CNN untuk klasifikasi citra buah (Maulana and Rochmawati, 2019) dan jenis kendaraan (Fadlia and Kosasih, 2019). Dilihat dari kesimpulan penelitian-penelitian tersebut membuktikan bahwa algoritma CNN memiliki akurasi yang baik untuk klasifikasi sebuah citra/gambar.

Python

Python adalah bahasa pemrograman interpretatif multiguna dengan filosofi perancangan yang berfokus pada tingkat keterbacaan kode. (Syahrudin and Kurniawan, 2018). Multi paradigm pemrograman didukung oleh python. Python menyediakan berbagai macam fitur, salah Bahasa pemrograman yang dinamis dan dilengkapi dengan manajemen memori secara otomatis. Saat ini, python dapat dijalankan di berbagai platform, diantaranya adalah : Java Virtual Machine, Palm, Amiga, Windows, Mac OS dan Linux/Unix.

Python didistribusikan dengan beberapa lisensi yang berbeda dari berbagai macam versi, namun python tidak bertentangan baik menurut definisi Open Source maupun General Public License (GPL).

Keras Framework

Keras Framework merupakan sebuah perangkat lunak yang berjalan diatas platform Python (Chollet, 2017) untuk mendukung proses komputasi cerdas diantaranya adalah untuk pengenalan citra (gambar) (Santoso and Ariyanto, 2018), Konversi Formulir(Saputra, Naf'an and Nurrochman, 2019) dan Klasifikasi (Sukardi1, Zainal Arifin2, 2017). Keras merupakan neural network library yang mudah digunakan (Keras Documentation, 2020). Fitur yang menonjol dari Keras yaitu: Keras menggunakan TensorFlow dan Theano sebagai backendnya. Keras dapat berjalan lancar di kedua CPU dan GPU. Keras mendukung hampir semua model jaringan saraf - sepenuhnya terhubung, konvolusional, pooling, recurrent, embedding, dan lain lain. Selanjutnya, model ini dapat dikombinasikan untuk membangun model yang lebih kompleks. Keras adalah kerangka kerja berbasis Python, yang membuatnya mudah untuk dideteksi dan dijelajahi atau dipelajari.

Penelitian terdahulu melakukan klasifikasi gambar dengan judul "Implementasi Deep Learning Berbasis Keras Untuk Pengenalan Wajah" (Santoso and Ariyanto, 2018). Hasil dari penelitian tersebut menunjukkan proses training dengan menggunakan Convolutional Neural Network menghasilkan tingkat akurasi kecocokan data sebesar 98.57% dan 90.57%

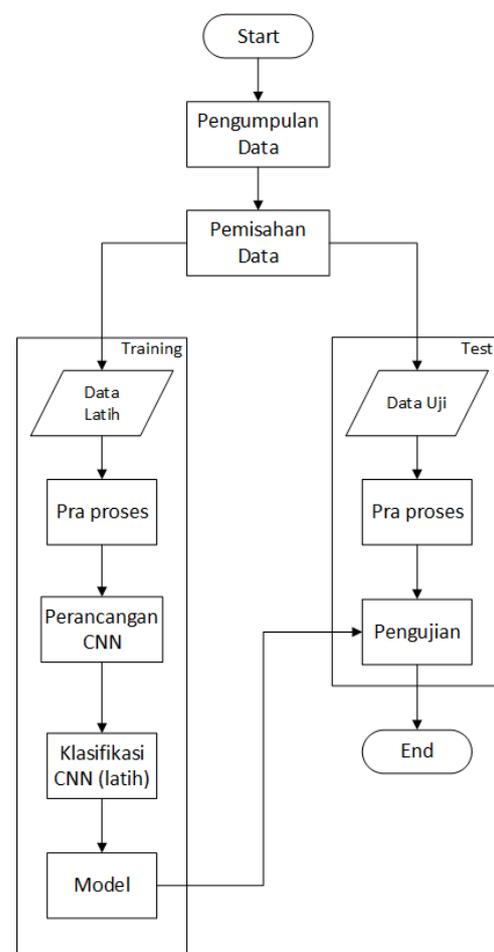
Tensorflow

Tensorflow adalah framework machine learning yang bersifat open source yang didukung oleh Google dalam rangka pengembangan penelitian berbasis komputer cerdas(Abadi *et al.*, 2016). Tensorflow menyediakan antarmuka untuk mengekspresikan algoritma machine learning secara fleksibel dan dapat berjalan diberbagai

sistem. Tensorflow merupakan salah satu library untuk data science paling populer dengan memiliki jumlah pengembang dan komunitas yang besar.

Metode Penelitian

Secara garis besar, proses klasifikasi data radar menggunakan Algoritma CNN dimulai dari pengumpulan data di lapangan. Data tersebut digunakan sebagai dataset untuk dilakukan proses *training* dan *testing*. Proses preprocessing selanjutnya dilakukan perancangan model Algoritma CNN dan dilanjutkan dengan melakukan pengujian model. Setelah model diuji maka akurasi akan didapatkan. Tahap akhir adalah menguji algoritma CNN dengan menggunakan *Confusion Matrix*

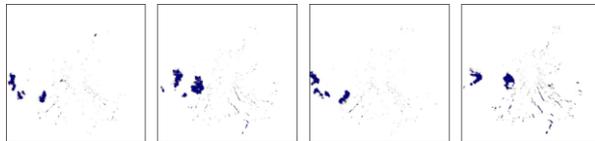


Gambar 1. Proses Klasifikasi Menggunakan CNN

Pengumpulan data

Data dikumpulkan langsung dari sistem SANTANU yang berada di kantor

LAPAN PSTA di pasteur, Bandung. Data berasal dari pengamatan tahun 2018 dan 2019. dari data ini kemudian diambil sampel untuk dijadikan data latih dan data uji. Sampel yang digunakan total sebanyak 1600 data untuk training dan 100 data untuk validasi.



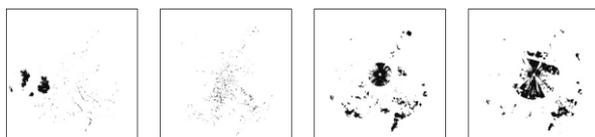
Gambar 2. Contoh Sampel Data dari SANTANU

Pembagian Data

Data yang dikumpulkan kemudian dibagi menjadi dua bagian yaitu data uji dan data latih dimana masing masing memiliki dua kelas klasifikasi yaitu klasifikasi data normal dan data tidak normal dengan persentase tertentu. Data uji akan digunakan dalam proses training model sedang data latih akan digunakan pada proses testing setelah model terbentuk.

Pra proses data

Sebelum melakukan training data terlebih dahulu harus diproses atau dipersiapkan, data dipersiapkan dengan cara mengubah ukuran data menjadi 200 pixel x 200 pixel dan mengubah citra menjadi grayscale karena yang klasifikasi akan berfokus pada bentuk data bukan pada warna data, proses ini juga bermanfaat untuk mengurangi beban belajar ketika proses training dilakukan. Gambar 3 contoh data yang telah di pra proses



Gambar 3. Pra Proses Data

Perancangan CNN

Pada proses ini ditentukan jumlah layer yang digunakan, jenis aktivasi, jumlah batch, jumlah epoch, ukuran convolution, ukuran pooling dan

beberapa parameter lainnya yang diperlukan.

Klasifikasi CNN

Diproses ini data yang telah disiapkan akan digunakan untuk melakukan *training* dengan kelas yang telah ditentukan yaitu data normal dan data tidak normal, dan akan menghasilkan sebuah model. Model ini kemudian akan digunakan dalam testing dan jika terbukti baik bisa diimplementasikan ke dalam sistem santanu

Pengujian

Proses ini akan menguji model yang telah hasilkan sebelumnya kemudian diuji dengan data uji yang telah disiapkan. Data uji ini akan mewakili data data baru yang nantinya dihasilkan oleh sistem SANTANU bertujuan untuk mengecek tingkat akurasi pada implementasi sesungguhnya. Tahap akhir adalah melakukan pengujian algoritma CNN menggunakan Confusion Matrix.

Hasil dan Pembahasan

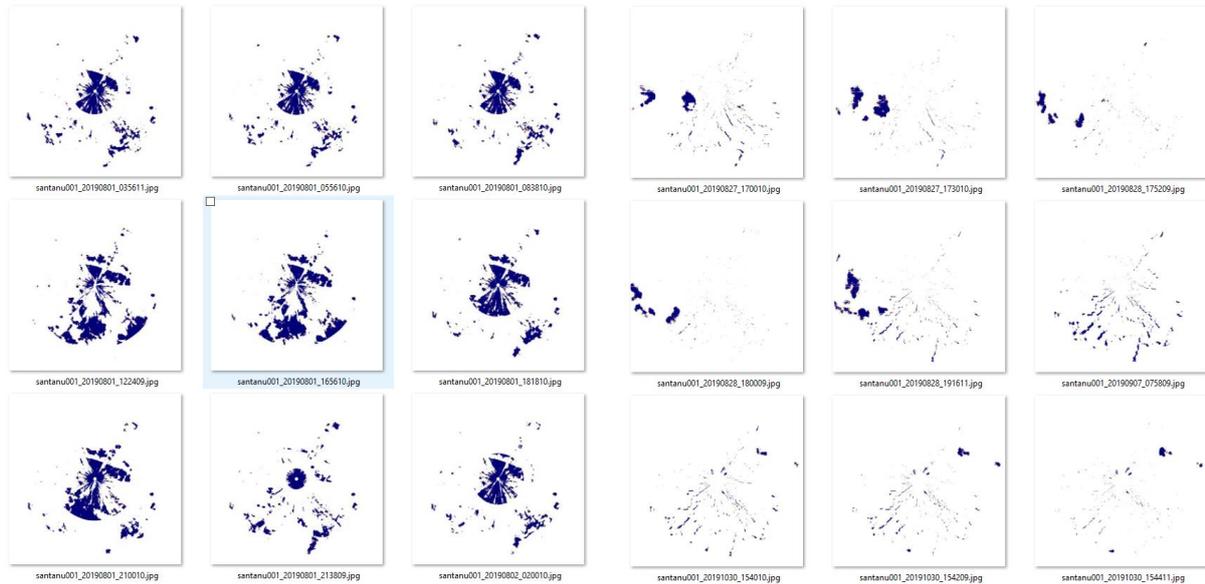
Pengumpulan Data

Dataset yang digunakan dalam penelitian ini merupakan dari sistem SANTANU yang berada di kantor PSTA, dan dibagi menjadi dua kelas yaitu kelas data tidak normal dan kelas data normal dengan pembagian seperti yang tertera pada Tabel 1.

Tabel 1. Pengumpulan Data Latih dan Uji

Data	Tidak Normal	Normal
Latih	800	800
Uji	50	50

Data Tidak normal dan data normal disajikan dalam bentuk visual untuk mempermudah pengguna membaca informasi yang sudah diolah dari data yang didapatkan dari radar. Sajian informasi ini dapat dilihat di situs web SANTANU LAPAN. Adapun contoh visual pembagian kelas data dapat dilihat pada Gambar 4.



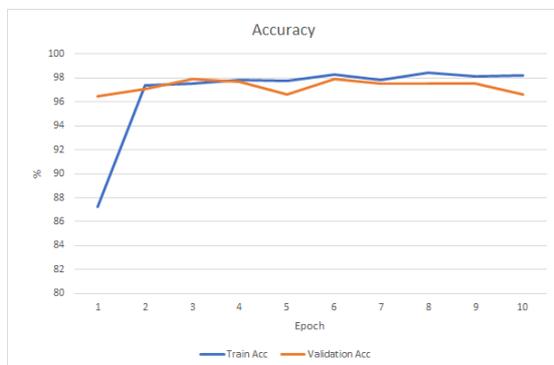
Gambar 4. Visual Pembagian Kelas Data

Processing Data

Training

Training dilakukan dengan menggunakan data latih yang berjumlah 800 data untuk kelas data normal begitu juga 800 data untuk kelas data tidak normal data ini kemudian akan dipisah kembali menjadi 70% untuk training dan 30% untuk validasi.

Training akan menggunakan 2 convolution dengan ukuran 3x3 dan pooling 2x2, layer dengan ReLU (Rectified Linear Unit) sebagai fungsi aktivasinya, 10 epoch dengan batch size sebesar 16 mendapatkan hasil sebagai terlihat dalam gambar 5.



Gambar 5. Grafik Data Akurasi

Epoch yang berbeda akan menghasilkan akurasi dan loss yang berbeda. Data loss terlihat pada gambar 6.



Gambar 6. Grafik Data Loss

Terlihat bahwa pada proses training mendapatkan akurasi sebesar 98.2% dan training loss sebesar 0.051 sedang pada proses validasi mendapatkan akurasi sebesar 96.6% dan loss sebesar 0.093.

Hasil ini menunjukkan bahwa model mampu melakukan kklasifikasi pada data SANTANU dengan baik yaitu mencapai 96.6% don loss yang sangat kecil hampir mendekati 0 yaitu sebesar 0.09.

Pengujian

Model yang telah dibangun sebelumnya akan diuji dengan data uji untuk melihat akurasinya, pengujian dilakukan pada 100 data dengan masing masing 50 data untuk setiap kelasnya.

Pengujian menghasilkan klasifikasi pada data normal berhasil mengenali 49 data sebagai data normal dan 1 data sebagai data tidak normal, sementara pengujian pada hasil data tidak normal mengenali 48 data sebagai data tidak

normal dan 2 data sebagai data normal, atau tergambar dalam tabel 2.

		Aktual	
		Data	Tidak Normal
Prediksi	Tidak Normal	48	1
	Normal	2	49
Total		50	50

Dari tabel tersebut didapatkan hasil :

1. True Normal : 49 dokumen
2. True Tidak Normal : 48 dokumen
3. False Normal : 1 dokumen
4. False Tidak Normal : 2 dokumen

Dengan hasil dari confusi matrix diperoleh akurasi:

$$\text{Akurasi} = (\text{True Normal} + \text{True Tidak Normal}) / (\text{True Normal} + \text{True Tidak Normal} + \text{False Normal} + \text{False Tidak Normal})$$

$$\text{Akurasi} = (49 + 48) / (1 + 48 + 49 + 2)$$

$$\text{Akurasi} = 0.97$$

$$\text{Akurasi} = 97\%$$

Akurasi algoritma CNN yang didapatkan dengan menggunakan *Confusion Matrix* adalah 0,97 atau 97% dalam pengklasifikasian data radar menjadi data normal dan data tidak normal. Untuk akurasi setiap kelas:

1. Data Normal
 Nilai Prediktif Normal = True Normal / (True Normal + False Normal)
 Nilai Prediktif Normal = 49 / (49 + 1)
 Nilai Prediktif Normal = 49/50
 Nilai Prediktif Normal = 0.98
 Nilai Prediktif Normal = 98%
 Hasil akurasi nilai prediktif normal mencapai 0.98 atau 98%, sehingga misclassification yang didapat hanya 12%.
2. Data Tidak Normal
 Nilai Prediktif Tidak Normal = True Tidak Normal / (True Tidak Normal + False Tidak Normal)
 Nilai Prediktif Tidak Normal = 48 / (48 + 2)
 Nilai Prediktif Tidak Normal = 48/50
 Nilai Prediktif Tidak Normal = 0.96
 Nilai Prediktif Normal = 96%
 Hasil akurasi nilai prediktif normal mencapai 0.96 atau 96%, sehingga misclassification yang didapat hanya 4%.

Berdasarkan hasil pengujian yang dilakukan, data radar berhasil dilakukan klasifikasi dengan nilai akurasi data klasifikasi menggunakan Algoritma CNN sebesar 96.6% dan nilai akurasi Algoritma CNN sebesar 97%.

Kesimpulan

Berdasarkan Model yang dibuat mendapat mendapatkan akurasi data radar sebesar 96.6%. Ini membuktikan penerapan Algoritma CNN memberikan hasil klasifikasi yang sangat baik, dimana hanya mendapatkan nilai *loss* akurasi sebesar 0,99. Klasifikasi berhasil dilakukan dengan baik menggunakan CNN dengan Tensorflow dengan akurasi yang didapatkan sebesar 97%. Berdasarkan akurasi tersebut dapat membuktikan bahwa penerapan CNN dengan Tensorflow memiliki kinerja yang sangat baik dalam teknik klasifikasi.

Daftar Pustaka

- Abadi, M. *et al.* (2016) 'TensorFlow: A System for Large-Scale Machine Learning', *12th USENIX Symposium on Operating Systems Design and Implementation*, pp. 265-283.
- Ari Nugroho, G. and Sinartra, T. (2018) *Sistem Pemantauan Hujan (Santanu)*. Bandung: Antasena.
- Chollet, F. (2017) 'Xception: Deep learning with depthwise separable convolutions', *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2017-Janua, pp. 1800-1807. doi: 10.1109/CVPR.2017.195.
- Eka Putra, W. S. (2016) 'Klasifikasi Citra Menggunakan Convolutional Neural Network (CNN) pada Caltech 101', *Jurnal Teknik ITS*, 5(1). doi: 10.12962/j23373539.v5i1.15696.
- Fadlia, N. and Kosasih, R. (2019) 'Klasifikasi Jenis Kendaraan Menggunakan Metode Convolutional Neural Network (Cnn)', *Jurnal Ilmiah Teknologi dan Rekayasa*, 24(3), pp. 207-215. doi: 10.35760/tr.2019.v24i3.2397.
- Keras Documentation (2020) Keras. Available at: <https://keras.io>.
- Li Deng and Dong, Y. (2013) 'Deep

- Learning Methods and Applications', *Foundations and Trends® in Signal Processing*. doi: 10.1136/bmj.319.7209.0a.
- Maulana, F. F. and Rochmawati, N. (2019) 'Klasifikasi Citra Buah Menggunakan Convolutional Neural Network', *Journal of Informatics and Computer Science*, 01, pp. 104-108. Available at: jurnalmahasiswa.unesa.ac.id > article.
- Santoso, A. and Ariyanto, G. (2018) 'Implementasi Deep Learning Berbasis Keras Untuk Pengenalan Wajah', *Emitor: Jurnal Teknik Elektro*, 18(01), pp. 15-21. doi: 10.23917/emitor.v18i01.6235.
- Saputra, W. A., Naf'an, M. Z. and Nurrochman, A. (2019) 'Implementasi Keras Library dan Convolutional Neural Network Pada Konversi Formulir Pendaftaran Siswa', *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, 1(10), pp. 524-531.
- Sukardi1, Zainal Arifin2, M. R. (2017) 'Klasifikasi Penentuan Gambar Berbasis Tensorform Dan Framework Dengan Algoritma Cnn', *Seminar Nasional Aptikom (Semnastikom)*, (November), pp. 1-4.
- Syahrudin, A. N. and Kurniawan, T. (2018) 'Input Dan Output Pada Bahasa Pemrograman Python', *Jurnal Dasar Pemrograman Python STMIK*, (January), pp. 1-7.