

SURAT PERNYATAAN BEBAS PLAGIARISME

Saya yang bertanda tangan di bawah ini:

Nama : Jonathan Andrew Pandapotan Simarmata
NIM : 1910314017
Program Studi : Teknik Elektro

Dengan ini menyatakan bahwa judul skripsi “ANALISIS PERFORMANSI METODE CONVOLUTIONAL NEURAL NETWORK (CNN) DAN SUPPORT VECTOR MACHINE (SVM) DALAM MENDETEKSI KEASLIAN DAN NOMINAL CITRA UANG KERTAS RUPIAH” benar bebas dari plagiarism dengan skor 21%. Apabila pernyataan ini terbukti tidak benar, maka saya bersedia menerima sanksi sesuai dengan ketentuan yang berlaku.

Demikian surat pernyataan ini dibuat untuk dipergunakan sebagaimana mestinya.

Jakarta, 14 Juli 2023

Yang Menyatakan,



Jonathan Andrew Pandapotan Simarmata

Pembimbing I



Achmad Zuchriadi P., S.T., M.T.

Pembimbing II



Fajar Rahayu I., S.T., M.T.

ANALISIS PERFORMANSI METODE CONVOLUTIONAL NEURAL NETWORK (CNN) DAN SUPPORT VECTOR MACHINE (SVM) DALAM MENDETEKSI KEASLIAN DAN NOMINAL CITRA UANG KERTAS RUPIAH

by Jonathan Andrew Pandapotan Simarmata

Submission date: 14-Jul-2023 10:05AM (UTC+0700)

Submission ID: 2130856807

File name: RevisiSkripsi-1910314017-JonathanAndrewPandapotanSimarmata.docx (1.67M)

Word count: 6701

Character count: 43468



ANALISIS PERFORMANSI METODE *CONVOLUTIONAL NEURAL NETWORK (CNN)* DAN *SUPPORT VECTOR MACHINE (SVM)* DALAM MENDETEKSI KEASLIAN DAN NOMINAL CITRA UANG KERTAS RUPIAH

SKRIPSI

**JONATHAN ANDREW PANDAPOTAN SIMARMATA
1910314017**

**PROGRAM STUDI TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS PEMBANGUNAN NASIONAL VETERAN
JAKARTA
2023**

ANALISIS PERFORMANSI METODE *CONVOLUTIONAL NEURAL NETWORK (CNN)* DAN *SUPPORT VECTOR MACHINE (SVM)* DALAM MENDETEKSI KEASLIAN DAN NOMINAL CITRA UANG KERTAS RUPIAH

Jonathan Andrew Pandapotan Simarmata

ABSTRAK

¹ Sulitnya membedakan uang asli dengan palsu mengharuskan masyarakat untuk meningkatkan kewaspadaannya terhadap keaslian uang yang dimiliki. Meskipun sekarang sudah banyak transaksi non-tunai namun demikian masih diperlukan adanya transaksi tunai dalam beberapa bidang. Terlebih pengecekan keaslian uang secara manual terbilang kurang efektif dan efisien karena apabila dilakukan dalam jumlah banyak memakan waktu dan cukup membuang energi. Akibatnya, sistem deteksi keaslian uang kertas rupiah perlu dibuat dan sistem ini ⁴⁵ dapat nantinya dapat digunakan dalam perangkat atau mesin transaksi tunai.. *Convolutional Neural Network (CNN)* yang merupakan salah satu metode untuk melakukan klasifikasi citra dan mendeteksi sebuah objek. CNN merupakan sebuah metode terbaik yang sering digunakan dalam memecahkan permasalahan image classification dan object detection. Pada penelitian ini akan dilakukan proses pengklasifikasian pada uang kertas rupiah dengan menggunakan kumpulan data ³ citra uang rupiah untuk mengklasifikasikan 6 kelas citra uang kertas rupiah yaitu 20.000 Asli, 20.000 Palsu 50.000 Asli, 50.000 Palsu 100.000 Asli, dan 100.000 Palsu ¹⁰ dengan menggunakan metode CNN dan SVM untuk melakukan klasifikasi citra uang kertas rupiah.

⁶ **Kata Kunci:** Uang Kertas Rupiah, Klasifikasi Citra, *Convolutional Neural Network (CNN)*, *Support Vector Machine (SVM)*, *Deep Learning*, *Machine Learning*

PERFORMANCE ANALYSIS OF THE CONVOLUTIONAL NEURAL NETWORK (CNN) AND SUPPORT VECTOR MACHINE (SVM) METHODS IN DETECTING THE AUTHENTICITY AND NOMINAL VALUE OF THE RUPIAH BANKNOTES

Jonathan Andrew Pandapotan Simarmata

ABSTRACT

The difficulty of distinguishing real money from fake money requires the public to increase their awareness of the authenticity of the money they have. Even though there are now many non-cash transactions, cash transactions are still needed in several fields. Moreover, checking the authenticity of money manually is somewhat less effective and efficient because if done large quantities, it takes time and is quite a waste of energy. As a result, a system for detecting the authenticity of rupiah banknotes needs to be developed and this system can later be used in devices or cash transaction machines. Convolutional Neural Network (CNN) is a method for classifying images and detecting an object. CNN is the best method that is often used in solving image classification and object detection problems. In this study, a classification process for rupiah banknotes will be carried out using a data collection of rupiah banknote imagery to classify 6 image classes of rupiah banknotes, namely 20,000 Genuine, 20,000 Counterfeit, 50,000 Genuine, 50,000 Counterfeit, 100,000 Genuine, and 100,000 Counterfeit using the CNN and SVM methods for classify the image of rupiah banknotes.

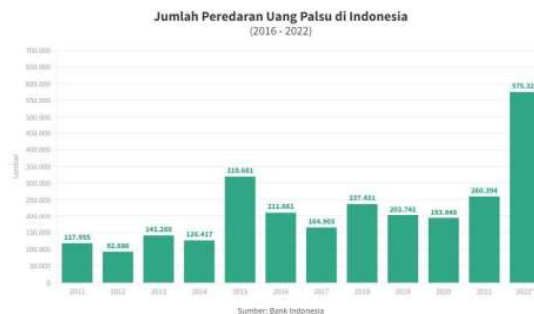
Keywords: *Rupiah Banknotes, Image Classification, Convolutional Neural Network (CNN), Support Vector Machine (SVM), Deep Learning, Machine Learning*

PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi semakin pesat perkembangannya terkhusus di bidang komunikasi dan informasi. Oleh karena itu, informasi dan komunikasi sangat mudah dan sangat cepat terjadi pada setiap orang. Hal tersebut sangat berpengaruh pada segala aspek di kehidupan yang terjadi setiap hari. Namun, teknologi yang berkembang saat ini baik secara langsung maupun tidak langsung menjadi pedang bermata dua. Hal tersebut karena kemajuan perkembangan teknologi yang dapat mendorong dan mempermudah peningkatan kesejahteraan serta kemajuan dalam peradaban manusia, sekaligus menjadikannya sebagai alat yang efektif untuk melakukan kegiatan ilegal/melawan hukum.

Dengan perkembangan dari teknologi saat ini, kejahatan yang disebabkan oleh penggunaan perkembangan teknologi juga semakin meningkat. Tindak pidana yang dimaksud merupakan perbuatan yang bertentangan dengan aturan hukum dan dapat meresahkan masyarakat. Salah satu tindak pidana dengan memanfaatkan perkembangan teknologi adalah pembuatan uang palsu. Peredaran uang kertas palsu umumnya terdiri dari pecahan Rp. 1.000 hingga pecahan Rp. 100.000 dengan jumlah banyaknya yang berbeda - beda.



Gambar 1.1 Jumlah Peredaran Uang Palsu di Indonesia

Sulitnya dalam membedakan uang rupiah asli dengan yang palsu membuat masyarakat harus meningkatkan kewaspadaannya terhadap keaslian uang yang diterima setiap bertransaksi. Meskipun saat ini sudah banyak diterapkan transaksi non-tunai namun keberadaan transaksi tunai masih diperlukan dalam beberapa bidang. Lebih lagi, pendeteksian keaslian uang rupiah secara manual

terbilang kurang efektif dan efisien karena apabila dilakukan dalam jumlah banyak memakan waktu dan cukup membuang energi. Akibatnya, sistem deteksi keaslian uang kertas rupiah perlu dibuat dan sistem ini juga nantinya dapat digunakan dalam perangkat atau mesin transaksi tunai.

Dalam melakukan proses deteksi keaslian uang kertas rupiah, ²¹terdapat beberapa metode yang dapat dilakukan salah satunya yaitu deteksi citra atau ¹disebut pengolahan citra digital (*digital image processing*). Pengolahan citra digital adalah suatu teknik pengolahan citra yang bertujuan untuk meningkatkan kualitas suatu citra agar dapat dengan mudah diinterpretasikan oleh manusia atau komputer yang berasal dari foto atau gambar bergerak. Penulis ingin melakukan penelitian sistem pengolahan citra dengan memfokuskan pada perbandingan antara metode klasifikasi ⁴¹*Support Vector Machine (SVM)* dan *Convolutional Neural Network (CNN)* untuk sistem pengolahan citra ini. Hal ini didasari untuk mendapatkan metode klasifikasi terbaik yang digunakan pada sistem pengolahan citra uang kertas. Judul dari penelitian yang akan penulis buat adalah “Analisis Performansi ⁵metode *Convolutional Neural Network (CNN)* dan *Support Vector Machine (SVM)* dalam ¹⁷Mendeteksi Keaslian dan Nominal Citra Uang Kertas Rupiah”. Alasan menggunakan algoritma CNN beserta SVM adalah karena algoritma *Deep Learning* dan *Machine Learning* tersebut merupakan yang terbaik dan umumnya digunakan dengan pengaplikasian pengolahan citra yang sederhana.

³⁰**1.2 Tujuan Penelitian**

Tujuan dari penelitian tugas akhir ini adalah:

- a. Membuat sistem klasifikasi uang kertas rupiah berbasis aplikasi pada *smartphone*.
- b. Untuk melakukan analisa perfomansi dari klasifikasi citra menggunakan metode SVM dan CNN dalam mendeteksi keaslian dan nominal uang kertas rupiah.

1.3 Rumusan Masalah

Rumusan masalah dalam penelitian tugas akhir ini adalah sebagai berikut.

- a. Bagaimana membuat aplikasi yang dapat mendeteksi keaslian dan nominal uang kertas rupiah?
- b. Bagaimana hasil Analisa peromansi untuk melakukan klasifikasi citra menggunakan metode SVM dan CNN dalam penerapan klasifikasi uang kertas?

1.4 Batasan Masalah

Batasan masalah dalam penelitian tugas akhir ini adalah sebagai berikut.

- a. Pembuatan hanya terfokus berupa sistem yaitu aplikasi pada *smartphone*.
- b. Citra uang kertas yang digunakan adalah citra uang kertas rupiah cetakan PERURI dengan emisi tahun 2022 pecahan Rp 20.000,00, Rp 50.000,00, dan Rp 100.000,00 dan uang palsu berupa cetakan (*print*) kertas HVS dengan kategori nominal yang sama.
- c. Uang kertas yang digunakan dalam kondisi baik dan tidak terlipat.
- d. Data citra yang digunakan adalah citra uang kertas rupiah yang diambil menggunakan kamera utama *smartphone*.
- e. Algoritma yang digunakan adalah *Convolutional Neural Network* (CNN) dan *Support Vector Machine* (SVM).

1.5 Sistematika Penulisan

Berikut ini merupakan sistematika penulisan yang berupa penjelasan susunan penulisan dari bab I sampai dengan bab V, adapun isi bab – bab tersebut adalah sebagai berikut.

BAB I PENDAHULUAN

Bab ini berisi latar belakang, rumusan masalah, tujuan, batasan permasalahan, dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Bab ini berisikan dasar ilmu yang mendukung dan penelitian terdahulu yang berkaitan dengan penelitian ini.

BAB III METODOLOGI PENELITIAN

Bab ini berisi rancangan penelitian berupa metode penelitian digunakan, prosedur penelitian berupa flowchart, dimana dalam tahap ini akan dilakukan pengolahan citra digital sesuai dengan metode dan algoritma yang telah ditetapkan. Penelitian ini menggunakan metode eksperimental untuk melakukan uji coba metode CNN & SVM dalam deteksi keaslian mata uang rupiah kertas.

BAB IV HASIL DAN PEMBAHASAN

Bab ini berisi proses uji coba berdasarkan parameter-parameter yang ditetapkan, dan kemudian dilakukan analisa terhadap hasil uji coba tersebut.

BAB V PENUTUP

Bab ini berisi kesimpulan yang dapat diambil dari penelitian ini serta saran untuk pengembangan selanjutnya dengan tujuan agar lebih bermanfaat untuk banyak orang.

12 BAB 2 LANDASAN TEORI

2.1 Uang Kertas

Uang kertas rupiah adalah alat untuk transaksi perdagangan. Alat tukar ini mempunyai bentuk lembaran dari bahan baik kertas ataupun bahan lainnya yang memiliki karakteristik layaknya kertas. Uang ini memiliki izin penggunaan sebagai sarana pertukaran di seluruh Republik Indonesia yang terdapat pada Undang – Undang No. 23 tahun 1999, yang diadopsi oleh Pemerintah Indonesia, dalam hal ini Bank Indonesia.

Suatu identitas diaplikasikan dalam uang rupiah berupa tanda dengan simbol unik untuk mencegah manipulasi atau penyelewengan terhadap uang rupiah. Menurut informasi dari Bank Indonesia, pemahaman tentang karakteristik pencetakan uang didasarkan pada gagasan bahwa semakin banyak penanaman (*security features*) sebuah uang, maka semakin tinggi nilai nominalnya. Secara umum, ciri-ciri keaslian uang rupiah dapat dikenali berupa unsur pengaman yang tertanam pada bahan uang dan teknik cetak yang digunakan, yaitu sebagai berikut:

- 1) *Watermark* (tanda air), dengan menggunakan cahaya akan terdapat simbol gambar yang dapat dilihat secara tembus pandang.
- 2) *Security thread* (benang pengaman), ketika terkena sinar ultraviolet benang yang ditanam atau dipasang ke dalam bentuk garis lurus vertikal akan berkilau. Pada gambar 2.1. merupakan contoh uang kertas rupiah yang asli dengan tampilan visual *Security Thread*.



Gambar 2.1. *Security Thread*

- 3) *Intaglio* (cetak dalam), tanda pengaman berupa cetak dalam akan terasa kasar ketika disentuh. Pada gambar 2.2. merupakan contoh tampilan visual *Intaglio*.



Gambar 2.2. *Intaglio*

- 4) *Rectoverso* (gambar saling isi) merupakan metode pencetakan yang digunakan pada bagian depan dan belakang dengan bentuk tertentu, ketika diterangi cahaya, terlihat menggabungkan atau mengisi. pada Gambar 2.3. merupakan uang kertas rupiah yang asli dari tanda pengaman *Rectoverso*.



Gambar 2.3. *Rectoverso*

- 5) *Optically variable ink* (tinta berubah warna) yaitu perubahan warna pada tinta yang tercetak apabila dilihat dari berbagai sudut pandangan. Berikut contoh uang kertas rupiah pada Gambar 2.4 dengan tampilan visual *Optically Variable Ink*.



Gambar 2.4. *Optically Variable Ink*

- 6) *Invisible ink* (tinta tidak tampak) dengan penyinaran ultraviolet akan terlihat perubahan tinta bagian tertentu yang sebelumnya tidak tampak pada kondisi normal. Pada Gambar 2.5 berikut terdapat tampilan visual *Invisible Ink* pada uang kertas rupiah.



Gambar 2.5. *Invisible Ink*

- 7) *Latent image* (gambar tersembunyi) dengan penglihatan sudut pandang tertentu, akan terdapat suatu gambar tersembunyi terdapat pada uang. *Latent Image* dalam uang kertas rupiah terdapat pada Gambar 2.6. dibawah ini.

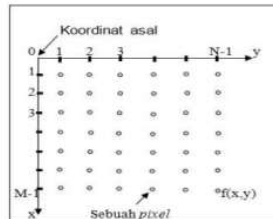


Gambar 2.6. *Latent Image*

2.2 Citra Digital

Dalam bentuk yang paling sederhana, citra adalah gambar atau representasi tampilan sebuah objek. Komputer tidak dapat mewakili visual analog; oleh karena itu, mereka tidak dapat memprosesnya secara langsung. Sebuah citra analog harus diubah menjadi citra digital sebelum dapat diproses oleh komputer. Gambar yang dapat diproses oleh komputer disebut citra digital. Sebuah citra yang dihasilkan oleh perangkat rekam data dapat optik, seperti foto, analog, seperti sinyal video yang ditampilkan di televisi, atau digital, yang dapat langsung disimpan pada media penyimpanan..

Sebuah citra digital adalah matriks di mana terdiri dari indeks baris dan kolom menandai titik dalam gambar, dan elemen matriksnya (disebut sebagai elemen pixel atau gambar) menandakan tingkat keabuan di titik tersebut. Pada gambar 2.7, citra digital akan dinyatakan berupa matriks $N \times M$ (baris \times kolom).



Gambar 2.7. Koordinat Citra Digital

Ketika citra diproses menggunakan komputer digital, citra pertama kali diubah menjadi nilai diskrit dari tingkat abu-abu sesuai posisi dimana gambar atau elemen piksel berada. Sebuah citra digital adalah sebutan yang diberikan untuk pemrosesan citra ini. Setiap citra digital berbeda dalam hal ukuran citra, resolusi, dan format lainnya. Citra digital biasanya memiliki bentuk persegi, lebar dan tinggi tertentu, dan diukur sebagai titik atau piksel.

2.3 Pengolahan Citra

Beberapa disiplin ilmu, termasuk fisika (optik, nuklir, gelombang, dll), elektronika, matematika, seni, fotografi, dan teknologi informasi, semuanya termasuk dalam bidang multidisiplin pemrosesan gambar. Komputer biasanya digunakan sebagai dasar untuk pemrosesan gambar dua dimensi yang merupakan pemrosesan gambar digital secara umum. Pengolahan gambar digital juga dapat diterapkan pada pengolahan data dua dimensi dalam arti yang lebih luas. Sebuah gambar digital adalah *array* yang mencakup nilai nyata dan kompleks, masing-masing diwakili oleh deret bit tertentu.

Pengolahan gambar digital adalah bidang yang menangani masalah seperti meningkatkan kualitas gambar, transformasi gambar, memilih fitur gambar terbaik untuk analisis, ekstraksi informasi, deskripsi objek, pengujian objek, dan kompresi. Untuk tujuan pemrosesan, transportasi, dan penyimpanan data, pengurangan data. Sebuah gambar berfungsi sebagai input untuk pemrosesan gambar, sementara gambar yang diproses berfungsi sebagai output..

2.3.1 Akuisisi Citra

Tahap awal dalam membuat gambar digital biasanya adalah akuisisi citra. Menemukan data penting dan mempelajari cara menyimpan citra digital adalah dua tujuan pengambilan citra. Alat-alat untuk pencitraan dipersiapkan terlebih dahulu dalam langkah ini. Imaging adalah proses mengubah gambar yang terlihat – seperti foto, lukisan, patung, dll – menjadi gambar digital. Ini sering dilakukan melalui kamera digital, scanner, dan kamera tradisional.

2.3.2 Segmentasi Citra

Proses **segmentasi citra** melibatkan memisahkan hal-hal yang berbeda satu sama lain atau antara objek dan latar belakang. Setiap objek dalam citra dapat dipisahkan menggunakan metode segmentasi ini dan kemudian digunakan sebagai input untuk operasi lain. Metode pemrosesan citra yang dikenal sebagai segmentasi gambar yang juga dapat dipahami sebagai cara untuk analisa dan identifikasi objek yang membutuhkan persepsi visual dengan memisahkan area (region) objek dari area latarnya (*background*).

Hal ini dapat diklaim bahwa segmentasi citra dilakukan untuk menjaga objek dalam gambar atau untuk membagi gambar menjadi bagian yang berbeda, masing-masing akan memiliki atribut yang sama untuk objek atau wilayah. Gambar dengan objek tunggal dibedakan dari latar belakang mereka. Segmentasi sering dilakukan sebagai langkah pertama dalam mengkategorikan objek. Karakteristik objek akan ditentukan setelah segmen gambar selesai. Ciri-ciri item tersebut dapat ditentukan dengan membandingkan lebar dan panjang objek, warna, atau bahkan teksturnya.

2.3.3 Preprocessing

Pre-processing adalah langkah dalam proses pengolahan citra yang memperbaiki atau meningkatkan citra asli sebelum proses selanjutnya. Untuk membuat citra yang lebih baik selama tahap pemisahan data, pengolahan citra menggunakan teknik pra-pengolahan. Pengujian pola dan teknik pra-pengolahan memiliki banyak kesamaan. Ilmu pengujian pola menggunakan pengukuran numerik dari karakteristik atau kualitas objek untuk mengkategorikan atau mengklasifikasikan sesuatu. Satu hal yang dapat diidentifikasi dan diberi nama adalah model itu sendiri. Sidik Jari adalah ilustrasi dari sebuah pola. Sebuah pola adalah kumpulan hasil pengukuran atau observasi yang dapat dicitrakan sebagai vektor atau matriks.

2.3.4 Principles Component Analysis

Secara umum, PCA adalah transformasi ruang vektor yang digunakan untuk mengurangi ukuran kumpulan data yang sangat besar. Dengan pemetaan, data asli dengan **7** banyak variabel dapat diinterpretasikan menjadi beberapa variabel saja. Tujuan dari metode ini adalah **13** untuk menemukan komponen utama yang akan digunakan. PCA biasanya digunakan untuk mengurangi atau memilih fitur dalam data tanpa mengubah karakteristik data secara signifikan.

2.3.5 Klasifikasi Citra

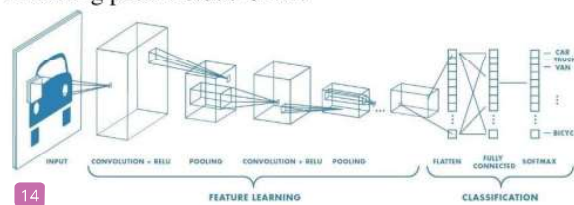
22 Klasifikasi citra yaitu cara untuk mengelompokkan piksel-piksel suatu citra ke dalam beberapa kelas sehingga setiap kelas dapat mendeskripsik¹² suatu entitas dengan karakteristik tertentu. Klasifikasi adalah teknik pengambilan data yang digunak⁵² untuk membuat model sampel data yang tidak terklasifikasi yang digunakan untuk mengklasifikasikan sampel data baru ke dalam kelas yang sesuai. Pada tahap klasifikasi citra, nilai parameter yang merepresentasikan properti dari objek pada setiap kategori digunakan sebagai data masukan. Data masukan kemudian diproses untuk mendapatkan formula yang digunakan untuk mengidentifikasi objek.

Ada dua proses utama dalam tahap klasifikasi, yaitu proses pelatihan dan proses pengujian. Dalam proses pelatihan, data pelatihan dianalisis menggunakan algoritma klasifikasi. Data uji kemudian digunakan untuk memastil³ keakuratan dari klasifikasi yang digunakan..

1) Convolutional Neural Network (CNN)

Salah satu algoritma *deep learning* yang dibuat oleh *Multilayer Perceptron* (MLP), *Convolutional Neural Network* (CNN), memproses data menjadi representasi dua dimensi seperti gambar atau suara. Menggunakan teknik pembelajaran terstruktur (*supervised learning*), CNN digunakan untuk mengkategorikan data label. Variabel yang ditargetkan dan data terlatih menjadi aspek penting dengan tujuan dari metode ini adalah untuk mengelompokkan suatu data pada kategori data yang sudah ada. Untuk deteksi, identifikasi, dan segmentasi objek, metode CNN sering digunakan. CNN menggunakan data gambar untuk belajar secara langsung sehingga menghilangkan persyaratan untuk ekstraksi fitur terpisah.

Terdapat beberapa langkah dan perencanaan arsitektur yang dapat digunakan secara teknis pada CNN. Sebuah peta fitur atau *feature map* berupa beberapa tabel merupakan tahapan dalam prosesan suatu masukan (*input*) hingga ke masukan (*output*). *convolution layer*, *pooling layer*, dan *fully connected layer* merupakan tiga lapisan utama pada setiap tahapan yang akan dirancang pada metode CNN.



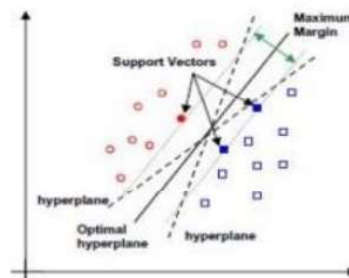
14 Gambar 2.8 Arsitektur Convolutional Neural Network

Pada Gambar 2.8 yaitu lapisan – lapisan pada metode CNN terdiri dari neuron tiga dimensi yang berupa lebar, tinggi, dan kedalaman. Dimensi kedalaman menunjukkan jumlah lapisan, sedangkan lebar dan tinggi menunjukkan ukuran lapisan. *Convolution* adalah tahap awal dalam arsitektur CNN yang didasarkan pada gambar-gambar tersebut. Penggunaan *kernel* dengan ukuran tertentu didasarkan pada perhitungan jumlah *kernel* yang akan digunakan memiliki keterkaitan dengan jumlah fitur yang dihasilkan. Setelah proses fungsi aktivitas selesai, proses berlanjut pada proses *pooling*. Biasanya, fungsi aktivasi dipertahankan dengan menggunakan fungsi Aktivasi ReLU (*Rectifier Linear Unit*). Sampai ada peta fitur yang cukup untuk pindah ke lapisan *fully connected* dan kelas output, proses ini dapat dilakukan secara berulang.

2) Support Vector Machine (SVM)

Tujuan utama dari Support Vector Machine (SVM) adalah untuk mengidentifikasi hyperplane terbaik di ruang input pemisahan setiap kelas. Dengan menggunakan fungsi kernel untuk menentukan margin maksimum dari hyperplane, maka akan didapat hyperplane optimal. Data yang telah diproses ke dimensi yang lebih lanjut menggunakan hyperplane dipisahkan menggunakan teknik kernel. Seperti yang ditunjukkan pada gambar 2.2, dengan memaksimalkan jarak antara *hyperplane* dan *margin* didapatkan akurasi tertinggi dapat diperoleh.

Dalam SVM mencoba untuk menemukan fungsi *splitter* atau pemisah sebagai batasan setiap kelas, yaitu pemisahan kelas berbeda dengan optimalisasi *hyperplane*, sebagaimana diilustrasikan pada gambar 2.10.



Gambar 2.9 Hyperplane pada Support Vector Machine (SVM)

2.3.6 Confusion Matrix

Confusion Matrix merupakan metode untuk mendapatkan hasil prediksi dari pelatihan model yang telah dijalankan. Hasil dari confusion matrix merupakan hasil klasifikasi sistem yang telah melakukan prediksi pada data kemudian dikategorikan terhadap data actual. Hasil dari *Confusion matrix* akan ditampilkan seperti pada tabel 2.1.

Tabel 2.1 Model Confusion Matrix

		Predicted Value	
		Positive (0)	Negative (1)
Actual Value	Positive (0)	TP	FN
	Negative (1)	FP	TN

Dimana :

TP = *true positive*, data aktual positif yang diprediksi benar

FP = *false positive*, data aktual negatif yang diprediksi positif

FN = *false negative*, data aktual positif yang diprediksi negatif

TN = *true negative*, data aktual negatif yang diprediksi benar

2.4 Library Keras & Scikit Learn

Keras adalah pustaka yang terdapat dalam pemrograman *python* berguna untuk memecahkan masalah terkait jaringan saraf. *Keras* dikembangkan untuk memungkinkan eksperimen cepat dengan jaringan saraf. Pemodelan keras mudah karena modul dasar sudah tersedia dan Anda tidak perlu menulis kode perhitungan satu per satu. Komputer yang dibuat dengan *Keras* dapat berjalan dengan baik saat menggunakan CPU atau GPU. Selanjutnya, terdapat *Scikit Learn* yang merupakan pustaka pada pemrograman *python* yang mengimplementasikan serangkaian pembelajaran mesin, pra-pemrosesan, validasi silang, dan algoritma untuk visualisasi melalui antarmuka terpadu. Manfaat *Scikit-Learn* mencakup berbagai algoritma klasifikasi, regresi, dan pengelompokan untuk masalah terkait data..

2.5 Penelitian Terkait Pengolahan Citra Uang Kertas Rupiah

Beberapa penelitian terdahulu yang berkaitan dengan klasifikasi uang kertas rupiah telah dilakukan oleh beberapa peneliti, sehingga penulis menjadikan hal tersebut sebagai acuan agar terhindar dari adanya duplikasi dalam penelitian yang akan dilakukan. Beberapa penelitian yang telah dilakukan terhadap citra uang kertas rupiah dengan menggunakan metode klasifikasi CNN dan SVM dapat dilihat pada tabel sebagai berikut.

Tabel 2.2 Perbandingan Studi Relevan

No	Judul	Objek	Data Sampel	Metode	Keterangan
1	Counterfeit Currency Detection using Deep Convolutional Neural Network (Kamble, K., 2019)	Uang Kertas Negara India	Total dataset yang digunakan sebanyak 10.000 gambar dengan bantuan proses augmentasi dari 100 gambar awal untuk setiap kelasnya yang terdiri dari ₹ 2000 asli, ₹ 2000 palsu, ₹ 500 asli dan ₹ 500 rupee palsu.	Convolutional Neural Network menggunakan model vgg16	Hasil eksperimen menunjukkan bahwa rasio pembagian data training-data validation sebesar 80% : 20% menghasilkan akurasi training 98.57% dan akurasi validation 96.55%.
2	Automated System for Indian Banknote Recognition using Image Processing and Deep Learning (Chowdhury, U.R. dkk, 2020)	Uang Kertas Negara India	Pecahan uang yang diteliti adalah ₹ 10, ₹ 20, ₹ 50, ₹ 100, ₹ 200, ₹ 500, dan ₹ 2000 yang masing berjumlah 10 gambar setiap nominalnya. Kemudian digunakan 34 gambar berbeda dengan nominal acak untuk dilakukan pengujian	k-Nearest Neighbors dan Convolutional Neural Network.	Hasil deteksi ciri uang pada metode k-NN menghasilkan akurasi sebesar 91,18% dan metode CNN menghasilkan akurasi sebesar 100%

3	Ethiopian Banknote Recognition and Fake Detection Using Support Vector Machine (Tessfaw, E. A., 2018)	Uang Kertas Negara Etiopia	Jumlah data 500 citra yang dibagi menjadi dua data set, yaitu 400 data train dan 100 data test dengan uang kertas nominal 1 etiopia birr, 5 etiopia birr, 10 etiopia birr, 50 etiopia birr, dan 100 etiopia birr	SVM dengan ekstraksi fitur LBP	Dengan pembagian data set 80% data uji dan 20% data latih dari total 500 citra menghasilkan akurasi terbaik sebesar 98%.
4	Mengenali Keaslian Mata Uang Kertas Rupiah Dengan Penerapan Metode Support Vector Machine (Sekarani, F. H., & Chamidah, N., 2020)	Uang Kertas Negara Indonesia	Citra uang kertas rupiah Tahun Emisi 2016 sejumlah 100 citra dengan nominal uang Rp 50.000,00 sebanyak 20 citra uang asli dan 24 citra uang palsu. Lalu untuk nominal uang Rp 100.000,00 menggunakan 30 citra uang asli dan 26 citra uang palsu.	SVM dengan ekstraksi fitur GLCM	Dalam proses identifikasi yang dilakukan, Sistem menghasilkan performa yang baik dengan mencapai rata-rata nilai akurasi sebesar 95%.

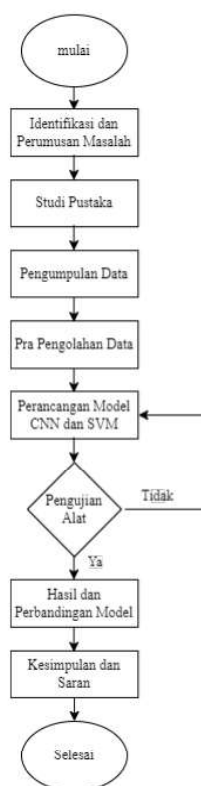
Pada penelitian kali ini saya akan melakukan perbandingan antara metode CNN dan metode SVM dalam mendeteksi keaslian dan nominal uang kertas rupiah. Perbandingan ini dilakukan dengan cara yang berbeda dari penelitian tertera pada tabel 2.2 dimana untuk mendeteksi keaslian dan nominal uang kertas rupiah, penulis akan melakukan pembuatan arsitektur CNN menggunakan skema 3 layer dengan parameter yang akan penulis tentukan tanpa menggunakan pre-trained model seperti vgg16. Kemudian untuk metode SVM, penulis akan mengubah penggunaan metode ekstraksi fitur menjadi metode seleksi fitur yaitu *Principal Component Analysis* (PCA).

BAB 3

4 METODOLOGI PENELITIAN

3.1. Kerangka Pikir

Dalam melaksanakan penelitian ini, penulis menyusun prosedur penelitian yang dituang dalam kerangka pikir pada gambar 3.1, beberapa langkah disajikan dalam diagram alir sebagai berikut:



Gambar 3.1 Diagram Alir

3.1.1. Identifikasi dan Perumusan Masalah

Pada tahap awal penelitian yang meliputi latar belakang dan identifikasi masalah yang ada di wilayah penelitian. Dalam perumusan masalah ditentukan tujuan dari masalah yang dijadikan sebagai solusi untuk memecahkan masalah, sebagai contoh besar kecilnya masalah.

3.1.2. Studi Pustaka

Tahap ini penulis mendalami dan memahami buku-buku, majalah penelitian dan literatur lainnya, seperti video tutorial, sebagai referensi materi terkait *image processing*, *SVM*, *machine learning*, *deep learning*, CNN dan wawasan lainnya yang masih memiliki keterkaitan dengan penelitian.

3.1.3. Pengumpulan Data

Prosedur pengumpulan data di dapatkan dengan melakukan pengambilan sampel citra uang kertas rupiah berupa uang rupiah asli dan uang rupiah palsu yang dicetak pada kertas HVS. Citra dari uang rupiah tersebut diambil menggunakan kamera *smartphone 12 mini* dengan bantuan fitur *night mode* dengan jarak ± 22 sentimeter diatas objek. Pengambilan citra uang dilakukan pada bagian depan dan belakang dengan dua kondisi yaitu kondisi pertama dengan bantuan 2 buah lampu penerang (*lighting*) LED sejauh ± 25 sentimeter dan 2 meter (lampu ruangan) diatas objek. Sedangkan, kondisi kedua dilakukan dengan bantuan 1 buah lampu penerang (*lighting*) ultraviolet tanpa adanya lampu ruangan. Penelitian dibagi menjadi enam kategori yaitu nominal uang Rp 20.000, Rp 50.000, Rp 100.000 asli serta nominal Rp 20.000, Rp 50.000, Rp 100.000 palsu dengan masing – masing kategori berjumlah 100 citra. Berikut disajikan tabel data citra yang telah dikumpulkan.

Tabel 3.1 Jumlah Sampel Citra

No	Kategori	Jumlah Sampel
1.	Rp 20.000 asli	100
2.	Rp 20.000 palsu	100
3.	Rp 50.000 asli	100
4.	Rp 50.000 palsu	100
5.	Rp 100.000 asli	100
6.	Rp 100.000 palsu	100
Total		600

3.1.4. Pra Pengolahan Data

Pada tahap pra pengolahan dilakukan pemrosesan awal dari dataset yang sudah dikumpulkan dan diklasifikasikan. Pemrosesan awal berupa *cropping* dan *resizing* pada setiap citra uang kertas rupiah di masing – masing kelasnya. Kedua hal ini dilakukan untuk menyesuaikan arsitektur algoritma CNN dan SVM agar kompatibel dengan citra masukan.

3.1.5. Perancangan Model CNN dan SVM

Pada tahap perancangan model, dilakukan pembuatan arsitektur pada algoritma CNN dan SVM yang akan digunakan dalam proses pelatihan dan pengujian data. Disamping itu, dataset yang ada dilakukan pembagian data uji dan data latih dengan ketentuan yang telah ditentukan oleh penulis. Pembuatan sebuah model CNN pada penelitian ini terdiri dari beberapa layer yang akan digunakan, seperti konvolusi, *max polling*, dan *fully-connected*. Kemudian dilakukan langkah pengoptimalan atau normalisasi dengan aktivasi fungsi dan optimizer serta ditentukan parameter apa saja yang dibutuhkan pada perancangan arsitektur ini. Untuk parameter yang digunakan antara lain adalah *epoch*, dan *learning rate*.

Epoch merupakan kondisi semua dataset yang sudah melalui proses training pada perancangan sistem diproses kembali mulai dari awal. Sedangkan *learning rate* digunakan untuk menghitung nilai koreksi bobot pada saat *training process*, saat nilai *learning rate* semakin besar maka proses

training akan berjalan lebih cepat, begitu juga sebaliknya.

Sedangkan untuk sebuah model SVM pada penelitian ini terdiri dari seleksi ²⁵ fitur menggunakan *Principal Component Analysis (PCA)* dan klasifikasi menggunakan metode *Support Vector Machine (SVM)*. PCA berfungsi untuk membantu model dalam mengenali citra dari setiap kelas agar saat proses komputasi, interpretasi data dapat lebih cepat. Sedangkan untuk menentukan klasifikasi menggunakan SVM, digunakan *kernel function* dan margin (disimbolkan dengan C) sebagai *hyperplane* yang berguna untuk pembatas antara setiap kelas dari input data tersebut.

3.1.6. Pengujian Alat/Sistem

Pengujian di lakukan untuk memeriksa rancangan *software* dapat berjalan dengan baik atau tidak. Pengujian pada penelitian ini ditinjau berdasarkan percobaan pelatihan model apakah model termasuk *good fit* atau mengalami error (*overfitting* atau *underfitting*). Model dapat dikategorikan *good fit* atau tidak dapat dilihat dengan menggunakan grafik pelatihan selama *training* atau *learning curve*.

3.1.7. Hasil dan Perbandingan Model

Pada tahap ini dilakukan interpresentasi hasil yang telah didapat dari proses pengolahan data dalam penelitian yang telah dilakukan. Selanjutnya dilakukan analisis dari hasil tersebut dan dibandingkan untuk masing – masing metode untuk menemukan keunggulan dari masing – masing metode dan menentukan metode yang terbaik.

¹² 3.1.8. Kesimpulan dan Saran

Pada bagian ini, menghasilkan kesimpulan dari semua temuan penelitian, yaitu temuan dari pemeriksaan perbandingan akurasi antara metode CNN dan metode SVM. Kemudian, saran dapat dibuat untuk lebih banyak peneliti untuk memperluas penelitian ini.

3.2. Perangkat Penelitian

Dalam melakukan penelitian, maka dibutuhkan perangkat keras dan perangkat lunak sebagai penunjang dilakukannya penelitian.

3.2.1. Perangkat Keras

Perangkat keras dan spesifikasi dari perangkat keras yang digunakan yaitu sebagai berikut:

1. Laptop atau Personal Computer
 - Model sistem : Macbook Air 7,1 11 inch (2015)
 - Prosesor : Intel Core I5 5250U quad-core processor (up to 3 MB L3 cache, up to 2.7 GHz)
 - Memori : 4GB DDR3
 - Storage : 512GB PCIe SSD
2. Lampu LED Ultraviolet
Merk : Gaxindo
Tipe : GX-M2048 (20 Watt)
3. Lampu LED
Merk : Hannochs
Tipe : Sonic (15 Watt)
4. Lampu LED
Merk : Hannochs
Tipe : Sonic (7 Watt)
5. Kamera Smartphone atau Kamera Digital
 - Model : Iphone 12 Mini
 - Resolusi : 12 Megapixel
 - Fitur Night Mode : Ya

3.2.2. Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan adalah sebagai berikut :

1. Sistem Operasi : Windows 10 Pro-64 Bit
2. Aplikasi yang digunakan :
 - a. Microsoft Word 2019 sebagai pengolahan dokumen,
 - b. Android IDE sebagai media penerapan model

- c. Jupyter Notebook dan Google Collaboratory sebagai pengolahan kode program *Python*,
- d. Adobe Photoshop CS6 2020 sebagai media pra-pengolahan data citra.

3.3. Timeline Pelaksanaan Penelitian

Pelaksanaan penelitian ini dilakukan secara sistematis, mulai dari ³⁹identifikasi masalah, studi literatur, pengumpulan data, hingga pengolahan dan analisis data, ⁴⁹evaluasi hasil serta penulisan laporan. Penelitian ini dilaksanakan selama 6 bulan, berikut ini merupakan *timeline* pada penelitian ini yang ditunjukkan pada Tabel 3.3. maka dibutuhkan perangkat keras dan perangkat lunak sebagai penunjang dilakukannya penelitian.

Tabel 3.2 ³³Timeline Pelaksanaan Penelitian

No	Kegiatan	Waktu					
		Jan	Feb	Mar	Apr	Mei	Juni
1	Identifikasi Masalah						
2	Studi Literatur						
3.	Pengambilan Data						
⁴⁹ 4.	Pengolahan Data dan Perancangan Sistem						
5.	Evaluasi Sistem dan Penerapan pada <i>Mobile Apps</i>						
6.	Kesimpulan Penelitian						
7.	Penulisan Laporan						

BAB 4

PEMBAHASAN DAN HASIL PENELITIAN

Tahap - tahap yang dilakukan pada penelitian ini dimulai dari pengumpulan data, pengolahan data dan perancangan sistem, pelatihan model, evaluasi model, sampai dengan *deployment* model ke *mobile apps* .

4.1 Pengumpulan Data

Tahapan awal pada penelitian ini adalah pengumpulan data citra. Data citra yang digunakan merupakan citra uang kertas rupiah bagian depan dan bagian belakang yang terdiri dari 6 kelas yaitu 20.000 Asli, 20.000 Palsu, 50.000 Asli, 50.000 Palsu, 100.000 Asli, dan 100.000 Palsu. *Dataset* tersebut diperoleh dari pengambilan citra menggunakan kamera *smartphone* 12 *mini* dengan bantuan fitur *night mode* dengan jarak ± 22 sentimeter diatas objek. Persebaran data pada tiap kelasnya ditunjukkan pada tabel 4.1

Tabel 4.1 Persebaran Data

Kelas <i>Dataset</i>	Jumlah
20.000 Asli	100
20.000 Palsu	100
50.000 Asli	100
50.000 Palsu	100
100.000 Asli	100
100.000 Palsu	100
Total	600

Berikut contoh citra uang kertas rupiah pada masing-masing kelas.



Gambar 4.1 Kelas 20.000 Asli



Gambar 4.2 Kelas 20.000 Palsu



Gambar 4.3 Kelas 50.000 Asli



Gambar 4.4 Kelas 50.000 Palsu



Gambar 4.5 Kelas 100.000 Asli



Gambar 4.6 Kelas 100.000 Palsu

4.2 Pengolahan Data dan Perancangan Sistem

Tahapan pengolahan data dan perancangan sistem ini merupakan tahapan dilakukannya pra proses data, perancangan model CNN, dan perancangan model SVM .

4.2.1 Pra Proses Data

Citra yang memiliki resolusi yang besar dan berbeda perlu disesuaikan agar menjadikan proses pengolahan data lebih cepat. Pada penelitian ini, dilakukan proses *cropping* pada ukuran panjang dan lebar citra dengan rasio 1 : 1 dan *resizing* sebesar 299×299 pixels menggunakan software Adobe Photoshop CS6. Contoh proses *cropping* dan *resizing* pada citra ditunjukkan pada gambar berikut.



Gambar 4.7 Proses *cropping* dan *resizing* citra uang kertas rupiah

4.2.2 Perancangan Model CNN

Pada penelitian ini proses utama yang harus dilakukan yaitu membuat model klasifikasi menggunakan CNN agar menghasilkan model yang mampu memberikan nilai akurasi yang terbaik. Hal pertama yang dilakukan yaitu dengan memasukkan alamat data dan membaca jumlah data yang telah dibagi setiap kelasnya menggunakan *Google Colaboratory* sebagai media pengolahan pemrograman *python*. Data yang terbaca sudah dibagi menjadi data latih dan data uji dengan rasio 80% : 20%. Data yang terbaca untuk masing – masing enam kelas yaitu 80 citra data latih dan 20 citra data uji.

```

set_1 = os.path.join('/content/drive/My Drive/Uang/train/20.000_asli')
set_2 = os.path.join('/content/drive/My Drive/Uang/train/20.000_palsu')
set_3 = os.path.join('/content/drive/My Drive/Uang/train/50.000-asli')
set_4 = os.path.join('/content/drive/My Drive/Uang/train/50.000_palsu')
set_5 = os.path.join('/content/drive/My Drive/Uang/train/100.000_asli')
set_6 = os.path.join('/content/drive/My Drive/Uang/train/100.000_palsu')

set_7 = os.path.join('/content/drive/My Drive/Uang/test/20.000_asli')
set_8 = os.path.join('/content/drive/My Drive/Uang/test/20.000_palsu')
set_9 = os.path.join('/content/drive/My Drive/Uang/test/50.000_asli')
set_10 = os.path.join('/content/drive/My Drive/Uang/test/50.000_palsu')
set_11 = os.path.join('/content/drive/My Drive/Uang/test/100.000_asli')
set_12 = os.path.join('/content/drive/My Drive/Uang/test/100.000_palsu')

print ('total data train uang 20.000 asli:', len(os.listdir(set_1)))
print ('total data train uang 20.000 palsu:', len(os.listdir(set_2)))
print ('total data train uang 50.000 asli:', len(os.listdir(set_3)))
print ('total data train uang 50.000 palsu:', len(os.listdir(set_4)))
print ('total data train uang 100.000 asli:', len(os.listdir(set_5)))
print ('total data train uang 100.000 palsu:', len(os.listdir(set_6)))
#train test
print('\n')
print ('total data test uang 20.000 asli:', len(os.listdir(set_7)))
print ('total data test uang 20.000 palsu:', len(os.listdir(set_8)))
print ('total data test uang 50.000 asli:', len(os.listdir(set_9)))
print ('total data test uang 50.000 palsu:', len(os.listdir(set_10)))
print ('total data test uang 100.000 asli:', len(os.listdir(set_11)))
print ('total data test uang 100.000 palsu:', len(os.listdir(set_12)))

```

```

total data train uang 20.000 asli: 80
total data train uang 20.000 palsu: 80
total data train uang 50.000 asli: 80
total data train uang 50.000 palsu: 80
total data train uang 100.000 asli: 80
total data train uang 100.000 palsu: 80

total data test uang 20.000 asli: 20
total data test uang 20.000 palsu: 20
total data test uang 50.000 asli: 20
total data test uang 50.000 palsu: 20
total data test uang 100.000 asli: 20
total data test uang 100.000 palsu: 20

```

29

Selanjutnya dilakukan proses *rescale* untuk membagi nilai RGB dari 0-255 dengan 255, sehingga didapatkan nilai RGB di rentan 0-1. Lalu, Pada *statement* pemrograman “train_generator” dan “valid_generator” digunakan sebagai proses pembangkitan data berdasarkan citra data latih dan citra data validasi. Kode pemrograman tersebut merubah data berupa *raw image* menjadi dataset yang terdiri dari 480 citra data latih pada 6 kelas dan 120 citra data uji pada 6 kelas untuk digunakan pada proses pembuatan model.


```

from keras.preprocessing.image import ImageDataGenerator

train_datagen = ImageDataGenerator(rescale=1./255
)
valid_datagen = ImageDataGenerator(rescale=1./255
)

train_generator = train_datagen.flow_from_directory('/content/drive/My Drive/Uang/train',
batch_size = 32,
target_size=(299,299),
class_mode="categorical",
seed=46)
validation_generator = valid_datagen.flow_from_directory('/content/drive/My Drive/Uang/test',
batch_size = 32,
target_size=(299,299),
class_mode="categorical",
seed=46)

```

```

Found 480 images belonging to 6 classes.
Found 120 images belonging to 6 classes.

```

Untuk mengetahui urutan kelas yang terbaca oleh mesin, digunakan kode pemrograman "label_map". Urutan citra uang rupiah yang terbaca yaitu kelas "100.000_Aslis" dengan label "0", kelas "100.000_Palsu" dengan label "1", "20.000_Aslis" dengan label "2" "20.000_Palsu" dengan label "3", "50.000_Aslis" dengan label "4", "50.000_Palsu" dengan label "5".

```

label_map=(train_generator.class_indices)
print(label_map)

{'100.000_aslis': 0, '100.000_palsu': 1, '20.000_aslis': 2, '20.000_palsu': 3, '50.000-aslis': 4, '50.000_palsu': 5}

```

Dalam pembuatan model, digunakan *Library* "Tensorflow" yang merupakan sebuah *framework* yang digunakan untuk melakukan penyelesaian masalah *machine learning* dan *deep learning*. *Library* "Keras" merupakan *library* yang biasa digunakan untuk permasalahan *deep learning* seperti CNN atau *Recurrent Neural Network* (RNN).

```

import tensorflow as tf
from tensorflow.keras import layers
from tensorflow.keras import regularizers

model = tf.keras.models.Sequential([
#konvolusi pertama
tf.keras.layers.Conv2D(16, (3,3), activation='relu', padding='same', kernel_initializer='he_normal', input_shape=(299, 299, 3)),
tf.keras.layers.MaxPooling2D(2, 2),
#konvolusi kedua
tf.keras.layers.Conv2D(32, (3,3), activation='relu', padding='same', kernel_initializer='he_normal'), tf.keras.layers.MaxPooling2D(2,2),
#konvolusi ketiga
tf.keras.layers.Conv2D(64, (3,3), activation='relu', padding='same', kernel_initializer='he_normal'), tf.keras.layers.MaxPooling2D(2,2),
tf.keras.layers.Conv2D(64, (3,3), activation='relu', padding='same', kernel_initializer='he_normal'), tf.keras.layers.MaxPooling2D(2,2),
tf.keras.layers.Conv2D(64, (3,3), activation='relu', padding='same', kernel_initializer='he_normal'), tf.keras.layers.MaxPooling2D(2,2),
#latten dropout
tf.keras.layers.Flatten(),
tf.keras.layers.Dropout(0.3),
#fully connected layers
tf.keras.layers.Dense(128, activity_regularizer=regularizers.l2(0.001), activation='relu'),
tf.keras.layers.Dropout(0.5),
tf.keras.layers.Dense(6, activation='softmax')
])

```

Dalam Model CNN, arsitektur dapat diatur dan dibuat dengan beberapa layer berbeda-beda yang terdiri dari *convolution layer*, *pooling layer*, *flatten layer*, dan *fully-connected layer*. Pada penelitian ini, arsitektur CNN yang dibuat digunakan 3 proses konvolusi dan 3 proses pooling. Konvolusi pertama menggunakan filter sebanyak 16 dengan ukuran kernel sebesar 3x3, fungsi aktivasi yang digunakan dalam konvolusi ini yaitu menggunakan ReLu. Pada penelitian ini digunakan maxpolling dengan ukuran kernel 2x2 mengambil nilai terbesar dari bagian pada area tertentu gambar sehingga menciptakan gambar baru. Pada proses konvolusi kedua menggunakan filter sebanyak 32 karena jumlah masukan pada layer kedua akan semakin kecil sehingga dibutuhkan banyak filter dalam mengekstrak informasi citra. Selanjutnya tetap menggunakan aktivasi ReLu, serta ukuran kernel dan maxpolling yang sama seperti konvolusi pertama. Proses konvolusi ketiga filter ditambah yaitu sebanyak 64, proses maxpolling masih sama seperti konvolusi sebelumnya dengan kernel polling sebesar 2x2, fungsi aktivasi tetap menggunakan ReLu.

Selanjutnya dilakukan *flatten* dan *dropout*. *Flatten* layer terhadap feature maps mengubah nilai masukan menjadi sebuah *array* hasil *polling* yang dilakukan untuk proses *Fully-connected* layer menghasilkan klasifikasi dari citra. *Dropout* Layer digunakan untuk mencegah terjadinya *overfitting* sehingga dapat mempercepat proses learning, pada *dropout* neuron dipilih secara acak dan tidak digunakan selama proses pelatihan, menghilangkan suatu neuron dan menghilangkan sementara jaringan yang ada. *Dense* layer berfungsi untuk menambahkan layer pada *fully-connected* layer. Terdapat *regularizer* yang berfungsi untuk membantu dalam menghindari error selama proses pelatihan model dan fungsi aktivasi *softmax classifier* sebagai proses klasifikasi karena memberikan hasil yang lebih baik serta lebih perseptif yang membantu dalam proses klasifikasi dalam banyak kelas.

```

model.summary()
Model: "sequential"
-----
Layer (type)                Output Shape                Param #
-----
conv2d (Conv2D)             (None, 299, 299, 16)       448
max_pooling2d (MaxPooling2D) (None, 149, 149, 16)       0
conv2d_1 (Conv2D)           (None, 149, 149, 32)       4640
max_pooling2d_1 (MaxPooling2D) (None, 74, 74, 32)         0
conv2d_2 (Conv2D)           (None, 74, 74, 64)         18496
max_pooling2d_2 (MaxPooling2D) (None, 37, 37, 64)         0
conv2d_3 (Conv2D)           (None, 37, 37, 64)         36928
max_pooling2d_3 (MaxPooling2D) (None, 18, 18, 64)         0
conv2d_4 (Conv2D)           (None, 18, 18, 64)         36928
max_pooling2d_4 (MaxPooling2D) (None, 9, 9, 64)           0
Flatten (Flatten)           (None, 5184)                0
dropout (Dropout)           (None, 5184)                0
dense (Dense)                (None, 128)                 663680
dropout_1 (Dropout)          (None, 128)                 0
dense_1 (Dense)              (None, 6)                   774
-----
Total params: 761,894
Trainable params: 761,894
Non-trainable params: 0

```

Dengan menggunakan kode “model.summary()”, maka akan menampilkan uraian tersebut dari model CNN yang telah dibuat. Jumlah parameter yang dilatih pada model ini sebanyak 761,894 parameter. Ukuran citra pada setiap layernya semakin berkurang, sebelum terakhir masuk ke *fully-connected* layer adalah 9x9 piksel dan jumlah filter yang digunakan sebanyak 64. Lalu dilakukan *reshape* menghasilkan 5184 neuron yang akan masuk pada *fully-connected* layer. Pada *hidden* layer, jumlah neuron yang digunakan sebanyak 512 neuron, dan pada akhirnya klasifikasi yang dilakukan sebanyak 6 kategori.

```

from tensorflow.keras.optimizers import Adam
adam = Adam(learning_rate = 0.001)

#compile
model.compile(loss = 'categorical_crossentropy', optimizer= tf.optimizers.Adam(), metrics=['accuracy'])

history=model.fit(
    train_generator, steps_per_epoch=3, epochs=100, validation_data=validation_generator, validation_steps=3
)

```

55 Dengan kode seperti diatas, dilakukan proses *compile* model dengan menggunakan *optimizer* Adam, *learning rate* sebesar 0.001, *loss function* menggunakan *categorical_crossentropy*, dan *metrics* menggunakan *accuracy*. Kemudian pada proses training model menggunakan data training yang ada pada *training_generator* dan data validation dari *validation_generator*, dengan epoch sebanyak 100 dan step_per_epoch sebanyak 3 step. Hasil 7 dari proses training adalah sebagai berikut.

```

Epoch 1/100
3/3 [-----] - 22s 8s/step - loss: 2.6999 - accuracy: 0.1771 - val_loss: 1.8002 - val_accuracy: 0.1562
Epoch 100/100
3/3 [-----] - 17s 6s/step - loss: 0.1497 - accuracy: 1.0000 - val_loss: 0.1561 - val_accuracy: 0.9688

```

Gambar 4.8 Hasil proses training model CNN

4.2.3 Perancangan Model SVM

Pada penelitian ini proses utama yang harus dilakukan yaitu membuat model klasifikasi menggunakan algoritma SVM. Hal pertama yang dilakukan yaitu dengan memasukkan alamat data untuk membaca data citra yang akan digunakan melalui *Google Colaboratory* sebagai media pengolahan pemrograman *python*. Terdapat kode pemrograman “classes” sebagai pernyataan urutan kelas citra uang rupiah yaitu kelas “100.000_Aslis” dengan label “0”, kelas “100.000_Palsu” dengan label “1”, “20.000_Aslis” dengan label “2” “20.000_Palsu” dengan label “3”, “50.000_Aslis” dengan label “4”, “50.000_Palsu” dengan label “5”.

```

import os

path = os.listdir('/content/drive/MyDrive/Data_uang/ready')
classes = ('100.000_Aslis':0, '100.000_Palsu':1, '20.000_Aslis':2, '20.000_Palsu':3, '50.000_Aslis':4, '50.000_Palsu':5,)

```

Kode dibawah ini digunakan menyatakan bahwa setiap kelas yang terbaca telah berhasil di simpikasi agar dapat digunakan pada proses selanjutnya.

```

flat_data_arr=[]
target_arr=[]
#please use datadir='/content' if the files are upload on to google collab
#else mount the drive and give path of the parent-folder containing all category images folders.
datadir='/content/drive/MyDrive/Data_uang/ready'
for i in classes:
    print(f'loading... category : {i}')
    path=os.path.join(datadir,i)
    for img in os.listdir(path):
        img_array=imread(os.path.join(path,img))
        img_resized=resize(img_array,(299,299,3))
        flat_data_arr.append(img_resized.flatten())
        target_arr.append(classes[i])
    print(f'loaded category:{i} successfully')

```

Selanjutnya data diubah menjadi bentuk array dan dinyatakan dalam x dan y. selanjutnya dilakukan proses pemisahan data train dan data test dengan rasio 80% data latih dan 20% data uji pada setiap kelasnya.

```

flat_data=np.array(flat_data_arr)
target=np.array(target_arr)
df=pd.DataFrame(flat_data)
df['Target']=target
x=df.iloc[:, :-1]
y=df.iloc[:, -1]
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=42,stratify=y)
print('Splitted Successfully')

```

Setelah dibagi data uji dan data latih, dilakukan *feature scaling* menggunakan *minmax scaler*, dimana *minmax scaler* mengubah data dari rentan 0 - 255 menjadi 0 - 1 agar data mengolah data. Kemudian dilakukan metode *dimensionality reduction* menggunakan metode *Principle Component Analysis* (PCA) agar mempermudah proses pembelajaran mesin dalam interpretasi data.

```

x_train = x_train/255
x_test = x_test/255
print(x_train.max(), x_train.min())
print(x_test.max(), x_test.min())

from sklearn.decomposition import PCA

print(x_train.shape, x_test.shape)

pca = PCA(.98)

pca_train = x_train
pca_test = x_test

```

Model yang sudah dilakukan *feature selection* dilanjutkan dengan proses training menggunakan algoritma SVM seperti kode berikut.

Hyperparameter pada metode SVM digunakan secara default. Artinya, nilai C sebesar 1 dan kernel yang digunakan adalah rbf. Pernyataan “.fit” digunakan untuk melakukan pelatihan dataset menggunakan model SVM.

```
from sklearn.svm import SVC
```

```
sv = SVC()  
sv.fit(x_train, y_train)
```

18

4.3 Evaluasi Sistem

Pada tahapan ini dilakukan Evaluasi terhadap model CNN dan model SVM yang telah dibuat. Lalu, hasil dari model terbaik antara kedua algoritma akan diterapkan pada aplikasi berdasarkan nilai *accuracy* tertinggi.

4.3.1 Evaluasi Model CNN

Setelah melakukan proses training, model yang sudah terlatih pada proses training dievaluasi menggunakan validation data untuk mendapat nilai akhir *accuracy*. Untuk melakukan evaluasi model terhadap data test set menggunakan kode sebagai berikut :

```
model.evaluate(validation_generator)
```

Kemudian didapatkan hasil output seperti gambar 4.9, terlihat didapatkan hasil evaluasi model dengan nilai loss 13.71% dan *accuracy* sebesar 98.33%.

```
4/4 [=====] - 5s 1s/step - loss: 0.1371 - accuracy: 0.9833  
[0.1370515376329422, 0.9833333492279053]
```

Gambar 4.9 Hasil Evaluasi Model

Kode dibawah merupakan kode untuk menampilkan grafik menampilkan akurasi dari model yang telah dipilih, grafik menunjukkan seberapa besar akurasi pada model selama dilatih.

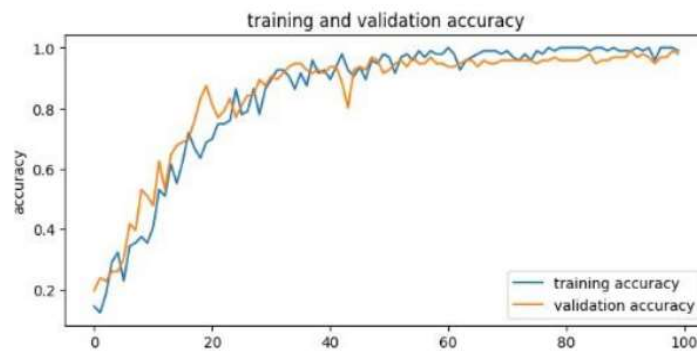
```

import matplotlib.pyplot as plt
acc = history.history['accuracy']
val_acc = history.history['val_accuracy']

plt.figure(figsize=(8, 8))
plt.subplot(2, 1, 1)
plt.plot(acc, label='training accuracy')
plt.plot(val_acc, label='validation accuracy')
plt.legend(loc='lower right')
plt.ylabel('accuracy')
plt.title('training and validation accuracy')
plt.show()

```

Gambar 4.10 menunjukkan grafik accuracy selama proses training dan validation berlangsung. Dilihat dari gambar grafik tersebut, model sudah good fit, tidak mengalami overfitting ataupun underfitting. Artinya, model dapat berjalan dengan baik tanpa error yang signifikan.



Gambar 4.10 Grafik Pelatihan Model CNN

Selanjutnya dilakukan prediksi untuk mendapatkan hasil pengujian model dari confusion matrix menggunakan validasi data. Pada kode dibawah ini, label yang digunakan merupakan kelas dari dataset yang telah ditentukan sebelumnya yaitu, label 0 untuk kelas “100.000 Asli”. Label 1 untuk kelas “100.000 Palsu”. Label 2 untuk kelas “20.000 Asli”. Label 3 untuk kelas “20.000 Palsu”. Label 4 untuk kelas “50.000 Asli”. Label 5 untuk kelas “50.000 Palsu”.

```

import sklearn
from sklearn.metrics import confusion_matrix
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns

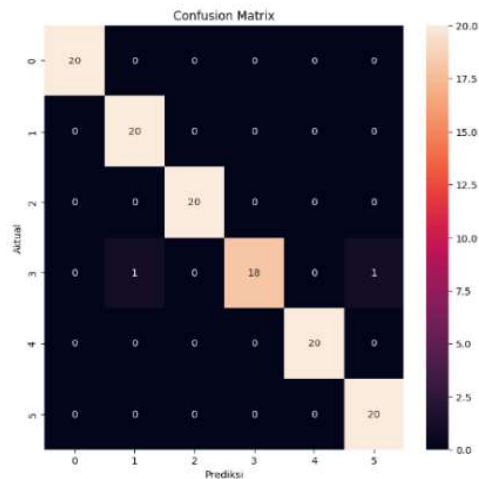
for x in label_map:
    print(label_map[x],":",x)

prediksi=model.predict_generator(test_data_generator)
y_pred=np.argmax(prediksi,axis=1)

plt.figure(figsize=(8,8))
sns.heatmap(confusion_matrix(test_data_generator.classes,y_pred),annot=True)
plt.title("Confusion Matrix")
plt.ylabel("Aktual")
plt.xlabel("Prediksi")

```

Pada gambar 4.11 menunjukkan bahwa dari total 120 data validasi yang tersebar pada 6 kelas, terdapat 1 kelas yang mengalami error dalam proses prediksi kelas terhadap actual kelas.



Gambar 4.11 *Confusion Matrix* pada Model CNN

18
4.3.2 Evaluasi Model SVM

Pada tahapan **evaluasi model**, model yang telah berhasil dilakukan training dinyatakan menggunakan kode di bawah ini untuk mendapat hasil score dari proses training yang sudah berlangsung.

```

print("Training Score:", sv.score(x_train, y_train))
print("Testing Score:", sv.score(x_test, y_test))

Training Score: 0.9729166666666667
Testing Score: 0.9666666666666667

```

Sesuai dengan output pemrograman diatas, nilai training yang didapat setelah diubah ke bentuk persen sebesar 97% dan nilai validasi sebesar 96.7%. Lalu, untuk memastikan proses training berjalan dengan baik dan tidak terdapat error, maka dibuat sebuah learning curve. Model yang baik apabila kurva training dan kurva validasi memiliki jarak yang berdekatan. Kode program dibawah ini digunakan untuk membuat learning curve.

```

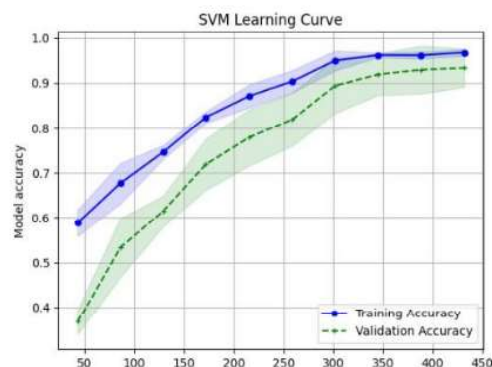
from sklearn.model_selection import learning_curve

train_sizes, train_scores, test_scores = learning_curve(estimator=sv, X=x_train, y=y_train,
                                                       cv=10, train_sizes=np.linspace(0.1, 1.0, 10),
                                                       n_jobs=1)

plt.plot(train_sizes, train_mean, color='blue', marker='o', markersize=5, label='Training Accuracy')
plt.fill_between(train_sizes, train_mean + train_std, train_mean - train_std, alpha=0.15, color='blue')
plt.plot(train_sizes, test_mean, color='green', marker='s', markersize=5, linestyle='--', label='Validation Accuracy')
plt.fill_between(train_sizes, test_mean + test_std, test_mean - test_std, alpha=0.15, color='green')
plt.title('SVM Learning Curve')
plt.xlabel('Number of Subset Samples')
plt.ylabel('Model accuracy')
plt.grid()
plt.legend(loc='lower right')
plt.show()

```

Pada gambar 4.12, kurva training dan kurva validasi memiliki jarak yang semakin lama semakin berdekatan. Artinya, proses pelatihan model berjalan dengan baik dan tidak terjadi *overfitting* maupun *underfitting*.



Gambar 4.12 Learning Curve pada model SVM

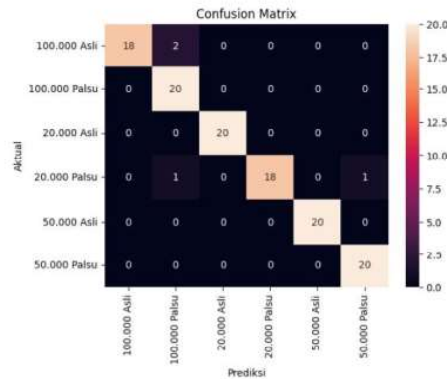
Kemudian, dilakukan pembuatan *confusion matrix* untuk melihat performa model SVM terhadap validasi data menggunakan kode pemrograman dibawah beserta label kelasnya.

```

cm = confusion_matrix(y_test, y_pred)
in.heatmap(cm, annot=True, xticklabels=["100.000 Asli", "100.000 Palsu", "20.000 Asli", "20.000 Palsu", "50.000 Asli", "50.000 Palsu"],
yticklabels=["100.000 Asli", "100.000 Palsu", "20.000 Asli", "20.000 Palsu", "50.000 Asli", "50.000 Palsu"])

```

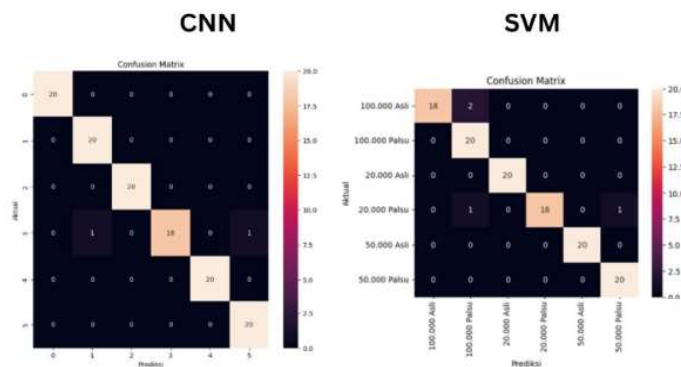
Pada gambar 4.12, menunjukkan bahwa dari total 120 data validasi yang tersebar pada 6 kelas, terdapat 2 kelas yang mengalami error dalam proses prediksi kelas terhadap actual kelas.



Gambar 4.13 confusion matrix model SVM

4.3.3 Penentuan Model Terbaik

Model dari dua percobaan algoritma CNN dan SVM memiliki performa akurasi yang tidak jauh berbeda. Berdasarkan hasil pelatihan model, masing – masing algoritma mendapatkan nilai accuracy sebesar 98.33% untuk CNN dan 96.67% untuk algoritma SVM. Dari Gambar 4.13 yang menunjukkan confusion matrix CNN dan SVM menampilkan bahwa terdapat selisih 1 kelas error dimana actual kelas dengan label 0 yaitu “100.000 Asli” dapat diprediksi sempurna oleh CNN. Oleh sebab itu nilai accuracy model SVM dapat lebih unggul dibandingkan dengan nilai accuracy model CNN.



Gambar 4.14 confusion matrix model CNN dan SVM

4.4 Penerapan Model pada Aplikasi

Pengimplementasian penelitian ini yaitu pada pembuatan mobile apps yang dapat digunakan pada sistem mobile berbasis android, model terbaik yang diambil pada penelitian ini yaitu model CNN dengan nilai accuracy sebesar 98.33% menggunakan software **Android Studio**.

4.4.1 Tampilan Awal

Tampilan awal ini merupakan pembuka ketika awal membuka aplikasi. Tampilan ini akan muncul beberapa detik lalu melanjutkan ke tampilan berikutnya. Berikut merupakan tampilan awal pada aplikasi.



Gambar 4.15 Tampilan awal pada aplikasi

4.4.2 Tampilan Utama

Tampilan utama merupakan halaman yang terbuka setelah tampilan awal pada aplikasi. Pada tampilan ini digunakan sistem klasifikasi menggunakan model terbaik yaitu model CNN yang telah dibuat sebelumnya. Untuk melakukan unggah gambar dapat menggunakan ambil gambar menggunakan kamera atau menggunakan gambar yang terdapat pada gallery. Dibawah ini merupakan tampilan utama yang telah dibuat



Gambar 4.16 Tampilan utama pada aplikasi

Apabila gambar berhasil diunggah, maka akan menampilkan hasil prediksi dan menampilkan gambar yang diunggah pada aplikasi seperti pada gambar dibawah ini.



Gambar 4.17 Tampilan hasil klasifikasi pada aplikasi

Dalam beberapa kali percobaan seperti contoh pada gambar 4.18, dapat disimpulkan bahwa pendeteksian uang dengan berbagai latar belakang citra yang berbeda dapat membuat sistem kurang maksimal dalam melakukan klasifikasi. Hal ini dapat dikarenakan minimnya variasi dari input dataset maupun minimnya pengembangan pra-pengolahan data yang menyebabkan sistem hanya mengenali data yang berlatar identik.



Gambar 4.18 Hasil Percobaan Klasifikasi pada *Mobile Apps*

5.1 Kesimpulan

Berdasarkan hasil pembahasan pada penelitian ini yang berjudul analisis performansi metode *Convolutional Neural Network* (CNN) dan *Support Vector Machine* (SVM) dalam mendeteksi keaslian dan nominal citra uang kertas rupiah emisi tahun 2022, maka dapat disimpulkan :

1. Sistem klasifikasi citra uang kertas rupiah menggunakan metode *Convolutional Neural Network* (CNN) dan *Support Vector Machine* (SVM) mampu menghasilkan model yang baik. Nilai accuracy yang didapat sebesar 98.33% untuk algoritma CNN dan 96.67% untuk algoritma SVM. Dalam hal pengolahan citra uang kertas rupiah, algoritma CNN memiliki nilai accuracy yang lebih unggul.
2. Pengimplementasian model dalam mendeteksi keaslian dan nominal uang kertas rupiah pada suatu *mobile apps* dapat digunakan namun kurang maksimal. Hal ini dikarenakan minimnya variasi dari input dataset maupun minimnya pengembangan pra-pengolahan data yang menyebabkan sistem hanya mengenali data yang berlatar identik

5.2 Saran

Setelah dilakukan penelitian ini, penelitian yang telah dibuat ini masih dapat dilakukan pengembangan lebih lanjut, maka dapat diberikan saran sebagai berikut :

1. Pada penelitian selanjutnya dapat menggunakan data citra/ sampel yang lebih banyak.
2. Modifikasi pada proses pra-pengolahan data dalam membantu memaksimalkan keakuratan pada sistem deteksi citra uang kertas rupiah contohnya seperti augmentasi data, perubahan ke *grayscale*, dan lain sebagainya.
3. Pengimplementasian model pada *mobile apps* dapat dibuat pada mobile berbasis IOS dengan tampilan yang memiliki banyak menu dan menarik.
4. Melakukan proses *deployment* yang lebih baik seperti membuat *real-time image recognition* untuk penerapan model lebih lanjut.

DAFTAR PUSTAKA

- [1] E. Ayalew Tessfaw, B. Ramani, and T. Kebede Bahiru, "Ethiopian banknote recognition and fake detection using support vector machine," *2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)*, 2018.
- [2] Adhao, R. and Pachghare, V. (2020) 'Feature selection using principal component analysis and genetic algorithm', *Journal of Discrete Mathematical Sciences and Cryptography*, 23(2), pp. 595–602. doi:10.1080/09720529.2020.1729507.
- [3] Fadilah, N.I., Rahayudi, B. and Furqon, M.T. (no date) *Implementasi algoritme support vector machine (SVM) Untuk Klasifikasi Penyakit Dengan Gejala demam, Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*. Available at: <https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/3366> (Accessed: 27 June 2023).
- [4] Pulung Nurtantio Andono, T.Sutojo, Muljono (2017) *Pengolahan Citra Digital, Google Books*. Available at: https://books.google.co.id/books?hl=en&lr=&id=zUJRDwAAQBAJ&oi=fnd&pg=PR3&dq=Andono%2C%2BP.%2BN.%2C%2B%26%2BSutojo%2C%2BT.%2B%282017%29.%2BPengolahan%2Bcitra%2Bdigital.%2BPenerbit%2BAndi.&ots=CiKIPcAU3N&sig=z2cWA439RPibFlzMcvCeHB2bLF4&redir_esc=y#v=onepage&q&f=false (Accessed: 27 June 2023).
- [5] Ndaru Aji Laksono (2020) *Ekstraksi Ciri local binary pattern (LBP) Dan Gray Level co-occurrence matrix (GLCM) pada KASUS motif batik yogyakarta dengan PENGKLASIFIKASI support vector machine (SVM)*, *Repository UPN Veteran Jakarta*. Available at: <https://repository.upnvj.ac.id/7127/> (Accessed: 27 June 2023).
- [6] Kamble, K. *et al.* (2019) 'Counterfeit currency detection using deep convolutional neural network', *2019 IEEE Pune Section International Conference (PuneCon)*

- [7] Kamble, K. *et al.* (2019) 'Counterfeit currency detection using deep convolutional neural network', *2019 IEEE Pune Section International Conference (PuneCon)* [Preprint].
- [8] Manaswi, N.K. (2018) 'Understanding and working with keras', *Deep Learning with Applications Using Python*, pp. 31–43. doi:10.1007/978-1-4842-3516-4_2.
- [9] Muhammad Alif Raihan (2021) *Pengenalan wajah dengan METODE convolutional neural network (CNN) Pada Citra Wajah Bermasker*, Repository UPN Veteran Jakarta. Available at: <https://repository.upnvj.ac.id/12015/> (Accessed: 27 June 2023).
- [10] Sekarani, F.H., Jayanta and Chamidah, N. (no date) *Mengenal Keaslian Mata uang kertas rupiah Dengan Penerapan metode support Vector Machine, Prosiding Seminar Nasional Mahasiswa Bidang Ilmu Komputer dan Aplikasinya*. Available at: <https://conference.upnvj.ac.id/index.php/senamika/article/view/629> (Accessed: 27 June 2023).
- [11] Sulistiyanti, S.R., Setyawan, F.X.A. and Komarudin, M. (1970) *Pengolahan Citra, Dasar Dan Contoh penerapannya*, University of Lampung. Available at: <http://repository.lppm.unila.ac.id/2976/> (Accessed: 27 June 2023).
- [12] Adawiyah, S.R. (2021) *Analisis Perbandingan kinerja metode canny Dan Fuzzy logic Dalam Deteksi Keaslian Mata Uang rupiah Kertas berdasarkan watermark, Go to start page!* Available at: <http://repository.unhas.ac.id/id/eprint/7079/> (Accessed: 27 June 2023).
- [13] Mounika, K. *et al.* (2021) 'Hyperspectral image classification using SVM with PCA', *2021 6th International Conference on Signal Processing, Computing and Control (ISPCC)*
- [14] Solikin, S. (2017) *Uang*, Google Books. Available at: https://books.google.co.id/books?hl=en&lr=&id=WaPWDgAAQBAJ&oi=fnd&pg=PA1&dq=Bank%2C%2BP.%2BP.%2BD.%2BS.%2BK.%2BP.%2B%282017%29.%2BUang%3A%2BPengertian%2C%2Bpenciptaan%2Bdan%2Bperanannya%2Bdalam%2Bperekonomian%2B%28Vol.%2B1%29.%2BPusat%2BPendidikan%2BDan%2BStudi%2BKebanksentralan%2B%28PPSK%29%2BBank%2BIndonesia.&ots=DLhrnKtflm&sig=tW0-B_b04a6_2ImHGRrQXmIGWX0&redir_esc=y#v=onepage&q&f=false (Accessed: 27 June 2023).
- [15] Wijayanto, A.A. (no date) *Pemalsuan Mata Uang sebagai kejahatan di Indonesia*, *Jurnal Hukum Khaira Ummah*. Available at: <https://jurnal.unissula.ac.id/index.php/jhku/article/view/2306> (Accessed: 27 June 2023).

- [17] Dhany Umar, "Diagnosis awal Pada Penyakit alzheimer menggunakan metode VGG-19 convolutional neural network (CNN) Berdasarkan Citra MRI dari Otak Manusia," Repository UPN Veteran Jakarta, <https://repository.upnvj.ac.id/22383/> (accessed Jun. 27, 2023).
- [18] Teris Ekamila Wahyundari Putri, "Penerapan deep learning Untuk Klasifikasi kesegaran daging SAPI Berbasis Mobile Apps," Repository UPN Veteran Jakarta, <https://repository.upnvj.ac.id/22097/> (accessed Jun. 27, 2023).
- [19] N. Silaparasetty, "Machine learning concepts with python and the jupyter notebook environment," SpringerLink, <https://link.springer.com/book/10.1007/978-1-4842-5967-2> (accessed Jun. 27, 2023).
- [20] B. Admin, "(artikel) Kenali FITUR Keamanan Pada uang rupiah emisi 2016," Generasi Baru Indonesia Kepulauan Riau - Komisariat Universitas Maritim Raja Ali Haji, <https://genbikepri.wordpress.com/2019/10/11/artikel-apa-saja-sih-fitur-keamanan-pada-uang-rupiah-emisi-2016/> (accessed Jun. 27, 2023).
- [21] Merdeka, "Intip, Hasil Kecanggihan teknologi terbaru Dalam Uang Baru 2022," merdeka.com, <https://www.merdeka.com/uang/intip-hasil-kecanggihan-teknologi-terbaru-dalam-uang-baru-2022.html> (accessed Jun. 27, 2023).
- [22] K. Relan, "Deploying flask applications," SpringerLink, https://link.springer.com/chapter/10.1007/978-1-4842-5022-8_6 (accessed Jun. 27, 2023).
- [23] Y. G. Indonesia, "Mengenal Seluk Beluk Uang," Google Books, https://books.google.com/books/about/Mengenal_Seluk_Beluk_Uang.html?id=eB3pLAmXLycC (accessed Jun. 27, 2023).
- [24] O. Kramer, "Scikit-Learn," SpringerLink, https://link.springer.com/chapter/10.1007/978-3-319-33383-0_5 (accessed Jul. 3, 2023).

ANALISIS PERFORMANSI METODE CONVOLUTIONAL NEURAL NETWORK (CNN) DAN SUPPORT VECTOR MACHINE (SVM) DALAM MENDETEKSI KEASLIAN DAN NOMINAL CITRA UANG KERTAS RUPIAH

ORIGINALITY REPORT

21 %
SIMILARITY INDEX

19 %
INTERNET SOURCES

6 %
PUBLICATIONS

5 %
STUDENT PAPERS

PRIMARY SOURCES

1	repository.unhas.ac.id Internet Source	3 %
2	dspace.uii.ac.id Internet Source	1 %
3	publikasi.mercubuana.ac.id Internet Source	1 %
4	docplayer.info Internet Source	1 %
5	repository.unsoed.ac.id Internet Source	1 %
6	www.researchgate.net Internet Source	1 %
7	repository.its.ac.id Internet Source	1 %
8	medium.com Internet Source	1 %

9	Submitted to Saint Leo University Student Paper	1 %
10	core.ac.uk Internet Source	1 %
11	repository.upnvj.ac.id Internet Source	1 %
12	repository.uin-suska.ac.id Internet Source	<1 %
13	conference.upnvj.ac.id Internet Source	<1 %
14	repository.ub.ac.id Internet Source	<1 %
15	text-id.123dok.com Internet Source	<1 %
16	Submitted to Universitas Trunojoyo Student Paper	<1 %
17	repositori.uin-alauddin.ac.id Internet Source	<1 %
18	Rian Kurniawan, Puput Budi Wintoro, Yessi Mulyani, Muhamad Komarudin. "IMPLEMENTASI ARSITEKTUR XCEPTION PADA MODEL MACHINE LEARNING KLASIFIKASI SAMPAH ANORGANIK", Jurnal Informatika dan Teknik Elektro Terapan, 2023 Publication	<1 %

19	Reno Supardi. "IMPLEMENTASI METODE BIT MATCHING UNTUK KEAMANAN PESAN TEKS MENGGUNAKAN VISUAL BASIC. NET", JURNAL TEKNOLOGI INFORMASI, 2018 Publication	<1 %
20	jst.publikasiindonesia.id Internet Source	<1 %
21	pt.scribd.com Internet Source	<1 %
22	Submitted to Syiah Kuala University Student Paper	<1 %
23	m.tribunnews.com Internet Source	<1 %
24	Submitted to Universitas Negeri Surabaya The State University of Surabaya Student Paper	<1 %
25	id.123dok.com Internet Source	<1 %
26	www.scribd.com Internet Source	<1 %
27	Afis Julianto, Andi Sunyoto, Ferry Wahyu Wibowo. "OPTIMASI HYPERPARAMETER CONVOLUTIONAL NEURAL NETWORK UNTUK KLASIFIKASI PENYAKIT TANAMAN PADI", TEKNIMEDIA: Teknologi Informasi dan Multimedia, 2022	<1 %

28	Submitted to Fakultas Ekonomi Universitas Indonesia Student Paper	<1 %
29	Submitted to Sriwijaya University Student Paper	<1 %
30	digilib.unila.ac.id Internet Source	<1 %
31	e-repository.perpus.iainsalatiga.ac.id Internet Source	<1 %
32	repository.unej.ac.id Internet Source	<1 %
33	Submitted to Politeknik Ilmu Pelayaran Semarang Student Paper	<1 %
34	Submitted to Universitas Brawijaya Student Paper	<1 %
35	Submitted to Universitas Islam Indonesia Student Paper	<1 %
36	news.bsi.ac.id Internet Source	<1 %
37	laptopworld.vn Internet Source	<1 %
38	ojs.unm.ac.id Internet Source	<1 %

39 repository.urecol.org <1 %
Internet Source

40 adoc.pub <1 %
Internet Source

41 ilkom.unila.ac.id <1 %
Internet Source

42 ojs.unimal.ac.id <1 %
Internet Source

43 pemrogramanmatlab.com <1 %
Internet Source

44 digilib.uinsby.ac.id <1 %
Internet Source

45 etd.repository.ugm.ac.id <1 %
Internet Source

46 etheses.uin-malang.ac.id <1 %
Internet Source

47 jurnal.untan.ac.id <1 %
Internet Source

48 ojs.uajy.ac.id <1 %
Internet Source

49 repository.unpkediri.ac.id <1 %
Internet Source

50 tsetyaernawati.wordpress.com <1 %
Internet Source

51

ecampus.pelitabangsa.ac.id

Internet Source

<1 %

52

eprints.walisongo.ac.id

Internet Source

<1 %

53

es.scribd.com

Internet Source

<1 %

54

repo.itera.ac.id

Internet Source

<1 %

55

repository.upi.edu

Internet Source

<1 %

56

stay-control.xyz

Internet Source

<1 %

Exclude quotes On

Exclude bibliography On

Exclude matches Off

