

Implementasi Algoritma Convolutional Neural Network Untuk Identifikasi Kode Seri Keramik Merk Arwana

Alfito Herdiansyah¹, Arief Hermawan², Sutarman³

^{1,2,3}Informatika, Fakultas Sains dan Teknologi, Universitas Teknologi Yogyakarta

¹hrdsh6@gmail.com *, ²ariefdb@uty.ac.id, ³sutarman@uty.ac.id

Abstract

Ceramic tiles currently have various types and motifs that might be distinguished, one of the options is by looking at the ceramic series code. When installing ceramics, sometimes many things happen that do not go according to plan. Generally, consumers often face obstacles when buying ceramics with motifs or colours that do not match because they forget to check the series code. The main objective of the research conducted is to create a system that is able to identify ceramic series codes using a convolutional neural network as its method and to determine the level of accuracy produced in its application. This study adapts the Convolutional Neural Network method to identify ceramic series code based on digital images. Classification is carried out using a digital image dataset obtained using a smartphone camera of 1000 images with 10 different classes. As a comparison, the dataset, namely 60% trainer data and 40% test data, produces the best accuracy of 96.17% on training data and 91.75% on test data.

Keywords: accuracy, ceramic, convolutional neural network, image, series code.

Abstrak

Ubin keramik saat ini memiliki berbagai jenis dan motif yang dapat dibedakan, salah satunya dengan melihat kode seri keramik. Saat melakukan pemasangan keramik, terkadang banyak hal yang terjadi tidak sesuai rencana. Umumnya konsumen sering menghadapi kendala saat membeli keramik dengan motif atau warna yang tidak sesuai karena lupa mengecek kode serinya. Tujuan utama dari penelitian yang dilakukan adalah menciptakan sebuah sistem yang mampu mengidentifikasi kode seri keramik menggunakan *Convolutional Neural Network* sebagai metodenya dan mengetahui tingkat akurasi yang dihasilkan dalam penerapannya. Penelitian ini mengadaptasi metode *Convolutional Neural Network* untuk mengidentifikasi kode seri keramik berdasarkan citra digital. Klasifikasi dilakukan dengan menggunakan *dataset* citra digital yang didapatkan menggunakan kamera *smartphone* sebanyak 1000 citra dengan 10 kelas yang berbeda. Sebagai pembandingan, dataset yaitu 60% data pelatih dan 40% data pengujian menghasilkan akurasi terbaik sebesar 96,17% pada data latih dan 91,75% pada data uji.

Kata kunci: akurasi, citra, *convolutional neural network*, keramik, kode seri.

©This work is licensed under a Creative Commons Attribution -ShareAlike 4.0 International License

1. Pendahuluan

Keramik merupakan satu dari banyaknya bahan yang digunakan dalam pembangunan dan dekorasi, baik untuk interior maupun eksterior [1]. Dengan perkembangan desain dan teknologi, keramik kini tersedia dalam berbagai jenis, motif, dan warna, sehingga konsumen dapat memilih produk yang cocok dengan kebutuhan estetika dan fungsional. Namun, dalam proses pemasangan keramik, seringkali terjadi kendala yang tidak sesuai rencana, seperti keramik pecah saat pemasangan.

Salah satu masalah utama yang dihadapi konsumen adalah ketika mereka harus membeli keramik yang sama dengan yang digunakan sebelumnya. Seringkali, konsumen lupa untuk memperhatikan kode seri keramik yang tertera pada kemasan atau produk itu sendiri. Ketidapahaman ini dapat mengakibatkan kesalahan saat pembelian ulang, di mana motif atau warna keramik baru tidak sesuai dengan yang diharapkan. Jika konsumen membeli keramik yang tidak cocok dengan produk lama, mereka mungkin harus membeli keramik baru lagi, yang tidak hanya lebih mahal, tetapi juga memerlukan biaya tambahan

untuk pemasangan ulang. Bahkan, jika keramik yang salah sudah terlanjur dipasang, konsumen bisa saja harus mengganti keramik yang sudah terpasang, yang tentunya menambah waktu dan biaya renovasi. Di tengah kemajuan teknologi saat ini, konsumen semakin menginginkan proses pembelian yang lebih cepat, mudah, dan bebas dari kesalahan. Salah satu solusi potensial untuk masalah ini adalah sistem identifikasi berbasis citra. Dengan sistem ini, konsumen dapat dengan mudah mengidentifikasi keramik yang mereka butuhkan hanya dengan memotret atau mengunggah gambar dari produk yang ada.

Dalam konteks identifikasi keramik, pemilihan metode yang tepat sangat penting untuk memastikan akurasi dan efisiensi sistem. Penggunaan *Machine Learning*, khususnya *Convolutional Neural Network* (CNN), merupakan teknologi yang tepat dan paling efisien untuk tugas ini [2]. Pembelajaran mesin adalah model statistik dan aplikasi komputer yang digunakan untuk belajar dari data dan membuat prediksi [3]. CNN dirancang untuk dapat mengolah data berbentuk citra [4]. CNN memiliki kemampuan untuk menangani data dengan kompleksitas tinggi, seperti gambar keramik

yang memiliki variasi warna, pola, dan desain. Melalui serangkaian lapisan konvolusional, CNN dapat mendeteksi fitur-fitur dasar seperti tepi, bentuk, dan warna, kemudian menggabungkan informasi tersebut untuk mengenali objek secara keseluruhan. Kemampuan ini menjadikan CNN sangat relevan untuk tugas identifikasi citra keramik, di mana tingkat akurasi dan ketelitian sangat penting agar konsumen dapat menemukan produk keramik yang sesuai dengan yang telah digunakan sebelumnya. Dengan CNN, sistem dapat mempelajari dan mengenali ribuan gambar keramik dari berbagai sudut pandang dan kondisi pencahayaan, meningkatkan kemampuan sistem untuk memberikan rekomendasi yang lebih tepat. Hal ini akan sangat berguna bagi konsumen yang mencari kesesuaian produk berdasarkan citra yang mereka miliki, memungkinkan proses pembelian yang lebih cepat, akurat, dan bebas dari kesalahan.

Beberapa peneliti telah melakukan penelitian menggunakan CNN dalam pengklasifikasian dan identifikasi objek. Penelitian yang dilakukan oleh Lutfi Hakim [5] mengungkapkan, mereka menggunakan dataset sebanyak 1024 foto dengan tujuh jenis motif batik Banyuwangi yang berbeda yaitu Moto pitik, Kopibesar, Sisikan, Gedegan, Paras gempal, dan Gajah oling. Algoritma CNN digunakan untuk klasifikasi motif batik Banyuwangi. Sebanyak 20% data uji digunakan pada proses prediksi, sedangkan 80% data latih digunakan pada proses pelatihan. Model arsitektur CNN yang disarankan mencapai nilai kinerja akurasi 63%.

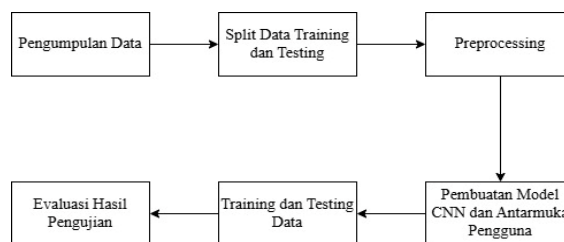
Penelitian lainnya dilakukan oleh Muhammad Rahman [6] mengungkapkan, mereka menggunakan dataset sebanyak 361 foto selada dengan empat jenis yang berbeda yaitu selada sehat, selada berpenyakit jamur, selada defisiensi nutrisi dan selada busuk. Dalam penelitian ini, algoritma CNN digunakan untuk mengukur kualitas tanaman. Pembagian rasio dataset pada penelitian ini 80:10:10 dengan masing masing merupakan *training*, *validation*, dan *testing*. Hasil akurasi terbaik menggunakan model arsitektur CNN yang disarankan yaitu mencapai 90% pada proses training dan pada data testing mencapai 84%. Penelitian-penelitian yang relevan inilah yang melandasi dilakukannya penelitian untuk identifikasi kode seri keramik serta mengembangkannya menjadi sistem yang dapat digunakan oleh konsumen.

Penelitian ini dimaksudkan untuk menangani kesalahan dalam pembelian keramik dengan mengimplementasikan algoritma *Convolutional Neural Network* (CNN) serta mengembangkannya menjadi aplikasi berbasis web yang dapat mengidentifikasi kode seri keramik merk Arwana berdasarkan citra digital. Dengan menggunakan metode ini, diharapkan dapat tercapai akurasi yang tinggi dalam pengenalan kode seri, sehingga memudahkan konsumen dalam pengambilan keputusan pembelian keramik. Hasil penelitian ini dapat digunakan secara luas dalam industri keramik untuk

meningkatkan pengalaman konsumen, mengurangi kesalahan pembelian, dan mempercepat proses renovasi atau pembangunan. Selain itu, sistem ini berpotensi diadaptasi untuk merek keramik lainnya, menjadikannya solusi universal dalam membantu konsumen memilih produk yang sesuai.

2. Metode Penelitian

Gambar 1 menunjukkan tahapan proses yang dikerjakan pada penelitian ini untuk menciptakan sistem yang dapat mengidentifikasi kode seri keramik berdasarkan citra digital.



Gambar 1. Tahapan Penelitian

2.1. Pengumpulan data

Citra atau gambar adalah data yang digunakan pada penelitian ini. Data yang digunakan pada penelitian ini yaitu berupa citra keramik. Data didapatkan dengan membeli sampel keramik sesuai dengan masing-masing kode seri keramik yang ada pada katalog. Pemilihan keramik dilakukan secara acak berdasarkan kode seri yang terdaftar dalam katalog produk sebanyak 10 kode seri keramik.

Selanjutnya dilakukan pengambilan citra keramik menggunakan kamera smartphone dengan resolusi 12 MP. Citra diambil dalam kondisi pencahayaan alami yang cukup, di ruang dengan pencahayaan yang konsisten untuk menghindari bayangan yang dapat mempengaruhi hasil pengenalan gambar. Setiap citra diambil dari sudut pandang yang berbeda, termasuk pandangan atas dan samping, untuk memastikan variasi sudut pandang yang cukup. Sudut pengambilan gambar dilakukan secara acak pada setiap produk keramik, dengan jarak pengambilan yang konsisten untuk memperoleh detail yang serupa pada setiap citra.

Data yang diperoleh dibagi menjadi 10 kelas yang setiap kelasnya berjumlah 100 data, lalu setiap kelas dibagi menjadi data pelatihan dan pengujian dengan perbandingan 60:40, sehingga jumlah keseluruhan dataset yang diperoleh yaitu sebanyak 1000 data citra. Tabel 1 menunjukkan jumlah data yang dikumpulkan untuk penelitian ini.

Tabel 1. Tabel *Software* dan *Hardware* Pendukung

No	Class	Jumlah	
		Data Training	Data Testing
1	AR2251BL	60	40
2	AR2251PK	60	40
3	AR2261BL	60	40
4	AR2281LB	60	40
5	AR2321BL	60	40

6	AR2321GN	60	40
7	AR2671BL	60	40
8	AR2671MN	60	40
9	AR2821BG	60	40
10	AR2821PK	60	40

Sampel data dari masing-masing kelas keramik yang berjumlah 10 kelas dapat dilihat pada gambar 2.



Gambar 2. Sampel Dataset

2.2. Preprocessing

Proses ini bertujuan untuk mempersiapkan gambar sebelum digunakan dalam model pembelajaran mesin. Pada tahap *preprocessing*, langkah pertama yang dilakukan adalah *cropping* pada *dataset*. Proses *cropping* dilakukan menggunakan *website roboflow*. *Cropping* dapat membantu menyingkirkan bagian yang tidak diperlukan, sehingga model hanya mempelajari fitur dari objek yang ingin dikenali [7]. Proses *cropping* pada penelitian ini memotong bagian tepi gambar dengan area objek utama sebagai fokus.

Setelah itu akan dilakukan proses segmentasi berupa menghapus latar belakang (*background*) dari gambar dan mengganti latar belakang menjadi warna hitam. Penghapusan latar belakang dilakukan menggunakan pustaka Python, yaitu *Library rembg*. Dalam konteks pembelajaran mesin, segmentasi membantu model untuk memahami struktur dan hubungan dalam gambar dengan lebih baik [8].

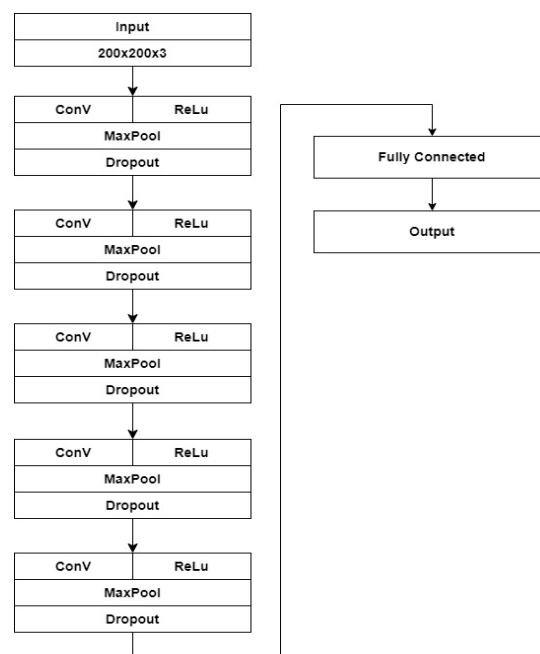
Setelah gambar input melalui proses *cropping* dan segmentasi, langkah terakhir yaitu dilakukan *resize*. Proses *cropping* menyebabkan ukuran gambar input menjadi bervariasi. Proses *resize* bertujuan untuk menjamin bahwa semua gambar memiliki dimensi yang seragam [9]. Pada proses terakhir ini gambar di *resize* menjadi 200x200 piksel. Ukuran ini dirasa cukup efisien pada beban komputasi.

2.3. Pembuatan Model

Dalam konteks pembelajaran mendalam, *Convolutional Neural Network* merupakan salah satu bentuk jaringan syaraf tiruan [10]. Jaringan Syaraf Konvolusional (*Convolutional Neural Network*) bekerja mirip dengan Jaringan Syaraf Buatan (*Artificial Neural Network*) konvensional [11]. Arsitektur CNN yang dibangun menggunakan framework PyTorch,

PyTorch menyediakan modul-modul yang digunakan untuk mendefinisikan lapisan jaringan. Secara umum, arsitektur CNN tersusun dari serangkaian tahapan atau lapisan [12].

Pada lapisan pertama, terdapat dua lapisan utama, yaitu *convolutional layer* (lapisan konvolusi) dan lapisan *pooling* (lapisan penyatuan) [13]. Lapisan konvolusi merupakan lapisan pada CNN yang berfungsi untuk memproses masukan yang diterimanya menggunakan operasi konvolusi. Setelah menerima masukan berupa gambar, lapisan konvolusi menerapkan filter atau kernel yang telah ditentukan sebelumnya ke masukan untuk melakukan operasi konvolusi [14]. Kemudian akan berpindah ke lapisan berikutnya untuk pemrosesan tambahan berdasarkan keluaran operasi konvolusi. Gambar 3 menunjukkan Ilustrasi model atau arsitektur CNN yang diterapkan dalam penelitian ini.



Gambar 3. Arsitektur CNN

Pada arsitektur CNN yang telah dirancang, terdapat 4 lapisan konvolusi yang bertujuan untuk mengambil fitur dari citra *input*. Setiap lapisan konvolusi menggunakan ukuran filter 3x3 yang dianggap efisien dalam banyak arsitektur CNN. Setiap lapisan konvolusi diikuti oleh fungsi aktivasi ReLU (*Rectified Linear Unit*), operasi *max pooling*, dan *dropout*. Penggunaan ReLU untuk menyajikan non-linearitas ke dalam jaringan, yang memungkinkan model mempelajari representasi data input yang lebih kompleks dan ReLU juga memiliki kemampuan untuk mempercepat konvergensi dalam pelatihan model [15].

Kemudian *max pooling* digunakan untuk mengurangi dimensi atau representasi fitur dan tetap mempertahankan informasi penting [16]. Lapisan *max pooling* dengan ukuran kernel 2x2 digunakan untuk mengurangi ukuran spasial (dimensi citra) dan memfokuskan pada fitur yang lebih dominan. Operasi

konvolusi sampai *pooling* akan mengumpulkan fitur atau informasi penting yang terdeteksi [17].

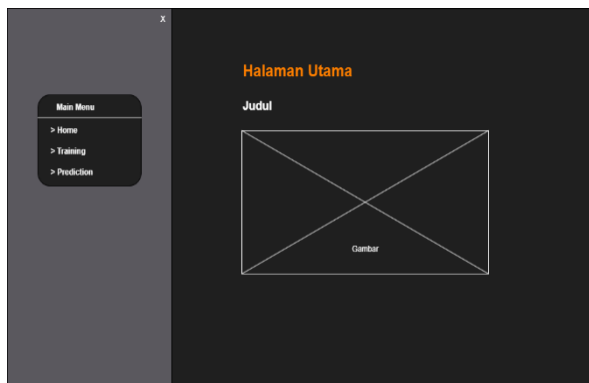
Selanjutnya penggunaan *dropout* bertujuan untuk mencegah terjadinya *overfitting* pada model. Untuk mencegah *overfitting*, setiap lapisan konvolusi dan lapisan *fully connected* diikuti dengan *dropout* dengan tingkat probabilitas 0.095. Fungsi *dropout* bekerja dengan mengabaikan beberapa *neuron* secara acak selama proses pelatihan. Selanjutnya, terdapat lapisan yang terhubung penuh (Linear), yang bertindak sebagai pengklasifikasi. Output dari lapisan konvolusi diubah menjadi bentuk vektor dengan dimensi yang sesuai sebelum disajikan ke lapisan ini. Lapisan ini memiliki *dropout* untuk mengendalikan *overfitting* [18]. *Dropout* merupakan metode regularisasi yang digunakan untuk mengatasi *overfitting* dengan tujuan meningkatkan akurasi pada pengujian, meskipun mungkin ada pengorbanan pada akurasi pelatihan [19].

2.4. Perancangan Antarmuka

Perancangan antarmuka pengguna pada penelitian ini menggunakan aplikasi Figma untuk merancang tampilan antarmuka. Antarmuka yang dirancang terdiri dari tiga halaman.

a. Rancangan Halaman Utama

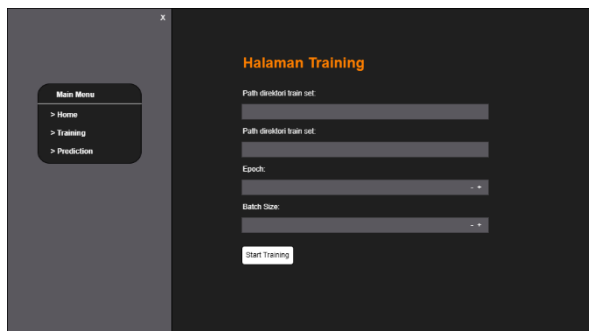
Gambar 4 merupakan rancangan dari halaman utama. Pada halaman utama terdapat bilah navigasi yang memungkinkan pengguna untuk berpindah ke halaman lain.



Gambar 4. Rancangan Halaman Utama

b. Rancangan Halaman Training

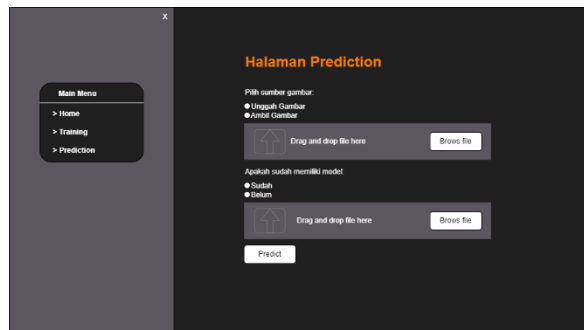
Rancangan halaman *training* disajikan pada gambar 5. Pada halaman *training* akan digunakan untuk proses *training* menggunakan model CNN.



Gambar 5. Rancangan Halaman Training

c. Rancangan Halaman Prediction

Halaman terakhir yaitu halaman prediction seperti yang disajikan pada gambar 6. Halaman ini digunakan untuk memprediksi gambar keramik baru.



Gambar 6. Rancangan Halaman Prediction

2.5. Evaluasi

Pada tahap ini, *Confusion Matrix* digunakan untuk evaluasi temuan pengujian. Matriks penilaian yang disebut matriks konfusi mengontraskan nilai aktual dengan temuan kategorisasi [20]. Matriks *f1-score*, akurasi, presisi, dan *recall* digunakan dalam penelitian ini.

Terdapat beberapa rumus untuk masing-masing nilai tersebut, yaitu:

Perhitungan akurasi:

$$\text{Akurasi} = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

Perhitungan precision:

$$\text{Precision} = \frac{TP}{TP+FP} \quad (2)$$

Perhitungan recall:

$$\text{Recall} = \frac{TP}{TP+FN} \quad (3)$$

3. Hasil dan Pembahasan

Pada tahap ini berkaitan dengan hasil implementasi antarmuka pengguna dan hasil pengujian sistem.

3.1. Implementasi Antarmuka Pengguna

Implementasi web bertujuan untuk membangun sistem untuk membantu pengguna dalam melakukan identifikasi kode seri keramik. Implementasi situs web ini berupa tahap pengembangan sistem dengan menggunakan bahasa pemrograman Python, *framework* PyTorch dan *framework* Streamlit. Pada penelitian ini penulis membuat tiga halaman pada antarmuka pengguna yaitu halaman utama, halaman training, dan halaman prediction. Fitur utama terletak pada halaman training dan halaman prediction, yang masing-masing digunakan untuk proses pelatihan dan prediksi.

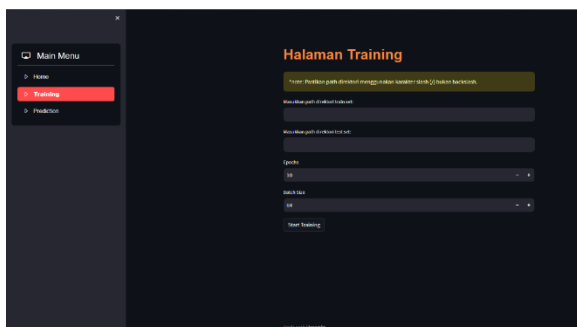
Pertama halaman utama, halaman utama ini merupakan

halaman pertama yang tampil pada antarmuka dan berfungsi untuk menyambut pengguna. Halaman ini mencakup bilah navigasi pada sebelah kiri halaman, yang memungkinkan pengguna menavigasi ke halaman lain. Bilah navigasi menggunakan elemen “option_menu()” yaitu komponen dari pustaka *streamlit-option-menu*. Pengguna dapat berinteraksi dengan sistem melalui bilah navigasi untuk memilih halaman yang diinginkan. Pengguna hanya perlu menekan tombol pada *Main Menu*. Gambar 7 menyajikan tampilan halaman utama.



Gambar 7. Halaman Utama

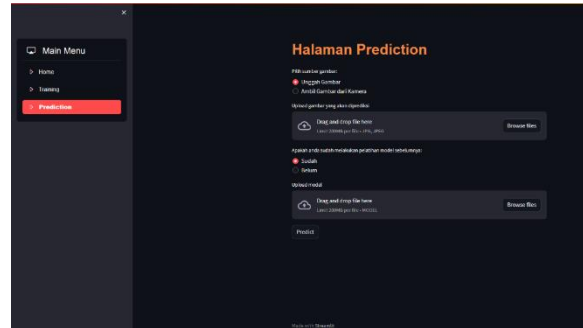
Kedua, halaman training pada gambar 8 merupakan halaman yang berfungsi untuk melakukan *training* pada dataset citra menggunakan algoritma CNN. Pada tampilan *training* ini terdapat beberapa kolom *input* seperti, *input directory*, *input epoch*, *input batch size* dan tombol “*start training*”. Kolom *input* menggunakan elemen “*st.text_input()*” dan “*st.number_input()*”. Kolom *input* pada halaman ini berfungsi untuk memasukkan direktori dataset yang akan digunakan dan penyetelan *hyperparameter*. Tombol “*start training*” pada halaman ini berfungsi untuk memulai proses *training* dataset citra menggunakan algoritma CNN. Tombol disini menggunakan elemen “*st.button()*”. Jika pengguna ingin memulai proses pelatihan, pengguna cukup dengan mengisi kolom *input* dan menekan tombol “*start training*”.



Gambar 8. Halaman Training

Ketiga, halaman *prediction* pada gambar 9 merupakan halaman yang berfungsi untuk melakukan prediksi atau identifikasi terhadap citra baru yang belum pernah digunakan pada proses *training*. Pada halaman ini pengguna dapat memilih metode input yaitu dengan memilih file atau melalui kamera yang tersedia. Untuk input melalui file, menggunakan elemen “*st.file_uploader()*” sedangkan input melalui kamera

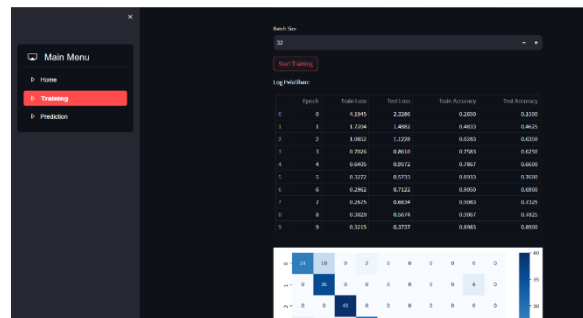
menggunakan elemen “*st.camera_input()*”. Selanjutnya pengguna akan diminta untuk memilih file model dari hasil *training* sebelumnya yang sudah disimpan pada penyimpanan lokal pengguna untuk digunakan pada proses prediksi. Proses prediksi akan berjalan setelah pengguna menekan tombol “*predict*”.



Gambar 9. Halaman Prediksi

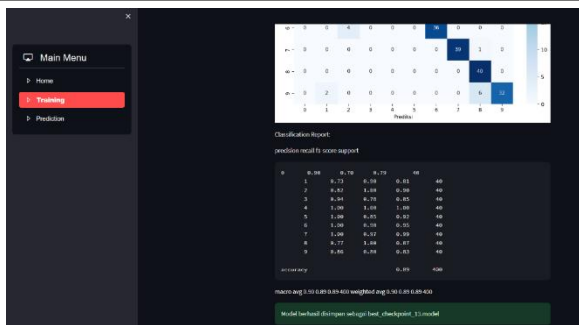
3.2. Hasil

Temuan dari penelitian ini dimaksudkan untuk menciptakan suatu sistem yang bisa memudahkan pengguna untuk mengidentifikasi kode seri keramik. Penelitian yang dilakukan penulis akan menerapkan *Convolutional Neural Network* (CNN) sebagai metodenya. CNN pada penelitian ini berfungsi sebagai pembuatan model yang akan ditanamkan ke dalam sistem. Uji coba dalam penelitian ini menggunakan dataset citra digital sebanyak 1000 citra dengan 10 kelas yang berbeda dengan perbandingan dataset yaitu 60% data *train* dan data *test* 40%. Hasil klasifikasi yang diperoleh dari penelitian ini bertujuan untuk mencari parameter yang paling efisien yang dapat digunakan untuk mengidentifikasi kode seri keramik.



Gambar 10. Halaman Proses Training

Gambar 10 memberikan gambaran hasil dari proses pelatihan menggunakan CNN. Hasil dari proses pelatihan pada halaman pelatihan akan memunculkan log hasil pelatihan, *confusion matrix*, dan laporan klasifikasi berupa metrik evaluasi. Gambar 11 menunjukkan hasil laporan klasifikasi.



Gambar 11. Halaman Hasil Training

Selanjutnya pada halaman *prediction* model dapat memprediksi gambar keramik baru dengan baik. Setelah tombol “*Predict*” ditekan, maka sistem dapat memprosesnya dengan sangat baik. Gambar 12 menunjukkan sistem yang telah dibuat dapat mengidentifikasi gambar keramik baru yang diinputkan oleh pengguna dan dapat menampilkan kode seri keramik berdasarkan gambar keramik yang diinputkan dengan benar.



Gambar 12. Halaman Hasil Prediction

Pada penelitian ini, eksperimen dilakukan pada proses pelatihan (*training*) dengan parameter epoch 10, 20, dan 30, serta rasio perbandingan dataset 60:40. Parameter yang digunakan pada proses pelatihan adalah jumlah *batch size* 32 dan menggunakan optimizer *Stochastic Gradient Descent* (SGD) dengan kecepatan belajar (*learning rate*) 0.009. Hal ini dilakukan untuk menentukan jumlah epoch yang tepat dan memastikan bahwa model dilatih dengan baik, tanpa *overfitting* (terlalu menyesuaikan data pelatihan) atau *underfitting* (tidak cukup belajar dari data).

Tabel 2. Tabel Perbandingan Jumlah Epoch

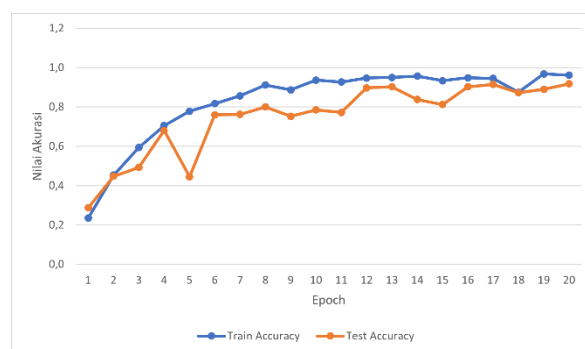
Epoch	Loss Training	Loss Testing	Akurasi Training	Akurasi Testing
10	0.3215	0.3737	89,83	89,00
20	0.1189	0.2658	96,17	91,75
30	0.0810	0.4336	96,83	81,00

Tabel 2 menunjukkan perbandingan nilai *loss* dan akurasi berdasarkan jumlah *epoch* yang berbeda. Pada *epoch* 10, model menunjukkan hasil yang cukup baik dengan *loss training* sebesar 0.3215 dan *loss testing* 0.3737. Akurasi pada data pelatihan mencapai 89,83%, sementara akurasi pada data pengujian adalah 89,00%, yang menunjukkan kinerja yang sebanding antara pelatihan dan pengujian.

Pada *epoch* 20, terjadi peningkatan yang signifikan. *Loss training* menurun menjadi 0.1189, dan *loss testing* juga berkurang menjadi 0.2658, menunjukkan bahwa model lebih terlatih dan lebih baik dalam memprediksi data pengujian. Akurasi pada data pelatihan meningkat menjadi 96,17%, dan akurasi pada data pengujian juga naik menjadi 91,75%, menunjukkan bahwa model lebih dapat menggeneralisasi dengan baik di luar data pelatihan.

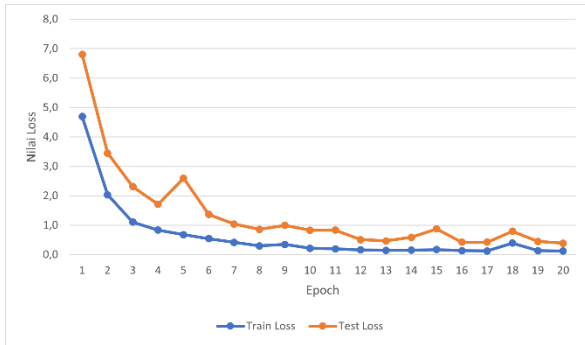
Namun, pada *epoch* 30, meskipun *loss training* terus menurun menjadi 0.0810, *loss testing* justru meningkat menjadi 0.4336, yang menunjukkan bahwa model mulai mengalami *overfitting*, di mana model terlalu terlatih pada data pelatihan dan tidak dapat menggeneralisasi dengan baik pada data pengujian. Akurasi pada data pelatihan meningkat sedikit menjadi 96,83%, namun akurasi pada data pengujian turun drastis menjadi 81,00%, menandakan penurunan kinerja model pada data yang belum pernah dilihat sebelumnya.

Dari perbandingan jumlah *epoch*, hasil terbaik didapatkan pada *epoch* 20, dengan rasio perbandingan dataset 60:40. Parameter yang digunakan pada proses pelatihan adalah jumlah *batch size* 32 dan menggunakan optimizer *Stochastic Gradient Descent* (SGD) dengan kecepatan belajar (*learning rate*) 0.009. Hal ini menunjukkan bahwa parameter *epoch* berpengaruh pada hasil performa model. Jumlah *epoch* yang terlalu sedikit dapat menyebabkan *underfitting* sedangkan jumlah *epoch* yang terlalu banyak dapat menyebabkan *overfitting*. Dapat dilihat pada gambar 13, grafik akurasi pada data *training* dan data *testing* mengalami peningkatan dan mendekati stabil seiring bertambahnya *epoch*. Selisih antara akurasi pada data *training* dan *testing* cukup kecil.



Gambar 13. Grafik Akurasi Terbaik

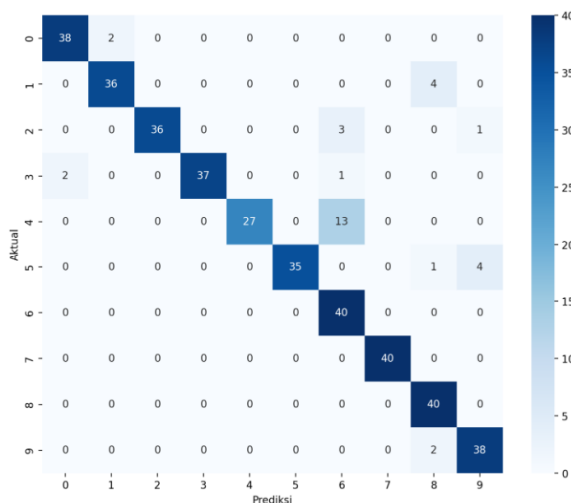
Gambar 14 juga menunjukkan penurunan nilai *loss* dan mendekati stabil seiring bertambahnya *epoch*. Selisih antara nilai *loss* pada data *training* dan *testing* juga cukup kecil dan nilai *loss* pada akhir pelatihan tidak terlalu tinggi. Kedua grafik tersebut menunjukkan bahwa pelatihan tidak mengalami *overfitting* maupun *underfitting*.



Gambar 14. Grafik Loss Terbaik

Selanjutnya untuk evaluasi model menggunakan *confusion matrix* dapat dilihat pada gambar 15. *Confusion matrix* membandingkan hasil prediksi dengan label sebenarnya untuk memberikan informasi komprehensif tentang kinerja model klasifikasi. Dapat dilihat bahwa untuk kelas 6, 7, dan 8, nilai True Positive (TP) adalah 40, yang menunjukkan bahwa ketiga kelas tersebut memiliki performa terbaik. Kelas 0, 1, 2, 3, 4, 5, dan 9 menghasilkan nilai True Positive (TP) lebih dari 35, yang berarti model dapat mengklasifikasikan contoh dengan cukup baik.

Kelas 4 menunjukkan kinerja yang lebih buruk karena memiliki banyak False Negatives (FN), yaitu 13, yang menunjukkan kelas 4 memiliki performa terburuk dan model kesulitan dengan data pada kelas ini. Perbedaan performa antara kelas-kelas dalam *confusion matrix* ini disebabkan oleh beberapa hal. Pertama, beberapa kelas mungkin memiliki visual yang mirip, contohnya dari segi warna, membuat model kesulitan untuk membedakan mereka. Kedua, beberapa kelas mungkin lebih sulit untuk dipelajari karena model CNN yang digunakan belum cukup optimal untuk menangkap fitur-fitur yang membedakan kelas tersebut.



Gambar 15. Confusion Matrix 20 epoch

Gambar 16 menunjukkan metrik evaluasi pada jumlah *epoch* 20. Dapat dilihat bahwa model memiliki akurasi keseluruhan 92% pada dataset uji, yang menunjukkan bahwa model sangat baik dalam mengklasifikasikan data secara keseluruhan. Model menunjukkan performa

yang sangat baik di sebagian besar kelas, dengan precision dan recall yang sangat tinggi untuk sebagian besar kelas. Namun, kelas 4 menunjukkan hasil yang lebih buruk dengan recall yang sangat rendah (0.68), meskipun precision tetap tinggi. Ini menunjukkan bahwa model kesulitan untuk mengidentifikasi contoh kelas 4 meskipun prediksi yang salah (False Positive) relatif rendah. Secara keseluruhan, F1-score menunjukkan performa yang baik di seluruh kelas, dan model menunjukkan keseimbangan yang cukup baik antara precision dan recall, dengan accuracy keseluruhan 92%.

0	0.95	0.95	0.95	40
1	0.95	0.90	0.92	40
2	1.00	0.90	0.95	40
3	1.00	0.93	0.96	40
4	1.00	0.68	0.81	40
5	1.00	0.88	0.93	40
6	0.70	1.00	0.82	40
7	1.00	1.00	1.00	40
8	0.85	1.00	0.92	40
9	0.88	0.95	0.92	40
accuracy	0.92			400

Gambar 16. Metrik Evaluasi 10 epoch

Secara keseluruhan, baik pada data pelatihan maupun data uji, nilai *loss* menurun dan cukup stabil pada epoch 20, yang menandakan model belajar dengan lebih baik dan bisa memprediksi lebih akurat setelah penambahan jumlah *epoch*. Perbedaan akurasi dari kedua pengujian juga menunjukkan tidak adanya tanda-tanda *overfitting* yang parah. Namun pada beberapa kelas model sulit untuk mengidentifikasi.

3.3. Pembahasan

Peningkatan jumlah epoch menyebabkan akurasi meningkat dan nilai prediksi menjadi lebih baik. Namun, jumlah epoch yang terlalu banyak dapat menyebabkan *overfitting*. Parameter yang digunakan dalam proses pelatihan adalah batch size 32, optimizer Stochastic Gradient Descent (SGD) dengan kecepatan belajar (learning rate) 0.009, dan jumlah epoch atau iterasi sebanyak 20. Menggunakan 1000 data citra, didapatkan hasil akurasi terbaik yaitu sebesar 96,17% pada data pelatihan dan menghasilkan akurasi sebesar 91,75% pada data pengujian.

Pengurangan loss dan peningkatan akurasi, baik pada data pelatihan maupun data pengujian menunjukkan bahwa model mengalami perbaikan yang signifikan ketika diberikan lebih banyak epoch untuk pelatihan. Perbedaan akurasi antara data pelatihan dan data pengujian juga menunjukkan bahwa model ini tidak mengalami *overfitting* yang parah. Namun, meskipun ada perbaikan signifikan, penggunaan lebih banyak epoch berisiko menyebabkan model menjadi terlalu cocok dengan data pelatihan, yang akan berdampak negatif pada kinerja model pada data uji yang belum pernah dilihat sebelumnya.

Jika dibandingkan dengan penelitian sebelumnya yang dilakukan oleh Rendi Setya Nugraha [21] menggunakan CNN untuk klasifikasi citra buah apel hijau, yang berhasil mencapai akurasi terbaik sebesar 96,88% pada data pelatihan dan 98,44% pada data validasi menggunakan 400 data citra. Hasil pada penelitian ini menunjukkan akurasi yang sedikit lebih rendah, baik pada data pelatihan maupun data pengujian. Namun, perbedaan ini dapat disebabkan oleh beberapa faktor, seperti perbedaan dalam jumlah data yang digunakan, teknik augmentasi yang diterapkan, atau variasi dalam pengaturan hyperparameter. Dalam penelitian buah apel, augmentasi data yang dilakukan telah meningkatkan kemampuan model dalam menggeneralisasi, sementara dalam penelitian ini, tanpa augmentasi data yang kompleks mungkin mempengaruhi hasil.

Untuk mendapatkan pemahaman lebih lanjut tentang bagaimana peningkatan jumlah epoch berdampak pada kinerja model, evaluasi yang lebih mendalam perlu dilakukan. Misalnya, penerapan augmentasi data yang lebih canggih dapat membantu mengurangi risiko overfitting dan meningkatkan kemampuan generalisasi model. Selain itu, eksperimen dengan struktur model CNN yang lebih kompleks atau pengaturan hyperparameter yang lebih optimal dapat memperbaiki hasil yang didapat.

Hasil pengujian pada penelitian ini menunjukkan potensi besar bahwa jaringan saraf tiruan, khususnya model Convolutional Neural Network (CNN), dapat digunakan untuk mengidentifikasi kode seri keramik dengan sangat baik dan cukup akurat. Ke depan, penelitian ini dapat dikembangkan lebih lanjut dengan meningkatkan jumlah data pelatihan dan eksperimen dengan teknik augmentasi citra, serta memperkenalkan model CNN yang lebih dalam untuk meningkatkan akurasi. Selain itu, potensi pengembangan model ini juga bisa mencakup aplikasi pada klasifikasi citra lainnya yang memiliki pola atau karakteristik visual yang serupa.

4. Kesimpulan

Penelitian dan analisis yang dijalankan, menyatakan bahwa metode *Convolutional Neural Network* (CNN) beroperasi secara efektif. Aplikasi berbasis web yang dibuat dapat menjalankan sistem identifikasi kode seri keramik dengan baik. Identifikasi kode seri keramik menggunakan CNN dalam penelitian ini menunjukkan kinerja yang optimal. Faktor-faktor yang meliputi jumlah lapisan, *epoch*, dan penggabungan lapisan *dropout* secara signifikan memengaruhi proses pelatihan. Model *Convolutional Neural Network* (CNN) pada proses *training* mendapat akurasi tertinggi yaitu sebesar 96,17% dan 91,75% pada proses *testing*. Dengan demikian, dapat disimpulkan bahwa penerapan metode CNN sangat efektif dalam mengidentifikasi kode seri keramik secara akurat dan dapat digunakan dengan baik pada aplikasi berbasis web. Dengan adanya sistem identifikasi berbasis citra yang akurat,

konsumen dapat lebih mudah mengenali dan membeli keramik yang tepat dengan hanya memotret atau mengunggah gambar keramik yang sudah ada.

Daftar Rujukan

- [1] A. I. Kulakov and A. U. Ri, "Ceramics in architecture," *IOP Conf. Ser. Earth Environ. Sci.*, vol. 751, no. 1, pp. 10–15, 2021, doi: 10.1088/1755-1315/751/1/012052.
- [2] L. Chipindu, W. Mupangwa, J. Mtsilizah, I. Nyagumbo, and M. Zaman-Allah, "Maize Kernel Abortion Recognition and Classification Using Binary Classification Machine Learning Algorithms and Deep Convolutional Neural Networks," *AI*, vol. 1, no. 3, pp. 361–375, 2020, doi: 10.3390/ai1030024.
- [3] A. Roihan, P. A. Sunarya, and A. S. Rafika, "Pemanfaatan Machine Learning dalam Berbagai Bidang: Review paper," *IJCIT (Indonesian J. Comput. Inf. Technol.)*, vol. 5, no. 1, pp. 75–82, 2020, doi: 10.31294/ijcit.v5i1.7951.
- [4] H. A. Pratiwi, M. Cahyanti, and M. Lamsani, "Implementasi Deep Learning Flower Scanner Menggunakan Metode Convolutional Neural Network," *Sebatik*, vol. 25, no. 1, pp. 124–130, 2021, doi: 10.46984/sebatik.v25i1.1297.
- [5] L. Hakim, H. R. Rahmanto, S. P. Kristanto, and D. Yusuf, "Klasifikasi Citra Motif Batik Banyuwangi Menggunakan Convolutional Neural Network," *J. Teknoinfo*, vol. 17, no. 1, p. 203, 2023, doi: 10.33365/jti.v17i1.2342.
- [6] M. Rahman, Asriyanik, and A. Pambudi, "Identifikasi Citra Daun Selada Dalam Menentukan Kualitas Tanaman Menggunakan Algoritma Convolutional Neural Network (CNN)," *J. Inform. dan Tek. Elektro Terap.*, vol. 11, no. 3, pp. 2830–7062, 2023, [Online]. Available: <http://dx.doi.org/10.23960/jitet.v11i3.s1.3438>
- [7] C. Shorten and T. M. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning," *J. Big Data*, vol. 6, no. 1, 2019, doi: 10.1186/s40537-019-0197-0.
- [8] F. Marpaung, F. Aulia, and R. C. Nabila, *Computer Vision Dan Pengolahan Citra Digital*. 2022. [Online]. Available: www.pustakaaksara.co.id
- [9] I. Kurniastuti, E. N. I. Yuliati, F. Yudianto, and T. D. Wulan, "Determination of Hue Saturation Value (HSV) color feature in kidney histology image," *J. Phys. Conf. Ser.*, vol. 2157, no. 1, 2022, doi: 10.1088/1742-6596/2157/1/012020.
- [10] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou, "A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 33, no. 12, pp. 6999–7019, 2022, doi: 10.1109/TNNLS.2021.3084827.
- [11] D. C. Agustin, M. A. Rosid, and N. Ariyanti, "Implementasi Convolutional Neural Network Untuk Deteksi Kesegaran Pada Apel," *J. Fasilkom*, vol. 13, no. 02, pp. 145–150, 2023, doi: 10.37859/jf.v13i02.5175.
- [12] N. IBRAHIM, S. SA'IDAH, B. HIDAYAT, and S. DARANA, "Klasifikasi Grade Telur Ayam Negeri secara non-Invasive menggunakan Convolutional Neural Network," *ELKOMIKA J. Tek. Energi Elektr. Tek. Telekomun. Tek. Elektron.*, vol. 10, no. 2, p. 297, 2022, doi: 10.26760/elkomika.v10i2.297.
- [13] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015, doi: 10.1038/nature14539.
- [14] K. Purcell and J. Brenner, "Pengembangan Model Klasifikasi Produk Furnitur Sebagai Visual Search Menggunakan Algoritma Convolutional Neural Network," vol. VIII, no. September, 2024.
- [15] Y. Kasnanda Bintang, H. Imaduddin, Y. Kasnanda, and B. Corresponding Author, "Pengembangan Model Deep Learning Untuk Deteksi Retinopati Diabetik Menggunakan Metode Transfer Learning," vol. 9, no. 3, pp. 1442–1455, 2024, [Online]. Available: <https://doi.org/10.29100/jipi.v9i3.5588>
- [16] I. Rafiedhia Pramutighna and A. Hermawan, "Pengenalan Potensi Racun dan Peningkatan Keamanan Pangan Dalam

-
- Jamur Menggunakan Convolutional Neural Network,” *J. Media Inform. Budidarma*, vol. 7, no. 4, pp. 1716–1726, 2023, doi: 10.30865/mib.v7i4.6372.
- [17] H. Mukhtar, F. Alfanico, H. Fu’adah Amran, F. Handayani, and R. Medikawati Taufiq, “Deep Learning Untuk Klasifikasi Kematangan Buah Mangrove Berdasarkan Warna,” *J. Fasilkom*, vol. 13, no. 3, pp. 563–569, 2023, doi: 10.37859/jf.v13i3.6292.
- [18] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: A simple way to prevent neural networks from overfitting,” *J. Mach. Learn. Res.*, vol. 15, pp. 1929–1958, 2014.
- [19] A. Peryanto, A. Yudhana, and R. Umar, “Rancang Bangun Klasifikasi Citra Dengan Teknologi Deep Learning Berbasis Metode Convolutional Neural Network,” *Format J. Ilm. Tek. Inform.*, vol. 8, no. 2, p. 138, 2020, doi: 10.22441/format.2019.v8.i2.007.
- [20] D. Krstinić, M. Braović, L. Šerić, and D. Božić-Štulić, “Multi-label classifier performance evaluation with confusion matrix,” *Comput. Sci. Inf. Technol.*, vol. 1, pp. 1–14, 2020.
- [21] R. Setya Nugraha and A. Hermawan, “Optimasi Akurasi Metode Convolutional Neural Network Untuk Klasifikasi Kualitas Buah Apel Hijau,” *J. Mnemon.*, vol. 6, no. 2, pp. 149–156, 2023, doi: 10.36040/mnemonic.v6i2.6730.