

SISTEM INFORMASI HARGA KOST BERBASIS CITRA VISUAL MENGGUNAKAN HYBRID RESNET DAN SUPPORT VECTOR MACHINE (SVM)

Jonathan Alonso^{1*}, Supatman²

^{1,2}Universitas Mercu Buana Yogyakarta, Jl. Wates Km. 10 Yogyakarta 55753, 6498211, Fax. 0274-6498213

Keywords:

Convolutional neural network (CNN); Support Vector Machine (SVM); klasifikasi harga kos; ResNet; aplikasi Flutter

Correspondent Email:

jonathansembiring69@gmail.com



Copyright © [JITET](http://www.jitet.org) (Jurnal Informatika dan Teknik Elektro Terapan). This article is an open access article distributed under terms and conditions of the Creative Commons Attribution (CC BY NC)

Abstrak. Mencari tempat kos yang pas dengan budget merupakan tantangan tersendiri bagi mahasiswa dan pekerja di kota-kota besar. Proses pencarian biasanya memakan waktu cukup lama karena harus datang langsung ke lokasi satu per satu untuk melihat kondisi dan fasilitas kos sebelum bisa memperkirakan harganya. Untuk mengatasi hal ini, penelitian ini mendesain sebuah sistem yang dapat mengenali kategori kos dengan fotonya menggunakan aplikasi teknologi berbasis deep learning. Metode yang diusulkan menggabungkan ResNet50 sebagai pengekstrak fitur gambar dengan SVM sebagai pengklasifikasi. Data yang digunakan berasal dari 1009 foto kos di wilayah Yogyakarta yang sudah dikelompokkan menjadi tiga kategori yaitu kelas rendah, menengah, dan tinggi. Untuk mendapatkan parameter terbaik pada proses learning, dilakukan optimasi menggunakan GridSearchCV dengan teknik 10-fold cross validation. Setelah diuji, sistem mampu mengklasifikasikan dengan tingkat akurasi 80.69%, nilai precision 81.01%, recall 80.03%, dan F1-score 80.79%. Sistem ini kemudian dikembangkan menjadi aplikasi smartphone berbasis Flutter yang terhubung dengan server Flask. Hasil pengujian aplikasi mampu memprediksi dengan tingkat keyakinan di atas 75%. Dengan adanya sistem ini, pencari kos bisa melakukan penilaian awal terhadap kategori dan perkiraan harga kos sebelum memutuskan untuk berkunjung langsung.

Abstract. Finding a boarding house that matches one's budget is a challenge for students and workers in large cities. The search process often takes considerable time because prospective tenants must visit each location to check its condition and facilities before estimating the price. To address this issue, this study designs a system capable of recognizing boarding house categories from photos using a deep learning-based application. The proposed method combines ResNet50 as an image feature extractor with SVM as a classifier. The dataset consists of 1,009 boarding house photos from the Yogyakarta area, grouped into three categories: low, middle, and high class. To obtain the optimal learning parameters, hyperparameter tuning was performed using GridSearchCV with 10-fold cross-validation. The experimental results show that the system achieves an accuracy of 80.69%, with a precision of 81.01%, recall of 80.03%, and F1-score of 80.79%. The system was then deployed into a Flutter-based smartphone application connected to a Flask backend server. Application testing indicates that the model can predict with a confidence level above 75%. With this system, boarding house seekers can conduct an initial assessment of the category and estimated price before deciding to visit the location in person.

1. PENDAHULUAN

Yogyakarta merupakan salah satu kota dengan jumlah pelajar dan pendatang yang tinggi, berdasarkan data dari Badan Perencanaan Pembangunan Riset dan Inovasi Daerah (BAPPERIDA) DIY, total siswa/mahasiswa di DIY diproyeksikan berjumlah sekitar 597.038 orang pada tahun 2025, sehingga kebutuhan akan hunian sementara seperti rumah indekos meningkat pesat. Variasi harga kos sangat beragam dan dipengaruhi oleh kualitas interior, fasilitas, serta kualitas bangunan. Pada platform online, informasi harga sering tidak dicantumkan sehingga calon penyewa hanya mengandalkan foto kamar untuk menilai kelas harga. Foto citra kamar sendiri sebenarnya menyimpan informasi visual, seperti material dinding, fasilitas kamar (Ac, Tv, Meja, Lemari, Sofa, dll) pencahayaan, dan dekorasi, yang dapat dijadikan sebagai indikator kategori harga.

Klasifikasi harga kos dapat membantu calon penyewa untuk mendapatkan gambaran awal untuk mengetahui estimasi harga dari kos tersebut sebelum melakukan survei ke lokasi. Pendekatan ini memudahkan pengguna karena proses penilaian dapat dilakukan melalui foto kamar tanpa perlu datang secara langsung. Sistem dapat membantu penyewa menyesuaikan pilihan hunian dengan kebutuhan dan anggaran yang dimiliki. Oleh karena itu, diperlukan pendekatan otomatis berbasis kecerdasan buatan untuk menganalisis informasi visual pada citra kamar kos secara lebih objektif.

Pendekatan pengolahan citra berbasis *Deep Learning* banyak digunakan dalam tugas klasifikasi karena mampu mengenali pola visual secara otomatis. Salah satu arsitektur yang sering dipakai adalah *Convolutional Neural Network* (CNN), yang dirancang untuk memproses data berbentuk gambar dan telah terbukti efektif dalam mengekstraksi fitur visual. Pada penelitian ini digunakan ResNet sebagai ekstraktor fitur. ResNet merupakan salah satu varian CNN yang memiliki mekanisme *residual learning* untuk mengatasi masalah hilangnya gradien pada jaringan dalam sehingga dapat menghasilkan performa ekstraksi fitur yang lebih stabil. Fitur yang dihasilkan kemudian diklasifikasikan menggunakan *Support Vector Machine* (SVM). SVM dipilih karena mampu menangani data

berdimensi tinggi dan memberikan performa klasifikasi yang stabil pada dataset yang berukuran sedang.

Meskipun penelitian-penelitian tersebut telah menunjukkan hasil yang baik dalam hal prediksi harga berbasis data tabular maupun klasifikasi berbasis citra, namun klasifikasi kategori harga kos berdasarkan informasi visual masih perlu dikembangkan. Padahal, kebanyakan dari calon penyewa membuat keputusan awal berdasarkan foto kamar yang ditampilkan dalam platform digital. Selain itu, implementasi sistem klasifikasi kos dalam bentuk aplikasi mobile yang dapat digunakan untuk memudahkan pengguna dalam mengambil keputusan untuk menentukan estimasi harga kos, yang dilihat dari citra visualnya.

Penelitian ini bertujuan untuk mengembangkan sistem klasifikasi harga kos berdasarkan citra visual dengan menggabungkan ResNet sebagai ekstraktor fitur dan SVM sebagai klasifikator. Dataset yang digunakan berupa 1.009 citra interior kamar kos yang diperoleh dari platform digital seperti Facebook, Instagram, TikTok, dan situs pencarian kos seperti Mamikos. Setiap citra diberi label ke dalam tiga kategori harga: kelas rendah, kelas menengah, dan kelas tinggi. Seluruh data diambil secara publik dan dibatasi pada wilayah Kota Yogyakarta. Sistem yang dikembangkan kemudian diimplementasikan dalam bentuk aplikasi mobile berbasis Flutter dengan backend Flask API untuk memudahkan calon penyewa melakukan penilaian awal. Kontribusi penelitian ini adalah memberikan pendekatan baru dalam klasifikasi kos berbasis informasi visual yang relevan dengan konteks platform digital, serta menyediakan perangkat bantu yang dapat digunakan masyarakat dalam proses pencarian hunian sementara.

2. TINJAUAN PUSTAKA

2.1 Computer Vision dan Klasifikasi Citra

Computer Vision adalah aspek dari kecerdasan buatan yang berorientasi pada pengolahan serta analisis gambar digital untuk secara otomatis mendapatkan informasi yang bermakna. Dalam pengklasifikasian citra, tujuan utama visi komputer adalah untuk mengelompokkan citra ke dalam kategori tertentu berdasarkan atribut visual seperti

warna, tekstur, pencahayaan, dan komposisi ruang. Kemajuan computer vision saat ini sangat dipacu oleh penggunaan deep learning, terutama Convolutional Neural Network (CNN), yang dapat mengekstrak fitur visual secara bertingkat dan lebih representatif dibandingkan dengan metode tradisional.

2.2 Convolutional Neural Network

Jaringan Saraf Konvolusional (CNN) adalah suatu arsitektur deep learning yang dirancang khusus untuk memproses data dengan format grid seperti gambar. CNN menggunakan operasi konvolusi untuk ekstraksi fitur lokal secara spasial, dilanjutkan dengan proses pooling yang bertujuan untuk mengurangi dimensi dan lapisan fully connected untuk memetakan ke ruang kelas[1]. CNN terbukti berhasil dalam berbagai tugas klasifikasi gambar karena kemampuannya untuk belajar fitur secara otomatis tanpa memerlukan ekstraksi manual.

2.3 ResNet dan Feature Extraction

ResNet (Residual Network) diluncurkan untuk mengatasi masalah vanishing gradient pada jaringan yang dalam dengan memanfaatkan residual block dan koneksi skip[2]. ResNet50 merupakan salah satu tipe ResNet yang memiliki 50 lapisan konvolusi dan menghasilkan vektor fitur berukuran 2048 pada lapisan terakhir sebelum lapisan fully connected. Dalam sejumlah penelitian, ResNet50 berperan sebagai pengambil fitur dengan memanfaatkan model yang telah disiapkan sebelumnya dari ImageNet karena kestabilan performanya pada dataset berukuran kecil hingga sedang, serta efisiensi dimensi fitur yang dihasilkan.

2.4 Support Vector Machine

Support Vector Machine (SVM) merupakan algoritma pembelajaran terawasi yang diperkenalkan oleh Vapnik dan sering dimanfaatkan untuk tugas klasifikasi pada data berdimensi tinggi[3]. SVM bekerja dengan membentuk hyperplane optimal yang memaksimalkan jarak antara kelas-kelas. Penerapan kernel, terutama Radial Basis Function (RBF), memungkinkan SVM untuk memetakan hubungan non-linear dalam ruang fitur berdimensi tinggi. Ciri-ciri ini menjadikan SVM cocok digunakan sebagai alat

pengklasifikasi untuk vektor fitur yang diperoleh dari ekstraksi CNN.

2.5 Penelitian Terkait

Beberapa penelitian menerapkan metode CNN maupun kombinasi CNN-SVM untuk klasifikasi citra di berbagai bidang. Penelitian pada klasifikasi pigmen kanker kulit menggunakan CNN mencapai akurasi 75% pada dataset ISIC yang berisi lebih dari 10.000 citra[4]. Penelitian lain membandingkan beberapa arsitektur untuk pengenalan ekspresi wajah menggunakan CNN, ResNet50V2, InceptionV3, dan DenseNet12 dengan total 14.248 gambar. Hasil menunjukkan bahwa ResNet50V2 memberikan performa terbaik dengan akurasi 79%[5]. Pendekatan hybrid CNN-SVM juga banyak digunakan, misalnya pada klasifikasi kesegaran daging ayam [6] dan klasifikasi kesegaran buah pare berbasis fitur citra[7]. Model hybrid lain yang menggabungkan CNN dan SVM untuk klasifikasi tanaman menggunakan data hyperspectral dan juga dilaporkan memberikan akurasi yang tinggi[8]. Selain itu arsitektur CNN seperti MobileNetV2 dan VGG16 yang digabungkan dengan SVM terbukti meningkatkan presisi klasifikasi pada berbagai objek visual, termasuk tumor otak dan komoditas pertanian[9]. Selanjutnya pada penelitian analisis sentiment klub tim nasional yang menggabungkan CNN dan SVM menghasilkan akurasi 73%[10].

Penelitian sebelumnya mengenai harga sewa kos di Yogyakarta umumnya menggunakan data tabular. Salah satu studi menggunakan metode machine learning seperti Random Forest untuk memprediksi harga berdasarkan variabel numerik, seperti harga sewa, wifi, tipe kos, lokasi, fasilitas dan memperoleh akurasi sebesar 75.62%. Pendekatan tersebut belum memanfaatkan data visual citra bangunan atau kamar kos sebagai sumber informasi. Oleh karena itu, diperlukan pendekatan yang memanfaatkan informasi visual citra kamar kos sebagai dasar klasifikasi kelas harga kost dengan menggabungkan CNN dan SVM.

Penelitian lain berfokus pada pengembangan aplikasi klasifikasi berbasis mobile. Seperti sistem klasifikasi sayuran menggunakan CNN yang diimplementasikan dalam aplikasi Android[11], serta sistem

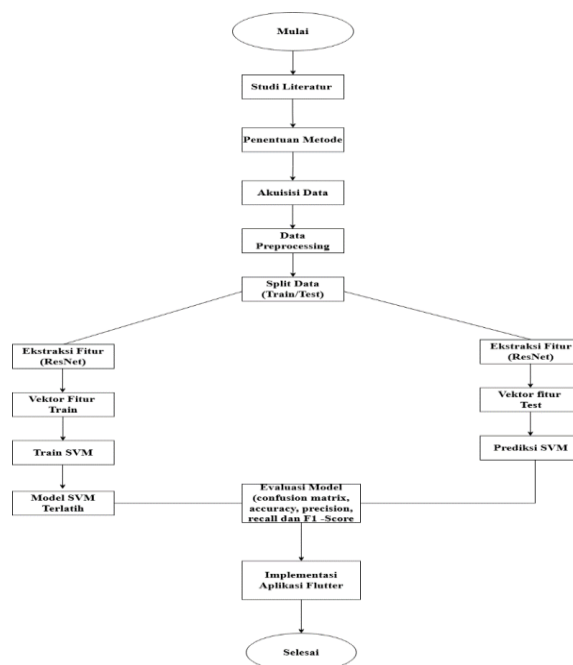
pendeteksi jenis sayuran berbasis CNN yang juga menunjukkan tingkat akurasi yang baik [12]. Penelitian lain mengembangkan sistem klasifikasi penyakit pada tanaman padi menggunakan algoritma CNN, yang mampu mendeteksi penyakit pada daun padi dan berhasil diimplementasikan dalam aplikasi Android untuk mengklasifikasikan gambar daun kedelai ke dalam beberapa kategori [13]. Selain itu, model CNN juga digunakan untuk klasifikasi ternak dengan akurasi tinggi pada citra sapi perah dan non-perah [14].

Terkait prediksi harga atau penilaian objek, beberapa penelitian masih menggunakan pendekatan berbasis data tabular. Sistem pendukung keputusan berbasis fuzzy Tsukamoto digunakan untuk memprediksi harga sewa kos berdasarkan variabel numerik seperti jarak, ukuran kamar, dan fasilitas, menghasilkan akurasi 72% [15]. Pendekatan serupa diterapkan dalam prediksi harga apartemen menggunakan metode *Random Forest* berbasis fitur tabular [16]. Pada bidang lain, algoritma SVM digunakan untuk memprediksi harga rumah dengan akurasi regresi mencapai 95.99% dan akurasi klasifikasi lebih dari 91% [17]. Selain itu, metode SVC juga digunakan untuk klasifikasi harga laptop dan diimplementasikan dalam aplikasi mobile, menghasilkan akurasi 92% setelah proses preprocessing dan feature engineering [18].

3. METODE PENELITIAN

Metode penelitian adalah representasi dari penelitian yang dilaksanakan dari tahap awal sampai akhir. Dalam Gambar 1 dijelaskan proses alur penelitian klasifikasi yang dimulai dari tahap studi literatur untuk memahami metode ResNet dan SVM, dilanjutkan dengan pengumpulan data dan preprocessing. Dataset dibagi menjadi dua bagian yaitu data pelatihan dan data pengujian dengan rasio 80:20. Data pelatihan digunakan untuk ekstraksi fitur dengan *ResNet* yang sudah di latih, menghasilkan vektor fitur yang selanjutnya dimanfaatkan untuk melatih model SVM. Sementara itu, fitur dari data uji diekstraksi dengan *ResNet* yang serupa, dan hasil vektor fitur digunakan untuk prediksi melalui model SVM yang telah dilatih. Output akhir dari model ini adalah penilaian kinerja model yang dikembangkan, yang mencakup confusion

matrix, akurasi, presisi, recall, dan F1-score untuk menilai tingkat akurasi klasifikasi pada dataset yang tersedia. Model yang telah terbukti efektif kemudian diterapkan ke dalam aplikasi Flutter agar dapat digunakan secara aplikatif.



Gambar 1. Alur penelitian

3.1 Pengumpulan Dataset

Data yang dipakai dalam penelitian ini adalah kumpulan berbagai gambar kos yang dikumpulkan melalui proses penelusuran dari berbagai sumber online. Setiap citra menggambarkan keadaan visual dari kos seperti tampilan bangunan kamar, interior, fasilitas, dan area pendukung lainnya. Seluruh data dibagi menjadi tiga kategori berdasarkan kisaran harga: rendah, menengah, dan tinggi. Total gambar yang dikumpulkan adalah 1.009 citra, kemudian dibagi menjadi data latih 807 citra dan data uji 202 citra dengan rasio 80:20 yang umum digunakan pada penelitian klasifikasi citra [4], [5], [7], [13], [14]. Pembagian ini bertujuan agar model mendapatkan representasi data yang memadai selama proses pembelajaran sekaligus memberikan sampel independen untuk menilai kinerja prediksi model. Sebelum melakukan tahap pengolahan, seluruh citra diperiksa kualitasnya untuk menghindari duplikasi dan gambar berkualitas rendah. Selanjutnya gambar akan diubah ukurannya menjadi 224×224 piksel

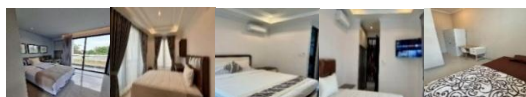
sesuai standar ResNet50 dan dinormalisasi ke rentang 0,1 sebagai tahap preprocessing yang sudah dilakukan pada penelitian sebelumnya [8], [19]. Berikut contoh gambar dataset sampel kelas kos yang dibagi menjadi 3 kategori yang ditunjukkan pada Gambar 2.



1) Kelas Rendah



2) Kelas Menengah



3) Kelas Tinggi

Gambar 2. Sample Dataset dari setiap kelas

3.2 Ekstraksi Fitur ResNet50

Penelitian ini menggunakan ResNet50 yang memiliki 50 lapisan konvolusi dan memanfaatkan *residual block* untuk mengatasi permasalahan *vanishing gradient* [16]. ResNet50 menggunakan bottleneck block sehingga proses ekstraksi fitur lebih efisien. Pada tahapan ekstraksi fitur, setiap citra diubah menjadi vector berdimensi 2048 sebagai representasi numeriknya. Pemilihan *ResNet50* sebagai ekstraktor fitur didasarkan pada kestabilan performanya dan efisiensi dimensi fitur yang dihasilkan. Model yang digunakan merupakan model yang telah melalui pelatihan/pretraining pada ImageNet yang telah dimodifikasi dengan menghapus *fully connected layer* dan menambahkan *Global Average Pooling* yang diterapkan pada beberapa penelitian berbasis CNN-SVM[6], [7], [8], [10], [19]. Seluruh parameter model dibekukan (freeze) sehingga ResNet50 hanya berfungsi sebagai ekstraktor fitur tanpa proses pelatihan ulang.

3.3 Klasifikasi Menggunakan SVM

Klasifikasi dilakukan dengan memanfaatkan algoritma *Support Vector Machine* (SVM) yang diperkenalkan oleh Vapnik[3], dan banyak digunakan dalam penelitian klasifikasi citra karena kemampuannya menangani dimensi fitur yang besar[6], [10], [19], [20] [7]. Menggunakan kernel RBF (*Radial Basis Function*) vektor fitur dengan dimensi 2048 yang dihasilkan melalui ekstraksi fitur ResNet50 digunakan sebagai masukan untuk melatih model SVM. Data dibagi menjadi 80% untuk data latih 807 citra dan 20% untuk data uji 202 citra dengan menggunakan metode *stratified sampling* untuk menjaga distribusi kelas yang seimbang. Proses dari pelatihan SVM ini dilakukan dengan *GridSearchCV* dan 10-fold cross-validation untuk menemukan kombinasi hyperparameter yang terbaik, yang sering digunakan pada penelitian hybrid CNN-SVM [9], [10], [19], [20]. Hyperparameter yang dioptimalkan mencakup nilai C (*regularisasi*) dalam rentang (1, 10, 50, 100), gamma dengan pilihan ('scale', 'auto', 0.001, 0.01, 0.1), kernel RBF, dan class weight 'balanced' untuk mengatasi ketidakseimbangan kelas. Model terbaik ditentukan berdasarkan skor akurasi tertinggi pada validasi silang, lalu digunakan untuk memprediksi data pengujian. Hasil prediksi dievaluasi dengan menggunakan confusion matrix, accuracy, precision, recall, dan F1-score untuk menilai kinerja model pada klasifikasi tiga kelas kategori harga kos.

3.4 Evaluasi Model

Setelah model dirancang dan dilakukan pelatihan serta pengujian dengan data uji, dilakukan beberapa tahapan perhitungan untuk menguji ketepatan serta kualitas dari model yang telah dibuat agar memiliki hasil yang akurat dan menghasilkan Tingkat akurasi yang tinggi serta nilai kesalahan yang kecil. Proses evaluasi juga bertujuan untuk mengetahui sejauh mana model dapat berfungsi dengan baik. Teknik evaluasi yang dilakukan adalah menggunakan metode confusion matrix dengan perhitungan accuracy, precision, recall, dan f1-score yang juga sering digunakan pada penelitian klasifikasi citra berbasis CNN dan CNN dan SVM [19], [20]

3.5 Implementasi

Implementasi sistem dilakukan sebagai tahap realisasi dari desain yang telah disusun sebelumnya. Model CNN-SVM yang telah dikembangkan diintegrasikan ke dalam aplikasi Android menggunakan Flask sebagai backend dan Flutter sebagai frontend. Penggunaan Flask sebagai API server sudah sering digunakan pada berbagai penelitian machine learning berbasis web atau mobile[1], [12], [18]. Flutter digunakan untuk frontend sebagai *framework* lintas platform yang mendukung Android, iOS, web, hingga desktop, dan cukup banyak digunakan dalam pengembangan aplikasi klasifikasi citra. Proses implementasinya mencakup integrasi model ke backend, pembuatan API untuk proses inferensi, pengembangan antarmuka, serta pengujian fungsionalitas untuk memastikan hasil prediksi akurat dan responsif.

4. HASIL DAN PEMBAHASAN

4.1 Hasil Pelatihan Model

Dataset penelitian terdiri dari 1009 gambar kos yang dibagi menjadi 3 kelas: Kelas Rendah (328 gambar/32.5%), Kelas Menengah (344 gambar/ 34.1%), dan Kelas Tinggi (337 gambar/33.4%). Data dibagi dengan rasio 80:20. Proses pelatihan dilakukan menggunakan *ResNet50* dengan ekstraksi fitur dan Support Vector Machine (SVM) sebagai klasifikator menggunakan *GridSearchCV* dengan 10-fold cross validation pada 20 kombinasi hyperparameter.

Hasil evaluasi model menunjukkan performa yang sangat baik dengan accuracy 80.69%, precision (macro) 81.01%, recall (macro) 80.03%, dan F1-Score (macro) 80.79%. Dibawah ini akan menampilkan rekap lengkap metrik evaluasi per kelas pada Tabel.

| Tabel 1. Hasil Klasifikasi | | | | |
|----------------------------|-----------|--------|----------|---------|
| | Prediksi | | | |
| | Precision | Recall | F1-Score | Support |
| Kelas Rendah | 0.8082 | 0.8939 | 0.8489 | 66 |
| Kelas Menengah | 0.7206 | 0.7101 | 0.7153 | 69 |
| Kelas Tinggi | 0.9016 | 0.8209 | 0.859 | 67 |
| Accuracy | | | 0.8069 | 202 |
| Macro avg | 0.8101 | 0.8083 | 0.8079 | 202 |
| Weighted avg | 0.8093 | 0.8069 | 0.8068 | 202 |

| === LAPORAN KLASIFIKASI === | | | | |
|-----------------------------|-----------|--------|----------|---------|
| | precision | recall | f1-score | support |
| Kelas Rendah | 0.8082 | 0.8939 | 0.8489 | 66 |
| Kelas Menengah | 0.7206 | 0.7101 | 0.7153 | 69 |
| Kelas Tinggi | 0.9016 | 0.8209 | 0.8594 | 67 |
| accuracy | | | 0.8069 | 202 |
| macro avg | 0.8101 | 0.8083 | 0.8079 | 202 |
| weighted avg | 0.8093 | 0.8069 | 0.8068 | 202 |

Gambar 3. Hasil Klasifikasi



Gambar 4. Confusion matrix

Confusion matrix pada Gambar 4 menunjukkan distribusi hasil prediksi model terhadap data uji. Hasil evaluasi mengindikasikan bahwa model memiliki performa paling seimbang pada klasifikasi Kelas Tinggi, yang ditunjukkan oleh nilai F1-Score tertinggi sebesar 85,94% dengan presisi mencapai 90,16%. Namun demikian, ditinjau dari aspek sensitivitas (recall), Kelas Rendah menunjukkan capaian tertinggi, dimana sebanyak 59 dari 66 sampel berhasil diklasifikasikan secara benar dengan nilai recall sebesar 89,39%. Di sisi lain, Kelas Menengah memperlihatkan performa yang relatif lebih rendah, dengan hanya 49 dari 69 sampel yang terklasifikasi dengan tepat, sehingga menghasilkan tingkat ketepatan sebesar 71,01%.

1. Kategori Rendah:

- True Positives (TP) = 59
- False Negatives (FN) = 7 + 0 = 7
- False Positives (FP) = 14 + 0 = 14
- Support = TP + FN = 59 + 7 = 66

$$Recall = \frac{TP}{TP+FN} = \frac{59}{59+7} = \frac{59}{66} = 0,8939 \quad (1)$$

Analisa: Nilai Recall sebesar 0,8939 atau 89,39% memperlihatkan bahwa model sangat sensitif dalam mengenali kelas rendah. Hampir seluruh data aktual kelas rendah berhasil dideteksi, model hanya sedikit salah memperkirakan ke kelas menengah.

$$Precision = \frac{TP}{TP+FP} = \frac{59}{59+14} = \frac{59}{73} = 0,8082 \quad (2)$$

Analisa: Nilai Precision sebesar 0,8082 atau 80,39% memperlihatkan model cukup baik, namun terdapat 14 data dari kelas menengah yang salah diprediksi sebagai kelas rendah yang menyebabkan penurunan nilai akurasi.

$$F1-Score = 2 \times \frac{0,8082 \times 0,8939}{0,8082 + 0,8939} = 0,8489 \quad (3)$$

Analisa: Nilai F1-Score kelas rendah sedikit di bawah kelas tinggi. Nilai recall yang tinggi, sebesar 89,39%, menunjukkan bahwa model dapat menemukan kamar kos dengan kategori harga rendah, sehingga hanya sedikit sampel Kelas Rendah yang tidak ditemukan. Namun, nilai presisi yang lebih rendah, sebesar 89,39%, sedikit memengaruhi nilai F1-Score Kelas Rendah.

2. Kategori Menengah

- True Positives (TP) = 49
- False Negatives (FN) = 14 + 6 = 20
- False Positives (FP) = 7 + 12 = 19
- Support = TP + FN = 49 + 20 = 69

$$Recall = \frac{TP}{TP+FN} = \frac{49}{49+20} = \frac{49}{69} = 0,7101 \quad (4)$$

Analisa: Nilai Recall sebesar 0,7103 atau 71,01% adalah nilai model terendah dibandingkan kelas lain. Menunjukkan bahwa model kesulitan mengenali karakteristik unik kelas menengah, sehingga 20 sampel sekitar 29% salah diklasifikasikan kelas rendah atau tinggi.

$$Precision = \frac{TP}{TP+FP} = \frac{49}{49+19} = \frac{49}{68} = 0,7206 \quad (5)$$

Analisa: Nilai Precision sebesar 0,7206 atau 72,06% mengindikasikan bahwa model memprediksi suatu gambar sebagai kelas menengah yang memiliki tingkat kebenaran sekitar 72%. Ini disebabkan oleh banyaknya gambar dari kelas rendah (7) dan kelas tinggi (12) salah diklasifikasikan yang diprediksi sebagai kelas menengah.

$$F1Score = 2 \times \frac{0,7206 \times 0,7101}{0,7206 + 0,7101} = 0,7153 \quad (6)$$

Analisa: Nilai F1-Score dibandingkan dengan dua kelas lainnya, Kelas Menengah menunjukkan nilai F1-Score yang lebih rendah karena kinerja model yang kurang optimal dari sisi presisi (72,06%) dan recall (71,01%). Nilai F1-Score yang lebih rendah menunjukkan bahwa karakteristik visual Kelas Menengah kurang tegas, yang membuatnya sulit bagi model untuk menyeimbangkan ketepatan dan sensitivitas klasifikasi.

3. Kategori Tinggi

- True Positives (TP) = 55
- False Negatives (FN) = 0 + 12 = 12
- False Positives (FP) = 0 + 6 = 6
- Support = TP + FN = 55 + 12 = 67

$$Recall = \frac{TP}{TP+FN} = \frac{55}{55+12} = \frac{55}{67} = 0,8209 \quad (7)$$

Analisa: Nilai Recall sebesar 0,8209 atau 82,09% memperlihatkan model cukup baik. Kesalahan utama terjadi karena 12 sampel kelas tinggi dianggap sebagai kelas menengah oleh model, dikarenakan beberapa fasilitas dianggap tidak terlalu mewah oleh model seperti kebanyakan pada kelas tinggi.

$$Precision = \frac{TP}{TP+FP} = \frac{55}{55+6} = \frac{55}{61} = 0,9016 \quad (8)$$

Analisa: Nilai Precision sebesar 0,9016 atau 90,16% memperlihatkan model ini memiliki nilai presisi yang tertinggi. Ini berarti jika model memprediksi Kelas Tinggi, kemungkinan besar prediksi tersebut benar, Model sangat jarang salah menebak gambar kelas lain sebagai kelas tinggi.

$$F1Score = 2 \times \frac{0,9016 \times 0,8209}{0,9016 + 0,8209} = 0,8594 \quad (9)$$

Analisa: Nilai F1-Score Kelas Tinggi memiliki nilai tertinggi dibandingkan kelas lainnya, yang

didukung oleh nilai presisi yang tinggi sebesar 90,16%, yang menunjukkan bahwa kesalahan klasifikasi dari kelas lain relatif rendah. Meskipun nilai recall sebesar 82,09% bukan yang tertinggi, kombinasi kedua metrik tersebut menunjukkan bahwa fitur visual Kelas Tinggi memiliki karakteristik yang lebih unik dan lebih mudah diidentifikasi oleh model.

4. Macro Average F1-Score

$$\text{Macro Average F1-Score} = \frac{\sum_{i=1}^K (F1_i)}{K} = \frac{0,8489 + 0,7153 + 0,8594}{3} = 0,8079 \quad (10)$$

Analisa: Nilai Macro Average, menurut Persamaan (4), nilai Macro Average F1-Score adalah 0,8079, atau 80,79 persen, yang menunjukkan bahwa model memiliki kinerja klasifikasi yang cukup seimbang di seluruh kelas.

5. Weighted Average F1-Score

$$\text{Weighted F1-Score} = \frac{\sum_{i=1}^K (n_i \times F1_i)}{N} = \frac{(66 \times 0,8489) + (69 \times 0,7153) + (67 \times 0,8594)}{202} = 0,8067 \quad (11)$$

Analisa: Nilai Weighted Average, menurut Persamaan (5), nilai Weighted Average F1-Score sebesar 0,8067, atau 80,67%, diperoleh. Nilai ini menunjukkan bahwa model bekerja dengan konsisten dan mampu membuat generalisasi yang baik dari semua data pengujian.

6. Nilai akurasi Model Pada DataTesting

$$\text{Akurasi} = \frac{\text{Total Prediksi Benar (TP)}}{\text{Total Dataset}} = \frac{59 + 49 + 55}{202} = \frac{163}{202} = 0,8069 \quad (12)$$

4.2 Implementasi Aplikasi Mobile

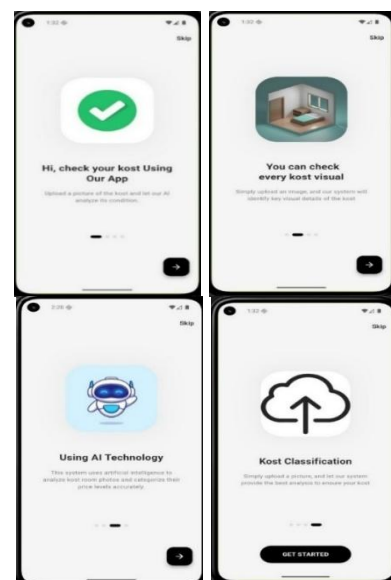
Aplikasi dikembangkan menggunakan Flutter Framework untuk frontend dengan integrasi model melalui Flask rest Api sebagai backend. Arsitektur sistem menggunakan client-server model dimana aplikasi mobile mengirimkan gambar ke server Flask untuk diproses, kemudian menerima hasil klasifikasi dalam format JSON. Proses klasifikasi dimulai ketika pengguna mengupload atau mengambil foto kos melalui aplikasi. Gambar kemudian dikirim ke endpoint Flask Api menggunakan http post request. Server melakukan

preprocessing gambar resize, normalization, kemudian menggunakan ekstraksi fitur *ResNet50*, dan klasifikator menggunakan SVM. Hasil prediksi berupa kelas rendah, menengah, tinggi, estimasi harga dan skor tingkat kepercayaan (confidence) dikirim kembali ke aplikasi dalam format JSON untuk ditampilkan kepada pengguna.

4.2.1 Halaman Welcome

Pada halaman welcome terdapat empat halaman tampilan awal yang akan muncul kepada pengguna baru yaitu:

1. Pengenalan aplikasi untuk melihat/mengecek kondisi kos.
2. Kemampuan sistem yang bisa mengidentifikasi detail visual citra kos.
3. Menggunakan teknologi *artificial intelligence* untuk menganalisis foto kamar lalu mengkategorikan berdasarkan kelasnya.
4. Menginstruksikan kepada pengguna untuk mengupload gambar kos agar dianalisis oleh sistem.
5. Dilengkapi juga dengan navigasi slide, page indikator, dan tombol skip untuk memudahkan pengguna.



Gambar 5. Halaman Welcome

4.2.2 Halaman Home

Tampilan halaman home menyediakan dua opsi input gambar yaitu, upload dari galeri

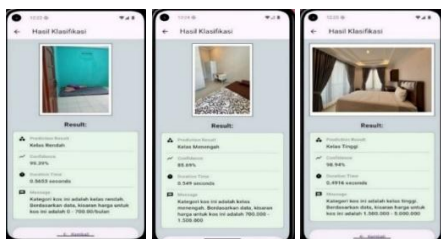
atau capture menggunakan kamera device. Tampilan dirancang untuk minimalis dan konsisten untuk memudahkan pengguna dalam berinteraksi dengan aplikasi.



Gambar 6. Halaman home

4.2.3 Halaman Result

Tampilan halaman result ini menampilkan hasil klasifikasi berupa kategori kos yang diprediksi oleh sistem, kelas rendah, kelas menengah, kelas tinggi, dan confidence score dalam persentase, serta message kategori kos dan estimasi harga untuk kelas kos tersebut.



Gambar 7. Halaman result

4.3 Pengujian Sistem

Dalam penelitian ini dilakukan pengujian sistem untuk menilai efektivitas sistem dalam mengklasifikasikan harga kos berdasarkan kelas rendah, kelas menengah, dan kelas tinggi. Tes ini dianggap berhasil jika data yang diujikan bisa dikelompokkan ke dalam kategori yang sesuai dengan basis data. Sebaliknya, pengujian dinyatakan gagal jika data uji tidak dapat dikelompokkan ke dalam salah satu kategori harga kos yang tersedia dalam database. Uji coba sistem dilaksanakan dalam dua fase: uji fungsional aplikasi dan uji akurasi model.

4.3.1 Hasil Pengujian Fungsional Aplikasi

Pengujian fungsional dilakukan pada seluruh fitur aplikasi yang mencakup unggah

gambar dari galeri, interaksi client-server, proses klasifikasi, dan tampilan hasil prediksi. Hasil dari pengujian itu akan ditunjukkan pada Tabel 2 yang memperlihatkan semua fungsi berfungsi dengan baik tanpa adanya error atau crash.

Tabel 2. Data Hasil Pengujian Fungsional Aplikasi

| No | Fitur | Skenario Pengujian | Hasil | Status |
|----|-------------------|----------------------------------|--|----------|
| 1 | Upload Image | Upload gambar kos dari galeri | Berhasil memuat dan menampilkan preview. | Berhasil |
| 2 | API Communication | Kirim gambar ke server Flask | Request dan response berhasil. | Berhasil |
| 3 | Klasifikasi | Prediksi kategori kos | Prediksi akurat sesuai model. | Berhasil |
| 4 | Confidence Score | Tampilan presentase keyakinan | Muncul dengan benar. | Berhasil |
| 5 | Loading indicator | Tampilan saat proses klasifikasi | Muncul dan hilang dengan tepat. | Berhasil |
| 6 | Error Handling | Koneksi server gagal/timeout | Menampilkan pesan error yang jelas. | Berhasil |
| 7 | Navigasi | Pindah antar halaman | Transisi smooth tanpa lag. | Berhasil |

4.3.2 Hasil Pengujian Akurasi Model

Tabel 3. Data Hasil Pengujian Akurasi Model

| Gambar | Kelas Target | Hasil Prediksi | Tingkat Kepercayaan | Durasi |
|--------|--------------|----------------|---------------------|----------------|
| | Kelas rendah | Kelas Rendah | 99.39% | 4.0908 seconds |

| | | | | |
|---|----------------|----------------|--------|----------------|
|  | Kelas Rendah | Kelas Rendah | 96.92% | 0.5614 seconds |
|  | Kelas Rendah | Kelas Rendah | 99.30% | 0.6676 seconds |
|  | Kelas Rendah | Kelas Rendah | 99.18% | 0.6055 seconds |
|  | Kelas Rendah | Kelas Rendah | 64.17% | 0.6131 seconds |
|  | Kelas Menengah | Kelas Menengah | 88.59% | 0.728 seconds |
|  | Kelas Menengah | Kelas Menengah | 80.82% | 0.6911 seconds |
|  | Kelas Menengah | Kelas Menengah | 73.85% | 0.5155 seconds |
|  | Kelas Menengah | Kelas Menengah | 67.37% | 0.5566 seconds |
|  | Kelas Menengah | Kelas Menengah | 51.36% | 0.8475 seconds |
|  | Kelas Tinggi | Kelas Tinggi | 91.45% | 0.714 seconds |
|  | Kelas Tinggi | Kelas Tinggi | 98.94% | 0.5438 seconds |
|  | Kelas Tinggi | Kelas Tinggi | 76.90% | 0.587 seconds |
|  | Kelas Tinggi | Kelas Tinggi | 90.77% | 0.5198 seconds |
|  | Kelas Tinggi | Kelas Tinggi | 85.69% | 0.7666 seconds |

4.4 Pembahasan

Hasil eksperimen ini menunjukkan bahwa integrasi metode *Hybrid ResNet50* sebagai ekstraktor fitur dan *Support Vector Machine* (SVM) sebagai klasifikator mampu memberikan performa yang signifikan dalam klasifikasi harga kos, dengan capaian akurasi keseluruhan sebesar 80.69%. Pengujian implementasi pada aplikasi *mobile* berbasis Android juga menunjukkan bahwa model mampu melakukan inferensi secara *real-time* untuk mengkategorikan citra kamar kos

kedalam tiga kelas: Rendah, Menengah dan Tinggi. Berdasarkan analisis dari *confusion matrix*, model menunjukkan efektivitas tinggi dalam mengidentifikasi Kelas Tinggi dan Kelas Rendah, yang dibuktikan dengan perolehan nilai *F1-score* masing masing sebesar 85.94% dan 84.89%. Namun, kelas menengah menunjukkan penurunan kinerja, dengan hanya 71.01% akurasi prediksi benar dan 14 sampel dianggap sebagai kelas rendah dan 6 sampel dianggap sebagai kelas tinggi. Ini menunjukkan tumpang tindih (*overlap*) karakteristik visual antara kos menengah dan rendah. Karena kondisi atau fasilitas kamar adalah kelas menengah, misalnya dari segi warna dinding atau perabotan, model sangat sulit untuk menarik batas (*decision boundary*). Oleh karena itu, karena kemiripan ruangan dan perabotan yang ada, model yang dikembangkan sering memprediksi kelas menengah sebagai kelas rendah. Nilai akurasi total model adalah 80,69% karena hasil akurasi kelas menengah yang rendah.

Dalam komparasi dengan literatur terdahulu, implementasi model *Deep Learning* ke dalam aplikasi *mobile* merupakan kebaruan dan keunggulan utama penelitian ini. Studi sebelumnya yang menggunakan metode *Random Forest* berbasis data tabular (variabel numerik) tercatat menghasilkan akurasi sebesar 75.62%. Penelitian ini membuktikan bahwa pendekatan berbasis citra visual (*computer vision*) mampu menangkap kompleksitas fitur yang tidak terwakili oleh data tabular, sehingga menghasilkan akurasi yang lebih tinggi. Penelitian ini menawarkan solusi yang efektif dan praktis melalui implementasi sistem. Aplikasi yang dikembangkan memungkinkan pengguna untuk memperoleh estimasi kategori dan harga sewa secara instan melalui unggahan citra, sehingga meningkatkan efisiensi waktu dalam proses survei lokasi secara langsung.

Secara keseluruhan, penelitian ini menghasilkan hipotesis bahwa optimasi model *ResNet50* dan *SVM* yang digunakan pada perangkat *mobile* adalah metode yang efektif untuk memfasilitasi penilaian harga kos. Sistem ini membantu penyewa membuat keputusan cepat. Sangat penting untuk dicatat bahwa model memiliki keterbatasan dalam membedakan kelas yang memiliki kemiripan

visual yang tinggi. Untuk penelitian selanjutnya, diperlukan peningkatan dataset yang lebih variatif. Ini akan menunjukkan batas fitur (batas keputusan) antar kelas dan melihat bagaimana menggabungkan variabel non-visual (seperti lokasi geografis) untuk meningkatkan ketepatan prediksi model.

5. KESIMPULAN

Dengan menggunakan metode Hybrid ResNet50 sebagai ekstraktor fitur dan Support Vector Machine (SVM) sebagai klasifikator, penelitian ini berhasil mengembangkan sistem klasifikasi harga kos berbasis citra visual ke dalam tiga kategori: Rendah, Menengah, dan Tinggi. Hasil pengujian menunjukkan bahwa sistem mampu mencapai akurasi sebesar 80.69%, presisi sebesar 81.01%, recall sebesar 80.03%, dan skor F1-nya sebesar 80.79%. Namun, identifikasi Kelas Menengah masih menghadapi kesulitan dengan nilai F1-Score sebesar 71,53%, yang disebabkan oleh kesamaan karakteristik visual antara kelas menengah, kelas rendah, dan kelas tinggi

Sistem ini telah berhasil diimplementasikan menjadi aplikasi mobile berbasis Flutter yang terhubung dengan server Flask API, yang mampu berjalan lancar pada perangkat Android dengan waktu klasifikasi rata-rata 3-5 detik. Pengujian fungsional membuktikan bahwa seluruh fitur aplikasi, mulai dari unggah foto hingga tampilan hasil prediksi beserta tingkat keyakinan (confidence score), berfungsi dengan baik. Secara praktis, aplikasi ini bermanfaat bagi pencari kos untuk melakukan penilaian awal terhadap kategori dan estimasi harga secara efisien hanya melalui media foto, tanpa harus melakukan survei fisik secara langsung.

DAFTAR PUSTAKA

- [1] S. S. Rambe, A. Asriyanik, and P. Prajoko, "PENERAPAN MODEL CONVOLUTIONAL NEURAL NETWORK (CNN) BERBASIS MOBILENETV2 UNTUK KLASIFIKASI TINGKAT KESEGERAN IKAN NILA," *J. Inform. Dan Tek. Elektro Terap.*, vol. 13, no. 3, Jul. 2025, doi: 10.23960/jitet.v13i3.6744.
- [2] K. W. Nugraha and N. Nurgiyatna, "Analisis Tren Penelitian Residual Network (ResNet) melalui Systematic Literature Review dan Bibliometric Analysis," *Sainteks*, vol. 22, no. 2, pp. 117–131, Oct. 2025, doi: 10.30595/sainteks.v22i2.27154.
- [3] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, Sep. 1995, doi: 10.1007/BF00994018.
- [4] Luqman Hakim, Z. Sari, and H. Handhajani, "Klasifikasi Citra Pigmen Kanker Kulit Menggunakan Convolutional Neural Network," *J. RESTI Rekayasa Sist. Dan Teknol. Inf.*, vol. 5, no. 2, pp. 379–385, Apr. 2021, doi: 10.29207/resti.v5i2.3001.
- [5] S. A. S. Mola, B. O. D. K. Wadu, A. N. Kenlopo, and V. C. K. Tunga, "Perbandingan Arsitektur ResNet50V2, InceptionV3, dan DenseNet121 dalam Klasifikasi Pengenalan Ekspresi Wajah," *JIKO J. Inform. Dan Komput.*, vol. 9, no. 2, p. 285, Jun. 2025, doi: 10.26798/jiko.v9i2.1584.
- [6] A. Agung Mujiono, K. Kartini, and E. Yulia Puspaningrum, "IMPLEMENTASI MODEL HYBRID CNN-SVM PADA KLASIFIKASI KONDISI KESEGERAN DAGING AYAM," *JATI J. Mhs. Tek. Inform.*, vol. 8, no. 1, pp. 756–763, Mar. 2024, doi: 10.36040/jati.v8i1.8855.
- [7] L. S. Arzeti, J. L. Putra, and T. Setiyorini, "MODEL HYBRID CNN-SVM UNTUK KLASIFIKASI KESEGERAN BUAH PARE BERDASARKAN CITRA," *J. Inform.*, vol. 9, no. 4, 2025.
- [8] S. Wan, M.-L. Yeh, and H.-L. Ma, "An Innovative Intelligent System with Integrated CNN and SVM: Considering Various Crops through Hyperspectral Image Data," *ISPRS Int. J. Geo-Inf.*, vol. 10, no. 4, p. 242, Apr. 2021, doi: 10.3390/ijgi10040242.
- [9] M. J. Adamu *et al.*, "Efficient and Accurate Brain Tumor Classification Using Hybrid MobileNetV2–Support Vector Machine for Magnetic Resonance Imaging Diagnostics in Neoplasms," *Brain Sci.*, vol. 14, no. 12, p. 1178, Nov. 2024, doi: 10.3390/brainsci14121178.
- [10] Y. Prastya, "HYBRID CNN-SVM UNTUK ANALISIS SENTIMEN KLUB TIM NASIONAL," *J. Inform. Dan Tek. Elektro Terap.*, vol. 13, no. 3S1, Oct. 2025, doi: 10.23960/jitet.v13i3S1.7764.
- [11] M. J. Akbar, M. W. Sardjono, M. Cahyanti, and E. R. Swedia, "PERANCANGAN APLIKASI MOBILE UNTUK KLASIFIKASI SAYURAN MENGGUNAKAN DEEP LEARNING CONVOLUTIONAL NEURAL NETWORK," *Sebatik*, vol. 24, no. 2, pp. 300–

- 306, Dec. 2020, doi: 10.46984/sebatik.v24i2.1134.
- [12] R. S. Budiawan and B. Hartono, "Pengembangan Sistem Pendeteksi Jenis Sayuran dengan Metode CNN Berbasis Android," *J. Inform. Dan Rekayasa Perangkat Lunak*, vol. 5, no. 1, p. 62, Mar. 2023, doi: 10.36499/jinrpl.v5i1.7833.
 - [13] B. Kusuma, T. I. Hermanto, and C. D. Lestari, "KLASIFIKASI JENIS PENYAKIT PADA TANAMAN PADI MENGGUNAKAN ALGORITMA CONVOLUTIONAL NEURAL NETWORK," *JIKO J. Inform. Dan Komput.*, vol. 9, no. 1, p. 40, Feb. 2025, doi: 10.26798/jiko.v9i1.1395.
 - [14] L. Maramis, I. Nurtanio, and H. Zainuddin, "Klasifikasi Sapi Perah dan Non-Perah Menggunakan Algoritma Convolutional Neural Network: Classification of Dairy and Non-Dairy Cattle Using the Convolutional Neural Network Algorithm," *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 5, no. 2, pp. 664–674, Apr. 2025, doi: 10.57152/malcom.v5i2.1824.
 - [15] R. K. Fitri and A. Iwan, "Sistem Pendukung Keputusan Dalam Memprediksi Harga Sewa Kos Menggunakan Fuzzy Tsukamoto Pada Kos Daerah Universitas Airlangga Kampus B", *JMI*, vol 13, no. 2, 2024.
 - [16] I. L. Mulyahati, "(Studi Kasus: Apartemen di DKI Jakarta Pada Website mamikos.com)," Skripsi, Universitas Islam Indonesia, Yogyakarta, 2020.
 - [17] F. R. Lumbanraja, R. A. Saputra, K. Muludi, A. Hijriani, and A. Junaidi, "IMPLEMENTASI SUPPORT VECTOR MACHINE DALAM MEMPREDIKSI HARGA RUMAH PADA PERUMAHAN DI KOTA BANDAR LAMPUNG," *J. Pepadun*, vol. 2, no. 3, pp. 327–335, Dec. 2021, doi: 10.23960/pepadun.v2i3.90.
 - [18] M. I. A. Supriyanto, A. A. R. Hasan, D. Dharmasesa, R. F. Aththar, S. A. Febrinato, and C. M. Sari, "Integrasi Mobile Aplikasi Untuk Klasifikasi Harga Laptop Menggunakan Metode Support Vector Classification Dan Logistic Regression," vol. 6, no. 4, 2025.
 - [19] S. Ahlawat and A. Choudhary, "Hybrid CNN-SVM Classifier for Handwritten Digit Recognition," *Procedia Comput. Sci.*, vol. 167, pp. 2554–2560, 2020, doi: 10.1016/j.procs.2020.03.309.