

Pengenalan Wajah dengan Algoritma *Support Vector Machine* dan *Sobel Edge Detection* Berbasis *Computer Vision* dan *Caffe Framework*

Muhammad Syarif Hidayatulloh, A. Yudi Permana dan Wahyu Hadi Kristanto

Jurusan Teknik Informatika, Universitas Pelita Bangsa

Jl. Inspeksi Kalimalang Tegal Danas. Cikarang Pusat.

E-mail : muhammadsyarifh30@gmail.com, yudi@pelitabangsa.ac.id, wahyu.hadikristanto@pelitabangsa.ac.id

Abstrak

Dalam penelitian ini, merumuskan masalah bagaimana menerapkan algoritma *support vector machine* dan ekstraksi fitur *edge detection* dengan metode sobel dalam melakukan pengenalan wajah. Pengenalan wajah dilakukan dengan menggunakan *caffe* untuk mendeteksi wajah dari citra yang diinputkan, kemudian dari wajah tersebut dilakukan *edge detection* dengan metode sobel dan algoritma *convolution*. Klasifikasi dilakukan dengan menggunakan algoritma *support vector machine* dengan metode *One Against All*. Dengan metode holdout untuk melakukan validasi dan *confusion matrix* untuk melakukan evaluasi diperoleh akurasi terbaik sebesar 100%. Dataset yang digunakan sebanyak 150 data yang terdiri dari 3 kelas. Proses pengambilan citra dataset diperoleh dengan jarak ± 1 meter, tinggi wajah dan kamera $\pm 1,2$ meter, luminans berkisar 3000-4000 lux dan menggunakan resolusi kamera 8 Mp. Dari dataset tersebut, dilakukan evaluasi berdasarkan jumlah dataset, yaitu 10, 20, 30 dan 40 data setiap kelas. Keseluruhan menghasilkan akurasi yang sama yaitu 100%. Berdasarkan akurasi yang diperoleh dapat disimpulkan bahwa metode yang digunakan cukup baik. Tetapi setelah dilakukan pengujian pengenalan wajah dengan input data citra baru, hanya mendapatkan akurasi sebesar 78% dengan menggunakan dataset sebanyak 10 dan 20 data. Kemudian mendapatkan akurasi sebesar 89% dengan 30, 40 dan 50 data yang digunakan. Data citra baru yang diinputkan sebanyak 9 data yang terdiri dari 3 kelas. Berdasarkan hasil tersebut, perlu dilakukan penelitian selanjutnya untuk memperbaiki citra hasil *edge detection* yang dapat mempengaruhi hasil klasifikasi.

Kata kunci : Pengenalan Wajah, *Support Vector Machine*, *Caffe*, Sobel

Pendahuluan

Manusia memiliki bentuk wajah yang berbeda, perbedaan ini menjadikannya salah satu identitas bagi setiap manusia. Wajah sebagai objek dua dimensi yang digambarkan dengan berbagai macam iluminasi, pose serta ekspresi wajah untuk diidentifikasi berdasarkan citra dua dimensi dari wajah tersebut [1]. Seiring dengan berkembangnya zaman, kebutuhan akan sistem cerdas yang mampu mengidentifikasi sebuah wajah manusia sangat diperlukan, salah

satunya dalam bidang keamanan rumah untuk menangkap wajah pencuri atau orang tidak dikenal yang mencoba mendekati rumah dengan memanfaatkan kamera CCTV. Akan tetapi sebelum sistem tersebut diterapkan, perlu dilakukan penelitian tentang performa metode yang digunakan dalam mengenali wajah tersebut, serta pengaruh jumlah dataset yang digunakan.

Sistem pengenalan wajah bekerja dengan mendeteksi objek wajah dari citra yang di-

masuk, kemudian mengklasifikasi citra wajah tersebut dengan citra yang telah tersimpan dalam sebuah basis data serta menemukan kecocokan wajah yang paling sesuai. Dalam melakukan klasifikasi dapat menggunakan algoritma *support vector machine* [2]. Beberapa penelitian tentang algoritma SVM untuk pengenalan wajah (*face recognition*) sebelumnya pernah dilakukan. Penelitian dengan menggunakan algoritma SVM Multi-Kernel mendapatkan akurasi sebesar 89%, dataset yang digunakan 10 macam wajah dengan masing-masing 20 citra, penambahan pertama 15 citra dan ke dua 10 citra [1]. Penelitian juga dilakukan dengan menggunakan algoritma histogram of oriented gradient (HOG) untuk membantu membandingkan hasil klasifikasi antara algoritma random forest dengan *support vector machine* dengan akurasi RF sebesar 97,17% sedangkan SVM sebesar 97,94% (K=20). Dataset yang digunakan berjumlah 800 citra terdiri dari 40 kelas [3]. Penelitian dengan menggunakan fitur SURF dengan SVM juga pernah dilakukan dengan akurasi yang dihasilkan Yalefaces 97.78 % dan UMIST 97.87 %, dataset yang digunakan berjumlah

Dari ketiga penelitian tersebut perlu diperhatikan mengenai pengaruh tahap preprocessing data serta jumlah dataset yang digunakan terhadap performa akurasi yang dihasilkan, meskipun penelitian yang ketiga menggunakan dataset yang lebih sedikit, akan tetapi mendapatkan akurasi lebih tinggi dari penelitian pertama dan kedua.

Penelitian ini mengusulkan metode dalam melakukan pengenalan wajah dengan memanfaatkan arsitektur layer *convolutional neural network* didalam *caffe framework* untuk mendeteksi wajah. Kemudian untuk menemukan performa akurasi hasil klasifikasi yang baik, penelitian ini akan menggunakan analisa citra *edge detection* (deteksi tepi) dengan menggunakan sobel operator karena lebih baik daripada metode lainnya seperti prewitt dan robert [5]. Meskipun penelitian selanjutnya menyimpulkan bahwa metode canny merupakan metode deteksi tepi yang lebih baik [6] [7], akan tetapi dalam hal efisiensi waktu eksekusi dan kompleksitas ruang sobel lebih baik [8]. Dalam melakukan *feature extraction* deteksi tepi, peneliti menggunakan algoritma *convolution*. Serta untuk memudahkan penelitian dalam mengolah citra, peneliti menggunakan

library OpenCV dengan bahasa pemrograman python.

Teknik Pengenalan Wajah

Pengenalan Wajah

Pengenalan wajah adalah salah satu sistem identifikasi yang dikembangkan berdasarkan perbedaan ciri wajah seseorang berbasis biometrik yang memiliki keakuratan tinggi[9]. Pengenalan wajah bekerja dengan mendeteksi citra wajah dari objek yang diinputkan kemudian dilakukan klasifikasi terhadap citra yang tersimpan didalam basis data.

Computer Vision

Computer vision merupakan disiplin ilmu yang bertujuan untuk memberikan kemampuan yang serupa, ke dalam komputer. Lebih tepatnya, *computer vision* berusaha mengembangkan metode yang mampu meniru salah satu kemampuan yang paling menakjubkan dari sistem visual manusia, menyimpulkan karakteristik dunia nyata 3D murni menggunakan cahaya terpantul ke mata dari berbagai benda [10].

CAFFE Framework

CAFFE merupakan *framework* deep learning opensource yang dikembangkan oleh *Berkeley Vision and Learning Center* (BVLC) dengan dibantu oleh komunitas aktif yang berkontribusi di Github . CAFFE memiliki arsitektur data stores dan komunikasi data dengan array 4 dimensi yang disebut dengan blobs sehingga dapat menggunakan memori dengan lebih efisien [11].

Sobel Edge Detection

Sobel edge detection merupakan salah satu metode yang digunakan untuk melakukan deteksi garis tepi. Deteksi tepi dilakukan dengan cara melakukan filtrasi citra dengan menggunakan sobel operator [7].

$$K_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, K_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

Kernel (K) merupakan filter sobel operator dengan K_x sebagai filter horizontal dan K_y sebagai filter vertikal. Dalam

melakukan filtrasi sobel mendapatkan hasil filtrasi horizontal I_x dan vertikal I_y . Kemudian dilakukan lokalisasi gradient G , besarnya G dan kemiringan θ dari gradien dihitung sebagai berikut:

$$|G| = \sqrt{I_x^2 + I_y^2} \quad (1)$$

$$\theta(x, y) = \arctan\left(\frac{I_y}{I_x}\right) \quad (2)$$

Dalam menghilangkan noise citra dilakukan smoothing (menghaluskan) citra dengan melakukan proses *convolution* dengan menggunakan filter Gaussian[6]. Fungsi gaussian (g) dengan koordinat x, y dikendalikan oleh varian σ^2 sesuai persamaan berikut [12]:

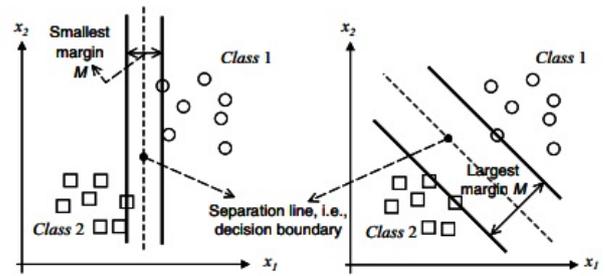
$$g(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\left(\frac{x^2+y^2}{2\sigma^2}\right)} \quad (3)$$

Holdout Validation

Holdout validation merupakan metode evaluasi yang paling sederhana. Hal itu dapat dilihat hanya dengan membagi dataset secara acak menjadi data training dan data test. Dalam membagi dataset dengan menetapkan 2/3 untuk data training dan 1/3 data untuk data test. Pelatihan / ujian perpecahan umumnya 60/40, 70/30, 80/20 atau bahkan 90/10 jika dataset relatif besar. Kemudian melatih data training (dengan menggunakan algoritma pembelajaran). Selanjutnya memprediksi label pada data test. Akurasi didapat dengan membandingkan kebenaran prediksi data test terhadap kesalahan prediksi [13].

Support Vector Machine

Support vector machine (SVM) adalah algoritma pembelajaran mesin supervised yang digunakan untuk menyelesaikan masalah klasifikasi atau regresi [10].



Gambar 1: *Hyperplane* Terbaik (Kanan) dan Kurang Baik (Kiri)

SVM bekerja dengan mendefinisikan batas antara 2 kelas dengan jarak maksimal dari data terdekat dengan menemukan *hyperplane* terbaik pada input space, dan mengukur margin *hyperplane*. *Hyperplane* dengan jumlah data n dapat diperoleh dengan persamaan:

$$w_l \cdot x_l + b = 0, \quad l = 1, 2, 3, \dots, n \quad (4)$$

Dalam melakukan pelatihan data, diperlukan parameter dan b dari fungsi diskriminan $d(x, w, b)$ [14].

$$d(x, w, b) = w^T x + b = \sum_{i=1}^n w_i x_i + b \quad (5)$$

Dimana $x, w \in R$ dan skalar b disebut dengan bias. Garis putus-putus pada gambar 1 merepresentasikan dari $d(x, w, b) = 0$. Dalam melakukan klasifikasi menggunakan persamaan berikut:

$$if = 0 = \text{sign}(d(x_p, w, b)) \quad (6)$$

Dimana 0 adalah notasi standar output dari pembelajaran mesin dan x_p merupakan data test. Dengan kata lain jika $(x_p, w, b) > 0$, pola x_p dimiliki kelas 1 ($0 = y_1 = +1$) dan jika $(x_p, w, b) < 0$, pola x_p dimiliki kelas -1 atau $0 = y_2 = -1$.

Confusion Matrix

Confusion matrix adalah matriks yang berisikan informasi tentang hasil prediksi klasifikasi dan data aktual yang dilakukan oleh sistem klasifikasi, lihat Tabel 1 [15].

Tabel 1: *Confusion Matrix*

		Nilai sebenarnya	
		TRUE	FALSE
Nilai Prediksi	TRUE	TP (True Positive) Correct result	FP (False Positive) Unexpected result
	FALSE	FN (False Negative) Missing result	TN (True Negative) Correct absence of result

$$Precision = \frac{TP}{TP+FP} \quad (7)$$

$$Recall = \frac{TP}{TP+FN} \quad (8)$$

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (9)$$

Dengan,

TP : Citra positif terdeteksi benar

FP : Citra positif terdeteksi salah

TN : Citra negatif terdeteksi benar

FN : Citra negatif terdeteksi salah

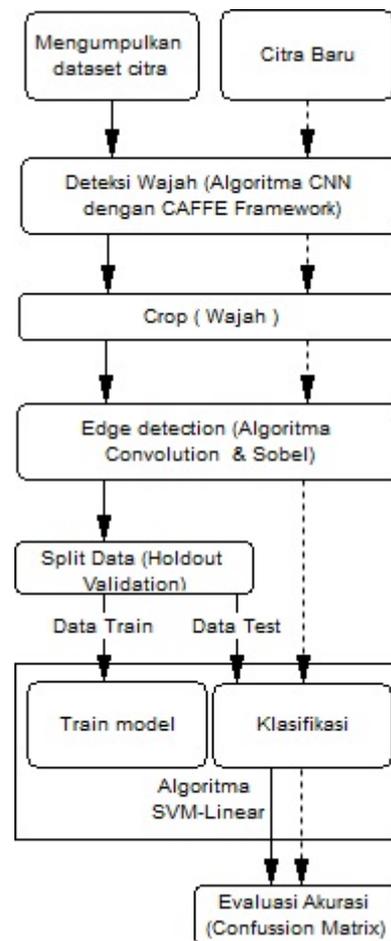
Precision adalah tingkat ketepatan antara informasi yang diminta oleh pengguna dengan jawaban yang diberikan oleh sistem. *Recall* adalah tingkat keberhasilan sistem dalam menemukan kembali sebuah informasi. Akurasi (*accuracy*) didefinisikan sebagai tingkat kedekatan antara nilai prediksi dengan nilai aktual.

Metode Penelitian

Penelitian ini dimulai dengan mengumpulkan dataset citra, dataset citra diperoleh dari 2 sumber yaitu mengambil dari galery dan melakukan pengambilan foto secara langsung dengan parameter yang telah ditentukan, lihat Tabel 2. Data terdiri dari 3 kelas. Proses pengambilan foto dilakukan 2 kali. Keseluruhan dataset yang diperoleh sebanyak 309 citra, lihat Tabel 3, 4 dan 5. Dari data yang diperoleh dibagi menjadi 2 kelompok, sebagai dataset dan citra baru. Citra baru merupakan data citra diluar data train dan data test. Berdasarkan dataset yang digunakan, proses input data dilakukan sebanyak 3 kali :

1. Data dengan pengambilan proses ke-1 sebanyak 50 data setiap kelas (150 data). Karena hasil akurasi yang diperoleh dibawah 90%, maka dilakukan perbaikan dataset citra.

2. Data dengan pengambilan proses ke-2 sebanyak 50 data setiap kelas. Karena hasil akurasi yang diperoleh lebih baik dari sebelumnya, maka untuk melihat pengaruh jumlah dataset terhadap akurasi yang diperoleh, kemudian dataset proses ke-2 diinputkan secara bertahap, yaitu 10, 20, 30, 40 dan 50 data dari masing-masing kelas.
3. Data citra baru, yang kemudian diklasifikasi dengan model hasil training data proses ke-2 dengan jumlah 10, 20, 30, 40 dan 50 data dari masing-masing kelas.



Gambar 2: Skema Metode Penelitian

Dari masing-masing dataset dan citra baru yang diinputkan kemudian dilakukan proses deteksi wajah dan *cropping* citra dengan menggunakan algoritma *convolutional neural network* pada *CAFFE framework*. Kemudian dilakukan *feature extraction* dengan metode sobel untuk memperoleh garis tepi citra dengan menggunakan persamaan (1), (2) dan (3). Hasil dari *feature extraction*, kemudian dataset dilakukan split data dengan metode holdout validation

dengan perbandingan data train 60% dan data test 40%[16]. Dari data train tersebut selanjutnya dilakukan training dengan algoritma *support vector machine* penyelesaian linear dengan menggunakan persamaan (4) dan (5). Dari hasil training dataset, masing-masing input diperoleh model hasil training. Hasil training model tersebut kemudian digunakan untuk melakukan klasifikasi data test dan citra baru. Proses klasifikasi dengan metode One Againts All, karena kelas yang digunakan lebih dari satu. Proses klasifikasi menggunakan persamaan (6). Dari hasil klasifikasi kemudian dilakukan evaluasi dengan metode *confusion matrix* sehingga masing-masing data yang diinputkan memperoleh *precision*, *recall* dan *accuracy*. Gambar 2 adalah skema metode penelitian yang dilakukan.

Hasil dan Diskusi

Mengumpulkan Dataset dan Citra Baru

Dalam mengumpulkan dataset dan citra baru, data citra diperoleh secara langsung (foto) dan tidak langsung(galery). Proses pengambilan dilakukan 2 kali. Hal ini dilakukan karena dengan data proses ke-1 akurasi yang diperoleh kecil dan hasil *edge detection* kurang bagus. Tabel 2 adalah parameter yang digunakan dalam proses pengambilan foto.

Tabel 2: Parameter Pengambilan Dataset Secara Langsung (Foto)

Proses	t(wajah)	t(kamera)	s	E_v	Resolusi
1	1,2m	1,2m	1,5m	500-1000 lux	8Mp
2	1,2m	1,2m	1m	3000-4000 lux	8Mp

Keterangan,

t(wajah) : tinggi wajah dari permukaan tanah

t(kamera) : tinggi kamera dari permukaan tanah

s : Jarak kamera dengan wajah

E_v : Luminans (Pencahayaan)

Resolusi : Resolusi Kamera yang digunakan

Pada pengambilan dataset proses ke-1, dihasilkan deskripsi dataset seperti disajikan pada Tabel 3.

Tabel 3: Deskripsi Dataset Proses Ke-1

No	Kelas (Nama)	Data Wajah	Dimensi (px)	Sumber
1	Edi	50	2448 × 3264	Langsung
2	Rochim	50	476 × 470 s/d 2048 × 2048	Tidak Langsung
3	Syarif	50	2448 × 3264	Langsung

Pada Kelas Rohim data diperoleh dari galery sehingga dimensi tidak sama. Kemudian Pada pengambilan dataset proses ke-2, dengan deskripsi dataset seperti tampak pada Tabel 4.

Tabel 4: Deskripsi Dataset Proses Ke-2

No	Kelas (Nama)	Data Wajah	Dimensi (px)	Sumber
1	Edi	50	2448 × 3264	Langsung
2	Rochim	50	2449 × 3264	Langsung
3	Syarif	50	2448 × 3264	Langsung

Data citra baru diperoleh dengan proses pengambilan data secara langsung tanpa parameter. Tabel 5 adalah deskripsi citra baru yang diperoleh.

Tabel 5: Deskripsi Dataset Citra Baru

No	Kelas (Nama)	Data Wajah	Dimensi (px)	Sumber
1	Edi	3	2448 × 3264	Langsung
2	Rochim	3	2449 × 3264	Langsung
3	Syarif	3	2448 × 3264	Langsung

Berdasarkan Tabel 3, 4 dan 5, jumlah dataset yang digunakan sebanyak 309 data citra. Gambar 3 sampai 5 adalah sampel dataset citra dan citra baru yang diperoleh.



Gambar 3: Sampel Dataset Proses Ke-1



Gambar 4: Sampel Dataset Proses Ke-2

digunakan adalah *CAFFE framework*, yang merupakan jaringan mirip dengan ResNet-10 serta menggunakan 140.000 gambar ukuran 300×300 untuk melatih model[18]. Proses deteksi dan crop citra wajah dilakukan menggunakan bahasa pemrograman python dan library OpenCV. Hasil dari proses deteksi dan crop citra wajah terlihat pada Gambar 6, dimana baris ke-1 merupakan citra kelas Edi, baris ke-2 kelas Rohim dan Baris ke-3 kelas Syarif.



Gambar 5: Sampel Dataset Citra Baru

Dari masing-masing sample citra, citra kiri merupakan kelas Edi, citra tengah merupakan kelas Rohim dan citra kanan merupakan kelas Syarif.

Edge Detection

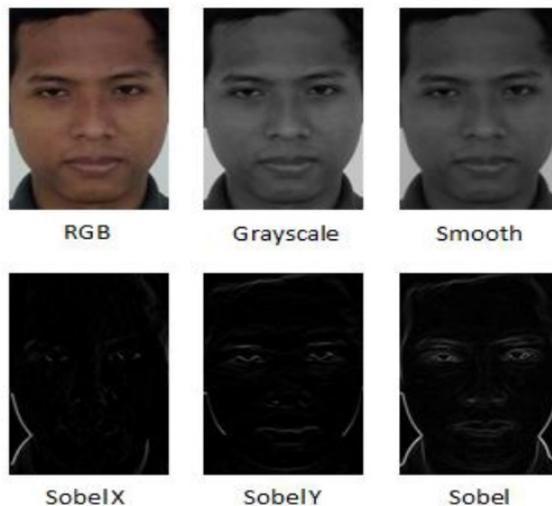
Setelah diperoleh citra wajah, selanjutnya dilakukan proses *edge detection* yang terdiri dari 3 tahap, yaitu transformasi warna RGB ke *grayscale*(G), filter untuk smoothing citra dan filter untuk deteksi tepi dengan sobel. Proses transformasi warna seperti yang dilakukan oleh Mai dan temannya[19], yaitu dengan mengalikan yaitu mengalikan setiap nilai pixel red(R), green(G) dan blue(B), dengan nilai berikut: $G=0,2989 \times R + 0,5870 \times G + 0,1140 \times B(10)$. Gambar 7 adalah sampel hasil proses *edge detection* citra.



Gambar 6: Sampel Dataset Citra Wajah (Proses Ke-2)

Deteksi dan Crop Citra Wajah

Dari dataset yang diperoleh kemudian dilakukan proses deteksi objek wajah dengan menggunakan pre-trained face detector model yang telah dibuat dengan *SSD framework* (Single Shot MultiBox Detector)[17]. SSD yang



Gambar 7: Sampel Proses *Edge Detection*

Kemudian proses smoothing citra dilakukan dengan melakukan *convolution* menggunakan filter dengan kernel 5×5 yang dibentuk menggunakan Gaussian, persamaan (3). Proses deteksi tepi citra dengan melakukan *convolution* menggunakan filter operator sobelX dan sobelY, kemudian dilakukan perhitungan titik gradien dari kedua sumbu.

Split Dataset

Proses split dataset dilakukan dengan menggunakan metode holdout validation, dimana sampel data test diambil secara acak. Kemudian perbandingan yang digunakan adalah 60:40 dimana 60% sebagai data training dan 40% sebagai data testing.

Training dan Klasifikasi

Proses training dan klasifikasi citra dilakukan dengan algoritma *support vector machine* dengan penyelesaian linear. Adapun metode multiclass yang digunakan adalah one against all, sehingga setiap input dataset proses training dilakukan sebanyak 3 kali yaitu, kelas Edi vs kelas Rochim dan Syarif, kelas Rochim vs kelas Edi dan Syarif dan kelas Syarif vs kelas Edi dan Rochim. Setiap proses kelas target dijadikan nilai positif(1) dan kelas bukan target dijadikan nilai negatif(-1) sebagai contoh kelas Edi (1) vs kelas Rochim dan Syarif (-1).

Dalam melakukan training, nilai pixel setiap citra digunakan sebagai perhitungan. Setelah melakukan proses *edge detection*, nilai pixel citra yang sebelumnya tiga dimensi menjadi dua dimensi. Dari dua dimensi tersebut kemudian dikonversi menjadi satu dimensi dengan bahasa pemrograman python. Hasil konversi pixel tersebut digunakan sebagai nilai x dan kelas (nama) digunakan sebagai nilai y, yang kemudian dimasukan ke dalam algoritma *support vector machine* untuk mendapatkan nilai w dan b. Nilai w dan b digunakan pada persamaan (6) untuk melakukan klasifikasi.

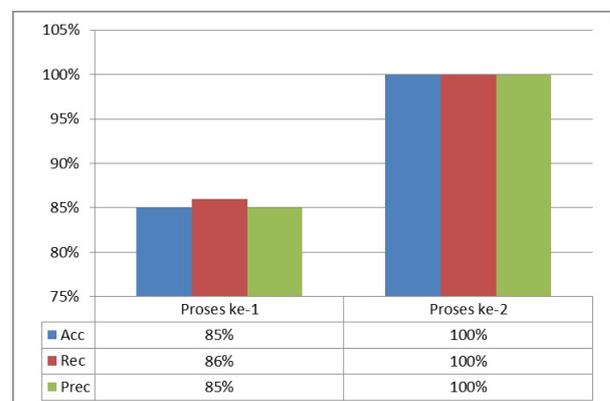
Proses klasifikasi baik data test maupun citra baru dilakukan dengan menggunakan persamaan (6). Dengan persamaan tersebut dapat mengklasifikasi data test yang diinputkan, apakah termasuk ke dalam kelas negatif atau kelas positif berdasarkan aturan one against all. Dimisalkan data test yang diinputkan adalah kelas Edi, kemudian nilai x kelas Edi tersebut dilakukan proses perhitungan dengan persamaan (6) dengan nilai w dan b hasil setiap proses perhitungan one against all masing-masing kelas dan menentukan dikelas mana data test yang diinputkan tersebut bernilai positif.

Proses perhitungan training dan klasifikasi dilakukan dengan menggunakan bahasa pemrograman python dengan library scikit-learn.

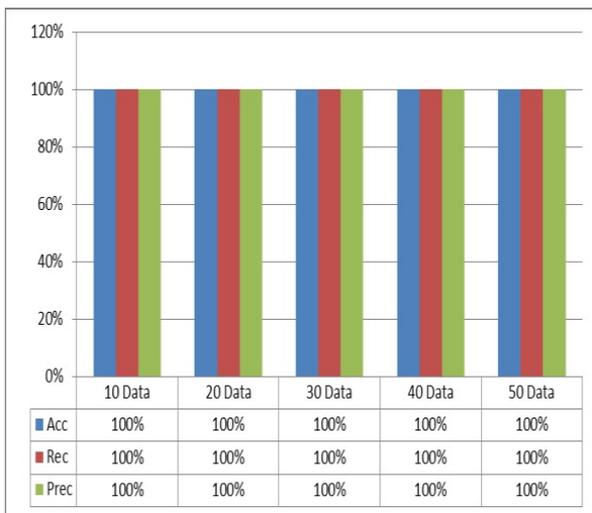
Evaluasi Klasifikasi

Setelah dilakukan klasifikasi data test dan citra baru, dilakukan evaluasi hasil klasifikasi dengan metode confusion matrix. Didalam penelitian ini terdiri dari 3 evaluasi klasifikasi. Evaluasi pertama, merupakan hasil klasifikasi data test dengan menggunakan dataset citra pengambilan proses ke-1 dengan input dataset sebanyak 50 citra setiap kelas. Hasil klasifikasi kemudian dilakukan perhitungan menggunakan persamaan (7), (8) dan (9), sehingga mendapatkan nilai *precision* sebesar 85%, *recall* sebesar 86% dan *accuracy* sebesar 85%.

Evaluasi kedua merupakan hasil klasifikasi data test dengan menggunakan dataset citra pengambilan proses ke-2 dengan input dataset sebanyak 50 citra setiap kelas. Berdasarkan hasil klasifikasi mendapatkan nilai *precision* sebesar 100%, *recall* sebesar 100% dan *accuracy* sebesar 100%. Perbandingan evaluasi pertama dan kedua tampak seperti pada Gambar 8. Kemudian dari hasil evaluasi yang dilakukan, dengan dataset ke-2 mendapatkan akurasi yang lebih baik dari dataset ke-1. Oleh karena itu, untuk melihat hasil evaluasi berdasarkan jumlah dataset setiap kelas, maka dari dataset ke-2 dilakukan input secara bertahap, yaitu 10, 20, 30, 40 dan 50 data setiap kelas. Hasil evaluasi klasifikasi pada setiap tahap mendapatkan hasil yang sama, yaitu mendapatkan nilai *precision* sebesar 100%, *recall* sebesar 100% dan *accuracy* sebesar 100%. Perbandingan evaluasi klasifikasi berdasarkan perbedaan jumlah dataset yang diinputkan tampak seperti pada Gambar 9.

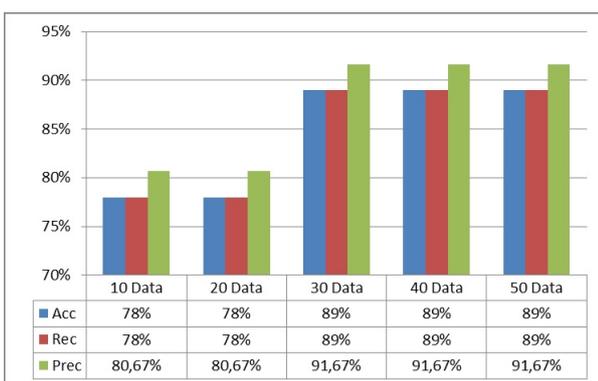


Gambar 8: Hasil Evaluasi Klasifikasi Berdasarkan Proses Pengambilan Dataset



Gambar 9: Hasil Evaluasi Klasifikasi Berdasarkan Jumlah Dataset Setiap Kelas

Evaluasi ketiga merupakan hasil klasifikasi dengan menggunakan data citra baru, dimana data training yang digunakan menggunakan dataset ke-2 yang diinputkan secara bertahap. Data citra baru yang digunakan sebanyak 9 data, dengan deskripsi seperti pada Tabel 5. Hasil evaluasi klasifikasi yang diperoleh dengan data training 10 dan 20 data mendapatkan nilai *precision* sebesar 78%, *recall* sebesar 78% dan *accuracy* sebesar 80,67%. Kemudian dengan data training 30, 40 dan 50 data mendapatkan nilai *precision* sebesar 91,67%, *recall* sebesar 89% dan *accuracy* sebesar 89%. Perbandingan hasil evaluasi klasifikasi citra baru tampak pada Gambar 10.



Gambar 10: Hasil Evaluasi Klasifikasi Input Citra Baru

Diskusi

Berdasarkan hasil penelitian yang dilakukan, evaluasi klasifikasi berdasarkan proses pengambilan dataset mengalami peningkatan, hal itu

dapat dibuktikan dengan nilai akurasi yang diperoleh. Dengan data proses ke-1 mendapatkan akurasi sebesar 85%. Setelah dilakukan perbaikan parameter proses pengambilan data (luminans dan jarak) serta menyeragamkan sumber data yang digunakan, akurasi yang diperoleh menjadi 100% meningkat sebesar 15%.

Kemudian dengan menggunakan data pengambilan proses ke-2, hasil evaluasi klasifikasi berdasarkan perbedaan jumlah dataset setiap kelas tidak mengalami perubahan. Hal itu dapat dibuktikan dengan hasil akurasi, *recall* dan *precision* adalah sama, yaitu 100%. Setelah dilakukan proses analisa, terdapat kemungkinan akurasi tersebut dapat diperoleh. Bahwa, setiap kelas memiliki nilai bobot yang kuat karena data test yang digunakan dari sumber data dan parameter yang sama dengan data train, meskipun dengan waktu pengambilan yang berbeda. Hal itu dapat dilihat setelah dilakukan klasifikasi data dengan menggunakan citra data baru.

Tabel 6: Pebandingan Hasil Deteksi Garis Tepi Berdasarkan Proses Pengambilan Citra

Proses	Citra	
	RGB	Hasil garis tepi
Ke-1		
ke-2		

Hasil evaluasi klasifikasi yang telah dilakukan dengan data citra baru, akurasi yang diperoleh dengan menggunakan 10 dan 20 data setiap kelas adalah 78%, *recall* 78% dan *precision* 80,67%. Kemudian akurasi yang diperoleh dengan menggunakan 30, 40 dan 50 data setiap kelas adalah 89%, *recall* 89% dan *precision* 91,67%. Sehingga berdasarkan hasil evaluasi yang diperoleh, dengan bertambahnya data latih setiap kelas dapat menambah *accuracy* dalam mengenali wajah. Selain itu, hasil deteksi garis tepi dengan citra yang diper-

oleh pada proses ke-2 lebih jelas dan sedikit noise dari proses ke-1. Hal itu terjadi setelah melakukan perbaikan luminans dan jarak pengambilan foto terhadap objek wajah. Perbandingan hasil deteksi garis tepi berdasarkan proses pengambilan citra tampak pada Tabel 6.

Tabel 7: Pebandingan Hasil Deteksi Garis Tepi Berdasarkan Sumber Data Citra

Sumber	Citra	
	RGB	Hasil garis tepi
Galery		
Foto Langsung		

Berdasarkan sumbernya, hasil deteksi garis tepi dengan melakukan pengambilan data citra secara langsung(foto) lebih tajam dan sedikit noise dari pada data citra yang berasal dari galery. Karena citra dari galery telah mengalami proses editing seperti smoothing, contrast, saturation dan lainnya. Perbedaan tersebut dapat dilihat dari sampel citra RGB (*galery*) sedikit tidak jelas berbeda dengan foto langsung. Begitu pula hasil *edge detection* yang diperoleh. Perbandingan hasil deteksi garis tepi berdasarkan sumber data citra tampak pada Tabel 7.

Penutup

Pengenalan wajah merupakan salah satu sistem identifikasi yang dikembangkan berdasarkan perbedaan ciri wajah seseorang mulai dari mendeteksi objek wajah kemudian mengklasifikasikannya. Dalam penelitian ini, mengusulkan metode pengenalan wajah dengan proses deteksi wajah menggunakan *caffe framework*, ekstraksi fitur menggunakan sobel *edge detection* dan klasifikasi menggunakan algoritma *support vector machine* dengan penyelesaian linear. Dengan metode yang diusulkan mendapatkan *accuracy* terbaik diperoleh sebe-

sar 100% setelah dilakukan perbaikan kualitas dataset citra. Kemudian berdasarkan perbedaan jumlah data perkelas, hasil *accuracy* yang diperoleh sama yaitu 100%. Dengan input citra baru, *accuracy* dengan jumlah data latih 30, 40 dan 50 data setiap kelas sebesar 89% lebih tinggi dari jumlah data latih 10 dan 20 data dengan *accuracy* 70%. Hal ini dapat dikatakan metode yang diusulkan cukup baik.

Dalam penelitian ini masih memiliki banyak kekurangan, oleh sebab itu untuk penelitian selanjutnya peneliti memberikan beberapa saran:

1. Melakukan penelitian untuk memperbaiki kemampuan hasil *edge detection* terhadap pengaruh kualitas citra berdasarkan parameter luminans dan jarak yang digunakan.
2. Setelah dilakukan input citra baru diperoleh klasifikasi yang tidak tepat. Perlu melakukan penelitian dengan hasil citra yang diperoleh menggunakan pendekatan teknis yang lain (jarak, luminans) serta memperhitungkan pengaruh ekspresi wajah pada dataset citra yang digunakan, terhadap hasil klasifikasi atau *accuracy* yang diperoleh.
3. Melakukan implementasi kedalam sistem secara real, seperti sistem keamanan rumah atau sistem autentikasi.

Daftar Pustaka

- [1] M. Athoillah, "Pengenalan Wajah Menggunakan SVM Multi Kernel dengan Pembelajaran yang Bertambah", JOIN (Jurnal Online Informatika), vol. 2, no. 2, hlm. 84–91, doi: 10.15575/join.v2i2.109, 2017.
- [2] R. A. Nugrahaeni and K. Mutijarsa, "Comparative analysis of machine learning KNN, SVM, and random forests algorithm for facial expression classification", dalam 2016 International Seminar on Application for Technology of Information and Communication (ISEMANTIC), Semarang, Indonesia, hlm. 163–168, doi: 10.1109/ISEMANTIC.2016.7873831, 2016.
- [3] E. Kremic and A. Subasi, "Performance of Random Forest and SVM in Face Recognition", The International Arab Journal of Information Technology, vol. 13, no. 2, hlm. 287–293, 2016.

- [4] B. Anand and P. K. Shah, "Face Recognition using SURF Features and SVM Classifier", *International Journal of Electronics Engineering Research.*, vol. 8, no. 1, hlm. 1–8, 2016.
- [5] G. N. Chaple, R. D. Daruwala, and M. S. Gofane, "Comparisons of Robert, Prewitt, Sobel operator based edge detection methods for real time uses on FPGA", 2015 *International Conference on Technologies for Sustainable Development (ICTSD)*, Mumbai, India, hlm. 1–4, doi: 10.1109/ICTSD.2015.7095920, 2015.
- [6] S. Das, "Comparison of Various Edge Detection Technique", *International Journal of Signal Processing, Image Processing and Pattern Recognition*, vol. 9, no. 2, hlm. 143–158, doi: 10.14257/ijsp.2016.9.2.13, 2016.
- [7] S. Kaur and I. Singh, "Comparison between Edge Detection Techniques", *International Journal of Computer Applications*, vol. 145, no. 15, hlm. 15–18, doi: 10.5120/ijca2016910867, 2016.
- [8] S. K. Katiyar dan P. V. Arun, "Comparative analysis of common edge detection techniques in context of object extraction", *IEEE TGRS*, vol. 15, no. 11 b, hlm. 68–79, 2014.
- [9] D. E. Pratiwi dan A. Harjoko, "Implementasi Pengenalan Wajah Menggunakan PCA (Principal Component Analysis)", *IJEIS*, vol. 3, no. 2, hlm. 175–184, 2013.
- [10] S. Khan, H. Rahmani, S. A. A. Shah, dan M. Bennamoun, "A guide to convolutional neural networks for computer vision", *Synthesis Lectures on Computer Vision*, vol. 8, no. 1, hlm. 1–207, 2018.
- [11] Y. Jia dkk, "Caffe: Convolutional architecture for fast feature embedding", dalam *Proceedings of the 22nd ACM international conference on Multimedia*, hlm. 675–678, 2014.
- [12] M. S. Nixon dan A. S. Aguado, "Feature Extraction and Image Processing", dalam *Academic Press*, 2 ed, 2008.
- [13] S. Raschka, "Model Evaluation, Model Selection, and Algorithm Selection in Machine Learning", arXiv:1811.12808 [cs, stat], 2018.
- [14] V. Kecman dan L. Wang, "Support vector machines: theory and applications", Springer, 2005.
- [15] P. N. Andono, T. Sutojo, dan Muljono, "Pengolahan Citra Digital", Penerbit Andi, 2017.
- [16] J. Awwalu dan O. Francisca, "On Holdout and Cross Validation: A Comparison between Neural Network and Support Vector Machine", *International Journal of Trend in Research and Development*, vol. 6, no. 2, hlm. 2394–9333, 2019.
- [17] G. Balu, "Computer Vision Exercises. Contribute to thegopieffect/computer_vision development by creating an account on GitHub", https://github.com/thegopieffect/computer_vision, 2019.
- [18] M. Jangid, P. Paharia, and S. Srivastava, "Video-Based Facial Expression Recognition Using a Deep Learning Approach", dalam *Advances in Computer Communication and Computational Sciences*, Springer, 2019, hlm. 653–660, 2019.
- [19] V. D. Mai, D. Q. Miao, and R. Z. Wang, "An Improved Method for Vietnam License Plate Location based on Mathematic Morphology and Measuring Properties of Image Regions", *AMM*, vol. 105–107, no. 1, hlm. 1995–1999, 2011.