

# Deteksi Kantuk pada Pengemudi Berdasarkan Penginderaan Wajah Menggunakan PCA dan SVM

Nur Ramadhani L. Q.<sup>1</sup>, Efri Suhartono<sup>1</sup>, Suci Aulia<sup>2</sup>, Sugondo Hadiyoso<sup>2</sup>

<sup>1</sup>Fakultas Teknik Elektro, Universitas Telkom

<sup>2</sup>Fakultas Ilmu Terapan, Universitas Telkom

Jl. Telekomunikasi No. 1, Terusan Buah Batu, Bandung 40257

e-mail: [suciaulia@telkomuniversity.ac.id](mailto:suciaulia@telkomuniversity.ac.id)

**Abstrak**—Mengantuk saat berkendara merupakan salah satu penyebab utama kecelakaan lalu lintas karena mengganggu tingkat fokus pengemudi. Oleh karena itu, diperlukan sebuah mekanisme deteksi kantuk otomatis pada pengemudi untuk memberikan peringatan atau alarm sehingga dapat terhindar dari kecelakaan. Pada penelitian ini, kami mendesain dan mensimulasikan sebuah sistem untuk mendeteksi kantuk melalui ekspresi menguap pada pengemudi. Akuisisi dilakukan dengan merekam wajah dari dua titik pengambilan gambar mencakup *dashboard* dan *mirror* depan di dalam mobil. Dari rekaman video tersebut, kemudian diambil menjadi beberapa citra dengan ukuran 128x82 piksel yang digunakan sebagai data pelatihan dan pengujian. Gambar ini kemudian diproses menggunakan *Principal Component Analysis* (PCA) untuk ekstraksi ciri dan diklasifikasikan menggunakan *Support Vector Machine* (SVM). Dari pengujian yang dilakukan, sistem menghasilkan akurasi tertinggi sebesar 98%. Performa terbaik ini diperoleh dengan SVM *kernel polynomial* pada posisi kamera di *dashboard*. Sementara itu, berdasarkan pengujian kompresi, citra yang masih dapat memenuhi kebutuhan sistem adalah 25% dari ukuran asli. Usulan metode deteksi kantuk pada studi ini diharapkan dapat diaplikasikan untuk deteksi kantuk secara real-time pada kendaraan.

**Kata kunci:** *deteksi kantuk, otomatis, pca, svm*

**Abstract**—Drowsiness while driving is one of the main causes of traffic accidents it affects the level of focus of the driver. Therefore, we need an automatic drowsiness detection mechanism for the driver to provide a warning or alarm so that an accident can be avoided. In this study, we design and simulate a system to detect drowsiness through the driver's yawn expression. The acquisition is made by recording the face from two shooting points including the dashboard and front mirrors in the car. From the video recording, then it is taken into several images with a size of 128x82 pixels which are used as training and testing data. This image is then processed using Principal Component Analysis (PCA) for feature extraction and classified using a Support Vector Machine (SVM). From the tests carried out, the system generates the highest accuracy of 98%. This best performance is obtained by SVM with polynomial kernel in the camera position on the dashboard. Meanwhile, based on compression testing, the image that can still meet system requirements is 25% of the original size. It is hoped that the proposed drowsiness detection method in this study can be applied for real-time drowsiness detection in vehicles.

**Keywords:** *drowsiness detection, automatic, pca, svm*

Copyright © 2021 Jurnal Rekayasa ElektriKa. All right reserved

## I. PENDAHULUAN

Salah satu faktor penyebab dari berkurangnya konsentrasi dari pengemudi adalah rasa kantuk. Rasa kantuk dapat membuat konsentrasi pengemudi menjadi menurun, terutama ketika jalanan sedang padat karena kemacetan yang panjang. Banyak pengemudi yang lebih memilih mengabaikan rasa kantuknya dan meneruskan perjalanannya hingga sampai tujuan, padahal hal ini dapat mengancam keselamatan pengemudi. Beberapa tahun terakhir ini, konsentrasi yang kurang saat sedang mengemudi menjadi salah satu penyebab terjadinya banyak kecelakaan [1].

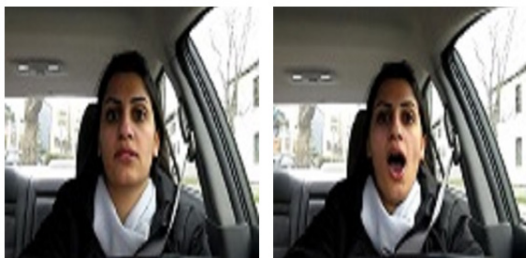
Sehingga pada penelitian ini diusulkan deteksi kantuk melalui ekspresi wajah pada pengendara sebagai

pencegahan kecelakaan. Ada beberapa metode yang biasa digunakan pada sistem pengenalan ekspresi wajah. Pada tahun 2015, penelitian yang dilakukan oleh Siddiqi, dkk. [2], melokalisasi ekspresi wajah menggunakan metode *stepwise linear discriminant analysis* dan *hidden conditional random field* dengan nilai akurasi 96,37%, namun dari hasil yang dicapai tersebut masih memiliki keterbatasan yaitu belum direalisasikan dengan *real environment* karena baru menggunakan 6 citra wajah. Pada tahun 2017, penelitian yang dilakukan oleh Qayyum, dkk. [3], menggunakan *stationary wavelet transform* sebagai ekstraksi ciri dan *neural network* sebagai klasifikasi antara wajah normal dan wajah berekspresi berdasarkan perubahan otot pada wajah, dengan nilai akurasi 94,28%. Azmi, dkk. [4] menggunakan metode *local*

*gabor binary pattern* sebagai ekstraksi ciri dan metode *K-nearest neighbors* dengan nilai akurasi tertinggi pada 96,25% dimana *local gabor binary pattern* merupakan penggabungan antara metode gabor dan metode *Local Binary Pattern* (LBP). Pada penelitian ini dicapai tingkat akurasi yang tinggi (96,25%), namun ekspresi wajah yang dideteksi terbatas pada ekspresi marah, sedih, menjijikan, bahagia, terkejut, takut, dan netral, sedangkan ekspresi wajah mengantuk belum ada.

Penelitian lainnya dengan *dataset* yang sama namun disimulasikan menggunakan metode yang berbeda, pertama adalah penelitian yang dilakukan oleh Kumari, dkk. [5] menggunakan metode LBP, *local gradient code*, dan *local directional pattern* dengan klasifikasi *K-nearest neighbor* dengan menggunakan *dataset* JAFFE mencapai *recognition rate* 88.26 %. Berikutnya studi oleh Chao, dkk. [6], menggunakan metode *specific expression LP* sebagai ekstraksi ciri dan *Support Vector Machine* (SVM) sebagai klasifikasi mencapai 94.88 %. Ketiga oleh Carcagni, dkk. [7], menggunakan metode *Histogram Of Oriented Gradients* HOG sebagai ekstraksi ciri dan SVM sebagai klasifikasi mencapai 72.2%. Keempat oleh Hasani, dkk. [8], menggunakan penggabungan metode LBP, HOG, dan *histogram of optical flow* dengan klasifikasi SVM.

Studi oleh Ghimere, dkk. [9], menggunakan metode LBP sebagai ekstraksi ciri dan SVM sebagai klasifikasi mencapai akurasi tertinggi 97.75%. Studi lainnya oleh Lopes, dkk. [10], mengusulkan metode *Convolutional Neural Network* (CNN) dengan nilai akurasi tertinggi pada 96,76%. Studi oleh Zeng, dkk. [11], membandingkan HOG dan LBP (sebagai ekstraksi ciri dan *deep sparse autoencoders* sebagai klasifikasi dengan nilai akurasi tertinggi pada 95.79 %. Sementara itu, studi oleh Farooq, dkk. [12], mengusulkan *Principal Component Analysis* (PCA) dan *Independent Component Analysis* (ICA) sebagai ekstraksi ciri dan metode *self organizing maps* sebagai klasifikasi dengan nilai akurasi tertinggi pada 96,55%. Penelitian lainnya dilakukan oleh De, dkk. [13], menggunakan metode PCA untuk mendeteksi ekspresi wajah terkejut, takut, marah, dan bahagia mencapai rata-rata *recognition rate* 85.38%. Berdasarkan beberapa review penelitian di atas, deteksi ekspresi mengantuk belum pernah dilakukan. Zhang, dkk. [14], menambahkan deteksi ekspresi mengantuk dengan menggunakan *Scale Invariant Feature Transform* (SIFT) metode sebagai ekstraksi ciri dan metode *Deep Neural Network* (DNN) sebagai klasifikasi mencapai rata-rata *recognition rate* 85.52%.



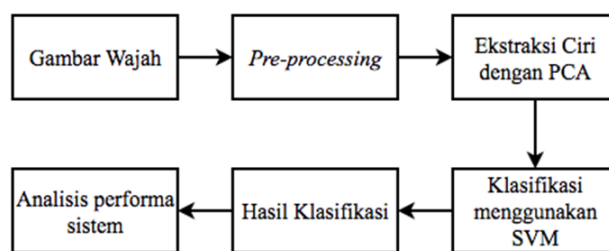
Gambar 1. Salah satu dataset pada *Yawning Detection Dataset* (YawDD)

Dari beberapa diskripsi penelitian tersebut, sejauh pengetahuan kami, belum dilakukan deteksi ekspresi wajah yang secara khusus ditujukan manfaatnya untuk keamanan berkendara. Maka pada penelitian ini diusulkan suatu sistem deteksi mengantuk berdasarkan ekspresi mengantuk pada pengemudi. Tujuannya adalah untuk membangun sistem otomasi deteksi mengantuk pada pengemudi sebagai salah satu tindakan preventif dalam mencegah kecelakaan akibat mengantuk saat berkendara. Pada penelitian ini digunakan citra wajah yang diambil dari dua posisi pengambilan, yaitu kamera pada posisi *dashboard* dan kamera pada posisi *mirror* depan. Pada penelitian ini metode PCA digunakan untuk proses ekstraksi ciri dengan pendekatan nilai *eigen*. Wajah dapat dengan mudah direkonstruksi dengan hanya mempertimbangkan sejumlah kecil informasi yang dapat diperoleh dengan menggunakan *eigenface* [15], [16] yang berhasil mencapai *detection rate* 92.93%. Sedangkan untuk metode klasifikasi, pada penelitian ini menggunakan SVM karena memiliki rata-rata tingkat akurasi tinggi seperti pada *review* penelitian di atas.

## II. MATERIAL DAN METODE

Sistem ini dirancang untuk mengenali ekspresi wajah dari suatu citra wajah menggunakan metode PCA dengan metode klasifikasi SVM. Sistem ini menggunakan *dataset* dari *Yawning Detection Dataset* (YawDD) [14]. YawDD merupakan sebuah *dataset* yang berisi dua set data video pengemudi yaitu pada posisi kamera berapa pada *mirror* dan posisi kamera berapa pada *dashboard* dengan berbagai karakteristik wajah, yang akan digunakan untuk pengujian algoritma dan model untuk sebagian besar deteksi mengantuk. Gambar 1 merupakan contoh gambar wajah yang digunakan pada studi ini. Setiap orang memiliki tiga atau empat video dan setiap video berisi kondisi mulut yang berbeda seperti normal dan mengantuk. *Dataset* ini menyediakan 322 video yang terdiri dari *driver* pria dan wanita [17]. Secara umum terdapat beberapa tahap proses dalam usulan sistem ini sebagaimana ditunjukkan pada Gambar 2.

Sinyal masukan merupakan citra wajah dari *dataset* YawDD yang akan dilanjutkan dengan *pre-processing*. Proses ini meliputi proses *cropping*, *resize*, dan kompresi. Proses *pre-processing* selanjutnya adalah mengubah citra hasil akuisisi menjadi citra *grayscale*. Citra keabuan tadi akan menjadi masukan untuk proses ekstraksi ciri menggunakan PCA, bagian wajah yang dilokalisasi oleh PCA disini adalah bagian mulut. Selanjutnya, hasil



Gambar 2. Sistem deteksi ekspresi wajah

ekstraksi ciri berupa *vector* ciri akan diklasifikasikan menggunakan metode SVM.

Pada penelitian ini, sistem yang dirancang terdiri dari dua proses yaitu *training* dan *testing*. Pada proses pelatihan data atau *training*, sistem menerima data yang kemudian akan disimpan dan dijadikan sebuah *database*. Pada proses pengujian, sistem akan menerima sebuah data dan melakukan proses teknik pengenalan ekspresi yang kemudian mengeluarkan sebuah teks berdasarkan ekspresi wajah yang terdapat pada citra uji.

#### A. Metode PCA

PCA merupakan sebuah teknik yang digunakan untuk menganalisa data dalam sebuah data. Metode PCA sering digunakan dalam pengenalan wajah, prediksi, dan lain-lain [18]. Metode ini berfokus untuk mereduksi data [19]. PCA akan mencari pola dan mengambil ciri dari data citra yang berdimensi tinggi dari sebuah *dataset* latih yang kemudian direduksi dari citra yang berdimensi tinggi menjadi citra berdimensi rendah [20]. *Eigenface* merupakan sebuah metode yang biasa digunakan untuk sistem pengenalan wajah. Metode *eigenface* merupakan komponen utama PCA untuk *training* awal pada *dataset* citra wajah. Setiap citra wajah dapat dilihat sebagai sejumlah vektor ( $N$ ) berdasarkan lebar dan tingginya ( $N = w \times h$  piksel). Citra wajah direpresentasikan sebagai vektor kolom  $N$  dimensi atau matriks  $N \times 1$ .

Tahapan yang dilakukan dalam proses PCA *Eigenface* adalah sebagai berikut [21]:

1. Citra wajah dengan ukuran dan format yang sama.

Kumpulan dari citra wajah *training set* sebanyak  $M$  citra wajah dengan lebar dan tinggi berturut-turut  $w$  dan  $h$ . Masing-masing citra wajah *training set* direpresentasikan sebagai vektor baris dimensi  $N$ , dimana  $N = w \times h$ . sekumpulan citra wajah dalam *training set* dapat dituliskan sebagai  $T_i = \{T_1, T_2, T_3, \dots, T_M\}$ , dimana setiap  $T_i$  adalah vektor dari dimensi  $N$  dan  $M$  adalah jumlah citra wajah dalam *training set*.

2. Mencari *average face*.

*Average face* adalah rata-rata piksel dari semua citra *training* yang ditunjukkan pada Persamaan 1 dan hasil matriksnya pada Persamaan 1, dimana  $m$  adalah jumlah citra pelatihan dengan indeks  $i$ .

$$\Psi = \frac{1}{m} \sum_{i=1}^m \Gamma_i, \quad (1)$$

dimana:

$\Psi$  = matrik rata-rata

$m$  = jumlah sampel citra pelatihan

$$\Psi = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_N \end{bmatrix}, \quad (2)$$

dengan mencari selisih rata-rata citra wajah (*average face*) terhadap masing-masing citra wajah dalam *training set*, didapat vektor selisih  $\{\Phi_1, \Phi_2, \Phi_3, \dots, \Phi_M\}$ .

$$\Phi_i = T_i - \Psi. \quad (3)$$

Kovarian antara dua *dataset* menunjukkan seberapa dekat keduanya berhubungan satu sama lain. Dalam hal ini kovarian matrik  $C$  merepresentasikan hubungan antara 2 matrik dan varian yang terdapat dalam *dataset*.

$$C = A \times A^T. \quad (4)$$

Dalam hal ini  $A^T$  adalah matrik *transpose* dari matrik  $A$ , dimana  $A$  adalah selisih dari masing-masing citra wajah *training set* dengan rata-rata dari citra wajah (*average face*)

3. Mencari vektor dan nilai *eigen* dari matrik kovarian  $C$ .

$$Eig\_f = A^T \times eig\_vector, \quad (5)$$

dimana  $ev$  adalah vektor eigen dan  $eval$  adalah nilai eigen. Kemudian vektor dikalikan dengan matriks  $A^T$  sehingga didapatkan *eigenface* dari citra referensi, tahap selanjutnya ialah tahap pengenalan dengan menggunakan klasifikasi.

#### B. Metode SVM

SVM adalah sebuah metode linier yang digunakan pada *feature space* berdimensi tinggi [22]. SVM banyak digunakan sebagai metode untuk mengklasifikasikan 2 buah kelas karena metode ini memiliki konsep dengan tujuan untuk menemukan *hyperplane* terbaik sehingga metode ini lebih efisien untuk dijadikan sebagai *classifier* [23], [24]. Dalam kasus pemisahan dua data, data dapat dengan mudah dipisahkan oleh satu garis dengan persamaan  $y = ax + b$ . Maka apabila data yang ingin dipisahkan lebih dari dua, kita dapat menggunakan *hyperplane*. Konsep dari *hyperplane* itu sendiri adalah dengan memisalkan  $x$  dengan  $x_1$  dan  $y$  sebagai  $x_2$ , maka kita akan dapatkan:

$$ax_1 - x_2 + b = 0, \quad (6)$$

dimana  $w$  adalah bobot vektor dan  $b$  adalah bias. Jika  $x = ($  dan  $w = (a - 1)$ , maka Persamaan (6) menjadi persamaan untuk *hyperplane* sebagai berikut:

$$x.w + b = 0. \quad (7)$$

Garis solid biru pada Gambar 3 merupakan *hyperplane* [25], yaitu garis pembatas antara kelas pertama dan kelas kedua. *Hyperplane* dapat digunakan untuk membuat prediksi sejumlah data atau kelas ( $i = 1, 2, 3, \dots, n$ ) dengan fungsi hipotesa pada persamaan berikut [26]:

$$\begin{aligned} x_i.w + b &\geq +1 \quad \text{for } y_i = +1 \\ x_i.w + b &< -1 \quad \text{for } y_i = -1. \end{aligned} \quad (8)$$

Algoritma SVM menggunakan sekumpulan fungsi matematika yang disebut sebagai kernel. Fungsi kernel merupakan suatu transformasi, yang mentransformasikan suatu data masukkan menjadi bentuk yang diperlukan. Adapun jenis kernel pada algoritma SVM adalah *kernel*

linier, kernel polynomial, kernel RBF (*Radial Basis Function*), dan Laplacian RBF. Kernel yang digunakan pada penelitian ini adalah kernel polinomial, yang ditentukan dengan Persamaan 9 berikut:

$$K(X_1, X_2) = \left\{ a + (X_1)^T, (X_1)^T \right\}^b, \quad (9)$$

dimana  $b$  adalah derajat polinomial.

### III. IMPLEMENTASI DAN PENGUJIAN

#### A. Tahap Pengujian Sistem

SVM tahap pengujian pada sistem ini terdiri dari beberapa tahapan, data diambil dari *dataset* video yang disediakan pada *Dataset YawDD* yang hanya diambil beberapa citra dari cuplikan video yang disediakan. Citra yang telah diambil disimpan menjadi sebuah *database* yang baru. Dari *YawDD*, diperoleh 400 *dataset* yang digunakan 50% sebagai data latih dan 50% sebagai data uji. Berikut ini adalah tahapan pengujian sistem:

##### 1. Tahap Pertama

Tahap pertama dalam pra-pemrosesan terdiri dari *cropping*, *resize*, dan kompresi. *Dataset* yang digunakan berasal dari *Dataset YawDD*, dimana *dataset* tersebut menyediakan video dengan 2 ekspresi yang berbeda yaitu normal dan kantuk dengan durasi video antara 1 sampai dengan 3 menit. Proses *cropping* dilakukan pada video dari *dataset YawDD*, pengambilan citra digital dilakukan dengan cara mengambil gambar pada video dengan selang waktu yang berbeda-beda.

Proses pengambilan gambar dilakukan pada saat peraga memperagakan ekspresi normal dan juga kantuk. Setelah proses *cropping* pada *dataset* selesai, dilakukan *resize*

pada citra sesuai dengan ukuran yang telah ditentukan yaitu 128x82 piksel. Proses *resize* dilakukan dengan tujuan agar ukuran data yang akan dilatih dan diuji sama. Dilakukan juga *resize* pada citra pengujian dengan ukuran 256x256 piksel, 512x512 piksel, dan 1024x1024 piksel. Tujuan dari *resize* saat akan dilakukan pengujian percobaan adalah untuk mengetahui pengaruh akurasi sistem terhadap citra yang telah di *resize*. Citra dari proses *resize* akan dikompresi sebesar 25% dan 50% menggunakan teknik kompresi JPEG. Tujuan dari proses kompresi adalah untuk mengetahui akurasi dan performansi sistem terhadap citra yang telah dikompresi. Citra yang telah dilakukan proses akusisi citra kemudian dipisahkan menjadi citra latih dan citra uji digunakan untuk tahapan *pre-processing* untuk pelatihan dan pengujian.

##### 2. Tahap Kedua

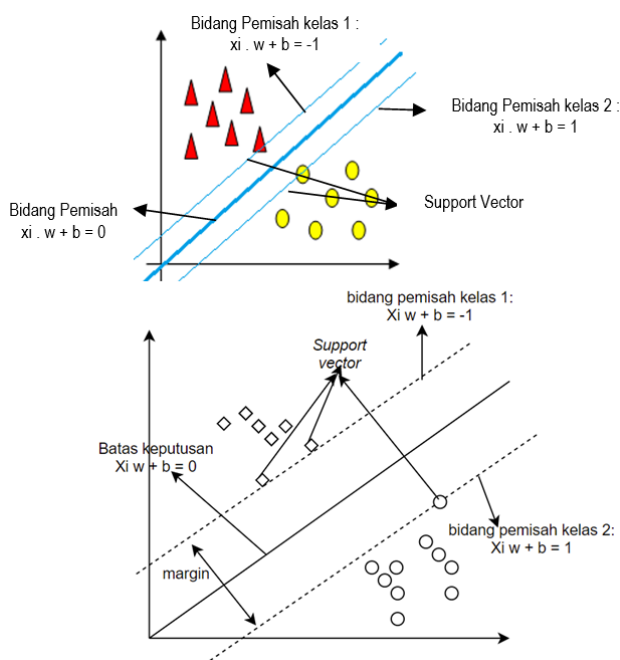
Tahap kedua dalam pra-pemrosesan gambar terdiri dari konversi ke *grayscale* dan deteksi wajah. Dilakukan pengubahan warna pada citra yang telah melalui proses akusisi citra dari gambar awal berupa RGB menjadi citra *grayscale*. Gambar 4 menunjukkan contoh citra hasil konversi dari RGB ke *grayscale*.

##### 3. Tahap Ketiga

Tahap ketiga yaitu tahap ekstraksi ciri. Citra yang telah melalui tahap *pre-processing* akan mengalami tahapan ekstraksi ciri dengan menggunakan metode PCA]]. Tahapan ini akan mendapatkan nilai *eigenface*. Data tersebut akan disimpan dalam data transformasi keseluruhan yang merupakan matriks transformasi dari citra yang tersegmentasi di dalam *boundingbox* sebagaimana ditunjukkan pada Gambar 5.

##### 4. Tahap Keempat

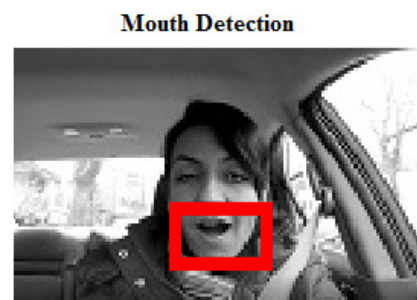
Setelah didapatkan hasil dari ekstraksi ciri yang merupakan data transformasi PCA, kemudian akan



Gambar 3. Metode klasifikasi SVM



Gambar 4. Salah satu *dataset* setelah proses konversi warna



Gambar 5. Hasil *mouth detection*



dilakukan pengklasifikasian citra menggunakan metode SVM dengan menerapkan strategi *one against one* dan *one against all*. Digunakan beberapa fungsi *kernel* pada pengujian sistem ini yaitu *kernel* RBF, linear, dan polinomial. Tahap akhir untuk pengujian sistem adalah menghitung akurasi sistem dan waktu komputasi yang optimal dengan beberapa parameter pengujian.

### B. Hasil Pengujian Sistem

#### 1. Pengujian kernel dengan parameter kernel option pada klasifikasi SVM

Pada pengujian *kernel* pada klasifikasi SVM dengan strategi OAA dan OAO dilakukan pada beberapa *kernel* yaitu *kernel* RBF, linear, dan *polynomial*. Pada pengujian *kernel* ini digunakan beberapa parameter *kernel option* untuk mencari *kernel* terbaik pada sistem ini. Parameter *kernel option* yang digunakan pada pengujian *kernel* adalah *kernel option* (derajat polinomial *b*) 9,10, dan 11.

Tabel 1. Hasil pengujian *Kernel Option* pada SVM OAA dan OAO. Berdasarkan Tabel 1 didapatkan *kernel* terbaik adalah *kernel polynomial* dengan akurasi rata-rata data uji tertinggi sebesar 84% dibandingkan dengan *kernel* RBF 51% dan *kernel* linear 49%.

#### 2. Pengujian kernel polynomial dengan menggunakan parameter kernel option pada SVM

Pada pengujian ini digunakan citra berukuran 128x82 piksel dan *kernel* yang digunakan adalah *kernel polynomial* dimana *kernel* ini merupakan *kernel* terbaik pada sistem ini berdasarkan hasil pengujian beberapa *kernel*. Pada pengujian ini digunakan beberapa *kernel option* untuk mencari *kernel option* terbaik pada sistem ini. Berdasarkan

Tabel 1. Hasil pengujian *Kernel Option* pada SVM OAA dan OAO

Jenis Kernel	Kernel Option	Posisi Kamera	Akurasi OAA	Akurasi OAO
RBF	9	Dashboard	51%	51%
		Mirror	51%	51%
	10	Dashboard	51%	51%
		Mirror	51%	51%
	11	Dashboard	51%	51%
		Mirror	51%	51%
Linear	9	Dashboard	49%	49%
		Mirror	49%	49%
	10	Dashboard	49%	49%
		Mirror	49%	49%
	11	Dashboard	49%	49%
		Mirror	49%	49%
Polynomial	9	Dashboard	49%	49%
		Mirror	91%	91%
	10	Dashboard	98%	98%
		Mirror	86%	86%
	11	Dashboard	89%	89%
		Mirror	91%	91%

Gambar 6 menunjukkan hasil pengujian *kernel polynomial* dengan parameter *kernel option* pada SVM. Citra yang diujikan berukuran 128x82 piksel, yaitu ukuran asli hasil *cropping* dengan variasi nilai *kernel option* 10, 11, dan 12.

Pengujian dilakukan pada dua grup citra *dataset* yaitu pada *dashboard* dan *mirror*. Hasilnya diperoleh tingkat akurasi tertinggi 98% pada *kernel option* 10 dengan posisi kamera pada *dashboard*. Apabila dihitung tingkat akurasi rata-rata dari pengujian pada variasi nilai *kernel option* 10, 11, dan 12, citra hasil akuisisi pada *dashboard* memiliki nilai yang lebih tinggi dibandingkan citra hasil akuisisi pada *mirror* yaitu 98% berbanding 86.67%.

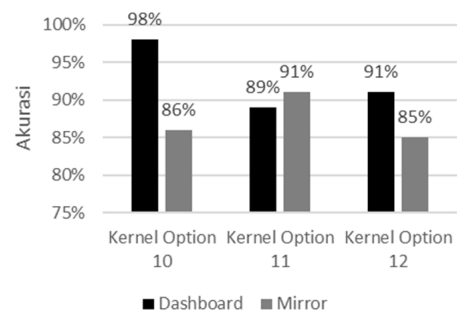
#### 3. Pengujian parameter rescaling citra terhadap akurasi sistem

Pada pengujian ini digunakan citra dengan posisi kamera berada dibagian *dashboard* dengan menggunakan parameter PCA *eigenface* dan SVM menggunakan strategi OAA dan OAO menggunakan *kernel polynomial* dan *kernel option* 10,11, dan 12. Tujuan dari *rescaling* ini adalah untuk mengetahui pengaruh setiap ukuran piksel terhadap tingkat akurasi. Untuk mencari dimensi terbaik pada sistem ini digunakan 3 dimensi citra yaitu 256x256 piksel, 512x512 piksel, dan 1024x1024 piksel.

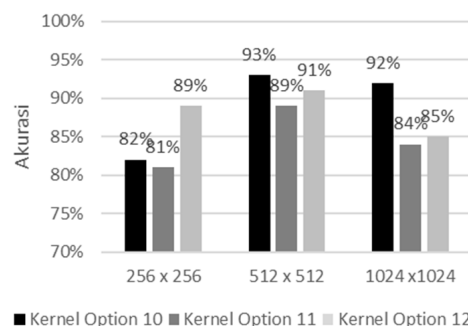
Berdasarkan Gambar 7 menunjukkan bahwa hasil dari pengujian *rescaling* pada citra memiliki tingkat akurasi tertinggi pada dimensi *rescaling* 512 x 512 piksel dengan nilai akurasi pada *kernel option* 10 sebesar 93% dan tingkat akurasi terendah pada dimensi *rescaling* 256x256 piksel dengan nilai akurasi pada *kernel option* 10 sebesar 82%.

#### 4. Pengujian pengaruh kompresi citra terhadap akurasi sistem

Pada pengujian ini digunakan citra berdimensi 512x512



Gambar 6. Akurasi *kernel option* terbaik



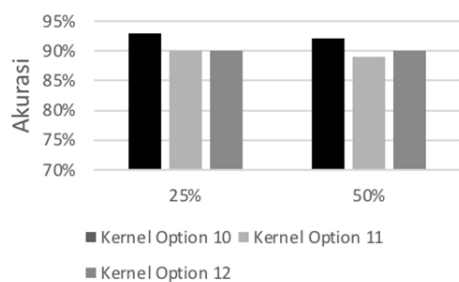
Gambar 7. Hasil pengujian parameter *rescaling* pada citra

piksel dengan *eigenface* pada PCA dan menggunakan *kernel polynomial* pada *kernel option* 10, 11, dan 12 pada SVM dengan posisi kamera pada *dashboard*. Pengujian ini bertujuan untuk mengetahui pengaruh performansi sistem saat dilakukan *lossless compression* terhadap citra. *Lossless compression* yang dimaksud disini adalah dengan mengurangi degradasi warna atau sering disebut dengan *color reduction* tanpa menghilangkan satupun informasi pada citra, sehingga ukuran citra hasil kompresi sama dengan sebelum kompresi.

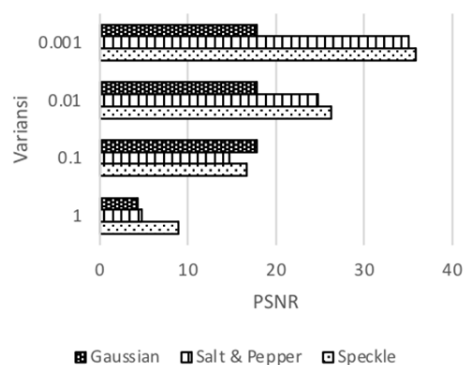
Pada Gambar 8 menunjukkan bahwa hasil akurasi dari pengujian kompresi pada citra yang telah dikompresi tidak mengalami kenaikan, sehingga dapat disimpulkan bahwa citra yang dikompresi tidak hanya mempengaruhi kualitas citra tetapi dapat mempengaruhi tingkat akurasi dari sistem. Semakin besar kompresi citra, semakin kecil akurasi sistem.

##### 5. Pengujian pengaruh noise terhadap akurasi

Berikut ini adalah data hasil pengujian dengan memberikan *noise* terhadap citra uji untuk mengetahui pengaruh *noise* terhadap akurasi sistem, dengan menggunakan dimensi 512x512 piksel pada *kernel polynomial option* 10 ( $b=10$ ) dengan posisi kamera pada *dashboard* sesuai dengan akurasi maksimal yang di dapatkan pada saat pengujian *rescaling* sebesar 93%. Noise yang diberikan pada pengujian ini ada 4 jenis yaitu *Gaussian*, *Salt & Pepper*, *Speckle*. Diujikan beberapa nilai variansi terhadap *noise* yang diberikan untuk mencari batas minimum variansi. Gambar 9 merupakan hasil pengujian pemberian variansi pada *noise*. Pada Gambar 9 menunjukkan bahwa pemberian *noise* terhadap citra uji akan mempengaruhi tingkat akurasi sistem. Tingkat akurasi sistem dan nilai PSNR akan menurun pada saat pemberian



Gambar 8. Hasil pengujian kompresi citra



Gambar 9. Pengaruh nilai variansi noise terhadap PSNR

*noise* pada titik variansi tertentu kecuali pada *noise poisson* yang tidak memiliki nilai variansi. Berdasarkan hasil pengujian, sistem paling handal terhadap *noise speckle* dibandingkan dengan *noise* lainnya berdasarkan nilai PSNR. Secara keseluruhan, hubungan nilai variansi pada *noise* berbanding terbalik dengan tingkat akurasi sistem.

##### a. Pengujian pemberian faktor pengali pada batas minimum variansi pada *noise Gaussian*.

Berdasarkan pengujian pemberian variansi pada *noise* didapatkan batas minimum nilai variansi pada *noise gaussian* berada pada nilai  $10^{-1}$  yang kemudian diberikan faktor pengali seperti pada Tabel 2.

Tabel 2. Pengujian pemberian faktor pengali pada batas minimum variansi pada *noise Gaussian*. Berdasarkan Tabel 2 menunjukkan bahwa batas maksimum pemberian faktor pengali nilai variansi pada *noise gaussian* adalah  $2 \times 10^{-1}$  dengan nilai PSNR yaitu 13,9259. Semakin besar variansi pada *noise gaussian* maka semakin kecil akurasi dari sistem, begitu pula sebaliknya.

##### b. Pengujian pemberian faktor pengali pada batas minimum variansi pada *noise Salt & Pepper*.

Berdasarkan pengujian pemberian variansi pada *noise* didapatkan batas minimum nilai variansi pada *noise salt & pepper* berada pada nilai  $10^{-1}$  yang kemudian diberikan faktor pengali seperti pada Tabel 3.

Berdasarkan Tabel 3 menunjukkan bahwa batas maksimum pemberian faktor pengali variansi pada *noise salt & pepper* adalah  $4 \times 10^{-1}$  dengan nilai PSNR yaitu 8,6955. Hubungan parameter nilai variansi pada *noise salt & pepper* berbanding terbalik dengan nilai akurasi sistem.

##### c. Pengujian pemberian faktor pengali pada batas minimum variansi pada *noise Speckle*.

Berdasarkan pengujian pemberian variansi pada *noise* didapatkan batas minimum nilai variansi pada *noise speckle* berada pada nilai  $10^{-1}$  yang kemudian diberikan faktor pengali seperti pada Tabel 4.

Tabel 2. Pengujian pemberian faktor pengali pada batas minimum variansi pada *noise Gaussian*

Jenis Noise	Faktor Pengali	Variansi	SNR	PSNR	Kecocokan Citra
Gaussian	2	$10^{-1}$	8,7662	13,9259	Benar
	3	$10^{-1}$	5,8970	11,0567	Salah
	4	$10^{-1}$	3,7706	8,9304	Salah

Tabel 3. Pengujian pemberian faktor pengali pada batas minimum variansi pada *noise salt & pepper*

Jenis Noise	Faktor Pengali	Variansi	SNR	PSNR	Kecocokan Citra
Salt & Pepper	2	$10^{-1}$	6,5468	11,7066	Benar
	3	$10^{-1}$	4,7860	9,9457	Benar
	4	$10^{-1}$	3,5358	8,6955	Benar
	5	$10^{-1}$	2,5697	7,7294	Salah
	6	$10^{-1}$	1,7833	6,9430	Salah

Tabel 4. Pengujian pemberian faktor pengali pada batas minimum variansi pada noise speckle

Jenis Noise	Faktor Pengali	Variansi	SNR	PSNR	Kecocokan Citra
Speckle	2	10-1	8,6455	13,8052	Benar
	3	10-1	6,9718	12,1315	Benar
	4	10-1	5,8676	11,0274	Benar
	5	10-1	5,1889	10,3486	Benar
	6	10-1	4,7361	9,8958	Benar
	7	10-1	4,4025	9,5623	Salah
	8	10-1	4,1210	9,2808	Salah

Berdasarkan Tabel 4. menunjukkan bahwa batas maksimum pemberian faktor pengali pada *noise speckle* adalah  $6 \times 10^{-1}$  dengan nilai PSNR yaitu 9,8958. Dimana jika diberikan faktor pengali yang lebih besar lagi pada *noise speckle* ini dapat mengurangi tingkat akurasi dan kecocokan citra.

#### d. Pengujian pemberian faktor pengali pada batas minimum variansi pada noise Poisson

Berdasarkan Tabel 5 menunjukkan bahwa *noise poisson* tidak memiliki nilai variansi karena *noise poisson* merupakan *noise* yang memiliki efek yang tetap sehingga tidak dibutuhkan adanya penambahan nilai variansi.

Didapatkan bahwa *noise poisson* cenderung stabil dikarenakan *noise* ini tidak memiliki nilai variansi. Dibuktikan pada Tabel 5 dengan beberapa pengujian yang dilakukan terhadap *noise* ini. Berdasarkan hasil simulasi yang dilakukan, deteksi kantuk ini sangat bergantung pada kondisi saat perekaman citra atau akuisisi. Banyak factor yang mempengaruhi proses akuisisi seperti kondisi cahaya, tipe mata, memakai kacamata atau tidak, memakai topi atau benda lainnya di atas kepala, dsb. Sama hal seperti percobaan yang dilakukan oleh Noni [27] masih perlunya optimasi disisi pencahayaan, sehingga diharapkan sistem dapat diterapkan untuk mengurangi tingkat kecelakaan yang diakibatkan kondisi pengemudi yang mengantuk.

#### IV. KESIMPULAN

Pada studi ini telah berhasil disimulasikan sebuah sistem untuk mendeteksi kantuk berdasarkan citra wajah pengemudi mobil. Sistem yang dibuat pada penelitian ini mampu mengenali ekspresi wajah pengemudi dan mengklasifikasikannya kedalam 2 kelas yaitu normal dan kantuk dengan menggunakan metode PCA dengan parameter *eigenface* sebagai ekstraksi ciri dan klasifikasi SVM menggunakan strategi *One Against One* (OAO) dan *One Against All* (OAA). Untuk mengetahui pemilihan jenis kernel terbaik diujikan jenis kernel RBF, *polynomial*, dan *linear* pada citra latih, hasilnya diperoleh tingkat akurasi rata-rata terbaik 84 % dengan menggunakan kernel *polynomial* dibandingkan dua jenis kernel lainnya. Pada pengujian, eksperimental dilakukan dengan parameter *kernel polynomial* dengan nilai *kernel option* 10, dengan

Tabel 5. Pengujian pada noise poisson

Jenis Noise	Percobaan Ke-	SNR	PSNR	Kecocokan Citra
Poisson	1	22.8591	28.4139	Benar
	2	22.8804	28.4310	Benar
	3	22.9167	28.4275	Benar
	4	22.9010	28.5160	Benar
	5	22.8321	28.5213	Benar
	6	22.8231	27.9828	Benar
	7	22.8001	28.0052	Benar
	8	22.8291	28.0261	Benar
	9	22.8267	28.0293	Benar
	10	22.8118	28.0297	Benar

ukuran citra asli 128x52 piksel, diperoleh citra hasil akuisisi pada *dashboard* memiliki nilai yang lebih tinggi dibandingkan citra hasil akuisisi pada *mirror* yaitu 98% berbanding 86.67%. Untuk mengetahui batas ukuran citra terbaik, diujikan *rescalling* citra dari 128x 52 piksel menjadi 256x256, 512x512, dan 1024x1024, diperoleh ukuran maksimal adalah pada 512x512 dengan maksimal kompresi 25%. Untuk mengetahui kehandalan sistem, diujikan dengan *noise speckle*, *salt and pepper*, *poisson* dan *gaussian*. Berdasarkan parameter nilai variansi yang diberikan pada  $10^{-1}$ , sistem memiliki nilai PSNR tertinggi saat diberi noise gaussian yaitu 13,92.

#### V. REFERENSI

- [1] T. Cvahte Ojsteršek and D. Topolšek, "Influence of drivers' visual and cognitive attention on their perception of changes in the traffic environment," *Eur. Transp. Res. Rev.*, vol. 11, no. 1, pp. 1–9, 2019.
- [2] M. H. Siddiqi, R. Ali, A. M. Khan, Y. T. Park, and S. Lee, "Human Facial Expression Recognition Using Stepwise Linear Discriminant Analysis and Hidden Conditional Random Fields," *IEEE Trans. Image Process.*, vol. 24, no. 4, pp. 1386–1398, 2015.
- [3] H. Qayyum, M. Majid, S. M. Anwar, and B. Khan, "Facial Expression Recognition Using Stationary Wavelet Transform Features," *Math. Probl. Eng.*, vol. 2017, no. 1, 2017.
- [4] R. Azmi and S. Yegane, "Facial expression recognition in the presence of occlusion using local Gabor binary patterns," in *ICEE 2012 - 20th Iran. Conf. Electr. Eng.*, pp. 742–747, 2012.
- [5] J. Kumari, R. Rajesh, and K. M. Pooja, "Facial Expression Recognition: A Survey," *Procedia Comput. Sci.*, vol. 58, pp. 486–491, 2015.
- [6] W.-L. Chao, J.-J. Ding, and J.-Z. Liu, "Facial expression recognition based on improved local binary pattern and class-regularized locality preserving projection," *Signal Processing*, vol. 117, pp. 1–10, Dec. 2015.
- [7] P. Carcagni, M. Del Coco, M. Leo, and C. Distanto, "Facial expression recognition and histograms of oriented gradients: a comprehensive study," *Springerplus*, vol. 4, no. 1, 2015.
- [8] B. Hasani and M. H. Mahoor, "Spatio-Temporal Facial Expression Recognition Using Convolutional Neural Networks and Conditional Random Fields," in *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*, May 2017.

- [9] D. Ghimire, S. Jeong, J. Lee, and S. H. Park, "Facial expression recognition based on local region specific features and support vector machines," *Multimed. Tools Appl.*, vol. 76, no. 6, pp. 7803–7821, 2017.
- [10] A. T. Lopes, E. de Aguiar, A. F. De Souza, and T. Oliveira-Santos, "Facial expression recognition with Convolutional Neural Networks: Coping with few data and the training sample order," *Pattern Recognition*, vol. 61, pp. 610–628, Jan. 2017.
- [11] N. Zeng, H. Zhang, B. Song, W. Liu, Y. Li, and A. M. Dobaie, "Facial expression recognition via learning deep sparse autoencoders," *Neurocomputing*, vol. 273, pp. 643–649, 2018.
- [12] F. Farooq, J. Ahmed, and L. Zheng, "Facial Expression Recognition Using Hybrid Features and Self-Organizing Maps," in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, 2017.
- [13] A. De, A. Saha, and M. C. Pal, "A human facial expression recognition model based on eigen face approach," *Procedia Comput. Sci.*, vol. 45, no. C, pp. 282–289, 2015.
- [14] T. Zhang, W. Zheng, Z. Cui, Y. Zong, J. Yan, and K. Yan, "A Deep Neural Network-Driven Feature Learning Method for Multi-view Facial Expression Recognition," *IEEE Trans. Multimed.*, vol. 18, no. 12, pp. 2528–2536, 2016, doi: 10.1109/TMM.2016.2598092.
- [15] S. Agrawal and P. Khatri, "Facial expression detection techniques: Based on Viola and Jones algorithm and principal component analysis," in *Int. Conf. Adv. Comput. Commun. Technol. ACCT*, vol. 2015-April, pp. 108–112, 2015.
- [16] P. F. Perdananti, B. Hidayat, and S. Aulia, "HOG and ICA Based Face Recognition System on A Surveillance Video," *Far East J. Electron. Commun.*, vol. 17, no. 5, pp. 995–1009, Oct. 2017.
- [17] K. Zhang, Y. Huang, Y. Du, and L. Wang, "Facial Expression Recognition Based on Deep Evolutional Spatial-Temporal Networks," *IEEE Trans. Image Process.*, vol. 26, no. 9, pp. 4193–4203, 2017.
- [18] S. Al-Aidid and D. Pamungkas, "Sistem Pengenalan Wajah dengan Algoritma Haar Cascade dan Local Binary Pattern Histogram," *J. Rekayasa Elektr.*, vol. 14, no. 1, pp. 62–67, 2018.
- [19] A. P. Gosavi and S. R. Khot, "Emotion recognition using Principal Component Analysis with Singular Value Decomposition," in *2014 International Conference on Electronics and Communication Systems (ICECS)*, Feb. 2014, pp. 1–5.
- [20] D. R. Anggraini, "Face recognition using principal component analysis and self organizing maps," in *2014 Third ICT International Student Project Conference (ICT-ISPC)*, Mar. 2014, pp. 91–94.
- [21] V. P. Kshirsagar, M. R. Baviskar, and M. E. Gaikwad, "Face recognition using Eigenfaces," *ICCRD2011 - 2011 3rd Int. Conf. Comput. Res. Dev.*, vol. 2, pp. 302–306, 2011.
- [22] I. R. Wijaya, U. N. Wisesty, and S. Al Faraby, "Analisis dan Implementasi Metode Gabor Filter dan Support Vector Machine pada Klasifikasi Sidik Jari," *Ind. J. Comput.*, vol. 2, no. 2, pp. 37–46, 2017, doi: 10.21108/indoic.2017.22.176.
- [23] S. Aulia, S. Hadiyoso, and D. N. Ramadan, "Analisis Perbandingan KNN dengan SVM untuk Klasifikasi Penyakit Diabetes Retinopati berdasarkan Citra Eksudat dan Mikroaneurisma," *J. ELKOMIKA -Teknik Elektro Itenas*, vol. 3, no. 1, pp. 75–90, 2015.
- [24] R. Fikri, F. Arnia, and R. Muharar, "Pengenalan Karakter Tulisan Tangan Jawi Menggunakan Metode New Relative Context dan SVM," *JNTETI*, vol. 5, no. 3, pp. 1–6, 2016.
- [25] B. Al Kindhi, T. A. Sardjono, and M. H. Purnomo, "Optimasi Support Vector Machine (SVM) untuk memprediksi adanya mutasi pada DNA Hepatitis C Virus ( HCV )," *JNTETI*, vol. 7, no. Agustus, 2018.
- [26] A. S. Indrawanti and E. P. Mandyartha, "Deteksi Limfoblas pada Citra Sel Darah Menggunakan Fitur Geometri dan Local Binary Pattern," *J. Nas. Tek. Elektro dan Teknol. Inf.*, vol. 7, no. 4, pp. 404–410, 2018.
- [27] N. Charimmah, E. Lanovia, K. Usman, and L. Novamizanti, "Deteksi Kantuk Melalui Citra Wajah Menggunakan Metode Gray Level Co- occurrence Matrix (GLCM) dan Klasifikasi Support Vector Machine (SVM)," in *SENER (Seminar Nasional Teknik Elektro)*, 2019, no. November 2019, pp. 174–185.