

Perbandingan Fase Ekspresi Menggunakan *Local Binary Pattern Histogram* untuk Pengenalan Ekspresi Mikro

Ulla Delfana Rosiani¹, Priska Choirina², Yessy Nindi Pratiwi³, Septiar Enggar Sukmana⁴

^{1,3,4} Program Studi Teknik Informatika, Jurusan Teknologi Informasi, Politeknik Negeri Malang, Kota Malang, 65141, INDONESIA, (tel.: 0341-404424; fax: 0341-404420, email: ¹rosiani@polinema.ac.id, ³yessynindi@gmail.com, ⁴enggar@polinema.ac.id)
² Program Studi Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Islam Raden Rahmat, Malang 65163, INDONESIA (tel.: 0341- 399099; fax: -; email: ²priska_choirina@uniramalang.ac.id)

[Diterima: 20 Juli 2023, Direvisi: 18 September 2023]

Corresponding Author: Ulla Delfana Rosiani

INTISARI — Ekspresi mikro merupakan representasi emosional yang terjadi secara spontan dan tidak dapat dikendalikan secara sadar serta bersifat sementara (durasi pendek) dengan gerakan halus (intensitas gerakan kecil), sehingga sulit dideteksi dengan mata manusia biasa. Gerakan otot ekspresi mikro dihasilkan hanya pada beberapa area kecil wajah, sehingga pengamatan pada area tertentu menghasilkan waktu komputasi yang lebih cepat dan memberikan informasi penting dibandingkan dengan pengamatan pada seluruh wajah. Penelitian ini mengusulkan pengurangan area dan fase pengamatan untuk pengenalan ekspresi mikro. Area yang diamati yaitu alis kiri, alis kanan, mata kanan, mata kiri, dan mulut pada dataset CASME II. Fase pengamatan ekspresi mikro dilakukan dengan menganalisis perbandingan pada fase *onset* hingga *offset* (“fullOAO”) dan hanya pada fase *onset*, *apex*, dan *offset* (“OAO”). Ekstraksi fitur dilakukan menggunakan metode sederhana dari *local binary patterns histogram* (LBPH), yang dapat merepresentasikan fitur lokal pada area wajah. Hasil terbaik dari metode yang diusulkan adalah fase “fullOAO” dengan akurasi 96,8% (menggunakan *support vector machine-radial basis function*, SVM-RBF) serta rata-rata waktu komputasi sebesar 0,192 ms per *frame* dan 10,473 ms per video. Pada jenis fase “OAO”, dihasilkan akurasi sebesar 87,7% dengan waktu komputasi 0,159 ms per *frame* dan 0,576 ms per video. Perbedaan akurasi dan waktu komputasi kedua jenis fase tersebut terjadi karena jumlah *frame* pada jenis “fullOAO” lebih besar daripada “OAO”, sehingga menghasilkan waktu pemrosesan dan data ekstraksi fitur dalam jumlah yang berbeda. Namun, penurunan akurasi sebesar 9% tidak berpengaruh terlalu besar karena tingkat akurasinya masih tergolong baik, yaitu di atas 80%. Selain itu, pengukuran yang tepat untuk waktu komputasi adalah waktu yang dibutuhkan untuk memproses setiap *frame* pada video masukan. Maka, dapat disimpulkan bahwa dengan metode yang diusulkan, dapat dihasilkan waktu komputasi cepat dan pengenalan yang tergolong akurat.

KATA KUNCI — Pengenalan Ekspresi Mikro, Fase Ekspresi, LBPH, Ekspresi Mikro.

I. PENDAHULUAN

Ekspresi wajah (*facial expression*) manusia merupakan faktor penting dalam komunikasi sosial. Komunikasi biasanya melibatkan komunikasi verbal dan nonverbal [1]. Komunikasi nonverbal ditunjukkan dengan gerakan pada wajah, kemudian disebut dengan ekspresi wajah. Ekspresi wajah adalah salah satu teknik yang cocok untuk mengetahui keadaan mental, sikap, dan niatan pada manusia [2]. Secara umum, ekspresi wajah meliputi beberapa jenis ekspresi, yaitu sedih, jijik, takut, marah, terkejut, dan senang [3]. Dalam ilmu psikologi, ekspresi wajah terbagi menjadi dua jenis, yaitu ekspresi makro (*macro-expression*) dan ekspresi mikro (*micro-expression*) [4]. Ekspresi wajah yang dapat diamati dengan mudah menggunakan mata telanjang dapat disebut sebagai ekspresi makro. Ekspresi makro memiliki gerakan pada otot wajah yang sangat jelas, sehingga dapat dengan mudah dianalisis [5].

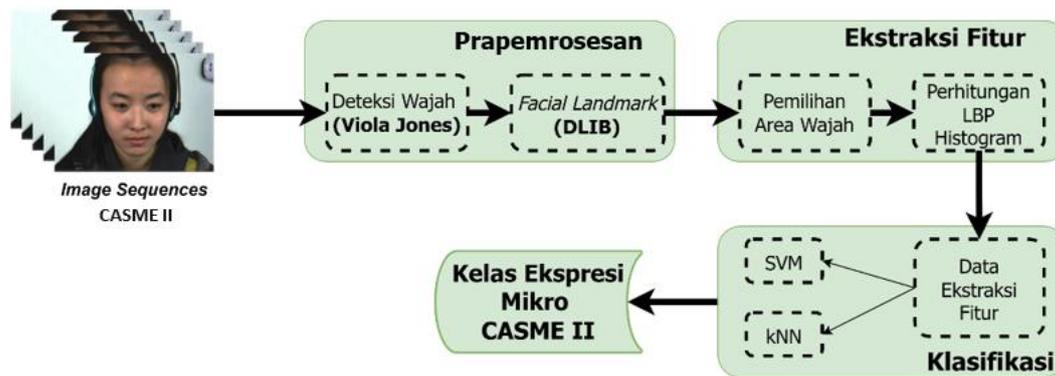
Berbeda dengan ekspresi makro, ekspresi mikro adalah gerakan otot pada wajah yang memiliki gerakan halus dan terjadi secara spontan. Ekspresi mikro biasanya berlangsung kurang dari 200 ms [6], yang tidak dapat disembunyikan bahkan oleh aktor profesional [7]. Oleh karena itu, ekspresi mikro dianggap sebagai refleksi secara nyata dari emosi pada manusia [8]. Ekspresi mikro memiliki tiga karakteristik utama, yaitu durasinya pendek, gerakannya halus, dan sulit disembunyikan [9]. Analisis ekspresi mikro sangat berguna dalam situasi berisiko tinggi, seperti tes pemulihan depresi, negosiasi dengan teroris, atau investigasi kriminal [7]. Oleh karena itu, analisis ekspresi mikro memainkan peranan penting

dalam menafsirkan emosi asli dari seseorang yang dapat diimplementasikan dalam kehidupan sehari-hari [10].

Ekspresi mikro merupakan gerakan wajah dinamis sesuai dengan urutan fase ekspresi berikut: *neutral-onset-apex-offset-neutral*. Kondisi awal merupakan fase netral, kemudian pada kondisi terdapat stimulan, gerakan wajah menuju ke fase *onset* [11]. Fase *onset* menunjukkan kondisi awal pada ekspresi dengan otot pada wajah mulai berkontraksi. Lalu, fase *apex* merupakan kondisi otot wajah berada di gerakan puncak, sehingga pada fase ini analisis dapat dengan mudah dilakukan [3]. Selanjutnya, fase *offset* adalah kondisi otot wajah mulai bergerak menuju posisi netral.

Penelitian terkait pengenalan ekspresi mikro tergolong baru dan telah berkembang dalam beberapa tahun terakhir. Penelitian ekspresi mikro yang telah dilakukan menerapkan berbagai macam metode yang bertujuan untuk mengenali ekspresi mikro secara otomatis. Terdapat beberapa metode yang sering digunakan dalam pengenalan ekspresi pada wajah, seperti *principal component analysis* (PCA) [12], *local binary pattern* (LBP) [13], dan *linear discriminant analysis* (LDA) [14]. Metode-metode tersebut telah berhasil diterapkan dengan hasil akurasi yang tergolong baik. Hasil tersebut tergantung pada berbagai hal. Selain dari pemilihan metode ekstraksi fitur, jenis fitur dan pemilihan metode klasifikasi menjadi poin penting dalam mendapatkan akurasi tinggi.

Metode PCA [15] telah diterapkan pada pengenalan ekspresi secara umum untuk menunjukkan ekspresi wajah,



Gambar 1. Alur kerja penelitian yang diusulkan.

tetapi metode tersebut tidak cocok untuk variasi iluminasi (*illumination variation*) [16]. Oleh karena itu, dilakukan pengembangan, yaitu metode LBP. Metode ini membutuhkan waktu komputasi yang lebih singkat dan toleransinya terhadap iluminasi lebih baik [13]. Salah satu perkembangan metode LBP adalah *local binary pattern histogram* (LBPH). Metode LBPH merupakan kombinasi algoritma metode LBP dan histogram, yaitu salah satu metode yang cocok untuk mendeskripsikan tekstur. Selain itu, metode LBPH dapat merepresentasikan fitur lokal pada area wajah. Hal ini mendukung pengenalan ekspresi wajah yang tidak terjadi pada area wajah besar. Metode LBPH juga merupakan metode yang relatif sederhana karena hanya melibatkan beberapa operasi perhitungan sederhana, seperti biner dan histogram, sehingga membutuhkan waktu komputasi yang lebih singkat.

Penelitian terdahulu mengadopsi teknik modifikasi metode LBP, yaitu *LBP-three orthogonal plane* (LBP-TOP) sebagai metode ekstraksi fitur [17], [18]. Metode ini memungkinkan digambarkannya tekstur citra digital dengan cara membagi gambar menjadi beberapa bagian kecil dari fitur yang akan diekstraksi. Fitur-fitur tersebut terdiri atas pola biner yang menggambarkan lingkungan piksel di area wajah yang dianalisis. Fitur-fitur tersebut kemudian diubah menjadi histogram fitur tunggal yang membentuk representasi dari gambar. Namun, penelitian tersebut memiliki kelemahan pada proses komputasi yang tergolong lama. Penyebabnya adalah pengamatan dilakukan di seluruh wajah. Hal ini mengakibatkan data fitur yang dihasilkan cukup besar dengan proses yang memakan waktu lama.

Ekstraksi fitur merupakan tahapan yang penting dalam pengenalan ekspresi mikro. Sistem dalam pengenalan ekspresi mikro akan menghasilkan akurasi yang baik ketika fitur yang diekstraksi dapat mengurangi variasi dalam kelas dan memaksimalkan variasi antarkelas. Representasi fitur yang bagus dapat menjamin proses pengenalan berjalan secara efisien dengan hasil yang akurat [19]. Dalam pengenalan ekspresi mikro, gerakan otot hanya muncul pada beberapa area kecil di wajah. Gerakan yang dominan terdapat pada dua bagian wajah, yaitu pada area alis dan mulut [20]. Sementara itu, menurut penelitian lain, pengamatan ekspresi mikro wajah pada area tertentu memerlukan waktu komputasi yang lebih singkat dan hasilnya lebih akurat dibandingkan dengan pengamatan pada seluruh wajah [21]. Pendeteksian komponen area wajah dapat dilakukan dengan berbagai cara, salah satunya mengadopsi metode *landmark point* dalam pembentukan area komponen wajah [22]. Dengan pengamatan pada area-area tertentu, durasi pengenalan ekspresi mikro menjadi lebih singkat dibandingkan dengan pengamatan yang dilakukan pada seluruh wajah [23]. Di sisi lain, akurasi bergantung pada

beberapa hal, di antaranya pemilihan jenis fitur yang digunakan [1].

Berdasarkan permasalahan yang telah dijelaskan, penelitian ini mengusulkan analisis perbandingan fase ekspresi mikro, yaitu “fullOAO” dan “OAO”, pada beberapa komponen wajah terpilih. Perbedaan mendasar dari dua jenis fase ekspresi tersebut adalah jumlah *frame* yang digunakan untuk proses analisis. Fase ekspresi “fullOAO” merupakan *frame* terpilih dari fase *onset* hingga *offset*, sedangkan fase “OAO” merupakan penggunaan *frame* hanya pada fase *onset*, *apex*, dan *offset*. Tujuan perbandingan fase ekspresi tersebut adalah untuk mengetahui hasil kinerja akurasi pengenalan terhadap penggunaan *frame* berdasarkan fase ekspresi pada *dataset*. Penggunaan *frame* terpilih dapat memberikan waktu pemrosesan yang lebih rendah dibandingkan dengan penggunaan keseluruhan *frame* pada *video clip* dari *dataset*. Area wajah terpilih untuk pengenalan ekspresi mikro adalah alis kiri, alis kanan, mata kanan, mata kiri, dan mulut. Pemilihan area tersebut dilakukan berdasarkan penelitian sebelumnya [21], [20], sebagai area wajah untuk ekstraksi fitur. Penerapan area wajah terbatas diharapkan dapat mempercepat proses pengenalan dan menghasilkan data hasil ekstraksi fitur yang representatif untuk proses klasifikasi.

II. METODOLOGI

Pada sistem pengenalan ekspresi mikro *real-time*, dibutuhkan respons yang cepat dengan akurasi yang tepat untuk mengidentifikasi suatu gerakan ekspresi. Oleh karena itu, waktu pemrosesan yang cepat dan akurasi yang tepat tidak dapat dipisahkan dalam membangun sistem pengenalan ekspresi mikro secara *real-time*. Pengenalan ekspresi mikro membutuhkan keseimbangan antara akurasi dan waktu pemrosesan. Secara umum, makin cepat suatu sistem dapat mengenali dan menginterpretasi ekspresi mikro pada wajah, makin baik kualitas respons pada sistem tersebut. Namun, dalam meningkatkan akurasi, sering kali digunakan metode yang lebih kompleks dan metode analisis yang canggih, sehingga waktu pemrosesan menjadi lebih lama.

Pada penelitian ini, diusulkan pengenalan ekspresi mikro pada wajah manusia menggunakan metode LBPH untuk ekstraksi fitur. Gambar 1 menunjukkan kerangka kerja yang diusulkan secara rinci. Penelitian ini memiliki tiga tahap utama, yaitu prapemrosesan, ekstraksi fitur, dan klasifikasi, seperti ditunjukkan pada Gambar 1. Penelitian ini bertujuan mengenali ekspresi mikro dalam urutan citra pada video pada basis data CASME II dari Chinese Academy of Science Micro-expressions. Kontribusi penelitian yang diusulkan yaitu pengurangan daerah pengamatan dan analisis perbandingan fase ekspresi, yaitu pada fase *onset* hingga *offset* (“fullOAO”)

dan hanya pada fase *onset*, *apex*, dan *offset* (OAO). Area komponen wajah tersebut yaitu alis kiri, alis kanan, mata kanan, mata kiri, dan mulut [21], [20]. Dengan menggunakan metode LBPH dalam proses ekstraksi fitur, fitur lokal pada gerakan area wajah terpilih dapat direpresentasikan. Metode LBPH menghasilkan sebuah fitur yang berfungsi untuk mengklasifikasi citra yang dikombinasikan dari LBP dan histogram. Metode LBPH merupakan salah satu teknik terbaru dari LBP untuk mengubah kinerja dalam hal pengenalan, termasuk pengenalan citra.

A. DATASET CASME II BERDASARKAN FASE EKSPRESI

Dataset CASME II merupakan *dataset* yang banyak digunakan di bidang analisis ekspresi wajah dan pengenalan ekspresi mikro [24]. *Dataset* ini berisi *video clip* ekspresi mikro spontan, yang merupakan ekspresi wajah yang sangat singkat dan terjadi tanpa disengaja serta dapat mengungkapkan emosi yang sebenarnya. *Dataset* CASME II terdiri atas 247 urutan video (*video sequences*) yang diambil dari 26 subjek, termasuk pria dan wanita. Setiap *video clip* biasanya berdurasi 5 s hingga 10 s dan direkam pada kecepatan 200 fps dengan resolusi spasial sekitar 640×480 piksel pada area wajah. *Dataset* ini mencakup tujuh emosi dasar (kelas ekspresi), yaitu *happiness*, *sadness*, *surprise*, *fear*, *disgust*, *anger*, dan *contempt*. Rincian data setiap *video clip* pada CASME II, seperti nama *file*, jumlah *frame*, label ekspresi, serta *frame* fase ekspresi (*onset*, *apex*, dan *offset*), tersedia pada *file* .xls dalam *dataset* tersebut.

Penelitian ini menggunakan empat kelas ekspresi terpilih, yaitu *disgust*, *sadness*, *surprise*, dan *happiness*. Pemilihan empat label ekspresi tersebut didasari oleh perbedaan jumlah video dari masing-masing kelas ekspresi pada CASME II. Perbedaan jumlah video disebabkan oleh sulitnya memperoleh ekspresi tersebut, sehingga menyebabkan distribusi *video clip* tidak merata [25]. Maka, *dataset* CASME II yang tersedia hanya menyediakan lima kelas ekspresi mikro, yaitu *disgust*, *fear*, *happiness*, *sadness*, dan *surprise*. Kelima kelas ekspresi memiliki jumlah video yang tidak seimbang, sehingga kinerja model juga menjadi tidak seimbang dan akan menghasilkan evaluasi data yang condong (bias) ke data mayoritas. Oleh karena itu, pada penelitian ini digunakan empat kelas ekspresi yang dipilih secara manual dengan mengacu pada *file* rincian yang tersedia pada CASME II.

Pemilihan manual dilakukan dengan mempertimbangkan jumlah *frame* yang hampir sama antara *onset* hingga *offset* untuk masing-masing video, seperti ditunjukkan pada Tabel I. *Frame* citra yang digunakan hanya pada *frame onset* hingga *offset* karena tujuan penelitian ini adalah untuk melakukan analisis perbandingan fase ekspresi *onset* hingga *offset* (“fullOAO”) dan hanya pada fase *onset*, *apex*, dan *offset* (OAO), dengan menggunakan area komponen wajah terpilih, yaitu area alis kiri, alis kanan, mata kanan, mata kiri, dan mulut. Sebagai contoh, data *Happiness* (17_EP01_15) memiliki jumlah *frame* pada video sebesar 223 citra. Jumlah *frame* antara *onset* hingga *offset* pada fase “fullOAO” dalam video tersebut adalah 42 *frame* citra, sedangkan untuk fase “OAO” berjumlah tiga *frame* citra karena hanya terdiri atas *frame onset*, *apex*, dan *offset*.

B. PRAPEMROSESAN

Tahapan prapemrosesan terdiri atas dua proses utama, yaitu deteksi wajah dan pencarian lokasi dari komponen wajah, seperti alis, mata, dan mulut. Pada tahapan pencarian lokasi wajah, diterapkan metode Viola Jones, sedangkan untuk pencarian komponen wajah, digunakan metode *landmark* wajah dari Dlib [26].

TABEL I
INFORMASI JUMLAH *FRAME* PADA SETIAP KELAS EKSPRESI

No	Kelas Ekspresi	Jumlah Video	Jumlah <i>Frame</i> Keseluruhan	Jumlah <i>Frame</i>	
				fullOAO	OAO
1	<i>Disgust</i>	7	2.071	269	21
2	<i>Happiness</i>	7	1.529	319	21
3	<i>Sadness</i>	7	2.788	281	21
4	<i>Surprise</i>	6	1.517	286	18
Jumlah <i>frame</i>		27	7.905	1.155	81

1) PENDETEKSIAN WAJAH MENGGUNAKAN VIOLA JONES

Pada tahapan ini, data masukan berbentuk sebuah video. Terhadap video tersebut kemudian dilakukan ekstraksi setiap *frame* dan menghasilkan urutan citra (*image sequences*). Setiap citra tersebut dikenai proses *grayscale* di seluruh citra, yang berfungsi untuk mengubah citra berwarna (RGB) menjadi citra keabuan. Selanjutnya, metode Viola Jones diterapkan untuk proses pencarian wajah pada setiap urutan citra yang telah diekstrak. Umumnya, metode ini disebut dengan *Haar cascade classifier* [27]. Metode ini memiliki waktu komputasi tergolong cepat karena dalam pencarian sebuah objek metode ini hanya melakukan pencarian pada pusat daerah yang memiliki peluang besar dari objek itu berada. Selain itu, hasil pencarian objek memiliki tingkat akurasi yang tinggi dan efisien dalam deteksi wajah [22]. Berikut merupakan penjelasan proses dari pendekatan untuk mendeteksi objek wajah.

1. Pelatihan data dilakukan untuk membentuk model dengan dua jenis data, yaitu data positif dan data negatif. Dalam hal ini, data positif merupakan data citra wajah, sedangkan data negatif adalah data noncitra wajah. Proses ini bertujuan untuk mempelajari perbedaan antara objek yang akan dideteksi.
2. Menentukan *Haar-like features* dengan cara mengurangi nilai rata-rata piksel pada daerah gelap dan terang. Fitur tersebut dianggap ada jika perbedaan nilai berada pada nilai ambang yang ditentukan.
3. *Integral image* digunakan untuk menentukan ada atau tidaknya *Haar-like features* pada citra yang diproses.
4. Citra integral pada (x, y) berisi jumlah piksel di sebelah kiri dan di atas, seperti pada (1).

$$II(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y') \quad (1)$$

dengan $II(x, y)$ adalah citra integral dan $i(x', y')$ adalah citra asal.

5. Pada (2) dan (3), $s(x, y)$ merupakan akumulasi baris kumulatif dari $s(x, -1) = 0$ dan $ii(-1, y) = 0$. Citra integral tersebut diformulasikan dalam satu lintasan melewati citra asli.

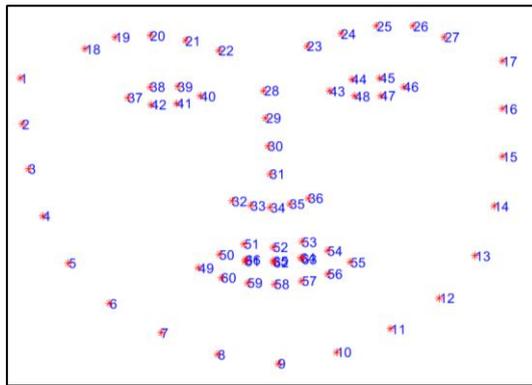
$$s(x, y) = s(x, y - 1) + i(x, y) \quad (2)$$

$$II(x, y) = II(x - 1, y) + s(x, y) \quad (3)$$

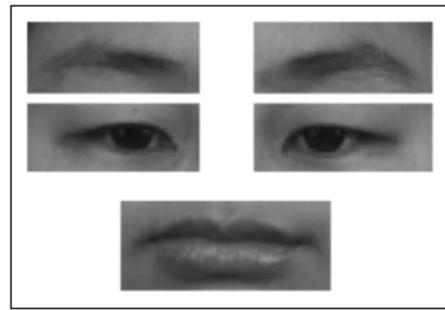
6. *Cascade classifier* digunakan untuk menghubungkan banyak fitur dengan mengombinasikan proses klasifikasi secara bertingkat agar lebih efisien.
7. Apabila wajah telah ditemukan, selanjutnya lokasi wajah ditandai dengan *region of interest* (RoI).

2) FACIAL LANDMARK DENGAN DLIB

Facial landmark merupakan deteksi komponen wajah berdasarkan titik-titik wajah yang menonjol, yang berfungsi sebagai pembentuk struktur wajah sebagai penanda komponen wajah tersebut. *Facial landmark* dapat mendeteksi komponen wajah dengan tepat, seperti area alis, mata, hidung, dan mulut.



Gambar 2. Struktur 68 titik dari Dlib untuk pembagian area komponen wajah.



Gambar 3. Hasil pemotongan komponen wajah untuk ekstraksi fitur.

Penelitian ini menerapkan sebuah *tool* Dlib dengan menggunakan *regression tree* untuk melokalisasi 68 titik yang telah terlatih [26]. Setelah RoI terbentuk dari hasil proses deteksi wajah sebelumnya, selanjutnya dilakukan penentuan titik-titik fitur untuk pencarian komponen wajah. Penerapan 68 titik oleh Dlib dilakukan pada area RoI yang telah ditentukan sebagai wajah, seperti ditunjukkan dalam Gambar 2 [26]. Tujuannya adalah untuk meminimalkan kesalahan dalam deteksi komponen wajah. *Regression tree* pada Dlib bekerja menggunakan *regressor*, yaitu membuat prediksi berdasarkan fitur. Fitur-fitur tersebut antara lain nilai-nilai intensitas piksel yang dihitung dari I dan dilakukan pengindeksan secara dinamis terhadap estimasi perulangan yang sedang terjadi. Penelitian ini menggunakan titik-titik *landmark* hanya pada area alis, mata, dan mulut, seperti tampak pada Gambar 2. Penggunaan titik dilakukan untuk menandai fitur komponen wajah sebagai berikut: alis kanan (titik ke-18 sampai ke-22), alis kiri (titik ke-23 sampai ke-27), mata kanan (titik ke-37 sampai ke-41), mata kiri (titik ke-43 sampai ke-47), dan mulut (titik ke-49, ke-55, ke-52 dan ke-58). Setelah titik-titik tersebut ditentukan, terhadap komponen yang telah dideteksi dilakukan pemotongan (*cropping*) dan dilakukan perubahan ukuran (*resize*) menjadi 100×50 piksel pada masing-masing komponen wajah, seperti ditunjukkan pada Gambar 3.

C. EKSTRAKSI FITUR DENGAN LBPH

LBPH merupakan metode yang digunakan dalam *computer vision* yang diterapkan untuk pengenalan sebuah objek. Metode ini melibatkan pemecahan gambar menjadi wilayah yang lebih kecil dan menghitung LBP untuk setiap wilayah. LBP adalah kode biner yang mewakili hubungan intensitas piksel dengan piksel tetangganya. Dengan menggabungkan histogram dan LBP dari semua wilayah dalam semua citra, sebuah vektor fitur dihasilkan. Vektor fitur ini digunakan untuk klasifikasi sebuah citra menggunakan algoritma pembelajaran mesin seperti *k nearest neighbor* (KNN) atau *support vector machine* (SVM). Metode LBPH dikenal dengan kesederhanaannya (*simplicity*) [28], [29], efisien, dan tangguh dalam perubahan iluminasi. Oleh karena itu, metode ini populer untuk pengenalan ekspresi mikro.

Gambar 4 merupakan ilustrasi proses dari metode LBPH yang diterapkan pada penelitian ini. Sebagai contoh, pada langkah pertama, citra masukan yang digunakan adalah citra alis kanan. Ukuran citra tersebut adalah $I_{w,h} = 58 \times 55$. Notasi I merupakan matriks citra dengan lebar (w) 58 piksel dan tinggi (h) 55 piksel. Metode LBP menggunakan citra masukan *grayscale* untuk mengurangi kompleksitas perhitungan dan fokus pada pola tekstur dalam citra. Kemudian, langkah kedua adalah membagi citra menjadi beberapa blok piksel *non-*

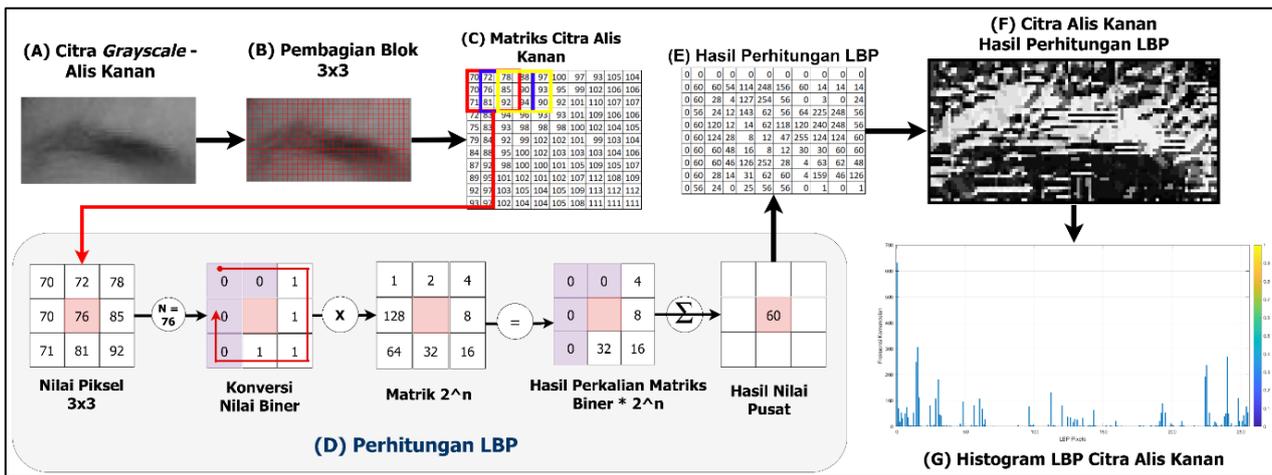
overlapping. Penelitian ini menggunakan pembagian blok 3×3 yang menghasilkan lima area fitur wajah yang telah ditentukan sebelumnya. Pembagian ukuran blok 3×3 bertujuan untuk merepresentasikan fitur yang lebih detail dalam menangkap pola-pola lokal dalam citra. Dengan mempertimbangkan piksel-piksel tetangga dalam blok, metode LBP dapat memberikan representasi fitur yang lebih baik. Gerakan ekspresi mikro memiliki karakteristik pergerakan halus (*subtle motion*). Makin kecil bagian blok yang diproses, makin detail fitur yang dihasilkan. Metode LBP dengan blok 3×3 menggunakan nilai tengah sebagai ambang batas (*threshold*) untuk piksel tetangganya.

$$LBP_{P,R} = \sum_{n=0}^{P-1} S(x_n - x_c) 2^n, \quad S(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (4)$$

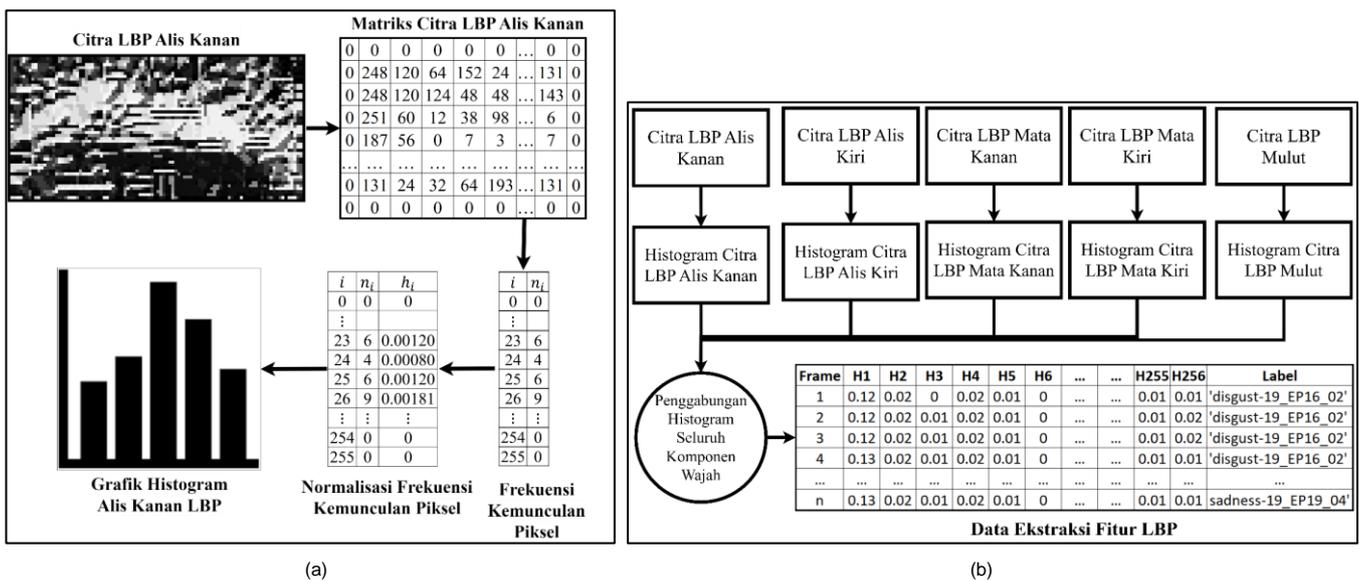
Pada (4), notasi (P,R) merupakan titik *sampling* (jumlah ketetanggaan) atau *n_point P* pada radius R . Notasi x_n merupakan nilai dari piksel tetangga, sedangkan x_c adalah nilai dari piksel pusat. Notasi 2^n merupakan nilai bobot dari jumlah pangkat n . $S(x)$ merupakan penentuan nilai ambang batas pada $S(x_n - x_c)$. Jika nilai x lebih dari atau sama dengan nol, $S(x)$ bernilai 1. Jika terjadi sebaliknya, $S(x)$ bernilai 0.

Setelah proses pembagian blok pada citra, perhitungan LBP pada penelitian ini menggunakan nilai $P = 8$ dengan radius $R = 1$ ($LBP_{8,1}$). Perhitungan LBP dimulai dari matriks $(0,0)$ atau area kiri atas. Pembagian blok tidak terjadi merata di seluruh citra, sehingga perlu dilakukan penambahan piksel dengan nilai nol (*zero padding*) di sekitar tepi gambar agar tidak terjadi kehilangan pergerakan pada citra. Matriks 3×3 dikonversikan ke dalam matriks biner dengan cara membandingkan nilai pusat (x_c) dengan nilai ketetanggaannya sesuai arah jarum jam. Jika nilai intensitas piksel tetangga lebih besar atau sama dengan nilai piksel pusat, nilai biner adalah 1. Jika yang terjadi adalah kondisi sebaliknya, nilai biner adalah 0. Setelah matriks biner terbentuk, proses selanjutnya adalah mengalikannya dengan bobot 2^n , yaitu $2^8 = 2^0 + 2^1 + 2^3 + 2^4 + 2^5 + 2^6 + 2^7$. Selanjutnya, dilakukan akumulasi hasil perkalian matriks biner dan bobot 2^n dari seluruh piksel dalam blok, sehingga membentuk nilai pusat baru yang mewakili blok tersebut.

Seluruh blok pada citra diproses sesuai dengan urutannya, sehingga membentuk sebuah matriks baru hasil perhitungan LBP. Matriks LBP pada seluruh komponen wajah direpresentasikan dalam histogram. Seluruh proses ini diilustrasikan pada Gambar 5(a). Histogram berfungsi untuk merepresentasikan distribusi frekuensi pola biner lokal yang ada pada citra. Setiap *bin* pada histogram mewakili satu pola biner lokal, sedangkan tinggi *bin* menunjukkan jumlah kemunculan pola tersebut. Persamaan (5) merupakan formula untuk membentuk histogram setiap komponen fitur wajah.



Gambar 4. Alur kerja metode LBPH pada citra grayscale alis kanan.



Gambar 5. Konversi ke histogram, (a) tahapan konversi matriks LBP ke dalam histogram, (b) histogram seluruh komponen area fitur.

Notasi $S(x)$ merupakan vektor jumlah frekuensi kemunculan piksel yang dikategorikan sesuai dengan nilai bin. Notasi n adalah frekuensi kemunculan; pada $LBP_{8,1}$, jumlahnya 256 dengan range nilai 0 sampai 255.

$$H_i = \sum_{x_c \in I(x,y)} f \left\{ LBP_{P,R} = \sum_{n=0}^{P-1} S(x) \right\} \quad (5)$$

$$S(x) = \begin{cases} 1, & \text{if } y \text{ is true} \\ 0, & \text{if } y \text{ is false} \end{cases}, \quad i = 0, 1, \dots, n - 1$$

$$h_i = \frac{n_i}{n}, \quad i = 0, 1, \dots, L - 1. \quad (6)$$

Perhitungan histogram pada setiap komponen wajah diawali dengan menentukan bin histogram sesuai pola LBP. Pada penelitian ini, digunakan pola 8 bit untuk LBP ($LBP_{8,1}$), sehingga jumlah bin histogram adalah 256. Jumlah bin sesuai dengan piksel pada citra, yaitu antara 0 hingga 255, berjumlah 256. Selanjutnya, frekuensi kemunculan setiap nilai LBP pada citra dihitung dengan membuat matriks yang berukuran sesuai dengan jumlah bin histogram, yaitu 0 sampai 255. Data histogram citra LBP memiliki rentang nilai yang beragam, sehingga dapat memberikan prediksi yang tidak akurat pada proses klasifikasi. Maka, diperlukan proses normalisasi data.

Dalam normalisasi data histogram pada (6), nilai n_i adalah jumlah piksel dari derajat keabuan, n merupakan jumlah seluruh piksel dalam citra, dan L adalah jumlah bin dalam histogram.

$$H_{i,j} = \sum_{x_c \in R_j} f \{ LBP_{P,R} = \sum_{n=0}^{P-1} S(x) \}, \quad j = 0, 1, \dots, m - 1. \quad (7)$$

Proses selanjutnya adalah penggabungan seluruh data histogram dari masing-masing komponen wajah. Penggabungan data histogram dilakukan dengan cara menjumlahkan frekuensi kemunculan sesuai dengan nilai bin. Persamaan (7) menunjukkan formula histogram untuk menyimpan seluruh area dari komponen wajah setiap citra yang ditandai dengan notasi m area pada setiap frame. Notasi $H_{i,j}$ merupakan nilai i dari (5) yang menunjukkan urutan komponen area wajah dari histogram j . Histogram regional (per area) digabungkan untuk membuat deskripsi global citra. Penggabungan seluruh histogram dari komponen area wajah diilustrasikan pada Gambar 5(b).

Data histogram hasil penggabungan seperti ditunjukkan pada Gambar 5(b) merupakan data ekstraksi fitur untuk proses klasifikasi. Pada penelitian ini, dilakukan analisis dalam membandingkan data fase ekspresi mikro, yaitu "fullOAO" (data citra dari frame onset hingga offset) dan "OAO" (data citra hanya frame onset-apex-onset). Informasi data ekstraksi

fitur memiliki tiga jenis atribut, yaitu *frame*, data histogram, dan label setiap data, seperti ditunjukkan pada Gambar 5(b) pada bagian data ekstraksi fitur. Atribut utama sebagai data fitur adalah hasil dari data histogram dengan *bin* 0 sampai 255 dengan frekuensi kemunculan dengan nilai ternormalisasi.

D. REDUKSI FITUR (CUR MATRIX DECOMPOSITION)

Histogram dari setiap komponen citra wajah menghasilkan dimensi yang besar, yang diilustrasikan dalam Gambar 5(b). Setiap *frame* pada citra menghasilkan fitur sejumlah 256, sehingga pada data tersebut perlu dilakukan eliminasi fitur yang tidak informatif untuk proses klasifikasi data. Reduksi fitur ini bertujuan untuk menghilangkan fitur yang tidak relevan atau *redundancy* serta mempertahankan informasi penting yang terkandung dalam *dataset* tersebut. Teknik *CUR matrix decomposition* dapat memperoleh representasi data yang lebih efisien dari data fitur sebelumnya dengan mempertahankan subset kolom dan baris yang paling penting. Hal ini dapat bermanfaat dalam kondisi data fitur asli memiliki dimensi baris yang besar karena teknik ini dapat mengurangi kompleksitas komputasi dan kebutuhan penyimpanan data.

Pada penelitian ini, proses reduksi fitur dilakukan menggunakan metode *CUR matrix decomposition*. Metode tersebut dapat memilih fitur yang menunjukkan pengaruh statistik yang tinggi, atau dapat berpengaruh besar, dari data matriks. Teknik ini memecah matriks masukan menjadi tiga matriks yang lebih kecil, yaitu *C*, *U*, dan *R*. Berikut ini merupakan langkah-langkah reduksi fitur dengan *CUR matrix decomposition*.

1. Seleksi kolom (*C*), yaitu memilih sejumlah kolom dari matriks asli (*A*) yang akan membentuk matriks dengan metode pemilihan acak oleh sistem.
2. Seleksi baris (*R*), yaitu memilih sejumlah baris untuk membentuk matriks *R* dengan pemilihan berdasarkan pentingnya baris (saat ini seluruh data *frame* pada data ekstraksi fitur dianggap penting).
3. Pembentukan matriks tengah (*U*), yaitu memilih subset kolom dari matriks ekstraksi fitur. Dalam memilih subset pada penelitian ini, digunakan *sampling* acak.
4. Perhitungan bobot dilakukan setelah mendapatkan matriks *C*, *U*, dan *R*. Bobot dihitung untuk setiap baris dan kolom yang terpilih menggunakan (8) dan (9).

$$W_{C(i,i)} = \frac{1}{\text{sqr}t(\|col_{i(A)}\|_2 \times \|col_{i(C)}\|_2)} \quad (8)$$

$$W_{R(i,i)} = \frac{1}{\text{sqr}t(\|row_{i(A)}\|_2 \times \|row_{i(R)}\|_2)} \quad (9)$$

dengan $W_{C(i,i)}$ adalah bobot untuk kolom terpilih dalam matriks *C* dan $W_{R(i,i)}$ adalah bobot terpilih untuk baris terpilih dalam matriks *C* dan *R*. Tujuan perhitungan bobot adalah untuk menentukan kontribusi relatif dari setiap baris dan kolom terpilih dari matriks masukan.

5. Aproksimasi matriks masukan dilakukan setelah mendapatkan matriks *C*, *U*, *R*, dan bobot. Aproksimasi matriks masukan A' dapat dihitung dengan menggabungkan ketiga matriks tersebut, yaitu menggunakan (10).

$$A' = C \times W_C \times U \times W_R \times R \quad (10)$$

dengan W_C dan W_R merupakan matriks diagonal bobot untuk kolom dan baris yang terpilih dari aproksimasi matriks masukan.

E. KLASIFIKASI EKSPRESI MIKRO MENGGUNAKAN KNN DAN SVM

1) K-NEAREST NEIGHBORS (KNN)

Penelitian ini mengimplementasikan metode KNN untuk membentuk sebuah model data. Prinsip dari metode ini adalah setiap titik pada data yang berdekatan akan berada pada label/kelas yang sama. Artinya, metode KNN dapat melakukan klasifikasi data baru berdasarkan kemiripan dari model yang tersedia [30]. Metode ini merupakan salah satu jenis pembelajaran berdasarkan pengaktualan yang fungsi-fungsi tersebut merupakan nilai kedekatan secara lokal. Pada penelitian ini, tahapan klasifikasi membandingkan tiga metode perhitungan jarak, yaitu Euclidean, Manhattan, dan Chebyshev. Berikut ini adalah langkah-langkah untuk menghitung KNN.

$$\text{Euclidean}(a, b) = \sqrt{\sum_{i=1}^n (x_{ai} - y_{bi})^2} \quad (11)$$

$$\text{Manhattan}(a, b) = \sum_{i=1}^n |x_{ai} - y_{bi}| \quad (12)$$

$$\text{Chebyshev}(a, b) = \max_i (|x_{ai} - y_{bi}|) \quad (13)$$

dengan *a* adalah kluster data pusat, *b* adalah data dalam atribut fitur, *i* merupakan notasi untuk setiap data pada *n*, *n* menunjukkan banyaknya data, x_{ai} menunjukkan data dalam kluster pusat hingga *k*, sedangkan y_{bi} adalah setiap objek data hingga ke-*k*.

Metode jarak Euclidean dianggap sebagai dasar dari metode kesamaan atau ketidaksamaan. Pada penelitian ini, digunakan (11) untuk menghitung jarak Euclidean antara elemen yang sesuai dari dua ruang vektor. Perhitungan jarak Chebyshev dilakukan menggunakan selisih terbesar dari dua data masukan, seperti pada (13). Sementara itu, metode jarak Manhattan merupakan metode lain untuk mengukur jarak antara dua vektor, seperti ditunjukkan pada (12).

2) SUPPORT VECTOR MACHINE (SVM)

Metode SVM digunakan dalam penelitian karena terbukti berhasil dalam pembelajaran statistika, pemodelan optimalisasi data, deteksi objek, serta pengenalan ekspresi mikro. Pemodelan klasifikasi dalam SVM memiliki konsep yang lebih baik secara matematis daripada teknik klasifikasi lainnya. Model dalam SVM dapat mengatasi masalah terkait klasifikasi secara linear maupun nonlinear. Dalam penelitian ini, fungsi dengan *kernel* linear, polinomial, *radial basis function* (RBF), dan *sigmoid* diterapkan pada SVM. Persamaan matematika dari keempat *kernel* tersebut terdapat dituliskan pada (14) sampai (16).

$$\text{Linear } K(x_i, x_j) = x_i^T x_j + c \quad (14)$$

$$\text{polynomial } K(x_i, x_j) = (c + x_i^T x_j)^b \quad (15)$$

$$\text{RBF } K(x_i, x_j) = \exp\left(\frac{\|x_i - x_j\|}{2S^2}\right) \quad (16)$$

dengan x_i dan x_j merupakan vektor masukan, $K()$ adalah fungsi *kernel*, *c* merupakan konstanta, dan *S* merupakan parameter *bandwidth* yang menentukan kecepatan turunnya kesamaan metrik saat sampel menjauh satu sama lain. Notasi x_i^T merupakan nilai transposisi dari vektor bobot *x*, sedangkan notasi *b* adalah nilai dari derajat polinomial.

III. HASIL DAN DISKUSI

A. PENGATURAN DATA

Penelitian ini membandingkan tingkat akurasi pada pengenalan ekspresi mikro ke dalam dua jenis fase ekspresi

TABEL II
 JUMLAH FRAME PERBANDINGAN DATA FASE EKSPRESI MIKRO

No	Jenis Fase	Jumlah Frame	Dimensi Fitur	Reduksi Dimensi Fitur
1	fullOAO	1.155	256	35
2	OAO	82	256	35

yang disajikan pada Tabel I. Jenis fase “fullOAO” menggunakan seluruh *frame* pada fase *onset* hingga *offset*, sedangkan jenis “OAO” hanya menggunakan tiga *frame*, yang terdiri atas *frame onset*, *apex*, dan *offset*. Penelitian ini menggunakan 27 video pilihan dari CASME II dengan empat label data. Deskripsi lebih lanjut disajikan dalam Tabel I. Tujuan pemilihan video adalah untuk mempertimbangkan jumlah *frame* dari setiap fase *onset* hingga *offset*, untuk menghindari ketidakseimbangan jumlah data. Dalam penelitian ini, pengenalan ekspresi mikro dilakukan menggunakan lima komponen wajah, yaitu alis kiri, alis kanan, mata kiri, mata kanan, dan mulut. Hasil dari metode LPBH adalah penggabungan jumlah frekuensi kemunculan pada lima komponen wajah yang memiliki bentuk runtun data kombinasi jumlah piksel dengan nilai 0 sampai 255. Dengan menggunakan lima komponen wajah, dapat disimpulkan bahwa data hasil ekstraksi fitur berdimensi tinggi.

Tabel II menyajikan informasi jumlah dimensi fitur pada “fullOAO” dan “OAO” sebesar 256 fitur pada setiap *frame*-nya. Proses histogram pada ekstraksi fitur menghasilkan distribusi data bernilai 0 dan bernilai sama pada beberapa piksel. Hal tersebut memberikan dampak pada proses klasifikasi, seperti *redundancy* data, sehingga diperlukan proses reduksi data menggunakan teknik *CUR matrix decomposition*. Deskripsi jumlah *frame* pada reduksi data, jenis fase, dan kelas ekspresi disajikan pada Tabel II. Dengan menggunakan teknik tersebut, dihasilkan jumlah fitur lebih kecil, yaitu 35 data, tetapi dianggap memiliki representasi lebih efektif dengan mempertahankan fitur yang paling penting atau signifikan.

B. PENGUKURAN KINERJA PENGENALAN EKSPRESI MIKRO

Dalam penelitian ini, SVM dan KNN dipilih sebagai metode untuk klasifikasi eksperimen. Pemilihan fungsi *kernel* untuk SVM dan metode pengukuran jarak sangat penting bagi kinerja masing-masing metode. Eksperimen dilakukan pada CASME II menggunakan metode ekstraksi fitur LBPH. Pengklasifikasian SVM dan KNN digunakan untuk mengevaluasi metode yang diusulkan menggunakan analisis *multi-subject*, *leave-one-out cross validation* (LOSOCV), dan *10-cross validation*. LOSOCV merupakan metode validasi silang yang digunakan untuk mengestimasi kinerja model atau algoritma pada *dataset* terbatas (memiliki jumlah yang sedikit). Penggunaan kedua teknik analisis tersebut bertujuan untuk mendapatkan estimasi kinerja model yang paling akurat. Dalam melakukan validasi keefektifan metode yang diusulkan dalam LOSOCV, urutan citra dari satu subjek diperlakukan sebagai data pengujian dan citra *frame* tersisa diperlakukan sebagai data pelatihan. Proses tersebut diulang sebanyak *k* kali, dengan *k* menunjukkan jumlah subjek dalam *dataset*. Kemudian, dilakukan perhitungan rata-rata hasil pengenalan semua subjek untuk membentuk akurasi pengenalan akhir.

Dalam *10-fold cross validation*, kumpulan data dibagi menjadi sepuluh bagian. Lalu, sembilan bagian data diambil sebagai data pelatihan secara bergantian dan satu bagian lainnya sebagai data uji. Tingkat yang benar diperoleh dari

TABEL III
 PERBANDINGAN *DISTANCE METRICS* KNN PADA FASE “FULLOAO” DAN “OAO”

KNN - fullOAO						
<i>k</i>	LOSOCV (%)			10-cross validation (%)		
	EC	MH	CS	EC	MH	CS
1	95,8	95,9	94,3	95,5	95,6	94
3	96,2	95,5	94,3	96,1	95,4	94,2
5	96,1	96,6	95	96,3	96,6	96,3
KNN - OAO						
<i>k</i>	LOSOCV (%)			10-cross validation (%)		
	EC	MH	CS	EC	MH	CS
1	84,0	87,7	81,5	85,2	87,7	80,2
3	81,5	84,0	81,5	79,0	85,2	77,8
5	76,5	79,0	74,1	65,4	77,8	67,9

Ket.: EC = Euclidean, MH = Manhattan, CS = Chebyshev

TABEL IV
 PERBANDINGAN *KERNEL* SVM PADA FASE “OAO” DAN “FULLOAO”

Kernel	fullOAO		OAO	
	LOSOCV (%)	10-Cross (%)	LOSOCV (%)	10-Cross (%)
Linear	53,9	53,7	64,2	63
Polynomial	96,3	96,3	81,5	81,5
RBF	96,7	96,8	74,1	71,6
Sigmoid	13,6	15,6	39,5	46,9

setiap pengujian dan nilai rata-rata dari tingkat yang benar digunakan untuk memperkirakan akurasi sebuah algoritma. Sebagai tahapan terakhir, penelitian ini melakukan *10-fold cross validation* dan mendapatkan nilai rata-rata sebagai akurasi tertinggi. Dalam penelitian ini, dua metode *cross validation* yang umum digunakan untuk mengevaluasi kemampuan metode klasifikasi dengan metode SVM dan KNN untuk pengenalan.

1) PENGUJIAN KLASIFIKASI DENGAN PERBANDINGAN NILAI *k* DAN METRIK JARAK PADA KNN

Pengujian KNN dilakukan dengan menggunakan sampel seperti tertera pada Tabel III. Perhitungan jarak dalam KNN menggunakan tiga perbandingan metode, yaitu Euclidean, Manhattan, dan Chebyshev. Penentuan nilai *k* pada KNN bergantung pada data masukan. Nilai *k* yang tinggi akan mengurangi efek dari *noise*, tetapi akan membentuk batasan dari setiap hasil klasifikasi. Penelitian ini menggunakan tiga macam nilai ganjil pada nilai *k*, yaitu 1, 3, dan 5. Karena jumlah label data bernilai genap, penelitian ini menggunakan nilai *k* ganjil.

Hasil perhitungan jarak Euclidean, Manhattan, dan Chebyshev terhadap akurasi KNN ditunjukkan pada Tabel IV. Pada data “fullOAO” dengan analisis LOSOCV dan *10-cross validation*, rata-rata hasilnya di atas 90% pada setiap nilai *k*. Hasil akurasi tertinggi menggunakan nilai *k* = 5 dan perhitungan jarak Manhattan, dengan akurasi LOSOCV dan *10-cross validation* sebesar 96,6%. Pada “fullOAO”, perbedaan nilai *k* tidak berpengaruh pada tingkat akurasi karena pada setiap nilai *k* dalam Tabel III, akurasi yang dihasilkan pada setiap metode di atas 90%. Pada data jenis fase “OAO”, penggunaan nilai *k* yang makin besar menghasilkan tingkat akurasi yang makin kecil pada setiap analisisnya. Nilai akurasi tertinggi terjadi pada *k* = 1 dengan pengukuran jarak Manhattan dengan nilai 87,7%. Manhattan memiliki kecocokan dengan data yang bersifat geometris, dengan jarak antartitik

dapat diukur berdasarkan perbedaan absolut antara koordinatnya pada setiap dimensi.

2) PENGUKURAN KLASIFIKASI DENGAN PERBANDINGAN KERNEL SVM

Pengujian dengan SVM dilakukan menggunakan sampel yang sama sesuai deskripsi sebelumnya. Dalam proses klasifikasi SVM, tiga parameter penting perlu dipilih dan disesuaikan. Parameter pertama adalah *kernel*. Penelitian ini membandingkan empat *kernel* yang umum digunakan pada SVM, yaitu *kernel* linear, polinomial, RBF, dan *sigmoid*. Parameter kedua adalah *penalty coefficient*, C , yang merupakan nilai toleransi kesalahan. Nilai tersebut dapat meminimalkan kesalahan pelatihan dan kompleksitas model. Makin tingginya nilai C menunjukkan bahwa kesalahan tidak dapat ditoleransi dan mudah terjadi *over-fit*. Makin kecil nilai C , makin mudah dilakukan *under-fit* [31]. Lalu, parameter ketiga adalah parameter γ , yang merupakan parameter dari fungsi RBF ketika dipilih sebagai *kernel*. Nilai lebar RBF akan memengaruhi *range of action* dari masing-masing *support vector* yang sesuai dengan Gauss, sehingga memengaruhi kemampuan secara umum. Tabel IV menyajikan hasil analisis dengan dua jenis fase, “fullAOA” dan “OAO”, dengan menggunakan LOSOCV dan 10-cross validation. Data “fullOAO” dengan *kernel* RBF pada kedua metode pembelajaran menghasilkan nilai tertinggi, yaitu 96,7% dan 96,8%. Metode SVM dengan *kernel* RBF dapat memisahkan data dengan dimensi tinggi yang sulit dipisahkan dengan metode linier. Berbeda dengan jenis data sebelumnya, pada “OAO” *kernel* polinomial dihasilkan akurasi tertinggi, yaitu 81,5% pada kedua metode tersebut. Hal ini terjadi karena *kernel* polinomial dapat menangkap hubungan antarfitur yang lebih kompleks dibandingkan dengan metode *kernel* lainnya.

C. PENGUJIAN WAKTU KOMPUTASI PENGENALAN EKSPRESI MIKRO

Selain melakukan pengujian kinerja akurasi, penelitian ini melakukan pengujian waktu komputasi dari algoritma yang diusulkan (lihat Tabel V). Waktu komputasi algoritma dianalisis pada prosesor 11th Gen Intel® Core™ i3-1115G4 @ 3,00 GHz dengan *random access memory* (RAM) sebesar 8 GB. Waktu komputasi yang dibutuhkan untuk mengekstrak fitur pada setiap *frame* rata-rata sebesar 0,192 ms untuk fullOAO dan 0,159 ms pada OAO. Waktu komputasi pada kedua jenis fase ekspresi hampir sama karena pada dasarnya setiap *frame* diproses dengan algoritma yang sama. Berbeda dengan pengujian per *frame*, pada pengujian per video dihasilkan waktu yang berbeda. Pada jenis fase “fullOAO”, dihasilkan rata-rata waktu komputasi 10,473 ms, sedangkan pada OAO adalah 0,576 ms.

Perbedaan waktu antara kedua jenis fase tersebut disebabkan oleh perbedaan jumlah *frame* yang diproses. Secara teknis, jumlah *frame* pada “fullOAO” lebih besar daripada pada jenis fase “OAO” (lihat Tabel I), sehingga berdampak pada waktu pemrosesan pada setiap video.

Implementasi sistem pengenalan ekspresi mikro pada umumnya dilakukan secara *real-time*, sehingga diperlukan waktu komputasi dalam pemrosesan setiap *frame* yang cepat. Pada dasarnya, karakteristik ekspresi mikro memiliki durasi yang cepat, dengan intensitas gerakan yang kecil dan halus, sehingga sulit diidentifikasi oleh mata manusia biasa. Dari pernyataan tersebut, dapat disimpulkan bahwa sistem pengenalan ekspresi mikro membutuhkan waktu komputasi yang cepat untuk memproses setiap *frame* pada video masukan.

TABEL V
PERBANDINGAN WAKTU KOMPUTASI SETIAP JENIS FASE

No	Jenis Pengujian	fullOAO (ms)	OAO (ms)
1	Rata-rata per <i>frame</i>	0,192	0,159
2	Rata-rata per video	10,473	0,576

TABEL VI
PERBANDINGAN AKURASI DENGAN PENELITIAN LAINNYA DENGAN LOSOCV

Penelitian	Ekstraksi Fitur	Classifier	Kinerja (%)
[32]	LBP-TOP	SMO	68,24
[33]	Bi-WOOF	SVM	58,85
[34]	OFF-ApexNet	CNN	88,28
Penelitian ini	LBPH	SVM	96,80

Pada penelitian ini, jenis fase “OAO”, khususnya pada pengujian video, menghasilkan waktu komputasi 20 kali lebih cepat daripada “fullOAO”. Hal ini disebabkan oleh perbedaan jumlah *frame* yang diproses pada kedua jenis *frame* tersebut. Pada jenis fase “OAO”, *frame* yang diproses merupakan *frame* inti (fase ekspresi mikro: *onset*, *apex*, dan *offset*), sedangkan pada “fullOAO”, *frame* yang diproses adalah dari *frame onset* hingga *offset*, sehingga membutuhkan waktu lebih lama.

D. HASIL KINERJA AKURASI DAN WAKTU KOMPUTASI TERHADAP JENIS FASE EKSPRESI

Pada bagian sebelumnya telah dilakukan tahapan klasifikasi menggunakan perbandingan metode KNN dan SVM dengan berbagai kombinasi *kernel* dan pengukuran jarak. Berdasarkan pengujian klasifikasi KNN dan SVM, diketahui bahwa pada jenis fase “fullOAO” dihasilkan akurasi yang hampir sama, yaitu 96,8% untuk SVM dan 96,6% untuk KNN. Sementara itu, pada jenis fase “OAO” terdapat sedikit penurunan akurasi dibandingkan dengan “fullOAO”, yaitu sebesar 81,5% pada SVM dan 87,7% untuk KNN. Penurunan akurasi tersebut terjadi karena perbedaan jumlah data dari setiap jenis fase ekspresi. Pada “fullAOA”, jumlah *frame* adalah sebesar 1.115, sedangkan pada “OAO” terdapat 82 *frame* (lihat Tabel II). Dari hasil kinerja akurasi tersebut, dapat disimpulkan bahwa makin banyak *frame* yang dianalisis, makin baik tingkat akurasi yang didapatkan.

Untuk mengevaluasi kinerja algoritma yang diusulkan pada penelitian ini, bagian ini membahas perbandingan dengan penelitian yang telah dilakukan dengan teknik berbeda, seperti ditunjukkan pada Tabel VI. Perbandingan penelitian pada Tabel VI seluruhnya menggunakan *dataset* CASME II dan teknik klasifikasi LOSOCV dengan metode klasifikasi yang berbeda, seperti *sequential minimal optimization* (SMO), *convolutional neural network* (CNN), dan SVM. Pada Tabel VI, penelitian lain menghasilkan akurasi tertinggi sebesar 68,24% dengan metode LBP-TOP dan SMO [32], sedangkan penggunaan *bi-weighted oriented optical flow* (Bi-WOOF) dan SVM menghasilkan akurasi sebesar 58,85% [33]. Referensi [34] menghasilkan akurasi lebih baik dari kedua penelitian sebelumnya dengan menggunakan OFF-ApexNet dan CNN, dengan akurasi sebesar 88,28%. Sementara itu, algoritma yang diusulkan pada penelitian ini menggunakan LBPH dan SVM, yang menghasilkan akurasi sebesar 96,8%.

Tingkat akurasi pada pengenalan ekspresi mikro sangat penting karena hasil yang akurat diperlukan untuk memahami perasaan dan emosi pada setiap data ekspresi masukan. Pada penelitian ini, diusulkan metode ekstraksi fitur menggunakan

TABEL VII
 KINERJA KOMPUTASI UNTUK PENELITIAN INI DAN PENELITIAN TERDAHULU

Penelitian	Ekstraksi Fitur	Prapemrosesan	Waktu per Frame (ms)
[17]	LBP-TOP	Face detection, face cropping, face alignment, dan onset-offset detection	2,980
[18]	LBP-TOP	Face detection, face alignment, face cropping, dan onset-offset detection	107,390
Penelitian ini	LBPH	Facial landmarks detection, facial component cropping	0,175

LBPH yang menghasilkan akurasi akhir dari hasil penelitian sebesar 96,8% pada SVM dan 87,7% pada KNN. Perbedaan akurasi tersebut tidak terlalu jauh, yaitu sebesar 9%, dengan akurasi terendah tergolong baik, yaitu lebih dari 80%. Selain itu, data ekstraksi fitur sebagai data masukan untuk klasifikasi adalah data yang sama, sehingga perbedaan akurasi tersebut tidak berdampak besar pada pengenalan ekspresi mikro. Hasil tersebut dapat dikategorikan baik karena metode yang diusulkan memiliki kinerja yang kuat dalam melakukan pengenalan ekspresi mikro pada area wajah. Pengenalan ekspresi mikro dilakukan dengan proses secara *real-time* dengan waktu yang singkat. Selain hasil akurasi yang baik, waktu pemrosesan untuk pengenalan perlu dipertimbangkan.

Tabel VII menyajikan rangkuman waktu pemrosesan pada setiap *frame* dari penelitian terdahulu [17], [18] dan penelitian yang diusulkan. Pada bagian ini, tidak disajikan kesimpulan akhir terkait perbedaan waktu komputasi setiap penelitian. Hal tersebut disebabkan oleh adanya perbedaan spesifikasi alat yang digunakan di setiap penelitian dan tahapan prapemrosesan yang diterapkan. Pada [18] digunakan *tensor unfolding* dengan *graphics processing unit* (GPU) menghasilkan waktu komputasi sebesar 107,39 ms, sedangkan pada [17] digunakan LBP-TOP berbasis GPU dengan platform CUDA, yang membutuhkan waktu komputasi sebesar 2,98 ms. Pada kedua penelitian tersebut, dibutuhkan tahapan prapemrosesan, termasuk deteksi wajah dari setiap *frame*, *face alignment*, *face cropping*, dan pendeteksian fase *onset* hingga *offset*. Beberapa tahapan tersebut membutuhkan waktu yang tergolong lama dalam pemrosesan. Di sisi lain, dalam penelitian ini, waktu pemrosesan yang dibutuhkan pada setiap *frame* seluruh komponen wajah terpilih adalah 0,175 ms. Waktu komputasi penelitian yang diusulkan menghasilkan proses yang lebih cepat dibandingkan dengan penelitian sebelumnya [17], [18]. Akan tetapi, penelitian yang diusulkan tidak mengimplementasikan proses pendeteksian fase *onset* hingga *offset* secara otomatis, sehingga membutuhkan waktu pemrosesan yang lebih singkat dibandingkan penelitian lainnya.

IV. KESIMPULAN

Pengenalan ekspresi mikro membutuhkan keseimbangan antara akurasi dan waktu pemrosesan. Secara umum, semakin cepat suatu sistem dapat mengenali dan menginterpretasi ekspresi mikro pada wajah, makin baik kualitas respons pada sistem tersebut. Namun, dalam meningkatkan akurasi sering kali digunakan metode yang lebih kompleks dan metode analisis yang canggih, sehingga waktu pemrosesan menjadi lebih lama. Pada penelitian ini, diusulkan metode sederhana,

yaitu LBPH untuk proses ekstraksi fitur. Perbandingan fase ekspresi pada ekspresi mikro berdasarkan fase inti (*onset*, *apex*, dan *offset*) diterapkan pada penelitian ini. Metode yang diusulkan diharapkan dapat memberikan waktu komputasi yang cepat dan pengenalan ekspresi mikro yang tepat. Pada penelitian ini, dilakukan perbandingan jenis fase ekspresi mikro, yaitu “fullOAO” (fase *onset* hingga *offset*) dan “OAO” (fase *onset*, *apex*, dan *offset*) dengan metode ekstraksi fitur LBPH.

Eksperimen pada penelitian ini menggunakan salah satu *dataset* ekspresi mikro, yaitu CASME II. Dalam penelitian ini, SVM dan KNN dipilih sebagai metode untuk klasifikasi eksperimental. Pemilihan fungsi *kernel* untuk SVM dan metode pengukuran jarak sangat penting untuk kinerja masing-masing metode. Dalam analisis multisubjek, LOSOCV dan 10-*cross validation* digunakan untuk melakukan validasi keefektifan metode yang diusulkan dalam LOSOCV, dengan urutan citra dari satu subjek diperlakukan sebagai data pengujian dan citra *frame* tersisa sebagai data pelatihan.

Pada “fullOAO” dihasilkan akurasi terbaik sebesar 96,8% (SVM-RBF), sedangkan pada “OAO” (KNN-Manhattan) diperoleh akurasi sebesar 87,7%. Perbedaan akurasi pada jenis fase ekspresi tersebut berbanding terbalik dengan waktu komputasi, yaitu pada “OAO” 20 kali lebih cepat daripada pada “fullOAO”. Pada “OAO” dibutuhkan waktu 0,576 ms, sedangkan pada “fullOAO” diperlukan waktu lebih lama, yaitu 10,473 ms. Perbedaan waktu komputasi tersebut disebabkan oleh perbedaan jumlah *frame* yang diproses pada kedua jenis fase ekspresi tersebut. Secara teknis, jumlah *frame* pada jenis “fullOAO” lebih besar daripada jumlah *frame* pada “OAO”, sehingga waktu pemrosesan juga berbeda.

Penurunan akurasi sebesar 9% pada kedua jenis fase tersebut tidak berpengaruh terlalu besar karena pada kedua jenis fase ekspresi tersebut didapatkan akurasi yang tergolong baik, yaitu di atas 80%. Selain itu, perbedaan waktu komputasi dari kedua jenis fase ekspresi bergantung pada jumlah *frame* setiap video masukan. Maka, pengukuran yang tepat untuk waktu komputasi adalah waktu yang dibutuhkan untuk memproses setiap *frame* pada video masukan. Pada penelitian ini, waktu pemrosesan setiap *frame* lebih cepat, yaitu 0,192 ms pada “fullOAO”, dibandingkan dengan penelitian terdahulu. Oleh karena itu, dapat disimpulkan bahwa dengan algoritma yang diusulkan, diperoleh waktu komputasi yang cepat dengan tingkat pengenalan ekspresi mikro tergolong akurat.

KONFLIK KEPENTINGAN

Penulis memastikan bahwa tidak ada konflik kepentingan, baik dalam keadaan tertentu atau kepentingan pribadi yang memengaruhi penyajian atau interpretasi dari hasil penelitian.

KONTRIBUSI PENULIS

Konseptualisasi, Ulla Delfana Rosiani dan Priska Choirina; metodologi, Ulla Delfana Rosiani dan Yessy Nindi Pratiwi; perangkat lunak, Yessy Nindi Pratiwi dan Septiar Enggar Sukmana; validasi, Ulla Delfana Rosiani dan Priska Choirina; analisis formal, Septiar Enggar Sukmana.

REFERENSI

- [1] L. Zhou, X. Shao, dan Q. Mao, “A Survey of Micro-Expression Recognition,” *Image, Vis. Comput.*, Vol. 105, hal. 1-11, Jan. 2021, doi: 10.1016/j.imavis.2020.104043.
- [2] P. Ekman, “Lie Catching and Microexpressions,” dalam *The Philosophy of Deception*, C. Martin, Ed., New York, AS: Oxford University Press, 2009.

- [3] P. Ekman dan W.V. Friesen, *Facial Action Coding System: A Technique for the Measurement of Facial Movement*. Palo Alto, AS: Consulting Psychologists Press, 1978.
- [4] M. Peng, Z. Wu, Z. Zhang, dan T. Chen, "From Macro to Micro Expression Recognition: Deep Learning on Small Datasets Using Transfer Learning," *2018 13th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG 2018)*, 2018, hal. 657-661, doi: 10.1109/FG.2018.00103.
- [5] C.H. Yap, C. Kendrick, dan M.H. Yap, "SAMM Long Videos: A Spontaneous Facial Micro- and Macro-Expressions Dataset," 2019, *arXiv:1911.01519*.
- [6] P. Ekman dan W.V. Friesen, "Nonverbal Leakage and Clues to Deception," *Psychiatry*, Vol. 32, No. 1, hal. 88-106, 1969, doi: 10.1080/00332747.1969.11023575.
- [7] H.-X. Xie, L. Lo, H.-H. Shuai, dan W.-H. Cheng, "AU-assisted Graph Attention Convolutional Network for Micro-Expression Recognition," *Proc. 28th ACM Int. Conf. Multimed.*, 2020, hal. 2871-2880, doi: 10.1145/3394171.3414012.
- [8] J. Ma, H. Tang, W.-L. Zheng, dan B.-L. Lu, "Emotion Recognition Using Multimodal Residual LSTM Network," *Proc. 27th ACM Int. Conf. Multimed.*, 2019, hal. 176-183, doi: 10.1145/3343031.3350871.
- [9] I.P. Adegun dan H.B. Vadapalli, "Facial Micro-Expression Recognition: A Machine Learning Approach," *Sci. Afr.*, Vol. 8, hal. 1-14, Jul. 2020, doi: 10.1016/j.sciaf.2020.e00465.
- [10] Y. Zhu, Z. Chen, dan F. Wu, "Multimodal Deep Denoise Framework for Affective Video Content Analysis," *Proc. 27th ACM Int. Conf. Multimed.*, 2019, hal. 130-138.
- [11] A.M. Buhari dkk., "FACS-Based Graph Features for Real-Time Micro-Expression Recognition," *J. Imaging*, Vol. 6, No. 12, hal. 1-20, Des. 2020, doi: 10.3390/jimaging6120130.
- [12] S.-J. Wang dkk., "Micro-Expression Recognition Using Robust Principal Component Analysis and Local Spatiotemporal Directional Features," dalam *Computer Vision - ECCV 2014 Workshops*, L. Agapito, M. M. Bronstein, and C. Rother, Eds., Cham, Swiss: Springer International Publishing, 2015, hal. 325-338, doi: 10.1007/978-3-319-16178-5_23.
- [13] Y. Wang, J. See, R. C.-W. Phan, dan Y.-H. Oh, "LBP with Six Intersection Points: Reducing Redundant Information in LBP-TOP for Micro-expression Recognition," dalam *Computer Vision - ACCV 2014*, D. Cremers, I. Reid, H. Saito, and M.-H. Yang, Eds., Cham, Swiss: Springer International Publishing, 2015, hal. 525-537, doi: 10.1007/978-3-319-16865-4_34.
- [14] P. Zhang dkk., "Micro-Expression Recognition System," *Optik*, Vol. 127, No. 3, hal. 1395-1400, Feb. 2016, doi: 10.1016/j.ijleo.2015.10.217.
- [15] M.N. Patil, B. Iyer, dan R. Arya, "Performance Evaluation of PCA and ICA Algorithm for Facial Expression Recognition Application," dalam *Proceedings of Fifth International Conference on Soft Computing for Problem Solving*, SocProS 2015, Vol. 1, M. Pant dkk., Eds., Singapura, Singapura: Springer, 2016, hal. 965-976, doi: 10.1007/978-981-10-0448-3_81.
- [16] W.-L. Chao, J.-J. Ding, dan J.-Z. Liu, "Facial Expression Recognition Based on Improved Local Binary Pattern and Class-Regularized Locality Preserving Projection," *Signal Process.*, Vol. 117, hal. 1-10, Des. 2015, doi: 10.1016/j.sigpro.2015.04.007.
- [17] X.R. Soh, V.M. Baskaran, A.M. Buhari, dan R.C.-W. Phan, "A Real Time Micro-Expression Detection System with LBP-TOP on a Many-Core Processor," *2017 Asia-Pacific Signal, Inf. Process. Assoc. Annu. Summit, Conf. (APSIPA ASC)*, 2017, hal. 309-315, doi: 10.1109/APSIPA.2017.8282041.
- [18] X. Hong, Y. Xu, dan G. Zhao, "LBP-TOP: A Tensor Unfolding Revisit," dalam *Computer Vision - ACCV 2016 Workshops*, C.-S. Chen, J. Lu, and K.-K. Ma, Eds., Cham, Swiss: Springer International Publishing, 2017, hal. 513-527, doi: 10.1007/978-3-319-54407-6_34.
- [19] N. Samadiani dkk., "A Review on Automatic Facial Expression Recognition Systems Assisted by Multimodal Sensor Data," *Sens.*, Vol. 19, No. 8, hal. 1-27, Apr. 2019, doi: 10.3390/s19081863.
- [20] P. Choirina, U.D. Rosiani, dan I.M. Fitriani, "Pengenalan Ekspresi Mikro Wajah Berdasarkan Point Feature Tracking Menggunakan Fase Apex pada Database Ekspresi Mikro," *Edu Komputika J.*, Vol. 9, No. 1, hal. 28-36, Jun. 2022, doi: 10.15294/edukomputika.v9i1.56600.
- [21] U.D. Rosiani dkk., "A Novel Approach on Motion Estimation for Micro-Expression Recognition Using Phase Only Correlation with All Block Search (POC-ABS)," *Int. J. Intell. Eng., Syst.*, Vol. 13, No. 6, hal. 546-559, 2020, doi: 10.22266/ijes2020.1231.48.
- [22] U.D. Rosiani, P. Choirina, S. Sumpeno, dan M. Hery P., "Menuju Pengenalan Ekspresi Mikro: Pendeteksian Komponen Wajah Menggunakan Discriminative Response Map Fitting," *J. Nas. Tek. Elekt., Teknol. Inf.*, Vol. 7, No. 2, hal. 204-211, Mei 2018, doi: 10.22146/jnteti.v7i2.424.
- [23] S.P.T. Reddy, S.T. Karri, S.R. Dubey, dan S. Mukherjee, "Spontaneous Facial Micro-Expression Recognition Using 3D Spatiotemporal Convolutional Neural Networks," *2019 Int. Joint Conf. Neural Netw. (IJCNN)*, 2019, hal. 1-8, doi: 10.1109/IJCNN.2019.8852419.
- [24] W.-J. Yan dkk., "CASME II: An Improved Spontaneous Micro-Expression Database and the Baseline Evaluation," *PLoS ONE*, Vol. 9, No. 1, hal. 1-8, Jan. 2014, doi: 10.1371/journal.pone.0086041.
- [25] "CASME II Database," [Online], <http://casme.psych.ac.cn/casme/e2>, tanggal akses: 26-Jun-2023.
- [26] M. Xu, D. Chen, dan G. Zhou, "Real-Time Face Recognition Based on Dlib," dalam *Innovative Computing*, C.-T. Yang, Y. Pei, dan J.-W. Chang, Eds., Singapura, Singapura: Springer, 2020, hal. 1451-1459, doi: 10.1007/978-981-15-5959-4_177.
- [27] T.Q. Vinh dan N.T.N. Anh, "Real-Time Face Mask Detector Using YOLOv3 Algorithm and Haar Cascade Classifier," *2020 Int. Conf. Adv. Comput., Appl. (ACOMP)*, 2020, hal. 146-149, doi: 10.1109/ACOMP50827.2020.00029.
- [28] N.M. Parsania, K.H. Solanki, dan A.R. Mehta. (3-4 Sep. 2021). Innovative Approach for Fingerprint Recognition Using LBP and PCA Algorithms. Dipresentasikan dalam 1st Int. Conf. Adv. Inf. Technol., Commun. (IC-AITC), [Online], <https://www.youtube.com/watch?v=8x3vsNylG0>
- [29] V. Esmaeili, M.M. Feghhi, and S.O. Shahdi, "Micro-Expression Recognition Using Histogram of Image Gradient Orientation on Diagonal Planes," *2021 5th Int. Conf. Pattern Recognit., Image Anal. (IPRIA)*, 2021, hal. 1-5, doi: 10.1109/IPRIA53572.2021.9483500.
- [30] D. Rahmawati dkk., "The Design of Facial Expression Detection System to Determine the Level of Customer Satisfaction Using K-Nearest Neighbor Method," *MATEC Web Conf.*, Vol. 372, 2022, hal. 1-5, doi: 10.1051/mateconf/202237206002.
- [31] H. Pan dkk., "Hierarchical Support Vector Machine for Facial Micro-Expression Recognition," *Multimed. Tools, Appl.*, Vol. 79, hal. 31451-31465, Nov. 2020, doi: 10.1007/s11042-020-09475-4.
- [32] A.K. Davison, W. Merghani, dan M.H. Yap, "Objective Classes for Micro-Facial Expression Recognition," *J. Imag.*, Vol. 4, No. 10, hal. 1-13, Oct. 2018, doi: 10.3390/jimaging4100119.
- [33] S.-T. Liang, J. See, K. Wong, dan R.C.-W. Phan, "Less is More: Micro-Expression Recognition from Video Using Apex Frame," *Signal Process.: Image Commun.*, Vol. 62, hal. 82-92, Mar. 2018, doi: 10.1016/j.image.2017.11.006.
- [34] Y. S. Gan dkk., "OFF-ApexNet on Micro-Expression Recognition System," *Signal Process.: Image Commun.*, Vol. 74, hal. 129-139, Mei 2019, doi: 10.1016/j.image.2019.02.005.