

Sistem Presensi Karyawan Berbasis Pengenalan Wajah Dengan Metode *Support Vector Machine*

David Setiyadi¹, Fauzun Atabiq^{1*}, dan Siti Aisyah¹

¹Politeknik Negeri Batam, Batam, Indonesia

*Email: atabiq@polibatam.ac.id

Abstrak—Sistem presensi saat ini yang ada pada instansi ataupun perusahaan masih banyak yang menggunakan sistem manual. Disisi lain, perusahaan-perusahaan tersebut juga telah memiliki aplikasi pengelolaan SDM *online*. Oleh karena itu, untuk efektifitas dan pengembangan sistem, perlu dilakukan pengembangan sistem presensi manual tersebut menjadi sebuah sistem yang dapat diintegrasikan dengan sistem pengelolaan SDM. Untuk itu, penelitian ini mengusulkan pengembangan sistem presensi berbasis pengenalan wajah yang diintegrasikan dengan aplikasi pengelolaan SDM. Sistem yang dibangun merupakan sistem deteksi dan pengenalan menggunakan *Support Vector Machine* yang di kombinasikan dengan metode *Histogram of oriented gradient*. Hasil pengujian sistem presensi menunjukkan hasil recall sebesar 77,78%, nilai *spesifitas* 32,22%, akurasi sistem 72,78%, dan kepresisian sistem mencapai 70,71%.

Kata kunci: Presensi, Pengenalan Wajah, *Support Vector Machine*, *Histogram of Oriented Gradients*

I. PENDAHULUAN

DALAM pengelolaan Sumber Daya Manusia (SDM) di sebuah instansi atau perusahaan, pencatatan kehadiran karyawan merupakan salah satu faktor penting. Hal ini dikarenakan sistem pencatatan kehadiran terhubung dengan sistem pengelolaan SDM yang biasanya mencakup sistem penggajian, *deadline* pekerjaan, hak cuti karyawan, dan sebagainya. Untuk itu banyak perusahaan memilih untuk mengembangkan ataupun menggunakan perangkat berteknologi tinggi untuk diterapkan pada sistem pengelolaan SDMnya.

Sistem pengelolaan SDM pada umumnya tidak terlepas pada sistem identifikasi SDM. Sistem identifikasi dan pengenalan ini biasanya diterapkan pada sistem forensik, dan sistem keamanan, seperti pengaksesan ruangan, perangkat/data/dokumen, dan laman website melalui berbagai metode akses. Metode akses ini biasanya membutuhkan media sebagai identitas pengguna seperti sidik jari [1][2], telapak tangan [3], iris mata [4] dan pengenalan wajah [5]. Berbagai metode juga telah dikembangkan di beberapa penelitian seperti metode *image quality enhancement*, *Deep Convolution Neural Networks* (DCNN) [4], *Wasserstein Convolutional Neural*

Network [5], *eigenface* [6] dan metode lain yang bersesuaian dengan jenis media identitasnya.

Pada penelitian Wardoyo [6], dibangun sistem pengenalan wajah menggunakan algoritma *eigenface* melalui aplikasi komputer. Proses pengenalan wajah diawali dengan pengambilan gambar menggunakan *webcam*, yang kemudian informasi mentah dari *pixel* citra direpresentasikan dalam metode *Principal Component Analysis* (PCA). Hasil yang diperoleh menunjukkan *sensivitas* 100%, *spesifitas* 55,55 %, dan akurasi 69,33 %. Namun, penelitian ini belum mengintegrasikan sistem dengan *website*, sehingga manajemen harus menuju lokasi presensi untuk mengambil data.

Oleh karena itu, penulis mengusulkan untuk mengembangkan sebuah sistem presensi karyawan suatu perusahaan melalui sistem pengenalan wajah menggunakan metode *Support Vector Machine*. Harapannya dengan adanya penelitian ini akan terbangun sistem presensi karyawan yang terintegrasi dengan sistem pengelolaan SDM perusahaan, meningkatnya kesadaran karyawan dalam melakukan presensi dan terciptanya keterbukaan informasi pengelolaan SDM antara karyawan dengan manajemen perusahaan.

II. TINJAUAN PUSTAKA

Pada pengenalan citra wajah secara umum ada dua jenis, yaitu sistem *image based* dan *feature based*. Pada kedua sistem ini memiliki sistem pengolahan data yang berbeda, pertama *image based* yaitu informasi mentah dari *pixel* citra digunakan untuk merepresentasikan citra dalam metode tertentu, misalnya transformasi *wavelet*, menggunakan *principal component analysis* (PCA) untuk mengklasifikasi identitas citra. Sistem *feature based* digunakan fitur yang di ekstraksi dari komponen citra wajah seperti bagian mata, hidung, mulut. Selanjutnya, fitur-fitur tersebut dihubungkan dan dimodelkan secara geometris [7]. Cara kerja dari pengenalan wajah dengan menyesuaikan *detail – detail* wajah pengguna dengan citra yang ditangkapnya melalui sebuah kamera pendeteksi. Kemudian, citra wajah yang ditangkap dibandingkan dengan *database* citra wajah yang telah disimpan dan dianalisis seberapa tinggi tingkat kecocokannya. Metode yang digunakan dalam

penelitian ini adalah *Histogram Of Oriented Gradient* dan *Support Vector Machine*.

A. Histogram of Oriented Gradient

Metode *Histogram of Oriented Gradients* (HOG) yaitu mengubah citra RGB (*Red, Green, Blue*) menjadi citra skala keabu-abuan, kemudian dilanjutkan dengan menghitung nilai gradien setiap *pixel*. Setelah nilai gradien didapatkan, proses selanjutnya dengan cara penentuan jumlah bin orientasi yang akan digunakan dalam pembuatan histogram atau *spatial orientation binning*. Pada proses *gradient compute*, gambar *training* terbagi menjadi beberapa sel yang akan dikumpulkan menjadi *block* (ukuran yang lebih besar). Sedangkan pada proses normalisasi *block* sendiri digunakan perhitungan dari geometri R-HOG. Hal ini dilakukan karena terdapat *block* yang saling tumpang tindih. Proses ini berbeda dengan proses pembuatan histogram citra yang menggunakan nilai-nilai intensitas *pixel* dari suatu citra atau bagian tertentu dari citra [8]. Hasil dari algoritma HOG menjadi *input* dalam algoritma klasifikasi citra lain seperti *Support Vector Machine* (SVM). Tujuannya adalah untuk memperoleh hasil yang baik.

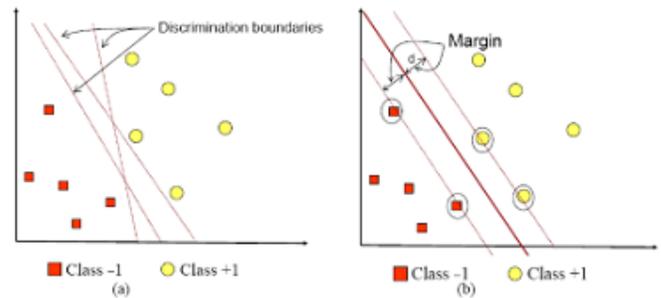


Gambar 1. Pemrosesan data menggunakan algoritma HOG

Pada Gambar 1 menjelaskan algoritma *HOG* dalam memproses sebuah gambar oleh Dalal dan Triggs. Navneet Dalal dan Bill Triggs dalam jurnalnya melakukan penelitian tentang *HOG* untuk deteksi manusia. Dalam percobaannya pada proses pertama normalisasi *gamma & colour* mengevaluasi beberapa representasi *input pixel* antara lain *grayscale, RGB, dan LAB colour space* opsional dengan persamaan *gamma*. Hasilnya yang paling baik menggunakan RGB dengan memberikan performa pada tingkat FPPW (dengan 1% pada 10^{-4} FPPW) (FPPW = *False Positives Per Window*), merupakan ukuran yang digunakan untuk mendeteksi secara umum terkait dengan klasifikasi dan deteksi sebuah objek). Pada proses *Compute gradient* adalah dengan menghitung nilai gradien horizontal dan vertikal menggunakan *Sobel Edge Detection*.

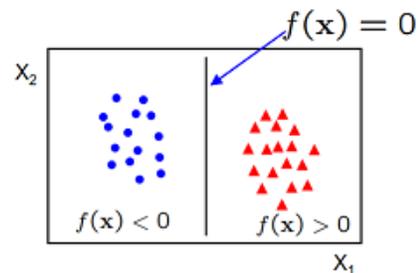
B. Support Vector Machine

Vapnik (1995) untuk pertama kalinya mengenalkan metode *Support Vector Machine* (SVM) di COLT-92. *Support vector machines* (SVM) adalah seperangkat metode pembelajaran terarah (*supervised learning*) yang digunakan untuk melakukan proses klasifikasi dan regresi [9]. Salah satu yang membedakan metode klasifikasi SVM dengan metode lain adalah SVM untuk memperoleh ukuran margin yang maksimal adalah dengan menemukan *hyperplane* terbaik. Margin merupakan jarak antara *hyperplane* dengan titik terdekat dari masing-masing kelas yang ada [8].



Gambar 2. SVM Hyperlane [8]

Gambar 2 *SVM Hyperlane* menunjukkan data beberapa *pattern* yang merupakan anggota dari dua buah *class +1* dan *-1*. Untuk *pattern* yang warna merah (kotak) menyimbolkan *pattern* yang tergabung pada *class -1*, sedangkan *symbol* warna kuning (lingkaran) menyimbolkan *pattern* pada *class +1*. Pada gambar 2 juga menunjukkan berbagai alternatif garis pemisah (*discrimination boundaries*). Usaha dalam menemukan garis (*hyperplane*) yang memisahkan kedua *class* tersebut disebut sebagai masalah klasifikasi. Dengan mengukur *margin hyperplane* dapat menemukan *hyperplane* pemisah terbaik antara kedua *class* tersebut dan mencari titik maksimalnya. *Pattern* yang paling dekat dengan margin ini disebut sebagai *support vector*. *Hyperlane* terbaik ditunjukkan oleh Gambar 3 *SVM Hyperlane* berupa garis solid yang menunjukkan *hyperplane* yang terbaik, yaitu garis yang terletak tepat pada di antara kedua *class*, sedangkan untuk *support vector* sendiri adalah titik merah dan kuning yang ada di dalam lingkaran hitam tersebut. Dan inti dari proses pembelajaran pada *Support Vector Machine* (SVM) adalah usaha untuk mencari lokasi *hyperplane*. Pada proses *learning*-nya sendiri SVM mendapat inputan dari *Feature HOG* [8].



Gambar 3. Bobot Positif dan Negatif pada *Linear Classifier* [10]

Pada Gambar 3 menjelaskan tentang klasifikasi SVM Linier untuk memisahkan *vector* dalam dimensi 2. Solusi *Soft margin* digunakan untuk mengatasi masalah dalam mencari *w* terbaik. Pada pelatihan data dan *feature SVM* diambil dengan mengambil sampel positif dan *negative* dari *HOG Descriptor*. Sampel positif artinya terdapat objek seperti yang di ditampilkan sedangkan *negative* artinya tidak terdapat objek.

III. METODE

A. Deskripsi Sistem

Pada pembuatan aplikasi presensi dibutuhkan perancangan perangkat lunak dan perangkat keras. Pada perancangan perangkat lunak terdiri dari 3 bagian yaitu aplikasi, *database* dan *website*. Sedangkan dari perangkat keras sistem akan dibuat dengan meletakkan kamera di meja dekat pintu masuk karyawan. Kamera akan menangkap wajah pegawai yang melakukan presensi dengan memosisikan wajah di depan kamera yang telah disediakan. Selanjutnya data hasil kamera diproses dengan pengenalan wajah untuk mendapat data unik id karyawan. Setelah itu data unik *id* dan waktu kedatangan atau pulang dikirim ke *database* server dengan simulasi *localhost* di komputer. Data pada server kemudian diolah untuk menghasilkan informasi mengenai presensi karyawan.

B. Pengumpulan Data

Tahap pertama dari penelitian ini adalah pengumpulan data. Tahap pengumpulan data adalah alat bantu yang digunakan oleh peneliti untuk memperoleh informasi yang dipakai pada proses penelitian. Pada tahap ini juga menyiapkan perangkat *webcam* yang digunakan untuk mengambil data citra. Untuk data karyawan telah tersedia pada *database server*. Penulis meminta ijin untuk mengakses *database*. Dengan menggunakan *query* (bahasa pemrograman *Sql* untuk melakukan permintaan data) untuk mendapatkan data karyawan. Kemudian data foto di kumpulkan dalam satu folder yang akan digunakan untuk proses *learning*. Foto yang digunakan adalah foto *Close up* yang diambil menggunakan *webcam* yang sama dengan *webcam* yang digunakan untuk pengujian alat.



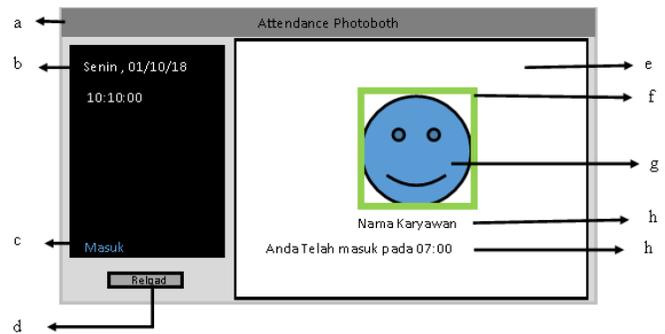
Gambar 4. Blok Diagram Sistem Presensi

C. Perancangan Software

Perancangan software pada penelitian ini adalah proses pengembangan sistem presensi karyawan yang meliputi perancangan antarmuka dan proses pengenalan wajah. Pada perancangan software tahapan pertama perancangan *block diagram* seperti yang ditunjukkan Gambar 4 dibawah ini.

Gambar 5 menjelaskan sistem yang digunakan untuk membangun aplikasi presensi. Aplikasi pengenalan wajah adalah alat yang digunakan untuk proses deteksi dan pengenalan wajah dibangun menggunakan *framework python* kemudian aplikasi presensi terhubung dengan *database* melalui koneksi *python* ke *local database postgres*. *Database* adalah aplikasi yang digunakan untuk menampung data pengujian. Aplikasi mengirim data ke presensi ke *database*, apabila data sudah ada maka *database* akan memberikan respon kepada aplikasi pengenalan wajah. *Database* digunakan adalah *postgresql*, karena hanya *postgresql* yang dapat terhubung langsung dengan *framework Web* yang digunakan. *Web*

Application adalah aplikasi yang digunakan penulis untuk menampilkan data hasil presensi dan alat pengelolaan SDM atau biasa disebut *system ERP*. Untuk *Web application* nya sendiri dibangun menggunakan *framework* Odoo.



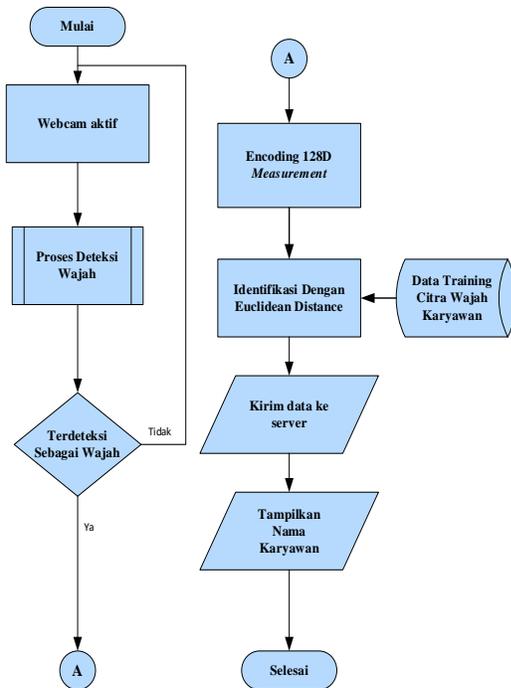
Gambar 5. Design Interface Aplikasi Presensi

Gambar 5 *Design Interface* Aplikasi Presensi menjelaskan desain antarmuka pada aplikasi presensi berikut poin-poin penjelasannya:

- 1) *Attendance Photobooth* : merupakan judul *form* aplikasi yang dibangun menggunakan *python*.
- 2) Tanggal dan jam : pada bagian kiri atas menunjukkan tanggal dan jam saat ini.
- 3) Masuk : merupakan status presensi untuk catat masuk atau pulang (*Default* sebelum pukul 12:00 otomatis sebagai presensi masuk, setelah pukul 12:00 sebagai Keluar).
- 4) Tombol *Reload* : digunakan untuk memperbarui *database* wajah dan *training* ulang *database* wajah yang ada pada *directory* *database*.
- 5) *Frame* aplikasi : menampilkan gambar *realtime* dari perangkat *webcam* Logitech C270.
- 6) Kotak hijau : untuk menandai posisi wajah dalam gambar, bisa lebih dari satu.
- 7) Area Wajah : wajah karyawan yang melakukan presensi.
- 8) Nama karyawan : Nama dari karyawan yang melakukan presensi ditampilkan di bawah kotak wajah.
- 9) Info waktu presensi : menampilkan info kapan terakhir masuk atau pulang apabila telah dicatat.

Aplikasi ini dibuat menggunakan bahasa *Python*, untuk penulisan program penulis menggunakan *Sublime-Text 3*. Pada tampilan *interface* dirancang menggunakan *Python Tkinter*. *Tkinter* merupakan *module* pendukung dalam membuat *GUI* (antarmuka) yang digunakan oleh Bahasa pemrograman *python*. *Library* *Open CV* digunakan untuk menampilkan hasil pemrosesan ke dalam *canvas* di antarmuka aplikasi. Proses pengenalan wajah dimulai dari pengambilan gambar melalui *webcam* kemudian dilanjutkan dengan penentuan posisi wajah menggunakan *HOG*, lalu dengan *facial landmark* untuk memosisikan wajah simetris selanjutnya dengan *SVM* untuk memprediksi karyawan yang melakukan presensi. Untuk sistem kerja program ini secara ringkas ditunjukkan oleh Gambar 6.

Pada Gambar 6 menjelaskan tentang proses pengenalan wajah, proses pertama *webcam* aktif untuk menangkap citra. Tahap selanjutnya proses deteksi wajah, untuk mengenali wajah dalam gambar adalah dengan menemukan bagian gambar yang paling terlihat mirip dengan pola wajah menggunakan *Feature HOG* dan *SVM*. Apabila dikenali sebagai wajah maka dilanjutkan ke proses berikutnya. Bila tidak maka akan kembali ke tahapan awal. Tahapan selanjutnya *Encoding 128D Measurement* pada proses ini *image* di *encoding* menjadi bentuk vektor yang berisi 128D[11]. Dengan menggunakan identifikasi *Euclidean distance image* hasil *encoding* di bandingkan dengan *image* pada *database* yang telah di-*training* dan disimpan dalam file format *numpy array*. Bila menemukan nilai terdekat atau maksimal level perbedaan 0,5 maka data dikirim ke *database* server data diolah agar menghasilkan informasi yang bisa digunakan sebagai laporan kehadiran.



Gambar 6. Diagram Alir Proses Pengenalan Wajah

1) Proses Deteksi Wajah dengan HOG

Proses deteksi wajah menggunakan HOG dijelaskan oleh diagram alir pada Gambar 7. Gambar 7 menjelaskan tahapan dari Akuisisi Citra sampai terdeteksi sebagai Wajah. Untuk Proses ini menggunakan *feature descriptor Histogram Of Oriented Gradient dan SVM*. Tahapannya sebagai berikut:

- a) Tahap pertama yaitu akuisisi citra, citra diperoleh dari memicu *webcam* menyala.
- b) Setelah akuisisi citra tahap selanjutnya yaitu *Resize Frame*.

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} s_x & 0 \\ 0 & s_y \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} \quad (1)$$

Resize Frame yaitu memperkecil ukuran gambar dengan fungsi scaling terhadap sumbu x dan y dengan faktor s_x dan s_y [12]. Disini gambar dengan resolusi 1280 x 720 *pixel* di *resize* 0,25 menjadi 320 x 180 *pixel*. Tujuan *resize* gambar ini agar mempercepat proses pengenalan wajah.

- c) Kemudian normalisasi warna, membuat warna menjadi *grayscale*. Transformasi dalam ruang RGB konversi dari warna RGB ke skala abu-abu.

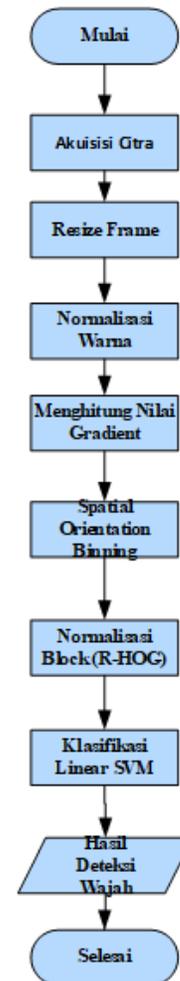
$$\text{RGB[A] to Gray} = L = 0,299 \cdot R + 0,587 \cdot G + 0,114 \cdot B \quad (2)$$

- d) Setelah itu menghitung nilai gradient pada tiap-tiap *pixel*. Ma

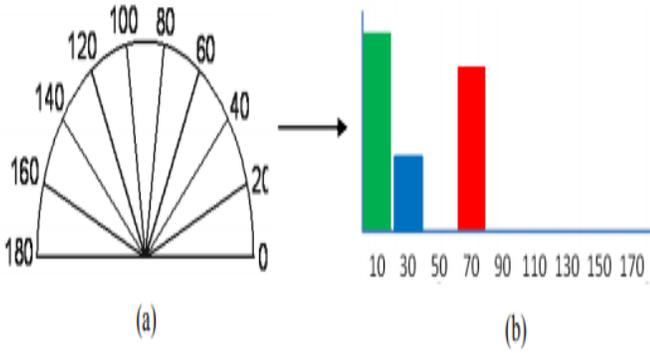
$$\text{Magnitude (besar gradient)} = g = \sqrt{g_x^2 + g_y^2} \quad (3)$$

$$\text{Orientasi gradient (dalam sudut)} = \theta = \arctan \frac{g_y}{g_x} \quad (4)$$

- e) Dilanjutkan dengan *spatial orientation binning* atau menentukan bin orientasi. Gambar dikelompokkan dalam ruang yang lebih kecil disebut sel. Angka tetap dalam bins didapat dari sudut *gradient* yang dibagi oleh histogram orientasi. Besar *gradient* dan orientasi digunakan untuk *vote* ke dalam histogram orientasi. Penulis membuat histogram gradien dalam 8x8 sel. Histogram berisi 9 *bins* yang sesuai dengan sudut 0, 20, 40 ... 160.

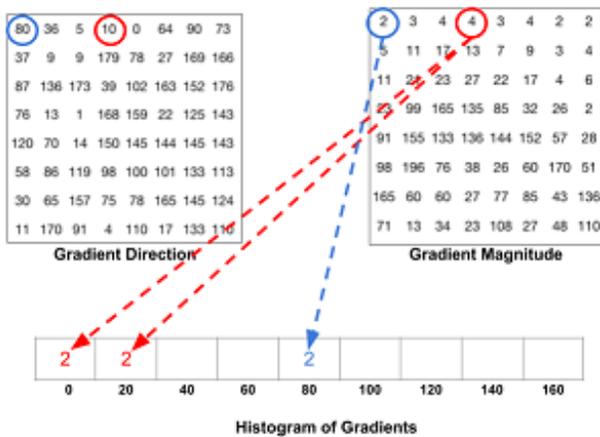


Gambar 7. Diagram Alir Proses Deteksi Wajah



Gambar 8. (a) Sudut Gradient, (b) Grafik Histogram pada set [8]

Gambar 8 masing-masing menjelaskan (a) adalah sudut *gradient*, nilai *magnitude* dalam suatu citra, sedangkan (b) adalah grafik histogram dalam setiap sel pada sebuah gambar.



Gambar 9. Contoh perhitungan menentukan bin orientasi [8]

Gambar 9 menjelaskan tentang proses perhitungan dalam menentukan bin orientasi, data *array* sebelah kiri adalah arah *gradient (direction)* sedangkan data sebelah kanan merupakan *gradient magnitude*. Posisi bin dipilih berdasarkan arah sedangkan nilai bin dipilih berdasarkan *magnitude*. Warna biru dengan *gradient direction* 80 dan *gradient magnitude* 2 maka berada pada posisi bin ke 5 dengan nilai 2. Sedangkan yang dilingkari merah nilai *gradient direction* 10, dan nilai *gradient magnitude* 4, karena 10 berada di antara 0 dan 20 maka *gradient* mengisi 2 bin.

f) Tahap selanjutnya normalisasi *block*. Dalam penelitian ini geometri yang digunakan adalah R-HOG. R-HOG umumnya kotak persegi, diwakili oleh tiga parameter: jumlah sel per blok, jumlah *pixel* per sel, dan jumlah *channel* per histogram sel. Dalal dan Triggs, dalam jurnal *HOG* untuk deteksi manusia, parameter optimal ditemukan menjadi empat sel 8×8 *pixel* per blok (16×16 *pixel* per blok) dengan 9 *channel* histogram. Menghitung normalisasi blok yaitu dengan cara menormalisasi respons kontras secara keseluruhan dan mengambil kelompok sel. Hal ini dilakukan dengan cara mengakumulasikan ukuran

histogram dari grup tiap-tiap sel yang disebut blok. Untuk menormalisasi setiap sel dalam blok menggunakan hasil proses, berikut adalah penghitungan histogram pada blok:

$$L2 - norm : v \rightarrow v / \sqrt{\|v\|_2^2 + \epsilon^2} \quad (5)$$

g) Tahapan terakhir adalah klasifikasi linear svm dan proses deteksi wajah. Dengan menggunakan *soft constrain* ($C=0,01$) untuk pelatihan SVM. Dataset yang digunakan untuk pelatihan, terdiri dari 2825 gambar yang diperoleh dari *dataset LFW* yang disediakan oleh Davis King, penulis Dlib. Kemudian menggunakan klasifikasi *sliding window* untuk deteksi posisi wajah dalam sebuah gambar dengan mencari bobot positif. Dengan *training data* (x_i, y_i) for $i = 1 \dots N$, dengan $x_i \in \mathbb{R}^d$ dan $y_i \in \{-1, 1\}$, pembelajaran klasifikasi $f(x)$.

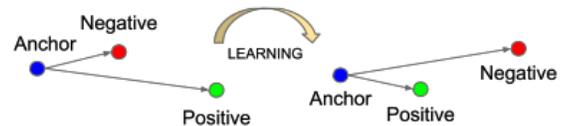
$$f(x_i) = \begin{cases} \geq 0 & y_i = +1 \\ < 0 & y_i = -1 \end{cases} \quad (6)$$

$$f(x) = w^T x + b \quad (7)$$

Dari rumus (6) dan gambar 4 Bobot Positif dan Negatif pada *Linear Classifier* maka proses pembelajaran *svm* dirumuskan dalam persamaan (7).

1) Proses *Encoding 128 Dimension measurement*

Pada proses ini bertujuan untuk menghasilkan 128 pengukuran pada setiap citra wajah yang dideteksi. Gagasan untuk mengurangi data mentah yang rumit seperti data citra ke dalam daftar angka yang dihasilkan *machine learning* telah banyak dilakukan penelitian. Pendekatan yang penulis gunakan dalam penelitian ini yang ditemukan oleh Schroff (2015)[12].



Gambar 10. Triplet Loss minimizes the distance

Gambar 10. *Triplet loss minimizes the distance* menjelaskan tentang proses learning bertujuan untuk mengurangi jarak antara anchor dan positif, dan maksimal jarak antara anchor dan negative. *Embedding* direpresentasikan dengan $f(x) \in \mathbb{R}^d$. Embed citra x kedalam d -dimensi *Euclidean space*. Facenet (Schroff) menggunakan 128 dimensi dan membuat model yang memetakan setiap wajah manusia secara umum. Ketika diberikan citra input model akan memberikan 128 *byte* data vektor numerik. Titik-titik *embedding* ini mudah dibandingkan dengan mengukur jarak *Euclidean*.

2) Proses Identifikasi dengan *Euclidean Distance*

Untuk identifikasi hasil vector citra test dengan vector citra database dengan mencari jarak terdekat antara vector citra test dengan vector citra database. Jarak (*distance*) ini digunakan untuk menentukan tingkat kesamaan dalam dua fitur vector [13]. Penulis menggunakan *Euclidean distance* dalam menghitung kesamaan 2 vektor. Akar kuadrat dari perbedaan 2 vektor dihitung dengan *Euclidean distance*. Pada citra wajah orang yang sama *Euclidean distance*

cenderung kecil. Untuk identifikasi diambil dari nilai *Euclidean distance* terendah dengan batas toleransi 0.5.

$$d_y = \sqrt{\sum_{k=1}^n (x_{ik} + x_{jk})^2} \quad (8)$$

Contoh pada perhitungan menunjukkan 2 citra wajah:

$$A = [0, 3, 4, 5]$$

$$B = [7, 6, 3, -1]$$

$$C = [1, 3, 4, 5]$$

Dengan C sebagai vektor test dan vektor A, vektor B merupakan database.

$$d_{AB} = \sqrt{(0-7)^2 + (3-6)^2 + (4-3)^2 + (5-(-1))^2}$$

$$d_{AB} = \sqrt{49+9+1+36} = 9,747$$

$$d_{AC} = \sqrt{(0-1)^2 + (3-3)^2 + (4-4)^2 + (5-5)^2}$$

$$d_{AC} = \sqrt{1+0+0+0} = \sqrt{1}$$

Dari hasil perhitungan diperoleh jarak citra paling kecil yaitu $\sqrt{1}$, maka hasil identifikasi menyimpulkan Vector A lebih mirip terhadap Vector C namun dari nilai batas toleransi di atas 0,5 maka *image* dianggap bukan bagian dari karyawan

D. Pembuatan Program

Langkah selanjutnya yaitu pembuatan program. Pembuatan program adalah proses menerjemahkan dari diagram alir ke dalam suatu bahasa pemrograman. Bahasa pemrograman python dipilih dalam penelitian ini karena bersifat *open source* dan dapat dijalankan pada sistem operasi Ubuntu. Sedangkan untuk penulisan program penulis menggunakan *software* Sublime-Text, karena memiliki fitur *indent tab* yang dibutuhkan oleh python.

E. Instalasi

Instalasi adalah alat perlengkapan yang di pasang pada posisi tertentu dan siap untuk digunakan.



Gambar 11. Instalasi Perangkat

Gambar 11 Instalasi Perangkat menjelaskan tentang perancangan perangkat yang akan dibuat. Perangkat untuk pengembangan sistem ini menggunakan kamera *webcam* Logitech C270 dan Komputer. Kamera dihubungkan ke komputer melalui kabel USB, komputer digunakan untuk memproses data hasil dari kamera. Selanjutnya data yang telah diproses berupa data tanggal dan id karyawan dikirim ke *server* dan *server* memberikan respons berupa peringatan apabila karyawan tersebut telah melakukan presensi sebelumnya.

F. Metode Pengambilan Data

Pada pengambilan data posisi kamera menghadap ke arah objek. Di penelitian ini, dilakukan pengambilan data dalam beberapa tahap yang meliputi:

1) Pengujian kemampuan deteksi berdasarkan jarak.

Pengujian ini bertujuan untuk mengetahui jangkauan aplikasi mendeteksi wajah dan jarak ideal dalam melakukan presensi. Pengujian kemampuan deteksi berdasarkan jarak dilakukan mulai dari 10 cm sampai 110 cm dengan posisi kamera mengarah ke objek dan wajah posisi menghadap ke kamera.

2) Pengujian kemampuan deteksi berdasarkan posisi kemiringan wajah.

Pengujian kemampuan deteksi berdasarkan posisi wajah bertujuan untuk mendapatkan posisi wajah yang ideal dalam melakukan proses presensi. Posisi wajah yang diuji adalah posisi *Pitch*, *Yaw*, dan *Roll* dengan *range* antara 50 derajat sebelum sampai 50 derajat sesudah. Dengan posisi nol derajat adalah posisi di depan kamera dengan posisi wajah menghadap kamera seperti ditunjukkan gambar 15 Posisi wajah. Pengujian dilakukan dengan mengarahkan kamera ke objek dan dengan menggerakkan posisi kepala.

3) Pengujian tingkat keberhasilan sistem presensi karyawan berbasis pengenalan wajah dengan metode support vector machine dengan cara mengintegrasikan hasil dari proses pengenalan wajah dengan *database system* pengelolaan SDM milik perusahaan. Sedangkan untuk tingkat keberhasilan pengenalan wajah parameter yang digunakan antara lain:

a) *Recall*

Recall (sensivitas) atau *True positif rate* adalah keberhasilan sistem dalam menemukan kembali informasi yang diperoleh dan dihitung dengan rumus.

$$Recall = \frac{TP}{TP + FN} \times 100\% \quad (9)$$

b) Spesifitas

Spesifitas atau *False positif rate* atau kekhususan merupakan nilai yang menunjukkan kesalahan pada saat proses identifikasi.

$$Spesifitas = \frac{FP}{TP + FN} \times 100\% \quad (10)$$

c) Akurasi

Akurasi adalah tingkat kedekatan antara nilai aktual dengan nilai prediksi. Dan keakurasian sistem pengenalan dilakukan dengan membagi jumlah klasifikasi benar dengan total pengujian.

$$Akurasi = \frac{TP + TN}{TP + FP + TN + FN} \times 100\% \quad (11)$$

d) Presisi

Presisi adalah tingkat ketepatan antara nilai yang diprediksi benar dengan tepat dengan nilai total prediksi benar dan dihitung dengan rumus [14].

$$Presisi = \frac{TP}{TP + FP} \times 100\% \quad (12)$$

Keterangan:

- TP (*true positive*) = Citra wajah yang dapat di identifikasi dan benar datanya ada pada *database* (pengujian benar).
- FP (*false positif*) = Citra wajah yang di identifikasi benar namun pada *database* merupakan data salah (salah mengidentifikasi).
- TN (*true negative*) = Citra wajah seharusnya di identifikasi salah dan diidentifikasi salah (pengujian benar).

- d. FN (*false negative*) = Citra wajah yang seharusnya diidentifikasi benar namun salah dalam mengidentifikasi.

IV. HASIL DAN PEMBAHASAN

A. Pengujian Kemampuan Berdasarkan Jarak

Untuk mengetahui jarak ideal dalam melakukan presensi, maka dilakukan pengujian jarak jangkauan terhadap kemampuan mengenali wajah dengan menggunakan Aplikasi Presensi. Pengukuran dilakukan dari jarak 10 cm karena bila dibawah 10 cm tidak dapat dideteksi sebagai wajah, pengujian dilakukan dengan objek yang sama. Hasil pengujian sebagian di tampilkan pada Tabel I.

TABEL I
PENGUJIAN DETEKSI JARAK

Hari ke-	Suhu (°C)	Kelembaban (%)
1	10	Tidak terdeteksi
2	20	Tidak terdeteksi
3	30	Terdeteksi
4	40	Terdeteksi
5	50	Terdeteksi
6	60	Terdeteksi
7	70	Terdeteksi
8	80	Terdeteksi
9	90	Terdeteksi
10	100	Tidak Terdeteksi
11	110	Tidak Terdeteksi

Dari data tabel I dapat dilihat jarak ideal dalam melakukan presensi yaitu antara 30 cm – 90 cm. Jarak ini dijadikan referensi untuk melakukan proses presensi

B. Pengujian Kemampuan Deteksi Berdasarkan Kemiringan

Pengujian kemampuan deteksi berdasarkan kemiringan bertujuan untuk mendapatkan posisi wajah yang ideal dalam melakukan proses presensi. Ada tiga jenis posisi kemiringan yang diambil datanya antara lain posisi Pitch, Roll, dan Yaw.

TABEL II
HASIL PENGUJIAN TERHADAP POSISI PITCH, ROLL, YAW

No	Sudut (°)	Pitch	Roll	Yaw
1	-50	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
2	-40	Tidak Terdeteksi	Terdeteksi	Tidak Terdeteksi
3	-30	Terdeteksi	Terdeteksi	Terdeteksi
4	-20	Terdeteksi	Terdeteksi	Terdeteksi
5	-10	Terdeteksi	Terdeteksi	Terdeteksi
6	0	Terdeteksi	Terdeteksi	Terdeteksi
7	10	Terdeteksi	Terdeteksi	Terdeteksi
8	20	Terdeteksi	Terdeteksi	Terdeteksi
9	30	Terdeteksi	Terdeteksi	Terdeteksi
10	40	Terdeteksi	Terdeteksi	Tidak Terdeteksi
11	50	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi

Dari data hasil pengujian kemiringan pada Tabel II diperoleh bahwa sistem mampu mendeteksi dan mengenali wajah dengan posisi terhadap Roll antara range -40 derajat atau 40 derajat ke arah kiri sampai 40 derajat ke arah kanan. Posisi

Yaw atau terhadap sumbu tegak lurus dengan gravitasi antara -30 derajat sampai 30 derajat. Sedangkan pada posisi Pitch alat mampu mendeteksi dari -30 derajat atau posisi menunduk sampai 40 derajat posisi menengadahkan.

TABLE III
STANDAR RANGE EFFECTIVITIES SISTEM

Sudut Pengujian	Kriteria
0° s/d 30° dan 0° s/d -30°	Efektif
30° s/d 50° dan -30° s/d -50°	Tidak Efektif
Di atas 50° dan Di bawah -50°	Sangat Tidak Efektif

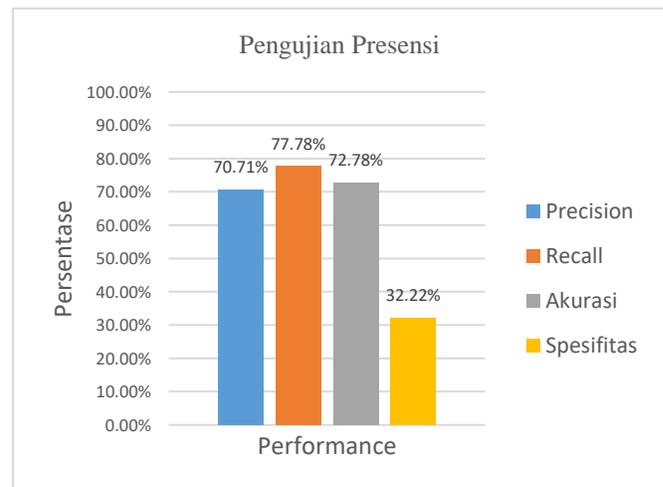
Dari tabel III diperoleh standar *range* efektivitas sistem antara lain : efektif apabila diuji pada sudut 0° s/d 30° dan 0° s/d -30° , tidak efektif apabila diuji pada *range* 30° s/d 50° dan -30° s/d -50°, sangat tidak efektif apabila diuji pada *range* di atas 50° atau di bawah 50°.

C. Pengujian Presensi

Pengujian Presensi bertujuan untuk menguji kestabilan sistem saat digunakan untuk proses presensi karyawan dengan melakukan simulasi presensi. Data diambil dengan data karyawan yang ada di basis data dan data bukan karyawan yang tidak ada di basis data.

Penjelasan istilah yang digunakan sebagai berikut:

- 1) TP atau *True Positif* adalah Prediksi benar pada aktual Benar.
- 2) FP atau *False Positif* adalah Prediksi benar pada aktual Salah.
- 3) FN atau *False Positif* adalah Prediksi salah pada aktual Benar.
- 4) TN atau *True Negatif* adalah Prediksi salah pada aktual salah



Gambar 12. Grafik Data Pengujian Presensi

Gambar 12 menampilkan data hasil pengujian presensi. Pengujian dilakukan dengan karyawan dan *visitor*, di mana untuk karyawan datanya telah dimasukkan dalam *database* sedangkan *visitor* datanya tidak. Pengujian diperoleh *Precision* 70,71%, *Recall* 77,78%, *Akurasi* 71,78%, dan *Spesifitas* 32.22 %. Pada pengujian presensi karyawan *False Negative* atau salah deteksi pada data yang telah di register sebanyak 20 dari 90 data. Pada pengujian presensi *Visitor* diperoleh *False positif* atau salah deteksi pada data *False* sebesar 29 dari total 90 pengujian.

TABLE IV
PENGUJIAN SIMULASI PRESENSI KARYAWAN

No	Nama	Pengujian Ke									
		1	2	3	4	5	6	7	8	9	10
1	Karyawan 1	TP	FN	TP	TP	TP	FN	TP	TP	FN	TP
2	Karyawan 2	FN	TP	TP	TP	TP	TP	FN	TP	TP	FN
3	Karyawan 3	TP	TP	TP	FN	FN	TP	TP	TP	FN	TP
4	Karyawan 4	TP	TP	TP	TP	TP	TP	TP	FN	TP	TP
5	Karyawan 5	TP	TP	TP	TP	FN	TP	TP	TP	FN	FN
6	Karyawan 6	TP	TP	TP	TP	TP	TP	TP	TP	TP	TP
7	Karyawan 7	FN	TP	TP	TP	TP	FN	TP	TP	TP	FN
8	Karyawan 8	TP	TP	TP	TP	FN	TP	TP	TP	TP	TP
9	Karyawan 9	FN	TP	TP	TP	FN	TP	TP	TP	FN	TP

Dari hasil pengujian simulasi presensi karyawan yang semuanya telah diregister diperoleh identifikasi benar atau nilai TP = 70 dan identifikasi salah atau FN = 20.

TABLE V
PENGUJIAN SIMULASI PRESENSI BUKAN KARYAWAN

No	Nama	Pengujian Ke									
		1	2	3	4	5	6	7	8	9	10
1	Visitor 1	FP	FP	FP	FP	TN	FP	FP	FP	FP	TN
2	Visitor 2	FP	FP	TN							
3	Visitor 3	TN	TN	TN	TN	TN	TN	TN	TN	TN	TN
4	Visitor 4	TN	TN	FP	FP	FP	TN	TN	TN	FP	FP
5	Visitor 5	TN	TN	FP							
6	Visitor 6	TN	TN	TN	TN	TN	TN	TN	TN	TN	TN
7	Visitor 7	TN	TN	TN	TN	FP	TN	TN	TN	TN	TN
8	Visitor 8	TN	TN	TN	TN	TN	TN	TN	TN	TN	TN
9	Visitor 9	TN	TN	TN	FP	FP	TN	FP	FP	FP	TN

Dari hasil pengujian simulasi presensi bukan karyawan atau tamu yang datanya tidak ada di database diperoleh identifikasi benar atau nilai TN = 61 dan identifikasi salah atau FP = 29.

$$\text{Recall} = \frac{70}{70+20} \times 100\% = 77,78\%$$

$$\text{Spesifitas} = \frac{29}{70+20} \times 100\% = 32,22\%$$

$$\text{Akurasi} = \frac{70+61}{70+61+29+20} \times 100\% = 77,78\%$$

$$\text{Presisi} = \frac{70}{70+29} \times 100\% = 70,71\%$$

V. KESIMPULAN

Berdasarkan analisa dari pengujian pengenalan wajah dalam aplikasi presensi maka dapat disimpulkan bahwa Hasil pengujian pengambilan gambar berdasarkan jarak menunjukkan hasil yang paling baik di antara 30 cm – 90 cm. Berdasarkan pengujian kemampuan berdasarkan kemiringan didapat bahwa sistem mampu mendeteksi dan mengenali wajah dengan posisi terhadap Roll antara range -40 derajat atau 40 derajat ke arah kiri sampai 40 derajat ke arah kanan. Posisi Yaw atau terhadap sumbu tegak lurus dengan gravitasi antara -30 derajat sampai 30 derajat. Sedangkan pada posisi Pitch alat mampu mendeteksi dari -30 derajat atau posisi menunduk sampai 40 derajat posisi menengadiah. Pada pengujian presensi didapat sistem dapat menemukan kembali informasi dengan baik ditunjukkan dengan hasil recall 77,78%, tingkat kesalahan saat proses identifikasi atau spesifitas 32,22%, kemudian tingkat kedekatan nilai aktual dan prediksi atau akurasi 77,78%, dan tingkat ketepatan nilai yang diprediksi benar dengan tepat atau presisi 70,71%.

REFERENSI

- [1] E. Rahmawati *et al.*, "Digital signature on file using biometric fingerprint with fingerprint sensor on smartphone," *Proceedings IES-ETA 2017 - International Electronics Symposium on Engineering Technology and Applications*, vol. 2017-December, pp. 234–238, Dec. 2017, doi: 10.1109/ELECSYM.2017.8240409.
- [2] W. Jiang, X. Wang, X. Song, Q. Liu, and X. Liu, "Tracking your browser with high-performance browser fingerprint recognition model," *China Communications*, vol. 17, no. 3, pp. 168–175, Mar. 2020, doi: 10.23919/JCC.2020.03.014.
- [3] B. P. Nguyen, W. L. Tay, and C. K. Chui, "Robust Biometric Recognition from Palm Depth Images for Gloved Hands," *IEEE Transactions on Human-Machine Systems*, vol. 45, no. 6, pp. 799–804, Dec. 2015, doi: 10.1109/THMS.2015.2453203.
- [4] A. Gangwar and A. Joshi, "DeepIrisNet: Deep iris representation with applications in iris recognition and cross-sensor iris recognition," *undefined*, vol. 2016-August, pp. 2301–2305, Aug. 2016, doi: 10.1109/ICIP.2016.7532769.
- [5] R. He, X. Wu, Z. Sun, and T. Tan, "Wasserstein CNN: Learning Invariant Features for NIR-VIS Face Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 7, pp. 1761–1773, Jul. 2019, doi: 10.1109/TPAMI.2018.2842770.
- [6] S. Wardoyo, R. Wiryadinata, and R. Sagita, "Sistem Presensi Berbasis Algoritma Eigenface Dengan Metode Principal Component Analysis," *Setrum: Sistem Kendali-Tenaga-elektronika-telekomunikasi-komputer*, vol. 3, no. 1, pp. 61–68, Mar. 2016, doi: 10.36055/SETRUM.V3I1.498.
- [7] Hanif Al Fatta, *Rekayasa Sistem Pengenalan Wajah*. Yogyakarta: Andi Offset, 2009.
- [8] Ari Kumiawan and Marzuki Syahfirin, "Aplikasi Deteksi Objek Menggunakan Histogram of Oriented Gradient Untuk Modul Sistem Cerdas Pada Robot Nao," Lampung, 2016.
- [9] A. Zisserman, "C19 Machine Learning lectures Hilary 2015, Lecture 2: The SVM classifier." 2015.
- [10] C. Cortes, V. Vapnik, and L. Saitta, "Support-vector networks," *Machine Learning 1995 20:3*, vol. 20, no. 3, pp. 273–297, Sep. 1995, doi: 10.1007/BF00994018.
- [11] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A Unified Embedding for Face Recognition and Clustering," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 07-12-June-2015, pp. 815–823, Mar. 2015, doi: 10.1109/CVPR.2015.7298682.
- [12] Emmanuel Agu, "Digital Image Processing (CS/ECE 545) Lecture 11: Geometric Operations, Comparing Images and Future Directions." Accessed: Dec. 29, 2021. [Online]. Available: <https://web.cs.wpi.edu/~emmanuel/courses/cs545/S14/slides/lecture01.pdf>
- [13] Darma Putra Westriningsih, *Pengolahan citra digital / Darma Putra*, 1st ed. Yogyakarta: Andi, 2010.
- [14] D. M. W. Powers and Ailab, "Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation," Oct. 2020, Accessed: Dec. 28, 2021. [Online]. Available: <https://arxiv.org/abs/2010.16061v1>