

Studi Kasus Deteksi Orang di Kelas dengan Fitur *Histogram of Oriented Gradients* dan *Local Binary Pattern* dengan model *Support Vector Machine*

Davin Jason Evan Raharjo

*Department of Computer Science
and Electronics*

Universitas Gadjah Mada

Yogyakarta, Indonesia

davinjasonevanraharjo@mail.ugm.ac.id

Aizar Hafizh Soejadi

*Department of Computer Science
and Electronics*

Universitas Gadjah Mada

Yogyakarta, Indonesia

Muhammad Dafa Wisnu Galih

*Department of Computer Science
and Electronics*

Universitas Gadjah Mada

Yogyakarta, Indonesia

muhammaddafawisnugalih@mail.ugm.ac.id

Ulima Muna Syarifah

*Department of Computer Science
and Electronics*

Universitas Gadjah Mada

Yogyakarta, Indonesia

ulimamunasyarifah@mail.ugm.ac.id

Bayu Dewanto

*Department of Computer Science
and Electronics*

Universitas Gadjah Mada

Yogyakarta, Indonesia

bayudewanto@mail.ugm.ac.id

Abstract—Deteksi orang dalam citra merupakan topik penelitian penting dengan aplikasi praktis seperti pengawasan keamanan dan manajemen sumber daya manusia. Dalam konteks pendidikan, deteksi orang di kelas membantu memonitor kehadiran siswa dan memantau proses pembelajaran secara real-time. Tantangan seperti variasi pose, pencahayaan, dan latar belakang yang kompleks memerlukan metode yang akurat. Penelitian ini mengusulkan kombinasi fitur *Histogram of Oriented Gradients* (HOG) dan *Local Binary Pattern* (LBP) untuk deteksi orang di kelas, menggunakan *Support Vector Machine* (SVM) sebagai klasifikator.

Index Terms—Deteksi orang, HOG, LBP, SVM, Analisis citra

I. PENDAHULUAN

Deteksi orang dalam citra menjadi salah satu topik penelitian penting dengan berbagai aplikasi praktis seperti pengawasan keamanan, analisis perilaku, dan manajemen sumber daya manusia. Di lingkungan pendidikan, khususnya, deteksi orang dalam kelas dapat membantu dalam berbagai aspek seperti memonitor kehadiran siswa, menganalisis interaksi antar siswa, dan memantau proses pembelajaran secara *real-time*. Metode yang efektif untuk mendeteksi orang dalam citra diperlukan untuk menghadapi tantangan yang ada, seperti variasi dalam pose, pencahayaan, dan latar belakang yang kompleks. Oleh karena itu, penelitian dalam bidang ini terus berkembang dengan tujuan menemukan teknik yang lebih akurat dan efisien.

Salah satu pendekatan yang menjanjikan adalah kombinasi fitur *Histogram of Oriented Gradients* (HOG) dan *Local Binary Pattern* (LBP), seperti yang diusulkan oleh Zeng dan Ma dalam [1]. Mereka mengembangkan metode deteksi kepala dan bahu yang menggunakan multilevel HOG-LBP

yang dikombinasikan dengan *Principal Component Analysis* (PCA) untuk pengurangan dimensi fitur. Teknik ini terbukti efektif dalam meningkatkan ketahanan dan akurasi deteksi di lingkungan dengan kondisi pencahayaan dan latar belakang yang beragam.

Penelitian lain yang juga memberikan kontribusi signifikan adalah yang dilakukan oleh Hassan et al. [2], yang mengembangkan sistem penghitung orang yang ditingkatkan dengan deteksi kepala-bahu di skenario kerumunan padat. Mereka menggunakan kombinasi HOG dan *Completed Local Binary Pattern* (CLBP) untuk mendeteksi area kepala dan bahu, yang bertujuan meningkatkan akurasi dalam lingkungan yang sangat padat.

Mengambil inspirasi dari kedua pendekatan ini, penelitian kami berfokus pada penggunaan HOG dan LBP untuk deteksi orang dalam kelas dengan memanfaatkan model *Support Vector Machine* (SVM) sebagai klasifikator. Model SVM dipilih karena kemampuannya dalam memisahkan data dengan margin optimal, yang sangat penting untuk menangani data citra yang kompleks dan beragam. Penelitian ini bertujuan untuk mengevaluasi efektivitas kombinasi HOG dan LBP dalam mendeteksi orang di kelas, serta mengatasi berbagai tantangan yang ada dalam permasalahan ini.

II. STUDI LITERATUR

Beberapa penelitian terdahulu telah mencoba metode-metode yang cukup serupa untuk melakukan deteksi wajah. Pada [1] dijelaskan bahwa penggabungan fitur HOG dan LBP memungkinkan untuk menangkap informasi bentuk berbasis gradien dan pola tekstur yang esensial untuk deteksi kepala dan bahu yang akurat. Dengan menggunakan Principal

Component Analysis (PCA) untuk mengurangi dimensi fitur, efisiensi komputasi meningkat tanpa mengorbankan kekuatan diskriminatif dari fitur yang digabungkan. Pendekatan *multi-level* ini memastikan ketahanan terhadap variasi dalam skala, pose, dan pencahayaan yang merupakan tantangan umum dalam mendeteksi orang di kerumunan.

Kemudian, [2] juga menjelaskan terkait penggunaan kombinasi HOG dan Completed LBP (CLBP) di mana metode ini digunakan untuk mendeteksi orang dalam kerumunan padat secara akurat berdasarkan area kepala-bahu. Integrasi CLBP dengan HOG dalam metode yang diusulkan [2] dapat menjadi inspirasi untuk mengadaptasi dan memperluas teknik serupa dalam lingkungan kelas atau ruangan yang beragam.

Berdasarkan kedua penelitian di atas dan juga beberapa penelitian serupa, seperti [3] dan [4], penulis termotivasi untuk menggabungkan teknik ini dengan SVM, dengan harapan dapat mengeksplorasi potensi peningkatan akurasi dan efisiensi sistem deteksi orang dalam berbagai kondisi ruangan.

III. METODOLOGI

Penelitian ini akan berjalan sesuai dengan alur pada Gambar 1. Pertama, dataset kelas dan dataset MPII dipersiapkan, di mana dilakukan *cropping* secara manual untuk kelas. Kedua dataset kemudian melalui tahap preprocessing yang meliputi *resizing*, *grayscale*, dan *Gaussian blur*. Setelah *preprocessing*, data dibagi menjadi data latih dan uji. Pada tahap ekstraksi fitur, *Histogram of Oriented Gradient* (HOG) dan *Local Binary Pattern* (LBP) digunakan untuk mengekstraksi fitur dari data latih dan uji. Data latih digunakan untuk melatih model menggunakan *Support Vector Machine* (SVM). Model yang terlatih divalidasi dan nantinya akan digunakan untuk prediksi dengan mekanisme *sliding window*. Jika objek terdeteksi dan *confidence score* melebihi *threshold*, maka akan dihitung sebagai orang dan akan *dibuat bounding box* pada window tersebut. Hasil akhir ditampilkan dan dianalisis sebelum proses berakhir.

A. Dataset

Dataset yang digunakan dalam penelitian ini terdiri dari dua sumber utama, yaitu *dataset* MPII yang diunduh dari Max Planck Institute for Informatics [5], serta *dataset* yang diambil menggunakan kamera ponsel. *Dataset* MPII berisi gambar-gambar orang dengan berbagai pose dan kondisi lingkungan, sementara *dataset* yang diambil sendiri difokuskan pada lingkungan kelas.

Untuk membentuk sampel positif, gambar-gambar dari kedua sumber tersebut di-*crop* pada bagian kepala hingga pundak. Proses ini dilakukan dengan hati-hati untuk memastikan bahwa setiap sampel positif mencerminkan bagian tubuh yang relevan untuk deteksi kepala-bahu. Total jumlah sampel positif yang diperoleh dari kedua sumber adalah 529 untuk *dataset training* dan 123 untuk *dataset testing*. Sampel negatif juga diambil dari gambar yang sama, tetapi dengan teknik *random cropping* pada bagian gambar yang tidak mengandung kepala hingga pundak. Proses ini dilakukan untuk memastikan bahwa sampel negatif bervariasi dan mencakup berbagai area gambar yang berbeda. Jumlah total sampel negatif yang dihasilkan

adalah 1177 untuk *dataset training* dan 263 untuk *dataset testing*.



(a) Contoh dataset positif



(b) Contoh dataset negatif

Fig. 2: Dataset

B. Histogram of Oriented Gradients

Histogram of Oriented Gradients atau HOG, adalah fitur deskriptor dalam bidang penglihatan komputer, yang digunakan untuk mengidentifikasi dan menemukan obyek di dalam gambar. Fitur deskriptor ini digunakan untuk mengurangi komputasi dan *noise* pada gambar yang fiturnya akan diekstraksi.

Hal pertama yang dilakukan pada dataset adalah mengubah gambar berwarna menjadi citra *grayscale*. Ini dilakukan dengan tujuan untuk mempertahankan informasi kontras pada citra semirip mungkin [6].

Formula yang paling sederhana untuk menerapkan *grayscale* adalah sebagai berikut:

$$I = \alpha_r R + \alpha_g G + \alpha_b B, \quad (1)$$



Fig. 3: Mengubah gambar menjadi *grayscale*

Analisis gradien citra pada HOG mengungkapkan arah dan besarnya perubahan intensitas dari satu piksel ke piksel disebelahnya.

$$G(x, y) = \sqrt{(G_x)^2 + (G_y)^2} \quad (2)$$

$$\theta = \alpha \cdot \tan \left(\frac{G_y}{G_x} \right) \quad (3)$$

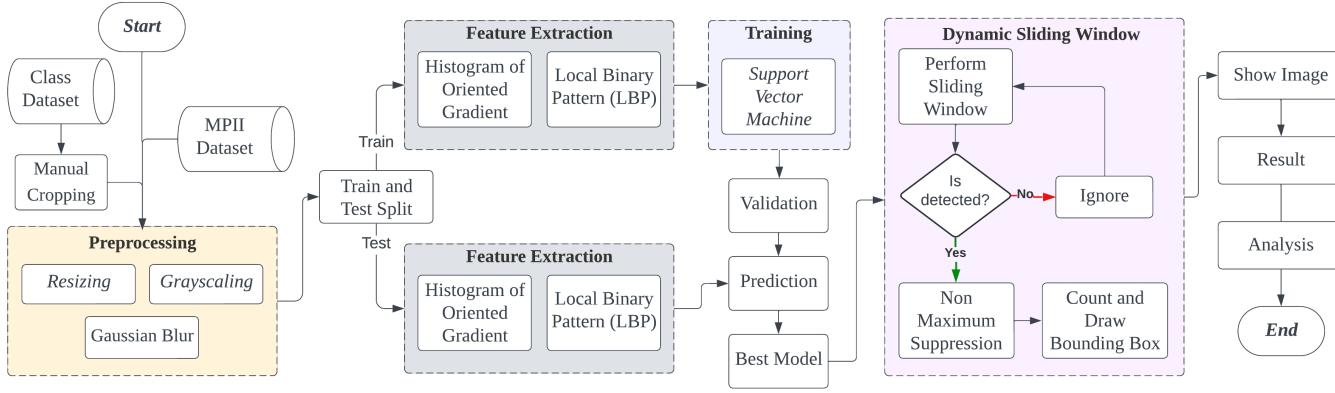


Fig. 1: Diagram Alir Penelitian

HOG membagi citra menjadi beberapa blok, dan dalam blok akan dibagi lagi menjadi beberapa sel yang berisi piksel-piksel. Histogram channel akan menyebar secara merata dari 0 sampai 180 derajat.

Langkah selanjutnya adalah menormalisasikan nilai gradien yang terhubung dengan komponen terdekat sebagai fitur. Fitur ini akan terhubung satu sama lain sehingga fitur yang diekstrak akan tetap memiliki nilai keterikatan dengan komponen tetangganya.



Fig. 4: HOG

Menormalisasikan blok adalah langkah dimana semua blok histogram akan dikombinasikan pada region untuk mencapai normalisasi. Formula yang digunakan ditunjukkan pada persamaan 4.

$$v_n = \frac{v}{\sqrt{\|v\|_2^2 + e^2}} \quad (4)$$

C. Local Binary Patterns

Local Binary Pattern adalah metode fitur ekstraksi untuk tekstur citra. LBP bisa mengekstraksi informasi lokal secara efisien dan tidak terpengaruh oleh perubahan grayscale monoton pada gambar.

Nilai LBP untuk piksel tengah (x, y) bisa didapatkan dengan formula:

$$LBP(x, y) = \sum_{p=0}^7 s(g_c - g_p)2^p \quad (5)$$

dimana $s(z)$ adalah sebuah fungsi *thresholding*.

$$s(z) = \begin{cases} 1 & z \text{ greater or equal than 0} \\ 0 & z \text{ less than 0} \end{cases} \quad (6)$$

Fig menunjukkan prosedur dasar untuk menghasilkan nilai LBP dari piksel lokal dalam 3×3 ketetanggaan dengan dasar persamaan 5 dan 6.

Selanjutnya area citra yang direpresentasikan dari histogram kode LBP akan diproses dengan:

$$H_{LBP}(i) = \sum_{x,y} \delta(i, LBP(x, y)), \quad (i = 0, \dots, 2^7) \quad (7)$$

yang mana delta adalah Kronecker delta yang didefinisikan sebagai

$$\delta(z) = \begin{cases} 1 & i \text{ equal to } j \\ 0 & i \text{ is not equal to } j \end{cases} \quad (8)$$

Pada pengekstraksian fitur kali ini, LBP yang digunakan adalah 3×3 . Hasil dari pengekstraksian fitur dengan LBP bisa dilihat pada gambar berikut.

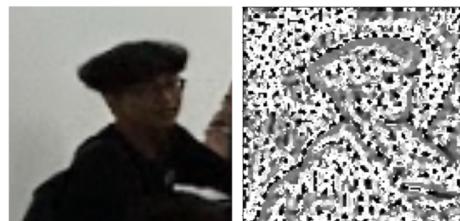


Fig. 5: LBP

D. Support Vector Machine

Algoritma *Support Vector Machine* (SVM) bertujuan menemukan garis atau permukaan yang memisahkan dua kelompok data sebaik mungkin. Jika data dapat dipisahkan secara linear, SVM akan mencari garis pemisah terbaik tersebut [7]. Namun, ketika data tidak bisa dipisahkan secara linear, SVM menggunakan fungsi *kernel* untuk memetakan data dari ruang berdimensi rendah ke ruang berdimensi tinggi, di mana data lebih

mudah dipisahkan dengan garis lurus [8]. Dengan demikian, SVM dapat menemukan garis pemisah optimal di ruang yang lebih tinggi tanpa menghadapi masalah "bencana dimensi", sehingga mampu mengklasifikasikan data yang sebelumnya sulit dipisahkan dengan benar [9]. Fungsi *kernel* dinyatakan dalam bentuk $K(x_i, x_j) = \phi(x_i)^T \phi(x_j)$.

Berikut adalah empat *kernel* dasar:

- Linear:

$$K(x_i, x_j) = x_i^T x_j \quad (9)$$

- Polinomial:

$$K(x_i, x_j) = (\gamma x_i^T x_j + r)^d, \gamma > 0 \quad (10)$$

- Radial Basis Function (RBF):

$$K(x_i, x_j) = \exp(-\gamma |x_i^T x_j|^2), \gamma > 0 \quad (11)$$

- Sigmoid:

$$K(x_i, x_j) = \tan(\gamma x_i^T x_j + r) \quad (12)$$

di mana γ , r , dan d adalah parameter-parameter *kernel*.

Dalam eksperimen, *kernel* RBF akan digunakan untuk mengklasifikasikan data fitur dari *dataset training*.

E. Sliding Windows

Teknik *sliding window* adalah pendekatan algoritmik yang melibatkan pemilihan *fixed-size subset* atau sebuah "window", dari sebuah *dataset* yang lebih besar. *Window* tersebut akan digerakkan sepanjang *dataset* secara bertahap sesuai dengan *stride* yang telah ditentukan.

Dalam eksperimen kali ini, *sliding window* akan digunakan untuk mengklasifikasikan objek yang berada di dalam *window*. Sebuah *window* dengan ukuran tetap akan tercipta dan operasi akan dimulai dari bagian kiri atas gambar. *Window* ini akan bergeser melintasi gambar. Untuk setiap posisi, model akan menghitung *confidence value* untuk bagian gambar yang berada di dalam *window*. Harapannya terdapat bagian dari objek di dalam batas *window* dan objek tersebut dapat terdeteksi.

Proses ini akan dilakukan secara berulang pada beberapa ukuran gambar berbeda dengan ukuran *window* yang sama. Pengulangan ini bertujuan untuk memastikan tidak ada detail yang terlewat selama proses deteksi. Skala *downscale* yang digunakan dalam eksperimen ini adalah 1.5.

Dalam eksperimen ini, proses *slidng window* akan dilakukan dengan *stride* dinamis berdasarkan hasil prediksi dari model. Apabila hasil prediksi model adalah 0, *stride* akan menjadi besar. Sebaliknya, ketika hasil prediksinya 1, posisi *sliding window* akan dibuat mundur, kemudian proses *sliding window* akan dilakukan dengan *stride* yang kecil.

IV. HASIL DAN PEMBAHASAN

A. Hyperparameter Tuning Model SVM

Untuk menemukan kombinasi parameter optimal yang meningkatkan kinerja model, hyperparameter tuning menggunakan GridSearchCV pada SVM. Dalam konteks ini, parameter grid mencakup beberapa nilai untuk setiap parameter seperti yang ditunjukkan pada Tabel I.



(a) Frame 1



(b) Frame 2

Fig. 6: Visualisasi *sliding window*

TABLE I: Parameter yang dimasukkan dalam parameter grid

Parameter	Values
C	[0.1, 1, 10]
gamma	[1, 0.1, 0.01]
kernel	['linear', 'rbf']

Setelah dilakukan hyperparameter tuning, didapatkan parameter paling optimal yaitu 'C' sebesar 1, 'gamma' sebesar 1, dan 'kernel' yang digunakan adalah linear. Laporan skor untuk model terbaik tersebut diberikan pada tabel II berikut.

TABLE II: Skor akurasi model terbaik

	precision	recall	f1-score
0	0.79	1.00	0.88
1	1.00	0.42	0.59
accuracy	0.82		
macro avg	0.89	0.71	0.74
weighted avg	0.86	0.82	0.79

Berdasarkan Tabel II, model memiliki presisi, *recall*, dan f1-score sebesar 0.79, 1.00, dan 0.88 untuk deteksi hasil negatif. Untuk deteksi hasil positif, model memiliki presisi sebesar 1.00, namun nilai *recall* dan f1-score hanya sebesar 0.42 dan 0.59 saja. Secara keseluruhan, model yang dilatih memiliki akurasi sebesar 0.82.

B. Hasil Sliding Window

Berikut adalah beberapa hasil penjalanan *sliding window* pada beberapa gambar:

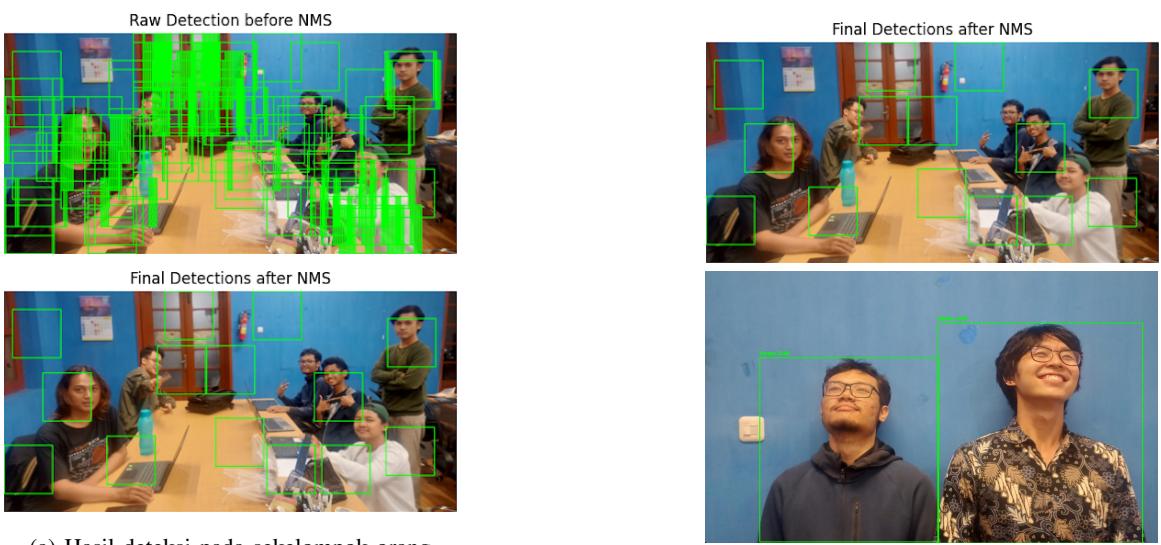
Jumlah objek yang terdeteksi pada gambar 7, 8a, 8b secara berturut-turut adalah 31, 14, dan 3 objek. Dari hasil penjalanan *sliding window* di atas, terlihat bahwa masih terdapat *false positive* dan *false negative*.

C. Perbandingan dengan Model Lain

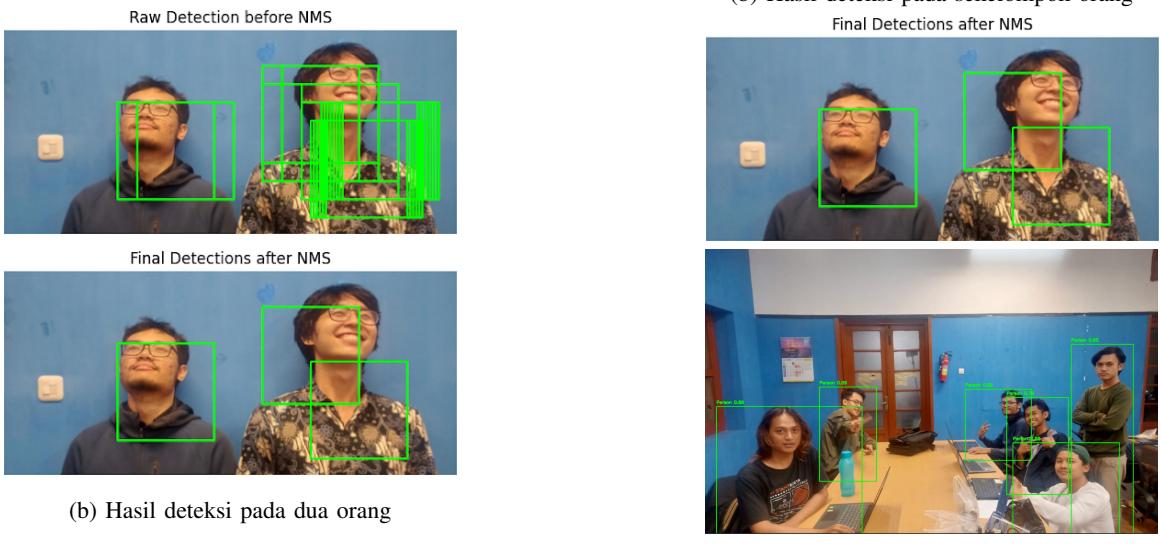
Berdasarkan Gambar 9c, dapat diperhatikan bahwa hasil model YOLOv5 sangat baik untuk mendeteksi adanya manusia dalam suatu frame jika dibandingkan dengan metode yang



Fig. 7: Hasil deteksi pada dataset kelas menggunakan *sliding window*



(a) Hasil deteksi pada sekelompok orang



(b) Hasil deteksi pada dua orang

Fig. 8: Hasil pada data terbatas menggunakan *sliding window*



Fig. 9: Perbandingan antara model YOLOv5 dan model yang dilatih

diajukan. Hal ini menjadi layak karena model YOLOv5 sendiri telah di-train dengan dataset COCO dengan jumlah gambar yang besar dan representatif.

V. KESIMPULAN DAN SARAN

Sebagai kesimpulan, dalam penelitian ini kami telah bereksperimen untuk mendeteksi dan menghitung jumlah orang di suatu kelas atau ruangan dengan pendekatan ekstraksi fitur dan *machine learning*. Dengan fitur HOG, LBP dan model SVM didapatkan skor akurasi validasi di gabungan dataset kelas dan MPII sebesar 0,82.

Meskipun nilai yang didapat dalam training dan validasi cukup baik, tetapi hasil testing dengan metode dynamic sliding window di data kelas masih kurang dapat menemukan dan mendeteksi dengan benar objek manusianya. Hal ini dimungkinkan karena beberapa faktor yaitu (1) Dataset yang tidak terlalu banyak, imbalance dan kurang representatif, (2) Ekstraksi fitur yang masih sederhana dan belum ada feature parameter tuning, dan (3) Model yang masih sederhana.

Untuk penelitian lebih lanjut, kami menyarankan adanya generasi dataset yang lebih baik untuk merepresentasikan objek manusia di gambar dengan berbagai macam keadaan. Selain itu, *feature parameter tuning* juga disarankan untuk dilakukan untuk mendapatkan hasil yang lebih optimal. Penggunaan model lain yang lebih mutakhir seperti Convolutional Neural Network (CNN), Haar-Cascade, maupun pre-trained model seperti You Look Only Once (YOLO) juga dapat diujicoba untuk mendapatkan akurasi yang lebih optimal.

REFERENCES

- [1] C. Zeng and H. Ma, "Robust head-shoulder detection by pca-based multilevel hog-lbp detector for people counting," in *2010 20th International Conference on Pattern Recognition*, IEEE, Aug. 2010.
- [2] M. A. Hassan, I. Pardiansyah, A. S. Malik, I. Faye, and W. Rasheed, "Enhanced people counting system based head-shoulder detection in dense crowd scenario," in *2016 6th International Conference on Intelligent and Advanced Systems (ICIAS)*, IEEE, Aug. 2016.
- [3] W. Zhou and S. Luo, "Pedestrian detection with improved lbp and hog algorithm," *OALib*, vol. 05, no. 04, p. 1–10, 2018.
- [4] M. Ghorbani, A. T. Targhi, and M. M. Dehshibi, "Hog and lbp: Towards a robust face recognition system," in *2015 Tenth International Conference on Digital Information Management (ICDIM)*, IEEE, Oct. 2015.
- [5] "MPII Human Pose Database — human-pose.mpi-inf.mpg.de." <http://human-pose.mpi-inf.mpg.de/>. [Accessed 12-06-2024].
- [6] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, IEEE.
- [7] Y. Liang, M. L. Reyes, and J. D. Lee, "Real-time detection of driver cognitive distraction using support vector machines," *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, p. 340–350, June 2007.
- [8] G. Zhang, F. Gao, C. Liu, W. Liu, and H. Yuan, "A pedestrian detection method based on svm classifier and optimized histograms of oriented gradients feature," in *2010 Sixth International Conference on Natural Computation*, IEEE, Aug. 2010.
- [9] F. Zhu, X. Yang, J. Gu, and R. Yang, "A new method for people-counting based on support vector machine," in *2009 Asia-Pacific Conference on Information Processing*, IEEE, July 2009.