

DETEKSI OBYEK MANUSIA PADA SISTEM KEAMANAN GEDUNG MENGUNAKAN WEBCAM

Fajar Hermawanto

Program Studi D3 Teknik Informatika

Politeknik Gorontalo

e-mail : fajar@poligon.ac.id, rulysinukun@poligon.ac.id

ABSTRAK

Keamanan merupakan prioritas yang utama pada sebuah gedung dari kejahatan pencurian maupun perampokan. Berbagai alat keamanan telah dikembangkan baik yang berteknologi sensor ultrasonik, inframerah, PIR, dan CCTV. Pada saat ini teknologi CCTV menjadi alternatif yang terbaik sebagai pengamanan, karena mampu memberikan informasi secara visual dari rekaman kejadian. Rekaman kejadian disimpan secara berkala, selanjutnya akan dievaluasi secara berkala atau saat terjadi suatu tindak kejahatan untuk mengetahui reka kejadian dan tersangka tindak kejahatan tersebut. Permasalahan yang ada saat ini adalah media penyimpanan terus melakukan perekaman hasil capture dari kamera secara terus menerus walaupun dilokasi pengintaian tidak ada manusia, sehingga kurang efektif dalam hal pencarian obyek manusia secara otomatis. Pengamatan video saat mencari tersangka dalam rekaman video pun akan membutuhkan waktu yang lama karena system penyimpanan video terjadi secara terus menerus. Permasalahan yang ada tersebut dapat diselesaikan melalui suatu program yang mampu mendeteksi obyek manusia secara otomatis. Deteksi obyek manusia melalui kamera secara otomatis dapat dilakukan menggunakan metode Hog Cascade. Metode ini mampu memberikan pendeteksian yang akurat secara realtime.

Kata kunci : Keamanan, Pencurian, Perampokan, CCTV, Hog Cascade,

1. Pendahuluan

Seiring dengan perkembangan zaman dan perkembangan teknologi yang semakin pesat, tingkat kejahatan semakin meningkat pula. Era globalisasi telah menuntut manusia untuk menciptakan keamanan dalam bekerja. Demikian halnya dengan sistem keamanan gedung untuk perkantoran maupun rumah mewah lainnya juga ikut berkembang.

Sistem keamanan yang ada antara lain menggunakan kamera CCTV melalui operator yang meminta. Ada juga dengan menggunakan sensor IR dan ultrasonik, tetapi dalam penggunaannya harus ada sumber dan sensor. Kamera CCTV merupakan sistem keamanan yang paling banyak digunakan saat ini. Data informasi kejadian yang dapat direkam secara visual melalui media penyimpan dan dapat diputar kembali dalam bentuk video.

Pengamatan video dilakukan secara berkala atau setelah terjadi tindak kejahatan. Pada saat pengamatan video, pencarian target tersangka pada video membutuhkan waktu yang lama untuk dapat menemukan obyek manusia dalam video. Penelitian deteksi pejalan kaki pada video menggunakan metode HOG (*Histograms of Oriented Gradient*) dilakukan oleh Achmad S dan Agus H. pada

penelitian tersebut menunjukkan hasil yang cukup baik dengan tingkat akurasi sebesar 73,25%.

Metode hog cascade digunakan untuk mendeteksi manusia pada penelitian yang dilakukan oleh Chengbin dkk. Pada penelitian tersebut menyajikan pendekatan deteksi manusia yang dapat memproses gambar dengan cepat dan mendeteksi obyek secara akurat. Fitur yang digunakan pada penelitian tersebut adalah cascade dari HOG (Histogram dari Oriented Gradien) dan LBP (Pola Binary local).

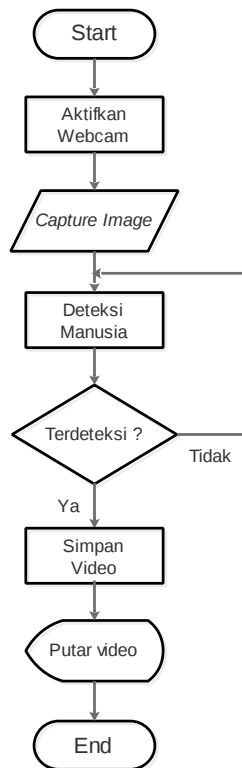
Dengan demikian bahwa untuk mempermudah dan mempercepat proses pengamatan video diperlukan suatu sistem yang mampu menyeleksi obyek yang dikehendaki dalam hal ini adalah manusia. Proses seleksi obyek dapat dilakukan oleh program secara otomatis menggunakan metode hogcascade. Pada penelitian ini akan membuat sistem yang mampu mendeteksi manusia pada video pengamatan yang direkam webcam.

2. Metode

2.1. Perancangan Sistem

Proses pencarian obyek manusia dalam video menggunakan metode hogcascade yang terdapat pada library opencv. Kamera webcam merekam lokasi kejadian secara realtime, proses pencarian

obyek manusia secara otomatis dilakukan oleh program pada komputer. Hogcascade merupakan file data dalam bentuk xml yang berisi data fitur manusia dalam bentuk *full body*.



Gambar 1 . Flow Chart Sistem Deteksi Manusia Pada Webcam

Pada flowchart pada gambar 1 yaitu saat algoritma dijalankan. Program dimulai dengan mengaktifkan webcam agar dapat menangkap informasi visual yang akan diproses pada komputer, selanjutnya didapatkan capture image untuk diproses menggunakan metode Hog dalam mendeteksi obyek khususnya manusia. Human detection menggunakan 2 langkah, yaitu HOG (Histogram of Oriented Gradients) descriptor dan SVM (Support Vektor Machine) classifier. Setelah mendeteksi adanya manusia. Algoritma berlanjut pada sistem penyimpanan file dalam bentuk video.

2.2. Histograms of Oriented Gradient (HoG)

Histograms of Oriented Gradient (HOG) sendiri merupakan metode deteksi obyek yang didasarkan pada tampilan dan bentuk obyek secara lokal. Ciri yang digunakan dalam metode ini adalah distribusi intensitas gradient beserta arahnya. Metode HOG banyak digunakan sebagai metode detector berbagai obyek. HOG digunakan untuk

mendeteksi pejalan kaki pada malam hari dengan kamera infra merah. Parameter yang dapat digunakan untuk mengoptimalkan metode HOG antara lain ukuran cell, ukuran dan normalisasi blok serta bobot dari histogram. Deskriptor HOG ditentukan berdasarkan 3 (tiga) langkah yaitu (a) menghitung gradient citra, (b) membuat histogram dari arah gradient untuk setiap cell dan (c) melakukan normalisasi histogram untuk setiap blok.

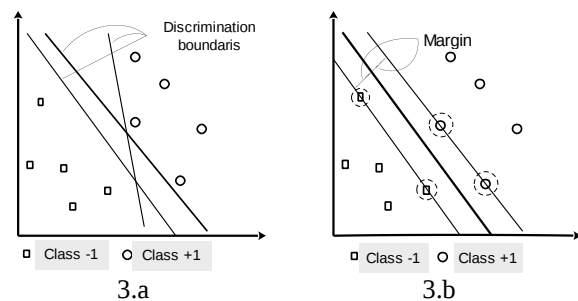


Gambar 2. Pedestrian Detection

2.3. SVM

SVM merupakan metode learning machine yang bekerja atas prinsip Structural Risk Minimization (SRM) dengan tujuan menemukan hyperplane terbaik yang memisahkan dua buah class (kategori) pada input space. Prinsip dasar SVM adalah linear classifier, dan sedang dikembangkan agar dapat bekerja pada non-linear.

Untuk mengecek apakah dalam window tersebut terdapat manusia atau tidak, digunakan SVM classifier untuk memisahkan *human* dan *nonhuman*.



Gambar 3. SVM dalam upaya mencari hyperplane terbaik untuk memisahkan class -1 dan +1

Pada SVM classifier dan algoritma klasifikasi yang berusaha memisahkan sebuah hyperplane optimal (Cristiani & Shawe-Taylor, 2000). Gambar 3.a menunjukkan ada beberapa *pattern* yang

merupakan anggota dari dua buah *class*: +1 dan -1. *Problem* klasifikasi dapat di terjemahkan dengan usaha menemukan garis (*hyperplane*) yang memisahkan kedua *class* tersebut. Berbagai alternative garis pemisah (*discrimination boundaris*) ditunjukkan pada gambar 3.a

Gambar 3.b menunjukkan penggunaan garis *hyperplane* yang memisahkan 2 *class* dengan margin tertentu. Formula dari linear *hyperplane* yang memisahkan kedua *class* : Nilai *w* dapat diperoleh dari rumus :

$$w = \sum_{i=1}^n a_i x_i y_i$$

dimana x_i merupakan vector contoh (HOG descriptor examples), y_i merupakan label data training positif atau negative (+1 atau -1) dan a_i merupakan konstanta pengali (multiplier).

3 Pembahasan

Berdasarkan metode penelitian yang telah diusulkan dan dijelaskan di atas, dilakukan serangkaian percobaan menggunakan perangkat lunak OpenCV. Beberapa fungsi dari OpenCV digunakan untuk mempermudah proses percobaan.

3.1. Frame Extraction

Frame yang digunakan pada penelitian ini dengan lebar 320 dan tinggi 240. Perekaman video menggunakan kamera webcam yang diletakkan pada salah satu sudut ruangan yang bersifat statis. Proses pengamatan dilakukan secara real time menggunakan fungsi `VideoCapture`(`CV_CAP_ANY`) pada `opencv`. Hasil capture dideklarasikan dalam bentuk *Mat*, variable *mat* yang selanjutnya akan diproses oleh fungsi SVM Detektor.

3.2. Background Modelling

Citra digital hasil *capture* dari webcam merupakan lingkungan yang statis sehingga variable yang digunakan untuk menentukan tingkat penajaman citra juga tetap. Lokasi webcam diletakkan dalam ruang dengan pencahayaan tetap yaitu cahaya dari lampu penerangan. Dalam keadaan yang seperti tersebut sangat memungkinkan dilakukan proses pengolahan untuk mengenali obyek yang diinginkan dikenali, dalam hal ini yaitu manusia.

3.3. Preprocessing

Normalisasi citra hasil capture dari webcam menggunakan fungsi dari `opencv`

yaitu `cap.set(CV_CAP_PROP_FRAME_WIDTH,x)` dan `cap.set(CV_CAP_PROP_FRAME_HEIGHT,y)`. *x* merupakan lebar frame dan *y*, merupakan panjang frame. Pada penelitian ini lebar dan panjang yang digunakan adalah 320 x 240. Normalisasi menggunakan normalisasi warna dan gamma sehingga warnanya menjadi lebih gelap.

3.4. Background Subtraction

Pada proses background subtraction citra hasil komputasi gradient menggunakan metode 1-D centered dengan matriks [-1, 0,1]. Dengan persamaan derivatif parsial untuk menghitung fungsi citra *f(x,y)* sehingga akan diperoleh nilai *x* dan *y* yang akan digunakan untuk menghitung gradient.

3.5. Image Segmentation

Pada proses segmentasi, citra hasil dari *background subtraction* dibagi atas daerah-daerah kecil yang disebut sel. Dalam sel masing-masing dibagi atas 9 channel yang memberikan vote untuk interval sudut tertentu, dan digunakan dalam menentukan histogram yang dibuat.

3.6. Object Classification

Setelah obyek-obyek tersegmentasi dengan baik, dilakukan seleksi kembali dengan melakukan klasifikasi sehingga didapatkan obyek pejalan kaki dan obyek bukan pejalan kaki. Seleksi sederhana dilakukan dengan melihat ukuran setiap obyek yang didapatkan dengan mendeteksi garis batas (*boundary*) untuk setiap obyek yang telah tersegmentasi. Pendeteksian garis batas untuk setiap obyek memanfaatkan fungsi yang sudah disediakan oleh OpenCV.

Citra hasil capture kamera webcam dideklarasikan dalam bentuk *Mat*, selanjutnya adalah proses pendeteksian obyek manusia yaitu menggunakan SVM Detector. Fungsi ini juga sudah disediakan oleh `opencv`, dengan fungsi tersebut pada saat dalam lokasi frame citra yang dapat ditangkap oleh webcam akan langsung dapat mendeteksi adanya obyek manusia yang dikenali.

3.7. Human Detection



Gambar 4. Deteksi tampak depan

Pendeteksian obyek manusia ditentukan setelah proses klasifikasi antara obyek manusia dan bukan manusia telah didapatkan. Pada saat obyek manusia dikenali ditandai dengan kotak warna hijau. Cara menampilkan kotak warna hijau dilakukan dengan fungsi *rectangle()* dari OpenCV. Hasil deteksi obyek manusia menggunakan webcam dapat dilihat pada gambar 4.



Gambar 5. Deteksi tampak belakang

Dari hasil pengujian pendeteksian manusia menggunakan webcam, program mampu mendeteksi posisi manusia dari beberapa arah. Pada gambar 4 menunjukkan manusia terdeteksi dari tampak depan. Sedangkan pada gambar 5 manusia mampu terdeteksi dari tampak belakang.



Gambar 6. Deteksi tampak samping

Pada gambar 6 deteksi manusia masih mampu bekerja walaupun tidak secara keseluruhan badan tertangkap oleh kamera.

4 Kesimpulan dan Saran

4.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan didapatkan kesimpulan sebagai berikut :

1. Jarak obyek terhadap kamera sangat berpengaruh terhadap hasil pendeteksian,

manusia akan terdeteksi oleh kamera jika minimal 2/3 bagian obyek harus tertangkap oleh kamera .

2. Posisi kamera terhadap obyek sangat berpengaruh pada hasil deteksi terhadap obyek. Posisi kamera tegak lurus terhadap obyek merupakan posisi yang paling baik dalam mendeteksi obyek manusia.

4.2 Saran

Untuk memaksimalkan kinerja dapat menggunakan webcam dengan resolusi yang lebih tinggi, agar hasil deteksi obyek bias ditingkatkan.

Daftar pustaka

- [1] Solichin, A. Harjoko, A. (2013). Deteksi Pejalan Kaki Pada Video dengan Metode Fastest Pedestrian Detector in The West (FPDW). Jurnal Ticom Vol.2 No.1 , hal 202-205
- [2] Zeng, C. Ma, H. Ming, A. (2010). Fast Human Detection Using Mi-SVM and A Cascade of HOG-LBP Features. *Proceedings of 2010 IEEE 17th International Conference on Image Processing*. Sep. 26-29. hal 3845-3848.
- [3] Jia, H X. Zhang Y. (2007). Fast Human Detection by Boosting Histograms of Oriented Gradients. *Fourth International Conference on Image and Graphics*. hal 683-687.
- [4] Kamijo, S. Fujimura, K. Shibayama, Y. (2010) Pedestrian Detection Algorithm for On-board Cameras of Multi View Angles. *IEEE Intelligent Vehicles Symposium University of California, San Diego, CA, USA, June 21-24*. Hal 973- 980.
- [5] Zarka, N. Alhalah, Z. Deeb, R. Real-Time Human Motion Detection and Tracking. Hal 1 – 6.
- [6] Tathe, S.V. Narote, S.P.(2013). Real-Time Human Detection and Tracking. *Annual IEEE India Conference (INDICON)*.
- [7] Bradski, G. Kaehler, A.(2008). *Learning OpenCV*.
- [8] Brahmabhatt, S. *Practical OpenCV. Hands On Projects For Computer Vision On The Windows, Linux And Raspberry Pi Platforms*.