

Deteksi Pengambilan Barang Dari Rak Dengan Pengolahan Citra

Irfan Haris Widyadhana 13517041
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
E-mail (gmail): 13517041@std.stei.itb.ac.id

Abstract—Salah satu pemanfaatan teknologi computer vision yang cukup sering digunakan yakni untuk kebutuhan surveillance, pada penerapannya komputer akan mengekstrak informasi dari suatu video yang diperoleh dari kamera surveillance untuk selanjutnya dapat dilakukan respons aksi lebih lanjut. Suatu aplikasi deteksi pengambilan barang dari rak yang digunakan untuk pencatatan proses belanja pada supermarket dapat dikembangkan dengan menggunakan proses-proses pengolahan citra yang sederhana. Dengan membandingkan frame-frame yang berurutan dapat diambil informasi yang merepresentasikan aksi pengambilan barang terjadi atau tidak.

Keywords—Citra, Warna, Gaussian Blur, Threshold, Kontur,

I. PENDAHULUAN

Teknologi computer vision merupakan salah satu cabang dari teknologi artificial intelligence yang spesifik membahas terkait citra gambar dan video. Bidang computer vision berusaha untuk melatih komputer agar mampu mendapatkan suatu informasi dari input citra yang diberikan. Salah satu contoh teknologi computer vision misalnya sistem pengenalan wajah dan sistem klasifikasi barang.

Pemanfaatan computer vision untuk pengolahan video memiliki sedikit perbedaan dengan teknik pengolahan citra gambar. Sebuah video tersusun dari serangkaian frame gambar ditampilkan dengan kecepatan tertentu sehingga untuk video terdapat tambahan dimensi waktu di dalamnya tidak hanya dimensi spasial gambar saja. Faktor ini menyebabkan dalam pengolahan video digital ekstraksi fitur dalam dimensi waktu perlu untuk dipertimbangkan.

Salah satu contoh pemanfaatan teknologi computer vision pada video yakni digunakan untuk kebutuhan surveillance. Komputer dapat mencatat terjadinya anomali yang terjadi pada suatu video surveillance untuk dilakukan tindakan lebih lanjut. Proses inferensi oleh komputer ini akan sangat membantu pekerjaan manusia jika dibandingkan dengan cara lama dimana manusia harus memantau video terus menerus dalam jangka waktu yang lama.

Perusahaan teknologi asal Amerika Serikat yang bernama Amazon memanfaatkan computer vision untuk surveillance ini pada toko supermarket mereka yang bernama Amazon Go

Store. Pada toko ini proses berbelanja yang dilakukan pengunjung dicatat oleh kamera pengawas yang terpasang di seluruh penjuru toko. Dengan teknologi computer vision, komputer bisa mengetahui barang apa yang diambil seseorang dan juga apakah orang tersebut mengembalikan barang yang sudah diambilnya kembali ke rak.

Keunggulan dari dilakukannya proses berbelanja seperti ini adalah pencatatan barang yang diambil seseorang dilakukan secara otomatis saat sedang berbelanja. Oleh karena itu pada Amazon Go Store tidak terdapat kasir untuk menghitung barang yang dibeli. Setelah selesai berbelanja, pembeli akan langsung keluar melalui gerbang yang disediakan dan saldo dari akun pembeli tersebut akan terpotong secara otomatis sesuai dengan jumlah barang yang dibelinya. Teknologi deteksi pengambilan barang ini merupakan milik perusahaan sehingga tidak terbuka untuk publik, tetapi sudah terdapat beberapa penelitian yang mencoba untuk mengembangkan teknologi yang serupa dengan berbagai teknik yang beraneka ragam.

Aplikasi deteksi pengambilan barang sederhana dapat dibangun dengan menggunakan metode yang sederhana yakni dengan membandingkan frame-frame yang berurutan pada video untuk diambil informasi yang merepresentasikan aksi pengambilan barang. Proses ekstraksi informasi ini dapat memanfaatkan metode-metode sederhana pada pengolahan citra seperti konversi warna, penebalan garis dan juga deteksi kontur. Jika aplikasi deteksi pengambilan barang ini berhasil dilakukan dapat dilanjutkan dengan membangun suatu model untuk melakukan klasifikasi aksi yang dilakukan apakah berupa mengambil atau mengembalikan barang ke rak dan juga model untuk mendeteksi dan mengklasifikasikan barang yang diambil.

II. LANDASAN TEORI

A. Citra

Citra merupakan representasi fungsi dari intensitas cahaya yang disusun di dalam suatu bidang dimensi. Dalam citra digital digunakan representasi matriks untuk menyimpan intensitas cahaya pada suatu koordinat titik di dalam matriks yang selanjutnya disebut dengan pixel. Nilai setiap pixel yang mengandung intensitas cahaya ini biasa disebut dengan nilai keabuan yang nilai rentangnya direpresentasikan dengan bit.

Citra diam yang biasa dikenal sebagai gambar citra biasa adalah sebuah citra tunggal yang tersusun hanya dari satu matriks. Sementara itu terdapat citra bergerak yang biasa dikenal dengan video merupakan rangkaian citra diam yang ditampilkan secara sekuensial sehingga memberi kesan sebagai gambar yang bergerak. Citra bergerak ini tersusun dari banyak matriks yang ditampilkan dengan sangat cepat sehingga perbedaan dari pixel-pixel dari tiap matriks penyusun yang berurutan ditangkap manusia sebagai suatu gerakan.

Citra digital berbeda dengan citra analog. Citra analog dihasilkan dari alat untuk akuisisi citra analog, sebagai contoh yakni mata manusia dan kamera analog. Gambaran yang tertangkap oleh mata manusia dan foto atau film yang tertangkap oleh kamera analog merupakan contoh dari citra analog. Citra analog memiliki tingkat resolusi yang sangat baik dibandingkan dengan citra digital.

Citra analog dibentuk dari sinyal analog yang bersifat kontinu, hal inilah yang menyebabkan citra analog hasilnya selalu lebih baik dibandingkan citra digital yang disusun dari sinyal digital yang sifatnya diskrit. Tetapi kelemahan yang dimiliki oleh citra analog adalah representasi kontinu yang dimiliki oleh citra analog tidak dapat disimpan dalam representasi komputer. Sehingga untuk representasi komputer lebih cocok digunakan citra dalam format digital.

Pengolahan yang dilakukan terhadap citra dilakukan dengan menganalisis citra terlebih dahulu. Terdapat beberapa analisis yang dapat dilakukan terhadap citra seperti mendeteksi noise, pengecekan kontras, deteksi blur dan lainnya. Setelah analisis citra dilakukan, dapat dilakukan proses pengolahan lebih lanjut untuk bisa didapatkan citra dengan kualitas yang lebih baik. Untuk citra digital dapat dilakukan proses pengolahan citra seperti melakukan blurring, penajaman resolusi, pencerahan warna dan lain sebagainya. proses pengolahan citra ini dilakukan dengan memodifikasi pixel-pixel yang ada di dalam citra dengan menggunakan fungsi-fungsi pengolahan citra.

B. Warna Citra

Dalam pengolahan citra warna merupakan hal yang cukup penting untuk diperhatikan, warna merupakan deskriptor yang memberikan ciri khas pada suatu citra. Pada citra digital format warna yang paling sering digunakan dan paling dikenal adalah format RGB kepanjangan dari red green blue. Dalam citra dengan format RGB suatu matriks citra tersusun dari setidaknya tiga matriks dimana tiap matriks tersusun merepresentasikan komponen warna red green blue dari citra. Jika ketiga matriks warna red green blue ini digabungkan akan menghasilkan suatu citra yang tampak berwarna-warni di layar. Dengan demikian citra RGB memiliki informasi yang lebih kaya daripada citra grayscale atau citra biner.

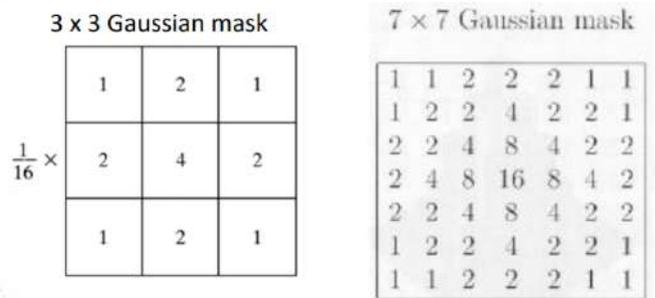
Citra dalam format RGB dapat dengan mudah dikonversikan menjadi citra biner maupun citra grayscale hanya dengan mengubah representasinya saja. Citra grayscale memiliki rentang yang sama dengan citra RGB tetapi hanya memiliki satu representasi matriks saja yang melambangkan derajat keabuan dari suatu pixel. Untuk konversi citra RGB

menjadi citra grayscale dapat dilakukan dengan mencari rata-rata dari nilai red green blue pada citra RGB, tetapi konversi dengan menggunakan perhitungan rata-rata ini tidak memberikan hasil yang bagus. Proses konversi citra RGB ke grayscale saat ini paling banyak menggunakan rumus $0.299R + 0.587G + 0.114B$ dimana R,G dan B masing-masing menandakan nilai pixel pada red green blue. Dapat dilihat dengan rumus tersebut nilai intensitas dari pixel green ditingkatkan dan intensitas dari pixel red dan blue diturunkan karena pengaruhnya yang cukup besar.

Dalam citra RGB, warna hitam penuh diperoleh dari menggabungkan pixel red green blue dengan nilai 0 semua, sementara untuk nilai putih penuh diperoleh dengan menggunakan nilai pixel maksimal pada pixel red green blue. Agar setiap pixel pada citra RGB dapat dikelompokkan menjadi dua kelas saja antara pixel hitam dan putih perlu dilakukan proses threshold yakni dengan menetapkan suatu batas dimana jika suatu pixel pada citra RGB nilainya kurang dari batas threshold maka akan di-set menjadi pixel hitam sementara jika nilainya melebihi threshold yang ditentukan maka nilai pixel akan di-set menjadi pixel putih. Dengan demikian setiap pixel pada citra RGB dapat dijadikan representasi biner hitam dan putih.

C. Gaussian Blur

Gaussian blur merupakan proses pengolahan citra dengan menggunakan proses penapisan atau konvolusi. Proses penapisan pada gaussian blur dilakukan dengan menggunakan filter gauss yang didapat dari penerokan atau sampling pada fungsi gaussian. Nilai pada setiap titik koordinat di dalam matriks filter gaussian semakin ke tengah nilainya semakin besar. Contoh dari matriks filter gaussian dapat dilihat pada gambar 1 berikut.



Gambar 1. Contoh matriks filter gaussian (Rinaldi Munir, Slide mata kuliah Interpretasi dan Pengolahan Citra 2020-2021)

Dengan menggunakan filter gaussian dapat diperoleh gambar citra hasil yang tampak lebih buram dibandingkan citra asli. Tingkat blur yang dihasilkan dari proses gaussian blurring dapat diatur dengan mengatur ukuran matriks filter yang digunakan. Ukuran matriks filter yang besar akan memberikan tingkat blurring yang lebih besar dibandingkan matriks filter yang ukurannya lebih kecil, dengan kata lain citra hasil proses akan lebih buram atau lebih tampak berbeda dibandingkan citra asli.

D. Thresholding

Thresholding atau juga dikenal operasi pengambangan adalah operasi untuk mengurangi batas rentang nilai pixel yang memungkinkan dari suatu citra. Terdapat beberapa jenis proses thresholding dan tiap jenisnya akan menghasilkan citra keluaran yang berbeda juga tergantung kebutuhan. Operasi thresholding paling sederhana adalah thresholding biner yakni mengurangi rentang citra pixel suatu citra sehingga rentangnya hanya menjadi dua nilai saja yakni 1 dan 0, biasanya untuk citra biner keluarannya adalah citra dengan warna hitam dan putih saja.

Contoh jenis lain dari proses thresholding seperti operasi threshold untuk mengurangi nilai maksimal dari pixel pada citra. Proses ini dilakukan dengan menetapkan batas threshold pada suatu nilai, kemudian pixel-pixel yang nilainya lebih dari threshold akan di-set nilainya menjadi nilai threshold sementara pixel lainnya yang nilainya kurang dari threshold nilainya tidak berubah. Begitu juga sebaliknya, thresholding dapat juga digunakan untuk meningkatkan nilai minimum dari pixel pada citra. Proses thresholding seperti ini juga dikenal dengan istilah truncate.

E. Kontur

Kontur didefinisikan sebagai rangkaian pixel-pixel pada tepi yang membentuk suatu batas daerah. Kontur merupakan proses lebih lanjut dari proses pendeteksian tepi yang bertujuan menemukan pixel-pixel tepi, dari pixel-pixel tepi ini akan diproses dan dihubungkan sehingga dapat dijadikan suatu representasi yang melambangkan suatu bentuk sederhana. Kontur pada umumnya digunakan pada computer vision untuk membantu pengenalan objek dan juga deteksi objek.

Metode paling mudah untuk melakukan deteksi objek dengan menggunakan kontur adalah dengan menetapkan suatu batas threshold minimum untuk menentukan apakah kontur yang terdeteksi adalah objek yang diinginkan atau tidak. Dengan menggunakan metode ini, kontur yang ukurannya lebih besar dari batas yang ditentukan merupakan objek yang ingin dideteksi. Sementara itu kontur-kontur lain yang ukurannya lebih kecil dari batas yang ditentukan dianggap sebagai objek lain yang tidak ingin dideteksi. Penentuan batas threshold untuk metode ini memerlukan pertimbangan resolusi dari citra dan juga rasio objek di dalam citra terhadap ukuran keseluruhan citra.

III. IMPLEMENTASI SOLUSI

Sistem deteksi aksi pengambilan barang yang dibangun bertujuan mengklasifikasikan suatu frame image dari video ke dalam kelas frame yang mengandung aksi atau kelas frame yang tidak mengandung aksi. Berikut ditampilkan gambar contoh frame yang tidak mengandung aksi dan yang mengandung aksi masing-masing ditunjukkan pada gambar 2 dan gambar 3.



Gambar 2. Contoh Frame yang tidak mengandung aksi



Gambar 3. Contoh Frame yang mengandung aksi

Sebelum diproses, gambar frame input akan dikonversi menjadi citra grayscale terlebih dahulu. Selain untuk menyederhanakan pemrosesan, pengubahan citra menjadi grayscale juga tidak berpengaruh terhadap deteksi aksi yang akan dilakukan. Hal ini disebabkan pada proses selanjutnya dilakukan pembuangan background image sehingga penggunaan citra RGB maupun grayscale tetap menghasilkan hasil yang sama. Setelah diubah menjadi citra grayscale dilakukan proses gaussian blur terhadap citra, proses ini bertujuan untuk menghilangkan noise pada citra. Proses gaussian blur dilakukan dengan menggunakan fungsi dari library opencv dengan menggunakan kernel filter gaussian berukuran 21x21.

Proses deteksi dilakukan dengan membandingkan antara dua frame yang berurutan untuk mengetahui adanya gerakan di dalam video. Dikarenakan posisi kamera yang diam menyebabkan latar belakang video tidak mengalami perubahan pada frame yang berurutan sehingga hanya pixel-pixel dimana terdapat gerakan yang mengalami perbedaan pada dua frame yang berurutan. Perbedaan pixel pada dua frame inilah yang akan diekstrak untuk dideteksi keberadaan suatu aksi. Berikut ditampilkan hasil proses perhitungan selisih frame dari gambar 3 dengan frame sebelumnya:



Gambar 4. Contoh hasil proses perhitungan selisih frame

Dapat dilihat pada gambar, terdapat gerakan yang jelas terlihat dari hasil pengurangan dua frame berurutan. Hasil ini diperoleh dengan menjalankan fungsi dari library opencv

```
cv2.absdiff(src1, src2)
```

Fungsi diatas menerima input parameter pertama berupa matriks gambar untuk frame pertama dan parameter kedua berupa matriks gambar frame kedua. Hasil keluaran fungsi diatas berupa matriks gambar baru hasil selisih dari frame kedua dikurangi frame pertama.

Hasil dari proses pengurangan kedua frame pada gambar ... masih dapat diperjelas lagi dengan melakukan proses thresholding pada citra. Proses ini dilakukan dengan fungsi:

```
cv2.threshold(src, thresh, maxval, type)
```

Fungsi ... menerima parameter pertama berupa delta frame hasil keluaran fungsi ..., parameter kedua berupa batas threshold, parameter ketiga merupakan batas nilai maksimum dari pixel hasil dari thresholding dan parameter keempat adalah metode thresholding yang dilakukan. Untuk implementasi aplikasi deteksi yang dikembangkan digunakan metode thresholding binary yang akan menghasilkan citra biner untuk keluarannya. Proses thresholding yang dilakukan mengikuti rumus pada gambar berikut.

$$dst(x, y) = \begin{cases} \text{maxval} & \text{if } src(x, y) > \text{thresh} \\ 0 & \text{otherwise} \end{cases}$$

Gambar 5. Rumus thresholding yang digunakan (https://docs.opencv.org/master/d7/d1b/group_imgproc_misc.html)

Bisa dilihat rumus pada gambar ..., nilai pixel akan di-assign menjadi batas nilai maksimal jika nilai pixel input berada diatas nilai threshold yang ditentukan, selain dari itu nilai pixel akan di-assign dengan batas nilai minimum dari pixel yakni 0. Dengan menggunakan rumus ini akan didapatkan image keluaran hasil proses thresholding yang berupa citra biner.

Pada eksperimen yang dilakukan digunakan nilai batas threshold yang cukup kecil yakni 5. Hal ini dilakukan dengan

tujuan untuk memperjelas hasil pengurangan dua frame yang telah dilakukan sebelumnya. Dengan demikian semua nilai pixel yang nilainya lebih dari 5 akan di-assign dengan nilai maksimal yakni 255 sehingga didapat citra hasil keluaran yang lebih mudah dideteksi. Hasil keluaran citra dari gambar 4 setelah dilakukan proses thresholding dapat dilihat pada gambar 6 berikut.



Gambar 6. Contoh hasil proses thresholding

Setelah proses thresholding dilakukan, citra hasil thresholding dapat kita buat lebih tebal lagi agar semakin mudah untuk dideteksi. Hal ini dapat dilakukan dengan menggunakan proses dilasi dengan menjalankan fungsi.

```
cv2.dilate(src, kernel, iterations)
```

Fungsi dilasi pada fungsi diatas, menerima parameter pertama yakni matriks citra yang menjadi input, kemudian parameter kedua berupa kernel untuk proses dilasi dimana dalam eksperimen ini digunakan nilai kernel default dari library opencv yakni matriks kernel berukuran 3x3. Untuk parameter ketiga merupakan banyaknya iterasi proses dilasi dilakukan dimana untuk eksperimen ini dilakukan iterasi sebanyak dua kali agar citra keluaran hasilnya semakin tebal. Fungsi dilasi pada fungsi ..., mengikuti rumus yang ditunjukkan pada gambar berikut.

$$dst(x, y) = \max_{(x', y') : \text{element}(x', y') \neq 0} src(x + x', y + y')$$

Gambar 7. Rumus dilasi yang digunakan (https://docs.opencv.org/3.4/db/df6/tutorial_erosion_dilatation.html)



Gambar 8. Contoh hasil proses dilasi

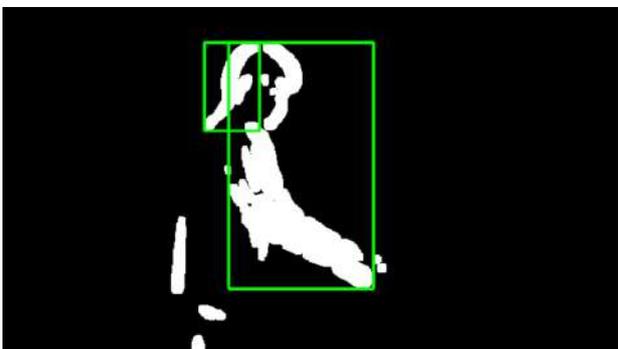
Hasil dari proses dilasi yang dilakukan dapat dilihat pada gambar, dapat dilihat bahwa objek pada citra keluaran yang dihasilkan ukurannya semakin membesar dibandingkan pada citra sebelum dilakukan proses dilasi. Ukuran objek yang besar ini akan memudahkan untuk proses pencarian kontur yang akan dilakukan selanjutnya

Untuk menentukan apakah terdapat aksi pengambilan barang pada frame saat ini dilakukan dengan menghitung luas kontur terbesar pada citra hasil pengurangan frame saat ini dengan frame sebelumnya, selanjutnya jika luas kontur terbesar lebih dari nilai batas threshold minimum yang ditetapkan maka pada frame saat ini dianggap ada aksi pengambilan barang yang dilakukan. Batas threshold minimum ini ditentukan dengan melakukan eksperimen untuk didapatkan nilai threshold yang mampu memberikan hasil terbaik.

Proses pencarian kontur pada citra dilakukan dengan menjalankan fungsi

```
cv2.findContours(image, mode,
                method)
```

Fungsi kontur menerima input pertama berupa array citra masukan, kemudian parameter kedua berupa mode dari algoritma retrieval kontur dimana dalam eksperimen ini digunakan algoritma retrieval kontur terluar dan untuk parameter ketiga merupakan algoritma yang digunakan untuk aproksimasi kontur dimana pada eksperimen ini digunakan metode chain approximation simple. Contoh gambar hasil dari pencarian kontur dapat dilihat pada gambar 9 berikut.



Gambar 9. Contoh hasil proses pencarian kontur

Dari hasil kontur yang didapatkan, dicari kontur dengan luas area yang paling besar, barulah luas kontur terbesar ini dibandingkan dengan batas threshold minimum untuk menentukan apakah pada frame saat ini ada aksi pengambilan barang yang dilakukan atau tidak.

IV. EKSPERIMEN DAN EVALUASI

Proses pengujian untuk mengetahui performa aplikasi deteksi yang dikembangkan dilakukan dengan melakukan pencarian batas threshold terbaik untuk menentukan apakah aksi yang sedang dilakukan pada frame saat ini. Batas threshold merupakan luas area minimum agar kontur yang

terdeteksi dianggap menandakan ada aksi yang terjadi. Untuk eksperimen ini akan diuji batas threshold luas bounding box minimum dari rentang 1000-4000 dengan interval 100 untuk setiap pengujian, pemilihan rentang ini didasari dari ukuran video uji yang memiliki resolusi

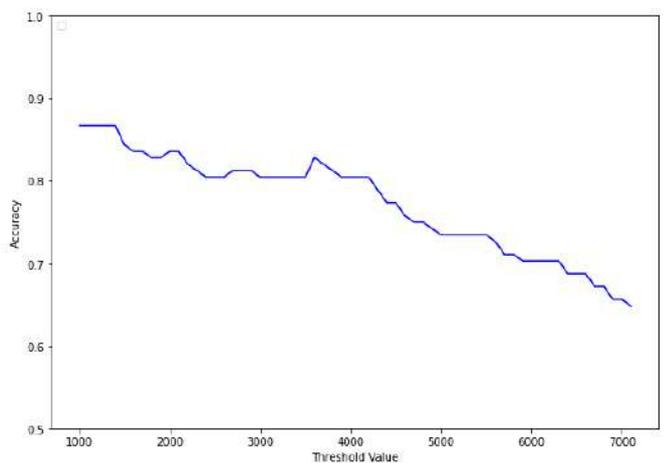
Proses pengujian dilakukan dengan hanya menggunakan satu buah video uji saja yang berdurasi 6 detik dan terdiri dari 128 frame. Video berisikan aksi mengambil barang biskuit dari rak yang disorot dari samping kanan. Resolusi dari video yang diambil mulanya berukuran 1920x1080, kemudian untuk menyederhanakan pemrosesan video uji diturunkan resolusinya menjadi 640x360.

Setiap frame pada video uji akan diberikan label yang menandakan apakah ada aksi atau tidak dalam frame tersebut. Kemudian aplikasi deteksi yang dibangun akan mendeteksi apakah ada aksi yang terjadi pada frame saat ini dengan membandingkan frame saat ini dengan frame sebelumnya, pengecualian terdapat pada frame pertama yang akan dibandingkan dengan frame itu juga, hal ini bukan suatu masalah dikarenakan pada frame pertama diasumsikan tidak ada aksi yang terjadi. Untuk pembagian label pada frame video yang akan diuji dapat dilihat pada tabel 1 berikut

TABEL 1. LABEL FRAME PADA VIDEO UJI

Frame	Label
1 -37	Bukan Aksi
38 - 106	Aksi
107 - 128	Bukan Aksi

Hasil dari pengujian untuk mendapatkan threshold luas area minimum agar kontur yang terdeteksi dianggap menandakan ada aksi yang terjadi dapat dilihat pada gambar grafik berikut.



Gambar 10. Grafik Batas Threshold Area terhadap Akurasi

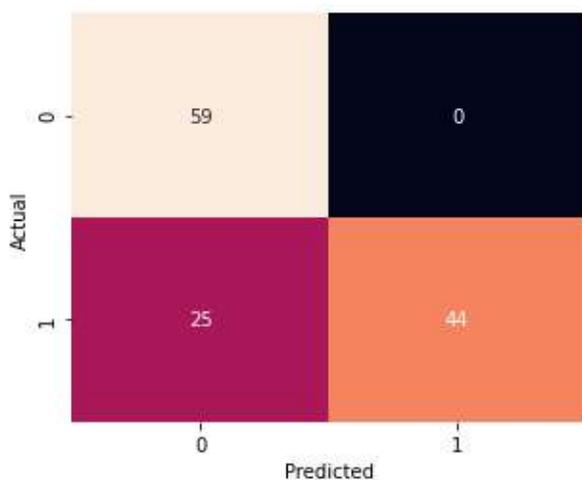
Bisa dilihat dari grafik pada gambar 10, akurasi terbaik diperoleh ketika menggunakan nilai batas threshold antara

1000 hingga 1400 dengan nilai akurasi yang diperoleh sebesar 86%. Nilai threshold minimum dari luas kontur yang semakin besar akan berdampak pada penurunan akurasi yang terus memburuk. Walaupun demikian penurunan akurasi yang terjadi hingga batas threshold 4000 tidak terlalu besar nilai penurunannya sehingga penggunaan nilai threshold hingga batas ini masih bisa digunakan.

Penurunan akurasi untuk setiap nilai threshold yang semakin besar disebabkan faktor batas luas threshold minimum yang bernilai kecil menyebabkan sedikit gerakan saja dapat dianggap sebagai aksi yang memenuhi. Gerakan yang cukup lambat menyebabkan frame tidak banyak berubah di dalam video sehingga luas kontur yang dihasilkan dari proses deteksi akan kecil pula. Di dalam video terdapat transisi frame yang walaupun di dalamnya terdapat aksi tetapi karena terlalu lambat perubahannya sehingga kontur yang dihasilkan akan kecil, permasalahan inilah yang cukup sulit untuk ditangani oleh batas threshold luas minimum yang bernilai besar sehingga semakin besar thresholdnya akan semakin menurun akurasinya.

Tetapi nilai batas threshold yang kecil juga memiliki kekurangan yakni terlalu sensitif terhadap perubahan aksi. Walaupun dengan nilai threshold yang kecil mampu mendeteksi aksi dengan lebih baik tetapi membuat aplikasi deteksi semakin sensitif. Dengan kata lain jika terdapat perubahan frame selain dari aksi pengambilan barang maka noise ini akan mudah dianggap sebagai aksi walaupun bukan hal itu yang sebenarnya ingin dideteksi.

Penggunaan nilai threshold sebesar 4000 membuat aplikasi deteksi pengambilan barang yang dibangun lebih robust terhadap noise. Tetapi terdapat tradeoff penurunan akurasi jika dibandingkan dengan nilai akurasi terbaik yang diperoleh dengan threshold sebesar 1000. Untuk batas threshold luas minimum sebesar 4000 diperoleh nilai akurasi sebesar 80%, nilai akurasi ini memiliki perbedaan yang cukup besar dengan selisih 6% dibandingkan akurasi jika menggunakan threshold 1000 sehingga dapat menjadi pertimbangan.



Gambar 11. Confusion matrix pada batas threshold 4000

Gambar 11 menunjukkan hasil confusion matrix dari hasil pengujian dengan batas threshold area minimum sebesar 4000. Kelas 1 melambangkan frame yang mengandung aksi sementara kelas 0 melambangkan frame yang tidak mengandung aksi di dalamnya. Dapat dilihat aplikasi deteksi salah memprediksi sebanyak 25 frame dari 128 frame dan semua kesalahan yang terjadi adalah aplikasi deteksi salah memprediksi 25 frame gambar yang mengandung aksi sebagai frame yang tidak mengandung aksi di dalamnya.



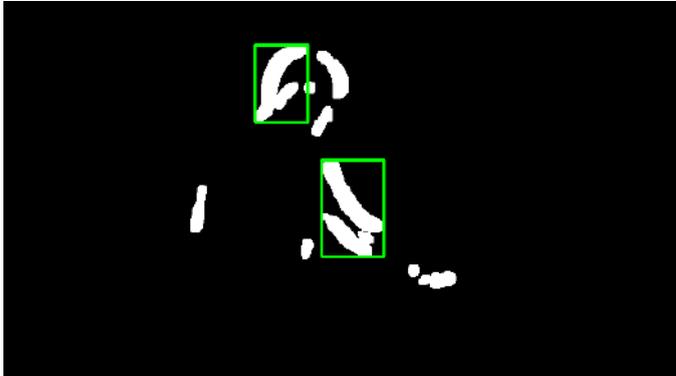
Gambar 12. Contoh frame yang salah diprediksi

Gambar 12 diatas menunjukkan salah satu contoh frame yang mengandung aksi tetapi salah diprediksi oleh aplikasi deteksi sebagai frame yang tidak mengandung aksi di dalamnya. Dapat dilihat pada gambar 12 tersebut aplikasi deteksi yang dikembangkan tidak mendeteksi adanya kontur pada selisih frame gambar. Gerakan yang cukup lambat pada bagian sekuens frame tersebut di video menyebabkan area kontur dari selisih frame yang dihasilkan kecil dan juga tidak banyak. Sekuens frame memiliki gerakan pengambilan barang yang lambat di dalamnya dibuktikan dengan kesalahan dari aplikasi mendeteksi frame salah yang posisinya berurutan yakni frame ke 57 hingga frame ke 81.

Faktor area kontur yang jaraknya cukup berjauhan juga berpengaruh terhadap gagalnya aplikasi menemukan kontur di dalam gambar. Kebanyakan kontur yang dihasilkan dari frame yang mengandung aksi memiliki kontur yang tidak menyambung satu dengan lain tetapi ada jarak sedikit diantaranya. Walaupun terpisah tetapi fungsi untuk mengambil

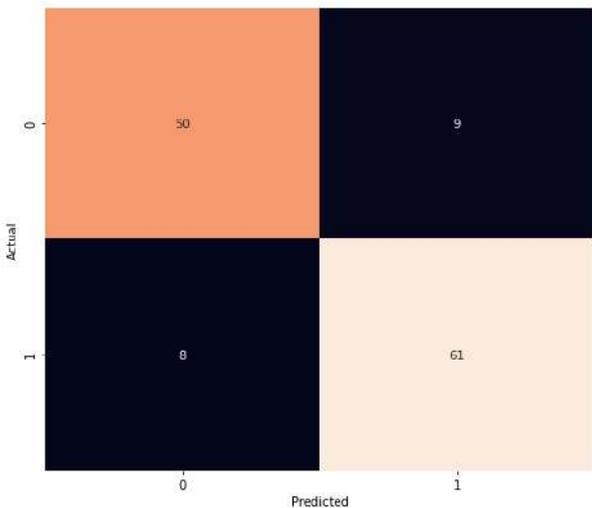
kontur dari gambar mampu untuk menangkap banyak kontur yang terpisah ini sebagai satu kontur yang berukuran besar.

Hasil kontur pada gambar 12 jika digunakan batas threshold luas minimum sebesar 1000 akan dideteksi sebagai frame yang mengandung aksi. Hasil deteksinya dapat dilihat pada gambar 13 berikut.



Gambar 13. Hasil deteksi kontur pada batas threshold 1000

Dapat dilihat pada gambar 13 diatas, dengan menggunakan threshold luas minimum 1000 terdapat area kontur yang memenuhi threshold tersebut. Walaupun kontur yang terdeteksi hanya merupakan sebagian kecil dari gerakan lengan tetapi itu sudah cukup untuk menyimpulkan bahwa pada frame ini terdapat aksi pengambilan barang yang terjadi. Selain itu terdapat area kontur juga yang terdeteksi pada bagian kepala, ini merupakan salah satu contoh noise yang ingin dihindari pada aplikasi deteksi pengambilan barang yang dibangun ini, pemilihan batas threshold luas minimum yang kecil menyebabkan kontur noise seperti ini menjadi kontur yang dianggap mewakili aksi pengambilan barang.



Gambar 14. Confusion matrix pada batas threshold 1000

Dapat dilihat pada gambar 14 menampilkan confusion matrix jika digunakan batas threshold sebesar 1000. Pemilihan

nilai threshold sebesar 1000 ini menyebabkan kesalahan prediksi pada frame yang seharusnya mengandung aksi menjadi menurun cukup banyak tetapi meningkatkan kesalahan prediksi pada frame yang tidak mengandung aksi. Hal ini disebabkan oleh luas area kontur yang merupakan noise memenuhi batas threshold pada gambar frame yang tidak mengandung aksi, sementara peningkatan akurasi pada frame yang mengandung aksi disebabkan oleh kontur yang mewakili aksi pengambilan barang yang sebelumnya tidak memenuhi threshold pada nilai 4000 menjadi dianggap sebagai aksi setelah nilai thresholdnya diturunkan.



Gambar 15. Contoh frame yang salah diprediksi karena mengandung aksi noise

Dapat dilihat pada gambar 15 diatas, kesalahan prediksi pada frame yang tidak mengandung aksi terdapat area kontur yang memenuhi batas threshold pada area ujung tangan. Walaupun pada bagian frame tersebut pada video bukan merupakan aksi pengambilan frame tetapi hanya sedikit gerakan pada ujung tangan tetapi sudah dianggap sebagai kontur yang mewakili aksi pengambilan barang.

Dengan demikian untuk aplikasi deteksi pengambilan barang ini akan lebih baik digunakan batas threshold luas minimum sebesar 4000 dengan pertimbangan yang sudah dijelaskan sebelumnya yakni agar aplikasi hanya mendeteksi kontur yang dianggap aksi pengambilan barang saja dan menghindari noise-noise kecil lain pada video. Walaupun memiliki perbedaan akurasi yang cukup besar jika dibandingkan dengan menggunakan threshold sebesar 1000 tetapi aplikasi deteksi dengan threshold sebesar 4000 memiliki

keunggulan yakni jika suatu frame diprediksi mengandung aksi maka pada frame tersebut benar-benar mengandung aksi pengambilan barang dengan confidence yang tinggi dikarenakan standard thresholdnya yang tinggi juga.

V. KESIMPULAN DAN SARAN

Aplikasi deteksi pengambilan barang dari rak yang sudah dibangun mampu mendapatkan akurasi hingga 80% pada data uji yang digunakan. Hasil ini diperoleh dengan menggunakan batas threshold area minimum sebesar 4000 pada video yang telah di resize menjadi resolusi 640x480. Pemilihan batas threshold yang semakin besar akan meningkatkan ketahanan aplikasi deteksi terhadap noise-noise kecil tetapi juga menyebabkan aplikasi rawan untuk menghasilkan kesalahan mendeteksi apabila kontur frame dari aksi terlalu kecil untuk dideteksi.

Saran pengembangan untuk kedepannya adalah mencoba untuk menambahkan pemrosesan tambahan sebelum dilakukan pencarian kontur agar hasil deteksinya bisa lebih baik atau mencoba metode lain selain pencarian kontur yang mampu mengidentifikasi aksi pengambilan barang dari rak.

ACKNOWLEDGMENT

Penulis mengucapkan terima kasih kepada Allah SWT karena atas berkat rahmat-Nya makalah ini dapat diselesaikan dengan baik. Penulis juga mengucapkan terima kasih kepada Bapak Dr. Rinaldi Munir, M.T. Sebagai dosen mata Kuliah Pengolahan dan Interpretasi Citra 2020-2021 yang telah memberikan ilmu dan juga penjelasan materi dengan baik. Tidak lupa juga penulis mengucapkan terima kasih kepada orang tua dan juga keluarga atas bantuan baik berupa materi maupun motivasi yang mendukung penulis untuk menyelesaikan penulisan makalah ini.

REFERENSI

- [1] Rinaldi Munir, Slide mata kuliah Interpretasi dan Pengolahan Citra 2020-2021
- [2] Soentanto, P., Hendryli, Janson., & Herwindiati, D. (2019). Object and Human Action Recognition From Video Using Deep Learning Models. 10.1109/ICSIGSYS.2019.8811081.
- [3] Dokumentasi OpenCV, <https://docs.opencv.org/master/index.html> diakses 8 Mei 2021 pada 11.00 WIB
- [4] <https://www.pocket-lint.com/gadgets/news/amazon/139650-what-is-amazon-go-where-is-it-and-how-does-it-work> diakses 8 Mei 2021 pada 17.00 WIB

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 25 Mei 2021



Irfan Haris Widyadhana 13517041