

Deteksi Dan Klasifikasi Cacat Pada Produk Kain Dengan Metode Pengolahan Citra Digital

*Arini Trianingsih, Nadhia Rizqy Amalia, Amelia Maslaqun Naila, Fetty Tri Anggraeny

Universitas Pembangunan Nasional "Veteran" Jawa Timur, Indonesia

Artikel Histori:

Disubmit: Juli 2024

Diterima: September 2024

Diterbitkan: Desember 2024

DOI

[10.33005/jifti.v6i2.164](https://doi.org/10.33005/jifti.v6i2.164)



ABSTRAK

Penelitian ini membahas tentang metode yang dapat digunakan untuk mendeteksi dan mengklasifikasi cacat pada kain menggunakan berbasis pengolahan citra digital. Tujuan dari penelitian ini adalah meningkatkan akurasi dan efisiensi deteksi cacat kain melalui pemrosesan citra yang lebih cepat dibanding dengan pengamatan manual. Dalam penelitian ini, gambar kerusakan kain diunggah dan program akan melakukan deteksi menggunakan pemrosesan gambar. Langkah pertama mengubah gambar menjadi grayscale. Selanjutnya, dilakukan penghilangan noise atau filtering, konversi biner melalui thresholding, serta mengaplikasikan operasi morfologi untuk perbaikan hasil citra. Kontur pada gambar kemudian dianalisis untuk mengidentifikasi area yang menunjukkan kerusakan pada kain, seperti kerusakan pada weft satu warp kain. Terdapat 7 jenis klasifikasi cacat yaitu hole, float, thick or thin weft, double pick, insufficient weft density, excessive weft density, dan broken weft. Klasifikasi ini dihitung berdasarkan ukuran kontur dan pengukuran tambahan lebar dan tinggi bounding box. Hasil penelitian ini menunjukkan tingkat keberhasilan deteksi cacat yang tinggi dengan klasifikasi yang akurat untuk setiap jenis kerusakan pada kain sehingga bermanfaat untuk inspeksi otomatis di industri kecil.

Kata Kunci: Deteksi Cacat Kain, Pengolahan Citra Digital, Klasifikasi Kerusakan, Kontur Citra, Operasi Morfologi.

PENDAHULUAN

Pakaian adalah salah satu kebutuhan dasar manusia, yang umumnya dibuat dari kain melalui berbagai teknik seperti tenun, atau rajut (Kumar dkk, 2020). Dalam industri tekstil, proses pembuatan kain melibatkan konversi serat menjadi benang dan kemudian menjadi kain. Salah satu metode yang sering digunakan adalah teknik tenun yang mencakup pola-pola dasar seperti *plain weave*, *satin weave*, dan *twill weave*. *Plain weave* merupakan pola paling sederhana, dengan pola silang antara benang lungsi dan pakan.

How to Cite:

Trianingsih, A., Amalia, N. R., Naila, A. M., & Anggraeny, F. T. (2024). Deteksi Dan Klasifikasi Cacat Pada Produk Kain Dengan Metode Pengolahan Citra Digital. *Jurnal Ilmiah Teknologi Informasi dan Robotika*, 6(2), 107-117. <https://doi.org/10.33005/jifti.v6i2.164>.

***Corresponding Author:**

Email : 21081010030@student.upnjatim.ac.id

Alamat : Jl. Rungkut Madya, Gn. Anyar, Kec. Gn. anyar,
Surabaya, Jawa Timur, 60294



This article is published under a [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/).

Kualitas kain menjadi perhatian utama dalam industri tekstil, karena cacat pada kain dapat mempengaruhi nilai jualnya, bahkan mengakibatkan penolakan oleh konsumen (Zhang, 2021). Berbagai jenis cacat dapat terjadi selama proses produksi, seperti lubang, float, cacat benang tebal atau tipis, double pick, serta kepadatan benang yang tidak sesuai. Oleh karena itu, deteksi dan klasifikasi cacat menjadi langkah penting dalam pengendalian kualitas.

Pengolahan citra memainkan peran penting dalam meningkatkan kualitas data visual untuk mendukung analisis berbasis pembelajaran mesin. Sebagaimana dijelaskan dalam penelitian sebelumnya, kualitas data gambar yang buruk, seperti adanya noise atau detail yang kurang jelas, dapat mempengaruhi akurasi model pembelajaran mesin, termasuk *Generative Adversarial Networks* (GANs), yang sangat bergantung pada pola-pola data yang relevan (Attaqwa dkk, 2024). Oleh karena itu, metode pengolahan citra seperti filtering, thresholding, atau peningkatan kualitas gambar dapat diterapkan untuk mendukung deteksi cacat kain secara otomatis.

Untuk mengatasi masalah tersebut, pengendalian kualitas kain menjadi salah satu prioritas utama dalam proses produksi tekstil. Secara tradisional, deteksi cacat dilakukan secara manual. Namun, metode ini memiliki keterbatasan, seperti ketergantungan pada keahlian, membutuhkan waktu yang cukup lama, dan kemungkinan terjadinya kesalahan manusia (Patel, 2022). Dengan kemajuan teknologi, berbagai pendekatan berbasis otomatisasi telah diperkenalkan untuk meningkatkan akurasi dan efisiensi deteksi cacat kain. Pendekatan ini memanfaatkan teknologi pengolahan citra digital, kecerdasan buatan, dan pembelajaran mesin untuk mendeteksi dan mengklasifikasikan cacat secara otomatis dan mudah.

Dalam penelitian ini, telah diterapkan metode deteksi cacat kain menggunakan pendekatan pengolahan citra, yang mencakup filtering untuk mengurangi noise, thresholding untuk konversi citra menjadi biner, serta operasi morfologi untuk memperbaiki struktur gambar (Chen dan Liu, 2023). Selanjutnya, deteksi kontur digunakan untuk mengidentifikasi dan mengklasifikasikan jenis cacat berdasarkan bentuk dan ukuran kontur yang ditemukan. Metode ini terbukti efektif dalam mendeteksi berbagai jenis kerusakan pada kain (Sharma dan Kumar, 2021).

METODE PENELITIAN

Diagram alur yang menjelaskan proses pendeteksian dan klasifikasi cacat pada kain menggunakan metode pemrosesan citra digital ditampilkan pada gambar 1. Proses dimulai dengan pengumpulan citra kain, yaitu pengambilan gambar kain menggunakan perangkat seperti kamera atau pemindai resolusi tinggi untuk memastikan detail kain terekam dengan baik. Setelah itu, citra dikonversi menjadi skala abu-abu agar pemrosesan lebih sederhana dan fokus pada intensitas piksel. Tahapan selanjutnya dilakukan penghilangan noise menggunakan filter median untuk membersihkan gangguan pada citra.

Setelah noise dihilangkan, dilakukan thresholding atau deteksi tepi untuk memisahkan bagian penting seperti area cacat dipisahkan dari latar belakang. Setelah segmentasi, penelitian melibatkan operasi morfologi untuk membersihkan noise pada area latar depan. Algoritma pencarian *Depth-First Search* (DFS) digunakan untuk menemukan dan mengumpulkan piksel-piksel yang saling terhubung dan membentuk sebuah batas atau kontur dari area cacat. Dari kontur yang ditemukan ini, kemudian dianalisis lebih lanjut

untuk mendapatkan informasi mengenai ukuran dan posisi cacat pada kain, yang kemudian digunakan untuk mengidentifikasi pola dan mendeteksi cacat berdasarkan penyimpangan dari pola kain yang diharapkan, jenis cacat tersebut seperti lubang(*hole*), benang terapung(*float*), kepadatan benang yang kurang atau berlebih (*double pick*), dan lainnya.

Hasil klasifikasi ini kemudian digunakan dalam tahap implementasi sistem, di mana sistem pendeteksian dan klasifikasi dapat dijalankan secara real-time. Metode ini memanfaatkan pola kain referensi untuk mengenali cacat pada pola yang serupa, dan hasilnya menunjukkan bahwa cara yang digunakan ini dapat membantu meningkatkan efisiensi dalam memeriksa kualitas kain. Diagram Alur Proses ditunjukkan pada Gambar 1.

Filtering

Pada langkah *filtering* dalam deteksi cacat kain tenun, tujuannya adalah untuk mengurangi noise atau gangguan dalam gambar sebelum melanjutkan ke tahap *thresholding* dan morfologi. Pada tahap ini, setiap piksel dalam gambar akan diganti dengan nilai rata-rata dari piksel-piksel di sekitarnya (dalam area jendela yang disebut *neighborhood* atau *kernel*). Proses ini dilakukan untuk setiap piksel gambar, kecuali pada piksel yang terletak di tepi (karena mereka tidak memiliki tetangga di semua sisi).

Rumus Mean filtering:

$$I'(x,y) = \frac{1}{N} \sum_{i=-k}^k \sum_{j=-k}^k I(x+i,y+j)$$

Dimana, $I'(x,y)$ merupakan nilai piksel baru pada posisi (x,y) setelah penerapan filter, lalu $I(x+i,y+j)$ merupakan nilai piksel disekitar piksel $I'(x,y)$ di dalam jendela filter. Selanjutnya N adalah jumlah total elemen dalam kernel, yaitu $N = (2k + 1)^2$, dengan k adalah setengah ukuran kernel (misalnya untuk kernel 3×3 , $k=1$), i dan j mengindeks piksel-piksel disekitar piksel pusat (x,y) dalam kernel, lalu $i=-k$ $j=-k$ $I(x+i,y+j)$ merupakan jumlah nilai-nilai piksel dalam area jendela atau kernel (x,y) , jendela ini biasanya berupa persegi dengan ukuran $(2k+1) \times (2k+1)$, dan $1/N$ adalah faktor normalisasi yang memastikan hasilnya adalah rata-rata dari nilai-nilai piksel yang ada dalam jendela tersebut. N adalah jumlah total piksel dalam kernel, yang dapat dihitung sebagai $N = (2k+1)^2$, dimana K merupakan jarak dari piksel pusat ke tepi kernel.

Thresholding

Pada langkah *thresholding*, proses ini bertujuan untuk mengubah gambar grayscale (abu-abu) menjadi gambar biner (hitam-putih), yang memudahkan dalam deteksi objek atau cacat pada kain tenun. Cara Kerja *Thresholding* dimulai dari penentuan nilai *threshold*, di mana setiap piksel dalam gambar grayscale memiliki nilai intensitas antara 0 (hitam) dan 255 (putih). Nilai *threshold* adalah angka yang digunakan untuk membedakan piksel yang akan dianggap objek (putih) dan latar belakang (hitam). Jika nilai intensitas piksel lebih besar atau sama dengan nilai *threshold*, maka piksel tersebut akan menjadi putih (255). Jika kurang dari nilai *threshold*, maka piksel tersebut akan menjadi hitam (0).

Untuk setiap piksel dalam gambar, rumus *thresholding* yang umum digunakan adalah:

$$Binary(x,y) = \begin{cases} 255 & \text{jika } I(x,y) \geq T \\ 0 & \text{jika } I(x,y) < T \end{cases}$$

Dimana, $I(x,y)$ adalah nilai intensitas piksel pada posisi (x,y) dalam gambar grayscale, T adalah nilai *threshold* yang ditentukan, dan $Binary(x,y)$ adalah nilai piksel pada posisi (x,y) dalam gambar biner setelah thresholding. Piksel ini akan menjadi 255 (putih) jika nilai intensitasnya lebih besar dari atau sama dengan *threshold*, dan 0 (hitam) jika kurang.

Operasi Morfologi

Dalam aplikasi deteksi cacat kain tenun, operasi morfologi dapat digunakan untuk mengurangi noise dan memperbaiki fitur penting yang mungkin terfragmentasi selama langkah thresholding. Contohnya, setelah proses thresholding, dapat menerapkan operasi opening untuk menghapus noise atau objek kecil yang tidak relevan, atau menggunakan operasi closing untuk mengisi celah atau lubang pada objek (misalnya pada kontur kerusakan kain). Langkah-langkah tersebut membantu dalam memperbaiki citra biner sebelum melakukan langkah selanjutnya, seperti pencarian kontur atau analisis lebih lanjut mengenai cacat yang ada pada kain tenun. Dengan menggunakan kedua operasi tersebut secara tepat, dapat menghasilkan citra yang lebih jelas dan terdefinisi, sehingga lebih mudah untuk melakukan deteksi cacat dan analisis lebih lanjut.

Pendeteksian Kontur

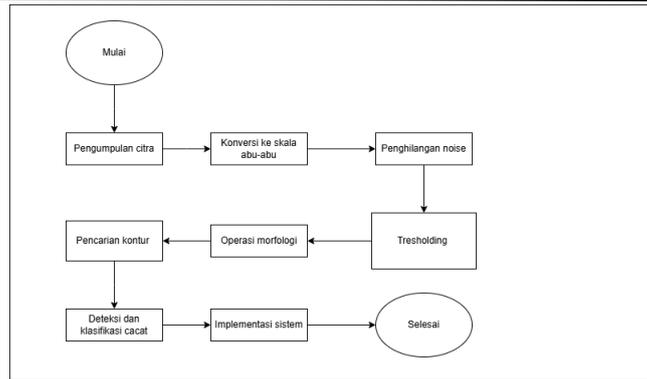
Proses pendeteksian kontur dilakukan melalui serangkaian tahapan pengolahan citra digital untuk mengidentifikasi dan menganalisis kerusakan pada tekstil. Pendeteksian kontur dilakukan menggunakan algoritma penelusuran tetangga (neighbor tracing) dengan pendekatan Depth-First Search (DFS). Algoritma ini memiliki karakteristik berikut:

- a) Memulai dari piksel putih yang belum dikunjungi
- b) Menelusuri piksel-piksel putih yang terhubung secara horizontal dan vertikal
- c) Mencatat koordinat setiap piksel dalam kontur
- d) Menandai piksel-piksel yang telah dikunjungi untuk menghindari pengulangan

Klasifikasi

Proses klasifikasi kerusakan tekstil dalam penelitian ini dilakukan melalui serangkaian tahapan pengolahan citra digital dengan menggunakan metode berbasis piksel dan kontur. Metode klasifikasi dirancang untuk mendeteksi dan mengidentifikasi berbagai jenis kerusakan pada tekstil secara otomatis. Klasifikasi kerusakan dilakukan berdasarkan karakteristik geometris kontur yang terdeteksi, dengan mempertimbangkan parameter:

- a) Panjang kontur
- b) Lebar kontur
- c) Tinggi kontur



Gambar 1. Alur proses penelitian
Sumber: Data Diolah

Berdasarkan parameter tersebut, penelitian mengklasifikasikan kerusakan ke dalam tujuh kategori utama:

Lubang (*Hole*)

- Kriteria: Panjang kontur > 1000 piksel.
- Indikasi: Kerusakan berbentuk lubang besar pada tekstil.

Benang Melayang (*Float*)

- Kriteria: Panjang kontur > 700 piksel, Lebar > 100 piksel, dan Tinggi < 50 piksel.
- Indikasi: Benang yang tidak terikat atau terlepas dari struktur dasar tekstil.

Cacat Benang Pakan Tebal atau Tipis (*Thick or Thin Weft*)

- Kriteria: Panjang kontur > 500 piksel, Lebar antara 50-100 piksel, dan tinggi < 50 piksel.
- Indikasi: Variasi ketebalan benang pakan yang tidak seragam.

Double Pick

- Kriteria: Panjang kontur > 300 piksel, Lebar < 50 piksel, dan Tinggi > 30 piksel.
- Indikasi: Benang pakan ganda atau tumpang tindih.

Kepadatan Benang Pakan Kurang (*Insufficient Weft Density*)

- Kriteria: Panjang kontur > 200 piksel, Lebar < 50 piksel, dan Tinggi < 30 piksel.
- Indikasi: Jarak antar benang pakan yang renggang.

Kepadatan Benang Pakan Berlebih (*Excessive Weft Density*)

- Kriteria: Panjang kontur > 100 piksel, Lebar > 20 piksel, dan Tinggi > 30 piksel.
- Indikasi: Benang pakan yang terlalu padat atau rapat.

Benang Pakan Putus

- Kriteria: Panjang kontur \leq 100 piksel, Lebar \leq 20 piksel, dan Tinggi \leq 20 piksel.
- Indikasi: Putusnya benang pakan pada tekstil.

HASIL DAN PEMBAHASAN

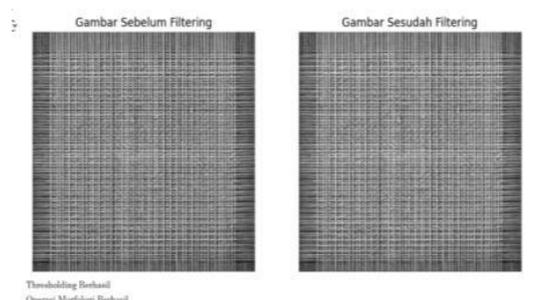
Pillow(PIL), pustaka *numpy*, pustaka *matplotlib*, *google colab*, dan bahasa pemrograman *python* digunakan untuk menguji deteksi dan klasifikasi cacat pada kain tenun berdasarkan pemrosesan digital. Semua gambar kain yang didapatkan diambil dari makalah “Simulasi dan pengenalan kain umum” oleh *Maroš Tunák dan Aleš Linka* dari Universitas Teknik di Liberec, Departemen Bahan Tekstil; Hálkova 6, Liberec 46117, Republik Ceko.

Membaca Data Gambar

Pengguna akan memasukkan file gambar yang nantinya file gambar tersebut akan disimpan di dalam sesi dan dapat diakses melalui path file. Setelah pengguna memasukkan gambar, langkah pertama yang dilakukan adalah membaca file gambar tersebut. Gambar yang diunggah akan dibuka dan dikonversi ke format grayscale (hitam-putih), di mana setiap piksel gambar diwakili oleh nilai intensitas antara 0 hingga 255, dengan 0 berarti hitam dan 255 berarti putih. Proses ini memungkinkan gambar menjadi lebih mudah diproses untuk analisis selanjutnya, seperti deteksi kontur atau pemrosesan citra lainnya. Setelah gambar dibaca, informasi mengenai dimensi gambar (seperti lebar dan tinggi) juga dapat diambil, yang kemudian dapat digunakan dalam proses-proses pengolahan gambar lebih lanjut. Gambar yang telah dibaca ini kemudian bisa ditampilkan untuk diperiksa atau dimanipulasi lebih lanjut sesuai kebutuhan.

Filtering atau Penghilangan Noise

Setelah pengguna mengunggah gambar, langkah berikutnya adalah melakukan pemrosesan untuk menghilangkan *noise* atau gangguan pada gambar. Proses ini bertujuan untuk memperbaiki kualitas gambar agar lebih mudah dianalisis. *Noise* pada gambar sering kali berupa titik atau piksel yang tidak diinginkan yang dapat mengganggu deteksi objek atau kontur. Oleh karena itu, gambar yang telah dibaca akan diproses untuk mengurangi pengaruh *noise*. Selama tahap ini, gambar akan diperlakukan dengan cara tertentu, seperti menghaluskan atau meratakan nilai piksel di sekitar setiap titik untuk memastikan hanya informasi yang relevan yang dipertahankan. Sehingga membuat gambar menjadi lebih bersih dan tajam seperti yang ditampilkan pada Gambar 2, sehingga fitur-fitur penting seperti kontur dan bentuk objek menjadi lebih jelas untuk dianalisis lebih lanjut. Proses ini sangat penting agar hasil dari tahapan berikutnya, seperti *thresholding* dan pencarian kontur, lebih akurat dan efisien.



Gambar 2. *Filtering*
Sumber: Data Diolah

Thresholding

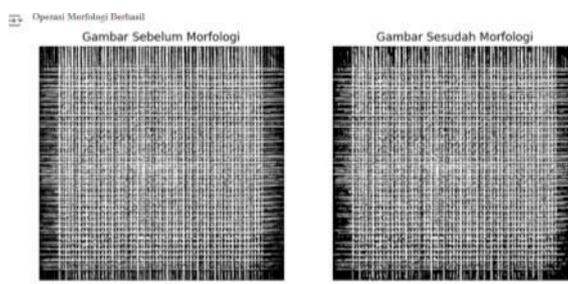
Pada tahap *thresholding*, gambar yang telah difilter sebelumnya dikonversi menjadi gambar biner, yaitu gambar yang hanya terdiri dari dua nilai piksel: hitam dan putih. Proses ini dilakukan dengan menentukan nilai ambang (*threshold*) tertentu. Setiap piksel dengan nilai intensitas di atas ambang tersebut akan diubah menjadi putih, sedangkan piksel dengan nilai di bawah ambang akan diubah menjadi hitam. *Thresholding* membantu memisahkan objek dari latar belakang dalam gambar, sehingga objek yang lebih terang akan terlihat jelas. Misalnya, pada gambar yang mengandung cacat atau kontur, *thresholding* digunakan untuk menyorot kontur tersebut dengan membuat bagian objek menjadi putih dan bagian lainnya menjadi hitam. Hal ini memudahkan dalam langkah-langkah berikutnya seperti analisis kontur atau pengolahan morfologi.

Operasi Morfologi

Pada langkah operasi *morfologi* dalam deteksi cacat kain, tujuannya adalah untuk memperbaiki atau memodifikasi struktur gambar biner (hitam-putih) yang diperoleh setelah langkah *thresholding*. Operasi *morfologi* digunakan untuk mengurangi *noise* dan memperjelas fitur penting seperti kontur, serta membantu dalam memisahkan atau menggabungkan objek dalam gambar. Secara keseluruhan, operasi morfologi dapat membantu dalam menghilangkan elemen yang tidak diinginkan dan memperbaiki struktur objek utama, seperti cacat yang ada pada kain, sehingga memudahkan untuk melakukan deteksi lebih lanjut, seperti pencarian kontur atau klasifikasi cacat seperti pada Gambar 3. Dengan kata lain, operasi ini berfungsi untuk memperbaiki kualitas gambar biner agar lebih mudah mendeteksi cacat yang signifikan dan mengabaikan *noise* atau gangguan lainnya.

Mencari Kontur

Pada proses mencari kontur ini berfungsi untuk mencari kontur dalam gambar biner dengan nilai piksel 255 (putih) dan 0 (hitam). Fungsi *cari_kontur* untuk mencari kontur dalam gambar biner dengan menggunakan algoritma pencarian *depth-first search (DFS)*. Setiap piksel yang ditemukan dan belum terdeteksi ditambahkan ke dalam *stack* dan akan ditandai telah terdeteksi. Proses ini berlanjut hingga seluruh piksel dalam satu kontur ditemukan yang kemudian disimpan dalam *list konturs*. Fungsi ini mengembalikan semua kontur yang terdeteksi dalam gambar.



Gambar 3. Operasi *morfologi*
Sumber: Data Diolah

Jumlah kontur yang ditemukan: 901

Gambar 4. Hasil kontur
Sumber: Data Diolah

```
Hasil Klasifikasi Cacat:  
Kontur 1: Lubang (Hole)  
Kontur 2: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 3: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 4: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 5: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 6: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 7: Double Pick  
Kontur 8: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 9: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 10: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 11: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 12: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 13: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 14: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 15: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 16: Cacat benang pakan tebal, atau tipis (Thick or Thin Weft)  
Kontur 17: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 18: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 19: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 20: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 21: Double Pick  
Kontur 22: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 23: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 24: Kepadatan benang pakan yang kurang (Insufficient Weft Density)
```

Gambar 5. Hasil klasifikasi cacat
Sumber: Data Diolah

```
Kontur 25: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 26: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 27: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 28: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 29: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 30: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 31: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 32: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 33: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 34: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 35: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 36: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 37: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 38: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 39: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 40: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 41: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 42: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 43: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 44: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 45: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 46: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 47: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 48: Kepadatan benang pakan yang kurang (Insufficient Weft Density)  
Kontur 49: Kepadatan benang pakan yang kurang (Insufficient Weft Density)
```

Gambar 6. Hasil klasifikasi cacat
Sumber: Data Diolah

Hasil output ini menunjukkan jumlah kontur (bentuk atau batas) yang berhasil dideteksi dalam sebuah gambar, yaitu 901 kontur. Terlihat hasil Capture sistem pada Gambar 4.

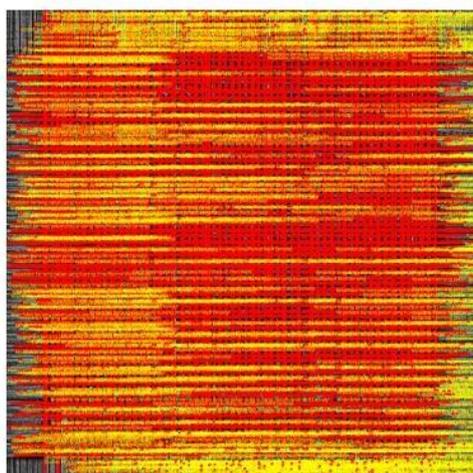
Klasifikasi Bentuk Kontur dan Type

Fungsi klasifikasi kerusakan yang ada pada kode program mengklasifikasikan jenis kerusakan berdasarkan panjang, lebar, dan tinggi kontur, Setiap kontur dianalisis, kemudian dikategorikan sesuai dengan kriteria pada Tabel 1. Fungsi ini mengembalikan daftar jenis kerusakan untuk setiap kontur berdasarkan kondisi tersebut. Setelah proses klasifikasi berdasarkan panjang, lebar, dan tinggi kontur selesai, Gambar 5 dan Gambar 6 adalah hasil klasifikasi jenis kerusakan yang ditemukan pada setiap kontur.

Klasifikasi Weft dan Warp

Fungsi ini digunakan untuk memvisualisasikan kontur - kontur yang terdeteksi pada gambar, serta menandai setiap jenis kerusakan yang terkait. Setiap kontur yang ditemukan akan digambarkan dengan titik - titik kecil pada posisi koordinatnya. Warna titik tersebut disesuaikan berdasarkan jenis kerusakan yang terdeteksi yaitu warna merah untuk kontur

yang menunjukkan kerusakan dan warna kuning untuk kontur yang tidak menunjukkan kerusakan. Selain itu, jenis kerusakan yang terdeteksi ditampilkan pada posisi titik pertama dari setiap kontur dalam bentuk teks berwarna hitam. Dengan cara ini, fungsi ini memberikan representasi visual yang jelas mengenai kondisi gambar berdasarkan klasifikasi kerusakan yang telah dilakukan. Pada gambar 7 adalah hasil dari deteksi dari kontur yang ditandai dengan warna merah (kerusakan), dan warna kuning (tidak ada kerusakan).



Gambar 7. Hasil deteksi kontur
Sumber: Data Diolah

Tabel 1
Kategori bentuk kontur dan tipe

| Kondisi | Jenis kerusakan |
|---|---|
| Panjang > 1000 | Lubang (<i>Hole</i>) |
| Panjang > 700 atau (Lebar > 100 dan Tinggi < 50) | Benang melayang (<i>Float</i>) |
| Panjang > 500 atau (50 < Lebar < 100 dan Tinggi < 50) | Cacat benang pakan tebal atau tipis (<i>Thick or Thin Weft</i>) |
| Panjang > 300 atau (Lebar < 50 dan Tinggi > 30) | <i>Double Pick</i> |
| Panjang > 200 atau (Lebar < 50 dan Tinggi < 30) | Kepadatan benang pakan yang kurang (<i>Insufficient Weft Density</i>) |
| Panjang > 100 atau (Lebar > 20 dan Tinggi > 20) | Kepadatan benang pakan yang berlebih (<i>Excessive Weft Density</i>) |
| Panjang <= 100 dan Lebar <= 20 dan Tinggi <= 20 | Benang pakan putus (<i>Broken Weft</i>) |
| Jika tidak ada kondisi yang cocok | Tidak ada kerusakan |

Sumber: Data Diolah

SIMPULAN

Jurnal ini bertujuan untuk mendeteksi dan mengklasifikasikan cacat pada kain melalui pengolahan citra digital. Dimulai dengan membaca citra dan mengkonversinya menjadi grayscale, program melakukan filtering menggunakan metode rata-rata lingkungan (*mean filter*) untuk mengurangi noise. Selanjutnya, dilakukan proses *thresholding* untuk mengubah citra menjadi gambar biner, memisahkan pola cacat dari latar belakang. Gambar biner ini diproses lebih lanjut menggunakan operasi morfologi, seperti *opening* dan *closing* untuk menyempurnakan struktur area yang relevan. Kontur kemudian diidentifikasi sebagai batas area cacat yang dianalisis berdasarkan panjang, lebar, dan tinggi guna mengklasifikasikan jenis kerusakan, termasuk lubang (*hole*), benang melayang (*float*), cacat benang tebal-tipis (*thick or thin weft*), dan kepadatan benang yang tidak sesuai (*insufficient dan excessive weft destiny*) dan benang pakan putus (*broken weft*). Hasil program ditampilkan melalui gambar asli yang telah dianotasi, di mana kontur cacat ditandai dengan warna dan diberi label jenis kerusakan. Pendekatan ini mempermudah pengguna dalam mendeteksi dan memahami lokasi serta jenis cacat pada kain, memberikan solusi otomatis yang efektif untuk inspeksi kualitas kain secara cepat dan akurat.

DAFTAR PUSTAKA

- Attaqwa, S. I., Puspaningrum, E. Y., & Saputra, W. S. (2024). Implementasi Contrast Limited Adaptive Histogram Equalization Dalam Pengolahan Citra Pada Algoritma Generative Adversarial Network. *JITET (Jurnal Informatika dan Teknik Elektro Terapan)*, 12(3S1), 4317-4322.
- Chen, W., & Liu, Y. (2023). Image Processing Techniques For Precise Fabric Defect Identification. *Computer Vision and Image Understanding*, 217, 103-118.
- Chen, W., & Liu, Y. (2023). Innovative Correlation And Hough Transform Methods In Fabric Defect Classification. *International Journal of Computer Vision and Image Processing*, 33(1), 78-92.
- Kumar, R., & Singh, A. (2020). Advanced Textile Manufacturing Techniques: A Comprehensive Review. *International Journal of Textile Science*, 39(2), 78-95.
- Patel, S., (2022). Automated Fabric Defect Detection: Transition From Manual To AI-Driven Quality Control. *Journal of Intelligent Manufacturing*, 33(4), 1123-1140.
- Patel, S., et al. (2022). Python-Based Image Processing Libraries For Industrial Quality Control: A Comparative Study. *Journal of Industrial Image Analysis*, 27(4), 210-225.

- Rodriguez, M., & Santos, P. (2021). High-Precision Fabric Defect Detection Using Deep Learning And Computer Vision. *Neural Computing and Applications*, 33(6), 2145-2160.
- Sharma, A., & Kumar, R. (2021). Deep Learning Approaches In Textile Defect Detection And Classification. *Neural Computing and Applications*, 33(6), 2045-2060.
- Sharma, A., & Kumar, R. (2022). Advanced Image Processing Techniques For Fabric Defect Detection: A Comprehensive Review. *Journal of Textile Research*, 45(2), 125-140.
- Zhang, L., et al. (2021). Machine Learning Approaches For Textile Quality Inspection: Challenges And Opportunities. *IEEE Transactions on Automation Science and Engineering*, 18(3), 1045-1059